



Twenty-One Day Online Training Manual on



Advanced Statistical and Machine Learning Techniques for Data Analysis Using Open-Source Software for Abiotic Stress Management in Agriculture

16 July – 5 August 2025 

Volume

02

Training Manual

Design of Experiments and Biometrical Analysis

Edited by

Dr. Santosha Rathod
Dr. Nobin Chandra Paul
Ms. Ponnaganti Navyasree
Mr. K Ravi Kumar
Dr. Prabhat Kumar

Organised by:

School of Social Science and Policy Support
ICAR-National Institute of Abiotic Stress Management, Baramati, Maharashtra - 413115

Design of Experiments and Biometrical Analysis

Editors

Santosha Rathod

Nobin Chandra Paul

Ponnaganti Navyasree

K Ravi Kumar

Prabhat Kumar

2025



School of Social Science and Policy Support
ICAR-National Institute of Abiotic Stress
Management, Baramati – 413115
Maharashtra, India



Title: Design of Experiments and Biometrical Analysis

Editors: Santosha Rathod, Nobin Chandra Paul, Ponnaganti Navyasree, K Ravi Kumar, Prabhat Kumar

Published by: ICAR-National Institute of Abiotic Stress Management, Malegaon Khurd, Baramati – 413115, Maharashtra, India.

Edition: I

Volume: 2

ISBN: 978-81-985897-3-6

Copyright: ICAR-National Institute of Abiotic Stress Management, Malegaon Khurd, Baramati, Pune – 413115, Maharashtra, India.

Citation:

Rathod, S., Paul, N. C., Ponnaganti, N., Kumar, K. R., & Kumar, P. (Eds.). (2025). *Design of Experiments and Biometrical Analysis: Training manual of the twenty-one-day online training programme on “Advanced statistical and machine learning techniques for data analysis using open-source software for abiotic stress management in agriculture”* (Vol. 2). ICAR-National Institute of Abiotic Stress Management. ISBN 978-81-985897-3-6.

CONTENTS

S No	Title	Page No
MODULE 3: Design Of Experiments and Biometrical Analysis		
1	Analysis of Design of Experiments in R	1-18
2	Split Plot and Strip Plot Data Analysis	19-26
3	Repeated Measures ANOVA	27-39
4	Group of Experiments	40-45
5	Analysis of Incomplete Block Designs	46-61
6	Response Surface Methodology	62-85
7	Generation Mean Analysis	86-99
8	Mating Design (Diallel and Line X Tester Design)	100-117
9	Path Analysis	118-126
10	Genotype-By-Environment Interaction and Stability Analysis	127-144
11	Basic Bioinformatics and QTL Analysis	144-154
12	Transcriptomic Analysis	155-177
13	Genome Wide Association Studies (GWAS)	178-193
14	Genomic Selection and Its Utilization for Crop Breeding	194-206
15	Selection Index in Plant Breeding	207-225
16	Metagenomics: Introduction and Applications	226-232
17	Meta-QTL Analysis and BioMercator 4.2.3 Workflow	233-244

Analysis of Design of Experiments in R

Santosh Patil

ICAR-Indian Agricultural Statistics Research Institute, New Delhi - 110012.

Email: san.santoshpatil@gmail.com

R. A. Fisher introduced the term ‘Analysis of variance’. The basic purpose of the analysis of variance is to test homogeneity of several means

The variation in numerical data may classified as

- Assignable cause- *i.e.* known cause
- Chance cause- *i.e.* error

Variation due to assignable cause can be measured and control and whereas variation due to error is beyond human control.

The increased efficiency and reduced experimental errors in experimental designs are achieved by THREE basic principles *i.e.*, Replication, Randomization, and Local control

Replication: -

Repeated application of treatment under investigation/experiment known as Replication

The Standard error of treatment mean is given as,

$$\text{S. E. } (\bar{x}) = \frac{S_x}{\sqrt{r}}$$

In general, the number of replications is chosen such that degree of freedom for error term is not less than 12. And minimum two replications may be used in experiment

Randomization: -

Allocation of the treatments to experimental units in such a way that an experimental unit has equal chance of receiving any treatment is known as randomization.

Example: -

Block 01	T2	T3	T1	T4
Block 02	T1	T4	T2	T3
Block 03	T4	T2	T3	T1

Role:

- Randomization removes human biases (assigning the treatments randomly).
- It introduces the independence of treatment in an experiment.

Local control: -

- Local control is a tool for maintaining greater homogeneity of experimental units within a block of an experiment. Local control is **also called blocking**.
- Blocking is done perpendicular to direction of heterogeneity.

Role:

- Local control form homogeneous block of experimental unit.
- It reduces the experimental error.
- It makes the design more efficient.
- It is also used to find the size and shape of experimental units.

Agricultural Experiment Designs:

Complete Block Design- are that design in which each block receives all treatment e.g. CRD, RBD, LSD whereas in case Incomplete Block Design each block does not receive all treatment e.g. BIBD, PBIBD

ASSUMPTIONS OF THE ANALYSIS OF VARIANCE

The general interpretation of the analysis of variance is valid only when certain assumptions are fulfilled as follows:

- **Additive Effects:** Treatment and environmental effects are additive in nature. *Tukey's 1-df Test for non-additivity*.
- **Independence of errors:** Experimental errors are independent in nature. A plot of the residuals will help to check independence of error.
- **Normal Distribution -** A plot of the residuals and Shapiro-Wilk test is recommended to check the normality of errors.
- **Homogeneity of Variance-** Experimental errors have common variance. Bartlett's and Levene's test are the most widely used for testing the homogeneity of several variances.

Failure to meet one or more of these assumptions affects the significance of the F test in the analysis of variance.

To know, mean performance of treatment how significantly different from other treatments the following commonly used multiple pair wise mean comparison test

- Least significant difference (lsd)
- Scheffe's
- Tukey's also known as honestly significant difference test" (HSD),
- Duncan's multiple range test
- Dunnett's-mostly used in Treatments Vs control;

-: Complete Randomized Design: -

The simplest design uses two essential principles of replication and randomization. In CRD we allocate 't' treatments completely at random to the n units, provided that i-th treatment appears in r_i units for $i = 1, 2, \dots, t$. $\sum r_i = n$ units,

Layout: Here as the whole field is homogenous in nature, we allotted the treatments randomly over whole field using random number table. We can see that the treatment T1, T2, T3, T4 & T5 occurs 4,3,4,2 & 2 times respectively.

T2	T3	T1	T5	T3
T1	T4	T3	T1	T2
T3	T5	T2	T4	T1

Linear Model of CRD is: $Y_{ij} = \mu + \tau_i + \varepsilon_{ij}$

Y_{ij} : observation from i-th treatment and j-th replicates.

μ : true mean effect

τ_i : i-th treatment effect

ε_{ij} : error/ effect due to unknown factors.

Hypothesis: -

H_0 : $\mu_1 = \mu_2 = \mu_3 \dots \dots \dots = \mu_t$

H_1 : at least one treatment mean μ_i differ from others

ANOVA sketch for completely randomized design with **t** Treatments, **r** replications

S.V.	df	SS	MSS	F cal
Treatment	t-1	SS _{trt}	M _{Strt}	M _{Strt} / MS error
Error	n-t	SS error	MS error	
Total	n-1	SS tot		

Merits:

- Flexibility to have any number of treatments and replicates for each treatment.
- Mostly used in Lab and pot experiments where experimental condition is homogenous.

Demerits: If experiment with more number of treatments then it is difficult to maintain homogenous condition of experimental units.

Randomized Block Design: -

The mostly commonly used design in agriculture experiments which uses all three essential principles of DOE i.e. replication, randomization and local control. In RBD we allocate 't' treatments completely at random to the each block separately to the n units, provided that i-th treatment appears in r units for $i = 1, 2, \dots, t$. $n = tr$ units,

In RBD, **number of blocks = number of replications**

Layout:

Let an experiment consist of 5 treatments, allotted in 3 blocks separately without any repetition in particular block

Block I	T2	T3	T1	T5	T4
Block II	T1	T4	T3	T5	T2
Block III	T3	T5	T2	T4	T1

Here experimental material between blocks is heterogeneous and within blocks homogeneous, allotment of the treatments randomly for each block using random number table. We can find that the treatment five treatments occurred each 3 times.

Linear Model of RBD is: $Y_{ij} = \mu + \tau_i + \beta_j + \varepsilon_{ij}$

Y_{ij} : observation from i-th treatment and j-th replicates.

μ : true mean effect

τ_i : i-th treatment effect

β_j : j-th block/replication effect

ε_{ij} :- error/ effect due to unknown factors.

Analysis in case of RBD Design:

Hypothesis: -

For testing Treatment effect

$$H_0 : \mu_1 = \mu_2 = \mu_3 \dots \dots \dots = \mu_t$$

H_1 : at least one treatment mean μ_i differ from others

For testing Block/Replication effect

H_0 : no difference in Block means

H_1 : significant difference in Block means

ANOVA sketch for randomized block design with **t** Treatments, **r** replications

S.V.	df	SS	MSS	F cal
Block	r-1	SSblk	MSblk	
Treatment	t-1	SSstr	MStrt	MStrt/ MS error
Error	(t-1)(r-1)	SS error	MS error	
Total	n-1	SS tot		

Merits: Used where one-way heterogeneity present in experimental material

Demerits: If number of treatments are more than it is difficult to maintain homogeneity in block also plot size will be reducing.

FACTORIAL EXPERIMENTS

Factorial experiments involve simultaneously more than one factor each at two or more levels. The experimenter finds the main effects and the interaction effects for different factors.

Example - RCBD with a 2x4 Factorial leads to combinations – (a0b0, a0b1, a0b2, a0b3, a1b0, a1b1, a1b2, a1b3). These eight treatment combinations may assign to blocks randomly as

Block 01	T3 a0b2	T7 a1b2	T2 a0b1	T6 a1b1	T4 a0b3	T5 a1b0	T1 a0b0	T8 a1b3
----------	-------------------	-------------------	-------------------	-------------------	-------------------	-------------------	-------------------	-------------------

Linear Model

$$Y_{ijk} = \mu + \alpha_i + \beta_j + (\alpha\beta)_{jk} + \gamma_k + e_{ijk}$$

Where: μ = Experiment mean

α_i = Effect of i^{th} level of factor A

β_j = Effect of the j^{th} level of factor B

γ_k = Effect of the k^{th} replicate

$(\alpha\beta)_{jk}$ = A x B interaction effect

e_{ijk} = Random error

Symmetric & Asymmetric factorial experiments-

If each factor has the same levels, then experiment is known as symmetrical factorial experiment ($p \times p = p^2$). If each factor has different levels, then an experiment is known as an asymmetrical factorial experiment ($p \times q$).

ANOVA sketch of Data A x B factorial experiment.

Source	df	SS	MS	F
Blocks	$r - 1$	SS(Block)		
Factor A	$a - 1$	SSA	MSA	MSA / MSerr
Factor B	$b - 1$	SSB	MSB	MSB / MSerr
A x B	$(a - 1)(b - 1)$	SS(AxB)	MS(AxB)	MS(AxB) / MSerr
Error	$(r-1)(ab-1)$	SSerr	MSerr	
Total (subplots)	$rab - 1$	SS tot		

THE SPLIT-PLOT TECHNIQUE

Definition: -The split-plot design involves assigning the levels of one factor to main plots and then assigning the levels of second factor to sub-plots within each main plot. The split-plot design results from a specialized randomization scheme for a factorial experiment.

Disadvantages of Split-plot designs

- Because of large plot size (main plot treat) and smaller plot size (sub plot treat), both factors are not tested with equal precision
- When there is more than two factors it becomes complex.

Split-Plot layout

A: main plot factor, 3 levels. (Spacing)

B: subplot factor, 5 levels(variety). With 02 replications.

Block I	S2	S3	S1	Block II	S3	S2	S1
	V2	V3	V4		V5	V4	V5
	V5	V4	V1		V1	V1	V1
	V3	V5	V5		V4	V5	V3
	V1	V2	V2		V3	V2	V2
	V4	V1	V3		V2	V3	V4

The linear model for the split-plot

The linear model for the split-plot with RBD main plots is

$$Y_{ijk} = \mu + R_k + \alpha_i + \gamma_{ik} + \beta_j + (\alpha\beta)_{ij} + \epsilon_{ijk}$$

Where: $k = 1, \dots, r$ indexes the reps,

$i = 1, \dots, a$ indexes the main plot levels, and

$j = 1, \dots, b$ indexes the subplot levels.

Y_{ijk} : observation

μ : mean

R_k : replication effect

α_i : Factor A effect

β_j : Factor B effect

$(\alpha\beta)_{ij}$: Interaction AB effect

γ_{ik} : error associated with the main plots

ϵ_{ijk} error associated with the subplots.

The variance σ^2_γ is usually larger than σ^2_ϵ .

Split-plot ANOVA

Source	Df	SS	MS	F
Blocks	$r - 1$	SS(Block)		
Factor A	$a - 1$	SSA	MSA	MSA / MS(MPE)
Error A= A*block	$(a - 1)(r - 1)$	SS(MPE)	MS(MPE)	
Factor B	$b - 1$	SSB	MSB	MSB / MS(SPE)
A x B	$(a - 1)(b - 1)$	SS(AxB)	MS(AxB)	MS(AxB) / MS(SPE)
Error	$a(r-1)(b-1)$	SS(SPE)	MS(SPE)	
Total (subplots)	$rab - 1$	SS		

Split-block (or strip-plot) design

In the strip-plot or split-block design the subplot treatments are applied in strips across an entire replication of main plot treatments.

Comparison of a 5 x 4 split-plot and a 5 x 4 strip-plot (Here only one replication is shown). In the strip-plot the terms main plot and subplot but there is no difference between the two (i.e., they are symmetric).

A3	A2	A1	A5	A4
B2	B1	B2	B3	B4
B1	B3	B1	B2	B3
B3	B2	B4	B4	B1
B4	B4	B3	B1	B2
Split-plot				

A3	A2	A1	A5	A4
B2	B2	B2	B2	B2
B4	B4	B4	B4	B4
B1	B1	B1	B1	B1
B3	B3	B3	B3	B3
Split-block or Strip-plot				

Note that the subplot treatments are continuous across the entire block or main plot, and thus each subplot treatment splits the block. Another term applicable to this layout is strip-plot, as both A and B treatments are in strips.

The A and B treatments are independently randomized in each replication.

Reasons for doing a split-block design

- Physical operations (e.g. tractor operation, irrigation, harvesting)
- The design tends to sacrifice precision in the main effects and improve precision in the interaction effects.

Linear model for the split-block design

The linear model for the split-block with RCB main plots is

$$Y_{ijk} = \mu + R_k + \alpha_i + \beta_j + \gamma_{ik} + \theta_{jk} + (\alpha\beta)_{ij} + \epsilon_{ijk},$$

Y_{ijk} : observation
 μ : mean
 R_k : replication effect
 α_i : Factor A effect
 β_j : Factor B effect
 $(\alpha\beta)_{ij}$: Interaction AB effect
 γ_{ik} : error associated with the main plots
 θ_{jk} : error associated with the subplots.
 ϵ_{ijk} : error associated with the interaction.

The ANOVA table for the RCBD split block design is

Source	Df	SS	MS	F
Blocks	$r - 1$	SS(Block)		
Factor A	$a - 1$	SSA	MSA	MSA / MS(MPE)
Error A= A*block	$(a - 1)(r - 1)$	SS(MPE)	MS(MPE)	
Factor B	$b - 1$	SSB	MSB	MSB / MS(STPE)
Error B= B*block	$(b - 1)(r - 1)$	SS(STPE)	MS(STPE)	
A x B	$(a - 1)(b - 1)$	SS(AxB)	MS(AxB)	MS(AxB) / MS(SPE)
Error C=A*B*block	$(a-1)(r-1)(b-1)$	SS(SPE)	MS(SPE)	
Total (subplots)	$rab - 1$	SS		

The package **agricolae** offers a broad functionality in the design of experiments, especially for experiments in agriculture and improvements of plants, which can also be used for other purposes.

It contains the analysis of: complete randomized blocks, split plot and strip plot, Latin, Graeco-Latin, augmented block designs, lattice, alpha, cyclic, balanced incomplete block designs. It also has several procedures of experimental data analysis, such as the comparisons of treatments of Waller-Duncan, Duncan, Bonferroni, Student-Newman-Keuls, Scheffe, Ryan, Einot and Gabriel and Welsch multiple range test or the classic LSD and Tukey; and non-parametric comparisons, such as Friedman, Durbin, Kruskal-Wallis, Median and Waerden, stability analysis, and other procedures applied in genetics, and also procedures in biodiversity and descriptive statistics,

```
> install.packages("agricolae")
```

```
> library(agricolae)
```

Layout for CRD, RBD, LSD, Split and Strip plot designs

- **Form layout for CRD design for five treatments with 4,3,4,4,3 replications respectively.**

```
trt <- c("A", "B", "C", "D", "E")
repeticion <- c(4, 3, 4, 4, 3)
outdesign <- design.crd(trt, r=repeticion, seed=777, serie=0)
outdesign
```

- **Form layout for RBD design for five treatments with 4 replications**

```
> trt <- c("A", "B", "C", "D", "E")
> repeticion <- 4
> outdesign <- design.rcbd(trt, r=repeticion, seed=-513, serie=2)
> book2 <- outdesign$book
> book2 <- zigzag(outdesign) # zigzag numeration
> print(outdesign$sketch)
```

```
      [,1] [,2] [,3] [,4] [,5]
[1,] "E"  "B"  "D"  "A"  "C"
[2,] "B"  "A"  "D"  "C"  "E"
[3,] "C"  "E"  "A"  "B"  "D"
[4,] "D"  "C"  "E"  "B"  "A"
```

- **Form layout for Latin square design for four treatments**

```
> trt <- c("A", "B", "C", "D")
> outdesign <- design.lsd(trt, seed=543, serie=2)
> print(outdesign$sketch)
      [,1] [,2] [,3] [,4]
[1,] "B"  "C"  "A"  "D"
[2,] "D"  "A"  "C"  "B"
[3,] "C"  "D"  "B"  "A"
[4,] "A"  "B"  "D"  "C"
```

- **Form layout for Split plot designs for main plot with four treatments and sub plot with three treatments.**

```
> trt1<-c("A","B","C","D")
> trt2<-c("s1","s2","s3")
> outdesign <- design.split(trt1,trt2,r=3,serie=2,seed=543)
> book10 <- outdesign$book
> head(book10)
> p<-book10$str1[seq(1,36,3)]
> q<-NULL
> for(i in 1:12)
+ q <- c(q,paste(book10$str2[3*(i-1)+1],book10$str2[3*(i-1)+2], book10$str2[3*(i-1)+3]))
> print(t(matrix(p,c(4,3))))
      [,1] [,2] [,3] [,4]
[1,] "D"  "B"  "A"  "C"
[2,] "B"  "C"  "A"  "D"
[3,] "D"  "B"  "A"  "C"
> print(t(matrix(q,c(4,3))))
      [,1]      [,2]      [,3]      [,4]
[1,] "s2 s1 s3"  "s1 s2 s3"  "s3 s1 s2"  "s3 s2 s1"
[2,] "s2 s3 s1"  "s1 s2 s3"  "s2 s3 s1"  "s1 s3 s2"
[3,] "s3 s1 s2"  "s2 s3 s1"  "s3 s1 s2"  "s1 s3 s2"
```

- **Form layout for Strip-plot designs for factor A with four treatments and factor B with three treatments.**

```
> trt1<-c("A","B","C","D")
```

```

> trt2<-c("s1","s2","s3")
> outdesign <-design.strip(trt1,trt2,r=3,serie=2,seed=543)
> book11 <- outdesign$book
> head(book11)
> t3<-paste(book11$trt1, book11$trt2)
> B1<-t(matrix(t3[1:12],c(3,4)))
> B2<-t(matrix(t3[13:24],c(3,4)))
> B3<-t(matrix(t3[25:36],c(3,4)))
> print(B1)
      [,1] [,2] [,3]
[1,] "D s2" "D s1" "D s3"
[2,] "B s2" "B s1" "B s3"
[3,] "A s2" "A s1" "A s3"
[4,] "C s2" "C s1" "C s3"
> print(B2)
      [,1] [,2] [,3]
[1,] "C s2" "C s1" "C s3"
[2,] "B s2" "B s1" "B s3"
[3,] "A s2" "A s1" "A s3"
[4,] "D s2" "D s1" "D s3"
> print(B3)
      [,1] [,2] [,3]
[1,] "A s3" "A s2" "A s1"
[2,] "B s3" "B s2" "B s1"
[3,] "D s3" "D s2" "D s1"
[4,] "C s3" "C s2" "C s1"

```

```

> head(aa)
  trt rep ear_plnt earlength grainwt grainyield
1  T1  R1   50.2    20.5    3.9    104.9
2  T2  R1   41.8    19.5    3.7     88.0
3  T3  R1   39.2    19.0    4.5     80.0

```

```
4  T4  R1  37.8  20.0  4.3  80.8
5  T5  R1  35.6  20.0  4.1  60.0
6  T6  R1  53.4  19.2  4.2  96.4
```

Completely randomised block design (CRD)

```
> model_1<-aov(grainyield ~trt,data=aa)
```

```
> summary(model_1)
```

```
          Df Sum Sq Mean Sq F value  Pr(>F)
trt         7  6248   892.6   11.36 2.77e-06 ***
Residuals  24  1885    78.6
```

```
---
```

```
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Interpretation:

Treatment effect is significant

Randomised block design (RBD)

```
> model_2<-aov(grainyield ~trt+repl,data=aa)
```

```
> summary(model_2)
```

```
          Df Sum Sq Mean Sq F value  Pr(>F)
trt         7  6248   892.6  12.369 3.42e-06 ***
rep         3   370   123.3   1.708  0.196
Residuals  21  1516    72.2
```

```
---
```

```
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Interpretation:

Treatment effect is significant at 1 % level of significance

Pair-wise treatment mean comparison

```
> library(agricolae)
```

```
> comparison_LSD= LSD.test(model_1,"trt",alpha=0.05,group=TRUE)
```

```
> comparison_LSD
```

```
$statistics
```

```

MSerror Df  Mean    CV t.value  LSD
78.55594 24 84.70312 10.46382 2.063899 12.93489

```

```
$groups
```

```
grainyield groups
```

```

T6 100.375  a
T2  98.250  ab
T4  91.625  abc
T7  90.975  abc
T1  85.675  bcd
T8  82.025  cd
T3  74.575  d
T5  54.125  e

```

Interpretation:

- CD value =12.93 @ 5% Level of significance
- Similar letters shows non significance of treatment means

```
comparison_LSD_2= LSD.test(model_2,"trt",alpha=0.05,group=TRUE)
```

```
> comparison_LSD_2
```

```
$statistics
```

```

MSerror Df  Mean    CV t.value  LSD
72.16674 21 84.70312 10.02927 2.079614 12.49212

```

```
$groups
```

```
grainyield groups
```

```

T6 100.375  a
T2  98.250  a
T4  91.625  ab
T7  90.975  ab
T1  85.675  bc
T8  82.025  bc
T3  74.575  c
T5  54.125  d

```

Interpretation:

- CD value =12.49 @ 5% Level of significance
- Similar letters shows non significance of treatment means

One can also use Duncan's Multiple Comparison test instead of LSD test

```
> dmrt=duncan.test(model_2,"trt",alpha=0.05,console=TRUE)
```

Analysis of Factorial experiment in RBD layout

Use fact.txt data

```
> fact=read.table("E:\\TNAU\\workshop\\fact.txt",header=TRUE)
> model<-aov(yield~rep+A+B+A:B,data=fact)
> summary(model)
```

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
rep	3	20.12	6.707	1.752	0.175473
A	3	80.16	26.719	6.982	0.000917 ***
B	2	38.17	19.086	4.987	0.012811 *
A:B	6	1.52	0.254	0.066	0.998692
Residuals	33	126.30	3.827		

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Interpretation:

- Effect of Factor A effect is significant at 1 % level of significance
- Effect of Factor B is significant at 5 % level of significance
- Interaction of AB is not significant

```
> outA<-LSD.test(model,"A",alpha=0.05, group=TRUE)
> outB<-LSD.test(model,"B",alpha=0.05, group=TRUE)
> df=model$df.residual
> summary(model)[[1]]$"Mean Sq"
```

```
[1] 6.7066667 26.7194444 19.0858333 0.2536111 3.8271212
```

```
> er=summary(model)[[1]]$"Mean Sq"[5]
```

```
> outAB<-with(fact,LSD.test(yield,A:B,df,er))
```

```
> outA
```

```
$statistics
```

MS	Error	Df	Mean	CV	t.value	LSD
3.827121	33	5.866667	33.34607	2.034515	1.624881	

```
$groups
```

```
yield groups
```

```
A4 7.383333 a
```

```
A3 6.875000 a
```

```
A2 4.900000 b
```

```
A1 4.308333 b
```

```
> outB
```

```
$statistics
```

MS	Error	Df	Mean	CV	t.value	LSD
3.827121	33	5.866667	33.34607	2.034515	1.407188	

```
$groups
```

```
yield groups
```

```
B3 7.0125 a
```

```
B2 5.7500 ab
```

```
B1 4.8375 b
```

Here no need to proceed for comparison of Interaction AB effect as it is non-significant in ANOVA table

Analysis of Split plot experiment in RBD layout

```
> model<-with(fact,sp.plot(rep,A,B,yield))
```

```
Analysis of Variance Table
```

```
Response: yield
```

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
rep	3	20.120	6.7067	0.5247	0.6761
A	3	80.158	26.7194	2.0906	0.1718

```
Ea 9 115.028 12.7809
```

```
B 2 38.172 19.0858 40.6562 1.962e-08 ***
```

```
A:B 6 1.522 0.2536 0.5402 0.7724
```

```
Eb 24 11.267 0.4694
```

```
---
```

```
Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
cv(a) = 60.9 %, cv(b) = 11.7 %, Mean = 5.866667
```

Interpretation:

- Effect of Factor A effect is non significant
- Effect of Factor B is significant at 1 % level of significance
- Interaction of AB is not significant

```
> outA<-with(fact,LSD.test(yield,A,model$gl.a,model$Ea))
```

```
> outB<-with(fact,LSD.test(yield,B,model$gl.b,model$Eb))
```

```
> outAB<-with(fact,LSD.test(yield,A:B,model$gl.b,model$Eb))
```

Here no need to proceed for comparison of factor A and Interaction AB effect as it is non-significant in ANOVA table

```
> outB
```

```
$statistics
```

```
MSerror Df Mean CV t.value LSD
```

```
0.4694444 24 5.866667 11.67887 2.063899 0.4999602
```

```
$groups
```

```
yield groups
```

```
B3 7.0125 a
```

```
B2 5.7500 b
```

```
B1 4.8375 c
```

Analysis of Strip plot experiment in RBD layout

```
> model<-with(fact,strip.plot(rep,A,B,yield))
```

```
Analysis of Variance Table
```

Df	Sum Sq	Mean Sq	F value	Pr(>F)
----	--------	---------	---------	--------


```
rep 3 20.120 6.7067
A 3 80.158 26.7194 2.0906 0.1718
Ea 9 115.028 12.7809
B 2 38.172 19.0858 66.3855 8.083e-05 ***
Eb 6 1.725 0.2875
B:A 6 1.522 0.2536 0.4784 0.8156
Ec 18 9.542 0.5301
---
```

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

cv(a) = 60.9 %, cv(b) = 9.1 %, cv(c) = 12.4 %, Mean = 5.866667

Interpretation:

- Effect of Factor A effect is non-significant
- Effect of Factor B is significant at 1 % level of significance
- Interaction of AB is not significant

```
> outA<-with(fact,LSD.test(yield,A,model$gl.a,model$Ea))
> outB<-with(fact,LSD.test(yield,B,model$gl.b,model$Eb))
> outAB<-with(fact,LSD.test(yield,A:B,model$gl.c,model$Ec))
```

Here no need to proceed for comparison of factor A and Interaction AB effect as it is non-significant in ANOVA table

```
> outB
$statistics
MSerror Df Mean CV t.value LSD
0.2875 6 5.866667 9.139607 2.446912 0.4638657

$groups
yield groups
B3 7.0125 a
B2 5.7500 b
B1 4.8375 c

> plot(outA, variation="SE")
> plot(outB, variation="SE")
```

```
> plot(outAB, variation="SE", las=2)
```

Suggested Readings:

Gomez, K.A. and Gomez, A. (1984) Statistical Procedure for Agricultural Research—
Hand Book. John Wiley & Sons, New York.

Steel and Torrie (1960) Principles and procedures of statistics. McGraw-Hill Book Co.11.

Wood, P.J. and Burley, J. (1991) A Tree for All Reasons. ICRAF.

Rangaswamy, R.(1995). A Text Book of Agricultural Statistics New Age International (P)
Limited, Publishers.

Nigam, A. K. and Gupta, V. K. (1979) Handbook of Analysis of Agricultural
Experiments. Publication of I. A. S. R.I. New Delhi.

Split Plot and Strip Plot Data Analysis

Aliza Pradhan

ICAR-National Institute of Abiotic Stress Management, Baramati, Pune, 413115

Email: alizapradhan@gmail.com

Split-Plot Design

In factorial experiments, it is sometimes necessary to accommodate factors that require different plot sizes. For example, treatments involving irrigation or tillage typically need larger plots, whereas those involving fertilizers or weedicides can be conducted on smaller plots. To manage such situations within the same experiment, the split-plot design has been developed. In this design, the factor requiring larger plots is assigned as main plot. Each main plot is then divided into smaller sub-plots, which are used for the second factor. Treatments are randomly assigned to the main plots and sub-plots accordingly.

Advantages

- Increased precision in the estimates of sub-plot treatments and interaction compared to factorial RBD with two factors
- Helps in saving the experimental material

Disadvantages

- Effects of the main plot treatments estimated with less precision
- Analysis becomes more complex when more than two factors or missing data occur.

Layout and Analysis

M: Main plot factor, 3 levels (Trash)

N: Subplot factor, 3 levels (Nutrient)

No. of replications: 03

Replication I			Replication II			Replication III		
M1	M2	M3	M2	M3	M1	M3	M1	M2
N1	N2	N3	N3	N2	N2	N1	N2	N3
N2	N3	N1	N2	N1	N3	N2	N3	N1
N3	N1	N2	N1	N3	N1	N3	N1	N2

The linear model for the split-plot design

The linear model for the split-plot with RBD main plots is

$$Y_{ijk} = \mu + r_i + m_j + e_{ij} + s_k + (ms)_{jk} + e_{ijk}$$

Where Y_{ijk} = the observation of i^{th} replication, j^{th} main plot and k^{th} sub-plot,

μ = overall mean

r_i = i^{th} replication effect

m_j = j^{th} main plot treatment effect

e_{ij} = main plot error or error (a)

s_k = k^{th} sub-plot treatment effect

$(ms)_{jk}$ = Interaction effect

e_{ijk} = error component for sub-plot and interaction or error (b)

Split-plot ANOVA

Sources of variation	df	SS	MS	F
Replication	$r - 1$	R SS	R MS	R MS/ E MS(a)
Factor A	$m - 1$	A SS	A MS	A MS/ E MS(a)
Error (a)	$(r-1)(m-1)$	SS(a)	E MS(a)	
Factor B	$s - 1$	B SS	B MS	B MS/ E MS(b)
A x B	$(m - 1)(s - 1)$	AB SS	AB MS	AB MS/ E MS(b)
Error (b)	$m(r-1)(s-1)$	E SS(b)	E MS(b)	
Total	$rms - 1$	TSS		

Example dataset

For example, suppose a researcher wants to study the effect of two different fertilizers methods and four fertilizer doses on plant growth.

Factor A (mainplot): Fertilizer methods

- F1 = Broadcasting
- F2 = Fertigation

Factor B (subplot): Fertilizer dose

- T1 = Control (0 kg per acre)
- T2 = 25 kg N per acre
- T3 = 50 kg N per acre
- T4 = 75 kg N per acre

Let's analyze the data in R.

```
# Creating data

library(dplyr)

set.seed(123)

Control <- rnorm(n = 6, mean = 15, sd = 3.4)
T25kg <- rnorm(n = 6, mean = 22, sd = 5.5)
T50kg <- rnorm(n = 6, mean = 45, sd = 9.5)
T75kg <- rnorm(n = 6, mean = 35, sd = 8.2)

yield <- as.data.frame(cbind(Control, T25kg, T50kg, T75kg))
yield$Method <- rep(c("Broadcasting", "Fertigation"), each = 3)
yield$Rep <- rep(c(1:3), times = 2)

df <- yield %>% tidyr::pivot_longer(!c(Rep, Method),
                                   names_to = "Dose",
                                   values_to = "yield")

df <- as.data.frame(df)

df
str(df)

# converting variables to factors

df$Method <- as.factor(df$Method)
df$Dose <- as.factor(df$Dose)

str(df)
```

```

# 'data.frame': 24 obs. of  4 variables:
#  $ Method: chr  "Broadcasting" "Broadcasting" "Broadcasting" "Broadcasting" ...
#  $ Rep    : int  1 1 1 1 2 2 2 2 3 3 ...
#  $ Dose   : chr  "Control" "T25kg" "T50kg" "T75kg" ...
#  $ yield  : num  13.1 24.5 48.8 40.8 14.2 ...

# converting variables to factors
df$Method <- as.factor(df$Method)
df$Dose <- as.factor(df$Dose)
str(df)

# 'data.frame': 24 obs. of  4 variables:
#  $ Method: Factor w/ 2 levels "Broadcasting",...: 1 1 1 1 1 1 1 1 1 1 ...
#  $ Rep    : int  1 1 1 1 2 2 2 2 3 3 ...
#  $ Dose   : Factor w/ 4 levels "Control","T25kg",...: 1 2 3 4 1 2 3 4 1 2 ...
#  $ yield  : num  13.1 24.5 48.8 40.8 14.2 ...

library(agricolae)

# Fitting ANOVA model for split plot design
model <- with(df,
  sp.plot(block = Rep,
    pplot = Method,
    splot = Dose,
    Y = yield))

#
# ANALYSIS SPLIT PLOT:  yield
# Class level information
#
# Method      :  Broadcasting Fertigation
# Dose       :  Control T25kg T50kg T75kg
# Rep        :  1 2 3
#
# Number of observations:  24
#
# Analysis of Variance Table
#

```

```

#
# Response: yield
#           Df Sum Sq Mean Sq F value    Pr(>F)
# Rep        2  173.77   86.89   8.0855 0.1100656
# Method      1    6.52    6.52   0.6067 0.5175765
# Ea         2   21.49   10.75    NaN      NaN
# Dose        3 2902.30  967.43 14.9754 0.0002329 ***
# Method:Dose 3   47.25   15.75   0.2438 0.8641451
# Eb        12   775.22   64.60    NaN      NaN
# ---
# Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
#
# cv(a) = 11.4 %, cv(b) = 28 %, Mean = 28.69679

```

The output of the model showed that only fertilizers application dose variable showed highly significant effect on the rice yield with probability value lower than 0.01. So next we shall proceed with mean comparison test to see differences in the mean values of variable Dose.

Getting Edf and EMS from sp.plot model

Error df for main plot factor (Method)

```
Edfa <- model$gl.a
```

```
Edfa
```

Error df for subplot factor (Dose)

```
Edfb <- model$gl.b
```

```
Edfb
```

Error MS for main plot factor (Method)

```
EMSa <- model$Ea
```

```
EMSa
```

Error MS for subplot factor (Dose)

```
EMSb <- model$Eb
```

```
EMSb
```

```

# [1] 2
# [1] 12
# [1] 10.74608
# [1] 64.60155

```

```
LSD <- with(df, LSD.test(y = yield,
```

```
trt = Dose,
```

```

DFerror = Edfb,
MSerror = EMSb,
alpha = 0.05,
group = TRUE,
console = TRUE))

```

```

#
# Study: yield ~ Dose
#
# LSD t Test for yield
#
# Mean Square Error: 64.60155
#
# Dose, means and individual ( 95 %) CI
#
#      yield      std r      LCL      UCL      Min      Max
# Control 16.52032  3.247008 6  9.370982 23.66966 13.09438 20.83122
# T25kg   21.67663  4.974571 6 14.527296 28.82597 15.04216 28.73245
# T50kg   45.43345 11.848624 6 38.284115 52.58279 26.31714 61.97567
# T75kg   31.15676  5.400990 6 24.007419 38.30609 26.24385 40.75112
#
# Alpha: 0.05 ; DF Error: 12
# Critical Value of t: 2.178813
#
# least Significant Difference: 10.11069
#
# Treatments with the same letter are not significantly different.
#
#      yield groups
# T50kg  45.43345    a
# T75kg  31.15676    b
# T25kg  21.67663   bc
# Control 16.52032    c

```

Interpretation of LSD output: Maximum yield was recorded in plots where nitrogen was applied at the rate of 50 kg per acre followed by 75 kg while the least production was recorded in control where nitrogen was not applied.

B) Split-Block (or Strip-Plot) Design

Another variant of the split-plot design is the strip-plot design, also known as the split-block design. This design is recommended when two factors are involved and both require large plot sizes. For example, it is suitable for experiments involving tillage and water management. The strip-plot design is also useful when higher precision is desired for the interaction between the two factors, rather than for the individual factors themselves.

In a strip-plot or split-block design, the subplot treatments are applied in strips that run across the entire replication of the main plot treatments. Although the terms "main plot" and "subplot" are used, there is no hierarchical difference between them in this design. Each subplot treatment forms a continuous strip across the entire block, effectively splitting the block. The name "strip-plot" comes from the fact that both factors are arranged in strips. Here, both the factors are independently randomized within each replication.

Layout and Analysis

T: Main plot/vertical strip factor, 3 levels (Tillage)

I: Subplot/horizontal strip factor, 3 levels (Irrigation)

No. of replications: 03

Replication I			Replication II			Replication III		
T1	T2	T3	T2	T3	T1	T3	T1	T2
I1	I1	I1	I3	I3	I3	I2	I2	I2
I2	I2	I2	I1	I1	I1	I3	I3	I3
I3	I3	I3	I2	I2	I2	I1	I1	I1

Linear model for the strip-plot design

$$Y_{ijk} = \mu + R_k + \alpha_i + \beta_j + \gamma_{ik} + \Theta_{jk} + (\alpha\beta)_{ij} + \epsilon_{ijk}$$

Where: k = 1, ..., r indexes the replications,

i = 1, ..., a indexes the main plot levels, and

j = 1, ..., b indexes the subplot levels.

Y_{ijk} : observation

μ : mean

R_k : replication effect

α_i : Factor A effect

β_j : Factor B effect

$(\alpha\beta)_{ij}$: Interaction AB effect

γ_{ik} : error associated with the main plots

Θ_{jk} : error associated with the subplots

ϵ_{ijk} : error associated with the interaction

Strip-Plot ANOVA

Source	Df	SS	MS	F
Replication	r-1	R SS	R MS	R MS/ E MS (a)
Factor A	a-1	A SS	A MS	A MS/E MS (a)
Error (a)	(r-1) (a-1)	E SS (a)	MS(MPE)	
Factor B	b-1	B SS	B MS	B MS/E MS(a)
Error (b)	(r-1) (b-1)	E SS(b)	E MS (b)	
A x B	(a-1) (b-1)	AB SS	AB MS	AB MS / E MS (c)
Error (c)	(r-1) (a-1) (b-1)	E SS(c)	E MS(c)	
Total	rab-1	T SS		

Analysis of strip-plot design in R

```
library(agricolae)
```

```
# Fitting ANOVA model for strip plot design
```

```
model <- with(df, strip.plot(block = Rep,
```

```
    pplot = Method,
```

```
    splot = Dose,
```

```
    Y = yield))
```

References

Cochran, W. G., & Cox, G. M. (1957). Experimental designs (2nd ed.). *John Wiley & Sons*.

Gomez, K. A., & Gomez, A. A. (1984). Statistical procedures for agricultural research (2nd ed.). *John Wiley & Sons*.

Mead, R., Curnow, R. N., & Hasted, A. M. (2017). Statistical methods in agriculture and experimental biology (3rd ed.). *CRC Press*.

Montgomery, D. C. (2020). Design and analysis of experiments (10th ed.). *John Wiley & Sons*.

Steel, R. G. D., Torrie, J. H., & Dickey, D. A. (1997). Principles and procedures of statistics: A biometrical approach (3rd ed.). *McGraw-Hill*.

Repeated Measures ANOVA

Vandita Kumari

ICAR-Central Arid Zone Research Institute, Jodhpur, Rajasthan, 342003

Email: vandita.iasri@gmail.com

1. Repeated measures design

A Repeated Measures design is an experimental design in which the same participants engage in each condition of the independent variable. In other words, every condition of the experiment involves the identical group of participants. This type of design is also referred to as within groups or within-subjects design. It entails obtaining multiple measures of a single variable from the same or matched individuals, either under varying conditions or across multiple time periods. This contrasts with a design where participants are randomly assigned to a treatment and stay on that treatment for the entirety of the trial. For instance, repeated measurements are gathered in a longitudinal study to evaluate changes over time, or patients receive a single treatment, with outcomes assessed over specific intervals (e.g., at 1, 4, and 8 weeks). Repeated measures ANOVAs are the suitable statistical tests for drawing conclusions regarding repeated measures designs.

2. Repeated measures ANOVA

2.1 Introduction

We can think back to the examples of paired t-tests, where we have a simple experiment involving only two experimental conditions. In paired-samples t-tests, we are limited to comparing two means. But what if our design included more than two experimental conditions? For instance, an experiment could consist of three levels for the independent variable, with each participant providing data for each of these three levels. Therefore, this scenario must be treated as an ANOVA issue. ANOVAs can assess whether there is a difference between two or more means. The repeated measures ANOVA serves as an extension of the paired t-test and is utilized to compare subjects across time. It is a statistical method that can incorporate both within-subject and between-subject factors. An example of this could involve evaluating a disease over the course of 12 weeks of treatment (for example, at weeks 0, 6, and 12). The design of the repeated measures ANOVA can be represented in a tabular format, as follows:

Subjects	Time/Condition		
	T1	T2	T3
S ₁	S ₁	S ₁	S ₁
S ₂	S ₂	S ₂	S ₂
S ₃	S ₃	S ₃	S ₃
S ₄	S ₄	S ₄	S ₄
S ₅	S ₅	S ₅	S ₅
S ₆	S ₆	S ₆	S ₆

This table outlines a study involving six participants (S₁ to S₆) who were assessed under three different conditions or at three distinct time points (T₁ to T₃). It is highlighted that "time/condition" may also be referred to as "treatment," which is classified as a within-subjects factor. All these terms pertain to the same concept, namely subjects undergoing repeated assessments at various time intervals or under different conditions/treatments.

2.2 Hypothesis for Repeated Measures ANOVA

The repeated measures ANOVA is utilized to evaluate whether there are differences among related population means. First, we define the null hypothesis, which asserts that the population means are equal:

$$H_0: \mu_1 = \mu_2 = \mu_3 = \dots = \mu_k$$

In this case, μ represents the population mean, and k denotes the number of related groups.

The alternative hypothesis (H₁) posits that the related population means are not equal (meaning at least one mean differs from another), i.e.

$$H_1: \text{at least two means are significantly different}$$

One major benefit of repeated measures ANOVA, as well as repeated measures designs in general, is the capability to separate out variability caused by individual differences.

2.3 Partitioning the Sums of Squares

Consider the overall formula for the F-statistic:

$$F = MS_{\text{Treatment}} / MS_{\text{Error}} = (SS_{\text{Treatment}}/df_{\text{Treatment}})/(SS_{\text{Error}}/df_{\text{Error}}) \quad (2.1)$$

In a typical design, variability arises from individual differences, which combines with the treatment and error components:

$$SS_{\text{Total}} = SS_{\text{Treat}} + SS_{\text{Error}} \quad (2.2)$$

$$df_{\text{Total}} = n - 1$$

In a repeated measures design, we separate subject variability from the treatment and error components. In this scenario, variability can be divided into between-treatments variability (or within-subjects effects, not accounting for individual differences) and within-treatments variability. The within-treatments variability can further be broken down into between-subjects variability (individual differences) and error (excluding individual differences):

$$SS_{\text{Total}} = SS_{(\text{between treatment})} + SS_{\text{within treatment}} \quad (2.3)$$

$$SS_{\text{Total}} = SS_{\text{Treat (excluding individual difference)}} + SS_{\text{Subjects}} + SS_{\text{Error}} \quad (2.4)$$

$$df_{\text{Total}} = df_{\text{Treat (within subjects)}} + df_{\text{between subjects}} + df_{\text{error}} = (k - 1) + (n - 1) + ((n - k)(n - 1))$$

Referring to the general structure of the F-statistic, it is evident that by removing the between-subjects variability, the F-value will increase because the error sum of squares will be smaller, leading to a reduced MSE. It is important to note that partitioning variability decreases the degrees of freedom for the F-test, so the significance of the between-subjects variability must be substantial enough to compensate for the reduction in degrees of freedom. If the between-subjects variability is minimal, this method may actually lower the F-value.

As with any statistical analysis, specific assumptions must be fulfilled to justify utilizing this test. In Repeated Measures ANOVA, the assumptions include: no significant outliers, a normality assumption where each level of the independent variable should be approximately normally distributed, and the Assumption of Sphericity, which is the repeated measures equivalent of homogeneity of variances.

2.4 Calculating a Repeated Measures ANOVA

Consider a scenario in which a 6-month exercise training program was conducted, and six participants had their fitness levels assessed at three different times: before the intervention, at the 3-months, and after the intervention. Their information is presented below, accompanied by some calculations.

Exercise Intervention				
Subjects	Pre-	3 Months	6 Months	Subject Means
1	45	50	55	50
2	42	42	45	43
3	36	41	43	40
4	39	35	40	38

5	51	55	59	55
6	44	49	56	49.7
Monthly Means	42.8	45.3	49.7	
Grand Mean: 45.9				

In this context, the null hypothesis (H_0) states that the average blood pressure remains constant across all time points (pre-, 3 months, and 6 months). The alternative hypothesis suggests that the average blood pressure varies significantly at one or more time intervals. The F-test associated with the repeated measures ANOVA examines whether the observed differences are genuine or merely due to random variation. To begin, we will compute SS_{Treat} .

Calculation of SS_{Treat}

$$SS_{Treat} = \sum_{i=1}^k n_i (\bar{x}_i - \bar{x})^2 \quad (2.5)$$

where, k represents the total number of conditions, n_i denotes the count of subjects within each specific (ith) condition, \bar{x}_i signifies the average for each (ith) condition, and \bar{x} indicates the overall average. Therefore, in the example provided above, we have:

$$\begin{aligned} SS_{Treat} &= 6[(42.8 - 45.9)^2 + (45.3 - 45.9)^2 + (49.7 - 45.9)^2] \\ &= 143.44 \end{aligned}$$

Calculation of $SS_{subjects}$

Each subject can be viewed as a separate block. In simpler terms, we consider each subject as a level of a distinct factor referred to as subjects. Consequently, we can compute $SS_{subjects}$ in the following way:

$$SS_{Subject} = m \sum_{j=1}^m (\bar{x}_j - \bar{x})^2 \quad (2.6)$$

where m represents the total number of subjects, \bar{x}_j mean of subject j, and \bar{x} the grand mean. As illustrated in our example, it is:

$$\begin{aligned} SS_{Subject} &= 3 \sum_{j=1}^6 (\bar{x}_j - \bar{x})^2 \\ &= 3[(50-45.9)^2 + (43-45.9)^2 + (40-45.9)^2 + (38-45.9)^2 + (55-45.9)^2 + (49.7-45.9)^2] \\ &= 658.3. \end{aligned}$$

Calculation of $SS_{\text{within treatment}}(SS_w)$

The within-groups variation (SS_w) is determined in a manner similar to that of independent ANOVA, represented as follows:

$$SS_w = \sum_{i=1}^k \sum_{j=1}^m (x_{ij} - \bar{x}_i)^2$$

In this equation, x_{ij} refers to the measurement of the j^{th} subject under condition i .

$$\begin{aligned} SS_w &= [(45-42.8)^2 + (42-42.8)^2 + (326-42.8)^2 + (39-42.8)^2 + (51-42.8)^2 + (44-42.8)^2 + \\ &\quad [(50-45.3)^2 + (42-45.3)^2 + (41-45.3)^2 + (35-45.3)^2 + (55-45.3)^2 + (49-45.3)^2 + \\ &\quad [(55-49.7)^2 + \dots + (56-49.7)^2] \\ &= 715.5 \end{aligned}$$

Calculation of SS_{Error}

SS_{Error} can be computed using either equation 2.3 or 2.4. These methods are:

$$SS_{\text{Error}} = SS_{\text{Total}} - SS_{\text{Treat (excluding individual difference)}} - SS_{\text{Subject}} \quad (2.7)$$

$$SS_{\text{Error}} = SS_{\text{within treatment}} - SS_{\text{Subject}} \quad (2.8)$$

Both methods for calculating the F-statistic necessitate determining SS_{Subject} ; however, one method allows for SS_{Error} to be derived using either equation 2.7 or 2.8. In this case, we will compute it by using the first method, which involves calculating $SS_{\text{within treatment}}$.

We can now find SS_{Error} by substituting values into equation 2.7:

$$\begin{aligned} SS_{\text{Error}} &= 715.5 - 658.3 \\ &= 57.2 \end{aligned}$$

Next, to compute the mean sum of squares for treatment (MS_{Treat}), we divide SS_{Treat} by its corresponding degrees of freedom ($k - 1$), where k represents the number of time points. In our example, this yields:

$$MS_{Treat} = \frac{SS_{Treat}}{k-1} = 71.72$$

Likewise, we calculate the mean sum of squares for error (MS_{Error}) by dividing SS_{Error} by the degrees of freedom $(m-1)(k-1)$.

$$MS_{Error} = \frac{SS_{Error}}{(m-1)(k-1)}$$

$$= 5.72$$

As a result, we can compute the F-statistic as follows:

$$F = \frac{MS_{Treat}}{MS_{Error}} = \frac{71.72}{5.72}$$

$$= 12.53$$

Next, we should look for the critical F-statistic for our F-distribution, considering the degrees of freedom for treatment (df_{Treat}) and error (df_{Error}), to assess if our F-statistic reveals a statistically significant outcome or not.

2.5 Result of a Repeated Measures ANOVA

The critical value for the F-statistic in the above scenario is $F(2, 10) = 12.53$. This indicates that we can refute the null hypothesis and support the alternative hypothesis. Therefore, we can determine that time has a statistically significant effect on fitness gained from exercise. Alternatively, we can state that the six-month exercise training program significantly impacted fitness levels.

2.6 Tabular Presentation of a Repeated Measures ANOVA

Typically, the findings of a repeated measures ANOVA are conveyed in the narrative format, as demonstrated above, rather than in a table when drafting a report. Nevertheless, many statistical software programs display the outcomes of a repeated measures ANOVA in a tabular format. The table below illustrates the format one might encounter.

Source	SS	df	MS	F
Treatment	SS_{Treat}	$k-1$	MS_{Treat}	$MS_{\text{Treat}} / MS_{\text{Error}}$
Subjects	SS_{Subject}	$m-1$	MS_{Subject}	$MS_{\text{Subject}} / MS_{\text{Error}}$
Error	SS_{Error}	$(k-1)(m-1)$	MS_{Error}	
Total	SS_{Total}	$N-1$		

Repeated Measures ANOVA Output Table

In some cases, statistical software may not display the “subjects” row in the output table, and the “total” row may also be missing. The F-statistic located in the first row (which corresponds to the time/conditions) is the value that will indicate whether there is a significant difference between at least two means. Referring to the example in Section 2.4, if we exclude the Subjects and Total rows, we have:

Source	SS	df	MS	F
Treatment	143.44	2	71.72	12.53
Error	57.2	10	5.72	

2.7 Repeated measures ANOVA in R

2.7.1 One-way repeated measures ANOVA

We will conduct the analysis in R utilizing a dataset that includes the self-esteem scores of 10 individuals measured at three different time points during a particular diet to assess whether their self-esteem has improved. A repeated measures ANOVA will be executed to analyze the influence of time on the self-esteem scores. The dataset can be found in the “datarium” package. Below are the R codes along with the interpretation of the results they produce.

```
>install.packages("datarium")

>install.packages("rstatix")

>library(rstatix)

>library(datarium)

>data("selfesteem")

# Combine columns t1, t2, and t3 into a long format

# Transform id and time into categorical variables
```

```
>selfesteem <- selfesteem %>%
  gather(key = "time", value = "score", t1, t2, t3) %>%
  convert_as_factor(id, time)
>head(selfesteem, 3)
```

Output:

```
# A tibble: 3 x 3
  id   time score
  <fct> <fct> <dbl>
1 1    t1    4.01
2 2    t1    2.56
3 3    t1    3.24
```

RCode:

#Calculate summary statistics for self-esteem scores by groups (time): average and standard deviation (sd).

```
>selfesteem %>%
  group_by(time) %>%
  get_summary_stats(score, type = "mean_sd")
```

Result:

```
# A tibble: 3 x 5
  time variable    n mean  sd
  <fct> <chr>   <dbl> <dbl> <dbl>
1 t1    score    10 3.14 0.552
```

```
2 t2 score 10 4.93 0.863
```

```
3 t3 score 10 7.64 1.14
```

The assumption of normality can be assessed by applying the Shapiro-Wilk test at each time point in the dataset. A p-value exceeding 0.05 indicates that the data follows a normal distribution. You can utilize the R code below to evaluate the normality of the data:

```
>selfesteem %>% group_by(time) %>% shapiro_test(score)
```

Following result is obtained after running the code

```
# A tibble: 3 x 4
```

	time	variable	statistic	p
	<fct>	<chr>	<dbl>	<dbl>
1	t1	score	0.967	0.859
2	t2	score	0.876	0.117
3	t3	score	0.923	0.380

From the result of Shapiro-Wilk's test we can see that the $p > 0.05$, therefore we can say that the self-esteem score was normally distributed at each time point.

The assumption of sphericity will be automatically checked during the computation of the ANOVA test using the R function `anova_test()` [rstatix package]. Therefore, further we run the following R code

```
>res.aov <- anova_test(data = selfesteem, dv = score, wid = id, within = time)
```

```
get_anova_table(res.aov)
```

```
#Result
```

	# Effect	DFn	DFd	F	p p<.05	ges
#1	time	2	18	55.469	2.01e-08*	0.829

From the result we find that the self-esteem score is statistically significantly different at the different time points during the diet, since $F(2, 18) = 55.5$, $p < 0.0001$. Here, η^2 is the generalized effect size (amount of variability due to the within-subjects factor).

Post-hoc tests

One can perform multiple pairwise paired t -tests between the levels of the within-subjects factor (here time). P-values are adjusted using the Bonferroni multiple testing correction method. The R code can be written as:

```
# pairwise comparisons

>pwc <- selfesteem %>% pairwise_t_test( score ~ time, paired = TRUE, p.adjust.method =
"bonferroni" )

>pwc
```

The result obtained as:

```
# A tibble: 3 x 10

. y. group1 group2  n1  n2 statistic  df      p p.adj p.adj.signif
* <chr> <chr> <chr> <int> <int>   <dbl> <dbl>   <dbl> <dbl> <chr>

1 score t1    t2     10  10   -4.97    9 0.000772  2e-3 **

2 score t1    t3     10  10  -13.2    9 0.000000334 1e-6 *****

3 score t2    t3     10  10   -4.87    9 0.000886  3e-3 **
```

The self-esteem score was statistically significantly different at the different time points, $F(2, 18) = 55.5$, $p < 0.0001$, generalized eta squared = 0.82. Post-hoc analyses with a Bonferroni adjustment revealed that all the pairwise differences, between time points, were statistically significantly different ($p < 0.05$).

2.7.2 Two-way repeated measures ANOVA

A two-way repeated measures ANOVA can be conducted to assess if there is a significant interaction between diet and time regarding the self-esteem score. We'll use

the selfesteem2 dataset in datarium package of R. The dataset contains the self-esteem score measures of 12 individuals enrolled in 2 successive short-term trials (4 weeks): control (placebo) and special diet trials. Each participant performed all two trials. The order of the trials was counterbalanced and sufficient time was allowed between trials to allow any effects of previous trials to have dissipated. The self-esteem scores were measured at three different times: at the start (t1), in the middle (t2), and at the conclusion (t3) of the trials. We aim to determine whether there is a significant interaction between diet and time regarding the self-esteem scores from this dataset.

```
data("selfesteem2", package = "datarium")
selfesteem2 %>% sample_n_by(treatment, size = 1)

## # A tibble: 2 x 5
##   id   treatment    t1    t2    t3
##   <fct> <fct>    <dbl> <dbl> <dbl>
## 1 4     ctr      92    92    89
## 2 10    Diet     90    93    95

# Gather the columns t1, t2 and t3 into long format.
# Convert id and time into factor variables
selfesteem2 <- selfesteem2 %>%
  gather(key = "time", value = "score", t1, t2, t3) %>%
  convert_as_factor(id, time)

# Inspect some random rows of the data by groups
set.seed(123)

selfesteem2 %>% sample_n_by(treatment, time, size = 1)

## # A tibble: 6 x 4
##   id   treatment time    score
##   <fct> <fct>    <fct> <dbl>
## 1 4     ctr     t1      92
## 2 10    ctr     t2      84
## 3 5     ctr     t3      68
```

```
## 4 11 Diet t1 93
```

```
## 5 12 Diet t2 80
```

```
## 6 1 Diet t3 88
```

```
res.aov <- anova_test( data = selfesteem2, dv = score, wid = id, within = c(treatment, time) )
```

```
get_anova_table(res.aov)
```

```
## ANOVA Table (type III tests)
```

```
##      Effect DFn DFd  F      p p<.05 ges
```

```
## 1 treatment 1.00 11.0 15.5 2.00e-03 * 0.059
```

```
## 2 time 1.31 14.4 27.4 5.03e-05 * 0.049
```

```
## 3 treatment:time 2.00 22.0 30.4 4.63e-07 * 0.050
```

There is a statistically significant two-way interactions between treatment and time, $F(2, 22) = 30.4$, $p < 0.0001$

Post-hoc tests

A significant two-way interaction indicates that the impact of treatment on the self-esteem score depends on the level of the time and vice versa. Therefore, one can decompose a significant two-way interaction into simple main effect and simple pairwise comparisons. For simple main effect a one-way model is run for the first variable (treatment) at each level of the second variable (time). Further, if the simple main effect is significant then multiple pairwise comparisons is performed to determine which groups are different. Again, this needs to be repeated considering time as first variable.

For a non-significant two-way interaction, one need to determine whether there is any statistically significant main effects from the ANOVA output.

References

Davis, C. S. (2009). Statistical Methods for the Analysis of Repeated Measurements. New York: Springer.

Hand, D. J., Crowder, M. J. (2017). Analysis of Repeated Measures. United States: CRC Press.

<https://cran.r-project.org/web/packages/rstatix/index.html>

<https://cran.r-project.org/web/packages/datarium/index.html>

R Core Team (2019). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. URL <https://www.R-project.org/>.

Group of Experiments

Susheel Sarkar

ICAR-Indian Agricultural Statistical Research Institute, New Delhi, 110012

Email: sarkar82@gmail.com

Introduction

In extensive experimental programs, it is essential to conduct multiple trials of a specific set of treatments, such as different varieties or fertilizers, across various locations or seasons. The locations chosen for these repeated trials are typically experimental stations situated within the region. The purpose of repeating the trials is to examine how the effects of treatments vary with different locations. More broadly, the goal of such repetitions is to identify which treatments are appropriate for specific areas, which is why the tests are conducted simultaneously at a carefully chosen range of sites.

Additionally, the aim of the research conducted at experimental stations is to develop recommendations for practitioners that encompass a broadly extensive population, whether in terms of space, time, or both. As a result, it is essential to verify that the outcomes derived from the research are applicable to multiple locations in the future and across a reasonably diverse geographical area.

An individual experiment will provide precise insights solely about the specific location where it takes place and the time of year in which it is conducted. Therefore, it has become standard practice to replicate experiments at various locations or over multiple time periods to arrive at reliable recommendations that consider variations from place to place or over time, or both. In instances of repeated experiments, suitable statistical methods for a joint analysis of the data must be adhered to, alongside the analysis of each individual experiment based on their specific objectives. In the combined data analysis, the primary areas of interest would be

- i) to estimate the average response to specified treatments and
- ii) to evaluate the consistency of responses across different places or occasions, meaning the interaction of treatment effects with specific locations or years.

The utility and importance of average response estimates rely on whether the response remains consistent across different locations or varies, which means it depends on the presence or absence of interaction.

The results of a set of trials may, therefore, be considered as belonging to one of the four categories:

- i) the experimental errors are homogeneous and there is no interaction,
- ii) the experimental errors are homogeneous and there is an interaction,
- iii) the experimental errors are heterogeneous and there is no interaction, and
- iv) the experimental errors are heterogeneous and there is an interaction.

The meaningfulness of average estimates of treatment responses would therefore, depend largely upon the absence or presence of this interaction analysis.

Analysis Procedure

For combined analysis or analysis for groups of Experiments following steps are to be followed

Step I: Construct an outline of combined analysis of variance over years or for places or environment, based on the basic design used. For example, the data of grain yield for four places, four treatments each treatment replicated five times is given in Table-1.

Step II: Perform usual Analysis of variance for the given data. Here the experiment conducted is in randomized complete block design. So, perform analysis of four places separately for the four places. This may be done either in SAS, SPSS or EXCEL software.

Step III: We have p error mean squares that belongs to p RBD conducted and we have to test the homogeneity of variances. Now we have following two situations:

Situation I: When $p = 2$ In this situation, we apply F-test to assess the homogeneity of variances. In this context, the null and alternate hypothesis are $H_0: \sigma_1^2 = \sigma_2^2$ and $H_0: \sigma_1^2 \neq \sigma_2^2$. Let Se_1^2 and Se_2^2 are the mean square errors (mse) for the two places. Then the value of F statistics will be Se_1^2/Se_2^2 and this value will be tested against the Table F value at n_1 and n_2 degrees of freedom at 5 % level of significance, where n_1 and n_2 are degrees of freedom (df) for error for the two places, respectively. If the computed F value exceeds the tabulated F value, then we reject the null hypothesis of homogeneity of variance, indicating that the data is heterogeneous across different locations; otherwise, it is homogeneous.

Situation II. When $p > 2$

In this situation, we apply Bartlett's Chi-square test. Here null and alternate hypothesis are

$H_0: \sigma_1^2 = \sigma_2^2 = \dots = \sigma_p^2$ against the alternate hypothesis H_1 : at least two σ_i^2 's are not equal, where σ_i^2 is the error variance of i^{th} year/location. Let $SE_1^2, SE_2^2, \dots, SE_p^2$ are the mse of p years respectively and n_1, n_2, \dots, n_p are the df for p years respectively.

Then the test statistics for testing the homogeneity of variances is

$$\chi_{p-1}^2 = \frac{\sum n_i \log \bar{s}_e^2 - \sum n_i \log s_{ei}^2}{1 + \frac{1}{3(p-1)} \left(\sum \frac{1}{n_i} - \frac{1}{\sum n_i} \right)}$$

Where $\bar{s}_e^2 = \frac{\sum n_i s_{ei}^2}{\sum n_i}$

If $n_i = n$

$$\chi_{p-1}^2 = \frac{n[\sum p \log \bar{s}_e^2 - \sum \log s_{ei}^2]}{1 + \frac{(p+1)}{3np}}$$

Where χ_{p-1}^2 follows chi-square distribution with $p-1$ df.

If the computed value of χ_{p-1}^2 exceeds the tabulated t value of χ_{p-1}^2 for $p-1$ degrees of freedom, the null hypothesis of variance homogeneity is rejected, indicating that the data is heterogeneous across different years; otherwise, it is considered homogeneous.

Step IV: If the variances of errors are not equal, then a combined analysis using weighted least squares is necessary, with weights being the inverses of the root mean square error. The weighted analysis is done by defining a new variable as $\text{newres} = \text{res} / \text{root mean square}$. This transformation is akin to Aitken's transformation. As a result, this new variable is homogeneous, allowing for combined analysis of variance to be conducted on it. If the error variances are equal, there is no need for data transformation.

Step V: Now, one can interpret the groups of experiments as a nested design with multiple factors embedded within one another. The various locations are considered as major blocks, with the experiments organized within these. Consequently, the comprehensive analysis of

data can be approached as a nested design. For the analysis, the replication-based data of treatments at each location offers valuable insights. A benefit of this analysis is that it allows for a further decrease in the sum of squares for error since an additional source of variability is extracted from the experimental error, thereby minimizing the experimental error. This might also result in a decrease in the coefficient of variation (CV).

Step VI: The subsequent step in the analysis is to evaluate the significance of the interaction between place and treatment. The question regarding the significance of the place treatment interaction, meaning whether the differences between treatments vary across different places, can be addressed by comparing the mean square for place treatment to the estimate of error variance using an F-test. If the mean square is deemed non-significant, it indicates that the interaction is not present. If we assume that this interaction does not exist, we can combine the sum of squares for treatment places with the error sum of squares to obtain a more accurate estimate of error for assessing the significance of treatment differences. Conversely, if the interaction is significant i.e., treatment effects differ across places, the suitable mean square for testing the significance of treatments is the mean square resulting from place treatment.

SAS code for group of experiments

```
data suman_MCM;
input Season$ rep trt PH NT PN PL FLL PW SPY KL KW KLKW TW IBR ZBR IPR
ZPR;
cards;
WS17 1 1 162.3 8.0 8.0 32.0 29.0 3.0 23.1 2.0 0.6 3.5 9.6 13.4 20.3 6.6 14.4
WS17 1 2 162.0 9.0 9.0 37.0 42.3 3.2 25.6 2.3 0.7 3.2 12.8 13.4 16.5 5.7 11.6
WS17 1 3 151.0 10.0 10.0 27.6 25.0 3.9 38.9 2.4 0.7 3.2 13.1 14.4 20.3 6.9 15.5
WS17 1 4 170.0 12.0 11.0 33.0 28.0 3.0 30.6 2.1 0.6 3.6 9.8 13.0 19.8 6.5 14.5
WS17 1 5 158.3 4.0 4.0 33.0 46.3 1.9 8.1 2.4 0.7 3.4 12.5 14.3 19.5 6.1 13.2
WS17 1 6 154.0 6.0 6.0 36.3 33.7 3.5 18.9 2.3 0.6 3.6 9.9 12.5 16.6 6.6 11.3
WS17 1 7 137.7 8.0 8.0 30.3 39.0 3.4 27.4 2.4 0.7 3.4 15.4 12.7 16.4 6.4 11.0
WS17 1 8 148.7 7.0 7.0 37.0 16.3 5.2 34.5 3.3 0.7 4.6 18.6 12.3 14.9 4.8 10.4
WS17 1 9 165.7 12.0 12.0 33.0 36.7 2.5 30.1 2.5 0.7 3.8 12.5 12.9 20.3 5.3 14.2
WS17 1 10 150.3 8.0 8.0 37.0 30.3 3.8 27.9 2.2 0.7 3.1 9.9 13.0 16.0 6.3 12.3
WS17 1 11 158.3 9.0 9.0 34.3 31.0 2.9 25.4 3.1 0.8 3.8 16.4 12.7 16.3 5.5 11.4
.
.
.
```

```

ods rtf file="SPY.rtf" style=journal;
proc sort;
by Season;
run;
ods trace on;
ods output overallanova=MSerror;
ods output LSmeans=lsmean;
proc glm data = suman_MCM;
class rep trt;
model SPY=rep trt;
means trt/lsd;
by Season;
quit;
ods output close;
ods output close;
ods trace off;
data required;
set MSerror(where=(source='Error') keep=Season source df ms);
run;
proc iml;
use required;
read all into a; /* use error variances in m1 variable*/
*a =m1[2:nrow(m1),ncol(m1)-1:ncol(m1)];/*from m1 extract variances and number of
observations */
v =0;ct = 0;nchi = 0;St = 0;

do i = 1 to nrow(a);          /* computing pooled variance */
    St = St + (a[i,1]-1)*a[i,2];
    v = v + (a[i,1]-1);
    ct = ct + 1/(a[i,1]-1);
end;
S = St/v;
dchi = (1 + (1/(3*(nrow(a)-1)))(ct-(1/v))); /*computing denominator of Bartlett's chi-
square statistic/
do i = 1 to nrow(a);
nchi = nchi + (a[i,1]-1)*(log(S/a[i,2]));
end;
chi = nchi/dchi;probability = 1 - probchi(chi,(nrow(a)-1));/*computing chi-square test
statistic and probability./
df = (nrow(a)-1);

```

```

print probability chi df S; /* printing chi-square test statistic value, probability and degree
of freedom*/
if probability >= 0.05 then Interpretation = "Data is Homogeneous at 5% level of
Significance";
else Interpretation = "Data is Heterogeneous at 5% level of Significance";
print Interpretation; /* testing and printing interpretation*/
pb = char(probability);
ods html body = 'mse.xls';
proc print data = required;
var Season ms;
run;
ods html close;
data transformed; /* This set of SAS statements transforms the data*/
set suman_MCM;
if Season="WS17" then
new_var=SPY/sqrt(0.0960354);
if Season="WS18" then
new_var=SPY/sqrt(2.8010354);
if Season="WS19" then
new_var=SPY/sqrt(1.8184596);
run;
proc glm data = transformed;
class Season rep trt;
model new_var SPY= Season rep(Season) trt Season*trt;
means Season trt/lsd;
run;
ods rtf close ;

```

Suggested Readings

https://www.researchgate.net/publication/384019558_Analysis_of_Experimental_Designs_using_SAS

Cochran, W. G., & Cox, G. M. (1957). Experimental designs (2nd ed.). John Wiley & Sons.

Gomez, K. A., & Gomez, A. A. (1984). Statistical procedures for agricultural research (2nd ed.). John Wiley & Sons.

Mead, R., Curnow, R. N., & Hasted, A. M. (2017). Statistical methods in agriculture and experimental biology (3rd ed.). CRC Press.

Montgomery, D. C. (2020). Design and analysis of experiments (10th ed.). John Wiley & Sons.

Steel, R. G. D., Torrie, J. H., & Dickey, D. A. (1997). Principles and procedures of statistics: A biometrical approach (3rd ed.). McGraw-Hill.

Analysis of Incomplete Block Designs

Manjunatha G R

Central Silk Board, Bengaluru – 560068

Email: mgr.dvg@gmail.com

When there are many treatments or complete blocks are unsuitable, incomplete block designs are useful. These were introduced by Yates to reduce variability better than randomized blocks or Latin squares. Precision of treatment effect increases with more replications. Similarly, comparing two treatments is more precise if they occur together more often. To ensure equal precision, treatments are placed so that:

- Each occurs once per block,
- Has equal replications, and
- Every pair occurs together equally or nearly equally.

When all treatment pairs occur together an equal number of times, the design is referred to as a Balanced Incomplete Block (BIB) design. If the frequency of pairwise replications differs, it is known as a Partially Balanced Incomplete Block (PBIB) design. These concepts are explained in detail by Cochran & Cox (1957). Designs such as Completely Randomized Design (CRD) and Randomized Complete Block Design (RCBD/RBD) are categorized as complete block designs. We will now focus on the balanced incomplete block design (BIBD) and the partially balanced incomplete block design (PBIBD), which fall under the category of incomplete block designs.

1. BALANCED INCOMPLETE BLOCK DESIGN (BIBD)

Balanced Incomplete Block (BIB) designs are highly efficient among binary incomplete block designs. However, they often need many replications and are not possible for all parameter combinations. Yates introduced them to better control heterogeneity than randomized blocks or Latin squares, especially with many treatments (Yates & Mather, 1963). The precision of estimating a treatment effect increases with more replications. Similarly, estimating the difference between two treatments is more precise if the pair appears together more often. This has been supported by Nigam & Gupta (1979), Pearce (1983), Federer (1985), and Montgomery (2012).

Balanced Incomplete Block (BIB) Designs: A BIB design is an arrangement of v treatments in b blocks each of size k ($k < v$) such that

- Each treatment occurs at most once in a block
- Each treatment occurs in exactly r blocks
- Each pair of treatments occurs together in exactly λ blocks.

Five parameters denote such design $D(b, k, v, r, \lambda)$.

The parameters b, k, v, r and λ are not chosen arbitrarily.

They satisfy the following relations:

- (i) $bk = vr$
- (ii) $\lambda(v-1) = r(k-1)$
- (iii) $b \geq v$ (and hence $r > k$)

A BIB design for $v = b = 7, r = k = 3$ and $\lambda = 1$ in the following:

1	3	7
2	4	1
3	5	2
4	6	3
5	7	4
6	1	5
7	2	6

Example of BIBD

Blocks	Treatments
B_1	T_1, T_2, T_5
B_2	T_1, T_2, T_6
B_3	T_1, T_3, T_4
B_4	T_1, T_3, T_6
B_5	T_1, T_4, T_5
B_6	T_2, T_3, T_4
B_7	T_2, T_3, T_5
B_8	T_2, T_4, T_6
B_9	T_3, T_5, T_6
B_{10}	T_4, T_5, T_6

Now we see how the conditions of BIBD are satisfied.

I. $bk=10*3 = 30$ and $vr= 6*5=30$

therefore $bk=vr$

II. $\lambda (v-1) = 2*5=10$ and $r (k-1) = 5*2 = 10$

Therefore $\lambda (v-1) = r (k-1)$

III. $b = 10 \geq 6$

Even if the parameters satisfy the relations, it is not always possible to arrange the treatments in blocks to get the corresponding design. The necessary and sufficient conditions to be satisfied by the parameters for the existence of a BIBD are not known.

Construction of BIB Design

Two Latin squares are pairwise orthogonal if, when superimposed, each symbol from one square pairs exactly once with each symbol from the other. Three or more Latin squares are mutually orthogonal if every pair of them is orthogonal. A full set of $s - 1$ mutually orthogonal Latin squares exists when $s = p^n$, where p is a prime. (Fisher and Yates, 1963)

To construct a BIB design using MOLS:

- Arrange $v = s^2$ treatments in an $s \times s$ array.
- Create s blocks by taking each row as a block.
- Create another s blocks by taking each column as a block.
- Superimpose a Latin square over the array. For each symbol in the Latin square, form a block with all treatments that share that symbol.
- This gives another s blocks.

Thus, each Latin square adds s more blocks, and the design becomes a BIB.

The **complementary design** of a BIB design is also a BIB design. For example, a BIB design with parameters: $v = 9$, $b = 12$, $r = 8$, $k = 6$, $\lambda = 5$ has a complementary BIB design with the same number of treatments ($v = 9$) but different values for the other parameters.

In general, if a BIB design has parameters: v , b , r , k , λ , then its complementary design has parameters:

- $v' = v$
- $b' = b$
- $r' = b - r$
- $k' = v - k$
- $\lambda' = b - 2r + \lambda$

A **symmetric BIB design** is one where, $v = b$ (number of treatments = number of blocks), or $r = k$ (number of replications = block size).

In such designs, any two blocks share exactly λ treatments and example with parameters $v = b = 7$, $r = k = 3$, and $\lambda = 1$.

A BIB design is called **α -resolvable** if its blocks can be grouped into t groups of m blocks each, such that “each treatment appears exactly α times in every group”.

If, in addition:

- Any two blocks within a group share q_1 treatments, and
- Any two blocks from different groups share q_2 treatments, then the design is called an **affine α -resolvable** BIB design.

The **dual of a BIB design** is formed by interchanging treatments and blocks. If the original design has parameters: v , b , r , k , λ , then the dual design has parameters: $v' = b$, $b' = v$, $r' = k$, $k' = r$.

Note: The dual is not always a BIB design. However, if the original is a symmetric BIB design, then the dual is also a BIB design.

Application of BIBD:

Application Area	Example	Purpose
Crop Variety Trials	Comparing 10 rice varieties in 5 blocks with limited land	Controls variability due to field heterogeneity
Fertilizer Trials	Testing 6 types of fertilizers on maize	Reduces the number of plots needed while maintaining balance
Pest/Disease Management Studies	Evaluating multiple pest control methods	Ensures fair comparison by equal pairing frequency
Multilocal Trials	Uniform testing across sites with limited replications per location	Useful when resources or land size constrain full replication

2. PARTIALLY BALANCED INCOMPLETE BLOCK DESIGN (PBIBD)

The balanced incomplete block designs have many advantages. They are connected designs as well as the block sizes are also equal. A restriction on using the BIBD is that they are not available for all parameter combinations. They exist only for certain parameters. Sometimes, they require a large number of replications also. This hampers the utility of the BIBDs. For example, if there are $v = 8$ treatments and block size is $k = 3$ (i.e., 3 plots in each block) then the total number of required blocks are $= 56$ and so using the relationship $bk \leq vr$, the total number of required replicates is .

Another key characteristic of the BIBD is its balanced efficiency. This indicates that the estimations of all treatment differences are achieved with equal precision. The partially balanced incomplete block designs (PBIBD) make some concessions on this aspect while aiding in the reduction of replication numbers. In simpler terms, pairs of treatments can be organized into various sets so that the accuracy of the estimated differences in treatment effects for all pairs within a set remains consistent. The partially balanced incomplete block designs remain connected like BIBD but no more balanced. Rather they are partially balanced in the sense that some pairs of treatments have the same efficiency whereas some other pairs of treatments have the same efficiency but different from the efficiency of earlier pairs of treatments. This will be illustrated more clearly in the further discussion.

Partially Balanced Association Schemes

A relationship that meets the following three criteria is referred to as a partially balanced association scheme with m associate classes.

- I. Any two symbols can be classified as first, second,..., or m^{th} associates and the relationship among these associations is symmetrical; that is, if treatment A is the i^{th} associate of treatment B, then treatment B is also the i^{th} associate of treatment A.
- II. Each treatment A within the set has a fixed number n_i of treatments within the set that are classified as the i^{th} associate, and this number n_i (for $i = 1, 2, \dots, m$) remains constant regardless of treatment A.
- III. If treatments A and B are i^{th} associates, the total number of treatments that are both the i^{th} associate of A and the k^{th} associate of B is consistent and does not depend on the specific pair of i^{th} associates A and B.

PBIBDs with two associate classes are commonly used in practical scenarios and can be categorized into various types based on the association scheme.

1. Triangular
2. Group divisible
3. Latin square with i constraints
4. Cyclic and
5. Singly – linked blocks.

General Theory of PBIBD

A PBIBD with m associate classes is characterized as follows. Assume there are v treatments available. Let b represent the number of blocks, and each block has k plots, meaning there are k plots in every block. The treatments are organized into b blocks in accordance with an m -associate partially balanced association scheme that adheres to the following conditions:

- (a) every treatment appears at most once per block,
- (b) every treatment is represented exactly in r blocks and

- (c) if two treatments are the i^{th} associates to one another, then they included together in exactly in λ_i ($i = 1, 2, \dots, m$) blocks.

The number λ_i is consistent regardless of the specific pair of i^{th} associates selected. It is not necessary for the values of λ_i to be unique, as some of them λ_i 's can be zero.

The parameters $b, v, r, k, \lambda_1, \lambda_2, \dots, \lambda_m, n_1, n_2, \dots, n_m$ are referred to as the first kind parameters, while the second kind parameters are termed as such. It should be noted that all of the design is derived from the association scheme under consideration. Only λ included in the definition of PBIBD.

If λ_i for all $i = 1, 2, \dots, m$ then PBIBD reduces to BIBD. So BIBD is essentially a PBIBD with one associate class.

Conditions of PBIBD

- **v:** Number of treatments
- **b:** Number of blocks
- **k:** Number of treatments per block
- **r:** Number of times each treatment appears across all blocks

$\lambda_1, \lambda_2, \dots$: Association numbers,

The parameters of a PBIBD are chosen such that they satisfy the following relations:

1. $bk = vr$
2. $\sum_{i=1}^m n_i = v - 1$
3. $\sum_{i=1}^m n_i \lambda_i = r(k - 1)$
4. $n p^k = n p^i = n p^j$

Example

Suppose we have **6 treatments** labelled **A, B, C, D, E, and F**, and we need to arrange them into **4 blocks**, with **3 treatments per block**.

Block	Treatments
1	A, B, C
2	A, D, E
3	B, D, F
4	C, E, F

Here:

- Each treatment appears in **two blocks** (so $r=2r = 2$).
- Some pairs appear **more frequently than others**:
 - **Highly associated pairs ($\lambda_1 = 1$)**: (A, B), (B, C), (A, C), (D, E), (E, F), (D, F) – These appear together in only one block.
 - **Less associated pairs ($\lambda_2 = 0$)**: (A, F), (B, E), (C, D) - These pairs **never appear together in any block**.

This is a **two-associate class PBIBD**, where some pairs occur together more frequently, and others never occur together.

Applications of PBIBD

Area	Example	Purpose
Genotype Trials	Comparing 50 genotypes of wheat	PBIBD allows comparison with more treatments than BIBD
Heterogeneous Fields	Trials on irregular terrain or fertility zones	Flexibility in blocking based on partial treatment relationships
Long-term Agronomy Trials	Rotational cropping systems where all combinations aren't feasible	PBIBD accommodates partial comparisons over time

PBIBD is useful in situations where:

- Some treatments require **more frequent comparisons** (e.g., standard drug vs. new drugs in clinical trials).

- Some treatments should **never be tested together** (e.g., chemical compounds in industrial experiments).
- There is a need for **more flexibility** compared to a standard BIBD.

This makes PBIBD a practical choice when perfect balance is impossible, but some level of structured comparison is still needed.

R code

```
install.packages("agricolae") # Run only if not installed

library(agricolae)

# Define Parameters

treatments <- 6 # Number of treatments

blocks <- 10 # Number of blocks

k <- 3 # Treatments per block

# Generate the BIBD Design

bibd_design <- design.bib(trt = treatments, k = k, r = NULL, seed = 123)

# View Design

print(bibd_design$book)

# Convert design to dataframe

data <- bibd_design$book

# Add simulated response values

set.seed(123)

data$response <- rnorm(nrow(data), mean = 50, sd = 10)

# View Data

head(data)

# Fit ANOVA Model for BIBD

bibd_model <- aov(response ~ block + treatment, data = data)

# ANOVA Summary Table

summary(bibd_model)
```

```

# Boxplot of Treatment Effects

boxplot(response ~ treatment, data = data, col = rainbow(6),
        main = "BIBD - Treatment Effects", xlab = "Treatment", ylab = "Response")

### PBIBD ###

install.packages("PBIBD") # Run only if not installed

library(PBIBD)

# Define PBIBD Parameters

treatments <- 7 # Number of treatments

blocks <- 7     # Number of blocks

k <- 3          # Treatments per block

lambda <- c(1, 2) # Association scheme

# Generate PBIBD

pbibd_design <- PBIBD(treatments, blocks, k, lambda)

# View Design

print(pbibd_design)

# Convert design to dataframe

data <- pbibd_design$Design

# Add simulated response values

set.seed(123)

data$response <- rnorm(nrow(data), mean = 50, sd = 10)

# View Data

head(data)

# Fit ANOVA Model for PBIBD

pbibd_model <- aov(response ~ Block + Treatment, data = data)

# ANOVA Summary Table

summary(pbibd_model)

# Boxplot of Treatment Effects

boxplot(response ~ Treatment, data = data, col = rainbow(7),

```

```
main = "PBIBD - Treatment Effects", xlab = "Treatment", ylab = "Response")
```

3. LATTICE DESIGN ANALYSIS (LSD)

Lattice designs are a type of **incomplete block design** used in agricultural, biological, and industrial experiments when there are a large number of treatments. These designs help in improving precision while reducing variability.

Yates (1936) introduced *lattice designs*, a class of resolvable incomplete block designs developed primarily for large-scale agricultural experiments. These include some BIBDs and PBIBDs, but not all lattice designs are balanced.

Lattice designs are useful when:

- A large number of treatments must be tested.
- Full replication (as in BIBDs) is impractical due to resource constraints.
- Flexible replication is needed (e.g., 2 replications = simple lattice, 3 = triple lattice, m = m -ple lattice).

Lattice designs are characterized by grouping incomplete blocks into replications. They are particularly suited for variety trials in plant breeding and agronomy.

Despite some limitations on permissible values of treatments and block sizes, lattice designs remain valuable tools in experimental agriculture—especially where v or $t > 100$, as in crop variety or hybrid evaluations.

Why Use Lattice Designs?

- When the number of treatments (t) is large, a Randomized Complete Block Design (RCBD) may not be feasible due to large block sizes.
- Lattice designs help reduce the size of blocks while maintaining control over variability.
- They improve efficiency by allowing more precise comparisons of treatments.

Basic Properties

- Treatments (t or v) are arranged in small blocks.

- Each treatment appears once per block.
- Each treatment is replicated across blocks to ensure statistical validity.

Types of Lattice Designs

(i) Simple Lattice Design

- Used when the number of treatments (t) is a perfect square ($t = k^2$).
- The experiment consists of k replications and k blocks per replication, each containing k treatments.

(ii) Triple Lattice Design

- Used when treatments are replicated three times.
- Treatments are grouped into blocks of size k, but each treatment appears in three different blocks.

(iii) Balanced Lattice Design

- A more balanced version of lattice designs where each treatment appears the same number of times across replications.
- Assume we have 9 treatments and 3 replications, with 3 treatments per block.

Replication	Block 1	Block 2	Block 3
Rep 1	T1, T2, T3	T4, T5, T6	T7, T8, T9
Rep 2	T1, T4, T7	T2, T5, T8	T3, T6, T9
Rep 3	T1, T5, T9	T2, T6, T7	T3, T4, T8

The general statistical model is:

$$Y_{ijk} = \mu + \tau_i + r_k + b_j(k) + e_{ijk}$$

where:

- Y_{ijk} = observed response for treatment ii in block jj within replication kk,
- μ = overall mean,

- = effect of treatment i ,
- = replication effect,
- = block effect within replication k ,
- = random error ($\sim N(0, \sigma^2)$)

Assumptions

1. Errors are independently and normally distributed ($\sim N(0, \sigma^2)$).
2. Block effects are random.
3. Treatments are fixed.

ANOVA Table for Lattice Design

Source of Variation	DF	SS	MS	F-ratio
Replication	$r-1$	SSR	MSR	MSR/ MSE
Blocks within Replication	$r(b-1)$	SSB	MSB	MSB/MSE
Treatments	$t-1$	SST	MST	MST/MSE
Error	DFE	SSE	MSE	
Total	$N-1$	SSTot		

Adjusting for Block Effects

- Since blocks are incomplete, we adjust treatment means using intra-block analysis.
- Block effects are treated as random and estimated separately.

Treatment Mean Comparisons

After ANOVA, if treatments are significant, post-hoc comparisons are used:

- Least Significant Difference (LSD)
- Tukey's HSD
- Duncan's Multiple Range Test (DMRT)

Advantages

- Reduces block size for better control of variability.
- More efficient than Randomized Complete Block Design (RCBD).
- Allows multiple replications to improve accuracy.

Disadvantages

- Requires specialized analysis due to incomplete blocks.
- More complex randomization than RCBD.
- May require adjustments for block effects.

Example:

The following table gives the synthetic yields per plot of an experiment conducted with $3^2 = 9$ treatments using simple lattice designs.

Replication 1			
Blocks	Treatments (yield per plot)		
1	1(8)	7(5)	4(3)
2	3(3)	6(2)	9(6)
3	8(3)	5(7)	2(3)

Replication 2			
Blocks	Treatments (yield per plot)		
4	8(2)	7(2)	9(7)
5	4(3)	5(3)	6(3)
6	2(2)	3(4)	1(6)

Analysis

Compute

$$\text{Grand total}(G) = 8+5+\dots+6 = 72$$

$$\text{No. of observation (n)} = 18$$

$$\text{Grand Mean } (\bar{y}) = G/n = 72/18 = 4$$

$$\text{No. of replications} = 2$$

$$\text{Block Size}(k) = 3$$

$$\text{Correction factor CF} = \frac{G^2}{n} = 288$$

$$\text{Total S.S. (TSS)} =$$

$$= 8^2 + 5^2 + \dots + 6^2 - 288$$

$$= 66$$

$$\begin{aligned}\text{Treatment S.S. unadjusted (SSTreat)} &= \\ &= (14^2 + \dots + 13^2) / 2 - 288 \\ &= 49\end{aligned}$$

$$\begin{aligned}\text{Block S.S. unadjusted (SSBloc)} &= \\ &= (16^2 + \dots + 12^2) / 3 - 288 \\ &= 9.33\end{aligned}$$

$$\begin{aligned}\text{Treatment S.S. adjusted (SSTreat)} &= \\ &= 51.44\end{aligned}$$

$$\begin{aligned}\text{Block S.S adjusted (SSB)} &= \text{SSTa} + \text{SSBu} - \text{SSTu} \\ &= 51.44 + 9.33 - 49.00 \\ &= 11.77\end{aligned}$$

$$\begin{aligned}\text{Error S.S. (SSE)} &= \text{TSS} - \text{SSBu} - \text{SSTa} \\ &= 66 - 51.44 - 9.33 \\ &= 5.23\end{aligned}$$

The ANOVA table is given as

Source	d.f	S.S.	M.S.	F
Blocks(unadj)	5	9.33		
Treatments(adj)	8	51.44	6.43	4.91
Blocks(adj)	5	11.77	2.35	1.79
Treatments(unadj)	8	49.00		
Error	4	5.23	1.31	
Total	17	66.00		

Treatments effects are not significantly different

$$\begin{aligned}\text{SE}(1) &= \\ &= \\ &= 1.32, \text{ if the treatments belongs to same block}\end{aligned}$$

$$\begin{aligned}\text{CD}(1) &= t(.05, 4) * \text{SE}(1) \\ &= 2.776 * 1.32 \\ &= 3.66\end{aligned}$$

$$\begin{aligned}\text{SE}(2) &= \\ &= \\ &= 1.477, \text{ if the treatments belongs to same block}\end{aligned}$$

$$\text{CD}(2) = t(.05, 4) * \text{SE}(2)$$

$$= 2.776 \times 1.32$$

$$= 4.100.$$

R code

```
install.packages("agricolae") # Run only if not installed
library(agricolae)
# Define number of treatments
treatments <- 9 # Must be a perfect square (e.g., 9, 16, 25)
# Generate Lattice Design
lattice_design <- design.lattice(trt = treatments, r = 3, seed = 123)
# View Design
print(lattice_design)
# Convert design to dataframe
data <- lattice_design$book
# Add simulated response values
set.seed(123)
data$response <- rnorm(nrow(data), mean = 50, sd = 10)
# View Data
head(data)
# Fit ANOVA Model for Lattice Design
lattice_model <- aov(response ~ block + treatment, data = data)
# ANOVA Summary Table
summary(lattice_model)
# Boxplot of Treatment Effects
boxplot(response ~ treatment, data = data, col = rainbow(9),
        main = "Lattice Design - Treatment Effects", xlab = "Treatment", ylab = "Response")
```

References

- Bose, R. C., and Nair, K. R. (1939). Partially balanced incomplete block designs. *Sankhya: The Indian Journal of Statistics*, 4, 337-372.
- Yates, F., and Mather, K. (1963). *Ronald Aylmer Fisher, 1890-1962*. The Royal Society London.

Response Surface Methodology for Optimization in Climate Change Studies

Eldho Varghese

ICAR-Central Marine Fisheries Research Institute

Email: eldho.varghese@icar.org.in

1. Introduction

In product and process optimization, the conventional One-Factor-At-a-Time (OFAT) method evaluates only one variable at a time while keeping others fixed, which fails to account for interaction effects and leads to subpar optimization. Conversely, factorial designs enable the identification of both significant factors and notable interactions among them with fewer tests compared to OFAT, yet they still do not predict the optimal factor level settings needed to achieve the desired outcome (minimum/maximum/target responses) within the experimental range. The drawbacks of these traditional methods are addressed by concurrently optimizing all influencing variables through Response Surface Methodology (RSM), which was introduced by Box and Wilson in 1951. RSM facilitates the exploration of the functional relationships between one or more response variables and a group of experimental variables or factors. These techniques are typically applied after identifying a “vital few” controllable factors to ascertain the factor settings that will optimize the response. Designs of this nature are generally selected when there is an expectation of curvature in the response surface.

RSM thus comprises a series of techniques that involve (i) establishing an experiment (designing an experiment) that can provide sufficient and dependable estimates of the response of interest, (ii) identifying a model that best represents the data gathered from the chosen design by performing suitable tests of hypotheses related to the model’s parameters, and (iii) finding the optimal conditions of the experimental factors that yield the highest (or lowest) value of the response.

RSM has numerous applications in the development, enhancement, and optimization of processes across various research areas, including agricultural studies, food science and technology, biological sciences, fisheries, biochemistry, analytical chemistry, and engineering, etc.

In recent years, RSM has also gained prominence in climate change research and agro-environmental modelling, where it is used to simulate, optimize, and quantify the effects of complex interactions among climatic variables such as temperature, precipitation, relative humidity, and CO₂ concentration. For example, RSM has been employed to assess crop response to varying temperature and water regimes, estimate greenhouse gas emissions under different land management scenarios, and design stress simulation experiments in controlled environments. Its ability to model nonlinear relationships and interactions makes it particularly suitable for climate-related studies where environmental variables rarely act independently.

Example 1: Overuse of nitrogen (N) compared to phosphorus (P) and potassium (K) presents serious agronomic and environmental challenges, especially under climate change conditions that alter nutrient uptake and soil dynamics. As temperature and rainfall patterns shift, the efficiency of nutrient application becomes more variable, yet farmers often rely heavily on nitrogenous fertilizers due to their immediate crop response and availability, while P and K remain underutilized. This imbalance not only reduces long-term soil fertility but also increases risks of greenhouse gas emissions and nutrient runoff. Traditional approaches that estimate optimal doses for N, P, and K individually do not account for nutrient interactions or climate-induced variability, leading to suboptimal recommendations. In contrast, Response Surface Methodology (RSM) enables simultaneous optimization of these inputs by capturing nonlinear relationships and interactions among nutrients and environmental factors. This method supports the development of balanced, source-specific fertilizer strategies tailored to changing climatic conditions, thereby enhancing both productivity and sustainability in crop production systems.

Example 2: Climate change is expected to significantly alter crop germination and early growth phases due to increased variability in temperature and soil moisture. Controlled environment experiments are commonly used to assess how seed germination responds to such stresses. The goal of this type of experiment is to identify the optimal combination of environmental conditions (e.g., temperature, soil moisture, and relative humidity) that supports maximum germination rate or seedling vigor. To assess the combined effects of air temperature, soil moisture, and relative humidity on the germination of chickpea (*Cicer arietinum*) in a growth chamber with the following levels selected based on realistic agro-climatic ranges:

	Factors	Levels
1.	Air Temperature (°C)	15 ⁰ C, 20 ⁰ C, 25 ⁰ C, 30 ⁰ C and 35 ⁰ C
2.	Soil Moisture (% field capacity)	30%, 50%, 70%, 90% and 110%
3.	Relative Humidity (%)	40%, 50%, 60%, 70%, 80%

In this scenario, response surface methodologies for three variables, each at five evenly spaced levels, can be utilized.

Example 3: Investigations into food processing are being conducted to enhance the value of agricultural products. The primary aim of these investigations is for the researcher to discover the optimal combination of values for various parameters that are crucial for the product. Specifically, let's say you're performing an experiment on the osmotic dehydration of banana slices to determine the ideal mix of sugar solution concentration, solution-to-sample ratio, and osmosis temperature. Below are the levels for the different factors:

	Factors	Levels
1.	Concentration of sugar solution	40%, 50%, 60%, 70% and 80%
2.	Solution to sample ratio	1:1, 3:1, 5:1, 7:1 and 9:1
3.	Temperature of osmosis	25 ⁰ C, 35 ⁰ C, 45 ⁰ C, 55 ⁰ C and 65 ⁰ C

In this scenario, response surface designs with three variables, each set at five evenly spaced levels, can be employed.

Example 4: In climate-sensitive aquaculture, optimizing fish culture conditions is essential to maintain productivity under increasing environmental variability. For instance, the larval development of Genetically Improved Farmed Tilapia (GIFT) is highly sensitive to changes in water salinity and temperature, both of which are directly influenced by climate change. Rising sea levels and coastal intrusion increase salinity in freshwater systems, while elevated temperatures affect metabolic and growth rates. To identify the optimal combination of salinity and temperature for GIFT larval rearing under such changing conditions, Response Surface Methodology (RSM) using a two-factor Central Composite Design can be effectively employed. This design allows the modeling of nonlinear responses and interaction effects, helping to develop robust aquaculture practices that are resilient to climate-induced stressors.

Example 5: In the field of analytical chemistry, anthocyanins (ACNs) have arisen as promising nutraceutical components for the creation of functional foods and dietary supplements. To increase the concentration of anthocyanins, it is essential to optimize enzyme-assisted

processing by focusing on specific levels of various factors. This can be accomplished through the use of response surface methodology, utilizing a three-level Box-Behnken design.

	Factors	Levels
1.	Enzyme Concentration (%)	0.05, 0.15 and 0.25
2.	Temperature (°C)	50, 60 and 70
3.	Time (minutes)	30, 60 and 90

2. Stages of Response Surface Methodology (RSM)

1. Fixing the objective of the study.
2. Screening phase or Screening Experiment involves selecting significant independent variables through the use of first-order response surface designs such as 2^v factorial designs (FD), fractional replicates of the 2^v factorial designs (FFD), Simplex designs, Plackett-Burman designs (PB), Definitive screening designs (DSD), and custom designs.
3. Regression modelling: For regression modeling, the regression equation is constructed using the effect terms that demonstrate statistical significance regarding the response. If the response is accurately modeled by a linear term of the independent variables, the corresponding function is a first-order model. However, if the model indicates a significant lack of fit (as assessed by ANOVA), then a first-order model would be insufficient; therefore, a polynomial of a higher degree (second-order design) should be utilized. First-order models are generally applied in screening experiments.
4. Experimentation using response surface design: When conducting experimentation with response surface designs, the appropriate second-order rotatable design should be chosen based on the selected experimental matrix, which takes into account the number of chosen factors, levels, and runs. The most frequently used second-order response surface designs consist of (i) 3^v factorial design, (ii) Box-Behnken design (BBD), and (iii) Central (face) Composite design (CCD/FFCD).
5. Model building and validation: The process of model building and validation involves assessing the adequacy of the fitted model based on various mathematical-statistical criteria, such as prediction error sum of squares (PRESS) residuals, lack-of-fit tests,

residual analysis, and the coefficient of determination (R^2). In some instances, a high R^2 value may not necessarily reflect the accuracy of the model; in those cases, the absolute average deviation (AAD) serves as the most reliable measure. Once the fitted model is deemed adequate, the necessary optimization technique can then be applied. If there is a significant lack of fit evident in the model, then a higher-order model should be considered.

6. Optimization of the response can be achieved through both graphical methods (like response surface plots and contour plots) and numerical approaches. For multi-response optimization, the desirability function approach is employed.
7. Verification of results: To confirm the desired optimum, a confirmatory trial should be conducted to validate the results.

3. Response Surface Models

Let there be v independent input or experimental variables, referred to as factors, denoted by x_1, x_2, \dots, x_v and a response variable y and there are N observations. The response is a function of input factors, i.e.,

$$y_u = f(x_{1u}, x_{2u}, x_{3u}, \dots, x_{vu}) + e_u \quad \dots\dots (1)$$

where $u=1,2,\dots,N$, x_{iu} is the level of the i^{th} ($i=1,2,\dots,v$) factor in the u^{th} treatment combination, y_u denotes the response obtained from u^{th} treatment combination. The function f describes how the response is related to the input variables, while e_u represents the random error associated with the observation, which is assumed to be independently and normally distributed with a mean zero and a common variance σ^2 .

In practical scenarios, the exact form of f is unknown, so it is approximated within the experimental range by a polynomial of an appropriate degree in the variables. Polynomials that effectively represent the true dose-response relationship are termed response surface models, and the designs that facilitate the fitting of these response surfaces and provide metrics for assessing their adequacy are known as response surface designs.

- If the function f is a polynomial of degree one, it is termed a first order (linear) response surface is

$$f(x_u) = \beta_0 + \sum_{i=1}^v \beta_i x_{iu} \quad \dots\dots (2)$$

- For a polynomial of degree two, the second order (quadratic) response surface is

$$f(x_u) = \beta_0 + \sum_{i=1}^v \beta_i x_{iu} + \sum_{i=1}^v \beta_{ii} x_{iu}^2 + \sum_{i=1}^{v-1} \sum_{j=i+1}^v \beta_{ij} x_{iu} x_{ju} \quad \dots\dots (3)$$

- With polynomial of degree three, the third order (cubic) response surface is

$$f(x_u) = \beta_0 + \sum_{i=1}^v \beta_i x_{iu} + \sum_{i \leq j=1}^v \beta_{ij} x_{iu} x_{ju} + \sum_{i \leq j \leq l=1}^v \beta_{ijl} x_{iu} x_{ju} x_{lu} \quad \dots\dots (4)$$

β_0 is a constant, β_i , β_{ii} , β_{iii} are the i^{th} linear, quadratic, cubic regression coefficient and β_{ij} , β_{ijl} are the $(i, j)^{\text{th}}$, $(i, j, l)^{\text{th}}$ interaction coefficient respectively.

Equation (1) can be expressed in matrix notation as

$$\mathbf{Y} = \mathbf{X}\boldsymbol{\beta} + \mathbf{e} \quad \dots\dots\dots (5)$$

Where $\mathbf{Y} = (y_1, y_2, \dots, y_N)'$ is an $N \times 1$ vector of observations and

- For linear regression, $\mathbf{X} = \mathbf{X}(\mathbf{L}) = [\mathbf{1}_v, \mathbf{x}]$ is an $N \times (v+1)$ matrix of linear regression with $\mathbf{x} = (x_1, x_2, \dots, x_v)$ is an $N \times v$ matrix of independent (explanatory) variables, $\boldsymbol{\beta}(\mathbf{L}) = (\beta_0, \beta_1, \beta_2, \dots, \beta_v)'$ is a $(v+1) \times 1$ vector of unknown parameters;
- For quadratic response surface model, $\mathbf{X} = \mathbf{X}(\mathbf{L}|\mathbf{Q})$ is an $N \times \binom{v+2}{2}$ matrix with $\mathbf{X}(\mathbf{Q}) = (x_1^2, x_2^2, \dots, x_v^2, x_1 x_2, x_1 x_3, \dots, x_{v-1} x_v)$ and $\boldsymbol{\beta}(\mathbf{L}|\mathbf{Q})$ takes $[(v+1)(v+2)/2] \times 1$ vector of the unknown parameters, where $\boldsymbol{\beta}(\mathbf{Q}) = (\beta_{11}, \beta_{22}, \dots, \beta_{vv}, \beta_{12}, \beta_{13}, \dots, \beta_{v-1,v})'$.
- For cubic response surface model, $\mathbf{X} = \mathbf{X}(\mathbf{L}|\mathbf{Q}|\mathbf{C})$ is an $N \times \binom{v+3}{3}$ matrix with $\mathbf{X}[\mathbf{C}] = (x_1^3, \dots, x_v^3, x_1^2 x_2, \dots, x_{v-1}^2 x_v, \dots, x_{v-2} x_{v-1} x_v)$ and $\binom{v+3}{3} \times 1$ is a vector of $\boldsymbol{\beta}(\mathbf{L}|\mathbf{Q}|\mathbf{C})$

corresponds to \mathbf{X} , $\mathbf{e} = (e_1, e_2, \dots, e_N)'$ is an $N \times 1$ vector of random errors distributed as $N(0, \sigma^2 \mathbf{I}_N)$.

By ordinary least squares, the estimates of β 's are

$$\hat{\beta} = (\mathbf{X}'\mathbf{X})^{-1} \mathbf{X}'\mathbf{Y}$$

and variance-covariance matrix is

$$D(\hat{\beta}) = (\mathbf{X}'\mathbf{X})^{-1} \sigma^2$$

With estimate of σ^2 can be obtained as

$$\hat{\sigma}^2 = \frac{(\mathbf{Y} - \mathbf{X}\hat{\beta})'(\mathbf{Y} - \mathbf{X}\hat{\beta})}{N - (v + 1)}$$

Since the variance of parameters $D(\hat{\beta})$ relies on the design matrix $(\mathbf{X}'\mathbf{X})$, one can select the design matrix \mathbf{X} with mutually orthogonal column vectors to ensure that the components of $\hat{\beta}$ of the estimates have zero pairwise correlation and are thus independent.

Orthogonality: A design is considered orthogonal if it is represented as diagonal, which means that the columns of \mathbf{X} are mutually orthogonal, resulting in the levels of corresponding variables being linearly independent. In this scenario, the elements of will be uncorrelated, facilitating the independent assessment of the significance of unknown parameters. Therefore, a design is classified as orthogonal if it provides independent information regarding the effects of various parameters in the model (Box and Hunter, 1957). Note: This pertains to the First Order Orthogonal Design [FOOD]. It is evident from the definition that orthogonal designs for models of higher degrees are not feasible. A type of orthogonal design can be achieved by representing the response function in terms of orthogonal polynomials. However, this approach has the drawback that the parameters of the new model are influenced by the design.

Rotatability: A design is considered rotatable if it maintains a constant prediction variance for all points that are equidistant from the design center. The variance associated with the estimate of the mean response at a specific point \mathbf{x}_0 is.

$$V(\hat{y}(\mathbf{x}_0)) = \sigma^2 \mathbf{x}_0' (\mathbf{X}'\mathbf{X})^{-1} \mathbf{x}_0$$

If the variance is uniform for all points x_0 that are situated the same distance from the design center, the design is recognized as possessing the property of rotatability. To guarantee orthogonality among parameter estimates and consistency in the variance of estimated responses at points equidistant from the design center, the x_{iu} values must adhere to the following Orthogonality-Rotatability conditions. When rotatability is achieved, the variance of the estimated response is expressed as a function of $\sum_{i=1}^v x_{i0}^2 = d^2$. Consequently, the variance of the estimated response at all points equidistant from the design center will be identical.

Orthogonality-Rotatability conditions of RSD:

$$\begin{array}{lll}
 1 & \sum_{u=1}^N x_{iu} = 0, & 2 \quad \sum_{u=1}^N x_{iu} x_{ju} = 0, \\
 & \forall i = 1, 2, \dots, v & \forall i \neq j = 1, 2, \dots, v \\
 & & 3 \quad \sum_{u=1}^N x_{iu}^2 = N\lambda_2 = \text{constant}, \\
 & & \forall i = 1, 2, \dots, v \\
 4 & \sum_{u=1}^N x_{iu}^4 = 3 \sum_{u=1}^N x_{iu}^2 x_{ju}^2 = 3N\lambda_4 = \text{constant}, & \forall i \neq j = 1, 2, \dots, v
 \end{array}$$

and all other sums of powers and products up to and including order four are zero

$$\begin{array}{l}
 5 \quad \frac{\lambda_4}{\lambda_2^2} > \frac{v}{v+2} \quad (\text{non-singularity condition}) \\
 6 \quad \sum_{u=1}^N x_{iu}^6 = 5 \sum_{u=1}^N x_{iu}^2 x_{ju}^4 = 15 \sum_{u=1}^N x_{iu}^2 x_{ju}^2 x_{ku}^2 = 15N\lambda_6 = \text{constant}; \text{ for all } i \neq j \neq k
 \end{array}$$

and all other sums of powers and products up to and encompassing order six are equal to zero.

$$7 \quad \frac{\lambda_2 \lambda_6}{\lambda_4^2} > \frac{v+2}{v+4} \quad (\text{non-singularity condition})$$

The designs satisfying conditions (1 to 3), (1 to 5), (1 to 7) are called First Order Rotatable Designs (FORDs)/ Orthogonal Designs, Second Order Rotatable Designs (SORDs) and Third Order Rotatable Design (TORD).

4. Response Surface Designs

Response Surface Designs (RSDs) are widely utilized in experiments to ascertain the connection between the response and a range of experimental factors (both quantitative and

qualitative) and to identify the combination of factor levels that produces optimal responses (Khuri and Cornell, 1996; Myers et al., 2016). RSDs find extensive applicability in the development, enhancement, and optimization of processes across various scientific fields. Numerous books can be referenced for a deeper understanding of the underlying principles of the theory, as outlined by Myers (1971), Khuri and Cornell (1996), Khuri (2006), Box and Draper (2007), and Myers et al. (2016). Additionally, several review papers discussing the concept of Response Surface Methodology (RSM) are available (Hill and Hunter, 1966; Mead and Pike, 1975; Myers et al., 1989; Myers et al., 2004; Khuri and Mukhopadhyay, 2010; Hemavathi et al., 2022). In this context, some of the frequently used RSDs will be examined.

Factorial designs are commonly employed in experiments with multiple factors where it is essential to explore the combined effects (main effects and interactions) of these factors on a response variable. A significant specific instance of the factorial design occurs when each of the v factors of interest has only two levels. Since each replication of this design consists of exactly 2^v experimental trials or runs, these designs are typically referred to as 2^v factorial designs. The category of 2^v factorial designs holds considerable importance in response surface methodology. More specifically, they are applied in three key areas:

1. The 2^v design (or a portion of it) is beneficial at the beginning of a response surface study, where screening experiments aim to pinpoint the critical process or system variables.
2. A 2^v design is frequently utilized to develop a first-order response surface model and to produce estimates for factor effects.
3. A 2^v design serves as a fundamental component for creating other advanced response surface designs. For instance, by supplementing a 2^v design with axial runs, one can derive a central composite design, which is among the most significant designs for fitting second-order response surface models.

The central composite design (CCD) is the most widely used category of second-order designs. It was proposed by Box and Wilson in 1951. The primary motivation for using CCD stems from its application in sequential experimentation. This design incorporates a two-level factorial or fractional design (resolution V) along with a set of 2^v axial or star points and several central points.

	x_1	x_2	.	.	.	x_v
$-\alpha$	0	0
α	0	0
0	$-\alpha$	0
0	α	0
.
.
.
0	0	$-\alpha$
0	0	α

The design consists of F factorial points, $2v$ axial points, and n_c center runs. The overall number of runs equals $F + 2v + n_c$. The factorial points serve as a variance optimal design for a first-order model or a model that includes a first-order plus two-factor interaction. Center runs provide insight into the presence of curvature within the system. When curvature is detected in the system, the inclusion of axial points enables efficient estimation of the pure quadratic terms.

To fit second-order response surfaces, Box and Behnken (1960) developed a set of effective three-level designs. This category of designs is grounded in the creation of BIB designs. In various RSM scenarios, the scope of research is often too extensive to conduct all runs uniformly. Consequently, second-order designs that facilitate blocking—specifically, the incorporation of block effects—are essential and fascinating to study. It is vital that the design points are allocated to blocks in a manner that minimizes their influence on the model coefficients. The desired characteristic is orthogonal blocking, which means that the block effects in the model are orthogonal to the coefficients of the model.

5. RSM using software

5.1. R Software for RSM

There are R packages that can be utilized for creating response surface designs as well as for conducting analysis.

A few examples have been included below.

Response-Surface Methods in R, Using rsm

Updated to version 2.10.2, 3 September 2020

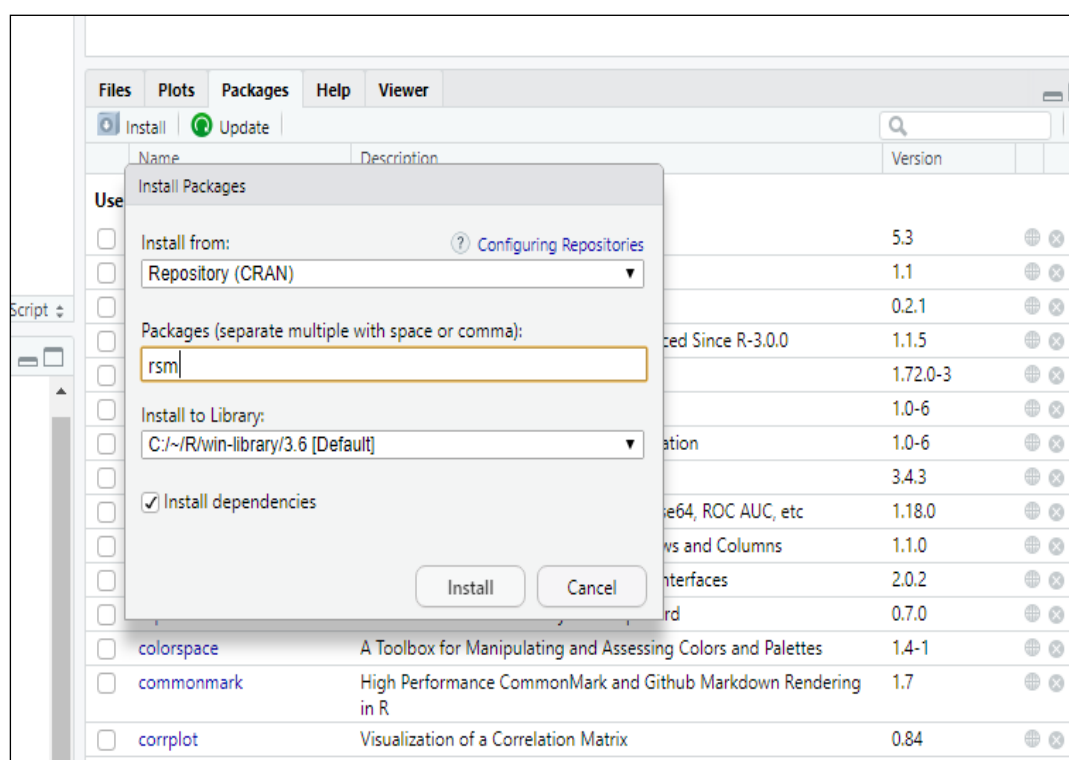
Russell V. Lenth
The University of Iowa

Abstract

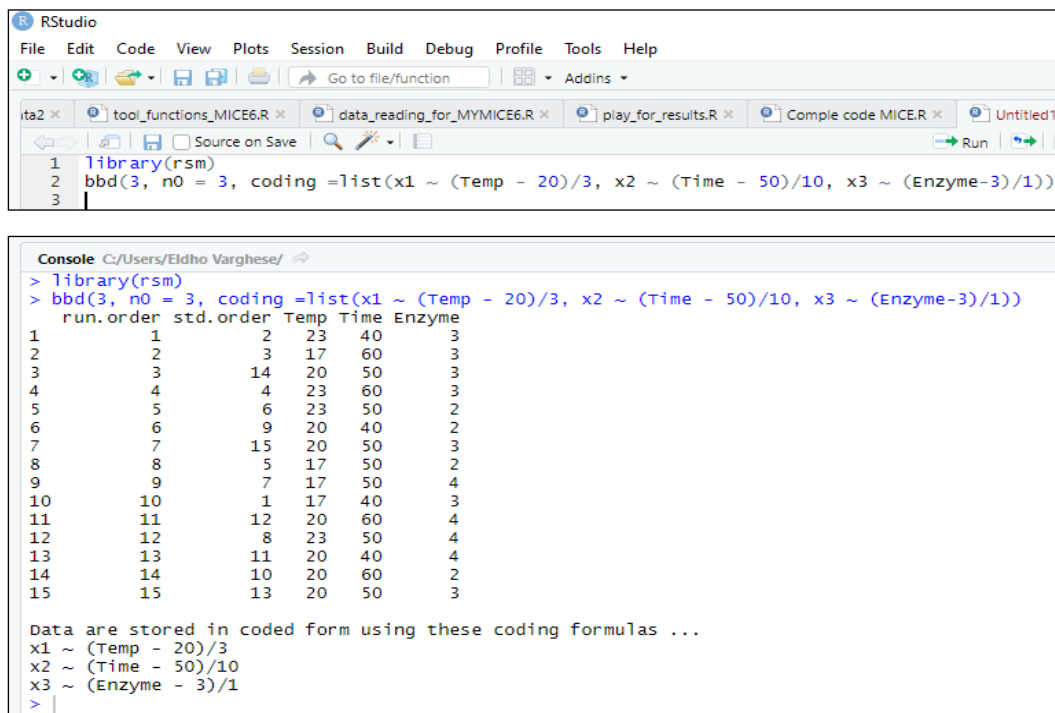
This introduction to the R package **rsm** is a modified version of Lenth (2009), published in the *Journal of Statistical Software*. The package **rsm** was designed to provide R support for standard response-surface methods. Functions are provided to generate central-composite and Box-Behnken designs. For analysis of the resulting data, the package provides for estimating the response surface, testing its lack of fit, displaying an ensemble of contour plots of the fitted surface, and doing follow-up analyses such as steepest ascent, canonical analysis, and ridge analysis. It also implements a coded-data structure to aid in this essential aspect of the methodology. The functions are designed in hopes of providing an intuitive and effective user interface. Potential exists for expanding the package in a variety of ways.

Keywords: response-surface methods, regression, experimental design, first-order designs, second-order designs.

To install the package named “rsm”



Generation of Box-Behnken design



The screenshot shows the RStudio interface. The script editor contains the following code:

```
1 library(rsm)
2 bbd(3, n0 = 3, coding = list(x1 ~ (Temp - 20)/3, x2 ~ (Time - 50)/10, x3 ~ (Enzyme-3)/1))
3
```

The console output shows the generated design matrix:

```
> library(rsm)
> bbd(3, n0 = 3, coding = list(x1 ~ (Temp - 20)/3, x2 ~ (Time - 50)/10, x3 ~ (Enzyme-3)/1))
  run.order std.order Temp Time Enzyme
1         1         2   23   40      3
2         2         3   17   60      3
3         3        14   20   50      3
4         4         4   23   60      3
5         5         6   23   50      2
6         6         9   20   40      2
7         7        15   20   50      3
8         8         5   17   50      2
9         9         7   17   50      4
10        10         1   17   40      3
11        11        12   20   60      4
12        12         8   23   50      4
13        13        11   20   40      4
14        14        10   20   60      2
15        15        13   20   50      3
```

Below the table, the console shows the coding formulas used:

```
Data are stored in coded form using these coding formulas ...
x1 ~ (Temp - 20)/3
x2 ~ (Time - 50)/10
x3 ~ (Enzyme - 3)/1
>
```

For fitting the model

4. Fitting a response-surface model

A response surface is fitted using the `rsm` function. This is an extension of `lm`, and works almost exactly like it; however, the model formula for `rsm` must make use of the special functions `F0`, `TWI`, `PQ`, or `SO` (for “first-order,” “two-way interaction,” “pure quadratic,” and “second-order,” respectively), because the presence of these specifies the response-surface portion of the model. Other terms that don’t involve these functions may be included in the model; often, these terms would include blocking factors and other categorical predictors.

Sample data taken from Design Resource Server

<https://drs.icar.gov.in/Analysis%20of%20data/Analysis%20of%20Data.html>

```
install.packages("rsm")
```

```
library(rsm)
```

```
attach(rsd)
```

```
analysis<- rsm(yield~ SO(N,S), data=rsd)
```

```
summary(analysis)
```

```
Call:
rsm(formula = yield ~ SO(N, S), data = rsd)

            Estimate Std. Error t value Pr(>|t|)
(Intercept) 4266.249755  181.973252  23.4444 4.517e-10 ***
N            40.906446   3.889763  10.5164 1.000e-06 ***
S            19.226334   9.724408   1.9771 0.07624 .
N:S          -0.007803   0.046409  -0.1681 0.86983
NA2          -0.174811   0.023204  -7.5335 1.985e-05 ***
SA2          -0.178741   0.145027  -1.2325 0.24597
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Multiple R-squared:  0.9632,    Adjusted R-squared:  0.9448
F-statistic: 52.31 on 5 and 10 DF,  p-value: 7.585e-07
```

```
Analysis of Variance Table

Response: yield
            Df Sum Sq Mean Sq F value    Pr(>F)
FO(N, S)     2 10942610  5471305 101.6134 2.269e-07
TWI(N, S)    1    1522    1522    0.0283  0.8698
PQ(N, S)     2 3137664 1568832  29.1364 6.742e-05
Residuals   10   538443    53844
Lack of fit 10   538443    53844
Pure error   0         0
Stationary point of response surface:
              N              S
115.85825  51.25385

Eigenanalysis:
eigen() decomposition
$values
[1] -0.1724072 -0.1811440

$vectors
      [,1]      [,2]
N -0.8514159 0.5244911
S  0.5244911 0.8514159
```

#To provide contour plot#

```
rsd.lm <- lm(yield~ poly(N,S, degree = 3), data = rsd)
```

```
contour(mpg.lm, N ~ S, image = TRUE)
```

```
detach(rsd)
```

5.2. Design Expert Software for RSM

(i) Construction of CCD

Go to main menu and click Central Composite under response surface tab

Enter the number of factors and continue

Design-Expert 12

File Edit View Display Options Design Tools Help

Standard Designs

- Factorial
 - Randomized
 - Regular Two-Level
 - Min-Run Characterize
 - Min-Run Screen
 - Multilevel Categorical
 - Optimal (Custom)
 - Miscellaneous
- Split-Plot
- Response Surface
 - Randomized
 - Central Composite**
 - Box-Behnken
 - Optimal (Custom)
 - Miscellaneous
 - Supersaturated
 - Definitive Screening
 - Split-Plot
- Mixture
- Custom Designs
 - Optimal (Combined)
 - User-Defined
 - Historical Data
 - Simple Sample

Central Composite Design

Each numeric factor is set to 5 levels: plus and minus alpha (axial points), plus and minus 1 (factorial points) and the center point. If categorical factors are added, the central composite design will be duplicated for every combination of the categorical factor levels.

Numeric factors: 3 (2 to 50) ☒ Horizontal ☐ Enter factor ranges in terms of ± 1 levels

Categorical factors: 0 (0 to 10) ☐ Vertical ☐ Enter factor ranges in terms of alphas

	Name	Units	Low	High	-alpha	+alpha
A [Numeric]	A		-1	1	-1.68179	1.68179
B [Numeric]	B		-1	1	-1.68179	1.68179
C [Numeric]	C		-1	1	-1.68179	1.68179

Type: Full Blocks: 1

Points

Non-center points: 14

Center points: 6

alpha = 1.68179 Options... 20 Runs

Cancel << Back Next >> Finish

Select the number of response variables and continue

Design-Expert 12

File Edit View Display Options Design Tools Help

Standard Designs

- Factorial
 - Randomized
 - Regular Two-Level
 - Min-Run Characterize
 - Min-Run Screen
 - Multilevel Categorical
 - Optimal (Custom)
 - Miscellaneous
- Split-Plot
- Response Surface
 - Randomized
 - Central Composite**
 - Box-Behnken
 - Optimal (Custom)
 - Miscellaneous
 - Supersaturated
 - Definitive Screening
 - Split-Plot
- Mixture
- Custom Designs
 - Optimal (Combined)
 - User-Defined
 - Historical Data
 - Simple Sample

Central Composite Design

Responses: 1 (1 to 999) ☒ Horizontal ☐ Vertical

	Name	Units
	R1	

Cancel << Back Next >> Finish

Layout of the design as follows:

Std	Run	Factor 1 A:A	Factor 2 B:B	Factor 3 C:C	Response 1 R1
1	1	-1	-1	-1	
3	2	-1	1	-1	
11	3	0	-1.68179	0	
17	4	0	0	0	
16	5	0	0	0	
5	6	-1	-1	1	
14	7	0	0	1.68179	
18	8	0	0	0	
10	9	1.68179	0	0	
4	10	1	1	-1	
20	11	0	0	0	
7	12	-1	1	1	
12	13	0	1.68179	0	
9	14	-1.68179	0	0	
19	15	0	0	0	
15	16	0	0	0	
8	17	1	1	1	
2	18	1	-1	-1	
13	19	0	0	-1.68179	
6	20	1	-1	1	

(ii) Construction of Box-Behnken Design

Go to main menu and click Box-behnken under response surface tab

Enter the number of factors and number of blocks and then continue

Box-Behnken Design

Each numeric factor is set to 3 levels. If categorical factors are added, the Box-Behnken design will be duplicated for every combination of the categorical factor levels. These designs have fewer runs than 3-Level Factorials.

Numeric factors: 4 (3 to 21) ☒ Horizontal

Categorical factors: 0 (0 to 10) ☐ Vertical

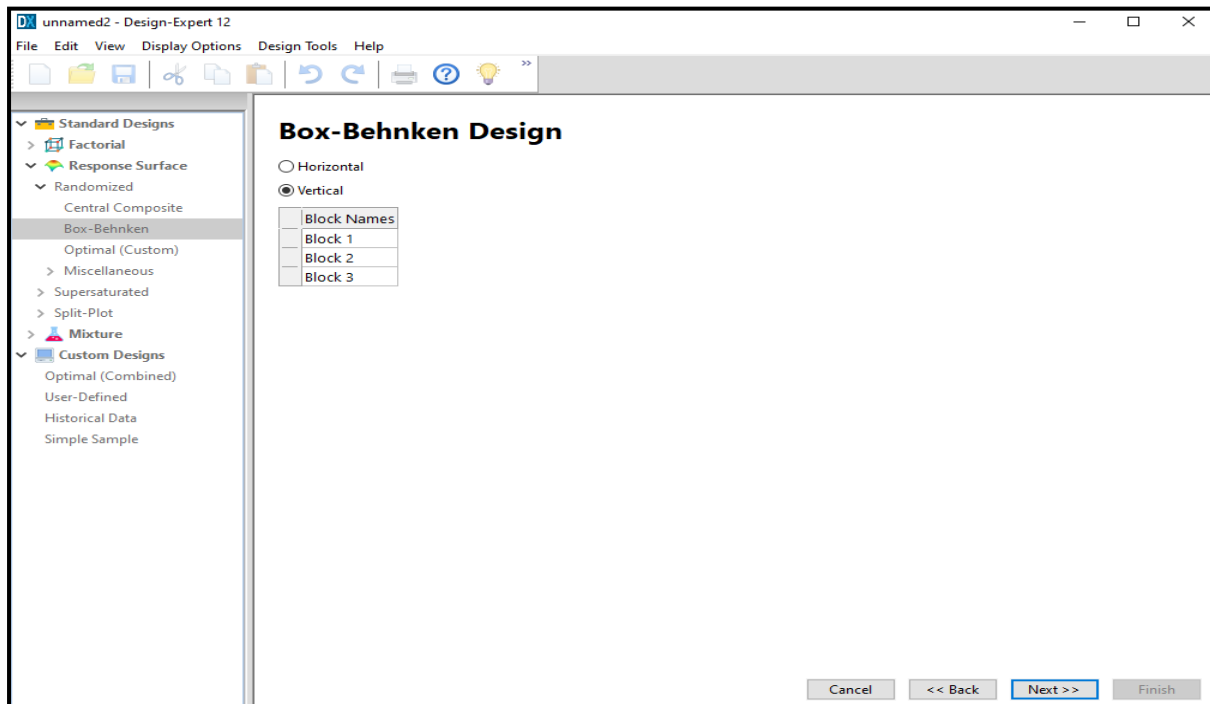
	Name	Units	Low	High
A [Numeric]	A		-1	1
B [Numeric]	B		-1	1
C [Numeric]	C		-1	1
D [Numeric]	D		-1	1

Blocks: 3

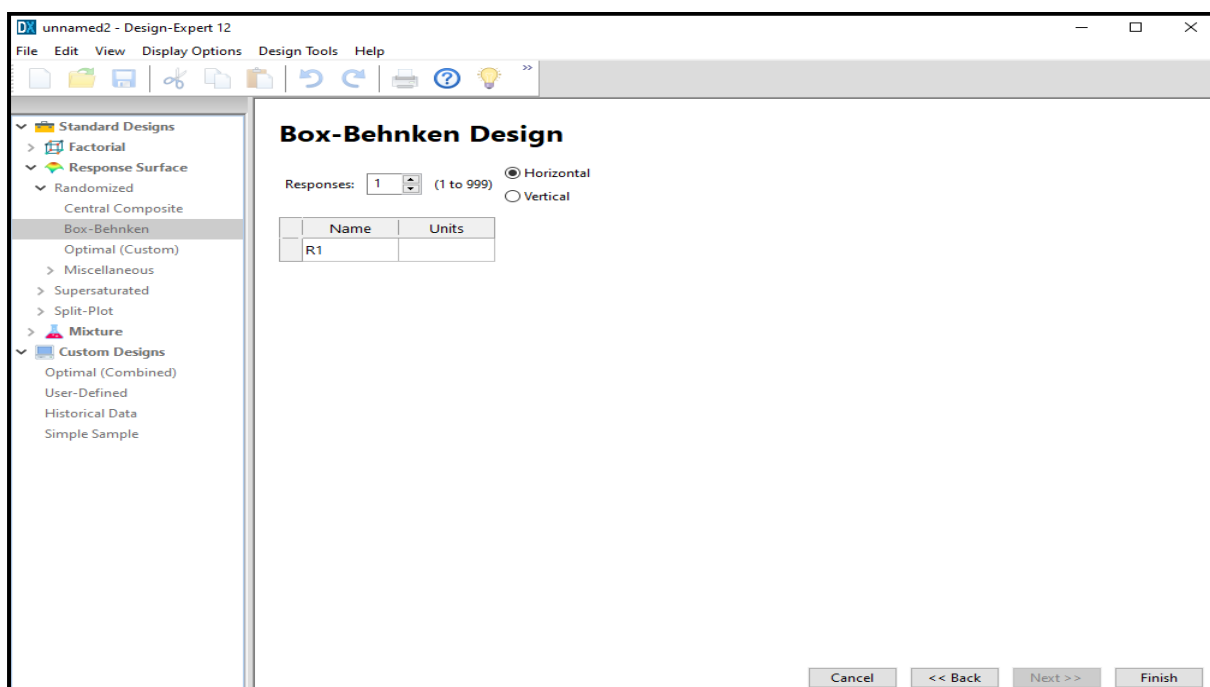
Center points per block: 2 (0 to 1000) 30 Runs

Cancel << Back Next >> Finish

Give the block labels (If required)



Select the number of response variables and continue



Layout of the design as follows:

Std	Block	Run	Factor 1 A:A	Factor 2 B:B	Factor 3 C:C	Factor 4 D:D	Response 1 R1
9	Block 1	1	0	0	0	0	
8	Block 1	2	0	0	1	1	
1	Block 1	3	-1	-1	0	0	
4	Block 1	4	1	1	0	0	
5	Block 1	5	0	0	-1	-1	
3	Block 1	6	-1	1	0	0	
10	Block 1	7	0	0	0	0	
2	Block 1	8	1	-1	0	0	
7	Block 1	9	0	0	-1	1	
6	Block 1	10	0	0	1	-1	
15	Block 2	11	0	-1	-1	0	
18	Block 2	12	0	1	1	0	
12	Block 2	13	1	0	0	-1	
13	Block 2	14	-1	0	0	1	
14	Block 2	15	1	0	0	1	
17	Block 2	16	0	-1	1	0	
11	Block 2	17	-1	0	0	-1	
20	Block 2	18	0	0	0	0	
16	Block 2	19	0	1	-1	0	
19	Block 2	20	0	0	0	0	
28	Block 3	21	0	1	0	1	
23	Block 3	22	-1	0	1	0	
25	Block 3	23	0	-1	0	-1	
27	Block 3	24	0	-1	0	1	
24	Block 3	25	1	0	1	0	
21	Block 3	26	-1	0	-1	0	
26	Block 3	27	0	1	0	-1	
22	Block 3	28	1	0	-1	0	
29	Block 3	29	0	0	0	0	
30	Block 3	30	0	0	0	0	

6. Practical Exercise

Exercise 6.1. (Taken from E book chapter written by V K Sharma and Rajender Parsad):

A central composite rotatable design was established to examine the impact of three fertilizer components on the yield of snap beans in field conditions. The fertilizer components and their corresponding amounts applied included nitrogen (N), ranging from 0.89 to 2.83 kg/plot; phosphoric acid (P₂O₅) from 0.265 to 1.336 kg/plot; and potash (K₂O), from 0.27 to 1.89 kg/plot. The primary response measured is the average yield of snap beans in kilograms per plot. The levels for nitrogen, phosphoric acid, and potash are coded, with the coded variables expressed as

$$X_1 = (N - 1.62)/0.71, X_2 = (P_2O_5 - 0.80)/0.31, X_3 = (K_2O - 1.08)/0.48$$

The values of 1.62, 0.80 and 1.08 kg/plot denote the centers for nitrogen, phosphoric acid, and potash, respectively. The experimental design utilizes five levels for each variable. The coded and actual levels of the variables are presented as follows.

Levels of x_i					
	-1.682	-1.000	0.000	+1.000	+1.682
N	0.42	0.91	1.62	2.34	2.83

P₂O₅	0.26	0.48	0.80	1.12	1.33
K₂O	0.27	0.60	1.08	1.57	1.89

A total of six central point replications were conducted to estimate the variance of experimental error.

The complete second-order model that will be fitted to obtain the values is

$$Y = \beta_0 + \sum_{i=1}^3 \beta_i x_i + \sum_{i=1}^3 \beta_{ii} x_i^2 + \sum_{i=1}^2 \sum_{i'=2}^3 \beta_{ii'} x_i x_{i'} + e$$

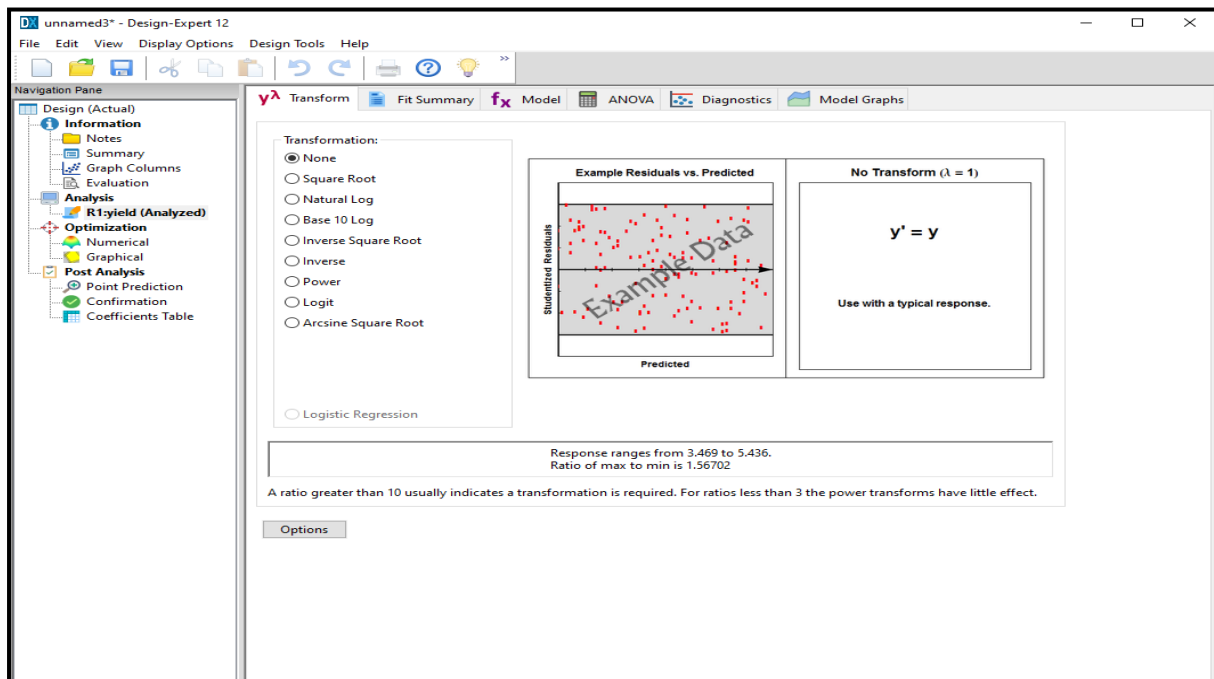
The table below presents the design settings of x_1 , x_2 and x_3 for N, P₂O₅, K₂O, along with the observed yields in kg at 15 design points.

Table 1. Central Composite Rotatable Design Settings in the Coded Variables x_1 , x_2 and x_3 , the original variables N, P₂O₅, K₂O and the Average Yield of Snap Beans at Each Setting

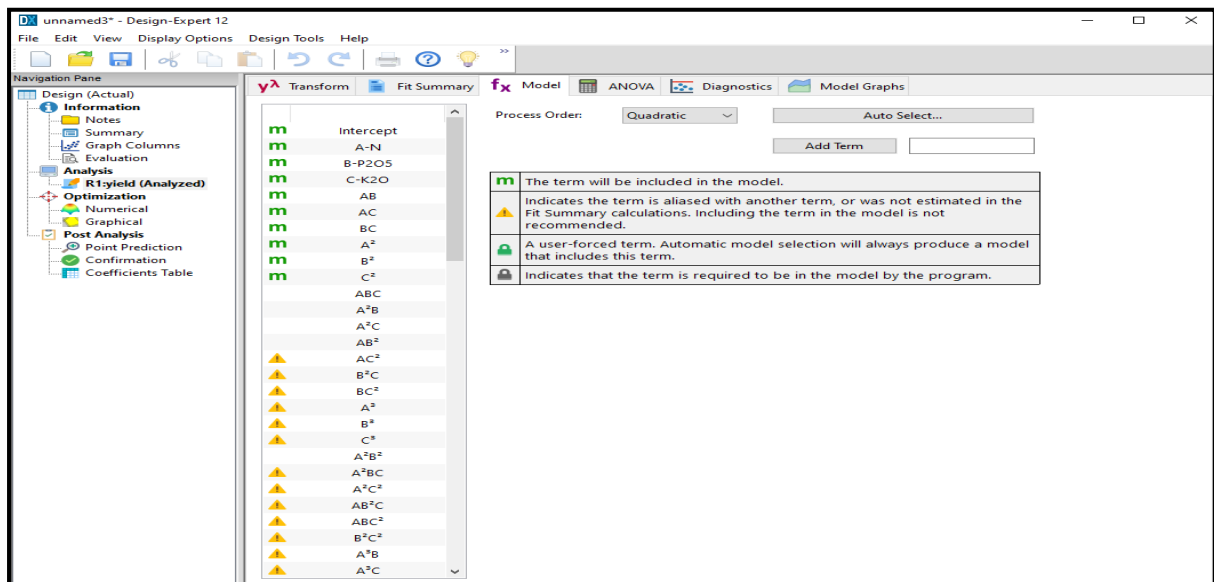
x_1	x_2	x_3	N	P ₂ O ₅	K ₂ O	Yield
-1	-1	-1	0.913	0.481	0.607	5.076
1	-1	-1	2.344	0.481	0.607	3.798
-1	1	-1	0.913	1.120	0.607	3.798
1	1	-1	2.344	1.120	0.607	3.469
-1	-1	1	0.913	0.481	1.570	4.023
1	-1	1	2.344	0.481	1.570	4.905
-1	1	1	0.913	1.120	1.570	5.287
1	1	1	2.344	1.120	1.570	4.963
-1.682	0	0	0.423	0.796	1.089	3.541
1.682	0	0	2.830	0.796	1.089	3.541
0	-1.682	0	1.629	0.265	1.089	5.436
0	1.682	0	1.629	1.336	1.089	4.977
0	0	-1.682	1.629	0.796	0.270	3.591
0	0	1.682	1.629	0.796	1.899	4.693
0	0	0	1.629	0.796	1.089	4.563
0	0	0	1.629	0.796	1.089	4.599
0	0	0	1.629	0.796	1.089	4.599
0	0	0	1.629	0.796	1.089	4.275
0	0	0	1.629	0.796	1.089	5.188
0	0	0	1.629	0.796	1.089	4.959

Analysis of data using Design Expert is as follows:

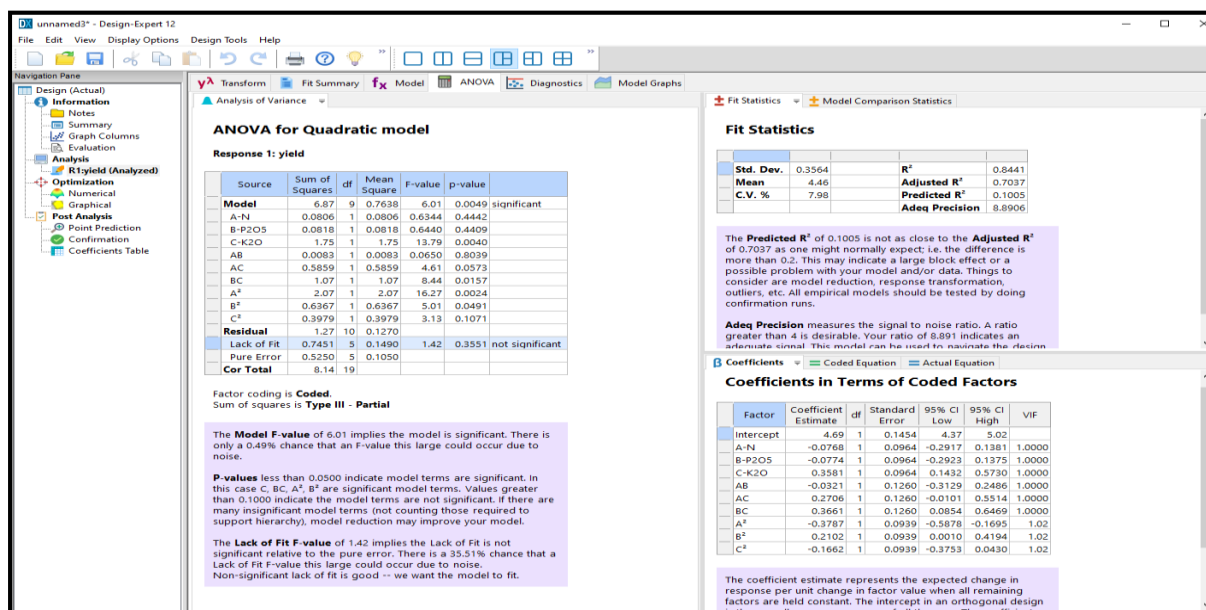
Check whether any transformation is required



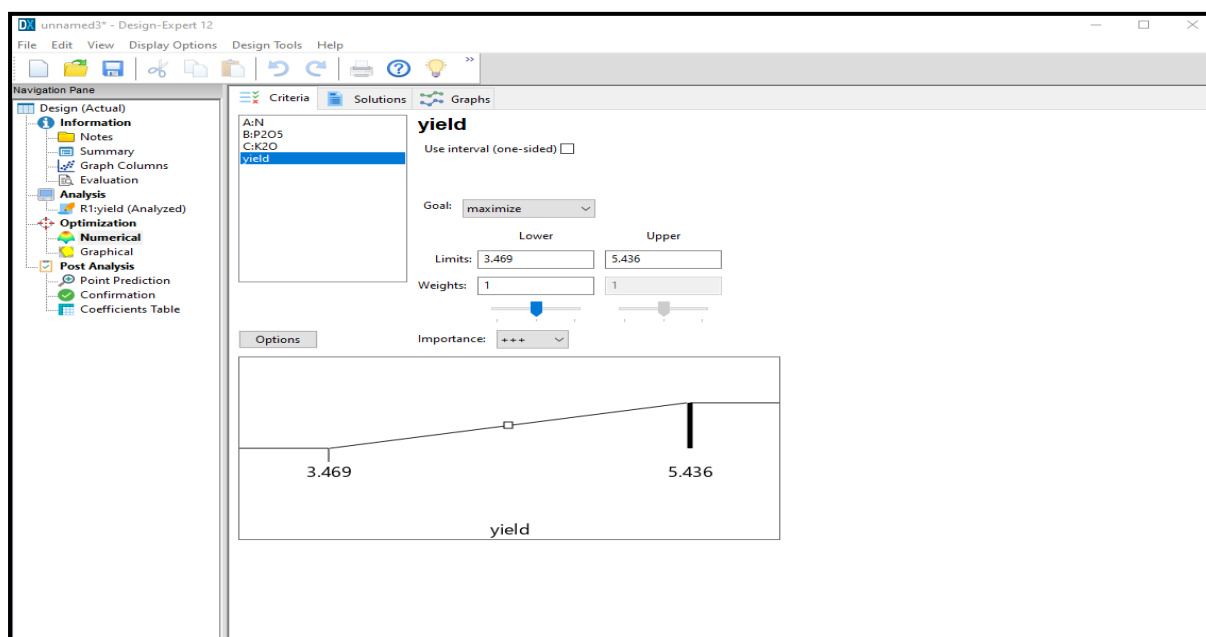
Select the model



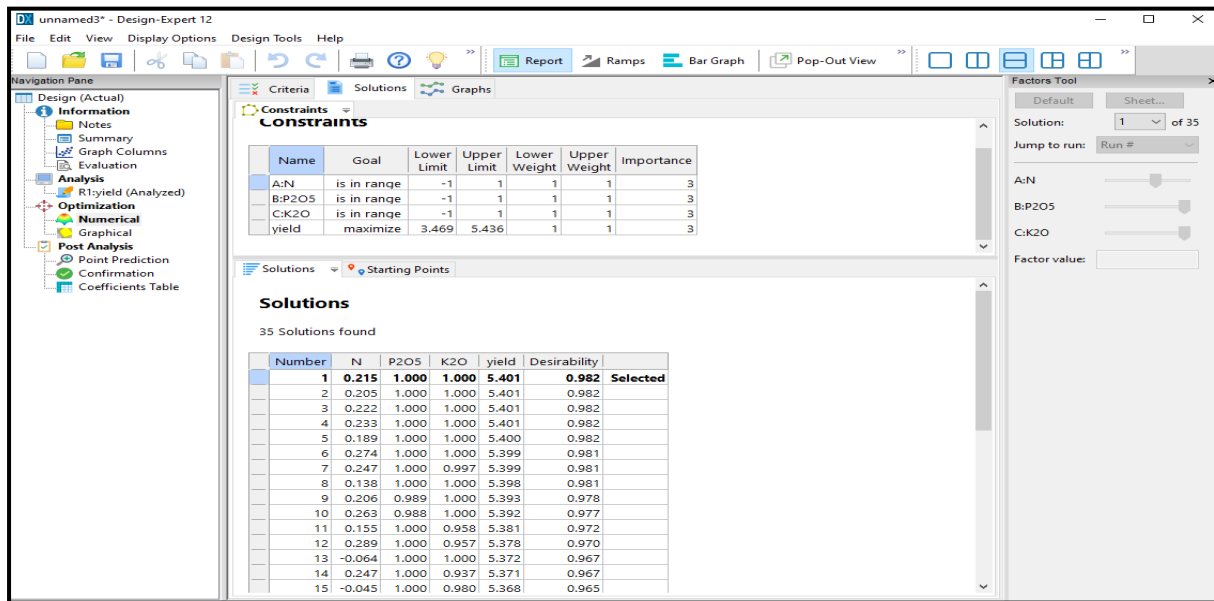
See for significance of the model fitted



Proceed for optimization and specify the target

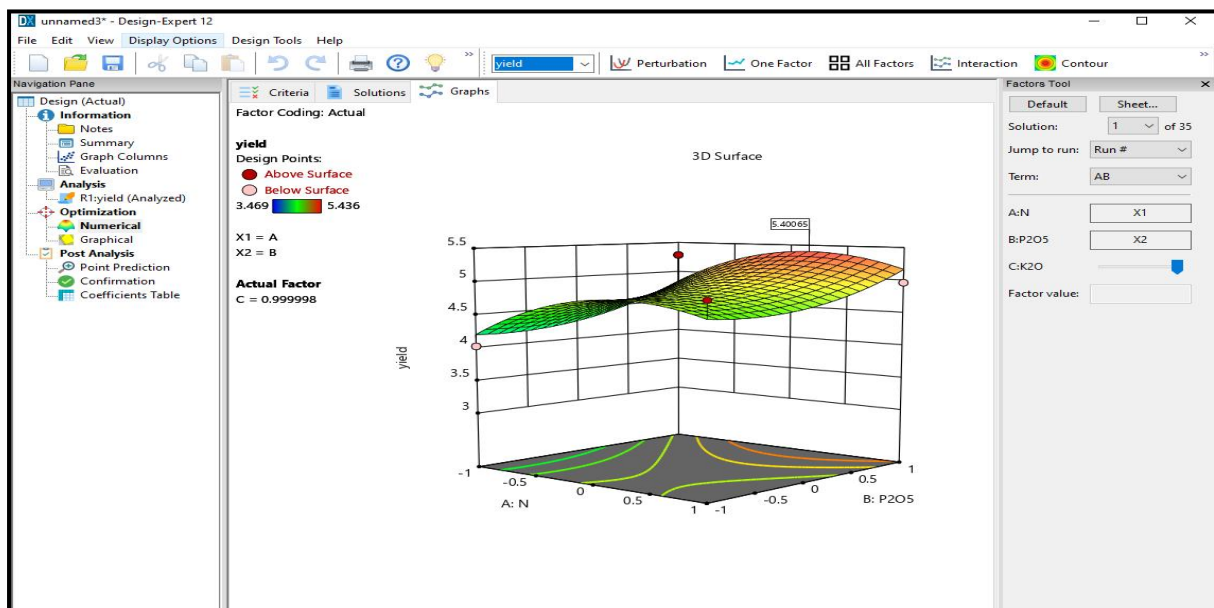


See for desirable solutions



See the confirmation report

Various surface plots can be generated – Contour and 3D surface plot



One can select a practically feasible combination of N, P and K.

7. Machine Learning Approaches for Optimization Trials

In recent years, machine learning (ML) approaches have emerged as powerful tools for optimization trials across various scientific domains, including agriculture, environmental modeling, and climate studies. Unlike traditional statistical methods that often require predefined model structures, ML algorithms such as Random Forests, Support Vector Machines, Gradient Boosting, and Artificial Neural Networks can model complex, nonlinear relationships between multiple input variables and response outcomes with minimal assumptions. These approaches excel in handling large, multidimensional datasets, capturing intricate interactions, and identifying patterns that may be overlooked by conventional techniques. In optimization trials, ML can be integrated with global search algorithms like Genetic Algorithms or Bayesian Optimization to identify optimal input combinations for maximizing or minimizing a desired outcome. Moreover, ML models offer adaptive learning, enabling continuous model refinement as new data become available, which is particularly valuable in dynamic and uncertain environments such as those affected by climate change.

7.1 Example: An Integrated Generalized Additive Model–Genetic Algorithm (GAM–GA) Optimization Approach in Climate Studies

To support the conservation of coral reef ecosystems, it is vital to identify potential climate refugia, areas where environmental stress is minimized and conditions remain relatively stable and favourable. In this study, an Optimal Environmental Window (OEW) was determined through an integrated approach combining Generalized Additive Models (GAMs) and Genetic Algorithms (GAs). GAMs were used to model nonlinear relationships between the Coral Resilience Index (CRI) and key environmental variables, while GAs were applied to optimize and select the best combination of predictor variables contributing to reef resilience. Secondary in situ data, including Live Coral Cover (LC), Damaged Coral Cover (DC), and Macroalgal Cover (MC), were collected from the Andaman and Nicobar Islands, a key coral reef region within India's Exclusive Economic Zone (EEZ). These metrics were used to compute the CRI, serving as a proxy for reef health and recovery potential. The integrated GAM–GA methodology enabled the identification of OEWs that represent optimal environmental conditions associated with low CRI values—indicative of resilient coral ecosystems. To ensure the robustness and reliability of the optimized conditions, bootstrapping techniques were employed to generate confidence intervals for both the predictor variables and the estimated

CRI responses. This data-driven framework offers critical insights into environmental thresholds essential for coral reef persistence and underscores the potential of OEW-based strategies in climate-resilient conservation and restoration planning (Joseph et al., 2025).

References

- Box, G. E. P. and Behnken, D. W. 1960. Some new three level designs for the study of quantitative variables. *Technometrics*, 2, 455–475.
- Box, G. E. P. and Wilson, K. B. 1951. On the experimental attainment of optimum conditions. *J. Roy. Statist. Soc., Ser. B*, 13, 1-45.
- Box, G. E. P., and N. R. Draper. 2007. Response Surfaces, Mixtures, and Ridge Analyses, 2nd ed. Hoboken, NJ: Wiley.
- Joseph, N.K., Varghese, E., Shafeeqe, M. and George, N. 2025. Optimal environmental window of coral reef bright spots in Andaman and Nicobar Islands: An integrated GAM–GA optimization approach. *Environmental Monitoring and Assessment*. Under Print.
- Hill, W. J. and Hunter, W. G. 1966. A review of response surface methodology, A literature survey. *Technometrics*, 84, 571- 590.
- Hemavathi, M., Varghese, E., Shekhar, S., Jaggi, S., Bhowmik, A. and Sathianandan, T.V., 2022c. Run order consideration for sequential third order rotatable designs. *Communications in Statistics-Simulation and Computation*, 1-14. DOI: 10.1080/03610918.2022.2039706
- Khuri, A. I. 2006. Response surface methodology and related topics. Singapore: World Scientific Publishing.
- Khuri, A. I., and J. A. Cornell. 1996. Response surfaces, designs and analyses, 2nd ed. New York: Dekker.
- Khuri, A. I. and Mukhopadhyay, S. 2010. Response surface methodology. Wiley interdisciplinary reviews. *Comput. Statist.*, 2(2), 128-149.

Mead, R. and Pike, D. J. 1975. A review of response surface methodology from a biometrics viewpoint. *Biometrics*, 31, 803-851.

Myers RH. 1971. Response surface methodology. Allyn and Bacon, Boston.

Myers R. H, Khuri A. I and Carter WH. 1989. Response methodology, 1966-1988, *Technometrics*, 31, 137-157.

Myers R. H, Montgomery D. C, Vining GG, Borror CM and Kowalski SM. 2004. Response surface methodology a retrospective and literature survey. *J Qual Technol*, 36 (1), 53-77.

Myers, R. H., D. C. Montgomery, and C. M. Anderson-Cook. 2016. Response surface methodology: Process and product optimization using designed experiments, 4th ed. Hoboken, NJ: John and Wiley Sons.

Generation Mean Analysis

Rafat Sultana

ICAR-National Institute of Abiotic Stress Management, Baramati, Pune, 413115

Email: rafat.hayat@gmail.com

Crop improvement relies on continuous selection and hybridization methods to enhance the genetic makeup of plants, aiming to breed superior phenotypes with greater economic value for humankind. This practice dates back thousands of years to the dawn of human civilization, beginning as an art carried out by farmers and evolving into a science driven by plant breeders. The main aim of plant breeding is to improve both qualitative (oligogenic) and quantitative (polygenic) traits, including yield, nutritional quality, resistance to abiotic and biotic stresses, and changes in maturity duration or growth habit (such as determinate vs. indeterminate growth or dormancy). Agronomically important traits like plant height, branching, and erect or trailing growth habits are also key targets of improvement. Additionally, breeding objectives often include eliminating toxic compounds, achieving non-shattering seed characteristics, promoting synchronous maturity, and introducing photoperiod and temperature insensitivity. Broader goals such as enhancing adaptability across environments and developing varieties suitable for new growing seasons in various crops are also central to modern plant breeding efforts. To improve these complex traits effectively, the choice of selection and breeding procedures for the genetic enhancement of any crop depends largely on understanding the type and relative contributions of genetic components—namely additive, dominance, and epistatic effects—as well as the presence of non-allelic interactions for each trait in the plant material. The complexity of many important traits, which are often controlled by multiple genes and their interactions, poses significant challenges for deciphering their inheritance patterns and improving them through selection.

1. Why is Generation Mean Analysis (GMA) Needed?

Generation mean analysis is a quantitative genetics method used to estimate the types and magnitude of gene effects controlling a trait in a crop or organism.

It relies on comparing the means of different generations derived from crossing two parents — typically: P_1 (Parent 1), P_2 (Parent 2), F_1 (First filial generation), and F_2 (Second filial generation) and their backcross. GMA offers a robust framework for dissecting the genetic

architecture underlying complex traits. By analysing the means of parental, filial, and backcross generations, GMA enables breeders to partition genetic effects into additive, dominance, and epistatic components. This understanding helps breeders design more efficient and targeted selection strategies for crop and improvement such as

- I. **Complex Traits Are Polygenic:** Most important crop traits (yield, stress tolerance, quality) are controlled by many genes (polygenes), each with small effects — often with additive, dominance, and epistatic interactions. Conventional selection alone cannot reveal these underlying genetic contributions.
- II. **Dissect Inheritance Patterns:** Breeders need to know if a trait is controlled mainly by additive effects (which respond well to selection) or if dominance and epistasis are involved (which require different strategies).
- III. **Guide Breeding Strategies:** By identifying the predominant type of gene action, GMA informs breeders when to apply selection (early vs. late generations) or whether hybrid breeding may be more effective.
- IV. **Detect Epistasis:** GMA (with scaling and joint scaling tests) can detect non-allelic interactions (epistasis), which are often responsible for failure of simple Mendelian expectations in segregating populations.
- V. **Practical & Cost-Effective:** Unlike genomic or molecular tools, GMA uses phenotypic data from traditional crosses, making it accessible and affordable, especially in resource-limited breeding programs

2. Principle behind Generation Mean Analysis (GMA)

The Core Principal of GMA is that the mean performance of different generations derived from a cross between two contrasting parents reflects the underlying genetic effects (additive, dominance, epistasis) controlling a quantitative trait. By analysing these means, it can be estimated how much each component contributes to phenotypic variation — even in complex traits influenced by multiple genes. Mendelian inheritance refers to the principles of biological inheritance discovered by Gregor John Mendel in 1865 and 1866, rediscovered in 1900 by Hugo de Vries and Carl Correns, and later popularized by William Bateson. Mendel's discoveries of how traits (such as color and shape) are passed down from one generation to the

next led the concept of dominant and recessive modes of inheritance. Hence, Generation Mean Analysis is fundamentally an application of Mendel's laws of segregation and independent assortment to populations of segregating generations, allowing estimation of genetic components controlling complex traits through their effects on generation means.

How is GMA linked to Mendelian genetics?

Generation Mean Analysis is grounded in Mendel's laws of inheritance, especially

Law of Segregation: Alleles segregate during gamete formation, leading to predictable genotype and phenotype frequencies in offspring. According to this law, during gamete formation, the alleles for each gene segregate from each other so that each gamete carries only one allele for each gene. During fertilization, when the gametes combine, the resulting offspring inherit one allele from each parent

Law of Independent Assortment: Genes at different loci assort independently (when loci are unlinked), which is key when modeling polygenic traits. This law states that genes of different traits can segregate independently during the formation of gametes which means the inheritance of one gene does not affect the inheritance of another gene. In other words, alleles of different genes segregate independently of one another during gamete formation. This principle assumes that the genes are located on different chromosomes or are far apart from each other on the same chromosome.

These laws mean that, for example:

- In F_1 , all individuals are heterozygous and their mean reflects dominance deviations if present.
- In F_2 , segregation of alleles produces expected proportions of homozygotes and heterozygotes, the F_2 mean reflects the combined contributions of additive and dominance effects, and any deviations point to epistasis.

Why is this important?

- GMA **extends** Mendel's single-gene principles to polygenic, quantitative traits by applying the same segregation and assortment laws to many genes at once.

- It connects **classic Mendelian theory with modern quantitative genetics**, helping breeders interpret complex inheritance patterns using basic genetic laws.

3. Partitioning genetic variance

Mendelian genetics predicts not only the distribution of genotypes but also how alleles contribute to genetic variance components (additive, dominance, epistatic), which is what GMA estimates from observed means. Genetic variance, on the other hand, refers to the variability in genetic traits observed within a population. Genetic variance arises from differences in the alleles carried by individuals in a population. These differences can be due to mutations, recombination etc. Genetic variance is important for natural selection and evolution. It allows for the presence of diverse traits within a population, and these traits can confer advantages or disadvantages in different environments. Genetic variance can be measured and quantified using various statistical methods, such as heritability, which estimates the proportion of variation in a trait that is due to genetic factors. Understanding both Mendelian inheritance and genetic variance is helpful for comprehending the transmission of traits from one generation to the next and the mechanisms underlying genetic diversity within populations.

The partitioning of genetic variance in plants refers to the analysis of the various sources of genetic variation within a population or species. It helps us to understand the relative contributions of different genetic factors to the observed phenotypic variation. Analysis of variance (ANOVA) separates the total phenotypic variance into different components, including the genetic variance, environmental variance, and interaction effects. The environmental variance represents the contribution of non-genetic factors such as growing conditions, soil fertility, and other environmental factors that can influence phenotypic variation.

The genetic variance can be partitioned into three major components: additive genetic variance, dominance genetic variance, and interaction (epistatic) genetic variance.

3.1 Additive Genetic Variance: This component of genetic variance is associated with the effects of individual genes that contribute additively to the phenotype. These genes typically have independent and cumulative effects on the trait. Additive genetic variance can be passed

from parents to offspring in a predictable manner and can be estimated using quantitative genetics methods such as narrow-sense heritability.

3.2 Dominance Genetic Variance: Dominance genetic variance arises when the combination of alleles at a particular locus produces a phenotypic effect that is different from the additive effects of the individual alleles. In other words, the heterozygote's phenotype differs from the average of the two homozygotes. Dominance effects can mask or enhance the expression of alleles at a particular locus, leading to additional sources of genetic variation.

3.3 Interaction (Epistatic) Genetic Variance: Interaction genetic variance, also known as epistatic variance, arises from the nonadditive interactions between genes at different loci. It accounts for the genetic variation resulting from the combined effects of multiple genes influencing a trait. Epistasis can be additive (when the combined effect is the sum of individual effects) or non-additive (when the combined effect is different from the sum of individual effects). By decomposing the genetic variance into these components, researchers can better understand the genetic structure of traits and assess the importance of different forms of genetic variation. This understanding is vital for plant breeders, as it assists them in crafting efficient breeding strategies aimed at enhancing desired traits, including yield, disease resistance, or stress tolerance.

4. Concepts of gene effects, gene frequencies, and their interactions:

Gene effects, gene frequencies, and their interactions are important concepts in the field of genetics and population genetics. Gene effects refer to the various ways in which genes can influence an organism's phenotype, or observable characteristics. Different types of gene effects are as follows;

4.1 Additive Effects: Additive gene effects occur when multiple genes contribute to a trait in a cumulative manner. Each gene has a small effect on the phenotype, and the effects are additive. For example, height in humans is influenced by multiple genes, and each gene contributes a small amount to the overall height.

4.2 Dominance Effects: Dominance gene effects occur when one allele (a variant of a gene) obscures the influence of another allele located at the same locus. When an individual possesses two different alleles for a gene, the dominant allele determines the phenotype while the recessive allele stays hidden. For instance, in Mendelian genetics, if a person receives a

dominant allele for a particular trait (like brown eyes) from one parent and a recessive allele (such as blue eyes) from the other parent, the dominant allele will determine the color of the eyes.

4.3 Epistatic Effects: Epistasis occurs when the effects of one gene mask or modify the effects of another gene. In other words, the interaction between two or more genes affects the phenotype. Epistasis can be either masking or modifying. Masking epistasis occurs when one gene masks the expression of another gene, while modifying epistasis occurs when the interaction between genes results in a modified expression of a trait. Epistatic effects can be observed in various traits, such as coat color in mammals.

4.4 Gene Frequencies: Gene frequencies, also known as allele frequencies, represent the relative abundance of different alleles within a population. Alleles are alternative forms of a gene that occupy the same locus on homologous chromosomes. The frequency of a particular allele is determined by counting the number of times it occurs in a population and dividing it by the total number of alleles at that locus. Gene frequencies are essential in population genetics and help in understanding patterns of genetic variation and evolution within a population. They can change over time due to several factors, including genetic drift, gene flow, mutation, natural selection, and non-random mating. The study of gene frequencies provides insights into population genetics, genetic diversity, and evolutionary processes.

5. Generation Mean Analysis and Testing Models:

The idea of generation mean analysis was introduced by Hayman (1958) and Jinks and Jones (1958) for evaluating gene action or variance components. This method relies on multiple generations and parental types, including F1, F2, and back crosses (B1 and B2). Mean values across replications were utilized to assess the genetic effects. The evaluation involved six or five generation populations in replicated field trials, and a biometrical model was created following the principles of generation mean analysis. This approach is valuable for determining primary effects such as additive and dominance, alongside digenic effects like additive x additive, additive x dominance, and dominance x dominance, as well as potential trigenic effects.

5.1 Scaling test (Testing of Models)

The scaling test was given by Mather (1949).

The means of the different generation's viz., P1, P2, F1, F2, B1 and B2 were used to test

$$A = 2B1 - P1 - F1 \dots\dots\dots(1)$$

$$B = 2B2 - P2 - F1 \dots\dots\dots(2)$$

$$C = 4F2 - 2F1 - P1 - P2 \dots\dots\dots(3)$$

$$D = 2F2 - B1 - B2 \dots\dots\dots(4)$$

P1, P2, F1, F2, B1, B2 are means of different generations over all replications.

The variances of the quantities A, B, C and D were calculated from the respective means of different generations.

$$VA = 4V(B1) - V(P1) - V(F1) \dots\dots\dots(5)$$

$$VB = 4V(B2) - V(P2) - V(F1) \dots\dots\dots(6)$$

$$Vc = 16V(F2) - V(F1) - V(P1) - V(P2) \dots\dots\dots(7)$$

$$VD = 4V(F2) - V(B1) - V(B2) \dots\dots\dots(8)$$

Where, VA, VB, VC, VD are variances of scales A, B, C and D respectively while V(P1), V(P2), V(F1) and V(F2) are variances of respective generations.

The standard errors of A, B, C and D respective by are obtained as $\sqrt{VA}, \sqrt{VB}, \sqrt{VC}, \sqrt{VD}$ and utilized for testing the significance of the deviations of the scales from zero. The significance of scales, A, B, C and D is determined by comparing the calculated and table 't' values.

$$tA = A / \sqrt{VA} \dots\dots\dots(9)$$

$$tA = B / \sqrt{VAB} \dots\dots\dots(10)$$

$$\frac{tA}{\sqrt{VC}} = C \dots\dots\dots(11)$$

$$tA = D / \sqrt{VD} \dots\dots\dots(12)$$

The significance of A and B scales indicate the presence of all the three types of non-allelic gene interaction viz., additive x additive, additive x dominance and dominance x dominance. The significance of C scale suggests dominance x dominance, while the D scale significance marks the presence of additive x additive type of non-allelic gene interactions.

5.2. Joint scaling test: Cavelli (1952) has proposed the Joint scaling test to test the presence of epistasis. In this test any combination of six populations are included at a time, whereas in individual scaling test only three or four populations were included in the study at a time which is the major drawback of this scaling test.

Estimation of gene effects with different models. The model to be used will depend on two factors, the first factor is presence or absence of epistasis and the second being the type and number of generations

5.2.1 Three parameter model

This method was proposed by Jinks and Jones (1958). When there is no indication of epistasis by scaling test, this method is followed. The gene effects of mean (m), additive (d) and dominance (h) were estimated

5.2.2 Five parameter model

This model is used when back crosses are not available. In this model five gene effects viz., Mean (m), Additive effect (d), Dominance effect (h), Additive x Additive interaction effect (i) and Dominance x Dominance interaction effect (l) but do not give information regarding Additive x Dominance interaction effect (j)

5.2.3 Six parameter model

The generation means were evaluated using the method proposed by Hayman (1959) to gain insights into the inheritance of different traits. These generation means served to estimate the six genetic parameters, namely (m), (d), (h), (i), and (j), of the digenic interaction model, which correspond to F2 mean, additive gene action, dominance genetic effect, additive × additive gene interaction effect, additive × dominance gene interaction effect, and dominance × dominance gene interaction effect, respectively, under the assumption that there is no linkage and no higher-order gene interaction. Using the generation means as reference points, the six genetic parameters were computed by following the relationship between the corresponding generation means and genetic effects.

Components of generation means

$$P1 = m + (d) + (i) \dots\dots\dots(13)$$

$$P2 = m - (d) + (i) \dots\dots\dots(14)$$

$$F1 = m + (h) + (l) \dots\dots\dots(15)$$

$$F2 = m + \frac{1}{2}(h) + \frac{1}{4}(l) \dots\dots\dots(16)$$

$$B1 = m + \frac{1}{2}(d) + \frac{1}{2}(h) + \frac{1}{4}(i) + \frac{1}{4}(j) + \frac{1}{4}(l) \dots\dots\dots(17)$$

$$B2 = m - \frac{1}{2}(d) + \frac{1}{2}(h) + \frac{1}{4}(i) + \frac{1}{4}(j) + \frac{1}{4}(l) \dots\dots\dots(18)$$

When scales A, B, C and D were significantly different from zero, a digenic interaction model was assumed and the following six parameters are estimated (Hayman, 1958).

$$d = B1 - B2$$
$$\text{Mean (m)} = F2 \dots\dots\dots(19)$$

$$\text{Additive effect } d = B1 - B2 \dots\dots\dots(20)$$

$$\text{dominant effect (h)} = F1 + 4(F2) - \frac{1}{2}(P1) - \frac{1}{2}(P2) + 2B1 - 2B2 \dots\dots(21)$$

$$\begin{aligned} \text{Additive x Additive interaction effect (i)} &= 2 \bar{B}^2_1 + 2 \bar{B}^2_2 + 4 \bar{F}^2 \dots\dots\dots(22) \\ \text{Additive x Dominance interaction effect (j)} &= 2 \bar{B}^2_1 - \bar{P}^2_1 - 2 \bar{B}^2_2 + \bar{P}^2_2 \dots\dots\dots(23) \\ \text{Dominance x Dominance interaction effect (l)} &= \bar{P}^2_1 + \bar{P}^2_2 + 2 \bar{F}^2_1 + 4 \bar{F}^2_2 - 4 \bar{B}^2_1 - 4 \bar{B}^2_2 \dots\dots\dots(24) \\ \text{Variances of the estimates of these parameters are obtained as follows:} & \\ Vm &= V\bar{F}^2 \dots\dots\dots(25) \\ Vd &= V\bar{B}^2_1 + V\bar{B}^2_2 \dots\dots\dots(26) \\ Vh &= V\bar{F}^2_1 + 16V\bar{F}^2_2 + 1/4V\bar{P}^2_1 + 1/4V\bar{P}^2_2 + 4V\bar{B}^2_1 + 4V\bar{B}^2_2 \quad (26) \quad Vi = 4V\bar{B}^2_1 + 1/4V\bar{B}^2_2 + 16V\bar{F}^2_2 \dots\dots\dots(27) \\ Vj &= 4V\bar{B}^2_1 + 1/4V\bar{P}^2_1 + V\bar{B}^2_2 + 1/4V\bar{P}^2_2 \dots\dots\dots(28) \\ Vl &= V\bar{P}^2_1 + V\bar{P}^2_2 + 4V\bar{F}^2_1 + 16V\bar{F}^2_2 + 16V\bar{B}^2_1 + 16V\bar{B}^2_2 \dots\dots\dots(29) \end{aligned}$$

Where, Vm is Variance of Mean effect, Vd is Variance of Additive effect, Vh is Variance of Dominance effect, Vi is Variance of Additive x Additive interaction effect, Vj is Variance of Additive x Dominance interaction effect and Vl is Variance of Dominance x Dominance interaction effect

The calculated ‘t’ value is referred to the ‘t’ table to test the significance. In each test, the degrees of freedom are sum of the degrees of freedom of various generations involved. When main or dominance effect (h) and interaction or dominance x dominance interaction effect (l) have similar signs + or –, the effect is beneficial and complementary. In this case selection may be delayed. While opposite in signs, the non-allelic interaction is duplicate type and under such situation bi parental mating is to be followed. If both additive and non-additive interactions were playing role for the inheritance of trait, then it is better to go for population improvement. Some of the model for GMA is as follows;

Hayman's Model: This is one of the earliest GMA models proposed by M. Hayman in 1954. It assumes that the genetic value of an individual is the sum of its sum of its additive and dominance genetic effects. Hayman's model allows estimation of additive, dominance, and epistatic genetic effects.

Jinks and Hayman's Model: Jinks and Hayman extended Hayman's model by including epistatic genetic effects. This model is particularly useful for predicting the performance of individuals in populations with complex genetic interactions.

Gardner and Eberhart model: This model, also known as the Method 3 of Griffing's model, is used for diallel cross experiments. It provides estimates of GCA and SCA effects similar to Griffing's method but assumes a different genetic model.

Griffing's method: This model, also known as the Method 2 of Griffing's model, is used for diallel cross experiments. It allows for the estimation of general combining ability (GCA) and specific combining ability (SCA) effects. GCA represents the average additive effect of a parent across all crosses, while SCA represents the specific interactions between parental genotypes.

North Carolina Design II: This model is an extension of the line \times tester analysis and is used in plant breeding programs with multiple testers. It allows the estimation of GCA and SCA effects, as well as tester-specific combining ability (TSCA) effects. These models and methods are just a few examples of the various approaches used in GMA. The choice of model depends on the specific breeding objectives, the genetic architecture of the traits under consideration, and the available resources and data. Researchers and breeders may adapt or develop new models based on their specific needs and research goals.

6. Applications of Generation Mean Analysis with Case Studies Generation mean analysis give estimates of components of mean which in turn provide information about the predominant type of gene action for the economically important yield components. The concepts of generation mean analysis is useful to detect the nonallelic interaction. The means and variances of different segregating and non-segregating generations indicate the nature of gene action and their interaction effects (Fisher et al., 1936; Mather, 1949 and Hayman, 1958).

7. Advancements in Generation Mean Analysis (GMA)

7.1 Integration with Mixed Models and REML

Traditional GMA relied on simpler linear models, but modern GMA can be integrated with mixed linear models (MLM) and restricted maximum likelihood (REML) methods. This allows more accurate estimation of genetic parameters by accounting for unbalanced data, missing values, or environmental effects. Traditional Generation Mean Analysis (GMA) relied on simpler linear models that assumed balanced data and ignored environmental variability. Modern GMA approaches integrate **mixed linear models (MLM)** and **restricted maximum likelihood (REML)** methods, which enable more accurate estimation of genetic parameters by effectively accounting for:

- I. **Unbalanced data**, common in field experiments where some generations or replicates may be missing.

- II. **Missing observations**, which often occur due to plot loss, pest damage, or environmental disruptions.
- III. **Environmental effects and genotype \times environment interactions**, improving the precision of estimates by separating genetic effects from environmental noise.
- IV. The use of MLM and REML also provides estimates of **variance components** associated with random effects (e.g., blocks, locations, years), which is crucial when conducting GMA across multiple environments or replications

7.2 Use of Computational Tools and Software:

Breeders now use powerful statistical software (e.g., SAS, R packages like AGD-R, asreml-R, lm, and lme4) for GMA, enabling simultaneous estimation of multiple traits, automatic calculation of standard errors, and formal hypothesis testing for scaling and joint scaling tests. To facilitate the implementation of GMA, several software packages have been developed. These packages provide user-friendly interfaces and computational tools for conducting generation mean analysis and related statistical procedures. Some examples include the R packages AGHmatrix (Amadeu et al., 2016) and AGHmatrix2 (Melo et al., 2022), which allow for GMA using additive genetic covariance matrices. (https://cran.rproject.org/web/packages/AGHmatrix/vignettes/Tutorial_AGHmatrix.html)

Other Software and Online Tools The other softwares tools which assist in analyzing the GMA are QTL Cartographer, GenStat, SAS, (agridat, GeneticsPed, and qtl.) packages of R software and Br Breeding View software. The website details were provided below

(<http://statgen.ncsu.edu/qtlcart/>) (<https://www.vsni.co.uk/software/genstat/>)

(https://www.sas.com/en_us/software/sas9.html). (<https://www.r-project.org/>).

(<https://www.breedingview.com/>).

7.3 GMA Combined with Molecular Data

Modern breeding programs increasingly overlay GMA results with QTL mapping, GWAS, or genomic selection data. This allows breeders to connect estimates of additive/dominance/epistatic effects from phenotypes with specific genomic regions — improving precision and understanding of complex traits. The combination of traditional GMA with Genomic selection in association with molecular markers are the powerful tools that can revolutionize the plant breeding by providing insights into genetic basis of traits and enhancing the selection efficiency. Genomic assisted GMA involves the genotyping of individuals of breeding population and QTL analysis to identify the region of genome related to trait of interest in combination of GMA. Statistical models are used to estimate the genetic effects and interactions of different alleles at specific loci. It helps in identifying the breeding programs and individuals with desirable marker genotypes and predicted breeding values can be selected for further breeding or advancement

7.4 Multivariate GMA

Instead of analyzing one trait at a time, recent developments allow multivariate GMA, which estimates genetic parameters for multiple correlated traits simultaneously — helping breeders understand genetic correlations and pleiotropy. The aim of multi-environment generation mean analysis is to ascertain the genetic and environmental effects on the performance of genotypes and their interaction. It helps in determining genotype by-environment interactions, which occur when the performance of genotypes differs across varied environments. Generation means are estimated by averaging the performance of genotypes across locations. The statistical models like AMMI (additive main effects and multiplicative interaction) and GGE (genotype plus genotypeby-environment interaction) models are employed to analyze the generation mean data. These models partition the total variation into genetic, environmental, and interaction components. With multi-environment generation mean analysis, genetic effects (main effects) of genotypes, environmental effects (main effects) of different environments, and the genotype-by-environment interactions can be understood by which how the genotypes perform under different conditions can be known and most promising genotypes for further breeding can be identified Software Tools for GMA

7.5 Bayesian GMA Approaches

Bayesian methods have been developed to estimate GMA parameters, providing more robust estimates, credible intervals, and flexibility to incorporate prior information or model complex interactions. In recent years, Bayesian approaches have been developed to estimate genetic parameters in GMA, offering several important advantages over classical (frequentist) methods, it is having Robust Estimates with Credible Intervals Unlike traditional methods that give only point estimates and confidence intervals, Bayesian GMA provides posterior distributions for each parameter (e.g., additive, dominance, epistatic effects), yielding credible intervals that more accurately reflect uncertainty — especially in small or unbalanced datasets common in breeding trials. It also allow to Use of Prior Information Bayesian frameworks allow breeders to incorporate prior knowledge (e.g., estimates from previous studies, literature, or expert opinion) into the analysis. This improves parameter estimation, especially when data are limited or traits have been studied extensively in related populations. Bayesian GMA can easily handle complex models, such as including multiple epistatic components, genotype \times environment interactions, or even multi-trait analyses — providing greater flexibility than standard linear models. Improved Fit for Non-Normal Data Bayesian methods are less sensitive to violations of assumptions (e.g., normality of residuals), which are common in phenotypic data from field experiments, making them more robust for real-world breeding data. E.g., Markov Chain Monte Carlo (MCMC) algorithms to estimate GMA parameters, Tools like JAGS, Stan, or R packages such as brms or rstanarm can fit Bayesian GMA models.

7.6 Application in Diverse Crops and Environments

Recent research extends GMA beyond major crops to underutilized or orphan crops (e.g., millets, pulses) and across highly variable environments — making it a critical tool in breeding for climate resilience and wider adaptation.

7.7 Better Experimental Designs

Modern GMA studies increasingly use replicated trials across multiple environments, improving precision and accounting for genotype \times environment interaction — something rarely addressed in earlier GMA implementations.

It's noteworthy that advancements in GMA are often closely tied to advancements in genomics, phenomics, statistical modeling, and computational resources. These interdisciplinary efforts contribute to a more comprehensive understanding of the genetic basis of complex traits and facilitate more efficient breeding strategies. They help in developing cultivars with enhanced traits to address various agricultural challenges and meet the needs of a growing population.

8. Reference:

- Cavalli, L. (1952). An analysis of linkage in quantitative inheritance. An analysis of linkage in quantitative inheritance.
- Falconer Douglas and Trudy F C Mackay. (2004). Introduction to quantitative genetics, 4 (56-72) by Pearson India
- Fisher, R.A. (1936). The use of multiple measurements in taxonomic problems. Ann. Eugen., 7:179-189.
- Hayman, B. (1954). The theory and analysis of diallel crosses. Genetics, 39(6), 789.
- Hayman, B.L. (1958). The separation of epistasis from additive dominance variation in generation means. Heredity, 12: 371- 390.
- Jinks, J., & Jones, R. M. (1958). Estimation of the components of heterosis. Genetics, 43(2), 223
- Mather, K. 1949. Biometrical Genetics. Dover Publications, Inc., New York. p 42-49.
- Ribaut, J. M., Betran, J., Monneveux, P., and Setter, T. (2012). "Drought tolerance in maize" in Handbook of Maize: Its Biology. eds J. L. Bennetzen and S. C. Hake (New York: Springer). 311–344. doi: 10.1007/978-0-387-79418-1_16

Mating Design (Diallel and Line X Tester Design)

Basavaraj PS and Boraiah KM

ICAR-National Institute of Abiotic Stress Management, Baramati, Pune, 413115

Email: bassuptl@gmail.com

Introduction:

A mating design is a structured method employed to plan and execute controlled crosses for generating progeny populations. The selection of suitable parental lines and the implementation of an appropriate mating strategy are essential components that influence the success of any plant breeding program (Khan et al., 2009). Breeders have the ability to direct the outcomes of such breeding efforts by carefully choosing parental combinations, controlling the frequency of parental use in crosses, and managing the number of progenies produced from each cross. These decisions are made to maximize genetic diversity, capture desirable traits, and efficiently meet the breeding objectives.

These are key factors to consider while selecting an appropriate mating design in plant breeding or genetic studies:

1. **Predominant Type of Pollination:** The type of pollination, whether the crop is primarily self-pollinated or cross-pollinated, influences the crossing strategy.
2. **Type of Crossing Used:** Crossing can be done either artificially (manual emasculation and pollination) or through natural pollination (by wind or insects).
3. **Type of Pollen Dissemination:** The crop may rely on wind (anemophily) or insects (entomophily) for pollen transfer.
4. **Presence of Male Sterility:** The availability of cytoplasmic male sterility (CMS) or genetic male sterility (GMS) systems facilitates hybrid seed production and simplifies crossing.
5. **Objective:** The design depends on whether the goal is breeding for varietal improvement or genetic studies like inheritance pattern analysis, gene mapping, or combining ability studies.

6. **Size of the Population Required:** The population size is determined based on the objectives—larger populations are often needed for quantitative trait analysis or genetic mapping, while smaller populations may suffice for preliminary breeding work.

Genetic Assumptions in Mating Designs

1. **Diploid Behaviour During Meiosis:** All mating designs assume diploid segregation during meiosis. This condition is also applicable to polyploid species if they exhibit disomic inheritance and function like diploids.
2. **Independent Distribution of Genes:** It is assumed that genes influencing the trait of interest are randomly and independently distributed among the parental lines, meaning there is no linkage or correlation among loci.
3. **No Non-Allelic Interactions (No Epistasis):** Ideally, epistatic interactions (interactions between genes at different loci) should be absent. However, in certain mating designs, such as triple test crosses and diallel crosses, epistasis can be detected and estimated if present.
4. **No Multiple Alleles at Trait-Controlling Loci:** The loci governing the character are assumed to carry only two alleles (biallelic), making genetic analysis simpler. Multiple alleles at the same locus complicate the interpretation.
5. **Absence of Reciprocal Differences:** Ideally, the direction of the cross (whether Parent A is the male or female) should not affect the results. Some mating designs test for these differences, and corrections can be applied if they are present.
6. **Use of Homozygous Lines in Diallel Crosses:** Diallel crosses are best conducted with homozygous parents, as heterozygosity can complicate the genetic interpretation. However, some designs can still accommodate heterozygous parents, though with more complex analysis.
7. **No Genotype \times Environment Interaction ($G \times E$):** Mating design analyses assume stable gene expression across environments. If $G \times E$ interactions exist, the material should be evaluated across multiple environments to estimate the extent of this interaction and its influence on trait expression.

Importance of Mating Designs in Plant Breeding

1. **Understanding Genetic Control of Traits:** Mating designs help in revealing the genetic basis of the traits, such as the type of gene action involved (additive, dominance, or epistasis) and the heritability of the character. This knowledge guides the choice of breeding strategies.
2. **Development of Breeding Populations:** They are used to create diverse breeding populations that form the foundation for selection and improvement, ultimately leading to the development of superior varieties.
3. **Estimation of Genetic Gain:** Mating designs enable breeders to predict the potential genetic improvement (genetic gain) that can be achieved through selection in a particular population or breeding scheme.
4. **Evaluation of Parental Lines:** Through systematic crossing and analysis, mating designs help assess the breeding value and combining ability of the parents, aiding in the identification of the most promising lines for further breeding efforts.

Major Mating Designs in Plant Breeding and Genetics

As outlined in key studies (Griffing, 1956b; Kearsey & Pooni, 1996; Hallauer et al., 2010; Acquaah, 2012), several mating designs are widely used in plant breeding and genetic studies. The selection of a mating design depends on the study objectives, biological characteristics of the crop, and available resources (time, space, and cost).

- Bi-Parental Matings
- Polycross
- Top Cross Design
- North Carolina Designs (I, II, III)
- Diallel Design
- Line \times Tester Design

Bi-Parental Matings:

Bi-parental mating is a fundamental controlled crossing strategy used in plant breeding and quantitative genetics for developing segregating populations and for analysing the inheritance of quantitative and qualitative traits. It is regarded as the simplest form of mating design, often referred to as paired crossing (Mather, 1982).

In a bi-parental mating design:

- A breeder randomly selects a set of n individuals from a genetically diverse population.
- These selected individuals are paired in two-parent combinations, such that each pair contributes to producing a full-sib family.
- The number of unique full-sib families produced from n individuals is $\frac{1}{2}n$, assuming each plant is used once in a pairwise cross (Acquaah, 2012).
- Controlled pollination is performed to produce true full-sib progenies, minimizing external contamination and selfing in cross-pollinated crops.

Genetic Analysis

The progenies of these crosses are grown and evaluated for one or more traits. The phenotypic variance (V_p) observed in the progenies is partitioned using Analysis of Variance (ANOVA) into:

- Between-family variance (σ^2_B): Reflects genetic differences between different parental combinations; mainly accounts for additive (A) and dominance (D) variance components.
- Within-family variance (σ^2_W): Captures the segregation variance within a family, contributed by recombination and residual environmental variance.

Thus, the total variance is expressed as:

$$V_P = V_{\text{between families}} + V_{\text{within families}} + V_{\text{environment}}$$
$$V_P = V_{\{\text{between}\ \text{families}\}} + V_{\{\text{within}\ \text{families}\}} + V_{\{\text{environment}\}}$$

(Hill et al., 1998).

Genetic Variance Components Estimated

Bi-parental mating helps estimate:

- Additive genetic variance (VAV_A): The cumulative effect of individual alleles.
- Dominance variance (VDV_D): The interaction between alleles at a locus.
- Environmental variance (VEV_E): The part of the variation due to non-genetic factors.

While epistasis is not directly estimated in simple bi-parental mating, further generations (F_2 , F_3 , recombinant inbred lines) can be used to dissect more complex genetic interactions.

Advantages

- Simplicity: Easy to implement, especially in self-pollinated crops.
- Creation of Segregating Populations: Forms the foundation for generating F_2 populations, backcross populations, and advanced mapping populations like recombinant inbred lines (RILs) or doubled haploids (DHs).
- Genetic Mapping: The resulting populations are ideal for QTL mapping, as they capture recombination events between two distinct genetic backgrounds.
- Estimate Heritability: Can be used to estimate broad-sense and narrow-sense heritability.

Limitations

- Only two parents contribute genetic diversity, limiting the allelic variation present in the segregating population.
- Complex gene interactions (epistasis) may not be fully captured.
- Requires careful selection of parental lines to ensure sufficient genetic contrast.

Applications

- Development of mapping populations for the study of quantitative trait loci (QTL).

- Genetic analysis of key characteristics such as drought tolerance, resistance to diseases, or yield.
- Early-stage breeding to combine desirable traits from two elite parents.

Polycross Mating Design:

Polycross mating is a natural open-pollination breeding method designed for species that are naturally cross-pollinated. It allows the random inter-mating of multiple genotypes, usually cultivars or clones, within an isolated crossing block, ensuring that progeny result from uncontrolled yet genetically meaningful crosses.

Concept and Procedure

- A set of selected parental lines (cultivars or clones) is planted together in an isolated area, preventing pollen contamination from outside sources.
- These entries cross-pollinate naturally, through agents such as wind or insects, within the block.
- Each progeny family is derived from seeds collected from one known female parent, but the male parent is unknown and randomly selected from the block. This results in half-sib families, since the female parent is known but the male parent could be any other plant within the crossing population.
- The term "polycross" specifically refers to the situation where a line or clone undergoes outcrossing with multiple other lines in a poly-parental setup.

Key Applications

- Development of synthetic cultivars in forage crops and vegetatively propagated plants.
- Used in recurrent selection programs for recombining desirable genes across several cycles.
- Helps generate broad-based genetic populations for further selection.

Genetic Implications

- Progeny from each female line in a polycross are half-sibs, having a common maternal parent and varying paternal contributions.
- Variance among progeny can be partitioned into:
 - Between-family variance: Reflecting the differences among maternal parents' genetic contributions.
 - Within-family variance: Representing the segregation of traits due to the genetic diversity of the pollen donors.
- The design primarily estimates General Combining Ability (GCA), which reflects additive genetic effects. GCA is critical in determining the potential of a parent to pass on favorable traits to its offspring.

Suitability and Target Crops

Polycross is particularly well-suited for:

- Obligate cross-pollinators, where controlled hand-pollination is impractical.
- Perennial and vegetatively propagated species, where the same clones can be used over multiple years.
- Examples of crops where polycross designs are commonly used:
 - Forage grasses and legumes
 - Sugarcane
 - Sweet potato
 - Forest trees and shrubs
 - Some tropical fruit trees

Genetic Analysis

The progeny from polycross mating can be analyzed using ANOVA, where the variation can be decomposed as follows (Falconer and Mackay, 1996):

$$VP = V_{\text{between maternal parents}} + V_{\text{within families}} + V_E$$

$$V_P = V_{\{\text{between}\ \text{maternal}\ \text{parents}\}} + V_{\{\text{within}\ \text{families}\}} + V_E$$

where:

- $V_{\text{between maternal parents}}$ $V_{\{\text{between}\ \text{maternal}\ \text{parents}\}}$ represents GCA variance.
- $V_{\text{within families}}$ $V_{\{\text{within}\ \text{families}\}}$ accounts for segregation and environmental variance within each family.
- V_E is the environmental variance.

Estimation of heritability (especially broad-sense heritability) becomes possible through such variance partitioning.

Advantages

- Simpler and more practical for crops where controlled crosses are laborious.
- Efficient for evaluating a large number of genotypes simultaneously.
- Useful for the early screening of parental materials for additive genetic effects.
- Promotes broad genetic recombination, improving the genetic base of breeding populations.

Limitations

- Unknown paternal contribution complicates the estimation of specific combining ability (SCA).
- Possible unequal pollen contribution from different male parents.
- Environmental factors may affect pollen dissemination and crossing success.

Top Cross design

The top cross design is a simple and effective mating approach where selected lines, clones, or inbred plants are crossed with a common tester parent, which could be a variety, inbred line, or hybrid with a well-known genetic background. This design was first introduced in maize breeding by Jenkins and Brunsen (1932) and later termed "top cross" by Tysdal and Grandall

in 1948. The main aim of this design is to assess the general combining ability (GCA) of new lines, helping breeders identify whether a line has favorable additive genes that contribute positively to hybrid performance. In this design, the selected lines are crossed with the tester in a one-way manner ($n \times 1$ crosses), and the resulting progenies form half-sib families, sharing the common tester as the male parent. Top cross is especially useful in the preliminary evaluation of inbred lines or exotic germplasm, requiring less crossing effort and simple data analysis compared to more complex designs like diallel crosses. While it efficiently estimates additive genetic effects (GCA), it does not provide information on specific combining ability (SCA) or gene interactions. This design is widely used in crops like maize and sorghum for early screening of inbred lines before advancing them to single-cross or multi-parent hybrid testing.

North Carolina Designs:

The North Carolina designs, also called biparental crosses or biparental matings, were developed by Comstock and Robinson (1948, 1952) to study genetic variation in plant breeding. These designs are used in the F_2 or later generations of a cross between two pure lines with contrasting traits. In this method, plants are randomly selected from the population and crossed in a specific pattern to generate progenies.

North Carolina designs are versatile tools that help breeders estimate key genetic components like additive variance (from individual gene effects) and dominance variance (from gene interactions). These estimates are made by analyzing half-sib families, where offspring share one parent (either the male or female) but have different other parents.

There are three main types of North Carolina mating designs — Design I, Design II, and Design III (Stuber, 2004; Acquaah, 2007). Each design follows a unique crossing pattern suited for specific genetic analyses. A common feature of all these designs is that the parents are randomly chosen from the base F_2 population, meaning they represent the genetic diversity of the population and are not pre-selected for specific traits.

These mating designs are widely used for partitioning genetic variance and understanding the inheritance of quantitative traits, helping breeders in making effective selections during crop improvement programs.

Diallel Cross Analysis:

Diallel cross analysis is a breeding method where a group of selected parents are crossed in all possible combinations, and the resulting progenies are evaluated. This design helps breeders to assess the combining ability of the parents, meaning how well a parent contributes to its offspring's performance. This method is especially useful for studying polygenic traits in self-pollinated crops. The diallel analysis methods for such traits were first developed by Jinks (1954) and Hayman (1954).

Types of Diallel Cross

1. Full Diallel

In a full diallel, all possible crosses are made in both directions, meaning each parent is used as both male and female.

- Full diallel with parents: Includes parents, direct crosses, and reciprocals. Total entries = p^2 (where p is the number of parents).
- Full diallel without parents: Includes only direct and reciprocal crosses, excluding parents. Total entries = $p(p-1)$.

2. Half Diallel

In a half diallel, crosses are made in only one direction (either male or female is fixed).

- Half diallel with parents: Includes direct crosses and parents. Total entries = $p(p+1)/2$.
- Half diallel without parents: Includes only direct crosses, excluding parents. Total entries = $p(p-1)/2$.

Approaches of Diallel Analysis

Hayman's Graphical Approach:

A visual method to estimate genetic components like dominance and additive effects.

Griffing's Numerical Approach (1956):

A statistical method that estimates General Combining Ability (GCA) and Specific Combining Ability (SCA). Griffing outlined four methods depending on whether parents and reciprocals are included:

- Method 1: p^2 entries (direct, reciprocals, and parents); used to estimate reciprocal effects.
- Method 2: $p(p+1)/2$ entries (direct crosses and parents); used when reciprocal differences are negligible. Most commonly used.
- Method 3: $p(p-1)$ entries (direct and reciprocal crosses); used when parents are excluded due to self-incompatibility.
- Method 4: $p(p-1)/2$ entries (direct crosses only); simplest form.

Combining Ability Models

There are two genetic models for analyzing combining ability:

- Fixed effect model (Model I): Parents are a fixed set of lines; conclusions apply only to these specific lines.
- Random effect model (Model II): Parents are considered random samples from a larger population; results apply to the whole population.

Genetic Information from Diallel Analysis (Griffing's Approach)

In Griffing's approach to diallel analysis, the genetic variation observed among the progenies from different crosses is separated into two main components:

- **Variation among half-sib families:** This represents the General Combining Ability (GCA) and reflects the additive genetic effects. GCA indicates how well a parent contributes its favorable genes to its offspring on average across all crosses.
- **Variation among full-sib families:** This provides the Specific Combining Ability (SCA), which measures the non-additive genetic effects like dominance and epistasis.

SCA captures the unique performance of specific parent combinations that cannot be explained by GCA alone.

Through the estimation of GCA and SCA variances, breeders can understand the relative importance of additive and dominance gene actions controlling the trait. High GCA variance suggests that additive effects are more important, while high SCA variance indicates dominance or interaction effects. This genetic information helps breeders choose suitable parents and decide whether selection or hybrid breeding would be more effective for trait improvement.

Griffings diallel analysis: Data input format (method 2)

This dataset (GriffingData2) contain data on 8 parents which are crossed in diallel fashion (Half diallel with parents). The experimental design was a RCBD with 4 replicates (blocks). It can be retrieved from R (DiallelAnalysisR package) using following code `write_xlsx(GriffingData2," GriffingData2.xlsx")`. Following table shows the sample of input data set.

Cross1	Cross2	Rep	Yield
1	1	1	104.86
1	2	1	88.66
1	3	1	109.76
1	4	1	128.1
1	5	1	128.36
1	6	1	74.4
1	7	1	91.82
1	8	1	48.08
.	.	.	.
.	.	.	.
.	.	.	.
.	.	.	.
.	.	.	.
2	7	1	84.16
2	8	1	96.92
3	3	1	77.94
7	8	4	112.46
8	8	4	81.48

```
#####Diallel Analysis#####
```

```
#####Required packages#####
```

```
install.packages("DiallelAnalysisR")
```

```
library(DiallelAnalysisR)
```

```
#####Importing data#####
```

```
Griffing1Data2<-read.table("clipboard", h=T)
```

```
attach(Griffing1Data2)
```

```
str(Griffing1Data2) library(readxl)
```

```
Griffing1Data2 <- read_excel("Griffing1Data2.xlsx")
```

```
View(Griffing1Data2)
```

```
#####Griffings diallel analysis#####
```

Diallel Analysis with Griffing's Approach Method 2 & Model 1

```
Griffing1Data2 <- Griffing( y = Yield , Rep = Rep , Cross1 = Cross1 , Cross2 = Cross2, data  
= GriffingData2 , Method = 2, Model = 1 )
```

```
names(Griffing1Data2)
```

```
Griffing1Data2
```

```
Griffing1Data2Means <- Griffing1Data2$Means
```

```
Griffing1Data2ANOVA <- Griffing1Data2$ANOVA
```

```
Griffing1Data2Genetic.Components <- Griffing1Data2$Genetic.Components
```

```
Griffing1Data2Effects <- Griffing1Data2$Effects
```

```
Griffing1Data2StdErr <- as.matrix(Griffing1Data2$StdErr)
```

```
sink("Griffing1Data2.txt")
```



```
Griffing( y = Yield , Rep = Rep , Cross1 = Cross1 , Cross2 = Cross2, data = GriffingData2 ,  
Method = 2, Model = 1 )
```

```
names(Griffing1Data2)
```

```
Griffing1Data2
```

```
Griffing1Data2Means <- Griffing1Data2$Means
```

```
Griffing1Data2ANOVA <- Griffing1Data2$ANOVA
```

```
Griffing1Data2Genetic.Components <- Griffing1Data2$Genetic.Components
```

```
Griffing1Data2Effects <- Griffing1Data2$Effects
```

```
Griffing1Data2StdErr <- as.matrix(Griffing1Data2$StdErr)
```

```
sink()
```

Diallel Analysis with Griffing's Approach Method 2 & Model 2

```
Griffing2Data2 <- Griffing( y = Yield , Rep = Rep , Cross1 = Cross1 , Cross2 = Cross2 , data  
= GriffingData2 , Method = 2 , Model = 2 )
```

```
names(Griffing2Data2)
```

```
Griffing2Data2
```

```
Griffing2Data2Means <- Griffing2Data2$Means
```

```
Griffing2Data2ANOVA <- Griffing2Data2$ANOVA
```

```
Griffing2Data2Genetic.Components <- Griffing2Data2$Genetic.Components
```

```
sink("Griffing2Data2.txt")
```

```
Griffing2Data2 <- Griffing( y = Yield , Rep = Rep , Cross1 = Cross1 , Cross2 = Cross2 , data  
= GriffingData2 , Method = 2 , Model = 2 )
```

```
names(Griffing2Data2)
```

```
Griffing2Data2
```

```
Griffing2Data2Means <- Griffing2Data2$Means
```

```
Griffing2Data2ANOVA <- Griffing2Data2$ANOVA
```

```
Griffing2Data2Genetic.Components <- Griffing2Data2$Genetic.Components
```

```
sink()
```

Line × Tester Analysis:

The Line × Tester analysis is an extension of the topcross design, where instead of using just one tester, multiple testers are used to evaluate the combining ability of several lines. This method was first proposed by Kempthorne in 1957. In this design, a set of selected lines (female parents) is crossed with a set of testers (male parents) in a one-to-one manner, producing a total of $f \times m = fm$ crosses, where f is the number of lines and m is the number of testers (Sharma, 2006). For example, if there are 10 lines and 5 testers, the total number of hybrids would be 50.

Line × Tester analysis is considered a simple and efficient mating design that enables breeders to estimate both General Combining Ability (GCA) and Specific Combining Ability (SCA). Unlike the topcross design, which only produces half-sib families, the Line × Tester method generates both full-sibs and half-sibs, allowing for the evaluation of both additive and non-additive genetic effects. This helps breeders in selecting the best parents and identifying the best hybrid combinations for further breeding and crop improvement programs.

Characteristics of a tester

The most desirable tester is one which provides maximum information about the performance of a line in cross combination under different environmental conditions.

- Broad genetic base
- Wider adaptability
- Low yield potential
- Low performance for other traits

Components of genetic variance in Line × Tester Analysis

Line × Tester Analysis partitions the variation of single crosses into three fractions viz., variation among male parents, variation among female parents and variation due to interaction of male and female parents.

Source	df	MS	Expected mean squares	
			Model I	Model II
Replication	$r - 1$			
Lines	$m - 1$	M_1	$\sigma^2 + rf \frac{1}{m-1} + \sum_i s_i^2$	$\sigma^2 + v_{sca} + rf_{gca(m)}$
Testers	$f - 1$	M_2	$\sigma^2 + rm \left(\frac{1}{f-1} \right) \sum_j s_j^2$	$\sigma^2 + rv_{sca} + rm_{gca(f)}$
Line x tester	$(m-1)(f-1)$	M_3	$\sigma^2 + r \left[\frac{1}{(m-1)(f-1)} \right] \sum_i \sum_j s_{ij}^2$	$\sigma^2 + rv_{sca}$
Error	$(r-1)(mf-1)$	M_4	σ^2	σ^2

Source: Sharma (2006)

Genetic Information from Line × Tester Analysis

The Line × Tester analysis provides valuable insights into the genetic control of traits by partitioning the total variation into:

- **General Combining Ability (GCA):** Reflects the additive genetic variance (VA) and represents the average performance of a line or tester across all its crosses. It is estimated from the covariance among half-sibs (Cov. HS).
- **Specific Combining Ability (SCA):** Represents the dominance or non-additive genetic variance (VD) and shows the unique performance of a specific cross beyond what is expected from the GCA of the parents. SCA is estimated as the difference between the covariance among full-sibs and twice the covariance among half-sibs (Cov. FS – 2 Cov. HS).

Thus, the genetic variances are related as:

- Additive variance (VA) \approx GCA variance
- Dominance variance (VD) \approx SCA variance

Through these estimates, breeders can determine whether additive effects (which can be fixed through selection) or dominance effects (exploited in hybrids) are more important for a

particular trait. This helps in selecting the best parents and in deciding whether to follow a breeding strategy focused on selection or hybrid development.

Line × Tester Analysis: Input data format

This dataset (LTdata) contain data on 5 lines with 3 testers. The experimental design was a RCBD with 4 replicates (blocks). It can be retrieved from R (LxT function, agricolae package) using following code write_xlsx(LxT," LTdata.xlsx"). Following table shows the sample of input data set.

replication	line	tester	yield
1	1	6	74.4
2	1	6	70.86
3	1	6	60.94
4	1	6	68
.	.	.	.
.	.	.	.
.	.	.	.
.	.	.	.
4		7	89.46
1		8	91.78
2		8	84.82
3		8	69.92
4		8	81.48

```
#####LinuxTesterAnalysis#####
```

```
#####Required packages#####
```

```
install.packages("agricolae")
```

```
library(agricolae)
```

```
#####Importing data#####
```

```
LTdata<-read.table("clipboard", h=T)
```

```
attach(LTdata)
```

```
str(LTdata)
```

```
library(readxl)
```

```
LTdata <- read_excel("LTdata.xlsx")
```

```
View(LTdata)
```

```
#####LinuxTesterAnalysis#####
```

```
output2<-with(LTdata,lineXtester(replication, line, tester, yield))
```

```
sink("output2.txt")
```

```
with(LTdata,lineXtester(replication, line, tester, yield))
```

```
output2
```

References;

Muthoni, J. and Shimelis, H., 2020, Mating designs commonly used in plant breeding: A review. Austarlian Journal of Crop Science, 14 (12): 1855-1869. 10.21475/ajcs.20.14.12.p2588.

Nduwumuremyi, A., Tongoona, P. and Habimana, S., 2013, Mating designs: Helpful tool for quantitative plant breeding analysis. Journal of Plant Breeding and genetics, 1 (3): 117-129.

Singh R.K. and Chaudhary, B.D., 1979, Biometrical Methods in Quantitative Genetic Analysis Kalyani Publishers.

Singh, P. and Narayanan, S. S., 1993, Biometrical techniques in plant breeding, Kalyani Publishers.

Advanced Concepts and Applications of Path Coefficient Analysis in Agricultural Research

Vinayaka

ICAR- Sugarcane Breeding Institute, Coimbatore, Tamil Nadu, 641007

Email: vinayaka.b3vs@gmail.com

1. Introduction

Understanding the interrelationships among multiple traits and their combined impact on a target trait, such as crop yield, is crucial in agricultural research. Traditional correlation analysis, while useful, fails to distinguish between direct and indirect relationships. Path coefficient analysis (PA), introduced by Sewall Wright (1920), overcomes this limitation by partitioning correlation coefficients into direct and indirect effects, offering clearer insight into causal relationships among traits. Dewey and Lu (1959) later adapted this method for plant breeding, marking the beginning of its widespread use in agriculture.

2. Concept and Scope of Path Coefficient Analysis

PA is a form of multiple regression analysis that quantifies the magnitude and direction of effects among variables using standardized partial regression coefficients. The major strength of PA lies in its ability to dissect a correlation into its causal components. According to Lleras (2005), PA enables researchers to distinguish whether a variable's influence on the dependent variable is direct or mediated through other variables.

In plant breeding, PA plays a critical role in identifying traits that significantly influence yield and should therefore be emphasized in selection programs (Singh & Narayanan, 1993).

3. Historical Development and Theoretical Foundations

The path analysis technique was initially developed for biological problems in genetics and evolution. Wright's (1920) original use involved partitioning correlation coefficients to understand heritable traits in guinea pigs. Subsequently, Blalock (1961) and Duncan (1966) extended PA to social sciences, enabling causal modelling in non-experimental designs.

In agriculture, Dewey and Lu (1959) applied PA to examine yield components in crested wheatgrass, setting a precedent for later studies in cereals, legumes, and oilseed crops.

4. Assumptions and Requirements of Path Analysis

Path analysis relies on several statistical assumptions:

- i. **Linearity and additivity:** All causal relationships are linear and additive.
- ii. **No multicollinearity:** Predictor variables should not be highly correlated.
- iii. **Recursive models:** Feedback loops are not permitted.
- iv. **Causal closure:** All important variables influencing the dependent variable must be included.

Failing to meet these assumptions can bias estimates of path coefficients, leading to misleading conclusions (Mahapatra *et al.*, 2020).

5. Types of Path Coefficients

Path coefficients can be derived under different conditions:

- **Phenotypic Path Coefficients:** Based on phenotypic correlations.
- **Genotypic Path Coefficients:** Based on genotypic correlations and more stable across environments.
- **Environmental Path Coefficients:** Derived from environmental correlations and help separate genotype-by-environment effects.

Among these, genotypic path coefficients are preferred in breeding programs due to their stronger reflection of inheritable traits (Singh and Chaudhary, 1979).

6. Methodological Framework

6.1 Data Collection and Preparation

Replicated trials with genetically diverse genotypes are essential for reliable estimation. Traits measured should include both yield and contributing characteristics such as plant height, biomass, and harvest index.

6.2 Statistical Procedure

- i. **Estimate variance and covariance matrices**
- ii. **Compute correlation coefficients**
- iii. **Calculate path coefficients using the formulae**

PA represents a standardized method of partitioning the partial regression coefficient, distributing the correlation coefficient into various measures of direct and indirect impacts of a set of independent variables on the dependent variable, which is yield. It is also known as cause-and-effect relationship. If a character is determined by the correlated characters, a path diagram must be formulated. Thus, we get a set of simultaneous equations as mentioned below:

$$R(X_1, Y) = a + r(X_1, X_2)b + r(X_1, X_3)c$$

$$R(X_2, Y) = r(X_1, X_2)a + b + r(X_2, X_3)c$$

$$R(X_3, Y) = r(X_1, X_3)a + r(X_2, X_3)b + c$$

Direct and indirect effects of characters using path coefficient analysis: The direct and indirect effects both at genotypic and phenotypic levels were estimated by taking quantitative response as dependent variable, using path coefficient analysis (Wright, 1921), (Dewey and Lu, 1959). More generalized case, the following equations were formed and solved simultaneously for estimating the various direct and indirect effects.

$$r_{1y} = P_{1y}r_{11} + P_{2y}r_{12} + P_{3y}r_{13} \dots + P_{ny}r_{1n}$$

$$r_{2y} = P_{1y}r_{21} + P_{2y}r_{22} + P_{3y}r_{23} \dots + P_{ny}r_{2n}$$

$$r_{ny} = P_{1y}r_{n1} + P_{2y}r_{n2} + P_{3y}r_{n3} \dots + P_{ny}r_{3n}$$

where 1, 2, ..., = independent variables; = dependant variable; $r_{1y}, r_{2y}, \dots, r_{ny}$ = Coefficient of correlation between casual factors 1 to on dependent character ; $P_{1y}, P_{2y}, \dots, P_{ny}$ = Direct effect of character 1 to on character . Considering the simultaneous equations given above can be matrix notation as: $CB = A$,

$$\begin{bmatrix} 1 & r_{12} & r_{13} & \dots & r_{1n} \\ r_{21} & 1 & r_{23} & \dots & r_{2n} \\ r_{31} & r_{32} & 1 & \dots & r_{3n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ r_{n1} & r_{n2} & r_{n3} & \dots & 1 \end{bmatrix} \begin{pmatrix} P_{1y} \\ P_{2y} \\ P_{3y} \\ \vdots \\ P_{ny} \end{pmatrix} = \begin{pmatrix} r_{1y} \\ r_{2y} \\ r_{3y} \\ \vdots \\ r_{ny} \end{pmatrix}$$

$$\text{Then, } B = C^{-1}A, \text{ where } C^{-1} = \begin{bmatrix} c_{11} & c_{12} & c_{13} & \dots & c_{1n} \\ c_{21} & c_{22} & c_{23} & \dots & c_{2n} \\ c_{31} & c_{32} & c_{33} & \dots & c_{3n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ c_{n1} & c_{n2} & c_{n3} & \dots & c_{nn} \end{bmatrix}$$

Direct effects were as follows:

$$P_{1y} = \sum_{i=1}^k c_{1i}r_{iy}, P_{2y} = \sum_{i=1}^k c_{2i}r_{iy}, \dots, P_{ny} = \sum_{i=1}^k c_{ni}r_{iy}$$

Residual effect, which measures the contribution of characters not considered, was obtained as:

$$P_{ry} = \sqrt{1 - \left(\sum_{i=1}^n P_{iy} r_{iy} \right)}$$

where, P_{ny} = direct effect of X_n on Y , and r_{ny} = correlation coefficient of X_n on Y .

6.3 Software Tools

- R Packages:** variability, agricolae
- SAS, SPSS, Genstat:** Also support path analysis via correlation and regression modules

The R code provided in Bharamappanavara (2023) simplifies the process for both replicated and unreplicated data, showcasing its utility in real-time breeding datasets.

6.4 Dataset and R Code for Replicated Data Path Analysis

Replicated data (trait names shortened for clarity) is given, that is, dependent variable: Grain yield (GY) and independent variables: DFF (Days to Fifty percent Flowering), PH (Plant Height), PL (Panicle Length), PW (Panicle Weight), HI (Harvest Index), GY (Grain Yield), etc.

Genotypes	Rep	DFF	PH	PL	PW	HI	TW	MILL	HRR	GY
G1	R1	122	104.6	21.2	81	42.59	22	86.42	30.75	61.2
G1	R2	120.4	90.4	22.4	86	45.51	20	67.48	39.32	65.3
G1	R3	120	98.6	22.8	79	43.2	20	70.48	42.32	61.3
.
.
.
.
G10	R1	124.6	94.6	20.7	100	54.48	13	80.57	35.9	71.8
G10	R2	128	87.2	20.2	106	59.06	12	61.29	31.5	74.3
G10	R3	133	87.6	19.8	111	55.92	12	64.29	30.57	68.5

R Code:

```
library(variability) #Required packages
library(variability)
```

```

#Importing data into R studio #
vardata<-read.delim("clipboard")
attach(vardata)
str(vardata)

# Alternate way of importing...
library(readxl)
vardata <- read_excel("vardata.xlsx")
view(vardata)

#Estimation of genetic parameters#
genvar <- gen.var(vardata[3:11], vardata$Genotypes, vardata$Replication)
genvar
sink("genvar.txt")
print(genvar)

#Genotypic and phenotypic correlations#
gencor <- geno.corr(vardata[3:11], vardata$Replication)
gencor
sink("gencor.txt")
print(gencor)
phecor<-pheno.corr(vardata[3:11], vardata$Replication)
phecor
vardata$Genotypes
vardata$Genotypes
sink("phecor.txt")
print(phecor)

#Genotypic and phenotypic path coefficients#
genpath<-geno.path(vardata[11], vardata[3:10], vardata$Genotypes, vardata$Replication)
genpath
sink("genpath.txt")
print(genpath)
phepath<-pheno.path(vardata[11], vardata[3:10], vardata$Genotypes, vardata$Replication)
phepath
sink("phepath.txt")
print(phepath)

#Path analysis for unreplicated data#
#Importing data set using following code or import from excel#
path=read.delim("clipboard")
attach(path)
str(path)

#load required package#
require(agricolae)

```

```
#Let us calculate correlation co-efficient#
x<-path[,c(1,2,3,4,5,6,7,8)] #here we have to define independent variables
y<-path[,9] #here we have to define dependent variable#
cor.y<-correlation(y,x)$correlation
cor.x<-correlation(x)$correlation

#Let us calculate calculate direct and indirect effects#
pathresult<-path.analysis(cor.x,cor.y)

#Let us save analysed data in table.csv#
write.table(pathresult,file="pathresults.csv", sep="," , col.names=NA, qmethod="double")
```

7. Interpretation of Results

- **High direct effect & high correlation:** Trait is a true contributor; prioritize in selection.
- **Low direct effect & high correlation:** Correlation is likely due to indirect effects.
- **High direct effect & low correlation:** Other traits suppress the trait's potential impact.

Residual effect estimates indicate the unexplained variation. If high, it suggests missing variables in the model.

8. Applications in Recent Research

8.1 Cereals

- **Rice:** Traits like panicle length, grains per panicle, and harvest index showed high direct effects on grain yield (Chakravorty *et al.*, 2021).
- **Wheat:** Flag leaf area and biological yield demonstrated substantial direct and indirect effects (Kumar *et al.*, 2022).

8.2 Pulses and Legumes

- **Chickpea:** Pod number and biomass yield are influential traits (Rani *et al.*, 2020).
- **Soybean:** Studies emphasize indirect effects of plant height via branches and pods (Kaur *et al.*, 2018).

8.3 Oilseeds and Fiber Crops

- **Sunflower:** Head diameter and seed filling duration positively affect yield (Mohana *et al.*, 2020).
- **Cotton:** Boll weight and number per plant are crucial for selection (Zhou *et al.*, 2021).

9. Challenges and Limitations

- **Violation of assumptions:** Linear additive models may not hold in polygenic systems.
- **Omitted variable bias:** Unmeasured traits can distort results.
- **Overfitting:** Including too many predictors can complicate interpretation.

Despite these, PA remains invaluable when applied cautiously with appropriate domain knowledge.

10. Future Directions

The integration of PA with machine learning and multi-trait selection indices is emerging. Bayesian path analysis and structural equation modelling (SEM) offer robust alternatives that handle latent variables and model complexity.

For instance:

- Bayesian SEM models allow uncertainty quantification (Lee *et al.*, 2019).
- Genome-wide PA is helping breeders link genomic regions to yield components (Rani *et al.*, 2023).

11. Conclusion

Path coefficient analysis remains a cornerstone in quantitative genetics and breeding research. By decomposing correlation into meaningful components, it aids breeders in pinpointing critical traits for selection. However, it requires careful design, proper data, and prudent interpretation. With advancements in statistical software and computational biology, the power and precision of path analysis in agronomy continue to expand.

References

- Bharamappanavara, M. (2023). Path Coefficient Analysis. In Rathod *et al.*, *Advanced Statistical Tools and Techniques for Biometrical Data Analysis*, ICAR-IIRR, pp. 35–42.
- Blalock, H.M. (1961). *Causal Inferences in Nonexperimental Research*. University of North Carolina Press.
- Chakravorty, S., Bera, S. and Das, S. (2021). Path coefficient and multivariate analysis for yield and its component traits in rice. *Journal of Genetics*, **100(21)**. DOI: 10.1007/s12298-020-00903-7.
- Dewey, D.R. and Lu, K. (1959). A correlation and path-coefficient analysis of components of crested wheatgrass seed production. *Agronomy Journal*, **51(9)**, 515–518. DOI: 10.2134/agronj1959.00021962005100090002x.
- Duncan, O.D. (1966). Path analysis: Sociological examples. *American Journal of Sociology*, **72(1)**, 1–16.
- Kaur, A., Sangha, M.K. and Kaur, G. (2018). Correlation and path coefficient analysis in soybean [*Glycine max* (L.) Merrill] under different environmental conditions. *Legume Research*, **41(1)**, 96–101. <https://doi.org/10.18805/LR-3785>.
- Kumar, A., Verma, R.P.S., Singh, M., Sharma, S.K. and Kumar, R. (2022). Dissecting direct and indirect contributions of yield-related traits in bread wheat (*Triticum aestivum* L.) using correlation and path coefficient analysis. *Molecular Breeding*, **42**, 123–134. <https://doi.org/10.1007/s11032-021-01215-0>.
- Lee, S.Y., Song, X.Y. and Lee, J.C.K. (2019). *Bayesian Structural Equation Modelling with Applications in Health and Behavioural Sciences*. Wiley Series in Probability and Statistics. ISBN: 978-1-119-42322-2.
- Lleras, C. (2005). Path analysis. In *Encyclopedia of Social Measurement*, Vol. 3, pp. 25–30. Elsevier.
- Mahapatra, S.K., Das, S., Mohanty, S. and Dash, A. (2020). Path analysis and its application in agriculture. *International Journal of Agriculture and Plant Science*, **2(2)**, 1–3.

- Mohana, T., Singh, D., Singh, M. and Sharma, S. (2020). Genetic variability, correlation and path coefficient analysis in sunflower (*Helianthus annuus L.*). *Helia*, **43(72)**, 97–108. <https://doi.org/10.2298/HEL2072097M>.
- Rani, M., Singh, R.K., Kumar, P., Yadav, R.K. and Sharma, P. (2023). Application of genome-wide path coefficient analysis for dissecting the genetic basis of grain yield in rice (*Oryza sativa L.*). *Theoretical and Applied Genetics*, **136**, 52.
- Singh, P. and Narayanan, S.S. (1993). *Biometrical Techniques in Plant Breeding*. Kalyani Publishers.
- Singh, R.K. and Chaudhary, B.D. (1979). *Biometrical Methods in Quantitative Genetic Analysis*. Kalyani Publishers.
- Wright, S. (1920). The relative importance of heredity and environment in determining the piebald pattern of guineapigs. *Proceedings of the National Academy of Sciences*, **6(6)**, 320–332.
- Zhou, L., Wang, L., Zhang, J., Zeng, H., Wang, X. and Yu, S. (2021). Genomic prediction of fiber yield in cotton using path coefficients. *The Plant Genome*, **14(3)**, e20100. <https://doi.org/10.1002/tpg2.20100>.

Genotype-By-Environment Interaction and Stability: Concepts, Analysis, And Applications

Ravi V. Mural

Department of Agronomy, Horticulture & Plant Science, Berg Agricultural Hall 208, Box, 2100A, Brookings, SD 57007

Email: ravi.mural@sdstate.edu

Introduction

The phenotype of a plant is determined by the complex interplay of its genotype (G) and the environment (E), with the potential for significant genotype-by-environment interaction ($G \times E$ or GEI). While the genetic makeup of an individual remains largely constant, the expression of traits can vary widely depending on environmental conditions. This means that the same genotype may exhibit different phenotypes in different settings, a phenomenon that lies at the heart of plant breeding and crop improvement.

Understanding $G \times E$ is crucial because it complicates the identification of superior genotypes. A variety that excels in one environment may perform poorly in another, not necessarily due to genetic inferiority, but because of specific adaptation to local conditions. Thus, breeders must evaluate genotypes across multiple environments to select those with either broad or specific adaptation, depending on breeding goals.

Key Definitions: $G \times E$, Stability, Adaptability, and Plasticity

In plant breeding and crop research, evaluating the performance of genotypes across multiple environments is fundamental. However, performance is rarely uniform across locations or seasons due to the complex interplay between genotype and environment. To interpret and harness this variation, we rely on four closely related but conceptually distinct terms: **genotype-by-environment interaction ($G \times E$)**, **stability**, **adaptability**, and **plasticity**. Each concept offers a different lens through which genotype performance can be assessed and understood:

- **Genotype-by-environment interaction ($G \times E$)** refers to the phenomenon in which the performance of genotypes changes relative to each other across environments. In other words, the effect of a genotype is not consistent in all conditions. If the ranking of genotypes shifts significantly from one environment to another, a strong $G \times E$ interaction is present. This interaction can be **crossover**, where genotype rankings

change across environments, or **non-crossover**, where performance changes in magnitude but not in rank. G×E complicates selection because a top-performing genotype in one location may not maintain superiority elsewhere. As a result, G×E is the underlying reason why plant breeders conduct multi-environment trials (METs).

- **Stability** refers to the consistency or predictability of a genotype's performance across diverse environments. A stable genotype shows minimal fluctuation in traits like yield despite environmental variation and contributes little to the G×E interaction.
- There are two main types of stability:
 - **Static (biological) stability** implies constant performance across environments, regardless of changes in environmental productivity. This type is ideal for traits like grain quality, where uniformity is preferred. However, it may not be desirable for yield if it results in low and unresponsive performance.
 - **Dynamic (agronomic) stability**, which is more relevant for yield and other quantitative traits, allows genotypes to respond positively to favorable environments while maintaining their relative superiority across conditions. A dynamically stable genotype tends to perform well in both high- and low-yielding environments and consistently ranks among the top performers.

While stability helps identify reliable genotypes, a stable genotype is not necessarily high yielding—it may simply perform predictably across sites..

- **Adaptability** captures the **capacity of a genotype to perform well in specific or diverse environments**. Unlike stability, which concerns performance variance, adaptability emphasizes **performance level**. A genotype with broad adaptability maintains high yield across many environments, while a genotype with specific adaptability excels in a limited set of environmental conditions (e.g., drought-prone or saline soils). In practice, adaptability is a more practical goal for breeders than stability alone, especially when targeting specific agroecological zones or production constraints. Notably, a genotype can be stable yet poorly adapted (yielding consistently low) or highly adapted to one environment but unstable overall.
- **Plasticity** on the other hand, refers to a genotype's **phenotypic responsiveness** to environmental conditions. It is the ability of a genotype to modify its growth, development, or yield in response to changes in the environment. Plasticity is not

inherently good or bad—it simply describes the degree to which a genotype can adjust its phenotype. High plasticity may be desirable in some cases (e.g., adjusting flowering time under different daylengths), but undesirable in others where consistency is preferred (e.g., grain quality traits). Importantly, plasticity is a **trait-level** property and not a direct measure of performance or superiority. It is often studied using reaction norms or slope-based models that capture how trait values change along an environmental gradient.

These four concepts are conceptually interconnected. G×E is the foundation—it explains why performance varies. Stability focuses on minimizing that variation, adaptability aims to exploit it in the most beneficial way, and plasticity describes how traits shift in response to it. Understanding and distinguishing among these terms is essential when designing breeding strategies, analyzing MET data, or making recommendations for cultivar release.

To help clarify the distinctions, the following table provides a side-by-side comparison of these four terms:

Attribute	G×E Interaction	Stability	Adaptability	Plasticity
Definition / Focus	Variation in genotype performance across environments	Consistency in genotype performance across environments	Performance level relative to environmental conditions	Phenotypic flexibility in response to environmental cues
Interpretation	Inconsistency in genotype rankings; underlies both stability and adaptability	Predictable performance; may be static (minimal change) or dynamic (predictable change)	Genotype fit to broad or specific environments; high performance when conditions are optimal	Ability to adjust traits like flowering or height; may be adaptive or non-adaptive
Goal in Breeding	Understand environmental sensitivity and manage variability	Ensure reliable performance across locations and years	Match genotypes to broad or specific target populations of environments (TPEs)	Enhance resilience, flexibility, or stress responsiveness
When Desired	When targeting diverse or highly variable environments	When consistent yield is required across unpredictable climates	When breeding for favorable, stress-prone, or high/low input zones	When breeding for climate resilience, stress tolerance, or variable growing conditions

Measure ment / Indicators	ANOVA interaction terms, mixed models, AMMI, GGE biplots	Regression slope ≈ 1 , low deviation, AMMI stability value, Shukla's variance, Kang's yield index	Mean performance, Finlay–Wilkinson regression slope (>1 = high adaptability), yield response	Reaction norms, slope of trait vs. environment, trait variance across environments
Example Use Case	Identifying stable or specifically responsive genotypes for regional recommendations	Selecting cultivars for multi-location or multi-year trials in variable climates	Recommending varieties tailored for optimal or stress-prone regions	Selecting genotypes that shift flowering under drought or delay emergence under cold

Experimental Design and Measurement

To assess stability and G×E interaction, breeders conduct multi-environment trials (METs) across different locations, years, or seasons. Common designs such as randomized complete block and lattice designs are used, particularly when evaluating a large number of genotypes. These trials help control environmental variation and generate the data necessary for robust statistical analysis of G×E and genotype stability.

Stability analysis relies on statistical models to partition phenotypic variance into components attributable to genotype, environment, and their interaction. The basic model is:

$$P_{ij} = \mu + G_i + E_j + (GE)_{ij}$$

Where,

P_{ij} is the phenotype of genotype i in environment j ,

μ is the overall mean,

G_i is the effect of the i^{th} genotype,

E_j is the effect of the j^{th} environment, and

$(GE)_{ij}$ is the interaction effect.

Table: Example of Genotype by environment interaction

Genotype	Environment		Difference
	E1	E2	(E effect)
G1	a	c	$\Delta 1 = c - a$
G2	b	d	$\Delta 2 = d - b$

Difference (G effect) $\Delta 3 = b - a$ (in E1), $\Delta 4 = d - c$ (in E2)

$$\begin{aligned} \text{GE interaction} &= (\Delta 2 - \Delta 1) = (\Delta 4 - \Delta 3) \text{ or } (d - b) - (c - a) = (d - c) - (b - a) \\ &= (\Delta 1 + \Delta 4) = (\Delta 2 + \Delta 3) \text{ or } (c - a) + (d - c) = (d - b) + (b - a). \end{aligned}$$

The genotype effect, $\Delta 3$, represents the difference between genotypes in environment E1 and $\Delta 4$ represents the difference between genotypes in environment E2. The environmental effect, $\Delta 1$, represents change attributable to environments for genotype G1 and $\Delta 2$ is the change attributable to environments for genotype G2.

$$\text{Total effect (T)} = G + E + GE = (d - a) \text{ or } GE = T - G - E.$$

Types of G×E Interaction: There are generally two forms of G×E interaction: **crossover** and **non-crossover** interaction. Crossover interaction occurs when the rank order of genotype performance changes from one environment to another. For example, Genotype A may outperform Genotype B in one location but not in another. This type of interaction can pose major challenges to breeding programs aimed at wide adaptation. Non-crossover interaction, on the other hand, occurs when performance changes in magnitude but not in rank, for instance, a high-performing genotype remains the top performer, although the yield level fluctuates.

Causes of G×E Interaction: G×E occurs when different genotypes respond differently to changes in environmental conditions. These interactions make it challenging to predict how a genotype will perform across diverse environments. Understanding the causes of G×E is critical for breeders aiming to develop varieties with broad adaptability or targeted performance in specific conditions.

- **Abiotic Stresses:** Abiotic stresses, such as drought, heat, cold, or nutrient deficiencies are major contributors to G×E interaction. When environmental factors deviate from

optimal levels, plants experience stress, and genotypes often differ in their ability to tolerate or respond to such conditions. For example, one genotype may maintain yield under drought by closing its stomata early, while another may fail to limit water loss and experience a drop in productivity. Similarly, some genotypes may exhibit symptoms of iron or boron deficiency in low-fertility soils, while others remain unaffected due to more efficient nutrient uptake mechanisms.

In many cases, these differences are genetically controlled and may be as simple as a single gene conferring tolerance to a particular stress. This explains why two otherwise similar genotypes may perform quite differently under stress but similarly in optimal environments.

At the cellular level, abiotic stress often triggers the accumulation of reactive molecules capable of damaging proteins, nucleic acids, and membranes. The degree of damage and the plant's ability to mitigate it can differ substantially among genotypes, contributing to observed interaction effects.

- **Biotic Stresses:** Biotic factors, such as diseases and insect pests, also drive $G \times E$ interactions. Resistance or susceptibility to pathogens often varies among genotypes. For instance, a sorghum line resistant to leaf blight may outperform others in environments with high disease pressure but show no advantage in disease-free environments. These differences in resistance can lead to "crossover" interactions, where the best-performing genotype in one environment is not the best in another.

Thus, understanding these causes helps breeders predict genotype performance under various conditions and informs the development of cultivars with either broad or targeted adaptation.

Adaptive Plasticity and Genotypic Variation due to Quantitative Nature of $G \times E$: A major contributor to genotype \times environment interaction is phenotypic plasticity—the ability of a genotype to modify its traits in response to environmental stimuli. This plasticity allows plants to adjust their growth, development, or physiology depending on environmental conditions, such as drought, nutrient availability, or temperature extremes. For instance, some maize genotypes may flower earlier under drought to escape stress, while others maintain fixed flowering times regardless of moisture availability.

Plasticity is closely tied to adaptability, a quantitative trait influenced by numerous genes. The extent and sophistication of a genotype's plastic responses determine whether it exhibits stable performance (i.e., low sensitivity to environmental variation) or high adaptability (i.e., strong positive response to improved conditions). Genotypes with high plasticity tend to perform well across variable environments, whereas those with low plasticity may show more consistent but less responsive behavior.

Importance in Plant Breeding: The presence of $G \times E$ interaction underscores the importance of multi-environment testing (MET) in crop improvement programs. Without evaluating genotypes across diverse environments, it is impossible to determine whether superior performance is due to genuine genetic advantage or favorable environmental conditions. MET helps in selecting genotypes that are either broadly adapted — performing well across most environments — or specifically adapted — excelling in targeted environmental niches. Stability analysis is therefore often coupled with $G \times E$ analysis to quantify how consistent a genotype's performance is across sites or seasons.

Implications for Breeding and Stability Analysis: GEI presents both challenges and opportunities in plant breeding. One of its primary consequences is a reduction in trait heritability, as it increases the environmental influence on phenotypic expression and confounds the partitioning of genotypic and environmental effects. This weakens the correlation between phenotypic and genotypic values, making it more difficult to accurately identify superior genotypes, especially when crossover interactions cause genotypic rankings to shift across environments.

In the context of breeding, GEI complicates selection decisions and reduces the effectiveness of evaluating genotypes in a single environment. Multi-environment trials (METs) become essential to distinguish true genetic potential from environmental noise. Breeders must then choose between two strategic approaches: (1) selecting for broad adaptation, where genotypes maintain high mean performance and stability across environments, or (2) pursuing specific adaptation, where genotypes are tailored to excel in particular target environments, even if their performance elsewhere is lower.

The magnitude of $G \times E$ also influences breeding program design and resource allocation. Developing separate breeding populations for different regions can yield greater genetic gains in the presence of strong interaction effects, particularly when genotypic rankings vary

drastically across locations. However, this approach is resource intensive. Conversely, selecting for general adaptation across multiple environments is more cost-effective, but may result in slower genetic progress due to reduced selection pressure for local adaptation.

To address these challenges, breeders have developed and adopted a suite of statistical tools collectively known as stability analysis, which includes both univariate and multivariate methods. These tools help quantify G×E and identify genotypes that are either stable (i.e., consistently performing across environments) or specifically adapted to certain environments. Stability analysis enhances the efficiency of selection decisions, enabling breeders to more effectively develop cultivars suited to variable or targeted environments.

Ultimately, understanding the quantitative nature of G×E and its implications helps breeders make informed decisions about trial design, selection strategies, and resource investment—ensuring that breeding programs are both scientifically sound and strategically aligned with environmental variability.

Statistical Methods for Analyzing G×E and Stability: Understanding and quantifying G×E interactions is essential in modern plant breeding. G×E complicates the selection of superior genotypes by reducing trait heritability and masking genotypic effects. Stability analysis provides tools to address these challenges, offering insight into genotype performance consistency across environments and guiding selection strategies based on either broad or specific adaptation.

A variety of statistical approaches are used to dissect G×E and assess stability:

- **Foundational Models: ANOVA and Mixed Models:** The most basic approach to analyzing G×E interaction is through **analysis of variance (ANOVA)** or **linear mixed models**, which partition total phenotypic variation into components attributable to genotype (G), environment (E), and their interaction (G×E). These models can determine whether G×E interaction is statistically significant, but they do not explain the nature of the interaction or the behavior of individual genotypes across environments.

To address these limitations, breeders turn to more advanced statistical frameworks that provide deeper insights into genotype stability, adaptability, and responsiveness.

- **Regression-Based Approaches:** Regression-based models—such as Finlay–Wilkinson and Eberhart–Russell—assess stability by regressing genotype performance on environmental means. A regression slope near 1 with low deviation from regression indicates average stability and broad adaptability. Genotypes with slopes < 1 tend to be more stable but less responsive to favorable environments, while slopes > 1 indicate high responsiveness but reduced stability. Eberhart–Russell also adds a stability variance term to quantify unpredictable behavior.
- **Variance-Based Indices:** Several variance-based indices quantify the contribution of individual genotypes to the overall $G \times E$ interaction:
 - Wricke’s ecovalence measures each genotype’s contribution to the interaction sum of squares.
 - Shukla’s stability variance estimates stability while accounting for environmental heterogeneity.
 - The coefficient of variation (CV) provides a simple measure of performance variability.

These indices help breeders distinguish between genotypes that are stable across environments and those that show high sensitivity or specific adaptation.

- **Multivariate Models and Biplot Analysis:** Advanced multivariate models are widely used for visualizing and interpreting $G \times E$ interactions:
 - **AMMI (Additive Main Effects and Multiplicative Interaction)** combines ANOVA with principal component analysis (PCA) to decompose $G \times E$ into interpretable components. Biplots generated from AMMI help identify genotypes with general or specific adaptability—those near the origin are considered stable, while those farther away may be specifically adapted to certain environments.
 - **GGE (Genotype + $G \times E$) Biplot** focuses on both genotype main effects and $G \times E$ interaction. It reveals “which-won-where” patterns and helps define mega-environments, making it a valuable tool for regional variety recommendations.
- **Stability Indices and Univariate Metrics:** Stability indices provide breeders with simplified metrics to rank genotypes based on both mean performance and consistency:

- Kang's yield-stability statistic combines mean performance and stability variance to rank genotypes for selection. For example, it combines yield and a stability measure (e.g., Shukla's variance) into a single index.
- Other widely used indices include:
 - Wricke's ecovalence: measuring each genotype's contribution to interaction sum of squares or the $G \times E$,
 - Shukla's stability variance (adjusted for environmental effects) estimates stability while accounting for environmental heterogeneity,
 - Coefficient of variation (CV) provides a simple measure of performance variability.

These univariate metrics are especially helpful in practical breeding decisions when a single index is needed to compare multiple genotypes across diverse environments.

Modern Tools and Practical Applications: Modern breeding increasingly relies on **linear mixed models** with **random effects**, **Best Linear Unbiased Predictors (BLUPs)**, **Bayesian approaches**, and **reaction norm models**, which model genotype performance across environmental gradients (e.g., temperature, rainfall). These methods allow breeders to predict performance in **untested environments**, improving selection accuracy and facilitating **genomic selection** pipelines.

Such models are especially valuable in large-scale breeding programs where the incorporation of **environmental covariates** improves the understanding of genotype responsiveness under complex, real-world conditions.

Software and Computational Tools: The R programming language offers a powerful platform for $G \times E$ and stability analysis, supported by numerous packages, including:

- metan: for MET analysis, stability indices, and AMMI/GGE visualization
- agricolae: for traditional ANOVA and index computation
- lme4 and asreml: for linear and mixed model fitting
- GGEbiplot and GGEbiplotGUI: for visual biplot construction
- stability: for comprehensive stability analysis (Finlay–Wilkinson, Eberhart–Russell, Wricke's ecovalence, Shukla's variance, AMMI, and GGE)

These tools allow breeders to process large datasets efficiently, compute stability statistics, and generate diagnostic plots (such as AMMI and GGE biplots) for interpreting genotype behavior across diverse conditions.

Integration of Stability and G×E Analysis in Breeding: In practical breeding programs, G×E interaction and stability must be considered together. The ideal genotype often combines high mean yield with low G×E interaction, indicating both performance and predictability across environments. However, breeding objectives may sometimes prioritize specific adaptation, particularly in marginal or stress-prone environments, where a genotype may consistently outperform others despite high G×E interaction.

Tools such as Kang's rank-sum, AMMI stability value (ASV), and multi-environment trial (MET) models help quantify and compare these trade-offs, enabling breeders to make context-specific decisions.

Practical Implications in Varietal Selection and Deployment: Stability and G×E analysis are crucial not only for selecting superior genotypes but also for defining mega-environments, guiding regional testing, and informing extension recommendations. These analyses help match varieties to their target population of environments (TPE), enhancing productivity and resilience under variable climatic and agronomic conditions.

By integrating traditional and modern tools, breeders can now more effectively account for G×E and stability in selection pipelines—supporting the development and deployment of varieties tailored to diverse and changing environments.

Conclusion: Genotype-by-environment interaction and stability analysis are foundational to modern plant breeding. They enable breeders to identify genotypes that are not only high yielding but also reliable across diverse and changing environments. By integrating robust experimental designs, advanced statistical methods, and powerful computational tools, researchers can effectively dissect G×E, quantify stability, and make informed recommendations for cultivar release and deployment. Mastery of these concepts and methods is essential for geneticists, agronomists, and data analysts working to deliver resilient, high-performing crops in the face of environmental variability.

R Code for Genotype by environment interaction and Stability analysis with metan

```
#install.packages("metan")
library(metan)
#install.packages("ggplot2")
library(ggplot2)
#install.packages("writexl")
library(writexl)
#install.packages("GGEbiplots")
library(GGEbiplots)
options(max.print = 10000)
#####data import#####
stabdata<-read.table("clipboard", h=T, stringsAsFactors = TRUE)
attach(stabdata)
str(stabdata)
options(max.print = 10000)

library(readxl)
stabdata <- read_excel("C:/Users/Lenovo/Desktop/UHS Bagalkot/Stability/stabdata.xlsx")

View(stabdata)
str(stabdata)
library(GGEbiplots)
##### factors with unique levels #####
stabdata$ENV <- factor(stabdata$ENV , levels=unique(stabdata$ENV ))
stabdata$GEN <- factor(stabdata$GEN, levels=unique(stabdata$GEN))
stabdata$REP <- factor(stabdata$REP, levels=unique(stabdata$REP))
str(stabdata)
##### Data inspection and cleaning functions#####
inspect(stabdata, plot=TRUE)
find_outliers(stabdata, var=GY, plots=TRUE)
find_outliers(stabdata, var=HM, plots=TRUE)
remove_rows_na(stabdata)
replace_zero(stabdata)
find_text_in_num(stabdata$PHT)
find_text_in_num(stabdata$YLD)

##### data analysis #####
##### descriptive stats #####
desc_stat(stabdata)
desc_stat(stabdata, stats="all")
ds <- desc_stat(stabdata, stats="all")
ds
write_xlsx(ds, "ds.xlsx")
histplot<-desc_stat(stabdata, hist = TRUE)
##### mean performances #####
##### mean of genotypes #####
```

```

mg <- means_by(stabdata, GEN)
mg
View(mg)
##### mean of environmnets #####
me <- means_by(stabdata, ENV)
me
View(me)
##### mean performance of genotypes across environments #####
mge <- stabdata %>%
  group_by(ENV, GEN) %>%
  desc_stat(GY, HM, stats="mean")
mge
View(mge)
#####two-way table#####
twgy<-make_mat(stabdata, GEN,ENV,GY)
twgy
twhm<-make_mat(stabdata, GEN,ENV,HM)
twhm
make_long(twgy)
make_long(twhm)
#####Exporting mean performances#####
sheets <- list("Genmean" = mg, "Envmean" = me, "Genmeaninenv"= mge,
"twowaygy"=twgy,"twowayhm"=twhm)
write_xlsx(sheets,"Mean performances.xlsx")
##### plotting performance across environments #####
## GY
GY1 <- ge_plot(stabdata, ENV, GEN, GY)
GY1
GY2 <- ge_plot(stabdata, ENV, GEN, GY, type=2)
GY2
arrange_ggplot(GY1, GY2)
## HM
HM1 <- ge_plot(stabdata, ENV, GEN, HM)
HM1
HM2 <- ge_plot(stabdata, ENV, GEN, HM, type=2)
HM2
arrange_ggplot(HM1, HM2)
#####Genotype-environment winners#####
win <- ge_winners(stabdata, ENV, GEN, resp = everything())
win
ranks <- ge_winners(stabdata, ENV, GEN, resp = everything(), type = "ranks")
ranks

sheets <- list("winner" = win, "winranks" = ranks)
write_xlsx(sheets,"GenEnvwinners.xlsx")
##### fixed effect models #####
#####Individual and Joint anova #####
indaov<-anova_ind(stabdata,ENV,GEN,REP,GY)

```

```

indaov
indaovout<-indaov$GY$individual
indaovout

jointaov<-anova_joint(stabdata,ENV,GEN,REP,GY)
jointaov
jointaovout<-jointaov$GY$anova
jointaovout
sheets <- list("indanova" = indaovout, "jointanova" = jointaovout)
write_xlsx(sheets,"indandjointanova.xlsx")
## Bartlett test
bartlett.test(stabdata$GY~stabdata$ENV, data = stabdata)
##### stability analysis:ANOVA Based Models
#####
ann <- Annicchiarico(stabdata, ENV, GEN, REP, GY)
print(ann)
eco <- ecovalence(stabdata, ENV, GEN, REP, GY)
print(eco)
Shu <- Shukla(stabdata, ENV, GEN, REP, GY)
print(Shu)

sink("anovabasedstability")
Annicchiarico(stabdata, ENV, GEN, REP, GY)
print(ann)
ecovalence(stabdata, ENV, GEN, REP, GY)
print(eco)
Shukla(stabdata, ENV, GEN, REP, GY)
print(Shu)
sink()
##### Regression based stability analysis: Eberhart and Russell's
regression mode #####
reg <- ge_reg(stabdata,ENV,GEN,REP,GY)
reg
regaov<-reg$GY$anova
regaov
regpara<-reg$GY$regression
regpara
plot(reg)

sheets <- list("regressionanova"=regaov, "regressions"=regpara)
write_xlsx(sheets,"regressionStabilitytp.xlsx")

#####non-parametric stability models#####
super <- superiority(stabdata, ENV,GEN, GY )
print(super)
fox <- Fox (stabdata, ENV,GEN, GY )
print(fox)

```

```
#####AMMI based stability analysis #####
ammiout<-performs_ ammi (stabdata,ENV,GEN,REP,GY)
ammiout
ammianova<-ammiout$GY$ANOVA
ammianova
ammipca<-ammiout$GY$PCA
ammipca
ammimeans<-ammiout$GY$MeansGxE
ammimeans
ammimodel<-ammiout$GY$model
ammimodel
#####AMMI indexes#####
ammiindex<-ammi_indexes(ammiout)
ammiindex
ammiindout<-ammiindex$GY
ammiindout
sheets <- list("ammianova" = ammiout$ANOVA, "ammipca" = ammiout$PCA, "ammimodel"=ammiout$model,
"ammimeans"= ammiout$MeansGxE, "ammiindex"=ammiindout)
write_xlsx(sheets,"ammistabilitytp.xlsx")
#AMMI biplots#
ammiplot1<-plot_scores(ammiout, x.lab = " Grain Yield")
ammiplot1
ammiplot2<-plot_scores(ammiout, type = 2, polygon = TRUE)
ammiplot2
ammiplot2<-plot_scores(ammiout, type = 2,col.env = "blue",
col.gen = transparent_color(),col.segm.env = "orange",
highlight = c("G1", "G2"),col.highlight = "darkcyan",axis.expand = 1.5)
ammiplot2
ammiplot3 <- plot_scores(ammiout, type = 4)
ammiplot3
arrange_ggplot(ammiplot1, ammiplot2, ammiplot3, tag_levels = "a", nrow = 1)

##### ammi based on weighted average of absolute scores
#####
waasammi <- waas(stabdata,ENV,GEN,REP,GY)
waasammi
waasanova<-waasammi$GY$ANOVA
waasanova
waasmodel<-waasammi$GY$model
waasmodel
wabs<- ammi_indexes(waasammi)
wabsout<-wabs$GY
wabsout
sheets <- list("waasanova" = waasanova, "wabsout"=wabsout)
write_xlsx(sheets,"waasammi.xlsx")
#####weighted average of absolute scores plots
#####
wassplot1 <- plot_scores(waasammi, type = 3)
```

```
wassplot1
```

```
wassplot2 <- plot_scores(waasammi, type = 2, polygon = TRUE)
wassplot2
```

```
#####GGE Model based stability analysis#####
ggemodel<-gge(stabdata, ENV, GEN, GY)
predict(ggemodel)
plot(ggemodel) #####A basic biplot
#####Biplot2:Mean performance vs stability#####
ggemodel<-gge(stabdata, ENV, GEN, GY, svp = "genotype")
plot(ggemodel, type = 2)
#####Biplot3: Which won where#####
ggemodel<-gge(stabdata, ENV, GEN, GY, svp = "symmetrical")
plot(ggemodel, type = 3)
#####Biplot4:Discriminateness and representativeness#####
ggemodel<-gge(stabdata, ENV, GEN, GY, svp = "3")
plot(ggemodel, type = 4)
#####Biplot5: Examine an environment#####
ggemodel<-gge(stabdata, ENV, GEN, GY, svp = "3")
plot(ggemodel, type = 5, sel_env = "E10")
#####Biplot6: Ranking environments#####
ggemodel<-gge(stabdata, ENV, GEN, GY, svp = "2")
plot(ggemodel, type = 6)
#####Biplot7: Examine a genotype#####
ggemodel<-gge(stabdata, ENV, GEN, GY, svp = "1")
plot(ggemodel, type = 7, sel_gen = "G8")
#####Biplot8: Ranking genotype#####
ggemodel<-gge(stabdata, ENV, GEN, GY, svp = "1")
plot(ggemodel, type = 8)
#####Biplot8: Ranking genotype#####
ggemodel<-gge(stabdata, ENV, GEN, GY, svp = "1")
plot(ggemodel, type = 8)
#####Biplot9: compare two genotypes genotype#####
ggemodel<-gge(stabdata, ENV, GEN, GY, svp="3")
plot(ggemodel, type = 9, sel_gen1 = "G8",sel_gen2 = "G10")
#####Biplot10: Relationship among environment#####
ggemodel<-gge(stabdata, ENV, GEN, GY, svp="2")
plot(ggemodel, type = 10)
```

```
stabdatagge <- make_mat(data_ge, GEN, ENV, GY) %>% round(2)
stabdatagge
GGEBiplot(Data = stabdatagge)
```

```
#####Multi-trait stability index#####
model<-waasb(stabdata, ENV, GEN, REP,resp = c(GY, HM), random = "all",mresp = c("h,
l"),wresp = c(60, 40))
get_model_data(model, what = "WAASBY")
```

```

index <- mtsi(model, index = "waasby", mineval = 0.7, verbose = FALSE)
print(index)
plot(index)
#####Wrapper function for stability analysis#####
gestats<-ge_stats(stabdata,ENV,GEN,REP,GY)
gestats
gestatsout<-gestats$GY
gestatsout
write_xlsx(gestatsout,"gestats.xlsx")
#####If ranks only to be extracted#####
ranksp <- get_model_data(gestats, "ranks")
ranksp
#####Spearman's rank correlation matrix between the computed stability indexes##
corplot<-corr_stab_ind(gestats, plot = FALSE, stats = "all" )
corplot
corplot2<-corr_stab_ind(gestats, plot = FALSE, stats = "ammi" )
corplot2
corplot3<-corr_stab_ind(gestats, plot = FALSE, stats = "par" )
corplot3
corplot4<-corr_stab_ind(gestats, plot = FALSE, stats = "nonpar" )
corplot4
#####Correlation coefficients with p values#####
coef_all <- corr_coef(data_ge2)
print(coef_all)
corplota<-plot(coef_all)
corplota
granum<-corr_plot(data_ge2)
granum

#####Estimation of path coefficients#####
pathcoef<-path_coef(data_ge2, resp = KW)
pathcoef<-path_coef(data_ge2, resp = KW, brutstep = TRUE)
pathl<-path_coef(data_ge2, resp = KW, pred = c(PERK, EP, NKR, PH,
NR,TKW,EL,CD,ED))
pathl

```

References:

- Lin, C. S., Binns, M. R., & Lefkovitch, L. P. (1986). Stability analysis: where do we stand?
1. *Crop science*, 26(5), 894-900.
- Sabaghnia, N., Karimizadeh, R., & Mohammadi, M. (2012). Genotype by environment interaction and stability analysis for grain yield of lentil genotypes.

Tai, G. C. (1971). Genotypic stability analysis and its application to potato regional trials. *Crop science*, 11(2), 184-190.

Caliskan, M. E., Erturk, E., Sogut, T., Boydak, E., & Arioglu, H. (2007). Genotype× environment interaction and stability analysis of sweetpotato (*Ipomoea batatas*) genotypes. *New Zealand Journal of Crop and Horticultural Science*, 35(1), 87-99.

Hossain, M. A., Rahman, L., & Shamsuddin, A. K. M. (2003). Genotype-environment interaction and stability analysis in soybean. *Journal of Biological Sciences*, 3(11), 1026-1031.

Basic Bioinformatics and QTL Analysis

Ashis Ranjan Udgata and Aravind K Konda

ICAR-Indian Institute of Pulses Research, Kanpur, UP, 208024

Email: ashisu93@gmail.com

Introduction:

Bioinformatics is a scientific discipline that combines computer science, mathematics, statistics, chemistry, and engineering to analyze, explore, integrate, and utilize data from biological sciences in research and development. It focuses on the storage, retrieval, analysis, and interpretation of biological data through the use of computer-based software and tools.

History of Bio-informatics:

- Bioinformatics began to take shape in the mid-1990s. From 1965 to 1978, Margaret O Dayhoff developed the first database of protein sequences, which was published yearly in a series titled “Atlas of Protein Sequence and Structure.”
- By 1977, DNA sequences had started to gradually appear in the literature, making it increasingly common to predict protein sequences by translating sequenced genes rather than through direct protein sequencing.
- In 1980, there was a sufficient number of DNA sequences to warrant the creation of the first nucleotide sequence database, GenBank, at the National Center for Biotechnology Information (NCBI) in the USA. NCBI became the main provider of databank information.
- The European Molecular Biology Laboratory (EMBL) was established at the European Bioinformatics Institute (EBI) in 1980, aimed at collecting, organizing, and distributing nucleotide sequence data along with related information.
- In 1984, the National Biomedical Research Foundation launched the Protein Information Resource (PIR).
- The DNA Data Bank was initiated by GemonNet in Japan in 1986. All these databanks work collaboratively and frequently share data.
- The management and analysis of the rapidly growing sequence data necessitated the development of new software and statistical tools.

Components of bioinformatics: There are three components of bioinformatics such as Data, Database and Database mining tools.

Data:

- ✓ Nucleic Acid Sequences
 - Raw DNA Sequences
 - Genomic sequence tags (GSTs)
 - cDNA sequences
 - Expressed sequence tags (ESTs)
 - Organellar DNA sequences
 - RNA Sequences
- ✓ Protein sequences
- ✓ Protein structures
- ✓ Metabolic pathways
- ✓ Gel pictures

Databases: A database is a large compilation of data related to a particular subject, such as nucleotide sequences or protein sequences, in a digital format. They serve as the digital repository for this information.

Nucleotide Sequence Databases: These are the major nucleotide sequence databases mostly used by several researchers, and other scientific community to study and apply different bioinformatics tools. The most commonly used databases are given below:

NCBI GenBank: www.ncbi.nlm.nih.gov/GenBank

EMBL: www.ebi.ac.uk/embl

DDBJ: www.ddbj.nig.ac.jp

The three databases are updated and shared daily, ensuring consistent accession numbers. There are no legal restrictions on the use of these databases; however, some sequences in the database are patented.

DDBJ (DNA Database of GenomNet, Japan) was established in 1986 through a collaboration with EMBL and GenBank. It is produced, maintained, and distributed by the National Institute

of Genetics in Japan, and sequence submissions can be made using a web-based data submission tool.

Protein data bases:

Protein databases are essential resources in bioinformatics and structural biology, providing comprehensive collections of protein sequence and structure information. These databases serve as valuable tools for researchers, enabling them to analyze, compare, and study proteins across various organisms. These databases play crucial roles in various research areas, including: Structural biology and protein folding studies, Drug discovery and design, Evolutionary analysis, Functional genomics. In proteomics research researchers can access these databases through web interfaces or programmatically via APIs, enabling integration with various bioinformatics tools and workflows. Regular updates and curation efforts ensure that these databases remain current and reliable sources of protein information for the scientific community. Some of the important protein database include SWISSPROT, Protein Information Resource (PIR) and TrEMBL etc.

Data mining tools: Database mining tools in bioinformatics are essential for extracting valuable information from large biological datasets. These tools help researchers to analyze and interpret complex data, identify patterns, and generate hypotheses. Some of the tool are as follows:

Analysis Tool	Function
BLAST (NCBI,USA)	Used for analyzing sequence data and identifying homologous sequences
ENTREZ (NCBI, USA)	Serves as a gateway to literature (abstracts), sequence, and structure databases
DNAPLOT (EBI, UK)	A tool for aligning sequences
LOCUS LINK (NCBI, USA)	Provides information on homologous genes
LIGAND (GenomNet, Japan)	A chemical database that enables searches for enzyme combinations and connects to all publicly available databases
BRITE (GenomNet, Japan)	A database for bio-molecular relations conveying information on transmission and expression, linking to all publicly accessible databases
TAXONOMY BROWSER (NCBI, USA)	Offers taxonomic classifications for various species along with genetic details
STRUCTURE	Supports the Molecular Modelling Database (MMDB) and tools for structural analysis

BLAST (Basic Local Alignment Search Tool)

BLAST is a fundamental algorithm in bioinformatics used for comparing biological sequence (DNA, RNA, Protein) information. It rapidly identifies regions of similarity between nucleotide or protein sequences by comparing a query sequence against a database. BLAST employs heuristic methods to find short matches and extend them, making it efficient for large-scale analyses. The algorithm provides statistical significance measures, such as E-values, to assess match quality. BLAST has various specialized programs for different sequence types and comparisons. It is widely used in genomics, molecular biology, and evolutionary studies for tasks like gene identification, function prediction, and exploring evolutionary relationships between organisms. The BLAST is a 3 step process.

- **Word search method:** Sequence is filled in order to remove complexity regions. Each of them prepares a set of query words (w) from the query sequence length I. Fixed length for proteins and nas are selected as 1 and 3.
- **Identification exact word method:** This alignment then searches the database for the neighbourhood word. Words having the score value equal or greater than neighbourhood score threshold (T) are taken for alignment. This conserved alignments are called as hits.
- **Maximum pair segment alignment:** In this process it extends the possible match as an ungapped alignment in both the direction that stops at maximum value. The matching criteria significance is matched by E value criteria. If E value $< 10^{-13}$ then the alignment is significant.

QTL Analysis: Application of QTL IciMapping

QTL mapping (Quantitative Trait Locus mapping) is a technique used in genetics and breeding to identify genomic regions associated with quantitative traits. Here are some key concepts related to QTL mapping:

1. **Quantitative Traits:** Quantitative traits are the traits that exhibit continuous variation and are influenced by multiple genetic and environmental factors. Examples include yield, height, weight, and protein content. Unlike qualitative traits (*e.g.*, flower color), which show discrete variation, quantitative traits are controlled by multiple genes and are influenced by environmental interactions (Lander and Botstein, 1989; Li *et al.*, 2007).

2. **QTL:** A Quantitative Trait Locus (QTL) refers to a genomic region that contains one or more genes affecting the expression of a quantitative trait. Each QTL can have an impact on the variation of the trait, contributing to its phenotypic variation. QTLs are typically identified by statistically linking genetic markers (*e.g.*, molecular markers) with the phenotypic variation observed in a population (Li *et al.*, 2007; Wang *et al.*, 2014).

3. **Mapping Populations:** QTL mapping requires the use of mapping populations, which are created by crossing individuals with contrasting phenotypic traits. The most commonly used mapping populations are bi-parental populations, such as F2 or recombinant inbred lines (RILs), generated from two parental lines with differing traits. These populations provide the genetic variation necessary for QTL identification (Lander and Botstein, 1989).

4. **Molecular Markers:** Molecular markers are DNA markers used to track genetic variation across individuals in a mapping population. They are typically short DNA sequences that can be easily assayed and genotyped. Common types of molecular markers include Single Nucleotide Polymorphisms (SNPs), Simple Sequence Repeats (SSRs), and Amplified Fragment Length Polymorphisms (AFLPs). These markers are genotyped across the mapping population and used to associate specific marker alleles with the phenotypic variation observed in the population (Meng *et al.*, 2015; Lander and Botstein, 1989; Li *et al.*, 2007).

5. **Linkage Analysis:** Linkage analysis is a statistical method used to detect the association between genetic markers and quantitative traits. It evaluates the co-segregation of genetic markers and phenotypic variation in the mapping population to infer the presence of QTLs. Various methods, such as interval mapping, composite interval mapping, and multiple QTL mapping, are employed to identify QTLs and estimate their effects.

6. **LOD Score:** The LOD (Logarithm of Odds) score is a statistical measure used in QTL mapping to assess the evidence of linkage between a marker and a quantitative trait. It quantifies the likelihood of observing the observed marker-phenotype association under the null hypothesis of no linkage. Higher LOD scores indicate a stronger association and provide evidence for the presence of a QTL in that genomic region (Lander and Botstein, 1989; Li *et al.*, 2007). **QTL Validation and Fine Mapping:** QTL mapping results need to be validated to ensure their reliability. Validation involves testing the presence and effects of identified QTLs in independent populations or environments. Fine mapping techniques, such as the use of additional markers or advanced genotyping technologies, can help narrow down the genomic regions containing the QTLs and identify the specific genes underlying the trait (population

(Meng *et al.*, 2015; Lander and Botstein, 1989; Li *et al.*, 2007). QTL mapping using the software package ICI Mapping (Integrated Composite Interval Mapping) is a widely used approach for identifying quantitative trait loci (QTLs) associated with complex traits in various crops, including chickpea. ICI Mapping is a comprehensive tool that integrates multiple QTL mapping methods and provides robust and accurate QTL detection. Here's a general overview of how QTL mapping using ICI Mapping is conducted:

Phenotypic Data: First, phenotypic data related to the trait of interest, such as seed protein concentration, is collected from a population of individuals. The phenotypic data should be accurate and consistent across the population.

Genotypic Data: Genotyping data, typically in the form of molecular markers, is obtained for the same set of individuals in the population. These markers can be simple sequence repeats (SSRs), single nucleotide polymorphisms (SNPs), or other marker types. The genotypic data should cover a sufficient number of markers across the genome to capture genetic variation.

QTL Model Construction: In ICI Mapping, various QTL mapping methods can be selected to construct QTL models. These methods include simple interval mapping (SIM), composite interval mapping (CIM), multiple-QTL mapping (MQM), and inclusive composite interval mapping (ICIM). The choice of method depends on the specific objectives and characteristics of the trait being studied (Wang *et al.*, 2007).

Statistical Analysis: Once the QTL models are constructed, statistical analysis is performed to detect and characterize the QTLs associated with the trait. ICI Mapping uses likelihood ratio tests (LRT) or interval mapping algorithms to calculate LOD (logarithm of odds) scores for each genomic region, indicating the strength of association between markers and the trait. Threshold values for LOD scores are determined based on permutation tests or other statistical methods (Meng *et al.*, 2015; Lander and Botstein, 1989).

QTL Mapping Results: The output of ICI Mapping includes QTL positions, LOD scores, additive and dominance effects, and other statistical information for each detected QTL. These results provide insights into the genomic regions that contribute to the variation in seed protein concentration.

QTL Validation and Fine Mapping: QTLs identified through ICI Mapping need to be validated using additional populations or experiments. Validation can involve different genetic mapping populations or environments to confirm the presence and stability of QTL effects. Fine mapping techniques, such as using additional markers or genotyping technologies, can be

applied to narrow down the genomic regions and identify candidate genes within the QTL regions. By utilizing ICI Mapping for QTL mapping, researchers can gain a deeper understanding of the genetic architecture underlying seed protein concentration in chickpea. The results can guide breeding programs by enabling marker-assisted selection and providing insights into the underlying genes and molecular mechanisms controlling this important trait.

QTL IciMapping (Integrated Software for Linkage Analysis and Genetic Mapping in Biparental Populations) Software:

Any genetic studies must utilize one or more genetic populations as their subjects. Regarding plant populations employed for genetic linkage analysis and QTL mapping, such as F₂, backcross (BC), doubled haploids (DH), and recombinant inbred lines (RIL), they can be divided into two main categories: temporary populations and permanent populations. In a temporary population like F₂ or BC, individuals may segregate following self-pollination. Conversely, in a permanent population such as DH or RIL, all individuals are genetically homozygous, ensuring that the genetic structure remains stable through self-pollination. Therefore, the phenotypic values of complex quantitative traits can be consistently measured via a replicated experimental design, and the same genotype can be evaluated across different environments (i.e., various locations over multiple years), facilitating more precise phenotyping and study of genotype (or QTL) interactions with the environment. As a result, permanent populations allow for better control of random environmental errors, enhancing the accuracy of QTL mapping (Li *et al.*, 2008). QTL ICI Mapping can manage twenty populations derived from a biparental cross, which includes both permanent and temporary types. Recently, permanent populations made up of a series of chromosome segment substitution (CSS) lines, also known as introgression lines, have been utilized for fine mapping of genes. CSS lines are typically developed through repeated backcrossing, aided by markers to select donor segments and control background genes. In the ideal scenario where each CSS line contains a single segment from the donor parent, standard analysis of variance (ANOVA), along with multiple mean comparisons between each line and the background parent, can be easily applied to determine if a segment in any CSS line harbors QTL responsible for the trait of interest. Unfortunately, creating a population with idealized CSS lines demands considerable labor and time. Typically, an initial CSS population features each line with several segments from the donor parent. Due to the extensive selection pressure involved in producing CSS lines, the frequencies of genes and markers within these lines do not follow the same trajectory as those

in standard mapping populations like F2, BC, DH, or RIL. QTL ICI Mapping employs a likelihood ratio test based on stepwise regression for these non-idealized CSS lines, which is also suitable for the idealized ones. A nested association mapping (NAM) population arises from a multiple-cross mating design that shares a common parent, offering high power and resolution through combined linkage and association analysis, as well as a wider genetic resource for quantitative trait evaluation. NAM populations can similarly be applied in QTL ICIMapping using a joint linkage mapping method. The QTL IciMapping software allows for conducting QTL mapping studies on the 20 biparental populations, CSS lines, and NAM populations. Additionally, the construction of linkage maps is restricted to the 20 biparental populations. Assuming the genotypes of the two parental lines are AA and BB, this would result in the frequencies of the three genotypes: AA, AB, and BB, within the 20 biparental populations.

QTL R code:

```
#install.packages(c("qtl", "bioseq")) #install if already not installed
library(qtl)
library(bioseq)

#create our first DNA sequence vector using the function dna()
x <- dna(Seq_1 = "ACCTAG", Seq_2 = "GGTATATACC", Seq_3 = "AGTC")
is_dna(x)
x
#we can select elements:
x[c("Seq_3", "Seq_1")]
x[2]
#the key difference between a DNA vector and a character vector is that DNA uses a
restricted alphabet.
#For DNA this alphabet is A, C, G, T, W, S, M, K, R, Y, B, D, H, V, N and -, which
correspond to the IUPAC symbols for DNA nucleotides.
#What happens if you include a forbidden character in a
sequence?#https://www.bioinformatics.org/sms/iupac.html
y <- dna("?AcGF")
y
#input a DNA Sequence
x_dna <- dna("ATGTCACCACAAACAGAGACT")
x_dna
#Transcribe the DNA
x_rna <- seq_transcribe(x_dna)
x_rna
# translate the sequence
x_aa <- seq_translate(x_rna)
x_aa
```



```

#reverse transcription
dna_from_rna <- seq_rev_transcribe(x_rna)
dna_from_rna
#compute the complement and the reverse complement of DNA and RNA sequences
x_dna_comp <- seq_complement(x_dna)
x_dna_comp_rev <- seq_reverse(x_dna_comp)
dna(x_dna, x_dna_comp, x_dna_comp_rev)
#STRING OPERATIONS
#pattern detection and selection
x <- dna("CTGAAACTG", "ATGAAACTG", "CTGCTG")
x[seq_detect_pattern(x, "AAAA")]
#or
x[seq_detect_pattern(x, "A{4}")]
#Alternatively, a biological sequence (i.e a DNA, RNA or AA vector) can be used as pattern.
x[seq_detect_pattern(x, dna("AAAA"))]
#This wont work. Guess why?
x[seq_detect_pattern(x, aa("AAAA"))]
# This works because W can be A or T.
x[seq_detect_pattern(x, dna("WAWA"))]
#it is important to find a pattern which contains ambiguous characters
seq_disambiguate_IUPAC(dna("WAWA"))
#If the AAAA pattern is an incorrect insertion, we may want to remove it from the sequences.
seq_remove_pattern(x, "A{4}")
#We can also replace a specific pattern with another sequence.
seq_replace_pattern(x, pattern = dna("AAAA"), replacement = dna("----"))
#if we want to replace the last 3 nucleotides with CCC
x <- seq_remove_pattern(x, "A{4}")
seq_replace_position(x, 4, 6, replacement = dna("CCC"))
#first data set analysis#####
data(hyper)
summary(hyper)
plotMissing(hyper)
# Genotype frequencies
geno.table(hyper)
data <- calc.genoprob(hyper, step = 1)
# Simple interval mapping (single QTL)
result <- scanone(hyper, method = "em") # EM algorithm for interval mapping
# Plot the LOD scores
plot(result)
perm <- scanone(hyper, method = "em", n.perm = 1000) #run upto 1000 permutation
threshold <- summary(perm, alpha = 0.05)
abline(h = threshold, col = "red", lty = 2)
summary(result, perms = perm, alpha = 0.05)
#2nd analysis####
data(fake.f2)
# take out several QTLs and make QTL object
qc <- c(1, 8, 13)
qp <- c(26, 56, 28)

```

```

fake.f2 <- subset(fake.f2, chr=qc)
fake.f2 <- calc.genoprob(fake.f2, step=2, err=0.001)
qtl <- makeqtl(fake.f2, qc, qp, what="prob")
# fit model with 3 interacting QTLs interacting
# (performing a drop-one-term analysis)
lod <- fitqtl(fake.f2, pheno.col=1, qtl, formula=y~Q1*Q2*Q3, method="hk")
summary(lod)
# fit an additive QTL model
lod.add <- fitqtl(fake.f2, pheno.col=1, qtl, formula=y~Q1+Q2+Q3, method="hk")
summary(lod.add)
# fit the model including sex as an interacting covariate
Sex <- data.frame(Sex=pull.pheno(fake.f2, "sex"))
lod.sex <- fitqtl(fake.f2, pheno.col=1, qtl, formula=y~Q1*Q2*Q3*Sex, cov=Sex,
method="hk")
summary(lod.sex)
# fit the same with an additive model
lod.sex.add <- fitqtl(fake.f2, pheno.col=1, qtl, formula=y~Q1+Q2+Q3+Sex, cov=Sex,
method="hk")
summary(lod.sex.add)
# residuals
residuals <- attr(lod.sex.add, "residuals")
plot(residuals)

```

Further reading:

- E.S. Lander, D. Botstein.(1989). Mapping Mendelian factors underlying quantitative traits using FLP linkage maps, *Genetics*,185–199.
- J. Wang, H. Li, X. Wan, W. Pfeiffer, J. Crouch, J. Wan. (2007). Application of identified QTL-marker associations in rice quality improvement through a design breeding approach, *Theory and Applied Genetics*, 115 (2007) 87–100.
- Malosetti, M., Ribaut, J.-M., & van Eeuwijk, F. A. (2013). The statistical analysis of multi-environment data: Modeling genotype-by-environment interaction and its genetic basis. *Frontiers in Physiology*, 4, 44. <https://doi.org/10.3389/fphys.2013.00044>

Transcriptomic Analysis

Prakash Kumar

ICAR-Indian Agricultural Statistical Research Institute, New Delhi-110012

Email: prakash289111@gmail.com

Introduction to Transcriptomics

Transcriptomics is the study of the complete set of RNA transcripts produced by the genome under specific conditions or in a particular cell type. It helps us understand **gene expression** patterns and regulation, giving insights into biological functions and disease mechanisms.

Applications of Transcriptomics

- Identifying disease biomarkers
- Studying gene regulation
- Uncovering alternative splicing events
- Understanding developmental processes
- Investigating responses to stress or drugs
- Precision medicine (cancer, immune diseases)

Tools & Resources

- **QC & Preprocessing:** FastQC, Trimmomatic
- **Alignment:** STAR, HISAT2, Bowtie2
- **Quantification:** Salmon, Kallisto, featureCounts
- **DE Analysis:** DESeq2, edgeR, limma
- **Annotation:** DAVID, Enrichr, GOstats
- **Visualization:** R (ggplot2, pheatmap), UCSC Genome Browser

1. Data Preprocessing

Transcriptomic data often come in raw formats from sequencing machines and require preprocessing before analysis. Preprocessing steps include quality control, adapter trimming, and read alignment. Quality control ensures the reliability of data, adapter trimming removes irrelevant sequences, and read alignment maps the reads to a reference genome.

Data imputation R code

```
# Load necessary libraries
library(impute)
library(dplyr)
# Simulated gene expression data with missing values
set.seed(123)
genes <- paste0("Gene", 1:10)
samples <- paste0("Sample", 1:5)
expression_data <- matrix(sample(c(1:10, NA), 50, replace=TRUE), nrow=10, ncol=5)
colnames(expression_data) <- samples
rownames(expression_data) <- genes
# Check the original expression data with missing values
head(expression_data)
# Data Preprocessing - Missing Value Imputation using KNN
# Impute missing values using the k-nearest neighbors (KNN) method
expression_data_imputed <- knnImputation(expression_data)
# Check the imputed expression data
head(expression_data_imputed)
#Creating a random string
> library(Biostrings)
> DNA_ALPHABET
[1] "A" "C" "G" "T" "M" "R" "W" "S" "Y" "K" "V" "H" "D" "B" "N" "-" "+" "."
> seq <- sample(DNA_ALPHABET[1:4], size = 24, replace = TRUE)
> seq
[1] "A" "G" "A" "T" "G" "C" "C" "T" "T" "C" "T" "C" "A" "C" "C" "G" "A" "A" "T"
[20] "A" "C" "A" "A" "T"
> seq <- DNAString(paste(seq, collapse = ""))
> seq
24-letter DNAString object
seq: AGATGCCTTCTCACCGAATACAAT
#Biostring basic functions
> alphabetFrequency(seq, baseOnly = T, as.prob = T)
  A      C      G      T  other
0.3333333 0.2916667 0.1250000 0.2500000 0.0000000
> reverseComplement(seq)
24-letter DNAString object
seq: ATTGATTCGGTGAGAAGGCATCT
> translate(seq)
```

```

8-letter AAString object
seq: RCLLTEYN
> seq[3:8]
6-letter DNAString object
seq: ATGCCT
#Biostring basic functions download FASTA file from open databases
# Download the sample FASTA file
>download.file("https://www.ncbi.nlm.nih.gov/sviewer/viewer.cgi?db=nucore&id=NM_001
301717.2&report=fasta", "sample.fasta")
downloaded 2319 bytes
# Read the downloaded FASTA file using Biostrings
sequences <- readDNAStringSet("sample.fasta")
# Print the sequences
>sequences
DNAStringSet object of length 1:
      width seq                      names
[1]  2191 CTCTAGATGAGTCAGTGGAGGGC...AAAAGTCTTTGGTAAATGGCAAA
NM_001301717.2 Ho...
> motif_pattern <- DNAString("CTAG")
# Find motifs
>motif_hits <- vmatchPattern(motif_pattern, sequences)
>for (i in 1:length(motif_hits)) {
  cat("Motif found in sequence", i, "at positions:",
      start(motif_hits[[i]]), "\n")
}

Motif found in sequence 1 at positions: 3 634 1237

```

2. Variant Calling

Variant calling is a critical step in genomic data analysis, where genetic variations (e.g., single nucleotide polymorphisms - SNPs, insertions, and deletions) are identified and compared to a reference genome. Variant calling algorithms use statistical models to distinguish true variants from sequencing errors.

Assume you have a CSV file named "sample_variants.csv" with the following content:

Chromosome	Position	Reference	Variant
chr1	1000	A	T
chr1	2000	C	G
chr2	1500	G	A

Now, let's use R to perform variant calling on this example dataset:

```
# Load necessary libraries
install.packages("VariantAnnotation")
library(VariantAnnotation)
# Read the CSV file
data <- read.csv("sample_variants.csv")
# Create a GRanges object for the variants
variants <- GRanges(
  seqnames = data$Chromosome,
  ranges = IRanges(data$Position, width = 1),
  ref = data$Reference,
  alt = data$Variant)
# Perform variant annotation
annotated_variants <- annotateVariants(variants)
# Display the annotated variants
print(annotated_variants)
#Output
GRanges object with 3 ranges and 2 metadata columns:
      seqnames  ranges strand |   ref     alt
      <Rle> <IRanges> <Rle> |<factor>  <factor>
[1]   chr1     1000    * |    A       T
[2]   chr1     2000    * |    C       G
[3]   chr2     1500    * |    G       A
```

seqinfo: 2 sequences from an unspecified genome; no seqlengths

In this example, we read the CSV file containing variant information, create a GRanges object representing the variants, and then use the `annotateVariants` function from the `VariantAnnotation` package to annotate the variants. The resulting annotated variants are displayed, including information about the chromosome, position, reference allele, and variant allele.

Please note that this is a simplified example using a small dataset. In real-world applications, variant calling involves more complex data preprocessing, quality filtering, and may require alignment to a reference genome before variant identification.

3. Genome Assembly

In some cases, genomic data may come from de novo sequencing projects without a reference genome. Genome assembly aims to reconstruct the full genome from these short reads. This process involves overlapping and assembling reads into contiguous sequences called contigs. Detailed description is given in next section.

4. Gene Expression Analysis

Gene expression analysis measures the activity of genes under specific conditions. It involves quantifying mRNA levels using techniques like RNA-seq. Differential gene expression analysis compares gene expression between different conditions to identify genes with significant expression changes.

Gene expression analysis typically involves preprocessing, normalization, differential expression analysis, and visualization. Here's a step-by-step example using a small dataset in R:

```
# Install and load necessary libraries
install.packages("limma")
library(limma)
Let's assume you have a CSV file named "gene_expression_data.csv" with columns 'Gene',
'Sample1', and 'Sample2'. Each row represents a gene's expression levels in two different
samples.
# Load the data
data <- read.csv("gene_expression_data.csv", header = TRUE)
# Extract gene names and expression values
gene_names <- data$Gene
expression_matrix <- data[, c("Sample1", "Sample2")]
# Optional: Convert expression values to matrix format
expression_matrix <- as.matrix(expression_matrix)
You can use methods like quantile normalization or variance stabilizing normalization (VSN)
for normalization. Here, we'll use quantile normalization from the preprocessCore package:
# Install and load necessary libraries
install.packages("preprocessCore")
library(preprocessCore)
# Perform quantile normalization
normalized_expression <- normalize.quantiles(expression_matrix)
We'll use the limma package for differential expression analysis. Let's assume you have a
design matrix where each sample is labeled with a condition (e.g., 'Control' and 'Treatment').
# Create a design matrix
design <- model.matrix(~0 + factor(c("Control", "Treatment")))
# Perform differential expression analysis using linear modeling
fit <- lmFit(normalized_expression, design)
contrast_matrix <- makeContrasts(Treatment - Control, levels = design)
fit_contrast <- contrasts.fit(fit, contrast_matrix)
fit_ebayes <- eBayes(fit_contrast)
# Extract differentially expressed genes
de_genes <- topTable(fit_ebayes, coef = 1, adjust.method = "BH", sort.by = "p", number = Inf)
Let's create a simple volcano plot to visualize the differential expression results:
# Install and load necessary libraries
install.packages("ggplot2")
```

```
library(ggplot2)
# Create a volcano plot
volcano_plot <- ggplot(de_genes, aes(x = logFC, y = -log10(P.Value))) +
  geom_point(aes(color = ifelse(P.Value < 0.05, "red", "black")), alpha = 0.7) +
  labs(x = "Log Fold Change", y = "-log10(P.Value)", title = "Volcano Plot") +
  theme_minimal()
# Display the plot
print(volcano_plot)
```

5. Functional Annotation

Functional annotation involves assigning biological functions to genes or genetic regions. It is essential for understanding the biological roles of genomic elements and can be achieved using various databases and tools. One common method for functional annotation is Gene Ontology (GO) enrichment analysis. Here's a small example of how to perform GO enrichment analysis using R:

```
# Install and load necessary libraries
install.packages("clusterProfiler")
library(clusterProfiler)
Assume you have a vector of differentially expressed gene names from the previous example:
# Differentially expressed gene names
de_gene_names <- de_genes$Gene
# Perform GO enrichment analysis using clusterProfiler
go_enrichment <- enrichGO(degene = de_gene_names,
  universe = gene_names, # All genes in your dataset
  OrgDb = org.Hs.eg.db, # Organism database (e.g., human)
  keyType = "SYMBOL", # Gene name type
  ont = "BP", # Biological Process ontology
  pvalueCutoff = 0.05, # P-value cutoff
  qvalueCutoff = 0.05) # Adjusted P-value (FDR) cutoff
# Plot the top enriched GO terms
barplot(go_enrichment, showCategory = 10)
```

6. Pathway and Network Analysis

Pathway and network analysis help uncover biological pathways and gene interactions. Pathway analysis identifies enriched pathways associated with differentially expressed genes, while network analysis models the relationships between genes and their interactions. Network analysis focuses on understanding interactions between genes or proteins within a network context. Here's a small example of how to perform pathway and network analysis using R:


```

# Install and load necessary libraries
install.packages("clusterProfiler")
install.packages("STRINGdb")
library(clusterProfiler)
library(STRINGdb)
Assume you have a vector of differentially expressed gene names:
# Differentially expressed gene names
de_gene_names <- de_genes$Gene
Perform Pathway Analysis using clusterProfiler:
# Perform KEGG pathway enrichment analysis using clusterProfiler
kegg_enrichment <- enrichKEGG(gene = de_gene_names,
                             organism = "hsa", # Human KEGG pathways
                             pvalueCutoff = 0.05,
                             qvalueCutoff = 0.05)

# Print the top enriched pathways
print(kegg_enrichment)
Network Analysis using STRINGdb:
# Create a STRING database object
string_db <- STRINGdb$new(version="11", species=9606) # 9606 for human
# Get interactions for the differentially expressed genes
interaction_data <- string_db$get_interactions(de_gene_names)
# Create a network plot using igraph and visNetwork
library(igraph)
library(visNetwork)
# Convert interactions to an igraph object
gene_network <- graph_from_data_frame(interaction_data, directed = FALSE)
# Customize network plot attributes
network_vis <- visIgraph(gene_network) %>%
  visNodes(color = "lightblue", shape = "circle") %>%
  visEdges(color = "gray") %>%
  visLayout(randomSeed = 123)
# Display the network plot
network_vis

```

7. Epigenomic Analysis

Epigenomics explores modifications to the genome that affect gene expression without altering the DNA sequence. Epigenomic analysis includes DNA methylation and histone modification studies, which play critical roles in development and disease. Here's a small example of how to perform a basic epigenomic analysis using DNA methylation data in R:

```
# Install and load necessary libraries
install.packages("minfi")
library(minfi)
Assume you have a DNA methylation dataset in the form of a MethySet object. This could be
a publicly available dataset or your own data in the appropriate format.
# Load and preprocess DNA methylation data
data("MsetExample")
methylation_data <- MsetExample
Quality control is crucial in epigenomic analysis to ensure data reliability.
# Quality control
qc <- getQC(methylation_data)
plotQC(qc)
You can identify differentially methylated regions (DMRs) between different groups.
# Define groups (for example, controls and cases)
groups <- factor(c("Control", "Case", "Control", "Case"))
# Perform DMR analysis
dmr_results <- DMRcate(methylation_data, group = groups)
Visualize the DMR results.
# Plot DMR results
plotDMR(dmr_results)
```

8. Association Studies

Genomic data analysis is extensively used in genome-wide association studies (GWAS) to identify genetic variants associated with specific traits or diseases. GWAS involves comparing genotypic data from cases and controls to discover genetic associations. It is a powerful approach that involves rapidly scanning markers across entire DNA genomes of numerous subjects to identify genetic variations linked to specific traits. By discovering new genetic associations, researchers can enhance strategies for detecting and managing these traits. GWAS is particularly valuable for uncovering genetic variations contributing to common, complex traits. In essence, GWAS relies on establishing correlations between genetic markers, often Single Nucleotide Polymorphisms (SNPs), and measurable traits within a population. The primary aim of GWAS is to pinpoint potential causal variants within genes or their regulatory elements that influence the target phenotype. This process contributes to a deeper comprehension of the genetic underpinnings of the trait. The typical stages of a GWAS encompass:

Genotype Calling and Quality Control: This involves determining genotypes from raw chip data and applying fundamental quality control measures.

Principal Component Analysis (PCA): PCA aids in detecting and, if necessary, correcting population stratification—a potential source of bias.

Genotype Imputation: Imputation employs linkage disequilibrium data from references like HapMap to predict missing genotypes.

Association Testing: Associations between individual SNPs and continuous or categorical phenotypes are statistically tested.

Global Significance Analysis and Correction: Multiple testing correction methods are applied to ensure robust significance thresholds.

Data Presentation: Visual aids like quantile-quantile and Manhattan plots facilitate effective presentation of results.

Cross-Replication and Meta-Analysis: Integration of association data from multiple studies, including cross-replication and meta-analysis, strengthens the findings.

Despite large-scale (meta-)studies involving thousands or even tens of thousands of samples, only a handful of candidate loci with highly significant associations are typically identified. Although these associations are replicated in independent studies, each locus explains a minute fraction (<1%) of the genetic variance underlying the phenotype. Various factors contribute to this outcome, necessitating ongoing research and advanced methodologies to unravel the complexity of genetic influences on traits.

The methodology of a Genome Wide Association Study (GWAS) revolves around the systematic examination of numerous variable points distributed across a genome. Given that these genetic variations are inherited in groups or blocks, it's unnecessary to test every single point individually. This approach entails swiftly scanning markers throughout the entire DNA or genomes of multiple subjects to uncover genetic variations linked to specific traits. Identifying novel genetic associations empowers researchers to develop enhanced strategies for managing these traits. The foundation of genome-wide association studies was made feasible by the advent of chip-based microarray technology capable of assaying over a million Single Nucleotide Polymorphisms (SNPs) or more. The primary platforms employed in most GWAS are Illumina and Affymetrix. The Affymetrix platform embeds short DNA sequences as spots on a chip, which discern specific SNP alleles through differential hybridization of

sample DNA. Conversely, Illumina employs a bead-based approach with slightly lengthier DNA sequences to identify alleles. While Illumina chips are pricier to manufacture, they offer superior specificity. Notably, technology for measuring genomic variation is rapidly evolving, with chip-based genotyping platforms progressively giving way to cost-effective next-generation sequencing methods for whole-genome sequencing. GWAS target two primary classes of phenotypes: categorical (binary, case/control) and quantitative traits. Statistically, quantitative traits are favored due to their enhanced power to detect genetic effects and often yield more interpretable outcomes. The design of a genetic association study varies based on several factors:

Scale of Study: It can be genome-wide or genomics-based.

Marker Design: Marker selection depends on the choice of markers like microsatellites, SNPs, or CNVs.

Subject Design: The study can adopt a candidate gene approach or a genome-wide screening approach.

In essence, the GWAS methodology involves strategically analyzing genetic variations to reveal significant associations between markers and traits, with the aim of advancing our understanding and management of complex genetic traits.

Genome-wide studies can be broadly categorized into three main types: cohort studies, family-based studies, and case-control studies. Each type has its own strengths and limitations, shaping their applications in genetic research.

Cohort studies involve subjects assumed to represent the broader population. Phenotypes are used to establish similarities among individuals, regardless of genetic variations. This method directly assesses risk and is less biased compared to case-control studies. However, cohort studies demand long-term follow-up and a substantial sample size, making them expensive and less suitable for studying rare traits.

Family-based studies assume that families are representative of the target population and both parents share the same genetic background. These studies are advantageous for assessing Mendelian inheritance and are less susceptible to spurious associations. Parent phenotypes aren't always necessary, allowing for investigations into imprinting. Simple logistics aid

association detection. Yet, family-based studies are cost-inefficient, possess low power, and are sensitive to genotyping errors.

Case-control studies involve subjects drawn from the same population, with cases representing all instances of the trait in question. These studies are straightforward, cost-effective, and accommodate a large number of cases and controls. They are optimal for investigating rare traits. However, case-control studies are susceptible to population stratification, and factors like batch effects and biases can introduce distortions. Additionally, case-control studies often lead to overestimations for common traits.

GAPIT R packages for GWAS

```
#install.packages("devtools")
#devtools::install_github("jiabowang/GAPIT3",force=TRUE)
library(GAPIT3)
library(GAPIT)
myY <- read.table("mdp_traits.txt", head = TRUE)
myG <- read.delim("mdp_genotype_test.hmp.txt", head = FALSE)
myGAPIT <- GAPIT(Y=myY, G=myG, PCA.total=3)
myGAPIT <- GAPIT(Y=myY[,1:2], G=myG, PCA.total=3, model="MLM")
#Tutorial 3: User defined Kinship and PCs
myKI <- read.table("KSN.txt", head = FALSE)
myCV <- read.table("Copy of Q_First_Three_Principal_Components.txt", head = TRUE)
myGAPIT <- GAPIT(Y=myY[,1:2], G=myG, KI=myKI, CV=myCV,)
#Tutorial 4: Genome Prediction
myGAPIT <- GAPIT(Y=myY[,1:2], G=myG, KI=myKI, PCA.total=3, model=c("gBLUP"))
#Tutorial 6: Numeric Genotype Format
myGD <- read.table("mdp_numeric.txt", head = TRUE)
myGM <- read.table("mdp_SNP_information.txt" , head = TRUE)
myGAPIT <- GAPIT(Y=myY[,1:2], GD=myGD, GM=myGM, PCA.total=3)
```

Genome-wide association studies (GWAS) are predominantly applied in disease-focused research, where these different study designs offer distinct advantages based on the specific research objectives and resources available.

9. Structural Variants

Genomic data analysis also focuses on identifying structural variants, such as large deletions, duplications, and inversions, which can cause genetic diseases or contribute to evolutionary processes.

10. Genomic Analysis using R-Programming Language

R is a popular open-source programming language used for statistical computing and graphics. It provides a wide range of packages for analyzing and visualizing large-scale biological datasets. R is a popular programming language for data analysis and visualization in computational biology and bioinformatics. Role of R-software used in big data bioinformatics and computational biology described in **Figure 1**.

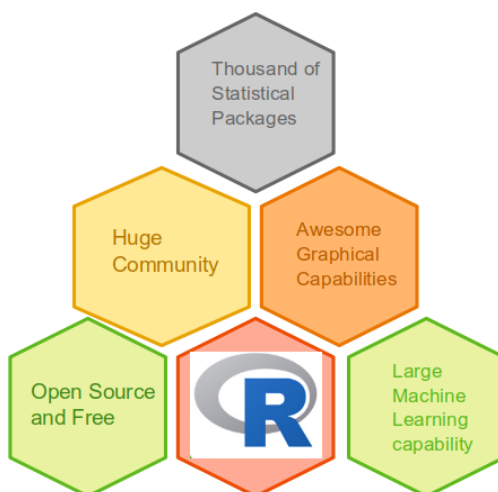


Figure 1. Overview of role of R-software used in big data bioinformatics and computational biology

Here are some examples of R code that can be used for Big Data analysis in this field:

10. 1. Data preprocessing:

a) Reading and cleaning data:

R can be used to read and import large datasets in various formats such as CSV, Excel, and TSV. The following code imports a CSV file called "input_file_name.csv" into R:

Table 1: Illustrative Data Table for Demonstrating Preprocessing with R

ID	Name	Age	Score
1	John	25	85
2	Jane	32	90
3	Michael	NA	75
4	Susan	28	88
5	Emily	22	92
2	Jane	32	90

```
# Read data from a CSV file
data <- read.csv("input_file_name.csv", header = TRUE)
# Clean data by removing NAs and duplicates
cleaned_data <- na.omit(data)
cleaned_data <- unique(cleaned_data)
```

b) Filtering data:

```
# Filter data to keep only rows where gene expression is above a certain threshold value
filtered_data <- data[data$expression > 10,]
```

10.2. Data analysis:

R has many built-in functions and libraries that can be used for data analysis. The following code calculates the mean, median, and standard deviation of a column in a data frame:

```
mean_value <- mean(data$column)
median_value <- median(data$column)
sd_value <- sd(data$column)
```

a) Gene expression analysis using DESeq2 package:

```
# Load DESeq2 package
library(DESeq2)

# Create a DESeqDataSet object
dds <- DESeqDataSetFromMatrix(countData = counts, colData = metadata, design = ~
condition)
# Run differential expression analysis
dds <- DESeq(dds)
# Get differentially expressed genes
results <- results(dds)
```

b) Clustering data:

```
# Create a distance matrix
dist_mat <- dist(data, method = "euclidean")
```

```
# Cluster the data using hierarchical clustering
```

```
hclust_res <- hclust(dist_mat)
```

```
# Visualize the clustering using a dendrogram
```

```
plot(hclust_res)
```

c) Machine learning: R provides several libraries for machine learning such as caret and mlr. The following code trains a linear regression model using the caret library:

```
library(caret)
```

```
model <- train(column_to_predict ~ ., data = data, method = "lm")
```

10.3. Data visualization:

R provides several powerful libraries such as ggplot2 for data visualization. The following code creates a scatterplot using ggplot2:

```
library(ggplot2)
```

```
ggplot(data, aes(x=column1, y=column2)) + geom_point()
```

a) Creating a heatmap:

```
# Load pheatmap package
```

```
library(pheatmap)
```

```
# Create a heatmap of gene expression data
```

```
pheatmap(data, scale = "row")
```

b) Creating a scatterplot:

```
# Create a scatterplot of gene expression data
```

```
plot(data$gene1, data$gene2, xlab = "Gene 1 expression", ylab = "Gene 2 expression")
```

10.4. Data storage:

```
# Save cleaned data to a CSV file
```

```
write.csv(cleaned_data, "cleaned_data.csv")
```

The following code loads a dataset of gene expression data and performs differential gene expression analysis using Bioconductor:

These are just a few examples of the many tasks that can be performed using R for Big Data analysis in computational biology and bioinformatics. The specific commands used will depend on the data being analyzed and the goals of the analysis.

11. NGS Data Analysis

A genome represents the entirety of an organism's cellular DNA, carefully organized to fit into chromosomes, ensuring proper packaging and enabling precise expression. This expression enables a cell to effectively transmit genetic information to subsequent generations. Genomes reside within cell nuclei, as well as within chloroplasts and mitochondria in plants. The timing, specificity, and degree of gene expression are dictated by the gene's sequence itself. Essentially, the arrangement of DNA's nucleotides along the strand is the key determinant. Therefore, comprehending the sequence of a gene, along with its neighboring regions within the genome, and even the complete genome, becomes crucial for grasping its intricate structure and complexity. This realm pertains to structural and functional genomics, where scientists unravel the life's code, aiming to fathom its intricacies and harness its potential benefits.

In the 1960s and 70s, the task of genome sequencing was seen as a formidable challenge and financially demanding endeavor. Prevailing theories suggested that the size of an organism's genome was directly linked to its overall size. However, these theories were contradicted by empirical evidence. Notably, it was discovered that the genome of a substantial mammal was smaller than that of a lily plant. Even prokaryotic organisms appeared to possess more DNA in their genomes than their single-cell confines could accommodate or efficiently utilize. A considerable portion of the genome was initially deemed redundant and coined "junk DNA," seemingly consuming the organism's resources and energy without apparent purpose. Subsequently, a transformative era of rapid progress dawned in comprehending the genetic code and its functioning in prokaryotic and eukaryotic organisms. The early strides in sequencing were marked by significant breakthroughs. In 1977, two distinct methods for DNA sequencing emerged. The first technique, known as Maxam-Gilbert sequencing, developed by scientists at Harvard University, employed specific chemicals to cleave radioactively labeled DNA at precise base positions. The second approach, pioneered by Frederick Sanger in England and termed the chain termination method (also referred to as the Sanger method), involved a DNA synthesis reaction using specialized forms of nucleotides that, upon integration into a DNA chain, halted further chain elongation.

A pivotal milestone in genome sequencing emerged during the latter part of the previous decade with the introduction of next-generation sequencers by 454 and Solexa. This heralded a transformative shift in the approach to genome sequencing. Earlier projects, spanning species

such as humans, Arabidopsis, and rice, primarily utilized the BAC by BAC (bacterial artificial chromosome) method. While effective, this approach was time-consuming, resource-intensive, and financially demanding. It relied on the availability of high-density molecular maps, a resource limited to a select few plants.

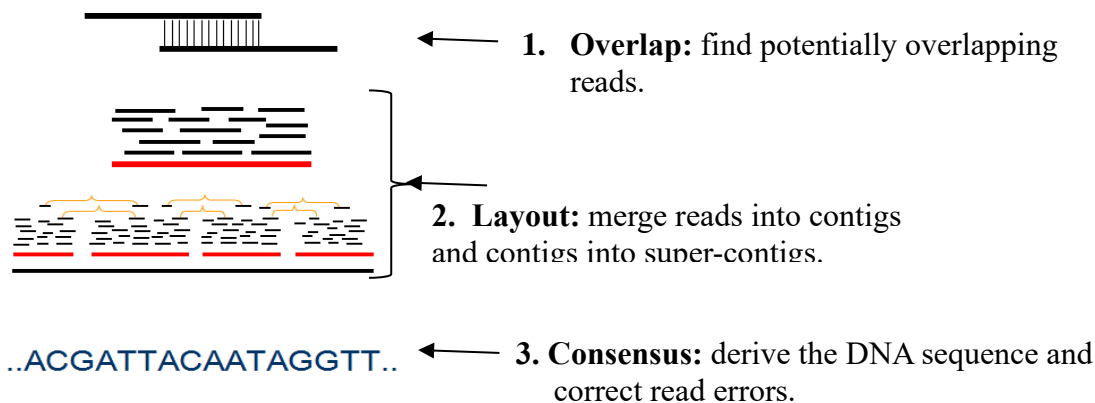
Significantly, two innovative systems, Pacific Biosciences and Oxford Nanopore, have emerged, providing the remarkable advantage of generating longer reads, with an average spanning of up to 10 kb or more. As a result, the landscape of genome sequencing has undergone a notable transformation. Today, the decoding and comprehensive analysis of genomes have become notably more accessible, eliminating the once imposing barrier of cost. Over the past decade, the cost of genome sequencing has considerably diminished, transitioning from multi-million-dollar endeavors, such as the sequencing of human and rice genomes, to a few hundred dollars for a complete genome. This excludes expenses associated with platforms and data analysis. Nonetheless, it's important to acknowledge that each of these technologies is not without its drawbacks. From a bioinformatics perspective, the task of assembling every genome presents challenges attributed to a range of factors. Particularly in the realm of plant genomics, these challenges are particularly pronounced and often appear unending. Plant genomes exhibit remarkable complexity due to several factors: 1) Encompassing larger genome sizes, 2) Displaying polyploidy (multiple sets of chromosomes), 3) Experiencing high heterozygosity (genetic diversity), 4) Possessing an epigenetic nature (gene regulation mechanisms influenced by chemical modifications), and 5) Featuring the presence of both mitochondrial and chloroplast genomes. The assembly of these intricate plant genomes poses a substantially more formidable challenge than that encountered with mammalian genomes. It underscores the intricate nature of plants' genetic makeup and underscores the ongoing pursuit to conquer the complexities inherent in understanding and decoding these genomes.

11.1. Genome Assembly:

Sequence assembly involves the intricate process of aligning and merging fragments of DNA sequences to reconstruct the original genetic sequence. This step is essential due to the limitations of DNA sequencing technology, which is unable to read entire genomes in a single sweep. Instead, it generates small fragments of genetic material, ranging from 20 to 1000 bases, depending on the specific sequencing technology employed. The recent strides in sequencing

techniques have led to the generation of extensive volumes of sequence data. However, the fragments produced by these advanced high-throughput methods are notably shorter compared to traditional Sanger sequencing.

The initial sequence assemblers emerged during the late 1980s and early 1990s as refined versions of simpler sequence alignment programs. These assemblers were developed to reconstruct the comprehensive genetic blueprint by piecing together myriad fragments produced by automated DNA sequencers. These sequencing instruments, known as DNA sequencers, laid the groundwork for subsequent advancements. Algorithms were crafted to facilitate whole-genome shotgun (WGS) fragment assembly. Prominent among these algorithms were Atlas, Arachne, Celera, PCAP, Phrap (www.phrap.org), and Phusion. These programs adopted an overlap-layout-consensus approach, wherein all the individual reads are systematically compared to one another in a pairwise manner. In essence, sequence assembly represents a critical endeavor in modern genomics, allowing scientists to reconstruct the intricate genetic puzzle from fragmented pieces and unveiling the comprehensive blueprint encoded within DNA.



The resultant draft genome sequence is crafted through the amalgamation of information gleaned from sequenced "contigs." Subsequently, this information is harnessed to construct "scaffolds," a process outlined in Figure 1. These scaffolds, in turn, are strategically positioned along the physical map of chromosomes, giving rise to a meticulously delineated "golden path."

Recent times have witnessed the emergence of novel sequencing methodologies. Among these, commercially available technologies encompass pyrosequencing (454 Sequencing), sequencing by synthesis (Illumina), and sequencing by ligation (SOLiD). Notably, the reads generated through these next-generation sequencing techniques are notably shorter compared to traditional Sanger reads. Due to their abbreviated length, these reads necessitate prolific

production and elevated coverage depths compared to earlier sequencing methodologies. In contrast to the extended overlaps facilitated by long reads, short reads within repetitive DNA regions exhibit fewer discernible differences for accurate assessment. These distinct challenges have prompted multiple research teams to engineer de novo assembly tools tailored to accommodate these exceedingly short reads. By addressing these intricacies, these specialized tools aim to pave the way for effectively assembling genomes even in the presence of the inherent limitations posed by short read lengths and repeat complexities.

Types of Sequencers and data format

Illumina	:	FASTQ
SoLID/ABI-Life:		FASTA
Roche 454	:	SFF
Ion Torrent	:	SFF or FASTQ

Types of Assembly

There are two type of assembly base on the availability of reference genome:

- a) **De novo Assembly:** Reads are aligned to each other to form a consensus sequence that are called contigs.
- b) **Reference genome assembly:** Here reads are aligned with the available reference genome to form a consensus sequences.

Genome Assembly techniques

General procedure for genome sequencing and assembly emphasizing the procedures that used at genome-sequencing centres-

- **Fragment readout:** The sequences of each fragment are determined using automatic base-calling software. Phred is the most widely used program.
- **Trimming vector sequences:** Shotgun reads often contain part of the vector sequences that have to be removed before sequence assembly.
- **Trimming low-quality sequences:** Shotgun reads contain poor quality base calls and removing or masking out these low-quality base calls often leads to more accurate sequence assembly. However, this step is optional and some sequencing centres do not

mask out low-quality base calls, relying on the fragment assembler to utilize quality values to decide true fragment overlaps.

- **Fragment assembly:** The shotgun data is input to a fragment assembler that automatically generates a set of aligned fragment called contigs.
- **Assembly validation:** Some contigs that assembled in the previous steps may be misassembled due to repeats. Since we do not have a priori knowledge on repeats in the targets DNA, it is very difficult to verify the correctness of assembly of each contig and this step is largely done manually. There are recent algorithmic developments on automatic verification of contig assemblies.
- **Scaffolding Contigs:** Contigs needs to be oriented and ordered. The mate-pair information is a primary information source for this step, thus this step is not achievable if the input shotgun is not prepared by reading both ends of clones.
- **Finishing:** Assuming that all contigs are assembled correctly and contigs are oriented and ordered correctly, we can close gaps between two contigs by sequencing specific regions that corresponds to the position of gaps.

De novo assembly of next-generation sequencing reads

Once Next-Generation Sequencing (NGS) reads have been generated, they undergo alignment to a known reference sequence or are subjected to de novo assembly. De novo assembly is a pivotal process employed to reconstruct the genome of organisms that have not been previously sequenced or lack a comparative reference genome. This intricate task involves the shotgun approach, wherein the organism's genome is fragmented into small pieces, each of which is sequenced individually and then reconstructed through computational techniques. The complexity of de novo assembly arises from the presence of segments within genomes that share identical sequences, commonly referred to as repeats. These repeats vary significantly in length, rendering the task of recovering the entire genome an arduous endeavor. Consequently, most de novo assembly tools focus on the generation of extended segments of the genome, termed contigs. While the process yields valuable insights, it falls short of providing a complete genome reconstruction.

Furthermore, the intricacy of de novo assembly escalates in proportion to the genome's size. Within the realm of de novo genome assembly, two primary categories stand out: Overlap Layout and Consensus (OLC), and De Bruijn graph-based methods. OLC methods, while

highly effective, tend to be computationally intensive. In contrast, De Bruijn graph-based methods offer efficiency, yet they come with a higher memory requirement. Notably, within the De Bruijn graph framework, several tools have been developed, each contributing to the arsenal of options available for genome assembly. These tools reflect the continuous refinement and diversification of approaches in pursuit of accurate and comprehensive genome reconstruction

Assembly for Double-Ended Short-read Sequencing Technologies

Emerging Pyrosequencing-like technologies hold immense promise, yet they come with a trade-off - the resulting read lengths are notably shorter compared to those generated by present sequencing platforms. This inherent brevity poses a challenge when encountering sequence repeats. Indeed, if a sequence repetition exceeds the length of the read, it ushers in an intractable ambiguity. A particular concern is the concise representation of the target within the shortest common superstring derived from a set of short reads. This representation often emerges as excessively compressed. To grapple with the complexities of repeat sequences, a solution emerges through the proposal of a variable-insert length, double-ended read protocol. This protocol entails fragmenting multiple target clones and employing gel electrophoresis to meticulously segregate fragments within a specific length range, denoted as $a \pm b\%$. This is equivalently expressed as fragments of lengths ranging from d to $d+w$, where d and w are designated integers. By embracing this innovative approach, the challenge of repeats can be systematically addressed, paving the way for enhanced accuracy and comprehensive insights in sequencing outcomes.

Issues and Problems of assembling complex genomes

Genome assembly is a very difficult computational problem, made more difficult because many genomes contain large numbers of identical sequences, known as repeats. These repeats can be thousands of nucleotides long, and some occur in thousands of different locations, especially in the large genomes of plants and animals.

One challenge to sequencing crop genomes is the vast difference in scale between the size of the genomes and the lengths of the reads produced by the different sequencing methods. While there may be a 10–500× difference in scale between the short reads produced by second-generation sequencing and modern Sanger sequencing, this is still dwarfed by the difference between Sanger read length and the lengths of complete chromosomes. As the sequenced

organisms became larger and more complex, the assembly programs employed in genome projects required progressively advanced techniques to manage:

- Terabytes of sequencing data that need to be processed on computing clusters;
- Identical and nearly identical sequences (referred to as repeats) that can, in the worst-case scenario, lead to exponential increases in the time and space complexity of algorithms; and
- Errors in the fragments produced by sequencing instruments, which can complicate the assembly process.

Table : Lists of prevalent de-novo assemblers

Name	Type	Technologies	Author	Late Updated
BySS	(large) genomes	Solexa, SOLiD	Simpson, J. et al.	2008 / 2011
ALLPATHS-LG	(large) genomes	Solexa, SOLiD	Gnerre, S. et al.	2011
AMOS	genomes	Sanger, 454	Salzberg, S. et al.	2002 / 2008
Arapan-M	Medium Genomes (e.g. E.coli)	All	Sahli, M. & Shibuya, T.	2011 / 2012
Arapan-S	Small Genomes (Viruses and Bacteria)	All	Sahli, M. & Shibuya, T.	2011 / 2012
Celera WGA Assembler / CABOG	(large) genomes	Sanger, 454, Solexa	Myers, G. et al.; Miller G. et al.	2004 / 2010
CLC Genomics Workbench & CLC Assembly Cell	genomes	Sanger, 454, Solexa, SOLiD	CLC bio	2008 / 2010 / 2011
Cortex	genomes	Solexa, SOLiD	Iqbal, Z. et al.	2011
DNA Baser	genomes	Sanger, 454	Heracle BioSoft SRL	2013
DNA Dragon	genomes	Illumina, SOLiD, Complete Genomics, 454, Sanger	SequentiX	2011
DNAnexus	genomes	Illumina, SOLiD, Complete Genomics	DNAnexus	2011
Edena	genomes	Illumina	D. Hernandez, P. François, L.	2008/2013

Euler	genomes	Sanger, 454 (,Solexa ?)	Farinelli, M. Osteras, and J. Schrenzel. Pevzner, P. et al.	2001 / 2006
Euler-sr	genomes	454, Solexa	Chaisson, MJ. et al.	2008
Forge	(large) genomes, EST, metagenomes	454, Solexa, SOLID, Sanger	Platt, DM, Evers, D.	2010

Conclusion

Genomic data analysis is a rapidly evolving field that plays a crucial role in understanding genetic variation, gene expression, and disease associations. By applying various computational and statistical methods, researchers can gain valuable insights into the complexities of genomes and improve our understanding of fundamental biological processes and human health. Transcriptomic analysis is a powerful tool for exploring gene expression patterns at a genome-wide level. Advances in RNA-Seq have revolutionized our ability to study dynamic changes in the transcriptome with high resolution and sensitivity.

References:

- Batzoglou, S.; Jaffe, DB; Stanley, K; Butler, J; Gnerre, S; Mauceli, E; Berger, B; Mesirov, JP et al. (January 2002). "ARACHNE: a whole-genome shotgun assembler". *Genome Research* 12 (1): 177–89. doi:10.1101/gr.208902. PMC 155255. PMID 11779843.
- Boisvert, Sébastien; Laviolette, François; Corbeil, Jacques (October 2010). "Ray: simultaneous assembly of reads from a mix of high-throughput sequencing technologies". *Journal of Computational Biology* 17 (11): 1519–33. doi:10.1089/cmb.2009.0238. PMC 3119603. PMID 20958248.
- Dohm, J. C.; Lottaz, C.; Borodina, T.; Himmelbauer, H. (November 2007). "SHARCGS, a fast and highly accurate short-read assembly algorithm for de novo genomic sequencing". *Genome Research* 17 (11): 1697–706. doi:10.1101/gr.6435207. PMC 2045152. PMID 17908823.
- Huse, S. M., Huber, J. A., Morrison, H. G., Sogin, M. L., and Welch, D. M. (2007) Accuracy and quality of massively parallel DNA pyrosequencing, *Genome Biol* 8, R143.
- Mardis, E. R. (2008) The impact of next generation sequencing technology on genetics, *Trends Genet* 24, 133–141.

- Michael C Schatz*, Jan Witkowski and W Richard McCombie (2012) Current challenges in de novo plant genome sequencing and assembly. *Genome Biology*, **13**:243
- Myers, E. W.; Sutton, GG; Delcher, AL; Dew, IM; Fasulo, DP; Flanigan, MJ; Kravitz, SA; Mobarry, CM et al. (March 2000). "A whole-genome assembly of *Drosophila*". *Science* 287 (5461): 2196–204. doi:10.1126/science.287.5461.2196. PMID 10731133.
- Pop, M. (2004) Shotgun sequence assembly, *Adv Comput* **60**, 193–248. 7. Pop, M., and Salzberg, S. L. (2008) Bioinformatics challenges of new sequencing technology, *Trends Genet* **24**, 142–149.
- Ronaghi, M., Uhlen, M., and Nyren, P. (1998) A sequencing method based on real-time pyrophosphate, *Science* **281**, 363–365.
- Zhang W, Chen J, Yang Y, Tang Y, Shang J, et al. (2011) A Practical Comparison of De Novo Genome Assembly Software Tools for Next-Generation Sequencing Technologies. *PLoS ONE* **6**(3): e17915. doi:10.1371/journal.pone.0017915.

GWAS: Genome Wide Association Studies

Anupam Singh

Shree Guru Gobind Singh Tricentenary University, Gurugram, Haryana 122505

Email: anupambiotech@gmail.com

GWAS stands for Genome Wide Association Studies. An examination of many common genetic variants in different individuals to see if any variant is associated with a trait. GWAS typically focus on associations between single-nucleotide polymorphisms (SNPs) and traits like major diseases.

The central goal of GWAS is to identify casual mutations that have an effect on a phenotype (any aspect of an organism that can be measured). A casual mutation is a position in the genome where an experimental manipulation of the DNA produces an effect on the phenotype on average. From a statistical point of view, a casual mutation occurs when $\text{Cov}(Y, X) \neq 0$ where Y are the value of the phenotypes and X the value of the genotypes. Genome-wide association analyses are aimed for detecting variants at genomic loci that are associated with complex traits in the population and, in particular, at detecting associations between common single-nucleotide polymorphisms (SNPs) and common diseases. Markers that are significantly associated with the phenotype are presumed to be in linkage disequilibrium (LD) with putative Quantitative Trait Loci (QTL). The goal of GWAS, is to test for association between the frequency of each of hundreds of thousands of common variants and a given phenotype, that exceed a conservative genome-wide threshold for association and then test these for evidence of replication. High statistical power, low probability of Type I error, use of covariates, and high resolution are the keys for success in GWAS.

A. Basic Guideline to Perform a GWAS Study: -

1. Read phenotypes and check the assumptions of the models.
 - (a) Check outliers.
 - (b) Normal distribution of errors. Possible transformation of the data.
2. Read genotypes and filter for:
 - (a) Markers with a proportion of missing data more than a particular threshold set by the researcher.
 - (b) Individuals with a high proportion of missing data.

- (c) Individuals with a high proportion of heterozygous.
- (d) Remove genotypes with a minor allele frequency (MAF) less than 5 %.
- (e) Remove genotypes that fail a Hardy-Weinberg test of equilibrium (Normally, use a conservative p-value cut-off of $<10^{-5}$).

3. Imputation of the genotype file, and performing the Kinship matrix.
4. Look for population structure effects.
5. Match phenotypes and genotypes.
6. Perform GWAS function from GAPIT or rrBLUP with and without population structure effects and Kinship matrix.
7. Manhattan and Q-Q plot graphs.
8. Interpretation and Validation.

What is GAPIT?

Genome Association and Prediction Integrated Tool

- Statistical package that is run in the R software environment
- Developed by Alex Lipka and Zhiwu Zhang
- Alexander E. Lipka et al. (2012) GAPIT: Genome Association and Prediction Integrated Tool. Bioinformatics. doi: 10.1093/bioinformatics/bts444
- Uses statistical tools implemented in other programs like TASSEL

GAPIT is a package that is run in the R software environment. R can be freely downloaded from <http://www.r-project.org>. We also recommend the integrated development environment RStudio which is also freely available at <http://www.rstudio.com>.

Installing GAPIT3: -

GAPIT3 can currently be installed in several ways.

- From source on the internet
- From GitHub
- From an archive

Installation from source at ZZlab:-

GAPIT can be loaded with a single function.

```
R> source("http://zzlab.net/GAPIT/GAPIT.library.R")
```

After loading the library, we'll need to source the GAPIT function as well.

```
R> source("http://zzlab.net/GAPIT/gapit_functions.txt")
```

Installation from GitHub:-

Installation can also be made from GitHub when the R package devtools is available.

```
R> install.packages("devtools")
```

```
R> devtools::install_github("jiabowang/GAPIT3",force=TRUE)
```

```
R> library(GAPIT3)
```

Installation from an archive:-

GAPIT can be installed from an archive such as *.tar.gz or *.zip archive. An archive can be downloaded from the "releases" page. If you would like the latest version of GAPIT from the GitHub site you may want to clone it and then build it (this may require Rtools on Windows).

```
bash$ git clone git@github.com:jiabowang/GAPIT3.git
```

```
bash$ R CMD build GAPIT3
```

Once an archive has been obtained it can be installed from a shell, similar to as follows.

```
bash$ R CMD INSTALL GAPIT3_3.1.0.9000.tar.gz
```

Or similarly from within R.

```
R> install.packages("GAPIT3_3.1.0.9000.tar.gz", repos = NULL, type="source")
```

Data Preparation: -

Phenotype Data

The user has the option of performing GWAS on multiple phenotypes in GAPIT. This is achieved by including all phenotypes in the text file of phenotypic data. Taxa names should be in the first column of the phenotypic data file and the remaining columns should contain the observed phenotype from each individual. Missing data should be indicated by either “NaN” or “NA”.

Taxa	EarHT	dpoll	EarDia
811	59.5	NaN	NaN
4226	65.5	59.5	32.21933
4722	81.13	71.5	32.421
33-16	64.75	64.5	NaN
38-11	92.25	68.5	37.897
A188	27.5	62	31.419
A214N	65	69	32.006
A239	47.88	61	36.064
A272	35.63	70	NaN
A441-5	53.5	67.5	35.008
A554	38.5	66	33.41775
A556	28	65	31.929
A6	109.5	80.5	31.5175
A619	36	61	40.63
A632	60	61	35.953
A634	54	59	35.601
A635	37	64	35.3005
A641	54.5	66	33.727
A654	39	64	NaN

Genotype Data: -

Hapmap Format

Hapmap is a commonly used format for storing sequence data where SNP information is stored in the rows and taxa information is stored in the columns. This format allows the SNP information (chromosome and position) and genotype of each taxa to be stored in one file.

Numeric Format

GAPIT also accepts the numeric format. Homozygotes are denoted by “0” and “2” and heterozygotes are denoted by “1” in the “GD” file. Any numeric value between “0” and “2” can represent imputed SNP genotypes. The first row is a header file with SNP names, and the first column is the taxa name. The “GM” file contains the name and location of each SNP. The first column is the SNP id, the second column is the chromosome, and the third column is the base pair position. As seen in the example, the first row is a header file.

Analysis GWAS: -

A Basic Scenario

The user needs to provide two data sets (phenotype and genotype) and one input parameter. This parameter, “PCA.total”, specifies the number of principal components (PCs) to include in the GWAS model. GAPIT will automatically calculate the kinship matrix using the VanRaden method²³, perform GWAS and genomic prediction with the optimum compression level using the default clustering algorithm (average) and group kinship type (Mean). The scenario assumes that the genotype data are saved in a single file in HapMap format. If the working

directory contains the tutorial data, the analysis can be performed by typing these command lines:

#Step 1: Set data directory and import files

```
myY <- read.table("mdp_traits.txt", head = TRUE)
myG <- read.table("mdp_genotype_test.hmp.txt", head = FALSE)
```

#Step 2: Run GAPIT

```
myGAPIT <- GAPIT(
Y=myY,
G=myG,
PCA.total=3
)
```

- **GLM**

The GAPIT uses Least Squares to solve the model. The GAPIT code for running a GLM is:

```
myGAPIT_GLM <- GAPIT(
Y=myY[,c(1,2)],
GD=myGD,
GM=myGM,
model="GLM",
PCA.total=5,
file.output=T
)
```

- **MLM**

EMMA method is used in GAPIT, the code of MLM is:

```
myGAPIT_MLM <- GAPIT(
Y=myY[,c(1,2)],
GD=myGD,
GM=myGM,
model="MLM",
PCA.total=5,
file.output=T
)
```

- **CMLM**

Compress Mixed Linear Model is published by Zhang in 2010. The code of CMLM is:

```
myGAPIT_CMLM <- GAPIT(
Y=myY[,c(1,2)],
GD=myGD,
GM=myGM,
model="CMLM",
```

```
PCA.total=5,
file.output=T
)
```

- **MLMM**

Multiple Loci Mixed linear Model is published by Segura in 2012. The code of MLMM in GAPIT is:

```
myGAPIT_MLMM <- GAPIT(
Y=myY[,c(1,2)],
GD=myGD,
GM=myGM,
model="MLMM",
PCA.total=5,
file.output=T
)
```

- **SUPER**

Settlement of MLM Under Progressively Exclusive Relationship is published by Qishan in 2014. The code of SUPER is:

```
myGAPIT_SUPER <- GAPIT(
Y=myY[,c(1,2)],
GD=myGD,
GM=myGM,
model="SUPER",
PCA.total=5,
file.output=T
)
```

- **Farm-CPU**

Fixed and random model Circulating Probability Unification (FarmCPU) is published by Xiaolei in 2016. The code of Farm-CPU in GAPIT is:

```
myGAPIT_FarmCPU <- GAPIT(
Y=myY[,c(1,2)],
GD=myGD,
GM=myGM,
model="FarmCPU",
PCA.total=5,
file.output=T
)
```

- **Convert HapMap format to numerical**

Many software requires genotype data in the numerical format. GAPIT can perform such conversion with a few lines of code as follows.

```
myG <- read.table("mdp_genotype_test.hmp.txt", head = FALSE)
myGAPIT <- GAPIT(G=myG, output.numerical=TRUE)
```

```
myGD= myGAPIT$GD
myGM= myGAPIT$GM
```

- **User-inputted Kinship Matrix and Covariates**

This scenario assumes that the user provides a kinship matrix and covariate file. The kinship matrix or covariates (e.g., PCs) may be calculated previously or from third party software e.g., STRUCTURE 2.3.4. When the PCs are input in this way, the parameter “PCA.total” should be set to 0 (default). Otherwise, PCs will be calculated within GAPIT, resulting in a singular design matrix in all model fitted for GWAS. The analysis can be performed by typing these command lines:

```
#Step 1: Set data directory and import files
```

```
myY <- read.table("mdp_traits.txt", head = TRUE)
```

```
myG <- read.table("mdp_genotype_test.hmp.txt", head = FALSE)
```

```
myKI <- read.table("KSN.txt", head = FALSE)
```

```
myCV <- read.table("Copy of Q_First_Three_Principal_Components.txt", head = TRUE)
```

```
#Step 2:
```

```
Run GAPIT myGAPIT <- GAPIT(
```

```
Y=myY,
```

```
G=myG,
```

```
KI=myKI,
```

```
CV=myCV
```

```
)
```

References: -

Lipka, A. E. et al. GAPIT: genome association and prediction integrated tool. *Bioinformatics* 28, 2397–2399 (2012).

Tang, Y. et al. GAPIT Version 2: An Enhanced Integrated Tool for Genomic Association and Prediction. *Plant J.* 9, (2016).

Wang J., Zhang Z., GAPIT Version 3: Boosting Power and Accuracy for Genomic Association and Prediction, *Genomics, Proteomics & Bioinformatics* (2021), doi: <https://doi.org/10.1016/j.gpb.2021.08.005>.

GWAS data analysis in rrBLUP: -

Genomic selection has been emerged in plant and animal breeding selection paradigm, with the advent of inexpensive and high-throughput genotyping technologies in the last decade. Genomic selection allows the prediction of the phenotypes of individuals based on known marker effects or genetic relationships (kinship-based), and in plants it has been used for predicting trait performance of hybrids and unrealized crosses. Genomic predictions can be made by estimating marker effects (rrBLUP). rrBLUP has been developed primarily for genomic prediction with mixed models (but it can also do genome-wide association mapping with GWAS). Ridge Regression (rr) is one of the first statistical method proposed for genomic selection was a called, where the ridge parameter (λ) can be observed in a mixed model framework as the σ^2_e / σ^2_u ratio between the residual and random effect variances. This can be applied in the genomic context where σ^2_u is the genetic variance and best linear unbiased predictor (BLUP) can be interpreted as the genomic estimated breeding values (GEBV), where the random effect refers to genotype effects and the variance-covariance structure is the additive or genomic relationship matrix (A or Ag). The genetic variance can also be interpreted in terms of marker effects in the form of marker-based BLUPs.

Phenotypes are considered following a normal probability model, there are a broad class of models that can apply to continuous and discrete phenotypes analysis. The R software is free and open-source statistical software that can be downloaded for Windows, Mac OS X, or Linux from <https://www.r-project.org/>. From the left side of the website click on CRAN (Comprehensive R Archive Network), select the appropriate CRAN Mirror, and select the appropriate operating system and install it into your computer following on-screen prompts for installation. Once in your computer, rrBLUP package needs to be installed by using the next command.

```
>install.packages("rrBLUP")
```

- **GWAS Study in rrBLUP**

Read phenotypes and check the assumptions of the models.

```
pheno <- read.csv("phenoat.csv",header=T);
```

```
dim(pheno)
```

```
head(pheno) ## GID ENV Yield
```

```

str(pheno)
hist(pheno$Yield,xlab="Yield",main="Histogram Yield")
shapiro.test(pheno$Yield)
boxplot.yield<- boxplot(pheno$Yield)
outliers <- boxplot.yield$out; outlier
pheno <- pheno[-which(pheno$Yield%in%outliers),]
shapiro.test(pheno$Yield)
pheno <- na.omit(pheno)

```

The code above indicates if there are outliers on the phenotypic data. After removing the outliers, we cannot reject the Shapiro–Wilk normality test indicating that yield data are now normal. Last line in code is to eliminate any possible missing data (NA).

Read genotypes and filtering.

```

geno <- read.csv("genoat.csv",header=T,row.names = 1);
dim(geno)
map <- read.csv("mapoat.csv",header=T,stringsAsFactors
=F,row.names=1); dim(map)
geno[1:5,1:5] ### View genotypic data.
map[1:5,1:3]

```

The next step is to filter the genotypic data. Filtering conditions will depend on researcher criteria. The code below represents a simple function to remove individuals and markers that does not met the criteria establish by the researcher. The function will remove individuals with more than a certain percentage of missing data, markers with a greater proportion of a threshold missing percentage, and also markers with a high proportion of heterozygous calls.

```

filter.fun <- function(geno,IM,MM,H){
#Remove individuals with more than a certain %
missing data individual.missing <- apply(geno,1,
function(x){
return(length(which(is.na(x)))/ncol(geno))
})
#Remove markers with certain % missing data
marker.missing <- apply(geno,2,function(x)

```

```

{return(length(which(is.na(x)))/nrow(geno))
})
length(which(marker.missing>0.6))
#Remove individuals with high heterozygous calls.
heteroz <- apply(geno,1,function(x){
return(length(which(x==0))/length(!is.na(x)))
})
filter1 <- geno[which(individual.missing<IM),
which(marker.missing<MM)]
filter2 <- filter1[, (heteroz<H)]
return(filter2)
}
geno.filtered <- filter.fun(geno[,1:3629],0.4,0.60,0.02)
geno.filtered[1:5,1:5];dim(geno.filtered)

```

The lower the minor allele frequency (MAF) the lower the statistical power, because MAF increases the variance of the phenotypes associated with the MAF alleles. Therefore, it is necessary to filter them and the most standard way is to eliminate markers with less than 5 % of MAF. Minor allele frequency will be removed using the A.mat function within the rrBLUP package.

Imputation of the genotype file, and performing the Kinship matrix.

The main idea behind imputation is to predict (or ‘impute’) the missing data based upon the observed data. Imputation is now routinely used to facility genotyped studies by increasing the power of the analysis. Here, “A.mat” function from rrBLUP package is used for imputation. A.mat has two options for imputation (<https://cran.r-project.org/web/packages/rrBLUP/rrBLUP.pdf>). One is to replace missing data with the population mean for that marker, or using an expectation maximization (EM) algorithm based on the multivariate normal distribution.

```

library(rrBLUP)
Imputation <- A.mat(geno.filtered,impute.method="EM",
return.imputed=T,min.MAF=0.05)
K.mat <- Imputation$A ; dim(K.mat) ### KINSHIP matrix
geno.gwas <- Imputation$imputed; dim(geno.gwas) #NEW geno data.

```

```
geno.gwas[1:5,1:5]## view geno
K.mat[1:5,1:5]## view Kinship
```

Look for population structure effects.

An important aspect of GWAS analysis involves examining the population structure (PS). The primary purpose of conducting this examination is that, due to different genetic histories, various subpopulations may exhibit differences in allele frequencies for numerous polymorphisms across the genome. If these populations display varying overall phenotypic values, any polymorphisms that show differing frequencies between the populations may be linked to the phenotype, even though they are not necessarily causal or in strong linkage disequilibrium with the true causal polymorphisms. To visualize the structure of our populations, we utilize principal component analysis (PCA) on genotypic data with the “*svd()*” function in R.

Population structure accounted by PCA is limited to correcting for spurious associations on a global level of genetic variation. Thereby, PS does not adequately capture the relatedness between individuals, and this relationship between genotypes (K, kinship matrix) needs also be taking into account on the analysis. Not taking into account of PS, K as well as a potential confounding between the phenotype and the genotype effects, could lead to unrealistic assessments in GWAS analysis.

```
geno.scale <- scale(geno.gwas,center=T,scale=F) # Data needs to be center.
Svdgeno <- svd(geno.scale)
PCA <- geno.scale%*%svdgeno$v #Principal components
colnames(PCA) <- paste("PCA",1:ncol(PCA),sep="")
PCA[1:5,1:5]

plot(round((svdgeno$d)^2/sum((svdgeno$d)^2)
,d=7)[1:10],type="o",main="Screeplot",xlab="PC
As",ylab="% variance")
PCA1 <- 100*round((svdgeno$d[1])^2/sum((svdgeno$d)^2),d=3); PCA1
PCA2 <- 100*round((svdgeno$d[2])^2/sum((svdgeno$d)^2),d=3); PCA2
Eucl <- dist(geno.gwas) # Euclidean distance
Fit <- hclust(Eucl,method="ward") # Ward criterion makes clusters with same size.
groups2 <- cutree(fit,k=2) # Selecting two clusters.
table(groups2) #Number of individuals per cluster.
plot(PCA[,1],PCA[,2],xlab=paste("Pcomp:",PCA1,"% ",sep=""),ylab=paste("Pcomp:",PCA2,
"% ",sep=""),pch=0,cex=0.7,col=groups2)
```

Match phenotypes and genotypes.

```
pheno=pheno[pheno$GID%in%rownames(geno.gwas),]
pheno$GID<-factor(as.character(pheno$GID),
```

```

levels=rownames(geno.gwas)) #to assure same levels on both files
##Creating file for GWAS function from rrBLUP
package X<-model.matrix(~-1+ENV, data=pheno)
pheno.gwas <- data.frame(GID=pheno$GID,X,Yield=
pheno$Yield) head(pheno.gwas)
geno.gwas <- geno.gwas[rownames(geno.gwas)%in%
pheno.gwas$GID,]
pheno.gwas <- pheno.gwas[pheno.gwas$GID%in%
rownames(geno.gwas),]
geno.gwas <- geno.gwas[rownames(geno.gwas)%in%
rownames(K.mat),]
K.mat <- K.mat[rownames(K.mat)%in%rownames(geno.
gwas),colnames(K.mat)%in%rownames(geno.gwas)]
pheno.gwas <- pheno.gwas[pheno.gwas$GID%in%
rownames(K.mat),]
geno.gwas <-geno.gwas[,match(map$Markers,colnam
es(geno.gwas))] head(map)
geno.gwas <- geno.gwas[,colnames(geno.gwas)%in%map$Markers]
map <- map[map$Markers%in%colnames(geno.gwas),]
geno.gwas2<- data.frame(mark=colnames(geno.gwas),
chr=map$chrom,loc=map$loc,t(geno.gwas))
dim(geno.gwas2)
colnames(geno.gwas2)[4:ncol(geno.gwas2)] <-rownames(geno.gwas)
head(pheno.gwas)
geno.gwas2[1:6,1:6]
K.mat[1:6,1:6]

```

Perform GWAS function from rrBLUP with and without population structure effects and Kinship matrix.

A statistically significant association between a genotypic marker and a particular trait is considered to be a proof of linkage between the phenotype and a casual locus. Generally, PS leads to spurious associations between markers and a trait, so that a statistical approach must account for PS. In this analysis, four different statistical models were performed.

- (a) Naïve model without controlling for PS or family relatedness (gwasresults).
- (b) Controlling for PS effects (Q model, gwasresults2).
- (c) Controlling just for relatedness (K model, gwasresults3).
- (d) Controlling for both Q and K effects (Q + K model, gwasresults4).

```

## gwasresults<-GWAS(pheno.gwas,geno.gwas2, fixed= colnames(pheno.gwas)[2:5], K=NULL,
plot=T,n.PC=0)
##gwasresults2<-GWAS(pheno.gwas,geno.gwas2, fixed=colnames(pheno.gwas)[2:5], K=NULL,
plot=T,n. PC=6)
##gwasresults3<-GWAS(pheno.gwas,geno.gwas2, fixed=colnames(pheno.gwas)[2:5], K=K.mat,
plot=T,n. PC=0)

```

```
##gwasresults4<-GWAS(pheno.gwas,geno.gwas2, fixed=colnames(pheno.gwas)[2:5], K=K.mat,
plot=T,n. PC = 6)
```

Manhattan and Q-Q plot graphs.

```
#Let's see the structure
```

```
str(gwasresults)
```

```
str(gwasresults)
```

```
#First 3 columns are just the information from markers and map.
```

```
#Fourth and next columns are the results from GWAS. Those values are already
```

```
#the -log10 pvalues, so no more transformation needs to be done to plot them.
```

```
pdf("Figure5.pdf",width = 7)
```

```
par(mfrow=c(2,2))
```

```
N <- length(gwasresults$Yield)
```

```
expected.logvalues <- sort( -log10( c(1:N) * (1/N) ) )
```

```
observed.logvalues <- sort( gwasresults$Yield)
```

```
plot(expected.logvalues , observed.logvalues, main="Naïve model(K=NULL,n.PC=0)",
```

```
  xlab="expected -log pvalue ",
```

```
  ylab="observed -log p-values",col.main="blue",col="coral1",pch=20)
```

```
abline(0,1,lwd=3,col="black")
```

```
N1 <- length(gwasresults2$Yield)
```

```
expected.logvalues1 <- sort( -log10( c(1:N1) * (1/N1) ) )
```

```
observed.logvalues1 <- sort( gwasresults2$Yield)
```

```
plot(expected.logvalues1 , observed.logvalues1, main="Q model (K=NULL,n.PC=6)",
```

```
  xlab="expected -log pvalue ",
```

```
  ylab="observed -log p-values",col.main="blue",col="coral1",pch=20)
```

```
abline(0,1,lwd=2,col="black")
```

```
N2 <- length(gwasresults3$Yield)
```

```
expected.logvalues2 <- sort( -log10( c(1:N2) * (1/N2) ) )
```

```
observed.logvalues2 <- sort( gwasresults3$Yield)
```

```
plot(expected.logvalues2 , observed.logvalues2, main="K model (K=Kmat,n.PC=0)",
```

```
  xlab="expected -log pvalue ",
```

```
  ylab="observed -log p-values",col.main="blue",col="coral1",pch=20)
```

```
abline(0,1,lwd=2,col="black")
```

```
N3 <- length(gwasresults4$Yield)
```

```

expected.logvalues3 <- sort( -log10( c(1:N3) * (1/N3) ) )
observed.logvalues3 <- sort( gwasresults4$Yield)
plot(expected.logvalues3 , observed.logvalues3, main="Q+K model (K.mat,n.PC=6)",
      xlab="expected -log pvalue ",
      ylab="observed -log p-values",col.main="blue",col="coral1",pch=20)
abline(0,1,lwd=2,col="black")
dev.off()

```

MANHATTAN PLOT

#False Discovery Rate Function

```

FDR<-function(pvals, FDR){
  pvalss<-sort(pvals, decreasing=F)
  m=length(pvalss)
  cutoffs<-((1:m)/m)*FDR
  logicvec<-pvalss<=cutoffs
  postrue<-which(logicvec)
  print(postrue)
  k<-max(c(postrue,0))
  cutoff<-(((0:m)/m)*FDR)[k+1]
  return(cutoff)
}

alpha_bonferroni=-log10(0.05/length(gwasresults$Yield)) ####This is Bonferroni correcton
alpha_FDR_Yield <- -log10(FDR(10^(-gwasresults$Yield),0.05))## This is FDR cut off

```

MANHATTAN PLOT

```

pdf("Figure6.pdf",width=8,height=8)
par(mfrow=c(2,2))
plot(gwasresults$Yield,col=gwasresults$chr,ylab="-log10.pvalue",
      main="Naïve model (K=NULL,n.PC=0)",xaxt="n",xlab="Position",ylim=c(0,14))
#axis(1,at=c(1:length(unique(gwasresults$chr))),labels=unique(gwasresults$chr))
axis(1,at=c(0,440,880,1320,1760))
abline(a=NULL,b=NULL,h=alpha_bonferroni,col="blue",lwd=2)
abline(a=NULL,b=NULL,h=alpha_FDR_Yield,col="red",lwd=2,lty=2)

```

```
legend(1,13.5, c("Bonferroni","FDR") ,
      lty=1, col=c('red', 'blue'), bty='n', cex=1,lwd=2)
```

```
plot(gwasresults2$Yield,col=gwasresults2$chr,ylim=c(0,14),ylab="-log10.pvalue",
     main="Q model (K=NULL,n.PC=6)",xaxt="n",xlab="Position")
axis(1,at=c(0,440,880,1320,1760))
abline(a=NULL,b=NULL,h=alpha_bonferroni,col="blue",lwd=2)
abline(a=NULL,b=NULL,h=alpha_FDR_Yield,col="red",lwd=2,lty=2)
legend(1.5,13.5, c("Bonferroni","FDR") ,
      lty=1, col=c('red', 'blue'), bty='n', cex=1,lwd=2)
```

```
plot(gwasresults3$Yield,col=gwasresults3$chr,ylim=c(0,14),ylab="-log10.pvalue",
     main="K model (K=K.mat,n.PC=0)",xaxt="n",xlab="Position")
axis(1,at=c(0,440,880,1320,1760))
abline(a=NULL,b=NULL,h=alpha_bonferroni,col="blue",lwd=2)
abline(a=NULL,b=NULL,h=alpha_FDR_Yield,col="red",lwd=2,lty=2)
legend(1.5,13.5, c("Bonferroni","FDR") ,
      lty=1, col=c('red', 'blue'), bty='n', cex=1,lwd=2)
```

```
plot(gwasresults4$Yield,col=gwasresults4$chr,ylim=c(0,14),ylab="-log10.pvalue",
     main="Q+K model (K=K.mat,n.PC=6)",xaxt="n",xlab="Position")
axis(1,at=c(0,440,880,1320,1760))
abline(a=NULL,b=NULL,h=alpha_bonferroni,col="blue",lwd=2)
abline(a=NULL,b=NULL,h=alpha_FDR_Yield,col="red",lwd=2,lty=2)##FDR gives inf for
Yield
```

```
legend(1,13.5, c("Bonferroni","FDR") ,
      lty=1, col=c('red', 'blue'), bty='n', cex=1,lwd=2)
dev.off()
```

WHICH ARE HITS?

```
which(gwasresults$Yield>alpha_bonferroni)
which(gwasresults$Yield>alpha_FDR_Yield)
which(gwasresults2$Yield>alpha_bonferroni)
which(gwasresults2$Yield>alpha_FDR_Yield)
```



```

which(gwasresults3$Yield>alpha_bonferroni)
which(gwasresults3$Yield>alpha_FDR_Yield)
which(gwasresults4$Yield>alpha_bonferroni)
which(gwasresults4$Yield>alpha_FDR_Yield)
markers.gwasresults4.bonf<- geno.gwas[,c(53,56,57,1054,1427)]#gwasresults3 and 4 have
same hits.
markers.gwasresults2.bonf <- geno.gwas

```

References

- Dehghan, A. (2018). Genome-wide association studies. *Genetic Epidemiology: Methods and Protocols*, 37-49.
- Evangelou, E., & Ioannidis, J. P. (2013). Meta-analysis methods for genome-wide association studies and beyond. *Nature Reviews Genetics*, 14(6), 379-389.
- Tam, V., Patel, N., Turcotte, M., Bossé, Y., Paré, G., & Meyre, D. (2019). Benefits and limitations of genome-wide association studies. *Nature Reviews Genetics*, 20(8), 467-484.
- Uffelmann, E., Huang, Q. Q., Munung, N. S., De Vries, J., Okada, Y., Martin, A. R., ... & Posthuma, D. (2021). Genome-wide association studies. *Nature Reviews Methods Primers*, 1(1), 59.

Genomic Selection and Its Utilization for Crop Breeding

Anantha M S¹, Santosha Rathod²

¹ICAR - Indian Institute of Rice Research, Rajendranagar, Hyderabad 500030

²ICAR - National Institute of Abiotic Stress Management, Baramati, Pune, 413 115

Email: anugenes@gmail.com

The role of Genomic-enabled Prediction in Plant Breeding began in the 1980s. With the emergence of various molecular marker systems, the availability of polymorphic markers for plant breeders and molecular biologists has significantly increased. Among these systems, single nucleotide polymorphisms (SNPs) stand out as the most prominent high-throughput genotyping (HTG) method. SNPs have been widely used in discovering quantitative trait loci (QTLs). Over 10,000 QTLs have been reported in more than 120 studies covering 12 different plant species (Bernardo, 2008), with the goal of improving quantitative traits of economic importance. Initially, molecular markers were integrated into conventional phenotypic selection (PS) by applying marker-assisted selection (MAS). For simple traits, MAS involves selecting individuals that possess QTL-associated markers with significant effects, while markers that do not show a significant association with a trait are excluded. However, attempts to enhance complex quantitative traits using QTL-associated marker detection have been unsuccessful due to the difficulties in identifying the same QTL across multiple environments (due to QTL environment interactions) or in varying genetic backgrounds (Bernardo, 2016).

Genomic selection is an upgrading form of marker-assisted selection for quantitative traits, and it differs from the traditional marker assisted selection in that markers in the entire genome are used to predict genetic values and the QTL detection step is skipped. Genomic selection holds the promise to be more efficient than the traditional marker-assisted selection for traits controlled by polygenes.

Linkage analysis used for QTL mapping is typically carried out on biparental populations, but it has a limited ability to identify marker–trait associations due to chromosomes exhibiting low recombination rates. Consequently, association mapping emerged in the early 2000s with the aim of enhancing the power of linkage analysis, enabling the identification of marker–trait associations in non-biparental populations and the fine mapping of chromosome segments characterized by high recombination rates. However, the main problem of fine-association

mapping is the limited power to detect rare variants that may be associated with economically important traits (Bernardo, 2016). Thus, the difficulty of association mapping and QTL detection lies in the identification and quantification of rare QTLs with minor effects for economically important traits that are highly influenced by the environment. However, due to the substantial reduction in the costs of SNP assays, the possibility of using high-density SNP arrays (containing tens of thousands of markers) has resulted in the development of statistical models to predict marker–trait association accurately, based on the genetic architecture of the trait being assessed (Crossa *et al.*, 2017).

Contrary to QTL and association mapping, Genomic Selection (GS) utilises all molecular markers for GP of the performance of the candidates for selection. Consequently, the purpose of GS is to estimate breeding and/or genetic values. GS integrates both molecular and phenotypic information from a training population (TRN) to derive the genomic estimated breeding values (GEBVs') for individuals in a testing population (TST) that have been genotyped but not phenotyped (Meuwissen 2001). Figure 1A illustrates the two basic populations in a GS program: the TRN data, whose phenotype and genotype are known, and the TST data, whose genetic values are to be predicted. GS replaces the need for phenotyping for a few selection cycles. The main advantages of GS compared to traditional phenotype-based selection in breeding include a reduction in the cost per cycle and the time needed for variety development. For instance, in maize breeding, a breeder can evaluate 50% of all available lines through testcrossing in first-stage multi-locational trials, allowing the phenotypic data to be used for predicting the other 50% by GS. Figure 1B illustrates the advantage of GS over PS with respect to: (i) potentially lowering costs by up to 50%; and (ii) saving time by selecting lines directly for stage II instead of going through stage I, as required in PS. This significantly lowers the expenses associated with forming testcrosses and evaluating them at each stage of multi-location evaluations. The time efficiency gained over PS may arise from the second selection cycle, which employs the TRN from the previous cycle to predict the new doubled haploid (DH) lines, thereby omitting the need for testcross formation and first-stage multilocation evaluation trials. Based on Genomic Selection process, the best lines could go directly to the second stage of multi-location evaluations.

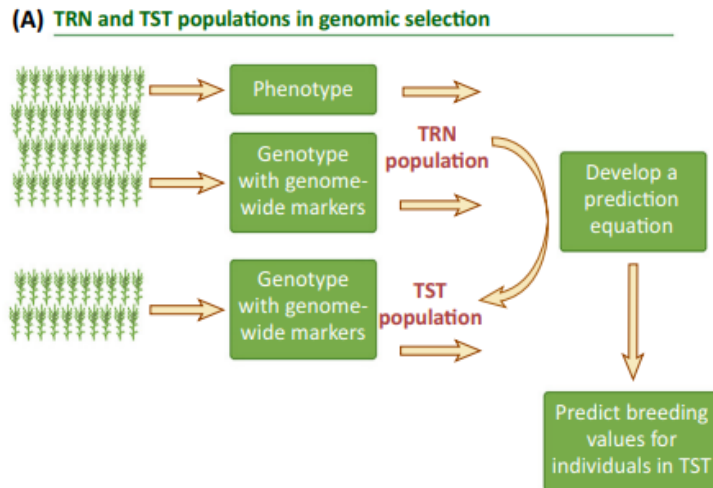


Figure 1 A

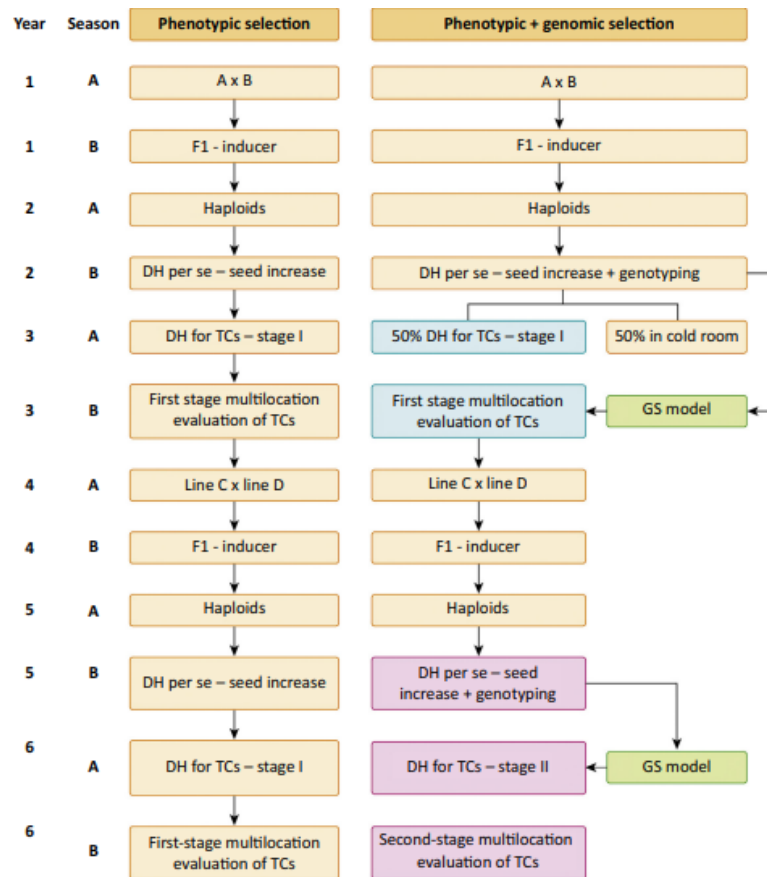


Figure 1 B

Figure 1. Populations Utilized in Genomic Selection and a Scheme of Phenotypic and Genomic Selection in Maize Breeding. (A) Genomic selection (GS) necessitates a training population (TRN) that has undergone both genotyping and phenotyping, along with a testing population (TST) that has been genotyped but not phenotyped.

(B) Reduction of cycle duration in maize through GS using doubled haploids (DH) crossed with a tester (TC). (Crossa *et al.*, 2017)

Genomic Selection estimates the breeding values (BVs) of the candidates chosen for selection. BVs consist of two components: the parental average (the mean BV of both parents) and the deviation of progeny performance from this average, which results from Mendelian sampling. In traditional breeding, the parental average is determined using pedigree data (when genealogy is accessible), allowing for the creation of a relationship matrix A among individuals. Mendelian sampling evaluates within-family variability, which is measured through progeny testing in multi-environment field trials. Genomic Selection takes advantage of dense markers to assess Mendelian sampling, thereby eliminating the need for extensive progeny phenotyping. This approach accelerates the process by shortening the breeding cycle while boosting the expected genetic gain and selection response over time; it also minimizes resource usage compared to extensive phenotyping. GS holds the potential of rapidly enhancing complex traits with low heritability and significantly lowering the cost involved in developing lines and hybrids. GS is also applicable to simple traits with higher heritability than complex traits, for which high GP accuracy is anticipated. The implementation of GS in plant breeding may face limitations due to 2 key factors: (i) the expenses associated with genotyping; and (ii) a lack of clear guidance on where Genomic Selection can be applied most effectively within breeding programs.

GS and GP have been utilised through two distinct approaches. One method concentrates on predicting additive effects in the early generations of a breeding program (F2:3) to enable a rapid selection cycle with a short interval (i.e., GS at the F2 level of a biparental cross). In this scenario, researchers aim to anticipate the breeding values (BVs) instead of the total genetic value; thus, additive linear models that compile the effects of the markers are adequate. The alternative method predicts individuals' complete genetic values by considering both additive and nonadditive (dominance and epistasis) effects, which allows for estimating the cultivars' performance (commercial value). The genetic values of lines are predicted for specific environments using an incomplete (sparse) multi-environment testing framework.

Several genetic and statistical factors hinder the effective use of GP. Genetic challenges arise from the size and diversity of the TRN population, as well as the heritability of the traits being predicted. Statistical challenges are linked to the high dimensionality of marker data, where the

number of markers (p) greatly exceeds the number of observations (n) ($p \gg n$), along with the multicollinearity present among markers (where adjacent markers show significant correlation).

Genomic-Enabled Prediction Models and Applications Coping with Complexity: The complexity of implementing GP in breeding occurs at various levels and is influenced by several factors. When a trait is influenced by a large number of loci, the accuracy of GP depends on several genetic factors: (i) the size and genetic variation within the training population (TRN) and its relationship with the test population (TST) (Pszczola *et al.*, 2012); specifically, whether the cultivars in the TRN are closely or distantly related to those in the TST group; (ii) the heritability of the traits being selected [complex traits characterized by low heritability and small marker effects are suitable for GS and GP, while traits that are less complex (with high heritability) can be effectively predicted using a limited number of markers that have relatively significant effects]; and (iii) for complex traits involving numerous markers that do not exhibit linkage disequilibrium (LD) with the quantitative trait loci (QTL), the accuracy of GP decreases (Daetwyler *et al.* 2010) but improves as heritability and TRN size increase. Studies have shown the significance of choosing an appropriate TRN population that enhances the accuracy of the predictions for non-phenotyped cultivars within the TST population (Isidro, *et al.* 2015). Depending on the trait, the enhancement in Genomic Prediction accuracy tends to reach a saturation point as the population size grows. A similar pattern has been observed regarding the number of markers (Lorenz, *et al.* 2012 and Arruda, *et al.* 2015).

A significant genetic-statistical challenge of genomic prediction (GP) models occurs when forecasting non-phenotyped individuals in particular environments (site-year combinations) by integrating genotype-by-environment (GxE) interactions into the statistical frameworks. Equally significant is the genomic complexity associated with GxE interactions for multiple traits; such interactions formulate trait and environmental structures that should be addressed using statistical-genetic models which utilise multi-trait, multi-environment variance-covariance, as well as genetic correlations among environments, traits, and between traits and environments, simultaneously. Untangling the complexities of multi-trait genomics and diverse environments necessitates a theoretical framework that considers these complex interactions (Montesinos-López, *et al.* 2016). Interestingly, employing GP to enhance disease resistance in wheat has proven to be difficult for two primary reasons: (i) selection for dominant resistance genes can be ephemeral owing to shifts in pathogen races; and (ii) breeding for minor resistance

genes with small effects throughout genomic selection (GS), which offers sustained resistance, may encounter the typical challenges associated with GS (Poland and Rutkoski, 2016).

Another level of complexity arises in GS statistical prediction models due to the scenario where the number of markers (p) exceeds the population size (n), combined with the high correlation among the predictors (markers). This situation leads to a deficiency in the rank of the predictor matrix, rendering it impossible to calculate least-squares estimates for the effects of markers. The challenges arise from factors such as the issues related to dimensionality; specifically, in models where p is much greater than n , which are not likelihood identified and susceptible to overfitting, there may be spurious features and data structures captured (refer to ‘The complexity of genomic selection and prediction’ and ‘Solution to an inverse problem’ in the supplementary online information). Potential solutions to these challenges include: (i) the use of penalized regression; (ii) variable selection techniques; and (iii) dimensionality reduction methods (e.g., principal components), which create a new set of uncorrelated predictors from the original markers, thus facilitating the application of univariate distributions and reducing the computation time for estimates and predictions. A fourth approach involves utilising statistical models that evaluate the complexities of GP alongside high-density marker platforms and $G \times E$ interactions, thereby enhancing the strength of GP models. Generally, theoretical investigations indicate good prediction accuracies for complex traits like grain yield and other traits assessed using independent random cross-validation data partitioning. In contrast to the common application of GP for predicting the performance of a single trait in TST populations using data from the same trait observed in TRN populations, the complexities of extending this approach to multi-trait GP indices have not been extensively explored, apart from a method proposed by Cerón-Rojas, which is based on the multi-trait Genomic Best Linear Unbiased Estimator (GBLUP) selection index, that worked well when tested on both simulated and real data sets (Crossa *et al.*, 2017).

Simulation and empirical findings derived from random cross-validation indicate that GS improves genetic gains by either accelerating the breeding process (rapid selection cycles) or increasing the efficiency of field evaluations. The outcomes of applying random cross-validation on maize and wheat breeding datasets demonstrate that GS can substantially improve prediction accuracy concerning pedigree and MAS for traits with low heritability. The application of GS in the breeding of maize and wheat shows its effectiveness in selection processes.

Breeding programs globally have been exploring and implementing GS and GP across various crops. Concurrently, significant research has led to the development of innovative statistical techniques that incorporate pedigree, genomic data, and environmental covariates (such as weather data) into statistical-genetic prediction models. GBLUP models are commonly employed in GP, and enhancing GBLUP to account for GxE interactions has increased the accuracy of predicting cultivars that have not yet been observed in specific environments. New approaches for evaluating the accuracy of GP in the context of discrete response variables (like ordinal disease data, rates, count data, etc.) have been introduced, along with Bayesian genomic models designed for the analysis of multiple traits across various environments. A computationally efficient Markov Chain Monte Carlo (MCMC) method has been established, which yields full conditional distributions of the parameters, facilitating exact Gibbs sampling for posterior distributions. Findings from simulated scenarios and two extensive data sets indicate that when the correlation among traits is significant, a model employing an unstructured covariance matrix is more effective than diagonal and conventional methods for enhancing prediction accuracy for grain yield. Conversely, in situations where correlations are low, the standard model suffices.

In a study on chickpeas, a total of 320 elite breeding lines were genotyped using Diversity Array technology (DArTseq) and phenotyped for yield-related traits across two environments under two treatments (i.e., rainfed and irrigated) during two different seasons. Multiple statistical models (RR-BLUP, Kinship GAUSS, Bayes Cp, Bayes B, Bayesian LASSO, and random forest regression or RFR) achieved high prediction accuracies for the targeted traits; however, minimal variation in prediction accuracy among the models was observed. Incorporating population structure into the model resulted in a slight enhancement of prediction accuracies for days to maturity (DM), days to flowering (DF), and seed dry weight (SDW), but not for seed yield (SY) (Roorkiwal *et al.* 2016, Varshney, R.K. 2016).

Integrating high-density marker platforms with GxE interactions enhances the accuracy of GP models; this has been thoroughly studied in bread wheat, maize, and legumes. In every GP model that includes GxE interactions, the accuracy compared to single-environment analyses improved by an average of 10–40% across all three crop types. The primary models utilised to evaluate GP accuracy by incorporating GxE interactions and their application to real data are outlined below.

Multienvironment trials for assessing GxE interactions play a crucial role in plant breeding by helping to select lines that perform well and exhibit stability across different environments. Burgueño et al. (2012) were pioneers in applying marker- and pedigree-based GBLUP models to evaluate GxE interactions in the context of genomic prediction, while Heslot et al. (2014) integrated crop-modeling data to analyze genomic GxE interactions. Jarquín et al. (2014) developed a reaction norm model that extends the traditional GBLUP model by incorporating both main and interaction effects of markers along with environmental covariates, utilising high-dimensional random variance-covariance structures.

Genetic Gains from Rapid Selection Cycle GS: There are few studies that assess the genetic gains achieved through a rapid selection cycle based on GS. The initial research confirming the potential of a rapid selection cycle in GP of biparental populations, along with earlier results from random cross-validation studies, was carried out by Massman *et al.* (2013) and indicated that GS enhanced maize genetic gains over time. Genetic advancements were also reported by Asoro *et al.* (2013) in oats and by Rutkoski *et al.* (2015) in wheat, illustrating that both GS and PS produced comparable realised genetic gains over time.

Another instance of genetic gains resulting from rapid-cycle GS on CIMMYT maize involves two biparental maize populations (F2:3) from Asia (CAP1 and CAP2) that were created and assessed for testcross performance in both drought and optimal conditions (Vivek *et al.*, 2017). The annual genetic gains for PS compared to GS in drought conditions were 0.067 t/ha versus 0.124 t/ha for CAP1, and 0.076 t/ha versus 0.104 t/ha for CAP2. In optimal conditions, the corresponding annual genetic gains for PS versus GS were 0.084 t/ha versus 0.140 t/ha for CAP1, and 0.123 t/ha versus 0.13 t/ha for CAP2. The findings of this research demonstrated that GS of superior plant phenotypes led to rapid genetic advancements in drought tolerance in maize.

The primary goal of GS is to lower the costs associated with phenotyping by utilising markers and to enhance genetic gains, while the purpose of high-throughput techniques is to assess high-density phenotypes for a vast number of individuals or breeding lines over time and varying locations using remote or proximal sensing methods. This approach can boost both the accuracy and intensity of selection and, as a result, improve the selection response while also reducing phenotyping costs. The main concept of high-throughput is to employ secondary traits

that are linked to grain yield, disease resistance, or end-use quality, which can be advantageous in the early-generation evaluation of lines.

Predicting hybrid performance in rice using genomic best linear unbiased prediction

Genomic selection for enhancing pure breeds relies on marker information, resulting in cost reductions by allowing early selection prior to measuring phenotypes. When this approach is applied to hybrid breeding, it is expected to be even more effective since the genotypes of hybrids are predetermined by their inbred parent lines. In the case of rice, researchers have introduced and employed a sophisticated technique to predict hybrid performance, utilising a subset of all possible hybrids as a training sample to estimate trait values for all potential hybrids. This technique is referred to as genomic best linear unbiased prediction. The technology utilised for hybrids is known as genomic hybrid breeding. They selected 278 hybrids randomly, originating from 210 recombinant inbred lines of rice, to serve as a training sample and used them to predict the performance of all 21,945 potential hybrids. The average yield of the top 100 selected hybrids demonstrates a 16% improvement compared to the average yield of all potential hybrids. This novel strategy of marker-based yield prediction for hybrids acts as a proof of concept for a new technology that could potentially transform hybrid breeding (Shizhong Xua *et al.*, 2014).

Genomic Selection and Association Mapping in Rice

Genomic Selection (GS) represents an innovative breeding technique where genome-wide markers are utilized to estimate the breeding value of individuals within a breeding population. Research has demonstrated that GS enhances breeding efficiency in dairy cattle and various crop species, with scientists assessing its effectiveness for the first time in breeding inbred rice lines. They conducted a genome-wide association study (GWAS) alongside five-fold GS cross-validation using a population of 363 elite breeding lines from the International Rice Research Institute's (IRRI) irrigated rice breeding program and herein report the GS results. The population was genotyped with 73,147 markers through genotyping-by-sequencing. The training population, the statistical method used to construct the GS model, the number of markers, and the traits assessed to evaluate their impact on prediction accuracy. For all three traits analyzed, genomic prediction models surpassed predictions based solely on pedigree information. Prediction accuracies ranged between 0.31 and 0.34 for grain yield and plant height, while flowering time achieved an accuracy of 0.63. Analyses of subsets from the

complete marker set suggest that utilizing one marker every 0.2 cM is adequate for genomic selection in this collection of rice breeding materials. The RR-BLUP statistical method exhibited the best performance for grain yield in cases where no significant effect QTL were identified by GWAS, whereas for flowering time, where a significant large effect QTL was found, the non-GS multiple linear regression method outperformed GS models. In the case of plant height, where four mid-sized QTL were identified by GWAS, random forest produced the most consistently accurate GS models. These findings indicate that GS, guided by GWAS insights into genetic architecture and population structure, has the potential to become a powerful tool for enhancing the efficiency of rice breeding as genotyping costs continue to decrease (Jennifer *et al.*, 2015).

Concluding Remarks: Numerous statistical techniques have been established to estimate unobserved individuals in genomic selection (GS). Generally, linear models (such as GBLUP) and machine-learning approaches have proven effective in identifying intricate patterns and making accurate decisions based on available data. Kernel-based techniques, including reproducing kernel Hilbert spaces (RKHS), have frequently provided reliable genomic predictions in plant science. Various statistical models derived from the conventional GBLUP that account for genotype-by-environment (G×E) interactions in genomic and pedigree assessments have significantly enhanced the accuracy of forecasting unobserved individuals across different environments. These GS prediction models can assist researchers in various fields to create plants that can withstand drought and heat by leveraging favorable G×E interactions. It is crucial to model multi-trait multi-environment scenarios to enhance the predictive accuracy of the performance of newly developed lines in the coming years.

The application of statistical models in advanced hyperspectral imaging technology for high throughput, along with genomic and pedigree data during early-stage testing, provides a chance to speed up genetic improvements by intensifying selection efforts. Deep machine-learning techniques that utilize neural networks seem to hold promise for enhancing the precision of genomic-enabled predictions. Genomic selection distinctly outperforms pedigree breeding and marker-assisted selection (MAS) in advancing genetic improvements for complex traits. The effective combination of genotyping platforms with accurate phenotyping systems will further improve prediction accuracy and expedite genetic progress by reducing the breeding cycle duration. Additional research is needed to integrate genomic selection with high throughput as a standard element in plant breeding initiatives.

Developing GP models for gene bank accessions will be crucial to access unexplored diversity and accelerating the integration of valuable traits into breeding programs. At present, GS is the most promising method of breeding for enhancing the speed of developing and releasing new genotypes; thus, utilizing GS to establish gene pools and populations from diverse gene bank accessions deserves thorough and focused investigation, particularly in light of the susceptibility of elite lines and hybrids to the severe impacts of climate change.

R codes to implement GS models (Source: Osval *et al.*, 2019)

```
# Clear the memory
rm(list=ls())
library(BMTME)
library(BGLR)
pheno=read.table(file="nlow.txt",header=T)
geno=read.table(file="Geno.txt",header=T)
head(pheno)
tail(pheno)
dim(pheno)
#Genomic relationship matrix
geno[1:10,1:10]
dim(geno)
#Design matrices#
LG <- cholesky(geno)
ZG <- model.matrix(~0 + as.factor(pheno$Line))
Z.G <- ZG%*%LG
pheno1 <- data.frame(GID = pheno[, 1], Env = pheno[, 2],
                    Response = pheno[, 3])
nCV=5
CrossV <- CV.KFold(pheno1, DataSetID = 'GID', K = nCV, set_seed = 123)
CrossV$CrossValidation_list
y2=(pheno[, 3])
length(2)
y=y2
tst_set=CrossV$CrossValidation_list[[1]]
tst_set
#Predictor eta=mu+G#
ETA=list(Gen=list(X=Z.G, model="BRR"))
y[tst_set]=NA
fm1=BGLR(y=y,ETA=ETA,nIter=1000,burnIn=500,verbose = F)
#str(fm1)
#Prediction of testing set#
predicted=c(fm1$yHat[tst_set])
Observed=y2[tst_set]
plot(Observed,predicted)
MSE=mean((Observed-predicted)^2)
MSE
```

```

Obs_Pred=cbind(Observed,predicted)
Obs_Pred
colnames(Obs_Pred)=c("Observed","Predicted")
head(Obs_Pred)
plot(Observed,predicted)
#Five fold cross validation with predictor Predictor eta=mu+G
Ave_MSE=c()
for (i in 1:nCV){
  y=y2
  #Five hold-out Cross validation#
  tst_set=CrossV$CrossValidation_list[[i]]
  tst_set
  #Predictor eta=mu+G#
  ETA=list(Gen=list(X=Z.G, model="BRR"))
  y[tst_set]=NA
  fm1=BGLR(y=y,ETA=ETA,nIter=1000,burnIn=500,verbose = F)
  #str(fm1)
  #Prediction of testing set#
  predicted=c(fm1$yHat[tst_set])
  Observed=y2[tst_set]
  plot(Observed,predicted)
  MSE=mean((Observed-predicted)^2)
  MSE
  Ave_MSE=c(Ave_MSE,MSE)
}
Ave_MSE
Ave1=mean(Ave_MSE)

```

Conclusion

Genomic selection facilitates the rapid selection of elite genotypes with accelerated speed of breeding cycle. Genomic selection (GS) is an advanced selection procedure over MAS. As it aims to use genome-wide markers to estimate the effects of all loci and thereby compute a genomic estimated breeding value (GEBV), to achieve more comprehensive and reliable selection. Major challenge in genomic prediction (GP) is well documented $p > n$ statistical problem. Many models have been proposed to overcome this problem. At the end, no single models are universally applicable in all kinds of data set, based on the assumptions of the data set one must choose proper model. On other hand, machine learning models are the promising alternative to statistical models in genomic selection.

REFERENCES:

- Arruda, M.P. et al. (2015) Genomic selection for predicting fusarium head blight resistance in a wheat breeding program. *Plant Genome*, 8, 1–12.
- Asoro, F.G. et al. (2013) Genomic, marker-assisted, and pedigree-BLUP selection methods for b-glucan concentration in elite oat. *Crop Sci.* 53, 1894–1906
- Bernardo, R. 2016. Bandwagons I, too, have known. *Theor. Appl. Genet.* 129, 2323–2332
- Crossa, J., Perez-Rodriguez, P., Cuevas, J., Montesinos-Lopez, O., Jarquin, D., de los Campos, G, Burgueno, J. ~ et al. (2017) Genomic selection in plant breeding: methods, models, and perspectives. *Trends Plant Sci.* 22, 961–975
- Pszczola, M. et al. (2012) Reliability of direct genomic values for animals with different relationships within and to the reference population. *J. Dairy Sci.* 95, 389–400
- Poland, J. and Rutkoski, J. (2016) Advances and challenges in genomic selection for disease resistance. *Annu. Rev. Phytopathol.* 54, 79–98
- Rutkoski, J. et al. (2015) Genetic gain from phenotypic and genomic selection for quantitative resistance to stem rust of wheat. *Plant Genome* 8, 1–10
- Shizhong Xua , Dan Zhuh , and Qifa Zhangb, 2014. Predicting hybrid performance in rice using genomic best linear unbiased prediction. *PNAS.* 111 (34) www.pnas.org/cgi/doi/10.1073/pnas.1413750111.
- Varshney, R.K. (2016) Exciting journey of 10 years from genomes to fields and markets: some success stories of genomicsassisted breeding in chickpea, pigeonpea and groundnut. *Plant Sci.* 242, 98–107.
- Vivek, B.S. et al. (2017) Use of genomic estimated breeding values results in rapid genetic gains for drought tolerance in maize. *Plant Genome* 10, 1–8

Selection Index in Plant Breeding

Parashuram Patroti

Centre on Rabi Sorghum (ICAR-IIMR, Regional station), Solapur, Maharashtra

Email: parashuram@millets.res.in

Introduction

Selection index, a pivotal quantitative genetic tool in plant breeding, facilitates the simultaneous improvement of multiple traits, enhancing the efficiency and precision of breeding programs. It is a statistical method employed to predict the breeding value of an individual based on multiple traits, each weighted by its economic or agronomic importance (Sivakumar *et al.*, 2017). The selection index is constructed as a linear combination of phenotypic values, with the weights assigned to each trait determined to maximize the correlation between the index and the aggregate genotype or breeding value (Villanueva *et al.*, 2006). The primary objective of utilizing a selection index is to identify and select superior individuals that possess a desirable combination of traits, thereby accelerating genetic progress (Batista *et al.*, 2021). In essence, the selection index serves as a powerful tool for plant breeders to make informed decisions regarding which individuals to select and propagate, ultimately leading to the development of improved crop varieties.

The efficacy of selection indices hinges on several key factors, including the accurate estimation of genetic variances and covariances among traits, the determination of appropriate economic weights, and the consideration of genotype-by-environment interactions. Genetic variances and covariances provide insights into the genetic architecture of the traits and their interrelationships, enabling breeders to predict the response to selection. Economic weights reflect the relative importance of each trait in terms of its contribution to overall economic value or agronomic performance, guiding the selection process towards the most desirable trait combinations. Furthermore, accounting for genotype-by-environment interactions is crucial to ensure that the selected individuals exhibit consistent performance across diverse environmental conditions.

Background of Plant Breeding

Plant breeding, at its core, is an evolutionary process guided by human intervention, aiming to create crop varieties that are better suited to meet human needs and preferences. Traditional breeding methods, relying on phenotypic selection and hybridization, have been instrumental in shaping the genetic makeup of our cultivated plants over centuries. Breeders historically selected superior individuals based on phenotypic observations, which served as proxies for their breeding value, or the anticipated performance of their progeny (Allier et al., 2019). In order to more accurately assess the breeding value of individuals, phenotypic selection has been supplemented by pedigree-based prediction of breeding values and, more recently, genomic prediction of breeding values, which makes use of the accessibility of affordable high-density genotyping (Allier *et al.*, 2019).

However, these conventional approaches often lack precision in manipulating and selecting specific genes, potentially leading to unintended consequences (Aziz & Masmoudi, 2024). With the advent of molecular biology and genomics, plant breeding has undergone a revolutionary transformation, enabling breeders to manipulate genes with unprecedented precision. The integration of molecular markers and genomic information has revolutionized plant breeding, enabling breeders to make more informed decisions and accelerate the pace of genetic improvement (Anand *et al.*, 2023). Modern breeding objectives have evolved beyond merely enhancing yield to encompass improved quality and other value-added characteristics (Baenziger *et al.*, 2006).

Importance of Selection in Plant Breeding

Selection is an important part of plant breeding because it helps breeders find and promote plants with the best traits, which eventually leads to the creation of better crop types. It is an ongoing cycle that involves choosing superior plants from a population and using them to produce the next generation, with the ultimate goal of improving desired traits such as yield, quality, disease resistance, and abiotic stress tolerance. Breeders can effectively gather and focus beneficial genes over time by carefully selecting plants, increasing the frequency of favourable alleles within the population. The selection process is also essential for preserving genetic diversity in breeding populations to guarantee continued genetic advancement in the long run (Li *et al.*, 2022).

Plant breeders can respond to changing consumer demands, environmental constraints, and market opportunities by using selection strategies that enable them to successfully adjust to new breeding goals. Marker-assisted selection has become a valuable tool in plant breeding, allowing for the selection of plants based on the presence of specific DNA markers associated with desired traits (Lema, 2018). Using selection indices, breeders can simultaneously select for multiple traits, considering their economic or agronomic importance (Pérez-de-Castro *et al.*, 2012). By integrating biotechnology and genetic engineering into plant breeding, breeders can now introduce novel genes and traits into crop plants that were previously unattainable through conventional breeding methods (Low *et al.*, 2018). This capability has revolutionized the improvement of crops by enabling the precise manipulation of plant genomes (Arora & Narula, 2017).

The selection index is a pivotal statistical tool in plant breeding, designed to enhance the efficiency and effectiveness of selecting superior individuals from a population (Valente *et al.*, 2013). In cases where selection pressure has been mostly focused on production, the genetic diversity of numerous functional traits has diminished as a result of the negative antagonistic correlation with productivity traits and the absence of selection pressure to improve them (Sánchez-Molano *et al.*, 2016). The selection index combines information from multiple traits into a single score, which is then used to rank and select individuals based on their overall merit (Rembe *et al.*, 2022). The selection index, typically represented as a linear combination of multiple traits, provides a comprehensive measure of an individual's genetic worth, enabling breeders to make more informed selection decisions (Chung & Liao, 2020). This approach is especially valuable when dealing with complex traits that are influenced by multiple genes and environmental factors (Govindaraj *et al.*, 2015).

Objectives of Selection Index

The primary objective of the selection index is to improve the overall genetic merit of a population by selecting individuals that possess the most desirable combination of traits (Natalini *et al.*, 2021). Maximizing genetic gain is the overarching goal of any breeding program, and the selection index is designed to achieve this by identifying individuals with the highest breeding values for the traits of interest. The key objectives of utilizing a selection index include simultaneous improvement of multiple traits, weighting traits according to their economic or agronomic importance, and accounting for correlations among traits. It is essential

to consider the heritability of each trait when constructing a selection index. By incorporating heritability estimates into the index, breeders can more accurately predict the response to selection and make more informed decisions. Selection indices are also used to maximize the probability of generating top-performing offspring from selection candidates (Niehoff et al., 2024).

Additionally, breeders can effectively manage resources by focusing selection efforts on the most promising individuals and traits. In situations where heritability is considerably greater under non-stressful conditions than it is under stressful ones, an index that combines data from both types of settings is anticipated to be more efficient than selection that is only based on stress environment evaluation (Baker, 1994). Breeders can create more resilient and adaptive crops by including stress tolerance traits in selection indices, allowing them to adapt to changing environmental conditions. Selection indices offer a transparent and repeatable framework for assessing and selecting individuals, improving consistency and objectivity in the breeding process.

When multiple traits are assessed, it is useful to create an index, known as a selection index, that integrates data on all of the characteristics related to the dependent variable, such as yield (Abu-Ellail et al., 2020). A selection index is constructed using statistical procedures to provide a single value representing the overall merit of an individual based on its performance across multiple traits (Rehman et al., 2019). The index is calculated by weighting each trait based on its economic value and genetic correlation with other traits (McCarthy et al., 2007; Miglior et al., 2017). The use of computer science breakthroughs in recent years has made it possible to test novel statistical methods for simulating uncertainty in multi-trait selection in order to enhance selection (Akdemir et al., 2018).

Theoretical Basis of Selection Index

The theoretical basis of the selection index lies in the principles of quantitative genetics and statistical prediction. The foundation of the selection index is the concept of breeding value, which represents the genetic merit of an individual for a particular trait. The use of selection indices relies on the assumption that the traits included in the index are heritable, meaning that a portion of the observed variation in these traits is due to genetic factors. The basic form of a selection index can be expressed as:

$$I = b_1X_1 + b_2X_2 + \dots + b_nX_n,$$

where I is the index value,

X_i is the phenotypic value for the i th trait,

and b_i is the weight assigned to the i th trait.

The weights assigned to each trait in the index are calculated to maximize the correlation between the index value and the aggregate genotype, which represents the overall genetic merit of an individual.

The selection index can be optimized using different statistical methods, such as multiple regression or best linear unbiased prediction. The weights are typically estimated using statistical methods that consider the genetic and phenotypic variances and covariances among the traits (Lopez-Cruz & Campos, 2021). The accuracy of the selection index depends on the accuracy of the estimates of genetic and phenotypic parameters, as well as the number of traits included in the index. The use of a selection index can improve the efficiency of selection by allowing breeders to simultaneously consider multiple traits and select individuals with the best combination of desirable characteristics (Shah et al., 2016).

The selection index allows researchers to give each genomic region the weight it deserves in a breeding program by integrating the methodology with affordable, high-density markers, which facilitates the transition from selection based on a combination of "infinitesimal" effects plus a few major QTL to selection that captures all QTL effects. However, for the foreseeable future, gathering high-quality trait information will remain essential to using these technologies (Haley & Visscher, 1998). The development of selection indices requires careful consideration of the objectives of the breeding program, the traits to be included in the index, and the statistical methods used to estimate the weights.

Basic Principles

The creation and application of selection indices are based upon certain fundamental principles that ensure their efficacy in enhancing genetic progress. It is crucial to carefully define the breeding goal in order to identify the traits that are most important for enhancing the population's overall merit. The economic value of each trait should be determined based on its contribution to the overall profitability or value of the crop.

Additionally, one should consider the genetic parameters, including heritability, genetic correlations, and phenotypic correlations, among the traits included in the index. These

parameters are used to determine the weights assigned to each trait in the index. Statistical methods, such as multiple regression or best linear unbiased prediction, are used to estimate the weights that maximize the correlation between the index value and the aggregate genotype. The breeder should consider the accuracy of the selection index to determine how well it predicts the breeding value of an individual.

The selection index should be validated using independent data sets to ensure that it accurately predicts the performance of individuals in different environments. Breeders should routinely monitor the genetic gain achieved through selection to ensure that the selection index is effective in improving the population's overall merit. Breeders can maximize genetic progress and create superior cultivars that satisfy the demands of farmers and consumers by adhering to these fundamental principles.

Mathematical Model of Selection Index

The mathematical model of the selection index provides a quantitative framework for combining information on multiple traits to predict the overall genetic merit of an individual (Villanueva *et al.*, 2006). The index is constructed as a linear combination of phenotypic values, with each trait weighted by a coefficient that reflects its relative importance and genetic relationship with other traits. The selection index (I) can be expressed mathematically as follows: $I = b_1X_1 + b_2X_2 + \dots + b_nX_n$. In this equation, I represents the index value, which is a measure of the overall merit of an individual; X_i represents the phenotypic value for the i th trait, which is the observed measurement of that trait for the individual; and b_i represents the weight assigned to the i th trait, which reflects its relative importance and genetic relationship with other traits.

The weights (b_i) are calculated to maximize the correlation between the index value (I) and the aggregate genotype (H), which represents the true genetic merit of an individual. The aggregate genotype is defined as a linear combination of the breeding values for the traits included in the index: $H = a_1g_1 + a_2g_2 + \dots + a_ng_n$. Here, H represents the aggregate genotype, which is a measure of the overall genetic merit of an individual; g_i represents the breeding value for the i th trait, which is the genetic contribution of the individual to the next generation for that trait; and a_i represents the economic weight assigned to the i th trait, which reflects its relative importance in determining overall merit.

The weights (b_i) in the selection index equation are calculated to maximize the correlation between the index value (I) and the aggregate genotype (H). This maximization is typically achieved using statistical methods that consider the genetic and phenotypic variances and covariances among the traits. The accuracy of the selection index depends on the accuracy of the estimates of genetic and phenotypic parameters, as well as the number of traits included in the index (Paikhomba *et al.*, 2014). The genetic gain can be considerably increased when crosses are selected based on their genomic usefulness criterion compared to selection based on mean genomic estimated breeding values (Lehermeier *et al.*, 2017). In each cycle of a line or a hybrid breeding program, lines are selected which serve as the parents of the crosses from which the base population of the next breeding cycle is derived (Osthushenrich *et al.*, 2017).

Assumptions Underlying Selection Index

The selection index method relies on several key assumptions to ensure its effectiveness and accuracy in predicting genetic merit. One critical assumption is the linearity of the relationship between the index and the aggregate genotype; this assumes that the traits included in the index combine additively to determine overall merit, without significant interactions or non-linear effects (Foulley & Rouvier, 1971).

Another assumption is the accuracy of the estimated genetic and phenotypic parameters. Accurate estimates of heritability, genetic correlations, and phenotypic correlations are essential for calculating the index weights and predicting the response to selection. The third important assumption is the normality of the data, where the phenotypic values and breeding values of the traits are normally distributed.

Additionally, the selection index assumes a constant economic weight for each trait across different environments and populations. However, economic conditions, consumer preferences, or market demands can change, rendering the original weights obsolete.

The selection index method also assumes that there is no genotype-by-environment interaction. Genotype-by-environment interaction occurs when the performance of a genotype differs across different environments. When genotype-by-environment interaction is present, the selection index may not accurately predict the performance of individuals in all environments.

Construction of Selection Index

The construction of a selection index is a multi-step process that requires careful consideration of the traits to be included, the estimation of genetic and phenotypic parameters, and the calculation of index weights. The breeder must define the breeding goal and identify the traits that contribute to it to begin constructing a selection index. The selection index is particularly useful in plant breeding programs, enabling breeders to improve multiple traits simultaneously (Batista *et al.*, 2021). Selection index can be constructed by several different methods.

First, the breeder must define the breeding objective and identify the traits that contribute to it. For example, a breeder may want to improve yield, disease resistance, and grain quality in a wheat variety. In order to better estimate the breeding value of individuals, phenotypic selection has been complemented by pedigree-based prediction of breeding values and more recently by genomic prediction of breeding values, taking advantage of the availability of cheap high-density genotyping (Allier *et al.*, 2019).

Second, the breeder must estimate the genetic and phenotypic variances and covariances among the traits. This information is needed to calculate the index weights and predict the response to selection. The larger the databases are growing, the better statistical approaches for genomic selection will be available (Weckwerth *et al.*, 2020).

Third, the breeder must assign economic weights to each trait. The economic weights reflect the relative importance of each trait in determining overall merit. Economic weights can be difficult to determine, especially for traits that do not have a direct economic value.

Fourth, the breeder must calculate the index weights. The index weights are calculated to maximize the correlation between the index value and the aggregate genotype. The selection index method is used to select the best individuals from a population based on their index values.

Defining Selection Objectives

Defining selection objectives is a crucial first step in constructing a selection index, as it sets the direction and priorities for the breeding program. The goal of defining selection objectives should be specific, measurable, achievable, relevant, and time-bound.

A well-defined selection objective provides a clear target for the breeding program and helps to guide the selection of traits to be included in the index. The selection objectives should be

aligned with the overall goals of the breeding program and should reflect the needs of the target market or consumer. The efficient estimation of marker effects in plant breeding is enhanced by the increasing volume of genotypic and phenotypic data (Xavier, 2019).

Choosing Traits to Include in the Index

Selecting the appropriate traits to include in a selection index is a critical decision that can significantly impact the effectiveness of the breeding program. Traits should be heritable, meaning that they are influenced by genetic factors and can be passed down from parents to offspring.

The traits should be easy to measure and should have a low cost of measurement. Traits should be genetically correlated with other important traits.

Estimating Genetic and Phenotypic Parameters

Estimating genetic and phenotypic parameters is a crucial step in constructing a selection index, as these parameters provide the foundation for calculating index weights and predicting response to selection. These estimates provide insights into the genetic architecture of the traits, the extent to which they are influenced by environmental factors, and the relationships among them (Dekkers *et al.*, 2021).

Heritability estimates the proportion of phenotypic variance that is due to genetic factors (Nagalakshmi *et al.*, 2018). Genetic correlation measures the degree to which two traits are influenced by the same genes. Phenotypic correlation measures the degree to which two traits are associated with each other, regardless of the underlying genetic or environmental causes.

With high-density molecular marker data and phenotypic data, genomic prediction is now a standard method in many plant and animal breeding programs (Heslot & Feoktistov, 2020). The usefulness criterion, which accounts for variation in progeny variance, is a measure of the gain that can be obtained from a specific cross (Lehermeier *et al.*, 2017).

Calculating Index Weights

Calculating index weights is a central step in constructing a selection index, as these weights determine the relative contribution of each trait to the overall index value. The index weights are calculated to maximize the correlation between the index value and the aggregate genotype (Osthushenrich *et al.*, 2018).

The selection index method is a powerful tool for improving multiple traits simultaneously in plant breeding programs (Heslot & Feoktistov, 2020). A selection index was used as a comprehensive index for selecting high-yielding genotypes and explained a substantial proportion of grain yield variation (Abdolshahi et al., 2015). Selection for traits like grains spike is a good selection criteria and can be effective for future breeding programs (Jan et al., 2015).

The basic assumption underlying genomic selection is the presence of markers throughout the genome, some of which exhibit direct linkage with causal loci (Lenz et al., 2017). Genomic prediction models are initially developed using phenotypic and genotypic data from a training population (Chung & Liao, 2020). These models can then estimate the genetic values of genotypes that haven't been phenotyped, enhancing evaluations, especially when phenotypic data is scarce, like in early selection phases (Werner et al., 2020). This is because genomic selection uses genome-wide markers to predict genomic estimated breeding values, allowing for early selection of superior individuals, potentially without needing to phenotype them (O'Connor et al., 2021).

Genomic selection's effectiveness hinges on improvements in genotyping technology, statistical computing, and user-friendly software. It aims to boost genetic gain per unit of time compared to phenotypic selection, yet its practical use in plant breeding is still developing (Werner et al., 2020). Maximizing the effectiveness of genomic selection within breeding programs is essential to boost genetic advancements (Atanda et al., 2020). Computational algorithms employing genomic prediction are vital for strategically selecting breeding individuals, determining optimal cross numbers, and managing progeny production under constraints (Zhang & Wang, 2022). The integration of genomic data has revolutionized plant breeding, facilitating the identification and selection of plants with desirable traits, like drought tolerance, thereby optimizing resource utilization and reducing the need for extensive field testing (Moeinizade et al., 2019).

Validating the Selection Index

Validating the selection index is a crucial step to ensure that it is effective in improving the desired traits in the target population. It is important to apply genomic selection at the appropriate stage of the breeding cycle (Juliana et al., 2018). By reducing generation intervals and accurately evaluating traits early on, genomic selection proves to be an effective tool for

accelerating genetic gain in plant breeding (Lin et al., 2014). Genomic selection enhances genetic progress by predicting performance, even before phenotypic characterization.

Applying the Selection Index in a Breeding Program

The selection index is used to evaluate and rank individuals in the breeding population. The selection index can also be used to predict the response to selection. By utilizing genomic selection, plant breeders can enhance selection intensity, shorten breeding cycles, and boost prediction accuracy (Allier et al., 2019). Integrating genotyping data into breeding programs requires efficient DNA extraction and marker production to enable timely selection decisions (Nti-Addae et al., 2019).

An extension to genomic selection, called optimal haploid value selection, predicts the best doubled haploid that can be produced from a segregating plant (Daetwyler et al., 2015). This method is particularly useful in hybrid breeding programs, enabling breeders to identify superior parental lines and hybrid combinations early in the breeding process (Chen et al., 2023). Recurrent genomic selection, which involves repeatedly selecting and intercrossing the best individuals based on genomic predictions, can lead to long-term genetic gain in breeding populations (Gorjanc et al., 2018). Genomic selection has the potential to transform plant breeding by accelerating the rate of genetic gain and improving the efficiency of breeding programs (Pégard et al., 2020). Genomic approaches, when combined with hybridization and selection strategies that are based on a physiological understanding, can increase rates of genetic gains (Reynolds et al., 2010). Balancing genotyping costs with the potential advantages of rapid genetic improvement is crucial for plant breeders (Cockerton et al., 2021).

Genomically enabled prediction has great potential to accelerate the rate of genetic gain in plant breeding programs, complementing traditional breeding and marker-assisted selection strategies (Krishnappa et al., 2021). Genomically assisted selection has been proven to improve yield in plant breeding and reduce the time between breeding cycles (Vanavermaete et al., 2020).

Types of Selection Indices

Different types of selection indices are available, each designed to address specific breeding objectives and genetic architectures. These indices vary in complexity and assumptions,

offering breeders a range of options to tailor their selection strategies. Selection index includes the base index, restricted selection index, and independent culling levels.

Smith-Hazel Index

The Smith-Hazel index, one of the earliest and most fundamental selection indices, forms the cornerstone of multi-trait improvement in plant breeding. The Smith-Hazel index is a linear combination of traits, weighted by their economic values and genetic variances and covariances. This index aims to maximize the genetic gain in an aggregate economic value, considering the genetic relationships among traits. The Smith-Hazel index relies on accurate estimates of genetic parameters, which can be challenging to obtain in practice. The Smith-Hazel index provides a powerful framework for selecting superior individuals based on multiple traits.

Base Index

The base index method involves creating a selection index by summing up the product of the economic weight and the breeding value for each trait. The economic weight for each trait indicates its relative importance in determining the overall merit of an individual (Niehoff et al., 2024). The base index is suitable for situations where the economic values of the traits are well-defined.

Restricted Selection Index

The restricted selection index is used when breeders want to improve certain traits without changing others. This method involves setting constraints on the selection process to prevent undesirable changes in specific traits. The restricted selection index is useful in situations where there are specific market requirements or regulatory constraints that need to be met.

The effectiveness of the selection index depends on the accurate estimation of genetic parameters and economic weights.

Economic Selection Index

The economic selection index aims to optimize the economic value of the selected individuals or lines. By incorporating economic data, the economic selection index helps breeders make informed decisions that align with market demands and profitability goals.

Package	Key Use Cases	Strengths	Notes
SelectionIndex	Classical, restricted, and ideal genotype indices	Direct functions for Smith-Hazel, restricted, and ideal index	Most recommended for teaching and applied breeding
sommer	BLUP estimation, variance components, genetic correlation	Can be used to derive genetic (G) matrix, heritability	Powerful for multi-environment data
rrBLUP	Genomic selection BLUPs	Can provide GEBVs for use in index	Often used before index construction
AGHmatrix	Relationship matrices from markers/pedigree	Helps build additive/dominance matrices	Supports genomic selection index
asreml-R	Mixed models for breeding data	Commercial, but gold standard in variance modelling	Needed for large breeding trials
lme4	Simple mixed models	Free and easy for single-trait models	Can extract BLUPs as inputs for index
psych	Trait correlation and factor analysis	Helps evaluate redundancy among traits	Useful for index trait selection
FactoMineR / PCAmixdata	PCA-based trait weighting or index	Helps construct synthetic indices	Useful when trait weighting is ambiguous
tidyverse (esp. dplyr , ggplot2)	Data wrangling and visualization	Helps format, filter, rank and plot index results	Essential for report-ready outputs
BGLR	Bayesian genomic prediction	For use in genomic selection indices	Can estimate marker effects and GEBVs

Step-by-Step Procedure to Construct a Selection Index in R

✓ Step 1: Prepare Your Data

You need:

- **Phenotypic data** (trait values per genotype)
- **Genotypic or pedigree data** (optional but useful for building relationship matrices)

Example phenotypic dataset:

Genotype	Yield	Height	Tiller
G1	50.2	145.0	4.1
G2	52.3	140.2	3.9
...

✓

✓ Step 2: Install and Load Required R Packages

```
install.packages(c("AGHmatrix", "SelectionIndex", "tidyverse"))
```

```
library (AGHmatrix)      # For relationship matrices
```

```
library (SelectionIndex) # To compute selection index
```

```
library (tidyverse)      # For data manipulation and plotting
```

✓ **Step 3: Simulate or Import Your Data**

A. Simulated Genotypic Data (if you don't have SNPs)

```
geno <- matrix(sample(0:2, 1000, replace = TRUE), nrow = 10)
```

```
rownames(geno) <- paste0("G", 1:10)
```

B. Simulated Phenotypic Data

```
traits <- data.frame(
```

```
  ID = rownames(geno),
```

```
  Yield = rnorm(10, 50, 5),
```

```
  Height = rnorm(10, 150, 10),
```

```
  Tiller = rnorm(10, 4, 1)
```

✓ **Step 4: Build Relationship Matrix (optional but recommended)**

Using SNP data → G-matrix via AGHmatrix

```
CopyEdit
```

```
G <- Gmatrix(SNPmatrix = geno, method = "VanRaden", ploidy = 2)
```

✓ **Step 5: Compute Phenotypic and Genetic Covariance Matrices**

```
trait_data <- traits[, c("Yield", "Height", "Tiller")]
```

```
# Phenotypic variance-covariance matrix (P)
```

```
P <- cov(trait_data)
```

```
# Define heritabilities for traits (estimated or assumed)
```

```
herit <- c(0.4, 0.5, 0.3)
```

```
# Genetic variance-covariance matrix (G)
```

```
G_mat <- P * diag(herit)
```

✓ **Step 6: Define Economic Weights**

Economic weights (vector **b**) reflect trait importance:

```
b <- c(1.0, -0.3, 0.5) # Ex: Increase Yield, reduce Height, moderate Tiller
```

✓ **Step 7: Compute the Selection Index**

```
index_result <- SelIndex(P = P, G = G_mat, b = b)
```

```
# Attach index to the genotype data
```

```
traits$SelectionIndex <- index_result$Index
```

✓ Step 8: Rank and Visualize Genotypes

```
traits %>%  
  arrange(desc(SelectionIndex)) %>%  
  ggplot(aes(x = reorder(ID, SelectionIndex), y = SelectionIndex)) +  
  geom_col(fill = "darkgreen") +  
  coord_flip() +  
  theme_minimal() +  
  labs(title = "Genotype Ranking by Selection Index",  
       x = "Genotype", y = "Selection Index")
```

Advantages and Limitations

The selection index offers several advantages, including increased selection efficiency and the ability to handle multiple traits simultaneously. However, it also has limitations, such as the requirement for accurate parameter estimates and the potential for reduced genetic diversity. When utilizing index selection in the case of non-additive traits, it's important to note a couple of issues: Firstly, CCPS selects animals individually, while non-additive effects manifest in the progeny and subsequent descendants of mating pairs (Li *et al.*, 2006). Consequently, the actual genetic gain observed in the offspring of CCPS-selected parents may not align precisely with predictions. Secondly, for traits governed by non-additive genetic factors, the relationship between an animal's genotype and its breeding value is not linear.

Advantages of Using Selection Index

Selection indices offer breeders a systematic and quantitative approach to improve multiple traits simultaneously. By combining information from multiple traits into a single index, selection indices can increase the efficiency of selection compared to single-trait selection (Li *et al.*, 2006). Selection indices allow breeders to balance the improvement of different traits based on their relative economic importance.

Limitations and Challenges

The success of selection indices depends on accurate estimates of genetic parameters such as heritabilities and genetic correlations. These limitations highlight the importance of careful

planning, data collection, and validation when using selection indices in plant breeding programs.

Although selection indices provide a rational means of selection, improvement is often limited to the traits included in the index. Another challenge is that traits may also have non-linear relationships or diminishing returns, which are not well-captured by linear selection indices.

Strategies to Overcome Limitations

Genomic selection offers a promising avenue to overcome some of the limitations associated with traditional selection indices (Jannink, 2010). By integrating genomic information into the selection process, breeders can improve the accuracy of selection and accelerate genetic gain (Esfandyari *et al.*, 2015). The effectiveness of genomic selection depends on the size and composition of the reference population used for the breeding objective (Grevenhof & Werf, 2015). By leveraging genomic data and optimizing selection strategies, breeders can minimize inbreeding while maximizing genetic gain (Sonesson *et al.*, 2012). Corrective mating programs are widely used in some species, and these can be modified to consider selection for economic merit adjusted for inbreeding depression (Weigel, 2001).

Conclusion

Selection indices play a crucial role in modern plant breeding by enabling breeders to make informed decisions and maximize genetic gain. Selection index theory has provided a valuable framework for breeders to make selection decisions when multiple traits are considered. By carefully considering the objectives, traits, and genetic parameters involved, breeders can design effective selection indices that drive crop improvement and meet the challenges of a changing world. However, in modern breeding programs, rapid genetic progress can lead to inbreeding via heavy impact of a few selected individuals or families (Weigel, 2001). Therefore, selection indices need to be combined with proper mating strategies to control inbreeding while maximizing genetic gain (Voss-Fels *et al.*, 2019). The creation of new populations from local landraces can help to alleviate the environmental effects impacting yields and can be used in breeding programs to select new and improved populations (Masoni *et al.*, 2019). Genomic selection can improve breeding efficacy by shortening the breeding cycle and facilitating the selection of candidate lines for creating hybrids without phenotyping in various environments (Liu *et al.*, 2019). Commonly employed marker-assisted selection strategies are not well suited for complex traits of agronomic importance, which necessitates

additional time for field-based phenotyping to pinpoint agronomically superior lines. Genomic selection represents an emerging alternative to marker-assisted selection that uses all marker information to calculate genomic estimated breeding values for complex traits, thus selections are made directly on GEBV without further phenotyping (Heffner et al., 2010). In ryegrass, simulations showed a four-year reduction in cycle time, with genetic gain doubling or tripling when GS is incorporated into the breeding program (Zhao et al., 2023). . It is essential to consider the optimization of mating strategies in GS breeding programs to balance short- and long-term genetic gain when selecting crosses (Allier et al., 2019).

Future Directions in Selection Index Research

Further research is needed to refine selection index methodologies, develop new approaches for incorporating non-linear relationships, and integrate genomic information into selection indices.

References

- Abdolshahi, R., Nazari, M., Safarian, A., Sadathossini, T. S., Salarpour, M., & Amiri, H. (2015). Integrated selection criteria for drought tolerance in wheat (*Triticum aestivum* L.) breeding programs using discriminant analysis. *Field Crops Research*, 174, 20. <https://doi.org/10.1016/j.fcr.2015.01.009>
- Abu-Ellail, F. F. B., Hussein, E., & El-Bakry, A. M. (2020). Integrated selection criteria in sugarcane breeding programs using discriminant function analysis. *Bulletin of the National Research Centre/Bulletin of the National Research Center*, 44(1). <https://doi.org/10.1186/s42269-020-00417-6>
- Akdemir, D., Beavis, W. D., Fritsche-Neto, R., Singh, A. K., & Sánchez, J. I. y. (2018). Multi-objective optimized genomic breeding strategies for sustainable food improvement. *Heredity*, 122(5), 672. <https://doi.org/10.1038/s41437-018-0147-1>
- Allier, A., Lehermeier, C., Charcosset, A., Moreau, L., & Teyssèdre, S. (2019). Improving Short- and Long-Term Genetic Gain by Accounting for Within-Family Variance in Optimal Cross-Selection. *Frontiers in Genetics*, 10. <https://doi.org/10.3389/fgene.2019.01006>
- Atanda, S. A., Olsen, M., Burgueño, J., Crossa, J., Dzidzienyo, D. K., Beyene, Y., Gowda, M., Dreher, K., Zhang, X., Prasanna, B. M., Tongoona, P., Danquah, E. Y., Olaoye, G., &

- Robbins, K. R. (2020). Maximizing efficiency of genomic selection in CIMMYT's tropical maize breeding program. *Theoretical and Applied Genetics*, 134(1), 279. <https://doi.org/10.1007/s00122-020-03696-9>
- Aziz, M. A., & Masmoudi, K. (2024). Molecular breakthroughs in modern plant breeding techniques. *Horticultural Plant Journal*. <https://doi.org/10.1016/j.hpj.2024.01.004>
- Babariya, S. G. P. K. R. and C. A. (2020). Selection indices for yield improvement in bread wheat (*Triticum aestivum* L.). *Electronic Journal of Plant Breeding*, 11(1). <https://doi.org/10.37992/2020.1101.056>
- Chen, S.-P., Tung, C., Wang, P.-H., & Liao, C. (2023). A statistical package for evaluation of hybrid performance in plant breeding via genomic selection. *Scientific Reports*, 13(1). <https://doi.org/10.1038/s41598-023-39434-6>
- Daetwyler, H. D., Hayden, M., Spangenberg, G., & Hayes, B. J. (2015). Selection on Optimal Haploid Value Increases Genetic Gain and Preserves More Genetic Diversity Relative to Genomic Selection. *Genetics*, 200(4), 1341. <https://doi.org/10.1534/genetics.115.178038>
- Gorjanc, G., Gaynor, R. C., & Hickey, J. M. (2018). Optimal cross selection for long-term genetic gain in two-part programs with rapid recurrent genomic selection. *Theoretical and Applied Genetics*, 131(9), 1953. <https://doi.org/10.1007/s00122-018-3125-3>
- Haley, C., & Visscher, P. M. (1998). Strategies to Utilize Marker-Quantitative Trait Loci Associations [Review of Strategies to Utilize Marker-Quantitative Trait Loci Associations]. *Journal of Dairy Science*, 81, 85. *Elsevier BV*. [https://doi.org/10.3168/jds.s0022-0302\(98\)70157-2](https://doi.org/10.3168/jds.s0022-0302(98)70157-2)
- Isaía, J., & Maich, R. (2009). Realized heritability estimates during a cyclical process of selection and intercrossing in bread wheat and hexaploid triticale. *Cereal Research Communications*, 37(4), 559. <https://doi.org/10.1556/crc.37.2009.4.10>
- Jan, S., Mohammad, F., & Khan, F. U. (2015). Genetic potential and heritability estimates of yield traits in F3 segregating populations of bread wheat. *International Journal of Environment*, 4(2), 106. <https://doi.org/10.3126/ije.v4i2.12630>

- Jannink, J. (2010). Dynamics of long-term genomic selection. *Genetics Selection Evolution*, 42(1). <https://doi.org/10.1186/1297-9686-42-35>
- Khan, K. D., Alex, R., Yadav, A., Sahana, V. N., Upadhyay, A., Mani, R. V., Kumar, T. S., Pillai, R. R., Vohra, V., & Gowane, G. R. (2025). Optimizing the Genomic Evaluation Model in Crossbred Cattle for Smallholder Production Systems in India. *Agriculture*, 15(9), 945. <https://doi.org/10.3390/agriculture15090945>
- Lenz, P., Beaulieu, J., Mansfield, S. D., Clément, S., Desponts, M., & Bousquet, J. (2017). Factors affecting the accuracy of genomic selection for growth and wood quality traits in an advanced-breeding population of black spruce (*Picea mariana*). *BMC Genomics*, 18(1). <https://doi.org/10.1186/s12864-017-3715-5>
- Nagalakshmi, R. M., Ravikesavan, R., Paranidharan, V., Manivannan, N., Firoz, H., Vignesh, M., & Senthil, N. (2018). Genetic variability, heritability and genetic advance studies in backcross populations of maize (*Zea mays* L.). *Electronic Journal of Plant Breeding*, 9(3), 1137. <https://doi.org/10.5958/0975-928x.2018.00142.4>
- Rehman, K., Sofi, P. A., Ara, A., & Dar, S. A. (2019). Multivariate analysis based on drought tolerance indices for screening drought tolerance in common bean (*Phaseolus vulgaris* L.). *Electronic Journal of Plant Breeding*, 10(1), 177. <https://doi.org/10.5958/0975-928x.2019.00021.8>
- Sánchez-Molano, E., Pong-Wong, R., & Banos, G. (2016). Genomic-Based Optimum Contribution in Conservation and Genetic Improvement Programs with Antagonistic Fitness and Productivity Traits. *Frontiers in Genetics*, 7. <https://doi.org/10.3389/fgene.2016.00025>
- Xavier, A. (2019). Efficient Estimation of Marker Effects in Plant Breeding. *G3 Genes Genomes Genetics*, 9(11), 3855. <https://doi.org/10.1534/g3.119.400728>
- Zhang, Z., & Wang, L. (2022). A look-ahead approach to maximizing present value of genetic gains in genomic selection. *G3 Genes Genomes Genetics*, 12(8). <https://doi.org/10.1093/g3journal/jkac136>
- Zhao, H., Lin, Z., Khansefid, M., Tibbits, J., & Hayden, M. (2023). Genomic prediction and selection response for grain yield in safflower. *Frontiers in Genetics*, 14. <https://doi.org/10.3389/fgene.2023.1129433>

Metagenomics: Introduction and Applications

Ratna Prabha, SC Mehta

ICAR-National Research Centre on Equines, Bikaner, Rajasthan, India,

Email: ratnasinghbiotech30@gmail.com

Overview

The field of metagenomics represents a revolutionary approach to understand microbial ecosystems through direct analysis of genetic material extracted from environmental systems. Metagenomics removes the traditional need of culturing individual microorganisms, enabling researchers to explore the vast universe of unculturable microbial life forms. Metagenomics studies provide unprecedented insights into the structural organization, biodiversity patterns, and metabolic capabilities of complex microbial systems across diverse habitats ranging from terrestrial to aquatic systems and host-associated environments. The integration of next-generation sequencing platforms has transformed this area, allowing for in-depth characterization of microbial community architecture and functional dynamics.

The computational demands of metagenomic research requires analytical frameworks and standardized processing protocols to ensure data integrity and meaningful biological interpretation. Advanced bioinformatics tools serve as the foundation for sequence assembly, functional annotation, and ecological analysis, enabling the identification of genetic polymorphisms, metabolic pathways, and inter-species relationships within complex microbial networks.

2. Agricultural Applications of Metagenomics Analysis

2.1. Soil Ecosystem Enhancement: Metagenomic analysis of agricultural soils reveals the intricate microbial networks responsible for biogeochemical cycling and fertility maintenance. This allows for the discovery of beneficial microorganisms that enhance soil productivity and plant nutrition, potentially reducing dependency on synthetic fertilizers.

2.2. Plant-Microbe Interaction Studies: Investigation of plant-associated microbial communities through metagenomic analysis promotes understanding of beneficial symbiotic relationships. This analysis identifies plant growth-promoting bacterial populations that enhance nutrient acquisition and stress tolerance in crop species.

2.3. Resistance Gene Identification: Metagenomic studies enables tracking of antimicrobial resistance determinants across agricultural landscapes, supporting the development of containment strategies to prevent their proliferation.

2.4. Climate Adaptation Strategies: Understanding plant-microbe interactions through metagenomic approaches contributes to developing climate-resilient agricultural practices by elucidating adaptive mechanisms in changing environmental conditions.

3. Different Metagenomics Approaches

3.1. Whole-Genome Shotgun Sequencing: This approach captures the complete genetic repertoire present in environmental samples through unbiased sequencing of entire genetic material. The methodology provides:

- Complete genomic coverage without selective targeting
- Elimination of PCR-associated biases through direct sequencing
- Requires extensive sequencing coverage to achieve adequate representation
- Generation of complex datasets demands advanced computational analysis
- Capability for comprehensive functional gene discovery and pathway elucidation
- Detailed characterization of microbial community composition including rare taxa

3.2. Targeted Amplicon Sequencing: This approach employs specific markers for microbial identification and characterization:

- Utilization of taxonomically informative gene sequences (e.g., 16S ribosomal RNA)
- Implementation of PCR amplification with associated methodological biases
- Reduced sequencing requirements offering cost-effective analysis
- Simplified datasets compatible with standard analytical tools
- Primary application in taxonomic profiling of microbial assemblages
- Widespread adoption in ecological and environmental research

4. Steps in metagenomics data analysis

An overview of the key steps involved in metagenomics data analysis is provided below:

4.1 Sample Acquisition and Processing: Environmental sample collection employs different techniques to prevent contamination and preserve microbial community integrity. Subsequent DNA extraction utilizes optimized protocols to maintain nucleic acid quality and quantity.

4.2 Library Construction and Sequencing: Preparation of sequencing libraries involves DNA fragmentation and adapter ligation, followed by high-throughput sequencing using platforms such as Illumina, PacBio, or Oxford Nanopore technologies.

4.3 Quality Control and Data Preprocessing: Raw sequencing data undergoes rigorous quality assessment using specialized software (FastQC, MultiQC) to identify issues associated with low-quality reads, adapter contamination, and sequencing artifacts. Data cleaning procedures remove poor-quality sequences and adapter sequences using tools like Trimmomatic or Cutadapt.

4.4 Sequence Assembly and Validation: Processed reads are assembled into contiguous sequences using specialized assemblers (SPAdes, MEGAHIT) designed for metagenomic data complexity. Assembly quality assessment employs tools like QUAST and MetaQUAST, followed by binning procedures to reconstruct metagenome-assembled genomes.

4.5 Taxonomic Assignment and Classification: Sequence identification utilizes reference databases (SILVA, Greengenes, NCBI) and classification tools (Kraken, QIIME) employing various approaches including marker gene analysis, k-mer classification, and alignment-based methods.

4.6 Functional Gene Annotation: Gene identification and annotation within assembled sequences uses specialized tools (Prokka, Prodigal, MetaGeneMark) to predict coding regions and assign functional categories.

4.7 Metabolic Pathway Analysis: Functional characterization involves mapping sequences to metabolic databases (KEGG, COG) using alignment tools (BLAST, DIAMOND) to understand biochemical pathways and cellular processes. Pathway reconstruction involves databases like MetaCyc and annotation tools such as eggNOG-mapper.

4.8 Diversity Assessment and Statistical Analysis: Alpha and beta diversity calculations quantify microbial diversity within and between samples using specialized packages (QIIME, R vegan). Statistical analysis identifies significant community differences across different conditions.

4.9 Data Visualization and Interpretation: Results presentation employs various graphical representations including heatmaps, taxonomic plots, and ordination analyses (PCA, PCoA) to illustrate community structure and diversity patterns. Network analysis visualizations reveal inter-species relationships and functional connections.

5. Computational Challenges in Metagenomic Analysis

5.1 Data Volume and Complexity: Metagenomic studies generate massive datasets ranging from gigabytes to terabytes per sample, requiring substantial computational infrastructure for

storage and analysis. The heterogeneous nature of microbial communities complicates sequence assembly and functional annotation processes.

5.2 Assembly Complexity: Metagenomic sequence assembly faces unique challenges due to mixed genomic material from multiple organisms, potentially resulting in chimeric assemblies and requiring advanced computational approaches. High polymorphism levels can cause divergent reads from identical genomic regions to be treated as separate loci.

5.3 Technological Considerations: Short-read sequencing technologies struggle with repetitive genomic regions and elevated error rates, while long-read platforms offer improved assembly continuity despite higher error frequencies. Specialized metagenomic assemblers have been developed to address these challenges.

6. Specialized Metagenomic Assembly Tools

6.1 MetaSPAdes Framework: This comprehensive toolkit handles uneven coverage and strain variations through multi-stage assembly approaches, providing high-quality results with adequate computational resources.

6.2 MEGAHIT: An efficient assembler optimized for large-scale metagenomic datasets, utilizing succinct de Bruijn graph algorithms with minimal memory requirements and high processing speed.

6.3 IDBA-UD System: Specifically designed for Illumina metagenomic reads, employing iterative approaches to enhance assembly quality and handle complex datasets.

7. Gene Prediction in Metagenomic Context

Metagenomic gene prediction addresses unique challenges including fragmented contigs, incomplete genes, and diverse genetic codes across multiple organisms:

- **Prodigal:** Fast, accurate prokaryotic gene prediction handling partial genes
- **MetaGeneMark:** Self-training algorithm adapting to sequence GC content
- **FragGeneScan:** Hidden Markov model approach for fragmented genes
- **MetaEuk:** Profile-based eukaryotic gene prediction
- **EukRep:** Separation of eukaryotic from prokaryotic sequences
- **VirFinder/VirSorter:** Specialized viral sequence identification

8. Advanced Annotation Methodologies

Modern metagenomic annotation faces challenges including data volume, low sequence similarity to reference databases, and potential annotation errors. Several innovative approaches address these limitations:

8.1.**MGS-Fast System**: Employs Bowtie2 for high-stringency DNA alignment with Galaxy workflow integration

8.2.**MetaStorm Platform**: Online server supporting custom reference databases with dual annotation pipelines

8.3.**MetaLAFFA Pipeline**: Snakemake-based workflow for comprehensive functional annotation

8.4.**FragGeneScan Method**: Combines error models with codon usage patterns for improved prediction

8.5.**MetaAnnotator Tool**: Focuses on exact k-mer matching with probabilistic taxonomic models

9. Taxonomic Binning approaches

Taxonomic binning involves assigning sequence fragments to taxonomic categories based on various characteristics including sequence similarity, compositional features, and coverage patterns. This process enables genome reconstruction and functional annotation of novel microbial species. For this purpose, different approaches are utilized. Reference-Dependent Methods utilize existing genomic databases for sequence classification, though limited by database completeness. Reference-Independent Methods are based on intrinsic sequence properties such as k-mer distributions and compositional signatures. Integrated Approaches combine multiple methodologies to enhance classification accuracy and efficiency. Different binning tools are mentioned below:

- **MGmapper**: Reference-based assignment with post-processing optimization
- **CH-Bin**: Convex hull distance-based clustering of high-dimensional feature vectors
- **TWARIT**: Combines alignment and composition-based approaches
- **MetaCoAG**: Dynamic bin adjustment using assembly graph information
- **MetaID**: Alignment-free n-gram approach for strain-level identification

10. Future Perspectives

The continuous evolution of metagenomic technologies promises enhanced understanding of microbial ecology and function. Emerging computational tools improve taxonomic binning precision and efficiency, facilitating deeper insights into microbial community dynamics and their ecological roles. The integration of advanced sequencing technologies with sophisticated analytical frameworks will continue to expand our comprehension of microbial diversity and its implications for environmental science, human health, and biotechnological applications.

Key Terminologies in Metagenomics

- Metagenomics: The study of genetic material recovered directly from environmental samples, bypassing the need for culturing organisms in the lab.
- Microbiome: Collective genomes of the microorganisms in a particular environment.
- Functional Metagenomics: Focuses on identifying and analyzing the functional capabilities of microbial communities.
- Metaproteomics: Study of all proteins expressed by a community of organisms in a complex sample at a single point in time.
- 16S rRNA Gene: A molecular marker widely used for classifying bacteria and archaea in metagenomic samples.
- Next Generation Sequencing (NGS): Advanced sequencing technologies that allow for the rapid sequencing of large amounts of DNA, crucial for metagenomic studies.
- Whole Genome Shotgun (WGS) Sequencing: A method where DNA is randomly fragmented and sequenced to reconstruct the entire genome.
- Gene Prediction: The process of identifying the locations of coding regions in genomic sequences.

Declaration of generative AI and AI-assisted technologies in the writing process

During the preparation of this manuscript, authors used SCOPUS AI and CLAUDE for certain content and citations. After using these tools/services, the author reviewed and edited the content as needed.

Selected References:

- Behera, B.K., Dehury, B., Rout, A.K. et al. Metagenomics study in aquatic resource management: Recent trends, applied methodologies and future needs. Gene Reports, 2021
- Delitte, M., Caulier, S., Bragard, C., Desoignies, N. Plant Microbiota Beyond Farming Practices: A Review. Frontiers in Sustainable Food Systems, 2021

- Dong, X., Strous, M. An Integrated Pipeline for Annotation and Visualization of Metagenomic Contigs. *Frontiers in Genetics*, 2019
- Novinscak, A., Zboralski, A., Roquigny, R., Fillion, M. Microbiome Genomics and Functional Traits for Agricultural Sustainability. *The Plant Microbiome in Sustainable Agriculture*, 2021
- Reddy, R.M., Mohammed, M.H., Mande, S.S. TWARIT: An extremely rapid and efficient approach for phylogenetic classification of metagenomic sequences. *Gene*, 2012
- Rho, M., Tang, H., Ye, Y. FragGeneScan: Predicting genes in short and error-prone reads. *Nucleic Acids Research*, 2010
- Roumpeka, D.D., Wallace, R.J., Escalettes, F. et al. A review of bioinformatics tools for bio-prospecting from metagenomic sequence data. *Frontiers in Genetics*, 2017
- Singh, B., Mal, G., Kumar, M. et al. Metagenomics in community and veterinary epidemiology. *Recent Trends and Advances in Environmental Health*, 2019
- Singh, S.P., Verma, N., Kumar, D., Gupta, S. Computational Challenges in Metagenomic Data Analysis. *Genomic Intelligence: Metagenomics and Artificial Intelligence*, 2024
- Snipen, L., Angell, I.-L., Rognes, T., Rudi, K. Reduced metagenome sequencing for strain-resolution taxonomic profiles. *Microbiome*, 2021
- Tessler, M., Neumann, J.S., Afshinnikoo, E., et al. Large-scale differences in microbial biodiversity discovery between 16S amplicon and shotgun sequencing. *Scientific Reports*, 2017
- Thatoi, H., Pradhan, S.K., Kumar, U. Applications of Metagenomics: Agriculture, Environment, and Health. *Applications of Metagenomics: Agriculture, Environment, and Health*, 2024
- Thomas, A.M., Segata, N. Multiple levels of the unknown in microbiome research. *BMC Biology*, 2019
- Yu, K., Zhang, T. Construction of Customized Sub-Databases from NCBI-nr Database for Rapid Annotation of Huge Metagenomic Datasets Using a Combined BLAST and MEGAN Approach. *PLoS ONE*, 2013

Meta-QTL Analysis and BioMercator 4.2.3 Workflow

Anupam Singh

Faculty of Agricultural Sciences, Shree Guru Gobind Singh Tricentenary University,
122505, Gurugram, Haryana, India
Email: anupambiotech@gmail.com

Meta-QTL Analysis

Introduction

Quantitative Trait Locus (QTL) mapping identifies genomic regions associated with complex traits, but individual studies suffer from limited resolution and reproducibility due to population-specific effects, environmental interactions, and statistical noise. Meta-QTL (MQTL) analysis overcomes these limitations by integrating multiple independent QTL studies into a unified statistical framework. This approach refines QTL positions, distinguishes stable genomic regions across environments, and enhances candidate gene discovery for marker-assisted breeding. The core principle involves projecting QTLs from diverse genetic maps onto a high-density consensus map using shared molecular markers. This harmonization allows cross-study comparisons and detects overlapping QTL regions. Statistical models (e.g., fixed- or random-effects) then estimate the optimal number of "true" MQTLs per chromosome. The Akaike (AIC) or Bayesian (BIC) information criterion evaluates model fit, penalizing overfitting to identify robust MQTLs.

Key Advantages:

1. Increased precision: Confidence intervals (CIs) narrow by 30–60% compared to individual studies (Goffinet & Gerber, 2000).
2. Biological validation: MQTLs with support from >5 underlying QTLs are likelier to represent causal loci (Veyrieras et al., 2007).
3. Breeding relevance: Stable MQTLs guide pyramiding of alleles for complex traits like drought tolerance.

Challenges involve data heterogeneity requiring standardized marker naming and map curation and biological variability (e.g., epistasis). Nevertheless, MQTL analysis remains indispensable for translating fragmented QTL data into actionable genetic insights.

Workflow of BioMercator 4.2.3

BioMercator 4.2.3 (Sosnowski et al., 2012) is a Windows-based tool automating MQTL analysis. Its workflow comprises six stages:

Install

BioMercator is a Java program; all you need is Java (v1.5 or above) installed on your machine. To install java see <http://www.java.com/> unzip the BioMercator archive file in a directory. On Windows: Double-click on the BioMercatorV4.jar to launch the program

On other OS: Open a terminal and execute the command line:

```
“java -jar BioMercatorV4.jar”
```

1. Data Preparation

Input Requirements:

- Genetic maps: Text files (per chromosome) with columns: `LinkageGroup, MarkerName, Position(cM)`.

```
Organism Genus=Zea
Organism Species=mays
Parent=b73
Parent=mo17
crossType=F2 intercross
popSize=242
mappingCrossType=SF2
mappingFunction=Haldane
mapName=IBM
mapUnit=cM
mapExpansion =0
mapQuality=3
locusLocation=0
chr=chr1
lg=linkage_group 1
1      phi097  34.9
2      cdo1081a      16.7
3      umc1977 12.8    EST
4      psb201b 23.7
5      umc157  46.6
6      psb115c 11.9    EST
7      bnl1007      23.4    EST
8      umc76   23.5
```

*Fig. 1: BioMercatorV3
example map file*

QTLs

```
mapName=IBM
QTL1 Trait 2 ID:0 P1 Y1 chr1 linkage_group 1 3 10 360.47 351.47 369.47
QTL2 Trait3 ID:0 P1 Y1 chr1 linkage_group 1 3 22 127.84 118.84 136.84
QTL3 Trait4 ID:0 P1 Y1 chr1 linkage_group 1 3 4 169.7 160.7 178.7
QTL4 Trait4 ID:0 P1 Y1 chr1 linkage_group 1 3 14 527.75 518.75 536.75
QTL5 Trait5 ID:0 P1 Y1 chr1 linkage_group 1 3 4 402.8 393.8 411.8
QTL6 Trait6 ID:0 P1 Y1 chr1 linkage_group 1 3 7 169.7 160.7 178.7
QTL7 Trait6 ID:0 P1 Y1 chr1 linkage_group 1 3 6 616.5 607.5 625.5
QTL8 Trait 7 ID:0 P1 Y1 chr1 linkage_group 1 3 11 323.59 314.59 332.59
QTL9 Trait 7 ID:0 P1 Y1 chr1 linkage_group 1 3 9 155.95 146.95 164.95
QTL10 Trait8 ID:0 P1 Y1 chr1 linkage_group 1 3 14 508.5 499.5 517.5
QTL11 Trait 9 ID:0 P1 Y1 chr1 linkage_group 1 3 8 638.5 629.5 647.5
QTL12 Trait10 ID:0 P1 Y1 chr1 linkage_group 1 15.39 7 345 335.3 356.8
QTL13 Trait13 ID:0 P1 Y1 chr1 linkage_group 1 4.6 5.6 241.53 233.45 248.85
QTL14 Trait14 ID:0 P1 Y1 chr1 linkage_group 1 3 15 625.5 565.6 685.4
QTL15 Trait14 ID:0 P1 Y1 chr1 linkage_group 1 3 15 59.08 51.6 67.2
```

Fig. 2: BioMercatorV3 example QTL file

QTL data: CSV files listing `Trait, LinkageGroup, PeakPosition, CI_Left, CI_Right, LOD_Score`.

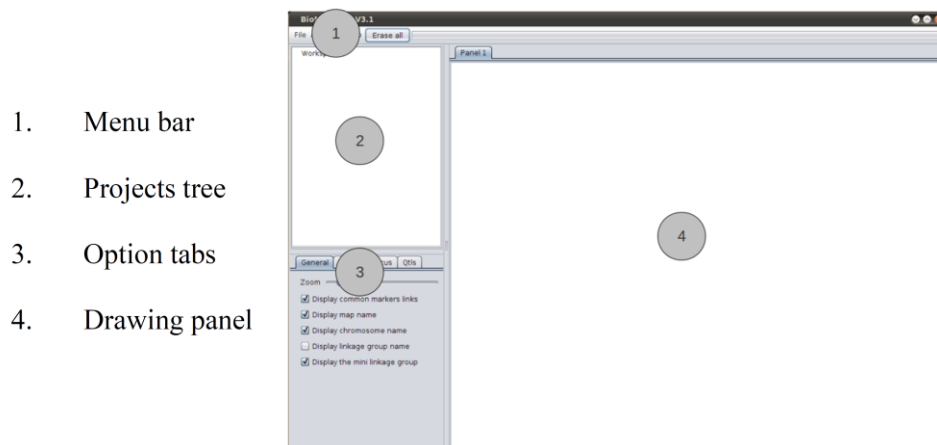
```
mapname=IBM
poptype=F2 intercross
chromosome ch1
>map
=====
Map:
Markers      Distance
1  isu110b    3.2 cM
2  phi097     0.0 cM
3  tub1       0.0 cM
4  bnlgl1124  1.2 cM
5  tub4b     38.2 cM
6  rz444d     9.0 cM
7  psb201b   11.8 cM
8  umc1977    0.9 cM
9  umc1071    2.9 cM
10 bnlgl1014  5.9 cM
11 umc1041    9.7 cM
12 cdo1081a  28.7 cM
```

Fig. 3: BioMercator V2 example map file

Preprocessing:

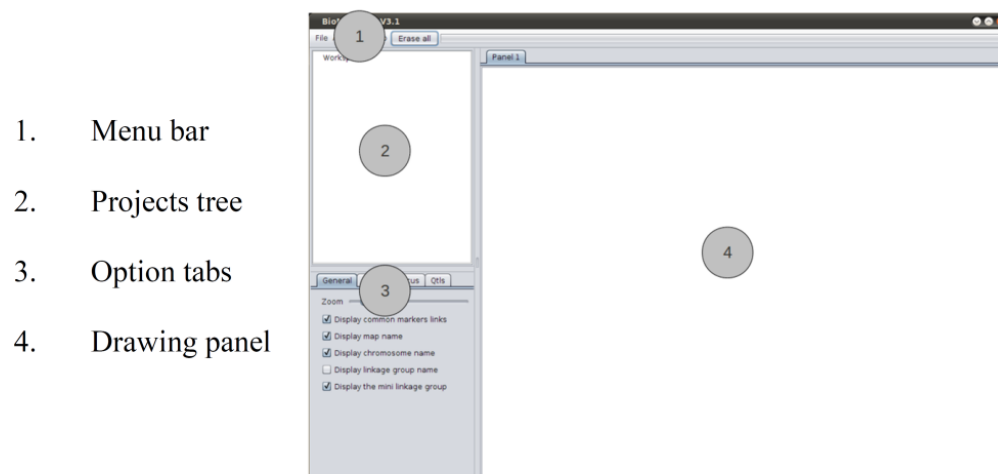
- Standardize marker names (e.g., "TaSNP1A_123" → "SNP123").
- Exclude maps with <10 markers/chromosome to ensure integration quality.

Graphical User Interface.

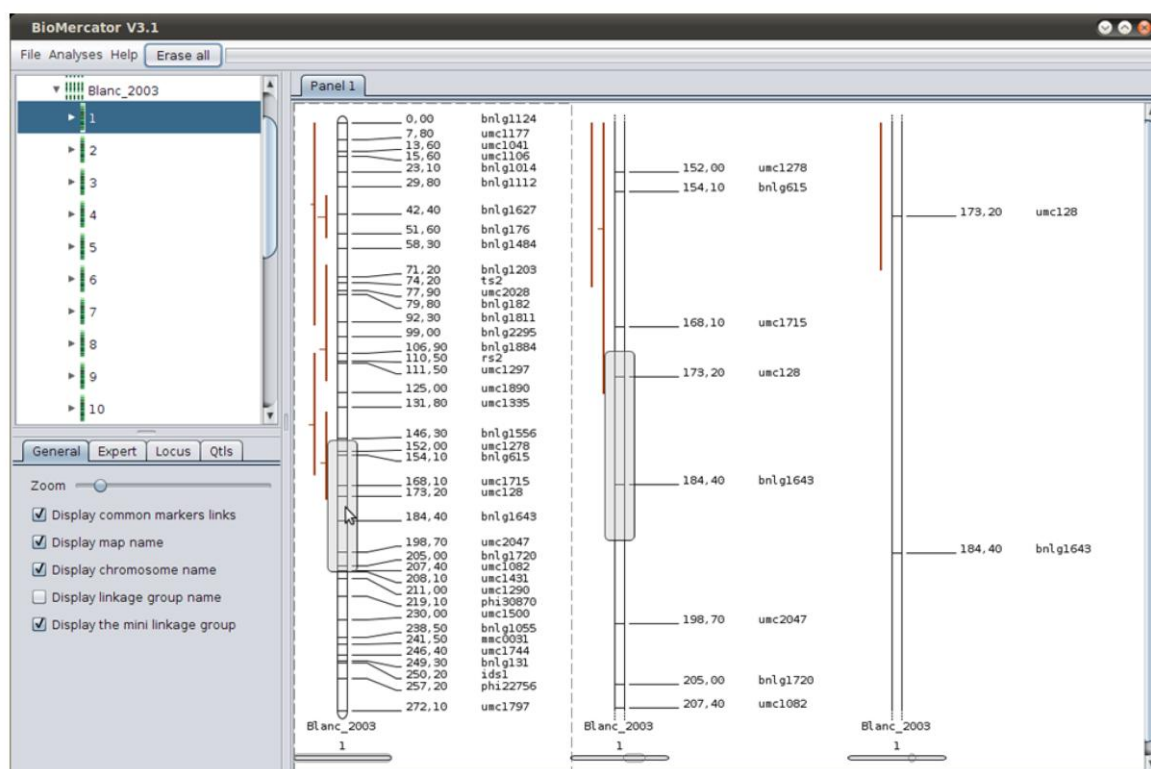
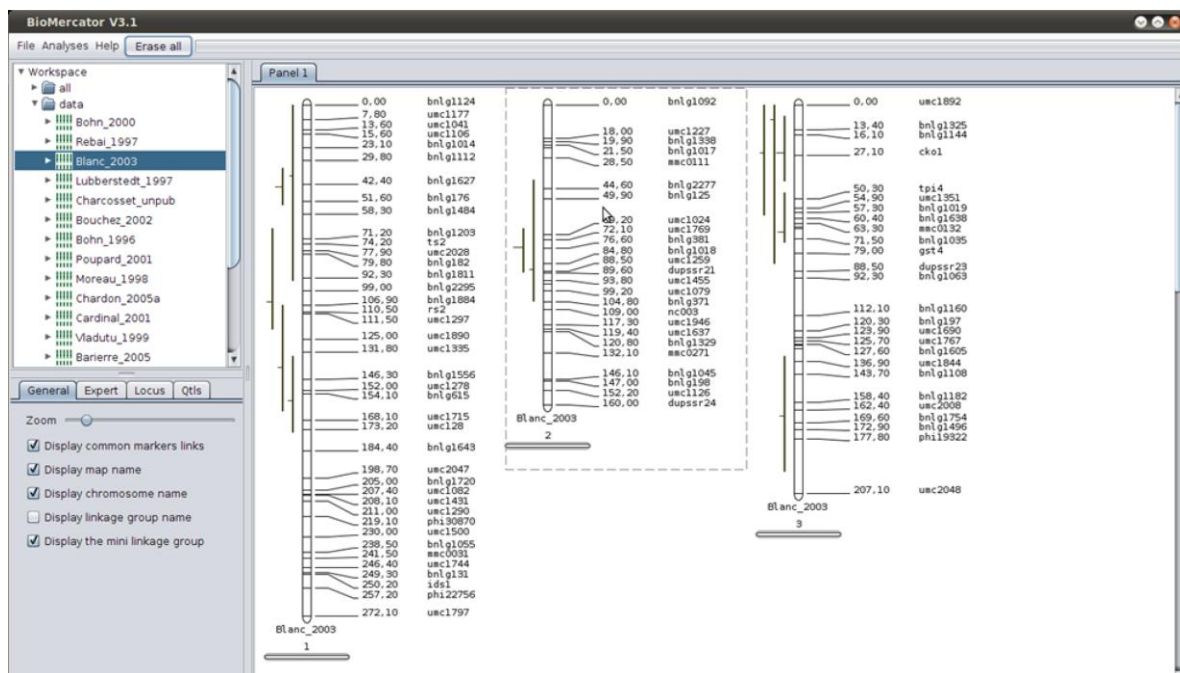


2. Project Setup

Graphical User Interface.



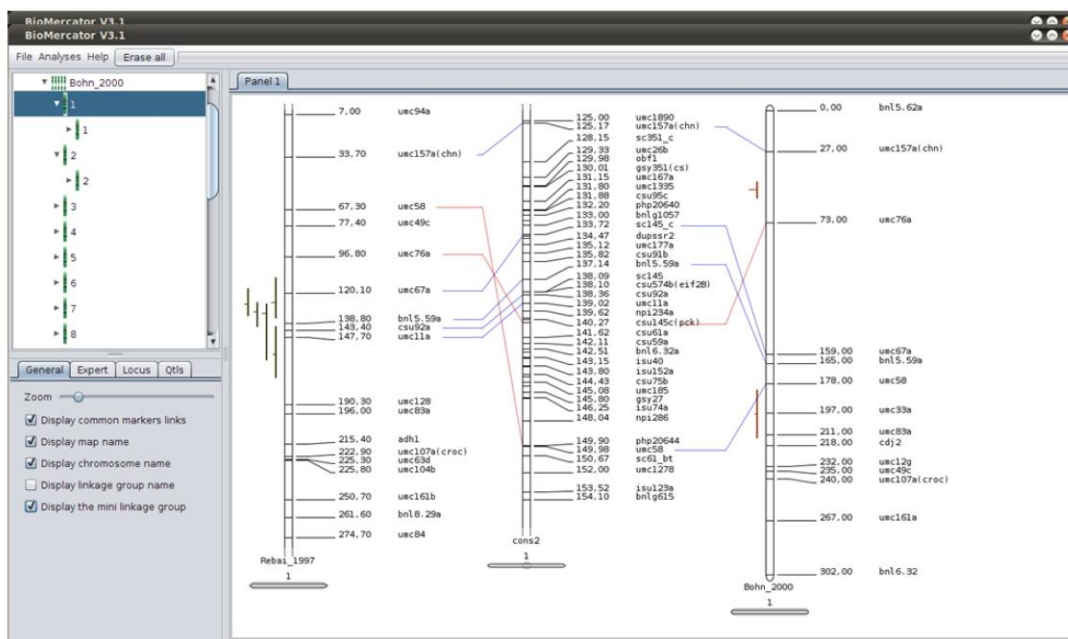
- Launch BioMercator → File → New Project → Define species (e.g., *Triticum aestivum*).
- Set map unit to centiMorgans (cM).



3. Map Integration

- Import maps: Maps → Import Map → Load individual linkage maps.
- Build consensus map:
- Tools → Consensus Map → Select maps for integration.

- Parameters:
 - Weighting: By marker number (favors data-rich maps).
 - Algorithm: Iterative re-weighting (default).
- Execute → Save output (e.g., "Consensus_Ch5B").



Dynamic comparisons

- Quality control:
 - Stress value <5% indicates minimal marker order distortion.
 - Resolve conflicts using physical map coordinates if available.
- 4. QTL Projection
 - Import QTLs: QTLs → Import QTLs → Load study-specific files.
 - Project QTLs:
 - QTLs → Project QTLs → Target "Consensus_Ch5B".

- Algorithm: Linear interpolation between flanking markers shared between source and consensus maps (Arcade et al., 2004).

- Output: Adjusted peak positions and CIs on the consensus map.

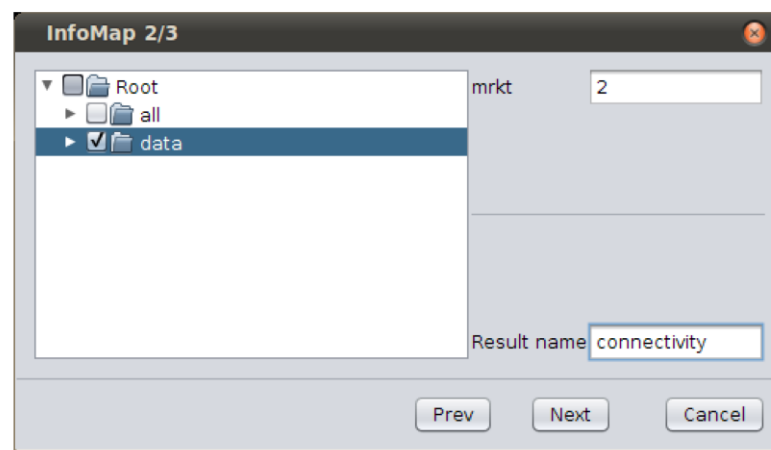
5. Meta-QTL Analysis

- Select chromosome and trait (e.g., "RootDepth_Ch5B").

- Tools → Meta-QTL Analysis → Configure:

Output:

- *_cmp.txt: lists the markers connection chromosome by chromosome
- *_mrk.txt: lists the number of common markers chromosome by chromosome



InfoMap analysis

- Model: Fixed-effects (assumes homogeneous genetic effects) or random-effects (accounts for between-study heterogeneity).

- Selection criterion: BIC (preferred for large datasets to avoid overfitting).

- Model search: Evaluates 1 to k MQTLs per chromosome.

- Execute:

- BioMercator identifies the optimal MQTL number minimizing BIC.

- Output includes:

- MQTL positions and 95% CIs.

- Goodness-of-fit statistics (R^2).

- List of contributing QTLs.

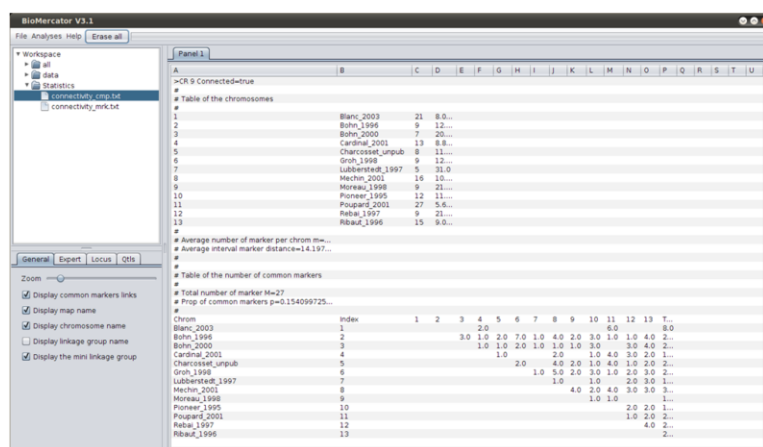
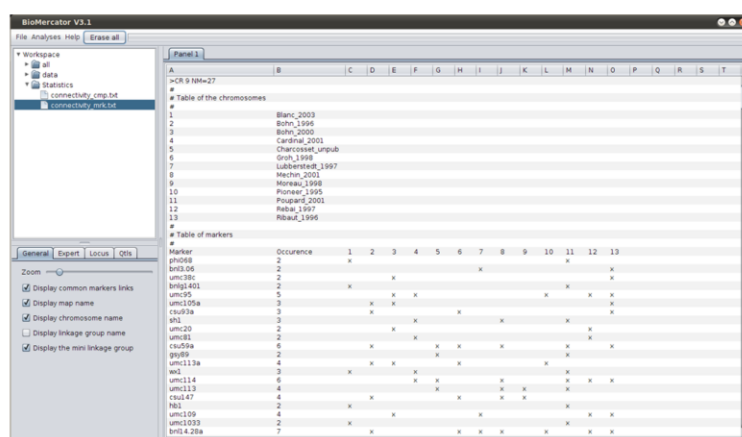


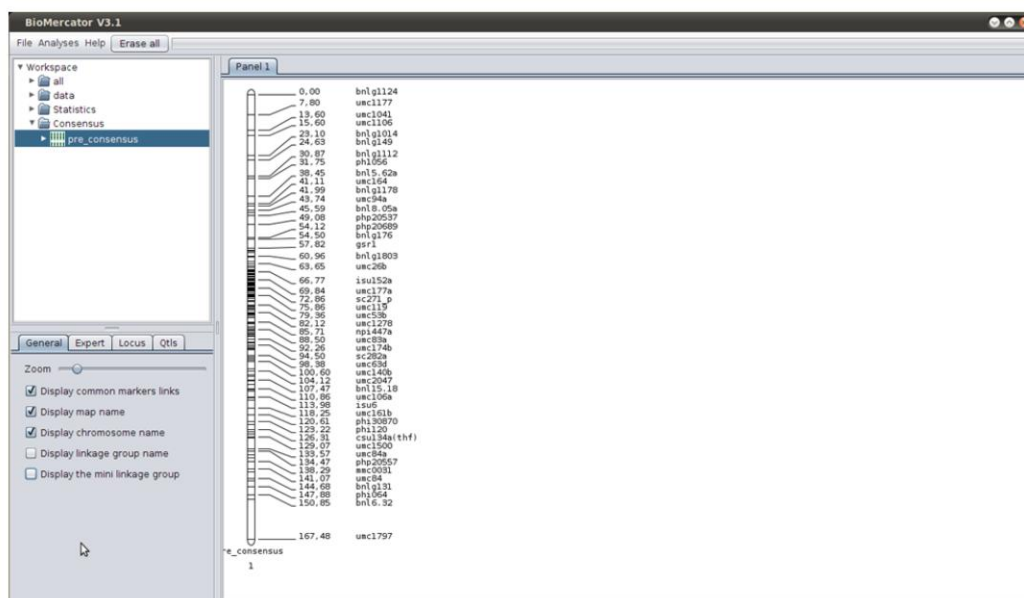
Fig. 16: InfoMap analysis - result



InfoMap analysis

6. Visualization & Validation

- Graphical output:
- View → Map Viewer displays:
- Consensus map (horizontal axis).



ConsMap Analysis

- Projected QTLs (color-coded bars).
- MQTLs (triangles with confidence intervals).
- Export as SVG/PDF for publications.

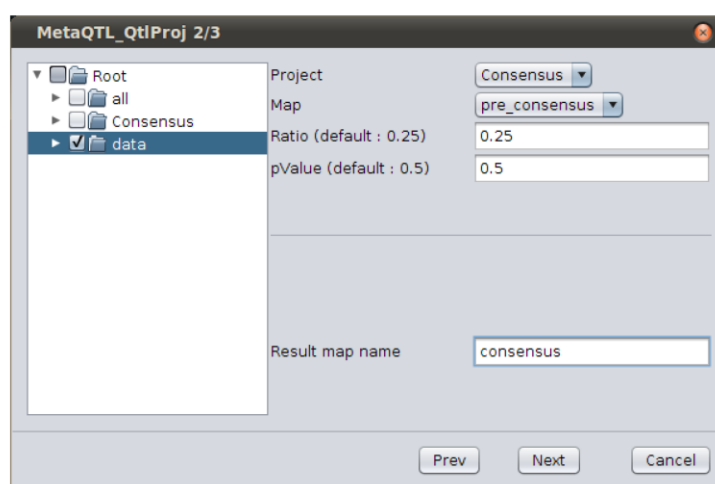
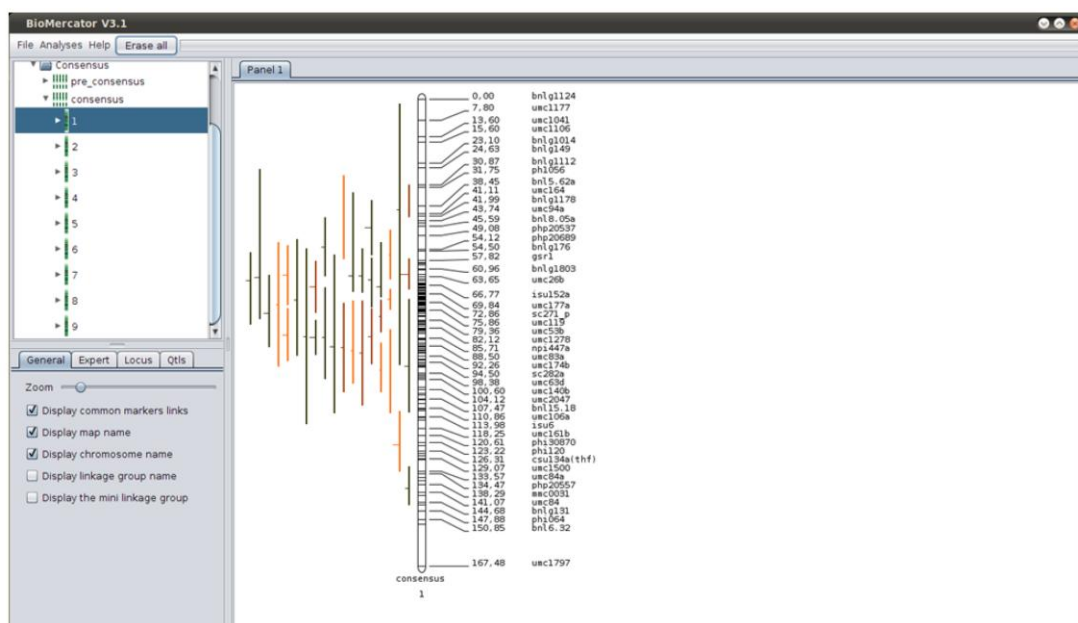


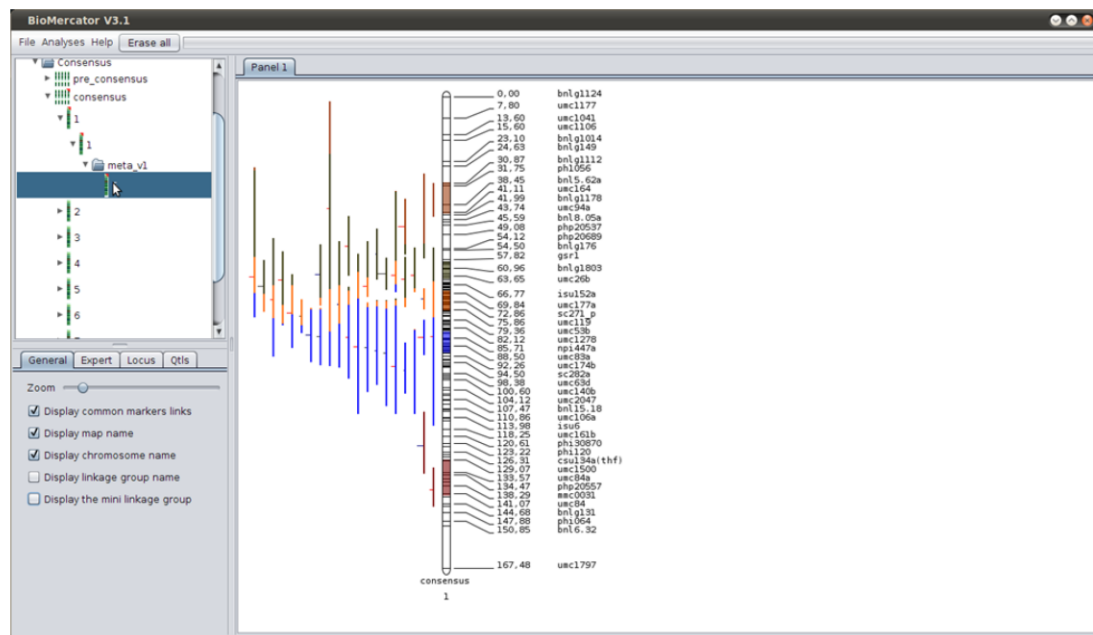
Fig. 20: QTLProj analysis



QTLProj analysis

- Biological interpretation:
- Prioritize MQTLs with:
 - Narrow CI (<5 cM).
 - High LOD support (average LOD >10).
 - Overlap with candidate genes (e.g., via EnsemblPlants).

Output:



Meta analysis - Visualisation

- Validation:

- Compare MQTLs with independent GWAS peaks or expression QTLs (eQTLs).
- Example: A wheat MQTL for grain yield (Chr3A) co-localized with TaGS5 gene validated by CRISPR (Liu et al., 2022).

Troubleshooting Common Issues

- QTL projection failure:

- Cause: Flanking markers absent in consensus map.
- Fix: Add missing markers or exclude the QTL.

- High stress in consensus map:

- Cause: Marker order conflicts between maps.
- Fix: Re-order markers using recombination data or exclude problematic maps.

- Overfitting in MQTL model:

- Cause: Too many MQTLs selected by AIC.
- Fix: Switch to BIC criterion.

References

1. Goffinet, B. & Gerber, S. (2000). Quantitative trait loci: a meta-analysis. *Genetics*, 155(1), 463–473.
2. Veyrieras, J. B. et al. (2007). High-resolution QTL mapping reveals epistasis and genotype× environment interactions affecting maize kernel composition. *Genetics*, 175(3), 1447–1460.
3. Arcade, A. et al. (2004). BioMercator: integrating genetic maps and QTL towards discovery of candidate genes. *Bioinformatics*, 20(14), 2324–2326.
4. Sosnowski, O. et al. (2012). BioMercator V3: an upgrade of genetic map compilation and QTL meta-analysis algorithms. *BMC Bioinformatics*, 13(1), 335.
5. Liu, Y. et al. (2022). A meta-QTL analysis highlights genomic hotspots for drought tolerance in wheat. *The Plant Genome*, 15(1), e20185.
6. Meta-QTL Theory: Chardon, F. et al. (2004). Genetic architecture of flowering time in maize. *Genetics*, 168(4), 2169–2185.

Software Access: BioMercator 4.2.3 is freely available at [Moulon INRAE](<http://moulon.inrae.fr/biomercator/>). Always consult the official manual for runtime parameters and dataset limitations.



Twenty-One Day Online Training Program on



Advanced Statistical and Machine Learning Techniques for Data Analysis Using Open-Source Software for Abiotic Stress Management in Agriculture

16 July – 5 August 2025

Chief Patron

Dr. K Sammi Reddy, Director, ICAR-NIASM

Patron

Dr. Nitin P Kurade, Head, SSSPS, ICAR-NIASM



Course Directors



Dr. Santosha Rathod
Dr. Nobin Chandra Paul
Ms. Ponnaganti Navyasree
Mr. K Ravi Kumar



Organised by:

School of Social Science and Policy Support

ICAR-National Institute of Abiotic Stress Management Baramati, Maharashtra - 413115

About ICAR-NIASM

ICAR-NIASM is the premier institute of ICAR established in 2009 at Baramati. The institute aims at exploring the avenues for the management of abiotic stresses affecting the very sustainability of national food production systems. Besides focusing on developing climate resilient solutions through cutting-edge technologies for managing abiotic stresses, NIASM also aims to enhance scientific capacity through multidisciplinary research and capacity building programs.

About the Training Program

This 21-day online training program offers hands-on experience in advanced statistical, machine learning, and deep learning techniques for analyzing agricultural data. The training is not limited to abiotic stress management; it is applicable across all research disciplines where data analysis plays a critical role. Participants will work with large datasets using open-source tools such as R, Python, QGIS, VassarStats, and BlueSky Statistics through practical, application-oriented sessions.

Key Objectives

The training program aims to:

- ✦ Train the participants in multivariate statistics, AI- ML, and deep learning, agroecological modeling tools, remote sensing &GIS
- ✦ Provide hands-on experience with open-source software
- ✦ Enable independent application of these techniques in research work

Course Content

The program combines theoretical foundations with hands-on practical sessions, enabling participants to apply these techniques to their own datasets efficiently.

Module 1: Software Tools for Data Analysis

- ✦ Pre-training session on Installation guide to R/Python/other tools
- ✦ Introduction to R
- ✦ Introduction to Python
- ✦ Introduction to Bluesky Statistics & VassarStats
- ✦ Data Visualization in R
- ✦ R Shiny and R packages

Module 2: Regression & Multivariate Statistical Methods

- ✦ Regression Analysis
- ✦ Regression for Categorical Data
- ✦ Nonlinear Growth Models
- ✦ Regularization Techniques in Regression Models
- ✦ Panel Data Regression
- ✦ Non-Parametric Analysis
- ✦ Data Classificatory Techniques (CA, DA)
- ✦ Data Reduction Techniques (FA, PCA)

Module 3: Design of Experiments & Statistical Genetics

- ✦ Analysis of Complete and Incomplete Block Designs
- ✦ Analysis of Incomplete Block Designs
- ✦ Analysis of Groups of Experiments (GOE)
- ✦ Response Surface Methodology
- ✦ Generation Mean Analysis
- ✦ Mating Designs
- ✦ Path Analysis
- ✦ Stability Analysis
- ✦ QTL Analysis
- ✦ Transcriptomic Analysis
- ✦ Genome Wide Association Studies (GWAS)
- ✦ Genomic Selection
- ✦ Selection Index
- ✦ Meta-QTL Analysis
- ✦ Meta-Genomics

Module 4: Machine Learning & Deep Learning Techniques

- ✦ Introduction to Machine Learning
- ✦ k-nearest neighbor (KNN)
- ✦ Artificial Neural Network
- ✦ Support Vector Machine
- ✦ CART and Decision Tree
- ✦ Random Forest Regression
- ✦ Extreme Learning Machine
- ✦ XGBoost
- ✦ Deep Learning: RNN, GRU, CNN, LSTM, Transformer DL
- ✦ ML Optimization Techniques
- ✦ Yield Forecasting using AI

Module 5: Time Series & Forecasting Methods

- ✦ Trend Analysis
- ✦ Time Series Analysis
- ✦ ARCH Family of Models
- ✦ Bayesian Forecasting Models
- ✦ Count Time Series Models
- ✦ Spatiotemporal Time Series Modelling
- ✦ Hybrid Modelling
- ✦ Ensemble Modelling
- ✦ VAR and Cointegration Analysis

Module 6: Spatial & Environmental Data Analysis

- ✦ Introduction to RS & GIS
- ✦ Introduction to QGIS
- ✦ Introduction to Google Earth Engine
- ✦ Spatial Interpolation Techniques
- ✦ Introduction to Sampling & Spatial Sampling Strategy
- ✦ Application of ML in RS & GIS (ASIS portal)
- ✦ Application of UAVs in agricultural data modelling

Module 7: Agro-Ecological Modelling

- ✦ Biomass Modelling & Carbon Sequestration using Allometric Models
- ✦ CMIP6 GCM Models
- ✦ Crop Simulation Modelling (DSSAT and APSIM)
- ✦ High-throughput Plant Phenotyping
- ✦ Assessment of Extreme Weathers

Module 8: Emerging & Interdisciplinary Topics

- ✦ Importance of Data Science in Agricultural Research
- ✦ Meta Analysis
- ✦ AHP and Grey Model: Technology Forecasting
- ✦ Markov Chain Analysis
- ✦ Social Network Analysis
- ✦ Bibliometric Analysis
- ✦ Economic Index Development
- ✦ Impact Assessment Modelling, Trend Impact Analysis
- ✦ Statistical Modelling in Disease Epidemiology
- ✦ Fuzzy Regression Analysis

Expected Learning Outcomes

- By completing this program, participants will master in advanced statistical, machine learning, and deep learning techniques to analyze complex agricultural and environmental datasets.
- They will gain hands-on experience with open-source tools (R, Python, QGIS and others) for data analysis, image processing, and designing efficient workflows to address real-world abiotic stress challenges.

Who Can Apply?

- Researchers and scientists from agriculture, climate science, environmental studies, and allied fields
- Data analysts looking for transition from classical statistics to ML/DL approaches
- Academicians and students seeking proficiency in open-source statistical and geospatial tools

Registration Fee

- ₹ 1000/- for students and research scholars
- ₹ 2000/- for scientists, researchers, faculty members, and working professionals from public organizations
- ₹ 5000/- for participants from private industries

Bank Account Details

Account Holder Name: ICAR UNIT-NIASM, Baramati

Account Number: 30862846914

Name of the Bank: State Bank of India

Branch Address: Afzalpurkar Building, Bhigwan Road,
Baramati, Maharashtra-413102

IFSC: SBIN0000321

UPI Code : icarniasmbmt@sbi

ICAR UNIT NIASM BARAMATI

SCAN & PAY



UPI ID: icarniasmbmt@sbi

Important Dates

- Last Date for Receipt of Applications: 30th June 2025
- Information to Selected Candidate: 2nd July 2025

Registration Link: <https://forms.gle/5cMmTxnS19DvWoc48>



Contact for Registration Related Queries

Dr. Santosha Rathod

Senior Scientist (Agricultural Statistics)
School of Social Science and Policy Support
ICAR-NIASM, Baramati
Mob: 9900912188

Dr. Nobin Chandra Paul

Scientist (Agricultural Statistics)
School of Social Science and Policy Support
ICAR-NIASM, Baramati
Mob: 8851954194

Ms. Navyasree Ponnaganti

Scientist (ABM)
School of Social Science and Policy Support
ICAR-NIASM, Baramati
Mob: 8639110291

Mr. K. Ravi Kumar

Scientist (Agricultural Extension)
School of Social Science and Policy Support
ICAR-NIASM, Baramati
Mob: 9133120921

Contact Email Id: ssspsniasm@gmail.com



हर कदम, हर डगर
किसानों का हमसफर
भारतीय कृषि अनुसंधान परिषद

Agrisearch with a human touch

Application Form

Twenty-One Day Online Training Program

on

Advanced Statistical and Machine Learning Techniques for Data Analysis Using Open-Source Software for Abiotic Stress Management in Agriculture

16 July to 5 August 2025

1.	Full Name (in BLOCK letters)					
2.	Highest degree with specialization					
3.	Present Institute Name					
4.	Address for Correspondence					
5.	E-mail address: <i>Telephone Number Mob/O/R:</i>					
6.	Date of Birth					
7.	Sex (Male/Female/other)					
8.	Education Qualification:					
	Degree	Subject	Year of passing	Class / Division / Equivalent	University / Institute	
	Bachelors Masters Ph.D. Any Other					
9.	Level of Knowledge in Statistics			Beginner / Intermediate / Expert		
10.	Level of Knowledge in R/ Python/ other software			Beginner/Expert		
11.	Area of ongoing research work					
13	Expectations from the training					

**Candidate must fill in all the details*

Signature of the Applicant with date

CERTIFICATE

It is certified that information furnished above is correct.

*Signature of the Recommending Authority
/ Head of the Department/ Institute along with Seal*



हर कदम, हर डगर
किसानों का हमसफर
भारतीय कृषि अनुसंधान परिषद

Agrisearch with a human touch