```
##Rice association analysis example
setwd('/home/williams/Desktop/gwas_examples/')

##Import phenotypes
riceFt <- read.csv('ricePhenoZhao.csv', header=TRUE)

##Import snp data. Snp data was previously imputed (i.e., missing snp scores were filled in by predicting the score based on map position and state of surrounding snps for each individuals.
riceSnps <- read.csv('rice413PanelGenoNaImp.csv', header=TRUE, stringsAsFactors=TRUE, row.names=1)

##Read in snp names separately and use them to name the columns. (The read.csv function adds an "X" to the front of column names if they are numbers, which is slightly annoying.)
cNames <- scan('rice413PanelGenoNaImp.csv', sep=',', nlines=1, what='character')[-1]
colnames(riceSnps) <- cNames


##Use simple linear regression to test for significance of marker-trait associations.
snp1 <- as.numeric(riceSnps[1, ])
ft <- riceFt[, 3]
model1900 <- lm(ft  ~ snp1, na.action=na.omit)


##### MODEL RELATIONSHIPS AND POP STRUCTURE ########

##Import data on subpop membership
riceStr <- read.csv('ricePopStrPca.csv')

##Import kinship matrix and convert to matrix. This was previously calculated using emma.kinship function.
riceK <- read.csv('riceKinMat.csv', row.names=1)

##Import SNP physical map
riceMap <- read.csv('sativas413map.csv')

##Analyze SNPs on chromosome 6 around 9e06 bp only to reduce time for in-class example. To analyze full set, just remove this part and replace riceSnps6 with riceSnps below
chr6 <- which(riceMap[, 2]==6 & riceMap[, 3]<8500000 & riceMap[, 3]>7000000)
riceSnps6 <- riceSnps[chr6, ]
riceMap6 <- riceMap[chr6, ]


#Some conversions of object types so EMMA works. Pheno data needs to contain only phenotypic data and needs to be a row vector
riceFt <- t(riceFt[, 3])
```

```
##To model population structure, we will use the first four PCs from a prior PCA analysis. These will be fit as fixed effects.
ricePca <- riceStr[, 10:13]

##EMMA wants everything to be a matrix.
riceFt <- as.matrix(riceFt)
riceSnps6 <- as.matrix(riceSnps6)
riceK <- as.matrix(riceK)
ricePca <- as.matrix(ricePca)


##We'll implement a mixed model to account for pop structure and relatedness using EMMA (Efficient Mixed-Model Association). You can download the EMMA script, save it, and call it up. The easiest thing to do at this point in time is to call it from an existing online location
source("http://www.maizegenetics.net/images/stories/bioinformatics/GAPIT/emma.txt")

## Call up a function from the EMMA package to fit the mixed linear model. This might take some time to run.
emmaOUT <- emma.ML.LRT(ys=riceFt, xs=riceSnps6, K=riceK, X0=ricePca)

##Extract p-values
pVals <- emmaOUT$ps

##Plot -log10(p-value)
library(ggplot2)
qplot(riceMap6[, 3], -log10(pVals), alpha=I(1/2), size=12, ylab='-log10(P)', xlab='Chr6 bp', legend.position='none') + theme_bw(base_size=15) + theme(legend.position='none')


##Determine effective number of tests for setting p-value to ensure reasonable experiment-wise error rate
##Method published by Li and Ji (2005)
##Method includes calculating correlation matrix among SNPs and performing eigenvalue decomposition. This is computationally intensive, so we already did this and will import eigenvalues.
eigVals <- read.csv('eigenValuesRice.csv', row.names=1)
eigVals <- as.matrix(eigVals)

nMrk <- dim(riceSnps)[2]


##Equation 5 in Li and Ji (2005)
Meff=0
for (i in 1:nMrk){
      if (abs(eigVals[i]) >= 1) I=1 else I=0
      fx = I + (abs(eigVals[i]) - floor(abs(eigVals[i])))
      Meff = Meff + fx
```

```
52.  }


53.  ###Assume we want an experiment-wise error rate (alpha_E) to be
     0.05
54.  ##Use Meff to calculate this. alpha_P is the comparison-wise
     error rate to use
55.  alpha_E = 0.05

56.  alpha_P = 1 - (1-alpha_E)^(1/Meff)

57.  ##Add line on Manhattan plot
58.  qplot(riceMap6[, 3], -log10(pVals), alpha=I(1/2), size=12,
     ylab='-log10(P)', xlab='Chr6 bp', legend.position='none') +
     theme_bw(base_size=15) + theme(legend.position='none') +
     geom_hline(y=-log10(alpha_P), size=2, colour='red', alpha=I(1/2),
     linetype=2)
```