

Maize Selection Tests

Investigating genome size variation in maize and teosinte. First, I plot the data to visualize the variation across altitudinal clines.

Read in the maize data.

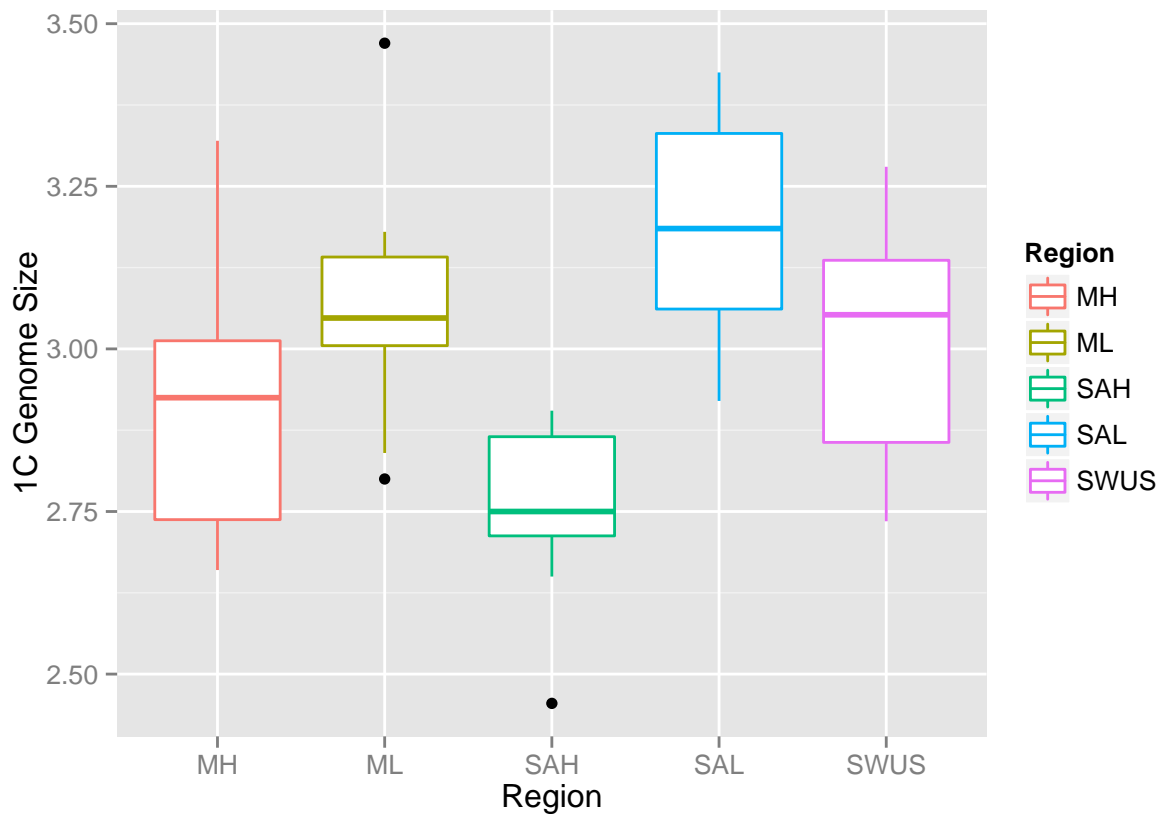
```
setwd("~/Documents/Projects/Genome_Size_Analysis")
library(ggplot2)
```

```
## Warning: package 'ggplot2' was built under R version 3.0.2
```

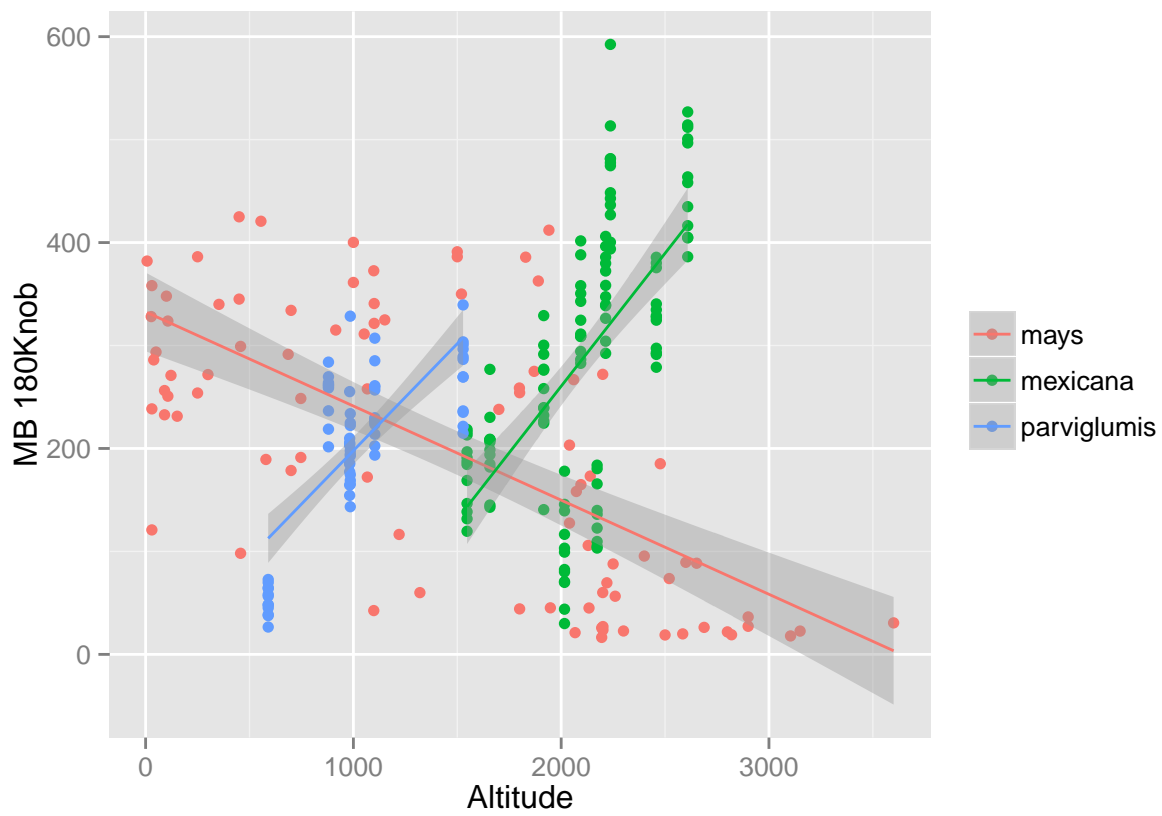
```
dataall <- read.csv("Master_Data_noNA.csv")
dataal <- subset(dataall, dataall$X1C_GS!="NA")
data <- subset(dataal, dataal$X1C_GS<3.6)
dmays <- subset(data, data$Species=="mays")
```

Plot the variation.

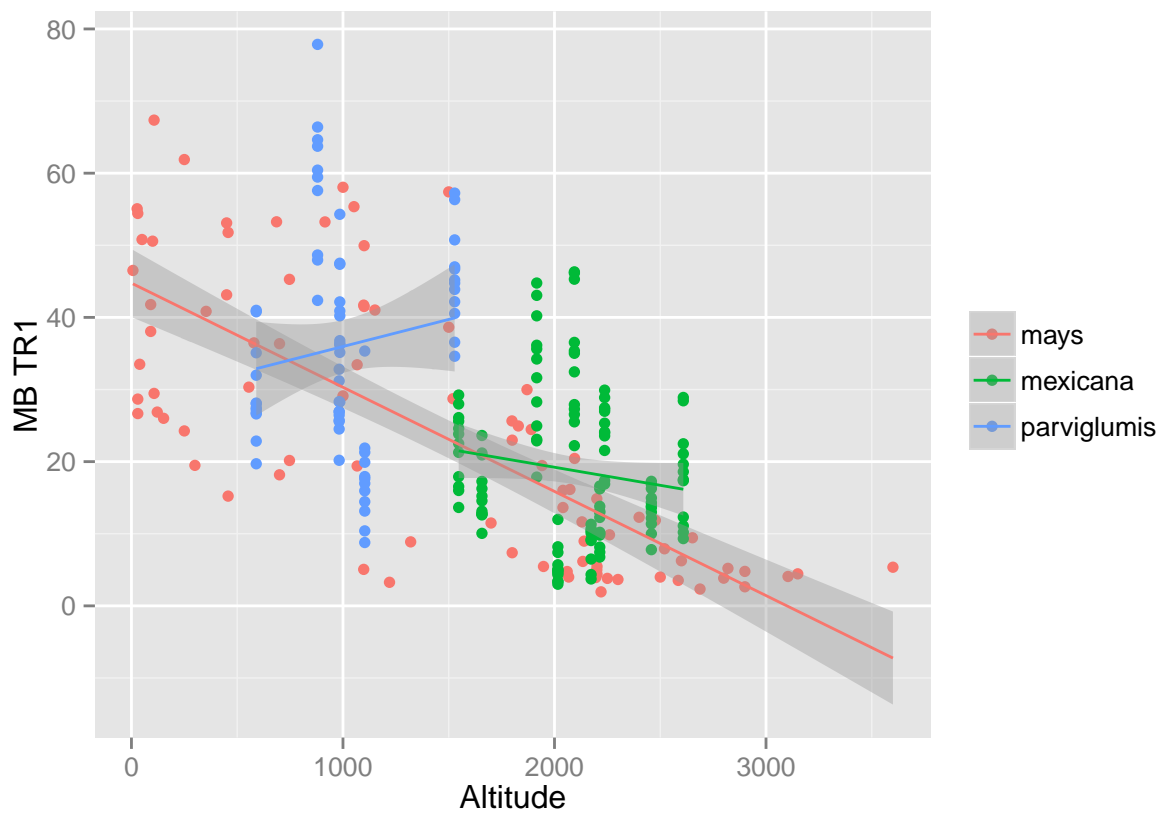
```
p1 <- ggplot(dmays, aes(Region, X1C_GS, color=Region)) + geom_boxplot() + ylab("1C Genome Size")
p1
```



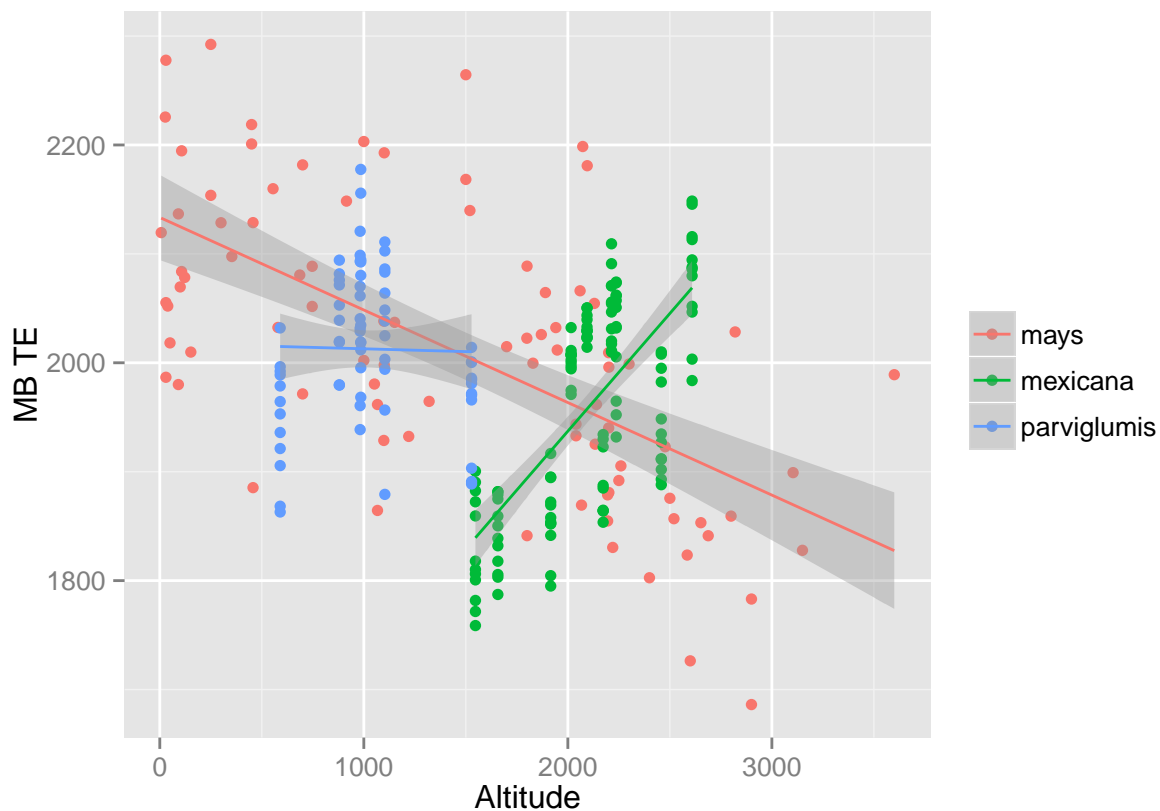
```
p2 <- ggplot(data, aes(Altitude, X180knobMB, color=Species)) + geom_point() + ylab("MB 180Knob") + theme_minimal()
p2
```



```
p <- ggplot(data, aes(Altitude, TR1MB, color=Species)) + geom_point()+ ylab("MB TR1") + theme(legend.ti
p
```



```
p <- ggplot(data, aes(Altitude, TotallTeMB, color=Species)) + geom_point() + ylab("MB TE") + theme(legend
p
```



Next, we set up the data for studying selection on genomic elements. Read in the genotype data with this code, and generate the genetic matrix with the following code.

```
setwd("~/Documents/Projects/Genome_Size_Analysis/Github_ParallelGS/SelectionTests/")
library("rrBLUP")
```

```
## Warning: package 'rrBLUP' was built under R version 3.0.2
```

```
geno <- read.csv("~/Documents/Projects/Genome_Size_Analysis/Github_ParallelGS/SNP_data/Landrace_noSWUS_r")
dt <- t(geno)
A <- A.mat(dt)
```

Read in the phenotype data, and make sure the order of the samples are the same as order in the genetic matrix.

```
pheno <- read.csv("~/Documents/Projects/Genome_Size_Analysis/Github_ParallelGS/PhenotypeData/Landraces_noSWUS_r")
#to make sure order is the same
tmp1 <- as.data.frame(colnames(geno))
names(tmp1)[1] <- "names"
tmp2 <- as.data.frame(pheno$FullID)
tmp <- setdiff(tmp1, tmp2)

phenoorder <- merge(tmp1, pheno, by.x="names", by.y="FullID", sort=FALSE)
```

Jeremy Berg's equation for testing selection.

```

library ( mvtnorm )

## Warning: package 'mvtnorm' was built under R version 3.0.2

EnvVarTest <- function ( phenos , kinship.mat , test.vector, verbose=F ) {

  # 'phenos' is a vector containing the phenotype
  # (i.e. number of repeats) for each individual; dimensions are N x 1
  # 'kinship.mat' is the kinship matrix; dimensions are N x N;
  # rows and columns need to be in the same order as the phenotypes in the vector
  # test.vector is the environmental factor of interest (in this case altitude)

  eigs <- eigen ( kinship.mat )
  # get eigendecomposition of kinship matrix
  rt.inv <- eigs$vec %*% diag ( sqrt(eigs$val) )
  # calculate inverse of the square root matrix
  rotated.phenos <- t ( rt.inv ) %*% phenos
  # rotate phenotypes from population space into principal component space
  test.vector <- test.vector / (sqrt ( 2 * sum ( test.vector^2 ) ) )
  # scale to be unit length after rotation
  #recover()
  rotated.vector <- rt.inv %*% test.vector
  # rotate environmental variable from population space into principal component space
  model <- lm ( rotated.phenos ~ 1+rotated.vector)
  # fit regression model
  r.sq <- cor.test ( rotated.phenos , rotated.vector )$estimate^2
  # get r^2
  ANOVA <- anova ( model )
  # get p value
  if(verbose){ print(ANOVA) }
  return ( c ( model$coef[2] , r.sq , ANOVA[5][[1]][1] ) ) # return
}

```

Now run the equations on each of the genetic phenotypes. Altitude is our environmental variable, and we will expand on doing this with other bioclim variables. The MB signifies megabases of the repeat, and the . is a replacement for the % symbol.

```
EnvVarTest(phenoorder$X180knobMB,A,phenoorder$Altitude)
```

```
## rotated.vector      cor
##      994.55885      0.07051      0.01526
```

```
EnvVarTest(phenoorder$X180knob.,A,phenoorder$Altitude)
```

```
## rotated.vector      cor
##      31.06811      0.07232      0.01396
```

```
EnvVarTest(phenoorder$TR1MB,A,phenoorder$Altitude)
```

```
## rotated.vector      cor
##      123.35322      0.05521      0.03250
```

```
EnvVarTest(phenoorder$TR1.,A,phenoorder$Altitude)
```

```
## rotated.vector      cor
##      3.78601      0.05539      0.03221
```

```
EnvVarTest(phenoorder$CentCMB,A,phenoorder$Altitude)
```

```
## rotated.vector      cor
##     -14.38804      0.08506      0.00747
```

```
EnvVarTest(phenoorder$CentC.,A,phenoorder$Altitude)
```

```
## rotated.vector      cor
##     -0.657070      0.088293      0.006373
```

Chunk for all of the teosintes. Using the population covariance matrices from Bayenv courtesy of Tim Beissenger, and the pop averages from my phenotypes stuff.

```
phenoteo <-read.csv("~/Documents/Projects/Genome_Size_Analysis/Github_ParallelGS/PhenotypeData/Teosinte")
```

```
genoteo <- read.csv("~/Documents/Projects/Genome_Size_Analysis/Github_ParallelGS/SNP_data/TeosinteAll_Bayenv")
```

```
EnvVarTest(phenoteo$X180knobMB,genoteo,phenoteo$Altitude)
```

```
## rotated.vector      cor
##      2.004e+03      1.455e-02      6.563e-01
```

```
EnvVarTest(phenoteo$X180knob.,genoteo,phenoteo$Altitude)
```

```
## rotated.vector      cor
##      66.55151      0.01456      0.65627
```

```
EnvVarTest(phenoteo$TR1MB,genoteo,phenoteo$Altitude)
```

```
## rotated.vector      cor
##      285.31281      0.03216      0.50634
```

```
EnvVarTest(phenoteo$TR1.,genoteo,phenoteo$Altitude)
```

```
## rotated.vector      cor
##      9.53122      0.03205      0.50708
```

```
EnvVarTest(phenoteo$CentCMB,genoteo,phenoteo$Altitude)
```

```
## rotated.vector      cor
##      124.82616      0.02166      0.58646
```

```
EnvVarTest(phenoteo$CentC.,genoteo,phenoteo$Altitude)
```

```
## rotated.vector      cor
##      4.20633      0.02153      0.58761
```

Took just averages from parv, run the code

```
phenoparv <-read.csv("~/Documents/Projects/Genome_Size_Analysis/Github_ParallelGS/PhenotypeData/Teosinte")
```

```
genoparv <- read.csv("~/Documents/Projects/Genome_Size_Analysis/Github_ParallelGS/SNP_data/TeosinteParv")
```

```
EnvVarTest(phenoparv$X180knobMB,genoparv,phenoparv$Altitude)
```

```
## rotated.vector      cor
##      100.1639      0.0832      0.5793
```

```
EnvVarTest(phenoparv$X180knob.,genoparv,phenoparv$Altitude)
```

```
## rotated.vector      cor
##      3.40691      0.09774      0.54633
```

```
EnvVarTest(phenoparv$TR1MB,genoparv,phenoparv$Altitude)
```

```
## rotated.vector      cor
##      -16.6504      0.2965      0.2640
```

```
EnvVarTest(phenoparv$TR1.,genoparv,phenoparv$Altitude)
```

```
## rotated.vector      cor
##      -0.4838      0.2971      0.2633
```

```
EnvVarTest(phenoparv$CentCMB,genoparv,phenoparv$Altitude)
```

```
## rotated.vector      cor
##      10.2871      0.3354      0.2284
```

```
EnvVarTest(phenoparv$CentC.,genoparv,phenoparv$Altitude)
```

```
## rotated.vector      cor
##      0.3521      0.3696      0.2005
```

Took just averages from mex, run the code

```
phenomex <-read.csv("~/Documents/Projects/Genome_Size_Analysis/Github_ParallelGS/PhenotypeData/Teosinte")
```

```
genomex <- read.csv("~/Documents/Projects/Genome_Size_Analysis/Github_ParallelGS/SNP_data/TeosinteMex_B")
```

```
EnvVarTest(phenomex$X180knobMB,genomex,phenomex$Altitude)
```

```
## rotated.vector      cor
##      -2.253e+03      7.595e-02      4.409e-01
```

```
EnvVarTest(phenomex$X180knob., genomex, phenomex$Altitude)
```

```
## rotated.vector      cor
##      -75.88198      0.07851      0.43296
```

```
EnvVarTest(phenomex$TR1MB, genomex, phenomex$Altitude)
```

```
## rotated.vector      cor
##      -165.4937      0.0982      0.3779
```

```
EnvVarTest(phenomex$TR1., genomex, phenomex$Altitude)
```

```
## rotated.vector      cor
##      -5.6704      0.1018      0.3689
```

```
EnvVarTest(phenomex$CentCMB, genomex, phenomex$Altitude)
```

```
## rotated.vector      cor
##      -105.90915      0.08813      0.40486
```

```
EnvVarTest(phenomex$CentC., genomex, phenomex$Altitude)
```

```
## rotated.vector      cor
##      -3.64500      0.09059      0.39808
```

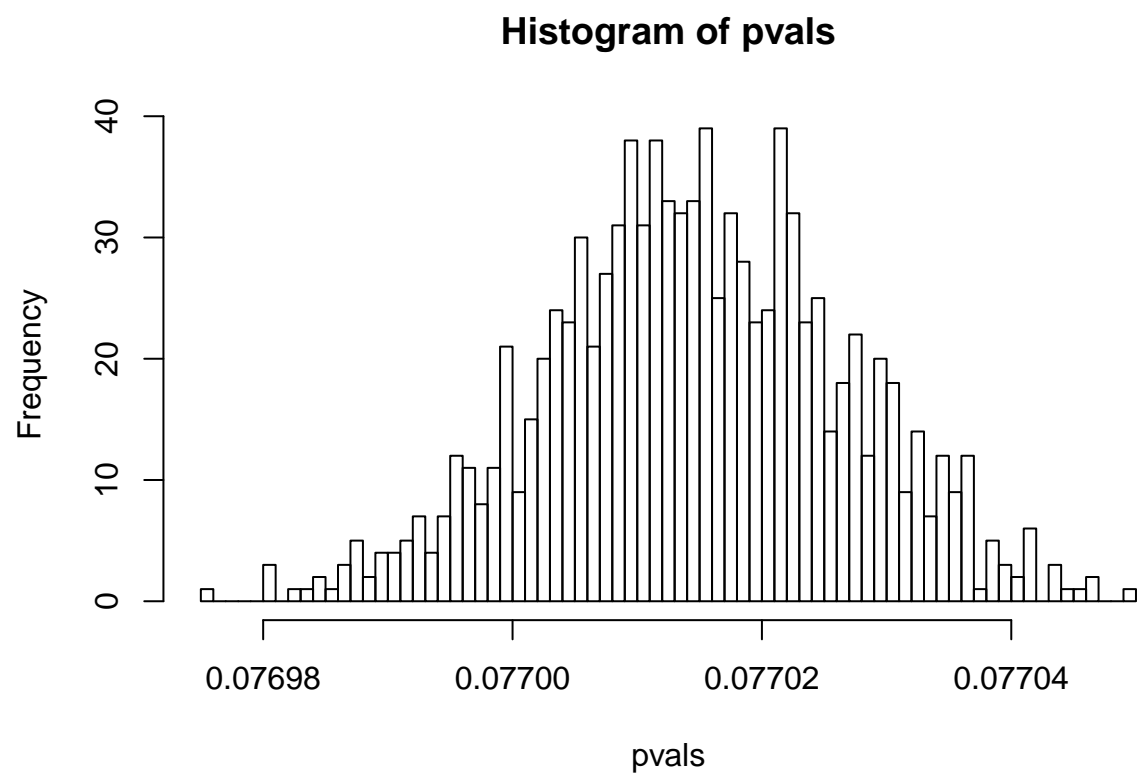
Power sims chunk. The rough draft but working version of it, cleaning it up with a function in the next go around.

```
#install.packages("MASS")
library(MASS)
```

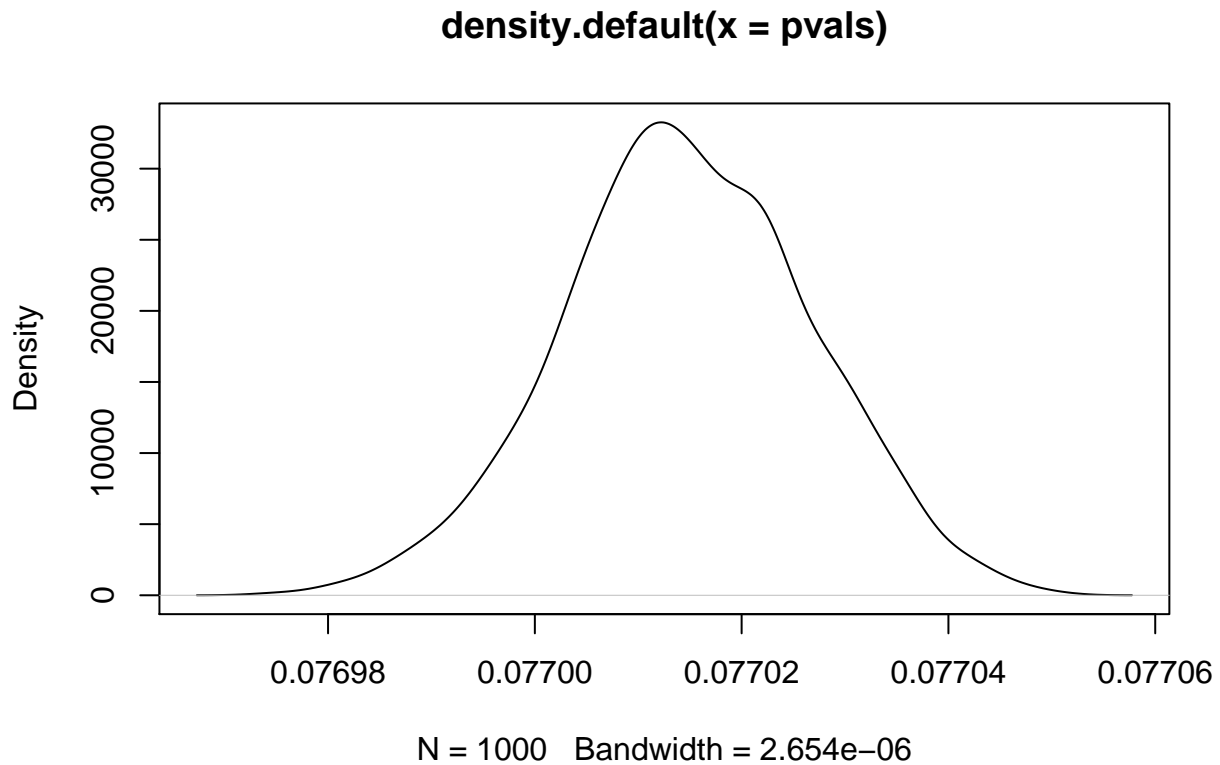
```
## Warning: package 'MASS' was built under R version 3.0.2
```

```
# p = mvn(my + Selection * Altitude, Va * Covariance)
# p will be an output of our phenotypes
mu <- rep(0, 83) #I think since we want 1K sims, we want to make the length of this 1k
Altitude <- phenoorder$Altitude
B <- 10
pizza <- mu + B * Altitude
#covariance matrix is given in A
VA=1 #additive genetics variance
simpheno <- mvrnorm(n=1000, pizza, VA*A) #n=number of samples we have, for maize 83, for allteo 16 pop
#pieces i am missing: where to stick in altitude? I think it is at the mu

pvals=apply(1:1000,function(X) EnvVarTest(simpheno[X,],A,Altitude)[3])
#EnvVarTest(simpheno[X,],A,Altitude,verbose=T)
hist(pvals,breaks=100)
```

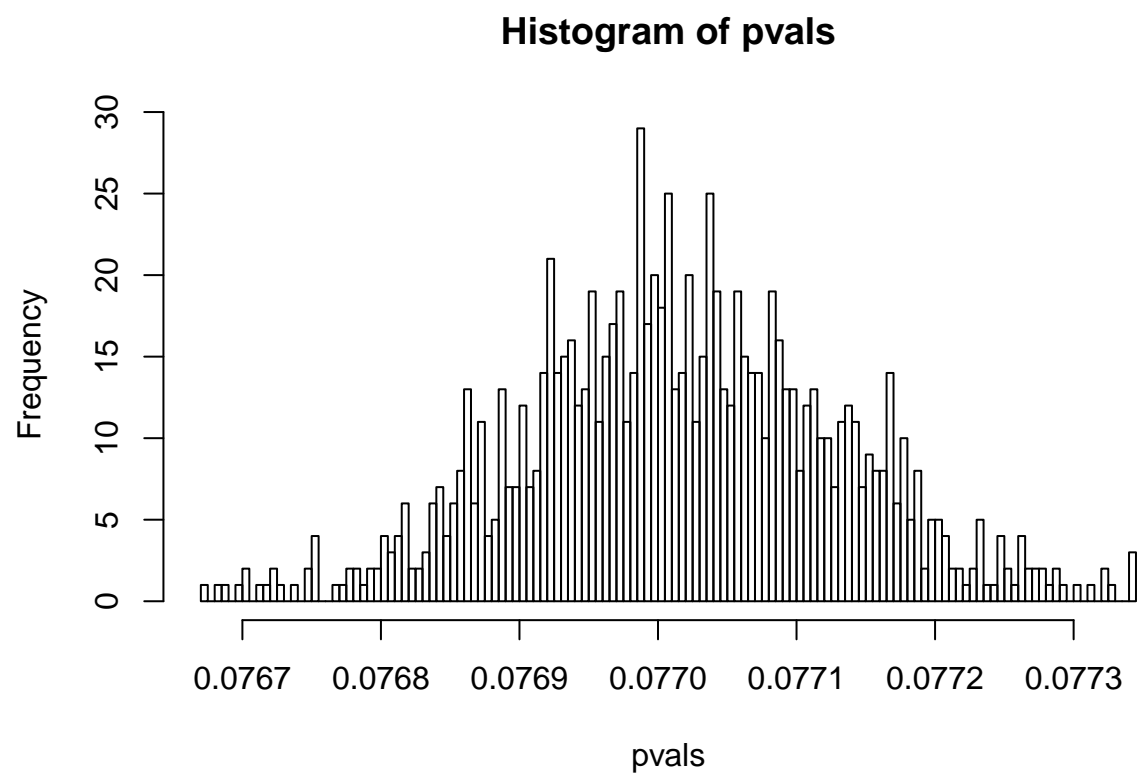
```
plot(density(pvals))
```



Function for sims. For the variables, here is an in text explanation: Samples is how many reps we want to run slope is our beta term, how strong selection is add.gen.var we assume to be 1, but is Va gen.covar.mat is our covariance matrix based on SNP genotypes, built earlier as matrix A altitudes is a vector of altitudes

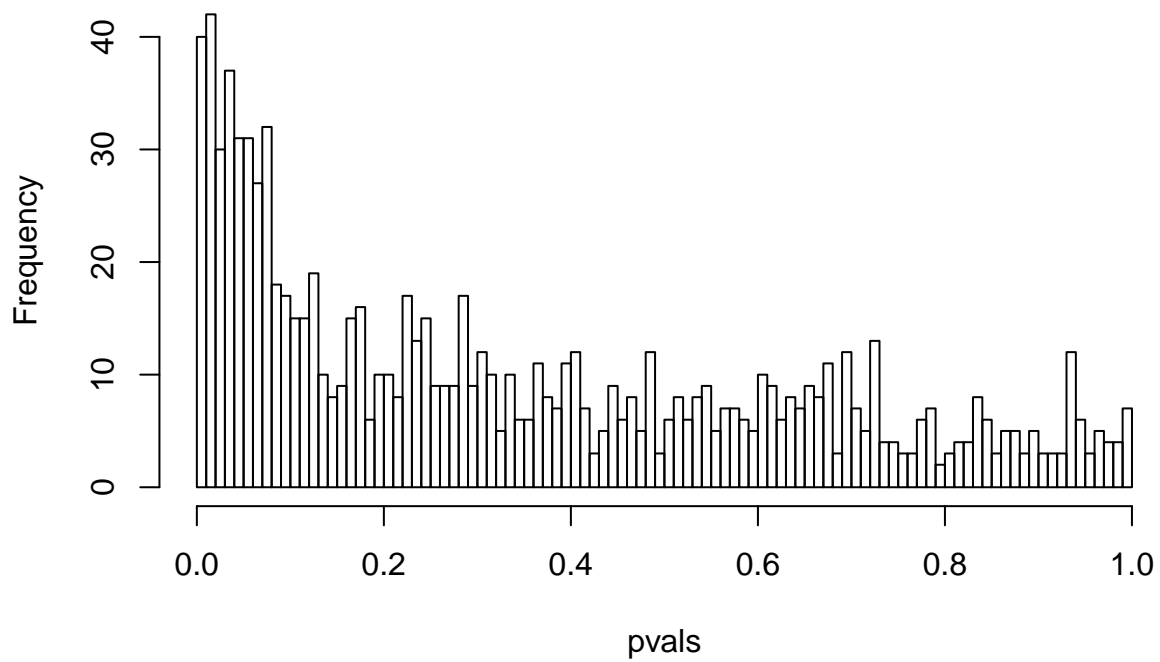
```
sim <- function ( samples , slope , add.gen.var , gen.covar.mat , altitudes , verbose=F) {
  mu <- rep(0, 83)
  Altitude <- altitudes
  GCM <- gen.covar.mat
  B <- slope
  env.vector <- mu + B * Altitude
  VA = add.gen.var
  simpheno <- mvrnorm(n=samples,env.vector, VA*GCM)
  pvals=sapply(1:samples,function(X) EnvVarTest(simpheno[X,],gen.covar.mat,Altitude)[3])
  hist(pvals,breaks=100)
}

sim(1000,1,1,A,phenoorder$Altitude)
```



```
sim(1000,0,1,A,phenoorder$Altitude)
```

Histogram of pvals



So from these histograms, we see the pvalues are all screwed up, and this is where we need help. If beta is greater than 0, it seems to focus all pvalues to 0.077, and does not change if we go to beta 1000 compared to beta 10. At beta zero, we would expect to see a uniform distribution of pvalues, and we don't. Any insight as to what we are doing wrong would be most helpful.