Introduction to Genomic Selection in R using the rrBLUP Package

Amy Jacobson
jaco0795@umn.edu
University of Minnesota

Learning Objectives

- Download the package and load the sample files
- Impute missing markers using A.mat()
- Define the training and validation populations
- Run mixed.solve() and determine accuracy of predictions

Overview of rrBLUP package

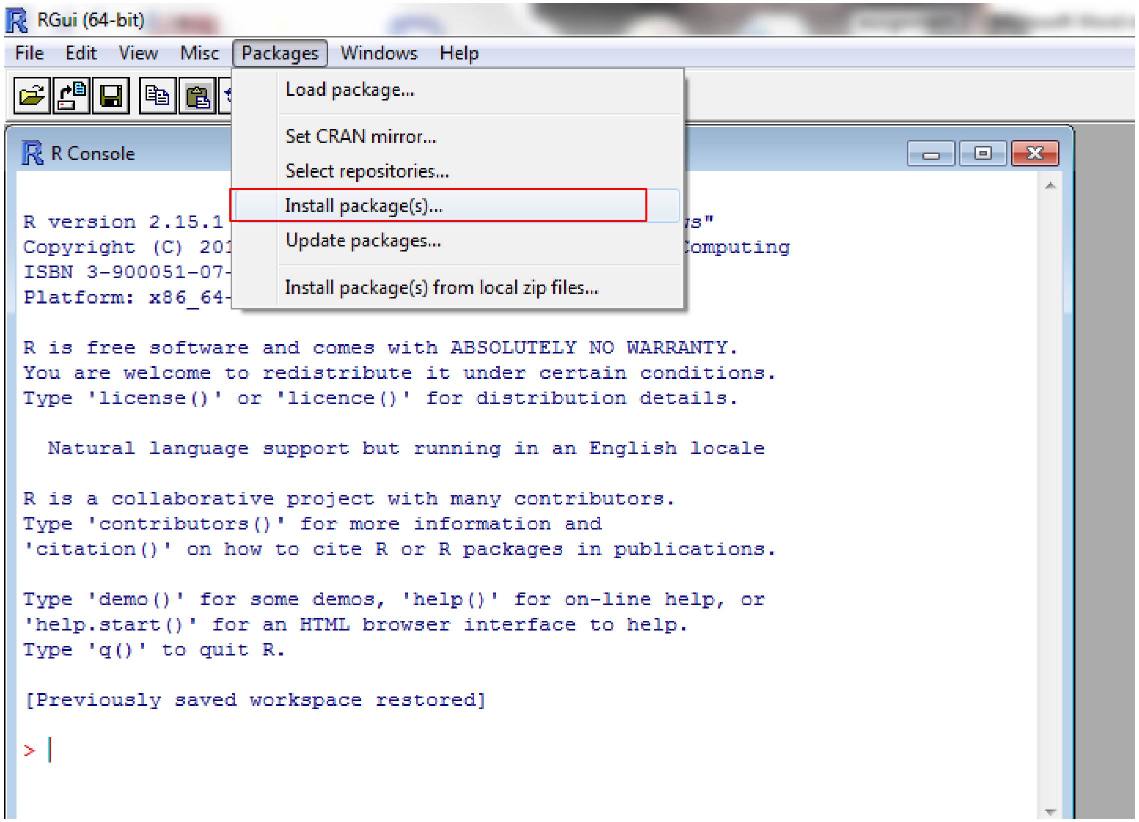
- Download from CRAN-version 4
  - Must use R version 2.14.1 or greater
- Uses ridge regression BLUP for genomic predictions
- Predicts marker effects through mixed.solve()
- A.mat() command can be used to impute missing markers
  - Mixed.sove does not allow NA marker values
- Define the training and validation populations

One Step vs. Two Step

- One step
  - Uses a mixed model analysis for the plot data
- Two step
  - Adjusted means are calculated across locations
  - Means are then used in ridge regression blup
- This webinar uses a two step approach
  - Computationally more efficient and faster

Install the rrBLUP Package

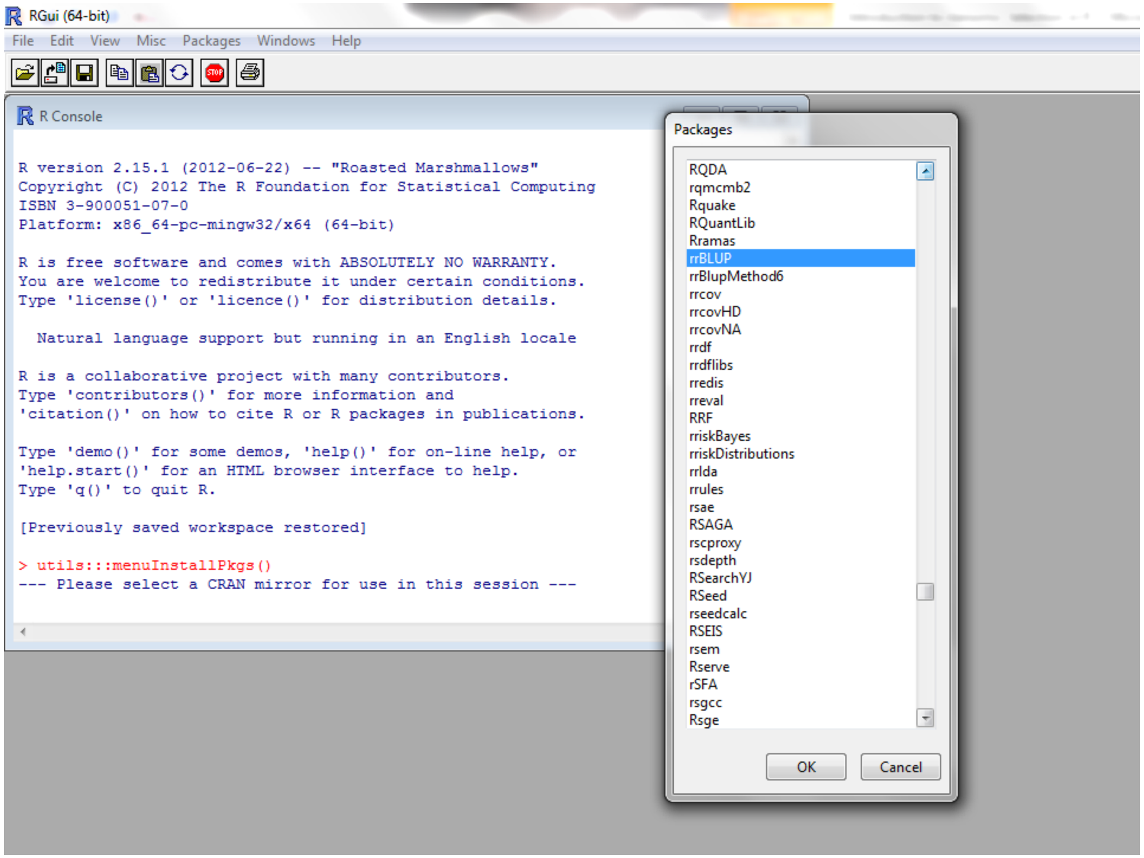
- Launch R->Packages->Install Package
- Select CRAN Mirror nearest you



```
R version 2.15.1                    s"
Copyright (C) 201                   Computing
ISBN 3-900051-07-
Platform: x86_64-

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

  Natural language support but running in an English locale

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

[Previously saved workspace restored]

> 
```

Install the rrBLUP Package

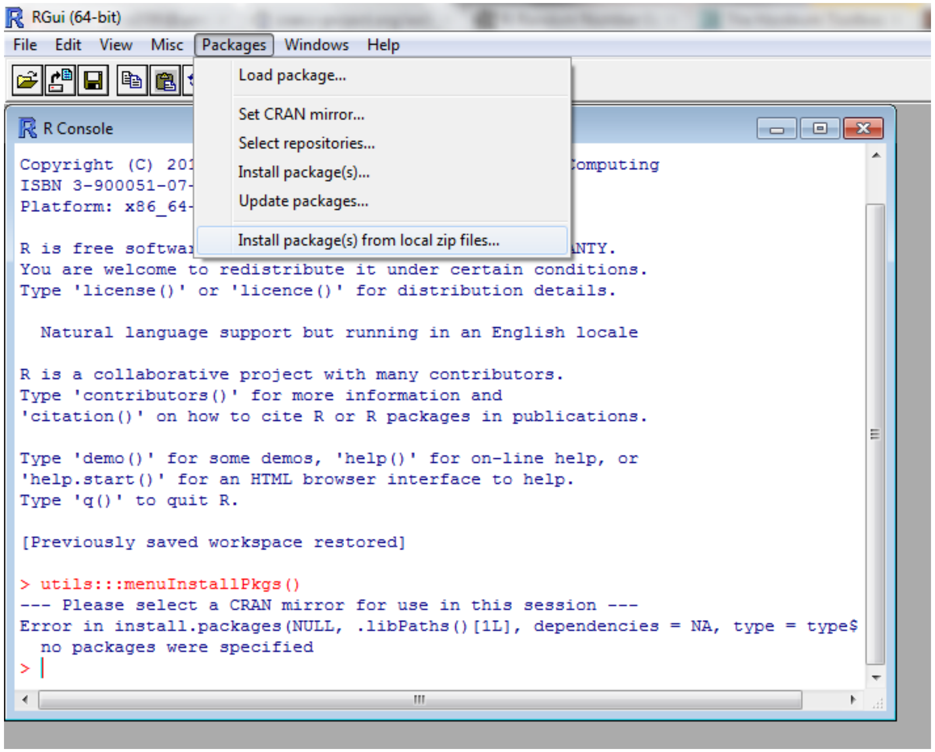
- Select the rrBLUP package



```
R version 2.15.1 (2012-06-22) -- "Roasted Marshmallows"
Copyright (C) 2012 The R Foundation for Statistical Computing
ISBN 3-900051-07-0
Platform: x86_64-pc-mingw32/x64 (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

  Natural language support but running in an English locale

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

[Previously saved workspace restored]

> utils:::menuInstallPkgs()
--- Please select a CRAN mirror for use in this session ---
```

Install the rrBLUP Package

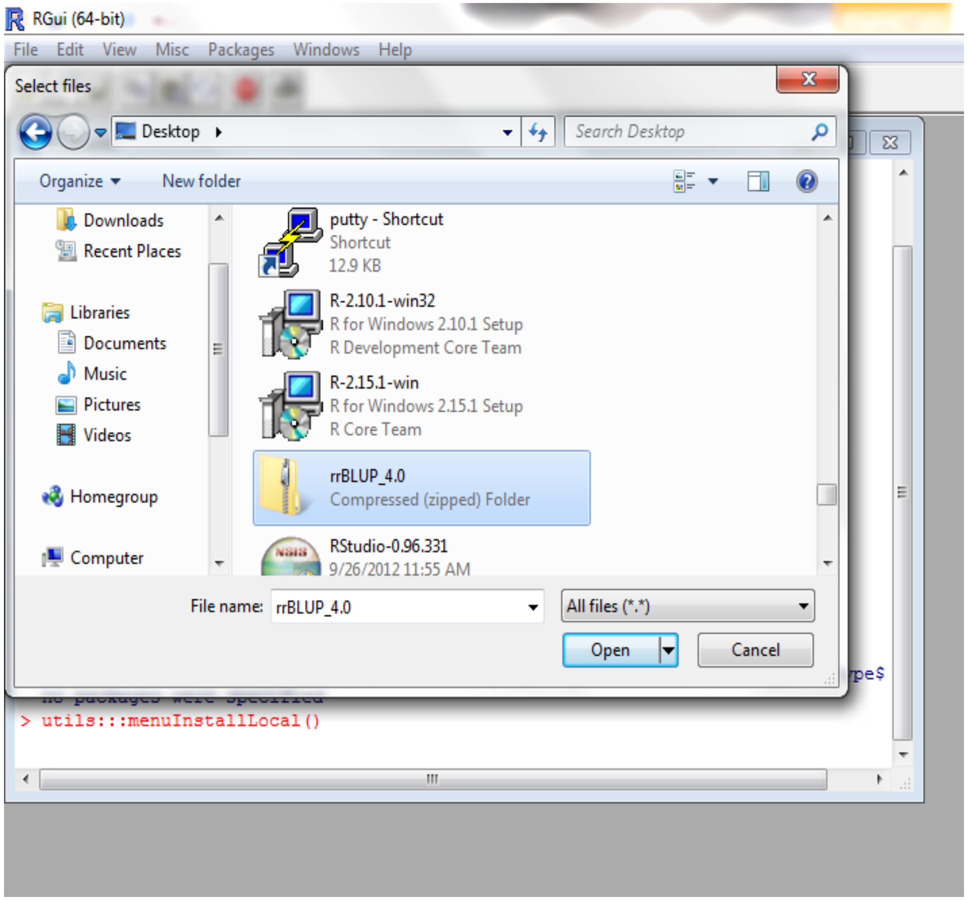
- Install the package by a zip file
- http://cran.r-project.org/web/packages/rrBLUP/index.html
- Packages->install package from local zip files



```
Copyright (C) 201                                    omputing
ISBN 3-900051-07-
Platform: x86_64-

R is free softwa                                    ANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

  Natural language support but running in an English locale

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

[Previously saved workspace restored]

> utils:::menuInstallPkgs()
--- Please select a CRAN mirror for use in this session ---
Error in install.packages(NULL, .libPaths()[1L], dependencies = NA, type = type$
  no packages were specified
> 
```

Install the rrBLUP Package

- Select the package from saved location



putty - Shortcut
Shortcut
12.9 KB

R-2.10.1-win32
R for Windows 2.10.1 Setup
R Development Core Team

R-2.15.1-win
R for Windows 2.15.1 Setup
R Core Team

rrBLUP_4.0
Compressed (zipped) Folder

RStudio-0.96.331
9/26/2012 11:55 AM

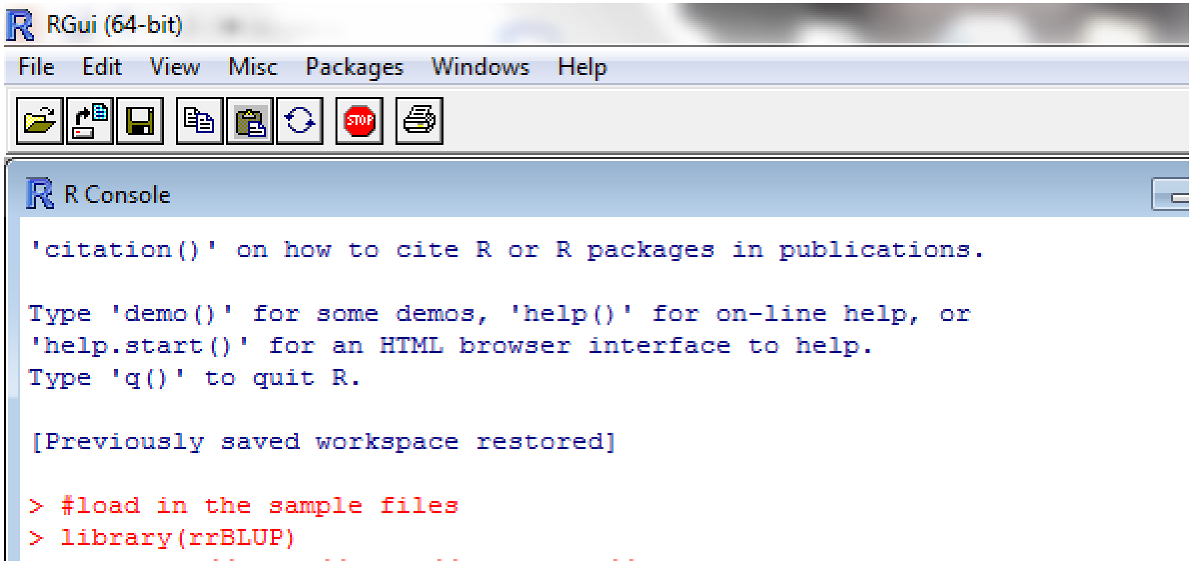
File name: rrBLUP_4.0

All files (*.*)

```
no packages were specified
> utils:::menuInstallLocal()
```

Install the rrBLUP Package

- Now that the package is installed, the library must be loaded every time R is opened



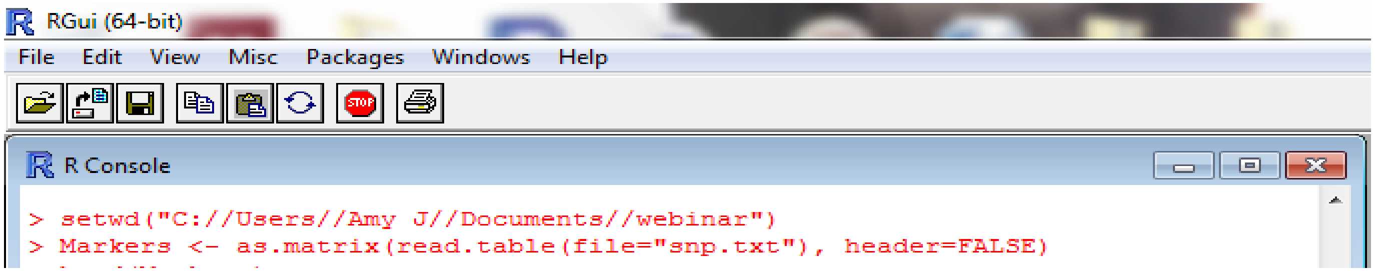
```
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

[Previously saved workspace restored]

> #load in the sample files
> library(rrBLUP)
```

Sample Files

- Files downloaded from the Hordeum Toolbox http://hordeumtoolbox.org/
- University of Minnesota barley breeding program preliminary yield trail-St. Paul location in 2009
- Phenotypic traits-yield, plant height and heading date
- 1178 markers, 164 NA markers
- 1 = homozygous for parent 1, 0 = heterozygous, and -1 homozygous for parent 2
  - Markers must be in the {-1,0,1} format for rrBLUP

Load the Sample Files

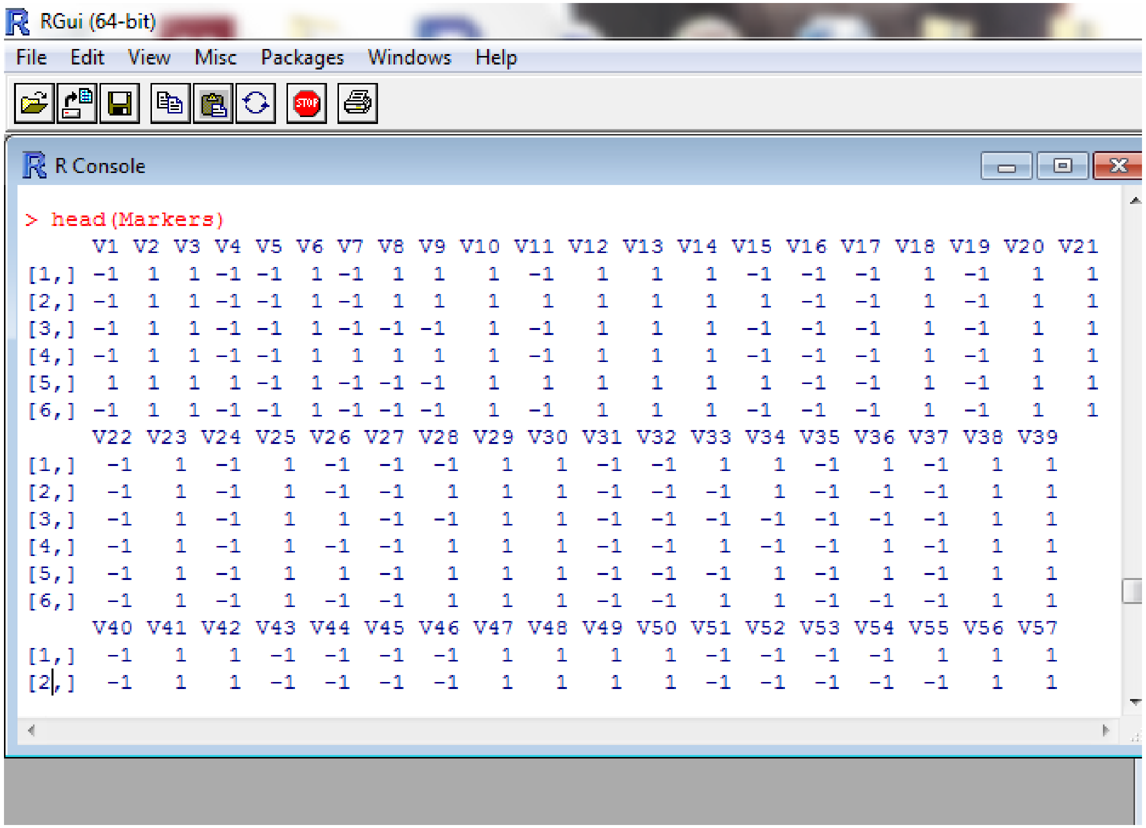
- Setwd()-Set the working directory to the location of the sample files
- Read.table command used for .txt files
- Read.csv command used for .csv files
- Header=F since sample marker file does not have a header with marker names



```
> setwd("C://Users//Amy J//Documents//webinar")
> Markers <- as.matrix(read.table(file="snp.txt"), header=FALSE)
```

Load the Sample Files

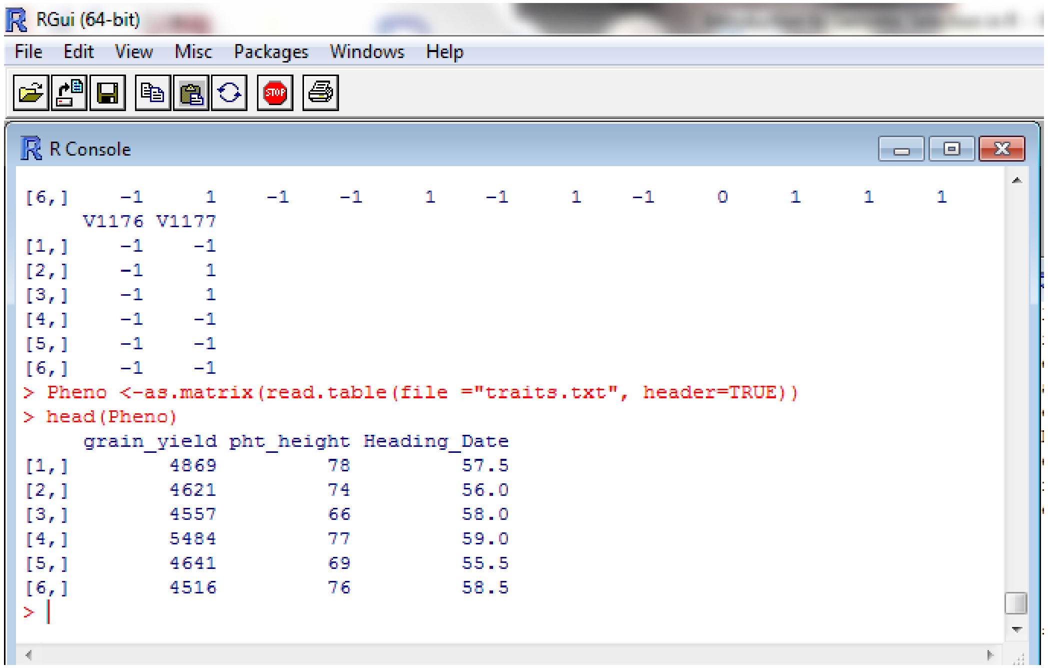
- head() command used to see the first 5 lines of a file
- Useful to see if data was loaded correctly



```
> head(Markers)
     V1 V2 V3 V4 V5 V6 V7 V8 V9 V10 V11 V12 V13 V14 V15 V16 V17 V18 V19 V20 V21
[1,] -1  1  1 -1 -1  1 -1  1  1   1  -1   1   1   1  -1  -1  -1   1  -1   1   1
[2,] -1  1  1 -1 -1  1 -1  1  1   1   1   1   1   1   1  -1  -1   1  -1   1   1
[3,] -1  1  1 -1 -1  1 -1 -1 -1   1  -1   1   1   1  -1  -1  -1   1  -1   1   1
[4,] -1  1  1 -1 -1  1  1  1  1   1  -1   1   1   1  -1  -1  -1   1  -1   1   1
[5,]  1  1  1  1 -1  1 -1 -1 -1   1   1   1   1   1   1  -1  -1   1  -1   1   1
[6,] -1  1  1 -1 -1  1 -1 -1 -1   1  -1   1   1   1  -1  -1  -1   1  -1   1   1
     V22 V23 V24 V25 V26 V27 V28 V29 V30 V31 V32 V33 V34 V35 V36 V37 V38 V39
[1,]  -1   1  -1   1  -1  -1  -1   1   1  -1  -1   1   1  -1   1  -1   1   1
[2,]  -1   1  -1   1  -1  -1   1   1   1  -1  -1  -1   1  -1  -1  -1   1   1
[3,]  -1   1  -1   1   1  -1  -1   1   1  -1  -1  -1  -1  -1  -1  -1   1   1
[4,]  -1   1  -1   1  -1  -1   1   1   1  -1  -1   1  -1  -1   1  -1   1   1
[5,]  -1   1  -1   1   1  -1   1   1   1  -1  -1  -1   1  -1   1  -1   1   1
[6,]  -1   1  -1   1  -1  -1   1   1   1  -1  -1   1   1  -1  -1  -1   1   1
     V40 V41 V42 V43 V44 V45 V46 V47 V48 V49 V50 V51 V52 V53 V54 V55 V56 V57
[1,]  -1   1   1  -1  -1  -1  -1   1   1   1   1  -1  -1  -1  -1   1   1   1
[2,]  -1   1   1  -1  -1  -1  -1   1   1   1   1  -1  -1  -1  -1  -1   1   1
```

Load the Sample Files

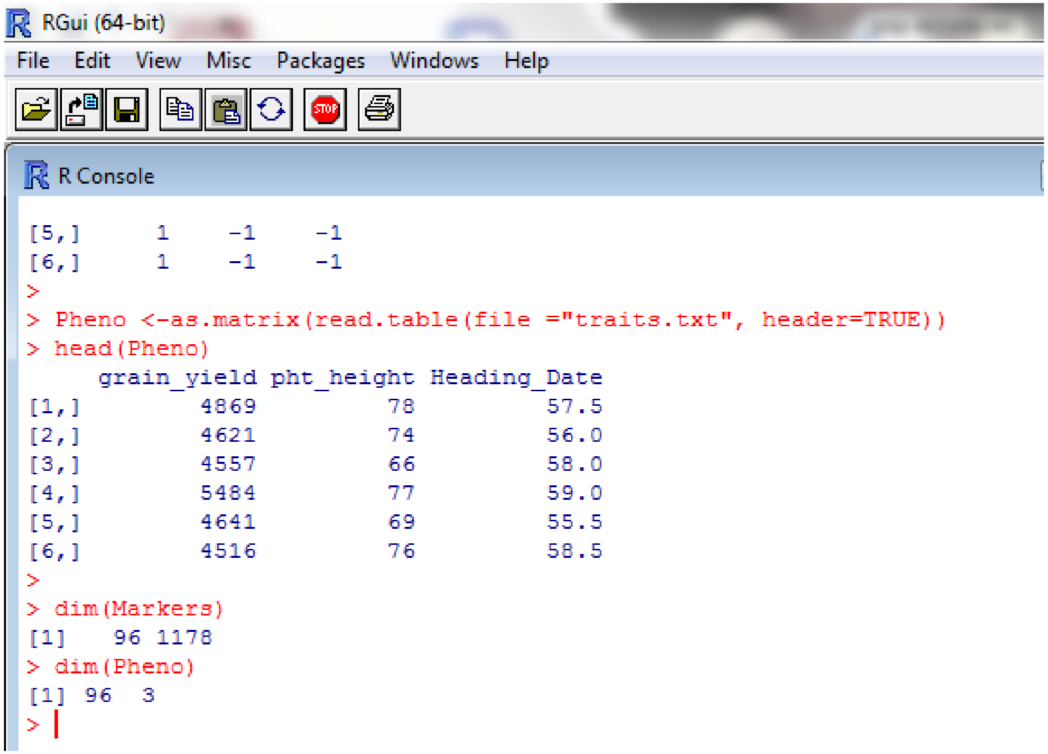
- Load the phenotype file and use the head command to see the first five lines
  - Header=T since phenotype files have column names
- Markers and phenotypes must be in matrix format



```
[6,]    -1     1    -1    -1     1    -1     1    -1     0     1     1     1
     V1176 V1177
[1,]    -1    -1
[2,]    -1     1
[3,]    -1     1
[4,]    -1    -1
[5,]    -1    -1
[6,]    -1    -1
> Pheno <-as.matrix(read.table(file ="traits.txt", header=TRUE))
> head(Pheno)
     grain_yield pht_height Heading_Date
[1,]        4869         78         57.5
[2,]        4621         74         56.0
[3,]        4557         66         58.0
[4,]        5484         77         59.0
[5,]        4641         69         55.5
[6,]        4516         76         58.5
> 
```

Load the Sample Files

- Determine the size of the matrices
- dim() command gives the number of rows and columns
- 96 observations and 1178 markers, 3 traits



```
[5,]     1    -1    -1
[6,]     1    -1    -1
>
> Pheno <-as.matrix(read.table(file ="traits.txt", header=TRUE))
> head(Pheno)
     grain_yield pht_height Heading_Date
[1,]        4869         78         57.5
[2,]        4621         74         56.0
[3,]        4557         66         58.0
[4,]        5484         77         59.0
[5,]        4641         69         55.5
[6,]        4516         76         58.5
>
> dim(Markers)
[1]   96 1178
> dim(Pheno)
[1] 96  3
> 
```

Learning Objectives

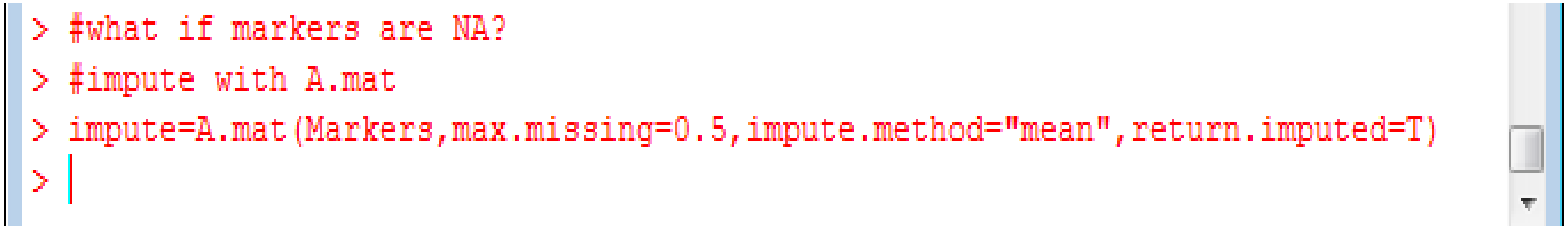
- Download the package and load the sample files
- **Impute missing markers using A.mat()**
- Define the training and validation populations
- Run mixed.solve() and determine accuracy of predictions

Impute Missing Markers

- rrBLUP mixed.solve() does not allow for missing markers
- Imputed value is the population mean for that marker
- Useful for SNP data since level of missing data is low
    - In the sample files 164 markers are missing out of 1178 (0.14%)
- A.mat also calculates the additive relationship matrix

Impute Missing Markers

- max.missing-maximum proportion of missing data
  - If 50% of markers are missing data then markers are not imputed
- impute method- imputes the mean of the markers
- return.imputed-prints out the imputed results if set to TRUE



```
> #what if markers are NA?
> #impute with A.mat
> impute=A.mat(Markers,max.missing=0.5,impute.method="mean",return.imputed=T)
> 
```

Impute Missing Markers

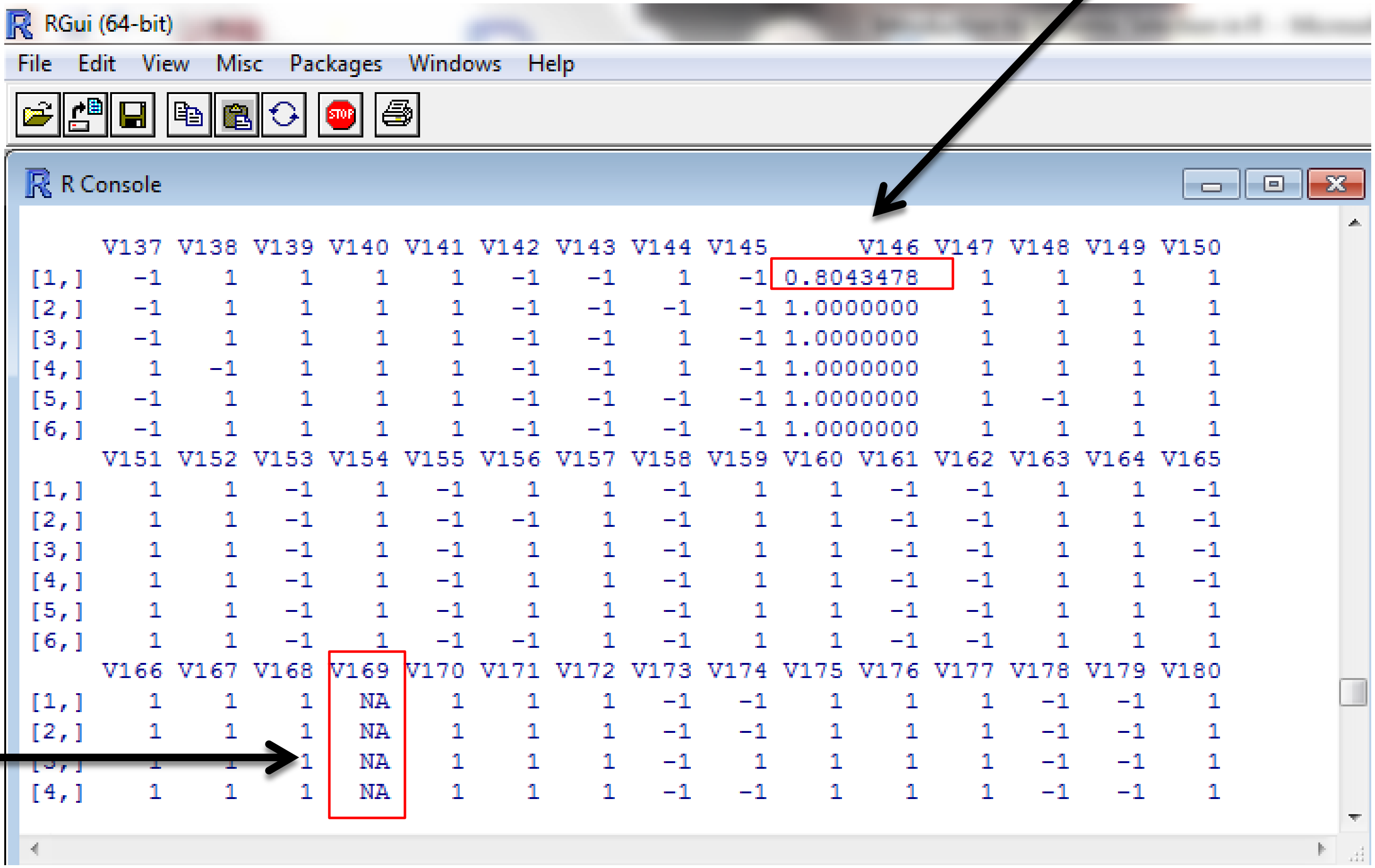
-
  ```
  >impute=A.mat(Markers,max.missing=0.5,impute.method="mean",return.imputed=T)
  ```

-
  ```
  > Markers_impute=impute$imputed
  ```
  - Rename imputed marker matrix as Markers_impute
- impute$imputed-returns the imputed marker matrix
- impute$A-returns the additive relationship matrix

Impute Missing Markers

- >impute$imputed

Imputed marker value

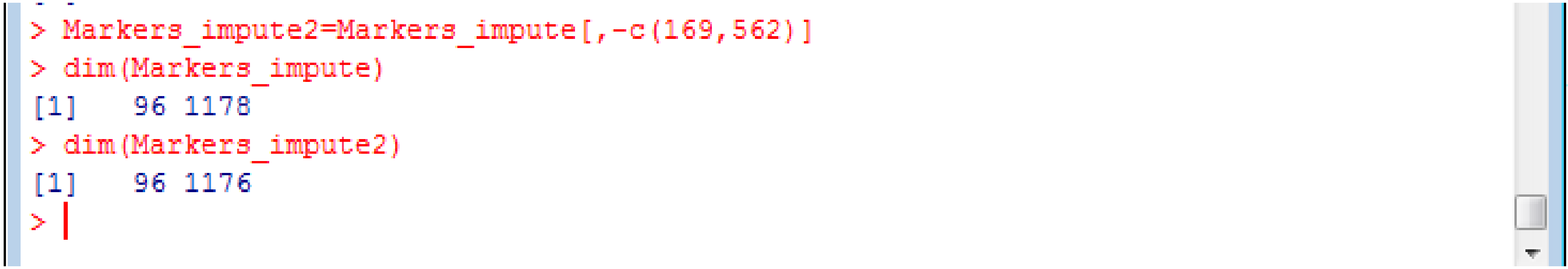


```
     V137 V138 V139 V140 V141 V142 V143 V144 V145      V146 V147 V148 V149 V150
[1,]   -1    1    1    1    1   -1   -1    1   -1 0.8043478    1    1    1    1
[2,]   -1    1    1    1    1   -1   -1   -1   -1 1.0000000    1    1    1    1
[3,]   -1    1    1    1    1   -1   -1    1   -1 1.0000000    1    1    1    1
[4,]    1   -1    1    1    1   -1   -1    1   -1 1.0000000    1    1    1    1
[5,]   -1    1    1    1    1   -1   -1   -1   -1 1.0000000    1   -1    1    1
[6,]   -1    1    1    1    1   -1   -1   -1   -1 1.0000000    1    1    1    1
     V151 V152 V153 V154 V155 V156 V157 V158 V159 V160 V161 V162 V163 V164 V165
[1,]    1    1   -1    1   -1    1    1   -1    1    1   -1   -1    1    1   -1
[2,]    1    1   -1    1   -1   -1    1   -1    1    1   -1   -1    1    1   -1
[3,]    1    1   -1    1   -1    1    1   -1    1    1   -1   -1    1    1   -1
[4,]    1    1   -1    1   -1    1    1   -1    1    1   -1   -1    1    1   -1
[5,]    1    1   -1    1   -1    1    1   -1    1    1   -1   -1    1    1    1
[6,]    1    1   -1    1   -1   -1    1   -1    1    1   -1   -1    1    1    1
     V166 V167 V168 V169 V170 V171 V172 V173 V174 V175 V176 V177 V178 V179 V180
[1,]    1    1    1   NA    1    1    1   -1   -1    1    1    1   -1   -1    1
[2,]    1    1    1   NA    1    1    1   -1   -1    1    1    1   -1   -1    1
[3,]    1    1    1   NA    1    1    1   -1    1    1    1    1   -1   -1    1
[4,]    1    1    1   NA    1    1    1   -1   -1    1    1    1   -1   -1    1
```

Marker value left NA if more than 50% missing data

Impute Missing Markers

- Remove markers that had more than 50% missing data
  - NA values are not allowed in mixed.solve
  - Two markers in the SNP file must be removed
  - Column 169 and 562
  - New dimensions show 2 less columns
- Use Markers_impute2 as marker matrix for estimating marker effects



```
> Markers_impute2=Markers_impute[,-c(169,562)]
> dim(Markers_impute)
[1]   96 1178
> dim(Markers_impute2)
[1]   96 1176
> 
```

Learning Objectives

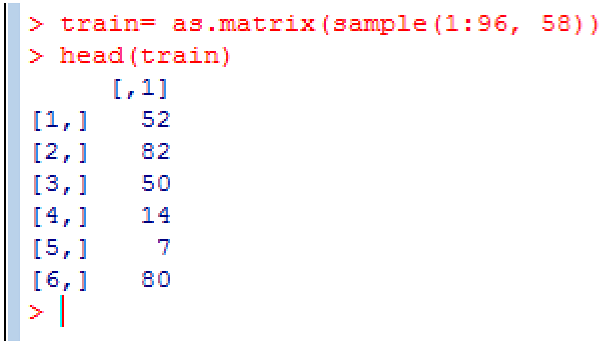
- Download the package and load the sample files
- Impute missing markers using A.mat()
- **Define the training and validation populations**
- Run mixed.solve() and determine accuracy of predictions

Training and Validation Populations

- Training population-genotyped and phenotyped
- Validation population-phenotype values estimated based on marker effects calculated from training population
- Code is set that 60% of the total population is the training population
    - 40% validation population

Training and Validation Populations

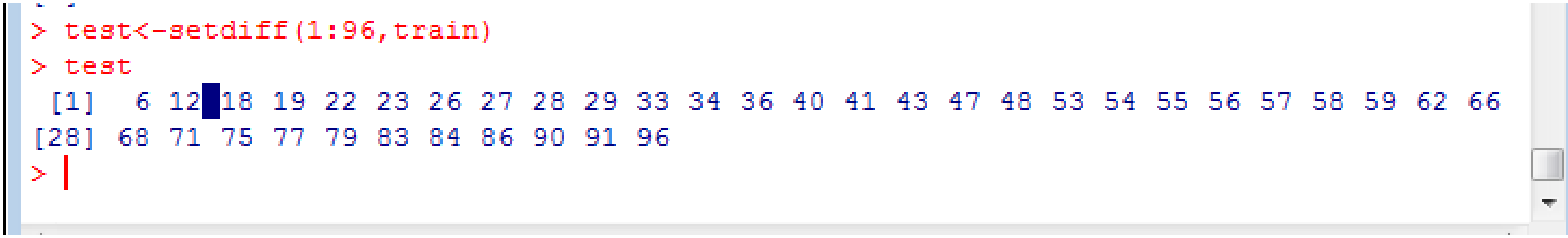
- 58 (60% of total population of 96) random numbers sampled to determine which individuals are in the training population
- Individuals are the row numbers for the phenotypes and marker matrices
- Sampled numbers will be different every time the code is run and will affect the correlation accuracy



```
> train= as.matrix(sample(1:96, 58))
> head(train)
     [,1]
[1,]   52
[2,]   82
[3,]   50
[4,]   14
[5,]    7
[6,]   80
> 
```

Training and Validation Populations

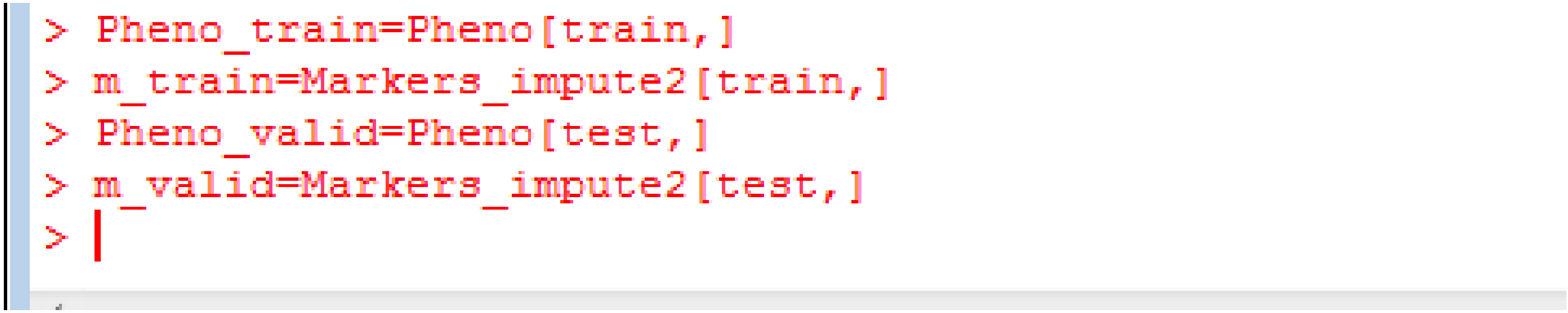
- Validation population is 40% of the total population
- setdiff() command determines the numbers that are not in the training population and will be part of the validation population



```
> test<-setdiff(1:96,train)
> test
 [1]  6 12 18 19 22 23 26 27 28 29 33 34 36 40 41 43 47 48 53 54 55 56 57 58 59 62 66
[28] 68 71 75 77 79 83 84 86 90 91 96
> 
```

Training and Validation Populations

- Pheno_train and m_train are the phenotype and marker matrices for the values in the training population
- Pheno_valid and m_valid will be the validation populations

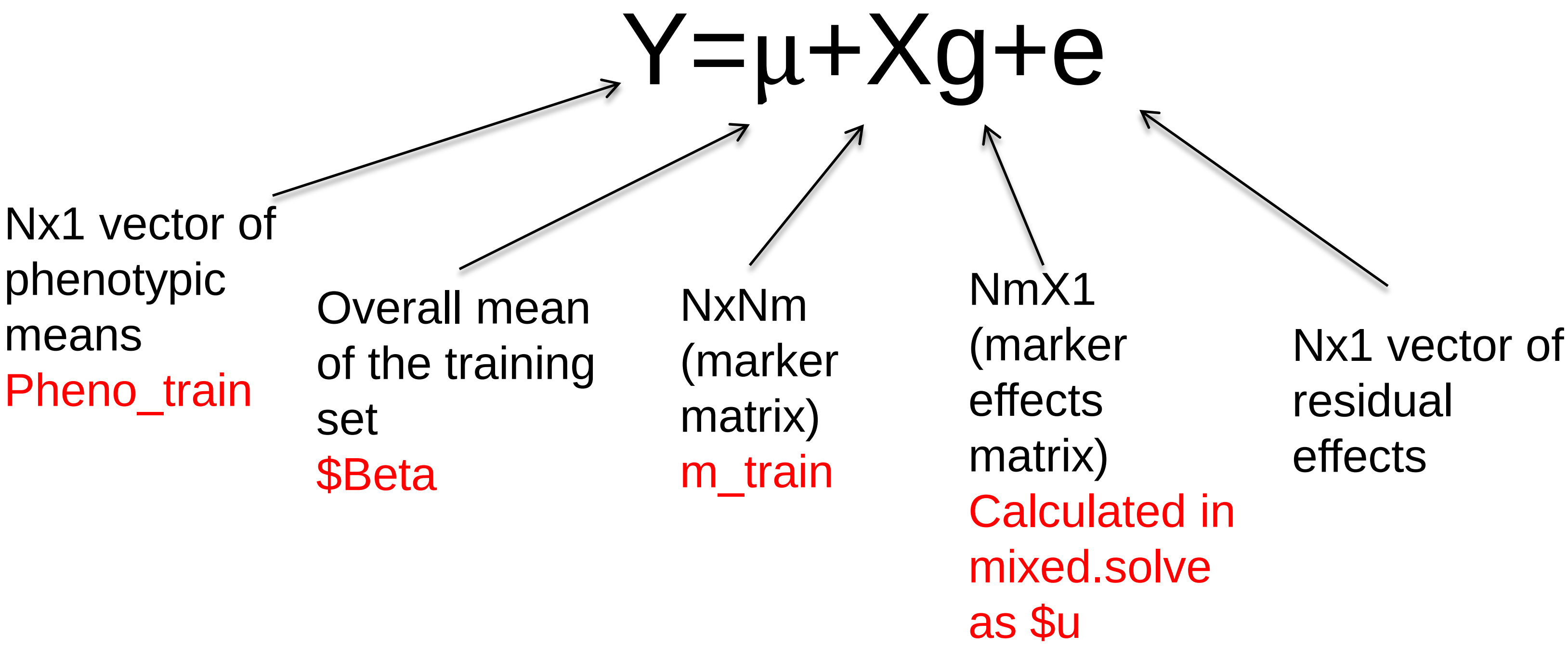


```
> Pheno_train=Pheno[train,]
> m_train=Markers_impute2[train,]
> Pheno_valid=Pheno[test,]
> m_valid=Markers_impute2[test,]
> 
```

Learning Objectives

- Download the package and load the sample files
- Impute missing markers using A.mat()
- Define the training and validation populations
- **Run mixed.solve() and determine accuracy of predictions**

Run mixed.solve



$$Y=\mu+Xg+e$$

Nx1 vector of phenotypic means
Pheno_train

Overall mean of the training set
$Beta

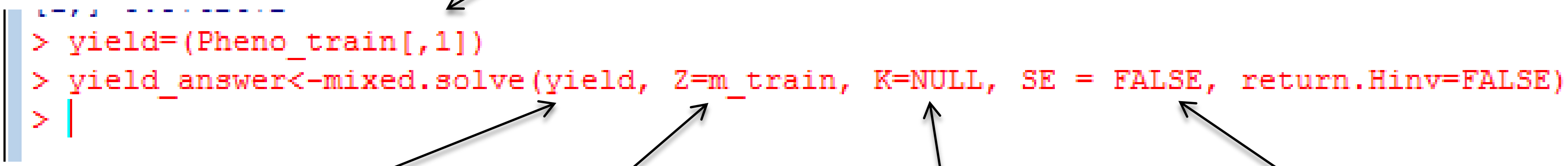
NxNm (marker matrix)
m_train

NmX1 (marker effects matrix)
Calculated in mixed.solve as $u

Nx1 vector of residual effects

Run mixed.solve

Yield is the first column of the pheno_train matrix



```
> yield=(Pheno_train[,1])
> yield_answer<-mixed.solve(yield, Z=m_train, K=NULL, SE = FALSE, return.Hinv=FALSE)
> 
```

Vector of observations

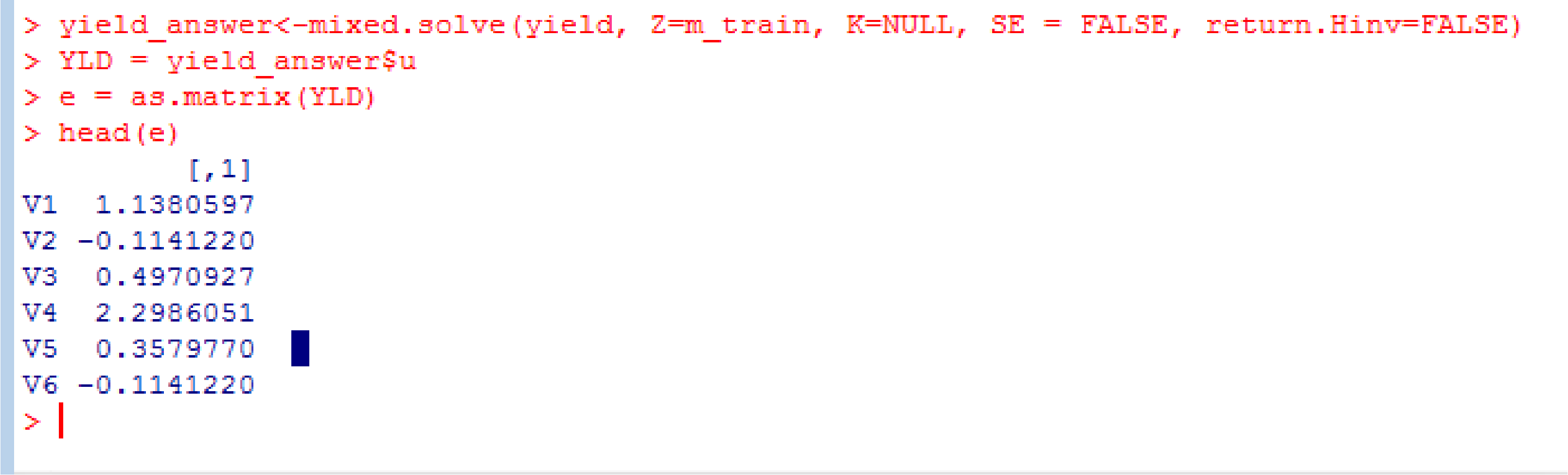
Design matrix of random effects (Markers)

K matrix is the identity matrix

Standard errors are not calculated

Run mixed.solve

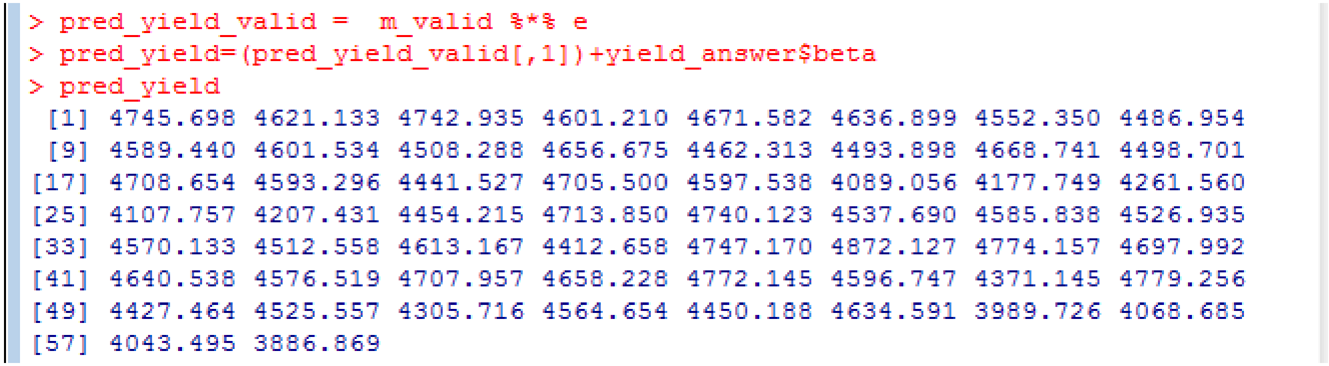
- Yield_answer$u is the output of the marker effects
- head(e) shows the marker effects for the first five markers



```
> yield_answer<-mixed.solve(yield, Z=m_train, K=NULL, SE = FALSE, return.Hinv=FALSE)
> YLD = yield_answer$u
> e = as.matrix(YLD)
> head(e)
          [,1]
V1  1.1380597
V2 -0.1141220
V3  0.4970927
V4  2.2986051
V5  0.3579770
V6 -0.1141220
> 
```

Run mixed.solve

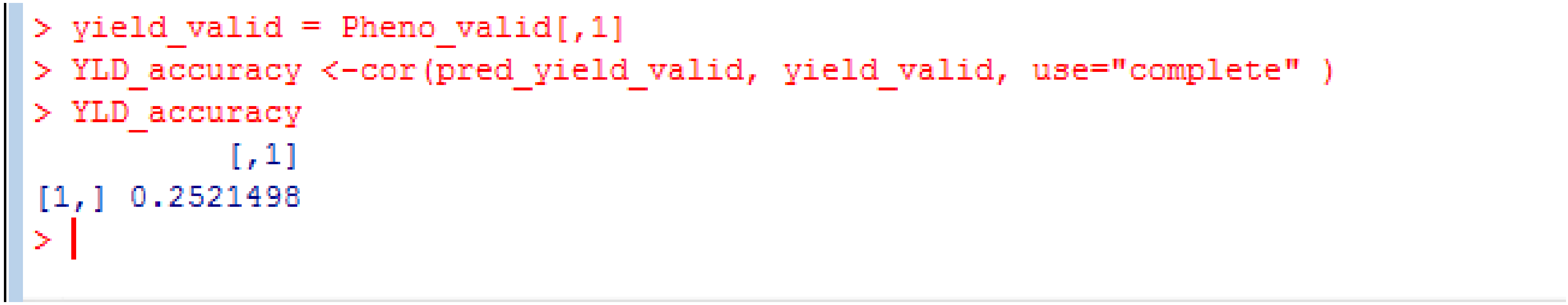
- m_valid*e = marker validation matrix times the marker effects
- Pred_yield=predicted yield based on the marker effects of the training population with the grand mean added in



```
> pred_yield_valid =  m_valid %*% e
> pred_yield=(pred_yield_valid[,1])+yield_answer$beta
> pred_yield
 [1] 4745.698 4621.133 4742.935 4601.210 4671.582 4636.899 4552.350 4486.954
 [9] 4589.440 4601.534 4508.288 4656.675 4462.313 4493.898 4668.741 4498.701
[17] 4708.654 4593.296 4441.527 4705.500 4597.538 4089.056 4177.749 4261.560
[25] 4107.757 4207.431 4454.215 4713.850 4740.123 4537.690 4585.838 4526.935
[33] 4570.133 4512.558 4613.167 4412.658 4747.170 4872.127 4774.157 4697.992
[41] 4640.538 4576.519 4707.957 4658.228 4772.145 4596.747 4371.145 4779.256
[49] 4427.464 4525.557 4305.716 4564.654 4450.188 4634.591 3989.726 4068.685
[57] 4043.495 3886.869
```

Determine Correlation Accuracy

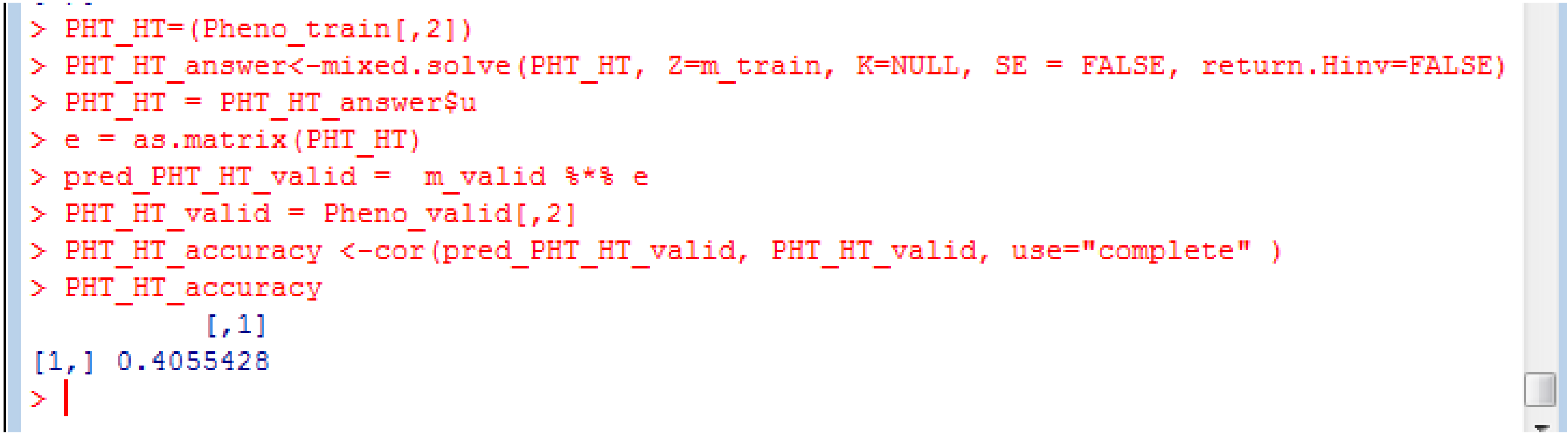
- Correlation between the predicted yield values and the observed yield values
- Accuracy will change slightly each time due to different individuals sampled for the training and validation populations



```
> yield_valid = Pheno_valid[,1]
> YLD_accuracy <-cor(pred_yield_valid, yield_valid, use="complete" )
> YLD_accuracy
          [,1]
[1,] 0.2521498
> 
```

Determine Correlation Accuracy

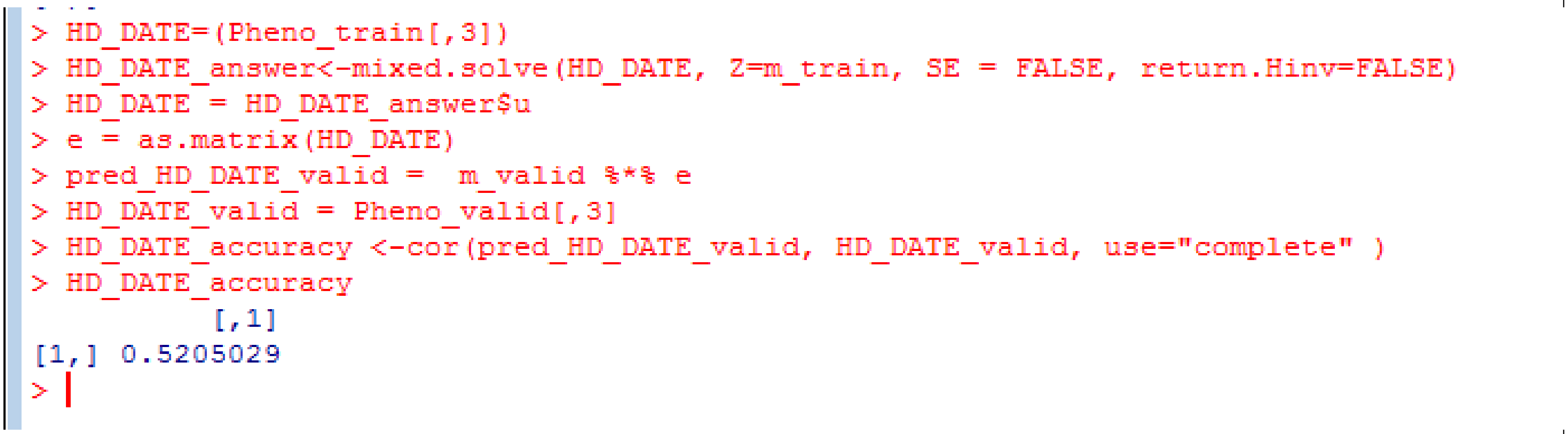
- Plant Height



```
> PHT_HT=(Pheno_train[,2])
> PHT_HT_answer<-mixed.solve(PHT_HT, Z=m_train, K=NULL, SE = FALSE, return.Hinv=FALSE)
> PHT_HT = PHT_HT_answer$u
> e = as.matrix(PHT_HT)
> pred_PHT_HT_valid =  m_valid %*% e
> PHT_HT_valid = Pheno_valid[,2]
> PHT_HT_accuracy <-cor(pred_PHT_HT_valid, PHT_HT_valid, use="complete" )
> PHT_HT_accuracy
          [,1]
[1,] 0.4055428
> 
```

Determine Correlation Accuracy

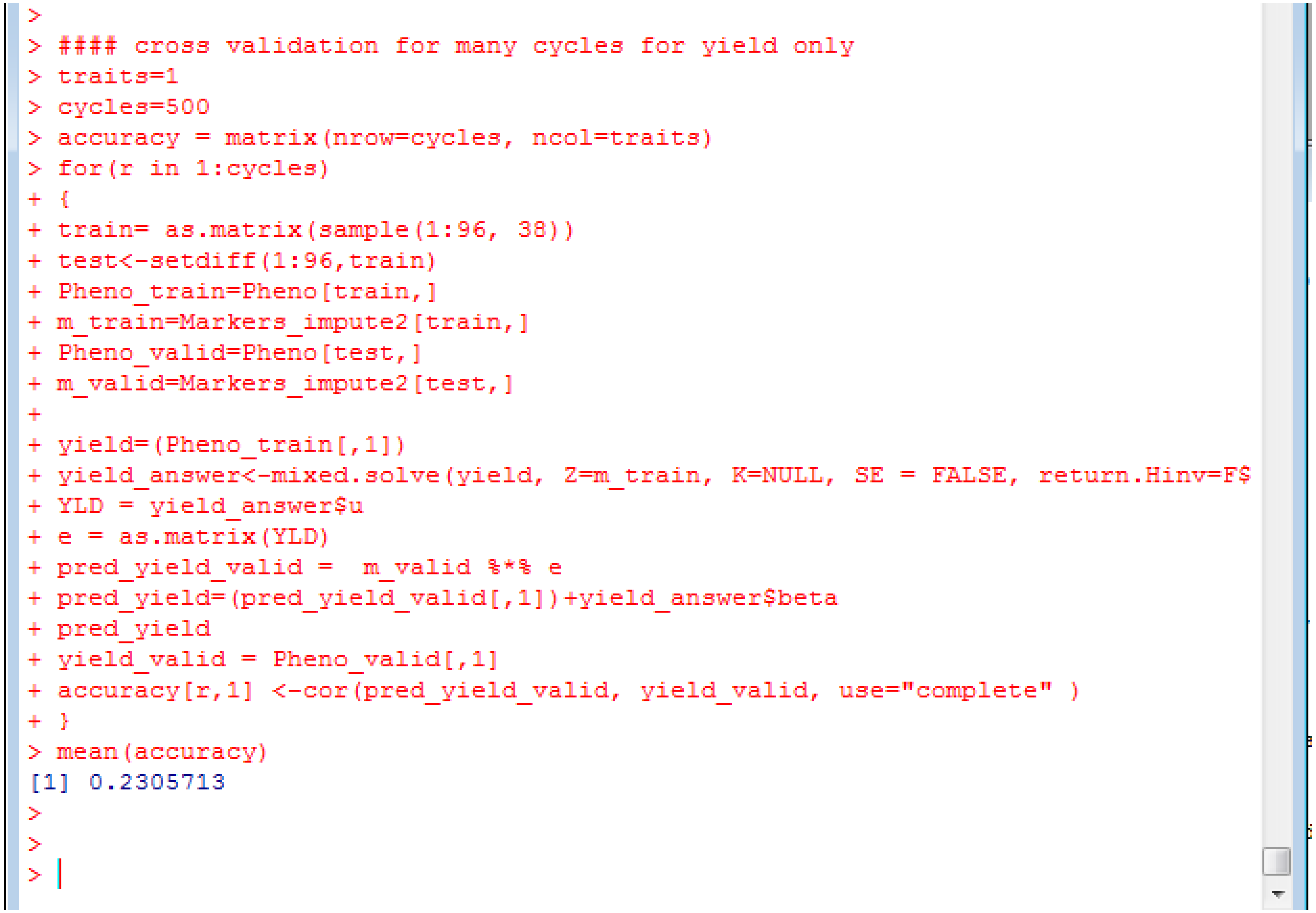
- Heading Date



```
> HD_DATE=(Pheno_train[,3])
> HD_DATE_answer<-mixed.solve(HD_DATE, Z=m_train, SE = FALSE, return.Hinv=FALSE)
> HD_DATE = HD_DATE_answer$u
> e = as.matrix(HD_DATE)
> pred_HD_DATE_valid =  m_valid %*% e
> HD_DATE_valid = Pheno_valid[,3]
> HD_DATE_accuracy <-cor(pred_HD_DATE_valid, HD_DATE_valid, use="complete" )
> HD_DATE_accuracy
          [,1]
[1,] 0.5205029
> 
```

Determine Correlation Accuracy

- Correlation accuracy with 500 iterations



```
>
> #### cross validation for many cycles for yield only
> traits=1
> cycles=500
> accuracy = matrix(nrow=cycles, ncol=traits)
> for(r in 1:cycles)
+ {
+ train= as.matrix(sample(1:96, 38))
+ test<-setdiff(1:96,train)
+ Pheno_train=Pheno[train,]
+ m_train=Markers_impute2[train,]
+ Pheno_valid=Pheno[test,]
+ m_valid=Markers_impute2[test,]
+
+ yield=(Pheno_train[,1])
+ yield_answer<-mixed.solve(yield, Z=m_train, K=NULL, SE = FALSE, return.Hinv=F$
+ YLD = yield_answer$u
+ e = as.matrix(YLD)
+ pred_yield_valid =  m_valid %*% e
+ pred_yield=(pred_yield_valid[,1])+yield_answer$beta
+ pred_yield
+ yield_valid = Pheno_valid[,1]
+ accuracy[r,1] <-cor(pred_yield_valid, yield_valid, use="complete" )
+ }
> mean(accuracy)
[1] 0.2305713
>
>
> |
```

Determine Correlation Accuracy

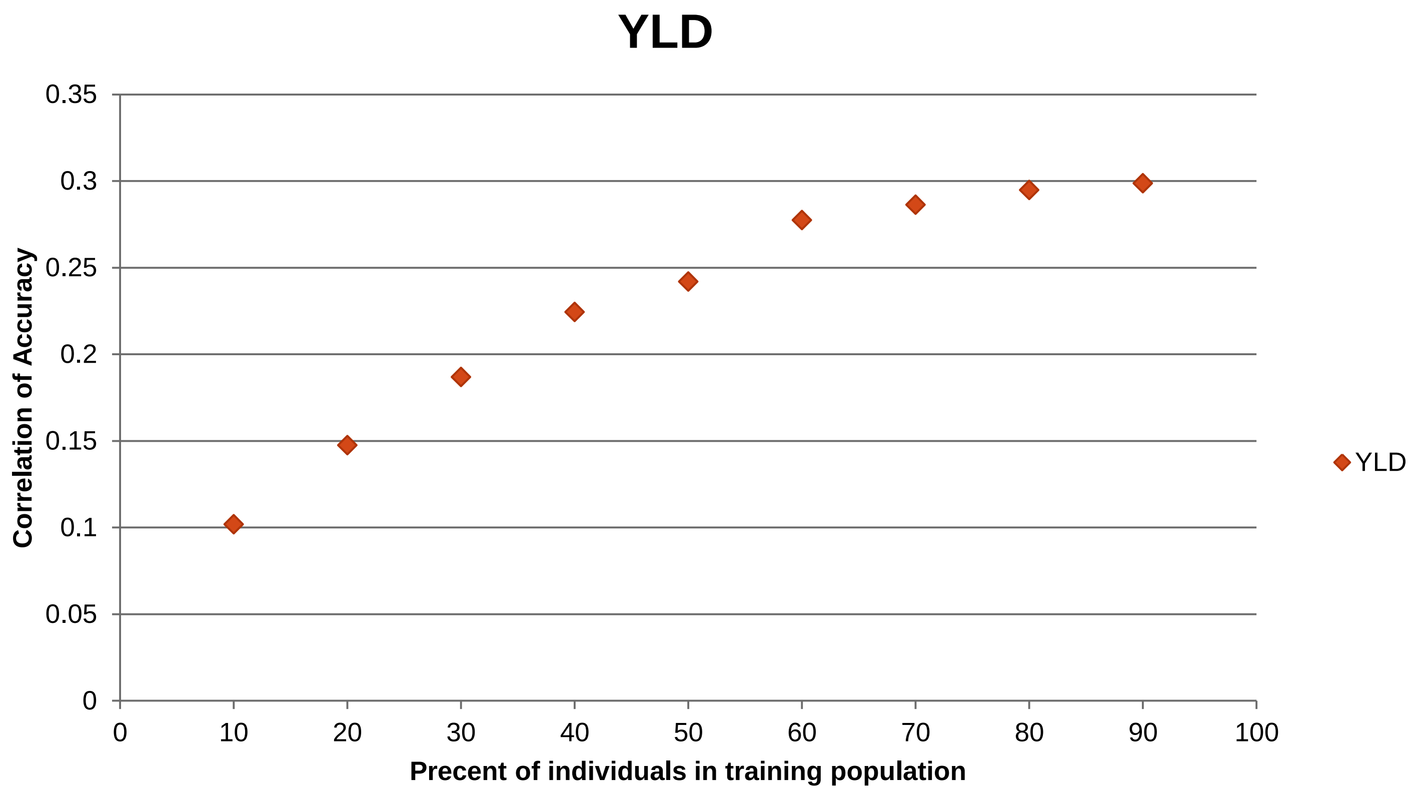
- Correlation accuracy is different for each trait

- Values will be different every time it is run since different lines will be included in the training or validation sets

- Accuracy is affected by training size, validation size, number of markers and heritability

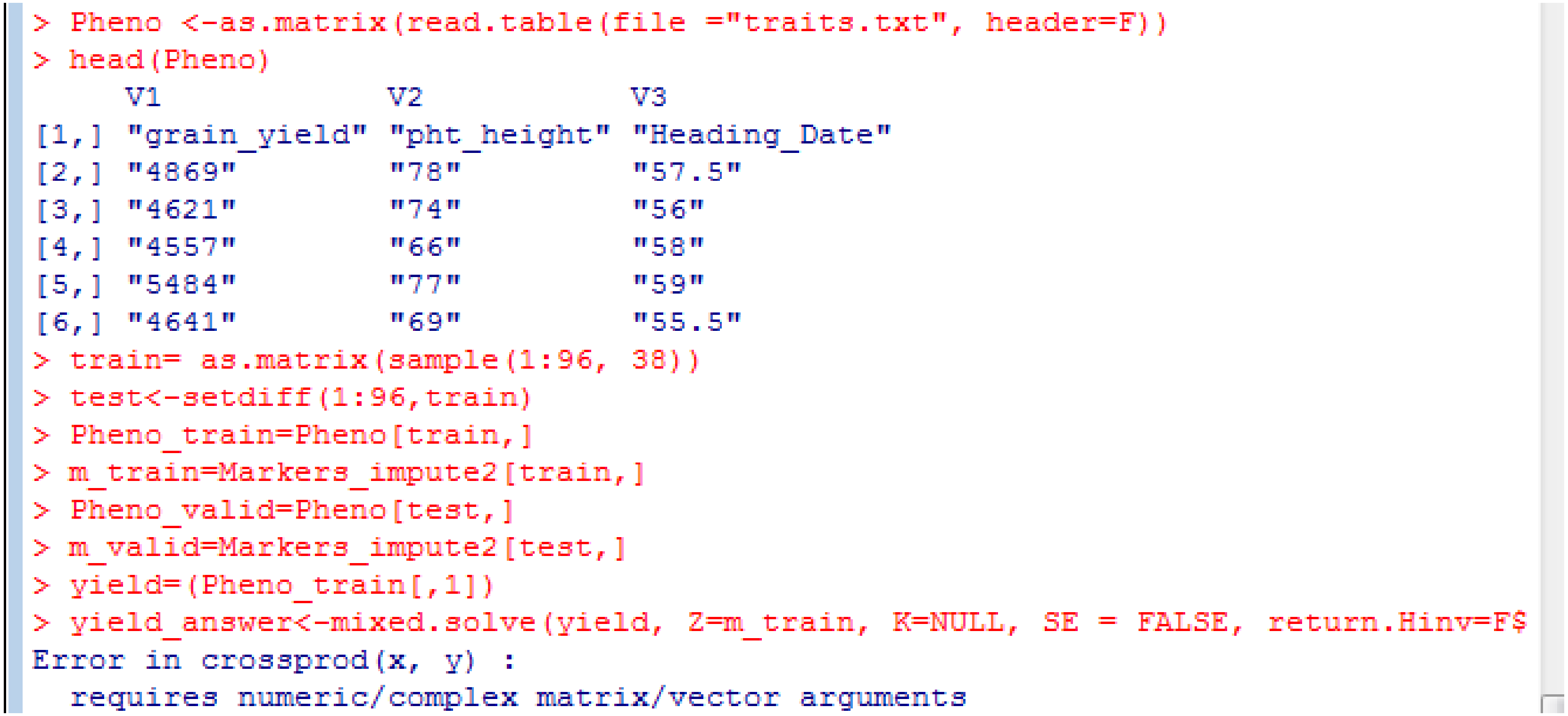
Determine Correlation Accuracy

- Effects of training population size on accuracy



Common Errors

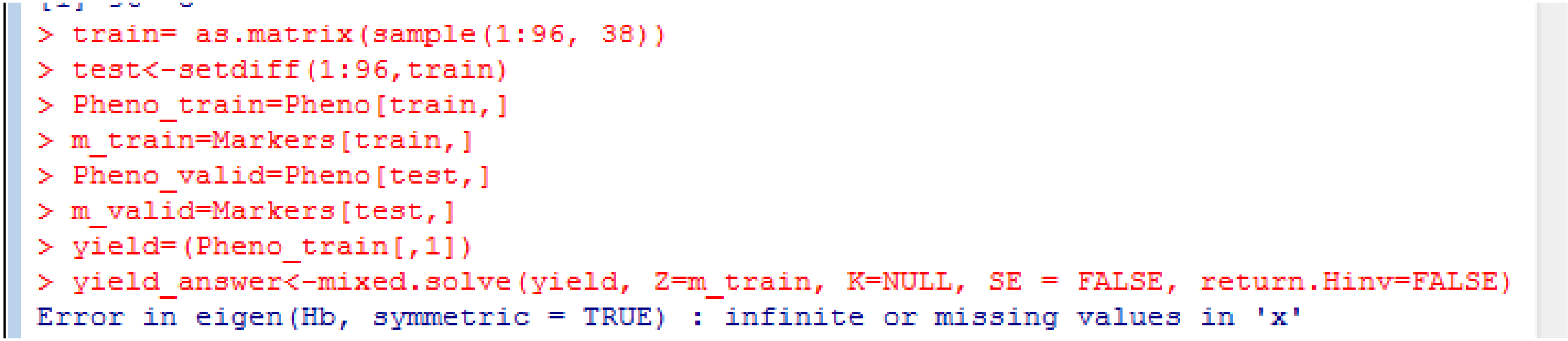
- Headers incorrectly input



```
> Pheno <-as.matrix(read.table(file ="traits.txt", header=F))
> head(Pheno)
     V1            V2           V3            
[1,] "grain_yield" "pht_height" "Heading_Date"
[2,] "4869"        "78"         "57.5"        
[3,] "4621"        "74"         "56"          
[4,] "4557"        "66"         "58"          
[5,] "5484"        "77"         "59"          
[6,] "4641"        "69"         "55.5"        
> train= as.matrix(sample(1:96, 38))
> test<-setdiff(1:96,train)
> Pheno_train=Pheno[train,]
> m_train=Markers_impute2[train,]
> Pheno_valid=Pheno[test,]
> m_valid=Markers_impute2[test,]
> yield=(Pheno_train[,1])
> yield_answer<-mixed.solve(yield, Z=m_train, K=NULL, SE = FALSE, return.Hinv=F$
Error in crossprod(x, y) : 
  requires numeric/complex matrix/vector arguments
```

Common Errors

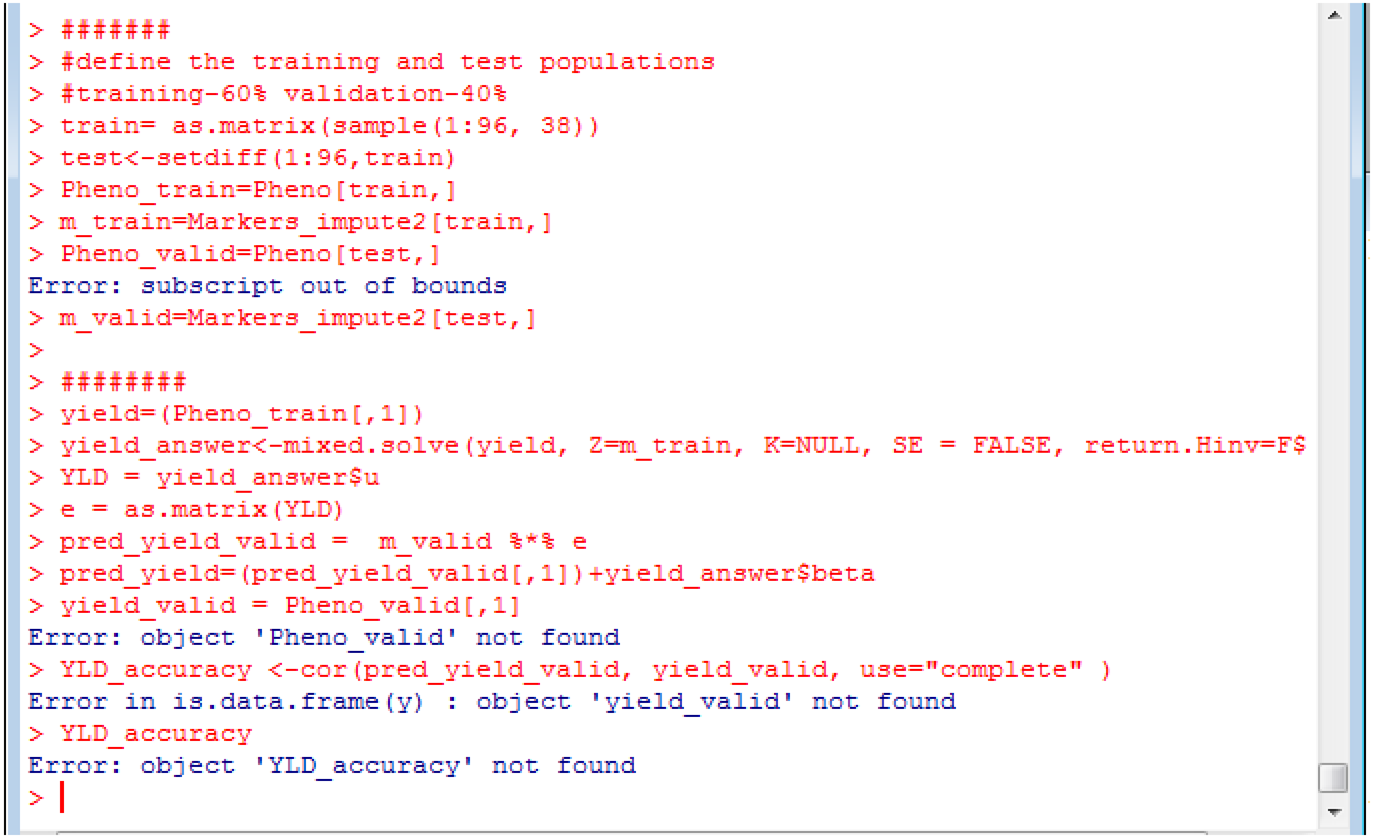
- NA Markers



```
> train= as.matrix(sample(1:96, 38))
> test<-setdiff(1:96,train)
> Pheno_train=Pheno[train,]
> m_train=Markers[train,]
> Pheno_valid=Pheno[test,]
> m_valid=Markers[test,]
> yield=(Pheno_train[,1])
> yield_answer<-mixed.solve(yield, Z=m_train, K=NULL, SE = FALSE, return.Hinv=FALSE)
Error in eigen(Hb, symmetric = TRUE) : infinite or missing values in 'x'
```

Common Errors

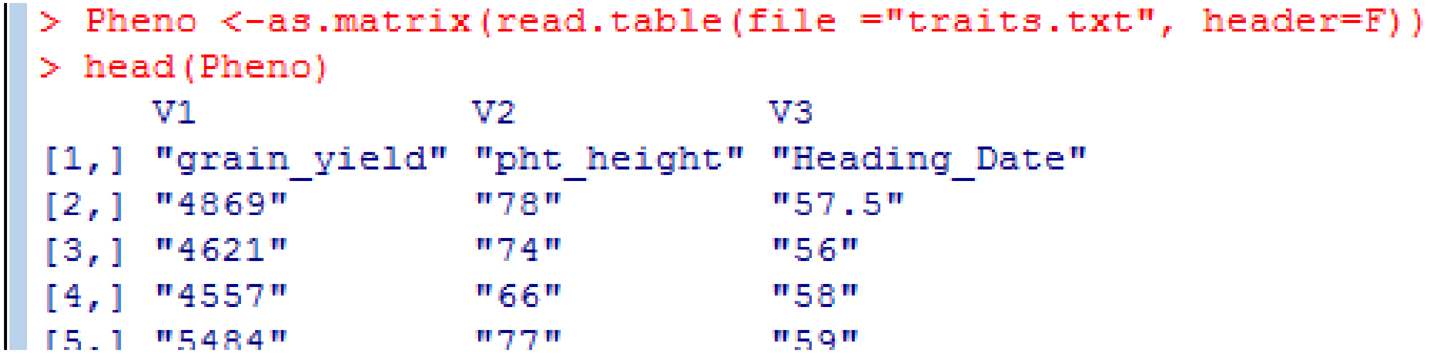
- Incorrect matrix dimensions
  - Removed one individual from phenotype matrix



```
> #######
> #define the training and test populations
> #training-60% validation-40%
> train= as.matrix(sample(1:96, 38))
> test<-setdiff(1:96,train)
> Pheno_train=Pheno[train,]
> m_train=Markers_impute2[train,]
> Pheno_valid=Pheno[test,]
Error: subscript out of bounds
> m_valid=Markers_impute2[test,]
>
> ########
> yield=(Pheno_train[,1])
> yield_answer<-mixed.solve(yield, Z=m_train, K=NULL, SE = FALSE, return.Hinv=F$
> YLD = yield_answer$u
> e = as.matrix(YLD)
> pred_yield_valid =  m_valid %*% e
> pred_yield=(pred_yield_valid[,1])+yield_answer$beta
> yield_valid = Pheno_valid[,1]
Error: object 'Pheno_valid' not found
> YLD_accuracy <-cor(pred_yield_valid, yield_valid, use="complete" )
Error in is.data.frame(y) : object 'yield_valid' not found
> YLD_accuracy
Error: object 'YLD_accuracy' not found
> |
```

Common Errors

- Read in values as characters instead of numeric
  - Quotes around values



```
> Pheno <-as.matrix(read.table(file ="traits.txt", header=F))
> head(Pheno)
     V1            V2           V3
[1,] "grain_yield" "pht_height" "Heading_Date"
[2,] "4869"        "78"         "57.5"
[3,] "4621"        "74"         "56"
[4,] "4557"        "66"         "58"
[5,] "5484"        "77"         "59"
```

Resources

- rrBLUP reference manual
  - http://cran.r-project.org/web/packages/rrBLUP/rrBLUP.pdf
- rrBLUP vingettes
  - http://cran.r-project.org/web/packages/rrBLUP/vignettes/vignette.pdf
- Endelman, J.B. 2011. Ridge regression and other kernels for genomic selection with R package rrBLUP. Plant Genome 4:250-255. doi: 10.3835/plantgenome2011.08.0024

Acknowledgements

Rex Bernardo

Emily Combs

Lian Lian

Chris Schaefer

Lisa-Marie Krchov

rrBLUP

Jeff Endelman

Funding

Monsanto Company

USDA SolCAP

USDA

United States Department of Agriculture

National Institute of Food and Agriculture

MONSANTO

SolCAP

David Francis

Shawn Yarnes

John McQueen

Dataset

TCAP Hordeum's

Toolbox

Questions?