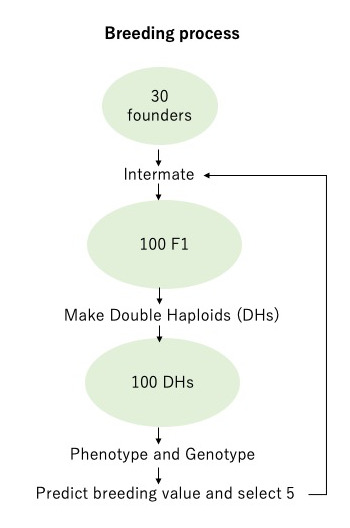
GS vs PS demo

The purpose of this demonstration is to:

1. Show how Genomic Selection (GS) can be used in a population improvement and line development program, and to
2. Show how to compare expected genetic gain from GS with expected genetic gain from phenotypic selection (PS) in order to determine which selection method is better for a given breeding scenario.

Consider the following breeding scenario:



This scenario is a basic recurrent selection program where selection is among double-haploid (DH) lines. Note that line development and population improvement is coupled, but this need not be the case and many other schemes are possible. This scenario was simulated for 2 cycles which generated a dataset containing 200 DH lines. 100 of the lines (group A) are from the first cycle of selection and 100 of the lines (group B) are from the second cycle of selection. There is raw phenotypic data on the 200 DHs, marginal means of phenotypic data computed using the raw phenotypic data, and marker data. Marker data is available for all the germplasm, although we will only use the data on the 200 DHs.

## Load packages

library(lme4)

## Loading required package: Matrix

library(arm)

## Loading required package: MASS

##   
## arm (Version 1.10-1, built: 2018-4-12)

## Working directory is /Users/jrut/Box/UIUC\_SQW\_Breeding\_Workshop/Jessica- Day 1 presentations/PM2 Exercise

library(rrBLUP)

## Load data

setwd("~/Box/UIUC\_SQW\_Breeding\_Workshop/Jessica- Day 1 presentations/PM2 Exercise")  
load("Breeding Population Data.RData")

## Look at the true breeding values

Because we are working with a simulated dataset, we know the true breeding values. Here we can see the data.frame of the true breeding values provided in the workspace.

head(tbv)

## phenoGID tbv  
## 1 1 1.3103430  
## 2 2 1.1549843  
## 3 3 2.0063847  
## 4 4 1.6952985  
## 5 5 2.0699542  
## 6 6 -0.7114222

## Look at the marker data

Here we can see the first 10 rows and first 10 columns of the marker data. The row names are the genotype IDs and the column names are the names of the markers. The markers are recoded as -1, 0, and 1 to reflect the dosage of one of the alleles, centered at zero.

M[1:10, 1:10]

## mrk1 mrk2 mrk3 mrk4 mrk5 mrk6 mrk7 mrk8 mrk9 mrk10  
## 1 -1 0 -1 0 0 0 -1 -1 1 -1  
## 2 -1 0 -1 1 -1 -1 -1 -1 -1 -1  
## 3 -1 0 -1 0 0 0 0 -1 1 -1  
## 4 -1 0 0 0 0 0 -1 -1 1 -1  
## 5 -1 1 -1 -1 -1 -1 -1 -1 1 -1  
## 6 -1 1 -1 -1 -1 -1 -1 -1 1 -1  
## 7 -1 0 -1 0 -1 -1 -1 0 0 -1  
## 8 -1 1 -1 -1 -1 -1 -1 -1 1 -1  
## 9 -1 1 -1 0 -1 -1 -1 -1 0 -1  
## 10 -1 0 -1 0 0 0 -1 -1 1 -1

## Look at the phenotypic data

Here we can see the first 6 rows of the phenotypic data. The phenotypic values are in the ‘pValue’ column. The genotype IDs are in the ‘phenoGID’ column. The phenotypic values are means across replicates. The phenotypic errors and standard errors are in the ‘error’ and ‘se’ columns respectively. The error is simply the square of the standard error. The level of error is different for each location.

head(pheno)

## phenoGID loc year pValue error se population  
## 1 131 1 1 -0.2018914 5 2.236068 A  
## 2 132 1 1 0.8654488 5 2.236068 A  
## 3 133 1 1 0.6071817 5 2.236068 A  
## 4 134 1 1 4.4765200 5 2.236068 A  
## 5 135 1 1 1.6748868 5 2.236068 A  
## 6 136 1 1 2.3879909 5 2.236068 A

Also check the structure of the phenotypic data

str(pheno)

## 'data.frame': 800 obs. of 7 variables:  
## $ phenoGID : Factor w/ 200 levels "131","132","133",..: 1 2 3 4 5 6 7 8 9 10 ...  
## $ loc : Factor w/ 4 levels "1","2","3","4": 1 1 1 1 1 1 1 1 1 1 ...  
## $ year : Factor w/ 2 levels "1","2": 1 1 1 1 1 1 1 1 1 1 ...  
## $ pValue : num -0.202 0.865 0.607 4.477 1.675 ...  
## $ error : num 5 5 5 5 5 5 5 5 5 5 ...  
## $ se : num 2.24 2.24 2.24 2.24 2.24 ...  
## $ population: chr "A" "A" "A" "A" ...

Create an environment column which is the combination of location and year. We will use this variable to capture the interaction between location and year

pheno$env<- paste(pheno$loc, pheno$year, sep="\_")  
head(pheno)

## phenoGID loc year pValue error se population env  
## 1 131 1 1 -0.2018914 5 2.236068 A 1\_1  
## 2 132 1 1 0.8654488 5 2.236068 A 1\_1  
## 3 133 1 1 0.6071817 5 2.236068 A 1\_1  
## 4 134 1 1 4.4765200 5 2.236068 A 1\_1  
## 5 135 1 1 1.6748868 5 2.236068 A 1\_1  
## 6 136 1 1 2.3879909 5 2.236068 A 1\_1

## Fit a mixed model to the data

Here we are going to use all the phenotypic available data on the 200 DHs to determine which are the best lines to use for the next cycle of breeding. We will use an ordinary mixed model that will assume that our lines are independently and identically distributed (iid). In other words, we will not tell the model that our lines are related and therefore the assumption (although incorrect) is that our lines are not genetically related. From this model we will identify the ‘best lines’ as thoes with the highest BLUP values. Note that because population A was evaluated in year 1 and population B was evaluated in year 2, population and year effects are completely confounded and we can only estimate phenoGID effects within year.

mod\_iid<- lmer(pValue~1+loc+(1|phenoGID), data=pheno, subset= which(pheno$year=='2'))

## Get the model results

rslt<- ranef(mod\_iid)$phenoGID  
rslt<- data.frame(phenoGID= row.names(rslt), blup\_iid= rslt[,1])  
rslt[order(rslt$blup\_iid, decreasing=TRUE),][1:10,] #shows the top ten lines

## phenoGID blup\_iid  
## 91 421 1.1290066  
## 80 410 1.1060454  
## 28 358 1.0415060  
## 100 430 0.9497925  
## 54 384 0.9495996  
## 69 399 0.8829963  
## 99 429 0.8437862  
## 18 348 0.8309908  
## 60 390 0.8219114  
## 56 386 0.8126943

## Compare with the true top ten

Notice that what the ‘best’ lines we identified based on BLUPs calculated from our phenotypic data does not correspond very well with the true best lines. This is because phenotype= true breeding value + error. For any given population, the greater the magnitude of the error, the lower the heritability and the lower the phenotypic selection accuracy.

tbv[order(tbv$tbv, decreasing=TRUE),][1:10,] #shows the true top 10 lines

## phenoGID tbv  
## 332 332 4.713768  
## 408 408 4.528768  
## 360 360 4.428033  
## 367 367 4.407856  
## 362 362 4.300402  
## 385 385 4.273832  
## 370 370 4.246234  
## 421 421 4.225093  
## 384 384 4.215880  
## 355 355 4.160330

## Compute the reliability of the BLUPs

se\_blupiid<- se.ranef(mod\_iid)$phenoGID  
vcomp<- data.frame(VarCorr(mod\_iid))  
Vg<- vcomp[vcomp$grp=='phenoGID', 'vcov']  
rel<- 1-se\_blupiid/Vg  
unique(rel)

## (Intercept)  
## 331 0.2623714

This is the square of the selection accuracy, thus the selection accuracy is sqrt(0.262)= 0.512

## Fit a genomic BLUP model to the data

Here we are going to use all the available data, both phenotypic and marker data, on the 200 DHs to determine which are the best lines to use for the next cycle of breeding. The best lines will be the ones with the highest genomic breeding values. To estimate breeding values we will use a genomic BLUP model, which is a mixed model with phenoGID as a random effect. phenoGIDs are assumed related according to a genomic relationship matrix.

First we will create the relationship matrix

Msub<- M[levels(pheno$phenoGID),] #marker data on the 200 DHs  
Gsub<- A.mat(Msub)

Next, create the design matrices and response vector

X<- model.matrix(pValue~1+env, data= pheno)  
Z<- model.matrix(pValue~0+phenoGID, data=pheno)  
y<- pheno$pValue

Next, pre-multiply the design matrices by the inverse of the residual covariance matrix. This accounts for heterogeneous errors.

sqrt.R<- pheno$se  
X2<- X/sqrt.R  
y2<- y/sqrt.R  
Z2<- Z/sqrt.R

Next, fit the model

mod<- mixed.solve(y2, Z2, K=Gsub, X2)

Next get the GEBVs for the 200 DHs

rsltGBLUP<- data.frame(phenoGID= names(mod$u), blup\_GBLUP= mod$u)  
head(rsltGBLUP)

## phenoGID blup\_GBLUP  
## 131 131 -0.6964888  
## 132 132 -1.2919768  
## 133 133 -2.4055189  
## 134 134 -0.5201939  
## 135 135 -0.3376844  
## 136 136 -0.5375650

Lets compare the results from GBLUP and iid BLUP. Lets also compare the results with the true breeding values, which is possible only because this is a simulated dataset.

bluprslt<- merge(rsltGBLUP, rslt, all.x = TRUE)  
head(bluprslt)

## phenoGID blup\_GBLUP blup\_iid  
## 1 131 -0.6964888 NA  
## 2 132 -1.2919768 NA  
## 3 133 -2.4055189 NA  
## 4 134 -0.5201939 NA  
## 5 135 -0.3376844 NA  
## 6 136 -0.5375650 NA

tail(bluprslt)

## phenoGID blup\_GBLUP blup\_iid  
## 195 425 0.7989017 -0.5407188  
## 196 426 1.4385499 -0.2087152  
## 197 427 0.6859209 -0.4314721  
## 198 428 0.2467362 -0.5197913  
## 199 429 1.1785286 0.8437862  
## 200 430 1.0273594 0.9497925

Lets also compare the results with the true breeding values, which is possible only because this is a simulated dataset.

allbv<- merge(bluprslt, tbv, all.x = TRUE)  
head(allbv)

## phenoGID blup\_GBLUP blup\_iid tbv  
## 1 131 -0.6964888 NA 0.79376562  
## 2 132 -1.2919768 NA -0.45229889  
## 3 133 -2.4055189 NA -1.59084425  
## 4 134 -0.5201939 NA 0.57614403  
## 5 135 -0.3376844 NA 2.41097282  
## 6 136 -0.5375650 NA -0.03453295

tail(allbv)

## phenoGID blup\_GBLUP blup\_iid tbv  
## 195 425 0.7989017 -0.5407188 3.802343  
## 196 426 1.4385499 -0.2087152 3.188705  
## 197 427 0.6859209 -0.4314721 3.575133  
## 198 428 0.2467362 -0.5197913 2.963424  
## 199 429 1.1785286 0.8437862 3.283049  
## 200 430 1.0273594 0.9497925 2.920454

Now we will compute the accuracy of the blups by correlating them with the true breeding values, again only possible becuase this is a simulated dataset. This tells us the accuracy of selection

cor(na.omit(allbv[,-1]))

## blup\_GBLUP blup\_iid tbv  
## blup\_GBLUP 1.0000000 0.4655944 0.5968753  
## blup\_iid 0.4655944 1.0000000 0.1813521  
## tbv 0.5968753 0.1813521 1.0000000

Notice that the accuracy of the blup\_iids is similar to what we computed earlier based on the reliability of the iid BLUPs. Also notice that the accuracy of selection with GBLUP is much higher than accuracy of selection with iid BLUP. This is because we use much more data in the model with GBLUP and we are able to borrow information from relatives.

## Comparing breeding strategies

Up until now we are using phenotypic data on our selection candidates of interest in the model, but what if we exclude that information to mimic a scenario where we select prior to collecting phenotypic information in order to reduce the breeding cycle duration? We could potentially increase the rate of genetic gain, but we need to know the level of selection accuracy in order to evaluate this possibility.

First, exclude population B phenotypic data NA

pheno<- pheno[-which(pheno$population=='B'),]

Next, create the design matrices and response vector

X<- model.matrix(pValue~1+env, data= pheno)  
Z<- model.matrix(pValue~0+phenoGID, data=pheno)  
y<- pheno$pValue

Next, pre-multiply the design matrices by the inverse of the residual covariance matrix. This accounts for heterogeneous errors.

sqrt.R<- pheno$se  
X2<- X/sqrt.R  
y2<- y/sqrt.R  
Z2<- Z/sqrt.R

Next, fit the model and get the GLUPs

mod<- mixed.solve(y2, Z2, K=Gsub, X2)  
rsltGBLUP<- data.frame(phenoGID= names(mod$u), blup\_GBLUP= mod$u)  
head(rsltGBLUP)

## phenoGID blup\_GBLUP  
## 131 131 -1.2060550  
## 132 132 -1.7143240  
## 133 133 -2.6219827  
## 134 134 0.1767888  
## 135 135 -0.8614000  
## 136 136 -0.5864202

Merge the GBLUPs with the true breeding values

gebv\_tbv2<- merge(rsltGBLUP, tbv, all.x = TRUE)  
head(gebv\_tbv2)

## phenoGID blup\_GBLUP tbv  
## 1 131 -1.2060550 0.79376562  
## 2 132 -1.7143240 -0.45229889  
## 3 133 -2.6219827 -1.59084425  
## 4 134 0.1767888 0.57614403  
## 5 135 -0.8614000 2.41097282  
## 6 136 -0.5864202 -0.03453295

Compute the correlation between our GBLUPs and the true breeding values for population B, which are in rows 201 to 400

cor(gebv\_tbv2[101:200, -1])

## blup\_GBLUP tbv  
## blup\_GBLUP 1.0000000 0.4522785  
## tbv 0.4522785 1.0000000

The accuracy of selection is 0.452, recall that we found the accuracy of selection when phenotypic data was included on the selection candidates to be 0.597. Lets assume that the breeding cycle duration for selection before phenotyping is 1.5 years, and breeding cycle duration for selection after phenotyping is 2.5 years. We can use the breeders equation to determine which approach is better. We can assume that the selection intensity and the additive genetic variance is the same for both methods. Then all we need to compare is the accuracy/time for both methods. 0.452/1.5= 0.301, and 0.597/2.5= 0.239. Thus GS with selection prior to phenotyping is better than GS with selection after phenotyping.

We can also compare with selection based on iid BLUPs. Based on our results, accuracy over time for this method is 0.18/2.5= 0.072. In our scenario, this approach is less effective.