Episode I: The QC Menace

Peter Fiorica

August 2, 2019

Introduction

If you are coming to this file from the preQC file, we will be starting with

`/home/peter/prostate_cancer/genotypes_dbGaP/preQC_bfiles` as the genotype files we will begin QC with. There is a total of 4769 individuals (2463 cases and 2306 controls) in the cohort that are both male and have phenotypes.

Quality Control

Note that the numbering nomenclature for the steps is not particularly related to anything. The brief description after the number is important to know what each PLINK command does.

QC Step 0: Sex and heterozygous halpoid check

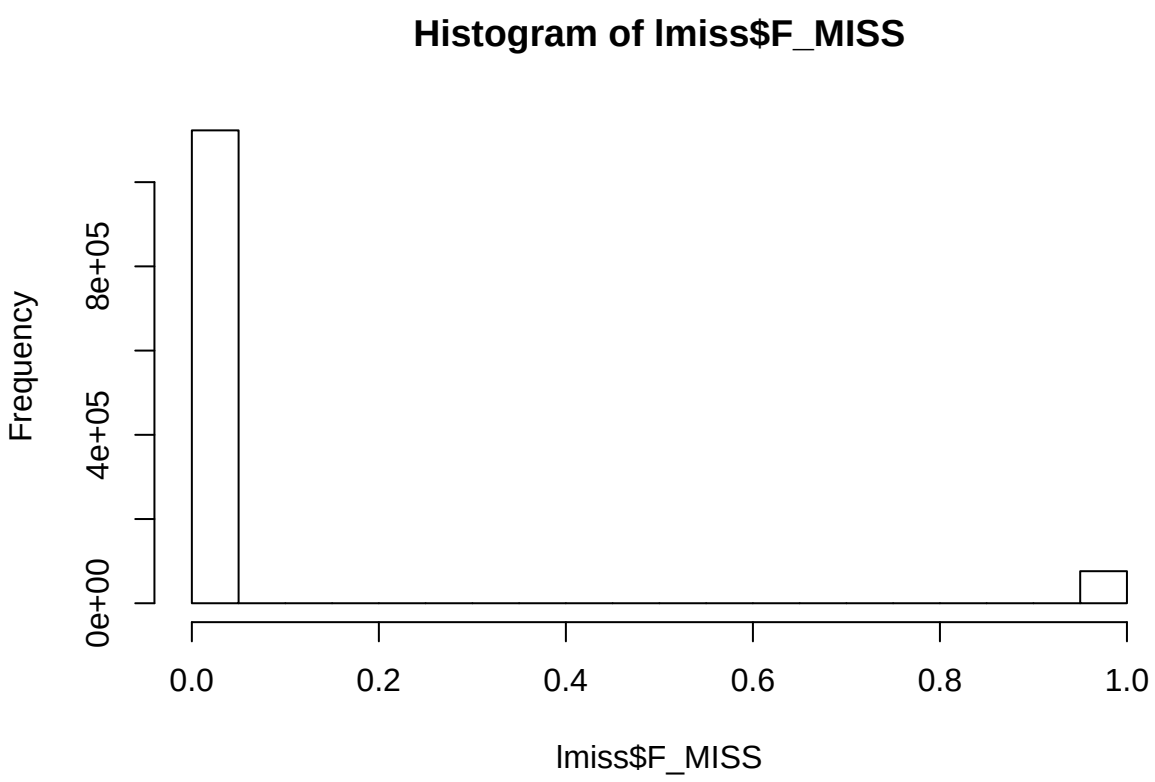
```
plink --bfile /home/peter/prostate_cancer/genotypes_dbGaP/preQC_bfiles
--set-hh-missing --make-bed
--out /home/peter/prostate_cancer/QC_Steps/step0/qcstep0nohh

plink --bfile /home/peter/prostate_cancer/QC_Steps/step0/qcstep0nohh
--check-sex --missing
--out /home/peter/prostate_cancer/QC_Steps/step0/qcstep0sexcheck


#We generated a missingness file here with the --missing flag,
#but we are going to do without the sex check.
```

QC Step 1: Identifying Unfiltered Genotyping Rate

```
plink --bfile /home/peter/prostate_cancer/QC_Steps/step0/qcstep0nohh
--missing --out /home/peter/prostate_cancer/QC_Steps/step1/qcstep1
```

QC Step 1A: Plotting Unfiltered Genotyping Rate

```
library(ggplot2)
library(dplyr)
library(data.table)
"%&%" = function(a, b) paste(a, b, sep = "")
my.dir <- "Z://prostate_cancer/QC_Steps/"
lmiss <- fread(my.dir %&% "step1/qcstep1.lmiss", header = T)
hist(lmiss$F_MISS)
```




```
# This creates a histogram of the missingness of the data
# before we filter by genotyping rate.
dim(lmiss)[1]
```

```
## [1] 1199187
```

```
# This tells us the number of SNPs we are working with before
# filtering by genotyping rate
table(lmiss$F_MISS < 0.01)
```

```
##
##   FALSE    TRUE
##  115233 1083954
```

```
table(lmiss$F_MISS < 0.02)
```

```
##
##   FALSE    TRUE
##   86889 1112298
```

```
sum(lmiss$F_MISS < 0.01)/(dim(lmiss)[1])
```

```
## [1] 0.9039074
```

```r
sum(lmiss$F_MISS < 0.02)/(dim(lmiss)[1])
```

```
## [1] 0.9275434
```

```r
# The percent of SNPs have a genotyping call rate of 98%
```

There are 1083954 SNPs that meet a genotyping rate of 0.99 and 1112298 SNPs that meet a genotyping rate of 0.98

QC Step 2: Filtering SNPs by Genotyping Rate

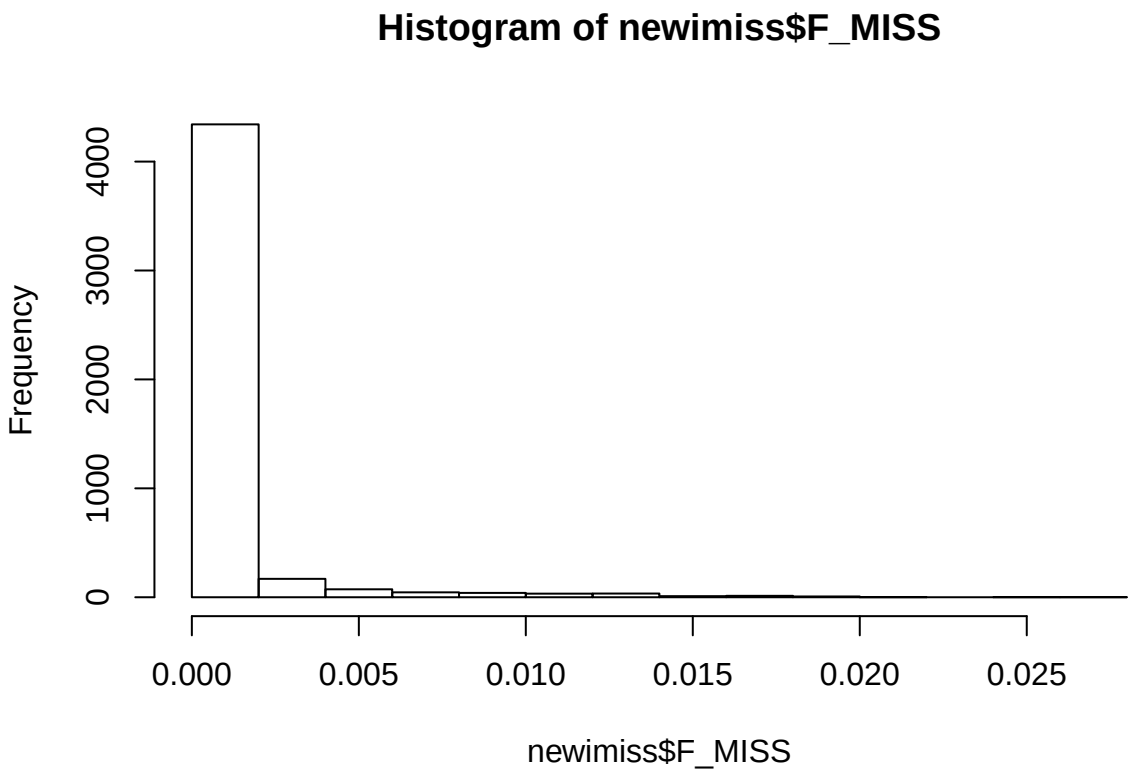
```
plink --bfile /home/peter/prostate_cancer/QC_Steps/step0/qcstep0nohh
--geno 0.01 --make-bed
--out /home/peter/prostate_cancer/QC_Steps/step2/qcstep2
```

QC Step 3: Identifying Filtered Genotyping Rate

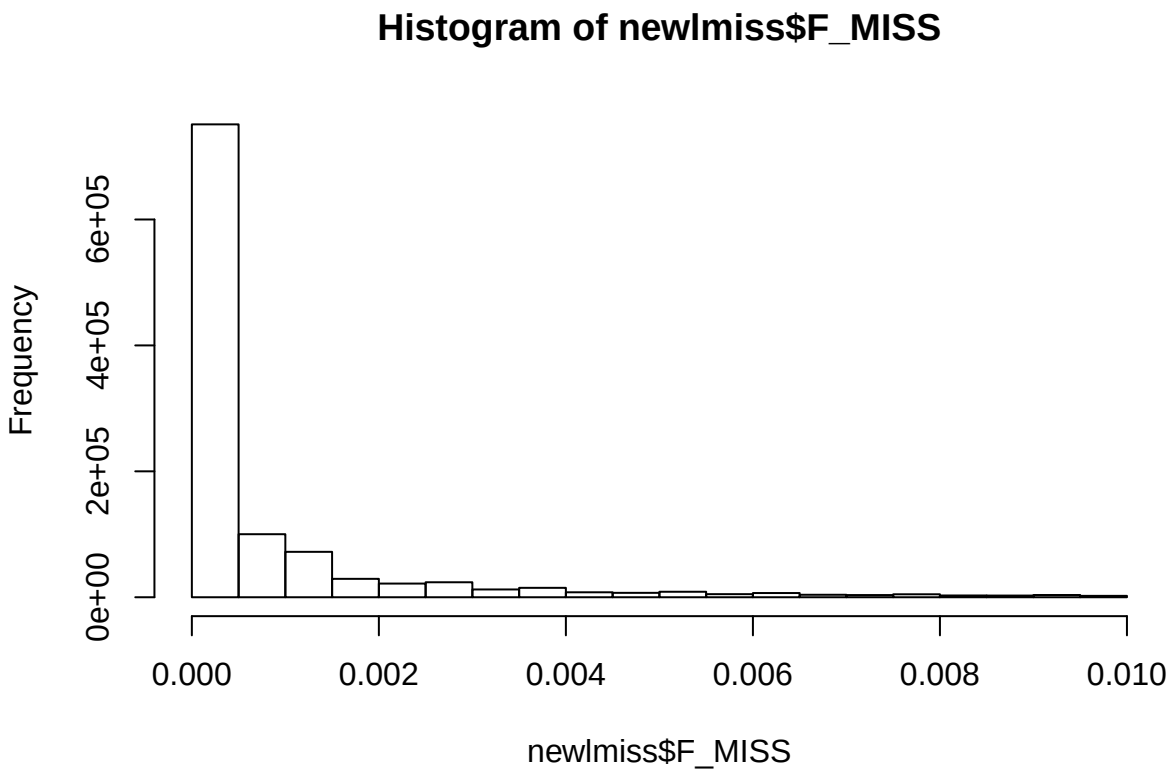
```
plink --bfile /home/peter/prostate_cancer/QC_Steps/step2/qcstep2
--missing --out /home/peter/prostate_cancer/QC_Steps/step3/qcstep3
```

QC Step 3A: Plotting Filtered Genotyping Rate

```r
options(tinytex.verbose = TRUE)
newimiss <- fread(my.dir %&% "step3/qcstep3.imiss")
hist(newimiss$F_MISS)
```




```
newlmiss <- fread(my.dir %&% "step3/qcstep3.lmiss")
hist(newlmiss$F_MISS)
```




```
dim(newlmiss)[1]
```

```
## [1] 1083954
```

QC Step 4: Filtering by HWE

```
plink --bfile /home/peter/prostate_cancer/QC_Steps/step2/qcstep2
--hardy
--out /home/peter/prostate_cancer/QC_Steps/step4/qcstep4
```

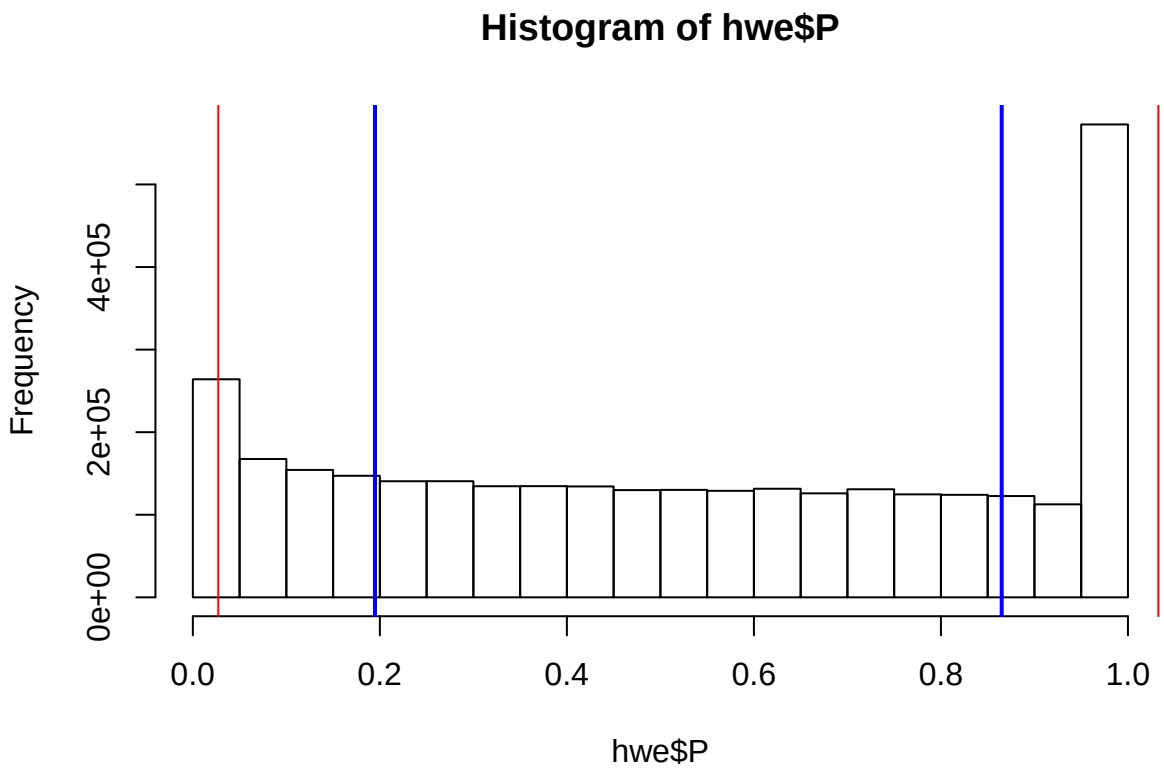
QC Step 4A: Plotting HWE Frequencies and Removing SNPs outside of HWE

We need to remove SNPs that are outside of HWE (P<1e-6)

```
options(tinytex.verbose = TRUE)
hwe <- fread(my.dir %&% "step4/qcstep4.hwe", header = T)
summary(hwe$P)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  0.0000  0.2281  0.5299  0.5324  0.8475  1.0000
```

```
hist(hwe$P)
abline(v = median(hwe$P) + sd(hwe$P), col = "blue", lwd = 2)
abline(v = median(hwe$P) - sd(hwe$P), col = "blue", lwd = 2)
abline(v = median(hwe$P) + 1.5 * sd(hwe$P), col = "red")
abline(v = median(hwe$P) - 1.5 * sd(hwe$P), col = "red")
```




```
table(hwe$P < 1e-06)
```

```
##
##   FALSE    TRUE
## 3238970   12892
```

```
table(hwe$P < 1e-06)/sum(table(hwe$P < 1e-06))
```

```
##
##       FALSE        TRUE
## 0.996035502 0.003964498
```

```
outlierSNPs <- as.data.table(subset(hwe$SNP, hwe$P <= 1e-06))
fwrite(outlierSNPs, my.dir %&% "step4/HWEoutlierSNPstoberemoved.txt",
    col.names = F, row.names = F, sep = " ", quote = F)
```

QC Step 4B: Removing Outlier SNPs

```
plink --bfile /home/peter/prostate_cancer/QC_Steps/step2/qcstep2
--exclude /home/peter/prostate_cancer/QC_Steps/step4/HWEoutlierSNPstoberemoved.txt
--make-bed --out  /home/peter/prostate_cancer/QC_Steps/step4/qcstep4b
```

QC Step 5: IBD Pruning

QC Step 5a: Calculating IBD values

```
plink --bfile /home/peter/prostate_cancer/QC_Steps/step4/qcstep4b
--indep-pairwise 50 5 0.3
--out /home/peter/prostate_cancer/QC_Steps/step5/step5a/QCStep5a
```

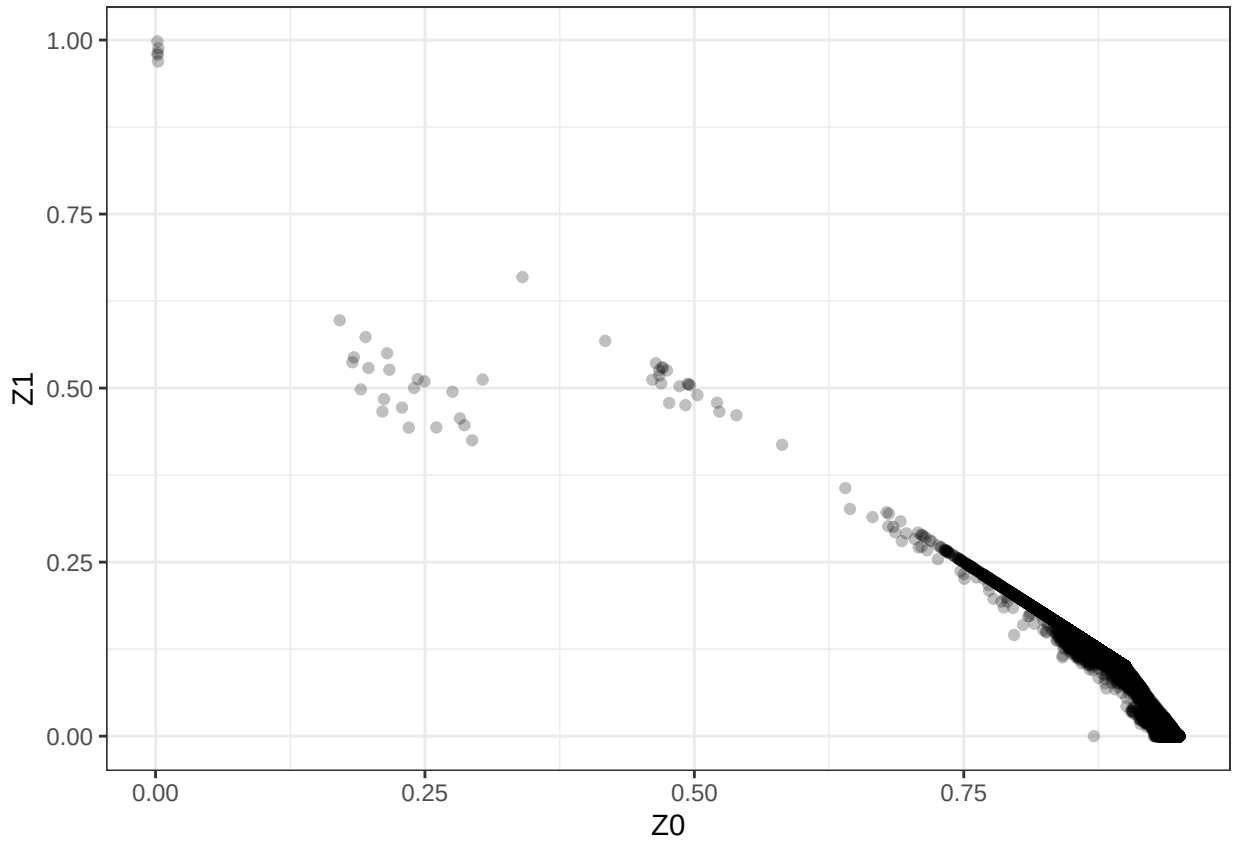
QC Step 5b: Extracting SNPs with excess IBD

```
plink --bfile /home/peter/prostate_cancer/QC_Steps/step4/qcstep4b
--extract /home/peter/prostate_cancer/QC_Steps/step5/step5a/QCStep5a.prune.in
--genome --min 0.05 --out /home/peter/prostate_cancer/QC_Steps/step5/step5b/QCStep5b
```

Initially, I did not have the `--min` flag included in the command above because it was too strict of a filter on my previous neuropsychiatric data. When I tried it this time, I had a file that was 113M lines.

Plotting IBD Values

```
ibd <- fread(my.dir %&% "step5/step5b/QCStep5b.genome", header = T)
ggplot(data = ibd, aes(x = Z0, y = Z1)) + geom_point(alpha = 1/4) +
    theme_bw()
```




```
# We have some parents, siblings, and other related in this
# data that we will need to remove.  For explanation, see
# Figure 4 of Turner et al. Current Protoc Hum Genet (2011).'
# Now we can check for duplicates in the data
dups <- data.frame()
for (i in 1:dim(ibd)[1]) {
    if (as.character(ibd$IID1[i]) == as.character(ibd$IID2[i])) {
        dups <- rbind(dups, ibd[i, ])
    }
}
dim(dups)
```

```
## [1] 0 0
```

```
hapmap <- filter(ibd, grepl("NA", IID1))
# No hapmap individuals. No surpise.
toExclude <- c(as.character(dups$IID1), as.character(hapmap$IID1))
a <- as.character(ibd$IID1) %in% toExclude
others <- ibd[a == FALSE, ]
# Isolating individuals that need to be removed.
toremove <- filter(others, PI_HAT >= 0.2)
write.table(toremove, my.dir %&% "step5/step5b/Relate.to.remove.txt",
    quote = FALSE, row.names = FALSE)
```

QC Step 5C: Identifying individuals with excess heterozygosity

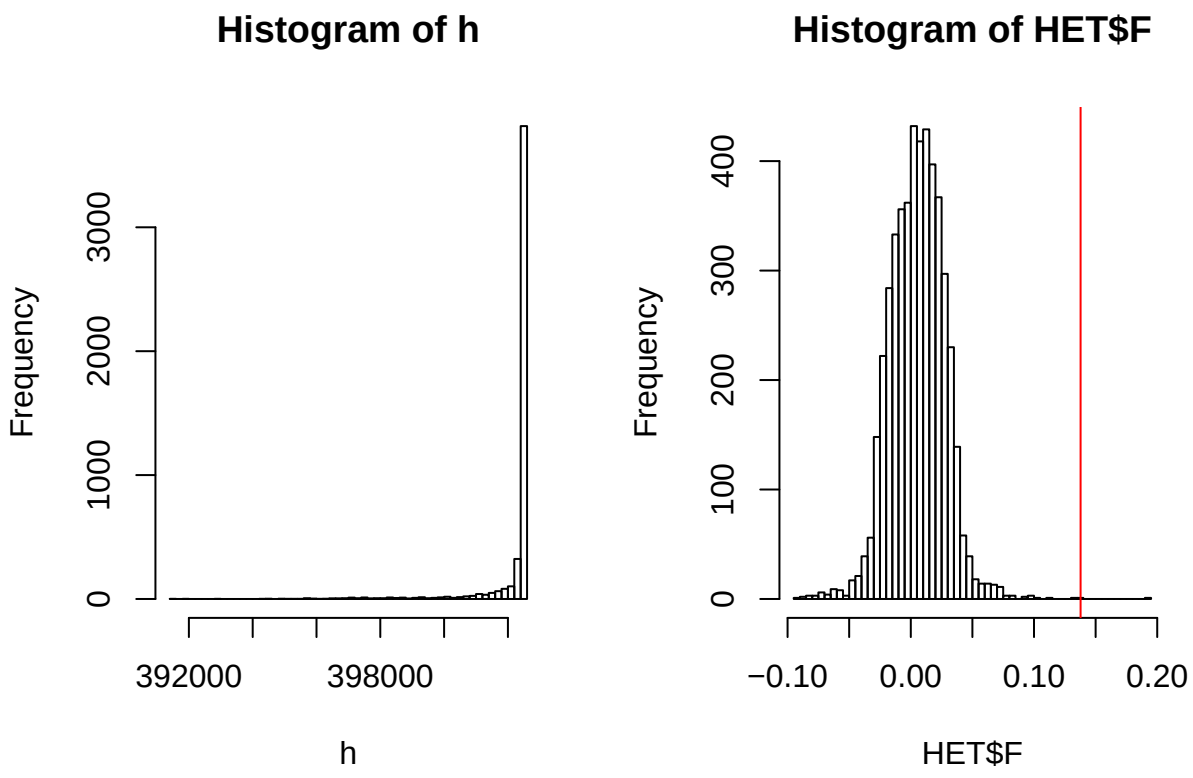
```
plink --bfile /home/peter/prostate_cancer/QC_Steps/step4/qcstep4b
--extract /home/peter/prostate_cancer/QC_Steps/step5/step5a/QCStep5a.prune.in
--het --out QCStep5c
```

Plotting Heterozygosity Data

```
HET <- fread(my.dir %&% "step5/step5c/QCStep5c.het", header = T)
h = HET$"N(NM)" - HET$"O(HOM)"/HET$"N(NM)"
oldpar = par(mfrow = c(1, 2))
hist(h, 50)
hist(HET$F, 50)
summary(HET$F)
```

```
##      Min.   1st Qu.    Median      Mean   3rd Qu.      Max.
## -0.093250 -0.009631  0.005951  0.005602  0.020400  0.190600
```

```
abline(v = mean(HET$F) + 6 * sd(HET$F), col = "red")
abline(v = mean(HET$F) - 6 * sd(HET$F), col = "red")
```



```r
sortHET <- HET[order(HET$F), ]
outliers <- data.table()

for (i in 1:length(sortHET$F)) {
    if (sortHET[i, 6] > (mean(sortHET$F) + 3 * sd(sortHET$F))) {
        outliers <- rbind(outliers, sortHET[i, ])
    }
    if (sortHET[i, 6] < (mean(sortHET$F) - 3 * sd(sortHET$F))) {
        outliers <- rbind(outliers, sortHET[i, ])
    }
}

hetoutliers <- select(outliers, FID, IID)
dim(hetoutliers)  #This tells us how many outliers there are.
```

```
## [1] 50  2
```

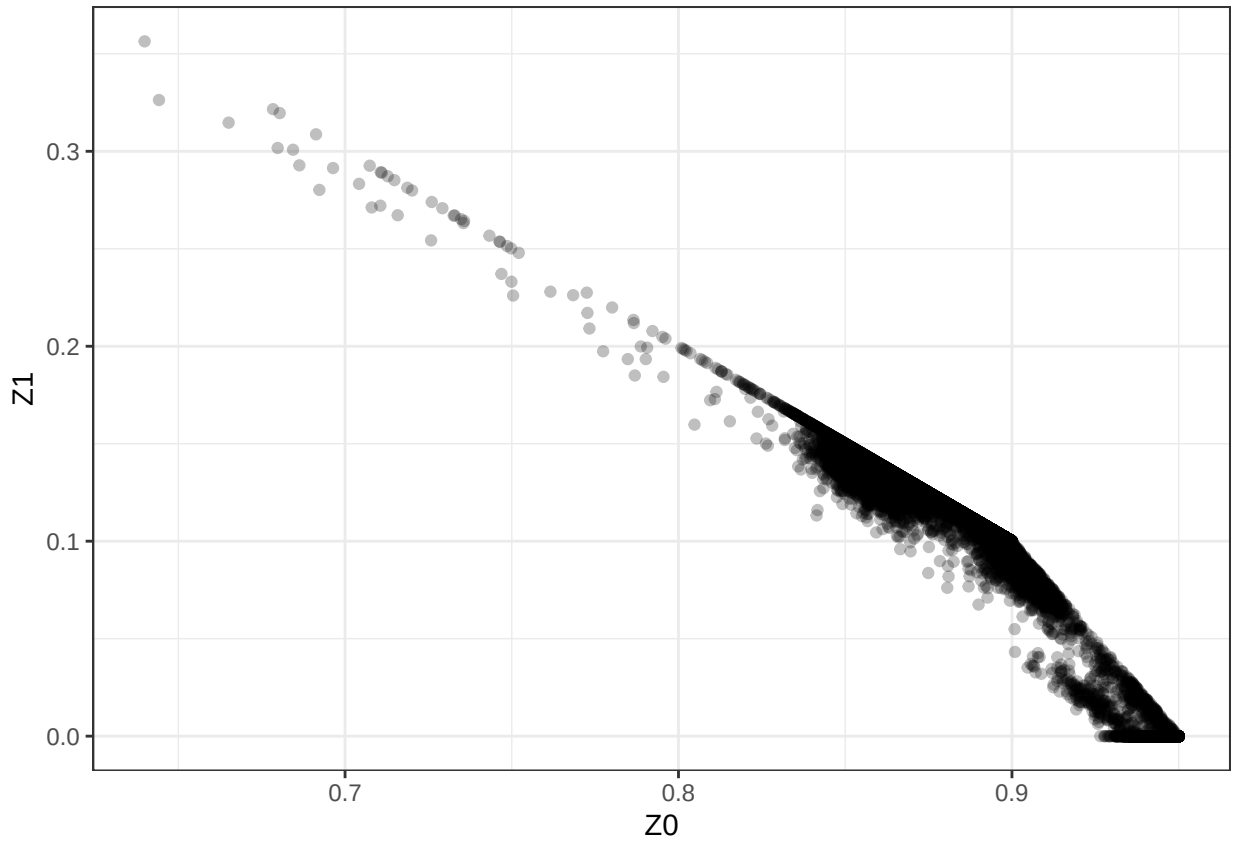
```r
fwrite(hetoutliers, "Z://prostate_cancer/QC_Steps/step5/step5c/hetoutliers.txt",
    quote = F, col.names = F, row.names = F, sep = " ")
```

QC Step 5D: Removing Heterozygosity Outliers

```
plink --bfile /home/peter/prostate_cancer/QC_Steps/step4/qcstep4b
--remove /home/peter/prostate_cancer/QC_Steps/step5/step5c/hetoutliers.txt
--extract /home/peter/prostate_cancer/QC_Steps/step5/step5a/QCStep5a.prune.in
--make-bed --out /home/peter/prostate_cancer/QC_Steps/step5/step5d/QCStep5d

plink --bfile /home/peter/prostate_cancer/QC_Steps/step5/step5d/QCStep5d
--extract /home/peter/prostate_cancer/QC_Steps/step5/step5a/QCStep5a.prune.in
--remove /home/peter/prostate_cancer/QC_Steps/step5/step5b/Relate.to.remove.txt
--genome --min 0.05 --out QCStep5D
```

Plotting IBD Filtered Data:

```r
options(tinytex.verbose = TRUE)
IBD <- fread(my.dir %&% "step5/step5d/QCStep5D.genome", header = T)
ggplot(data = IBD, aes(x = Z0, y = Z1)) + geom_point(alpha = 1/4) +
    theme_bw()
```



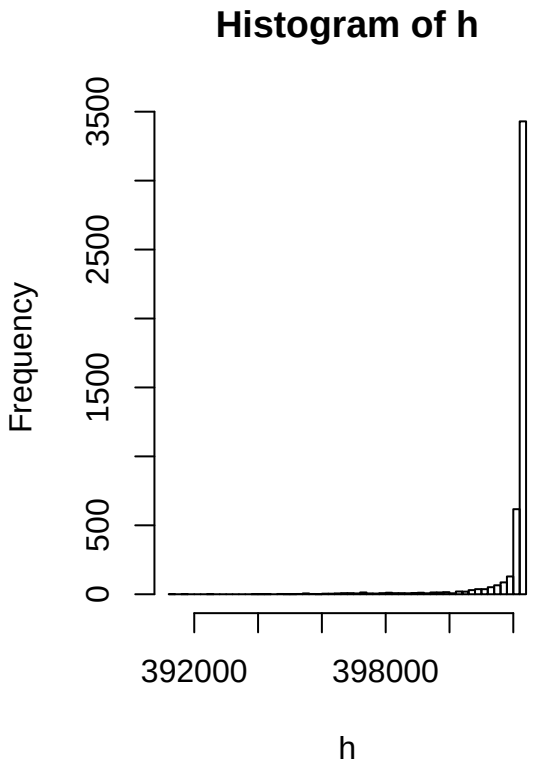
QC Step 5E: Second Heterozygosity Check

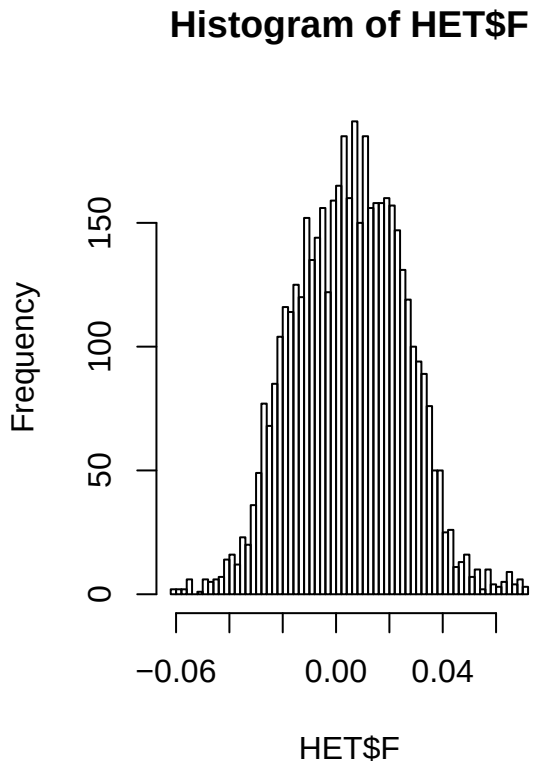
```
plink --bfile /home/peter/prostate_cancer/QC_Steps/step5/step5d/QCStep5d
--het --out /home/peter/prostate_cancer/QC_Steps/step5/step5e/QCStep5e
```

```
options(tinytex.verbose = TRUE)
HET <- fread(my.dir %&% "/step5/step5e/QCStep5.het", header = T)
h = HET$"N(NM)" - HET$"O(HOM)"/HET$"N(NM)"
oldpar = par(mfrow = c(1, 2))
hist(h, 50)
hist(HET$F, 50)
summary(HET$F)
```

```
##      Min.   1st Qu.    Median      Mean   3rd Qu.      Max.
## -0.060470 -0.009887  0.005533  0.005187  0.019810  0.071000
```

```
abline(v = mean(HET$F) + 6 * sd(HET$F), col = "red")
abline(v = mean(HET$F) - 6 * sd(HET$F), col = "red")
```






```
sortHET <- HET[order(HET$F), ]
outliers <- data.table()

for (i in 1:length(sortHET$F)) {
    if (sortHET[i, 6] > (mean(sortHET$F) + 3 * sd(sortHET$F))) {
        outliers <- rbind(outliers, sortHET[i, ])
    }
    if (sortHET[i, 6] < (mean(sortHET$F) - 3 * sd(sortHET$F))) {
        outliers <- rbind(outliers, sortHET[i, ])
    }
}

hetoutliers <- select(outliers, FID, IID)
dim(hetoutliers)
```

```
## [1] 18  2
```

```
# These outliers are individuals from the data that was after
# we removed the initial outliers.  This would be too
# stringent to remove these outliers.
```

QC Step 5F

```
plink --bfile /home/peter/prostate_cancer/QC_Steps/step5/step5d/QCStep5d
```

```
--extract /home/peter/prostate_cancer/QC_Steps/step5/step5a/QCStep5a.prune.in
--remove /home/peter/prostate_cancer/QC_Steps/step5/step5b/Relate.to.remove.txt
--make-bed --out /home/peter/prostate_cancer/QC_Steps/step5/step5f/QCStep5f
```

QC Step 6: Principal Component Analysis

QC Step 6A: Merge with HapMap

```
plink --bfile /home/peter/prostate_cancer/QC_Steps/step5/step5f/QCStep5f
--bmerge /home/wheelerlab1/Data/HAPMAP3_hg18/HM3_ASN_CEU_YRI_Unrelated_hg18_noAmbig
--make-bed
--out /home/peter/prostate_cancer/QC_Steps/step6/step6a/step6a
```

QC Step 6B: Exclude Missing SNPs or SNPs with +3 Alleles

```
plink
--bfile /home/wheelerlab1/Data/HAPMAP3_hg18/HM3_ASN_CEU_YRI_Unrelated_hg18_noAmbig
--exclude /home/peter/prostate_cancer/QC_Steps/step6/step6a/step6a-merge.missnp
--make-bed
--out /home/peter/prostate_cancer/QC_Steps/step6/step6b/step6b
```

QC Step6C: Merge Attempt 2 with Excluded SNPs

```
plink --bfile /home/peter/prostate_cancer/QC_Steps/step5/step5f/QCStep5f
--bmerge /home/peter/prostate_cancer/QC_Steps/step6/step6b/step6b
--out /home/peter/prostate_cancer/QC_Steps/step6/step6c/step6c
```

QC Step6D: Run PCA

```
plink --bfile /home/peter/prostate_cancer/QC_Steps/step6/step6c/step6c
--geno 0.01 --maf 0.05 --chr 1-22 --pca 10 --out QCStep6D_PCA
```

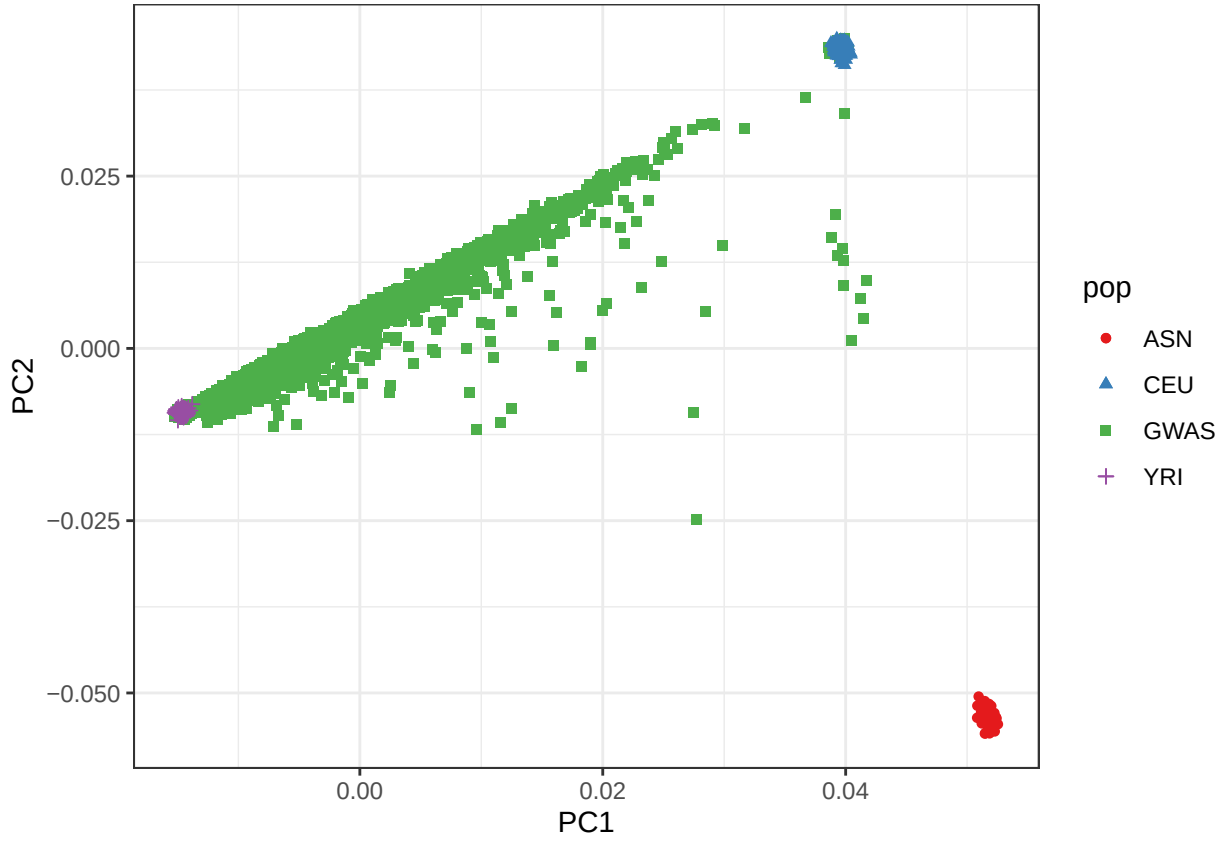
```r
options(tinytex.verbose = TRUE)

hapmappopinfo <- read.table(my.dir %&% "step6/pop_HM3_hg18_forPCA.txt") %>%
    select(V1, V3)
colnames(hapmappopinfo) <- c("pop", "IID")

fam <- fread(my.dir %&% "step6/step6c/step6c.fam", header = F) %>%
    select(V1, V2)
colnames(fam) <- c("FID", "IID")

popinfo <- left_join(fam, hapmappopinfo, by = "IID")
```

```
## Warning: Column `IID` joining character vector and factor, coercing into
## character vector
```

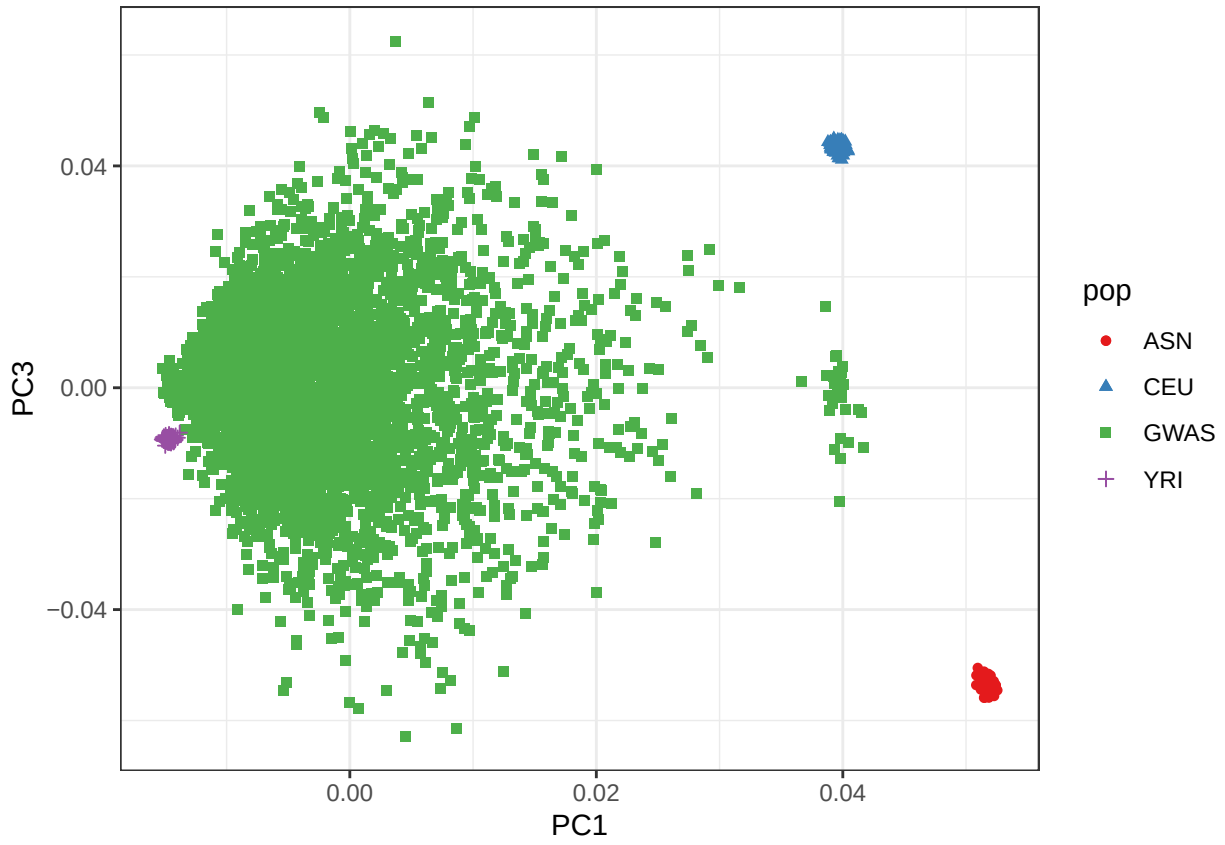
```r
popinfo <- mutate(popinfo, pop = ifelse(is.na(pop), "GWAS", as.character(pop)))
table(popinfo$pop)
```

```
##
##  ASN  CEU GWAS  YRI
##  170  111 4674  110
```

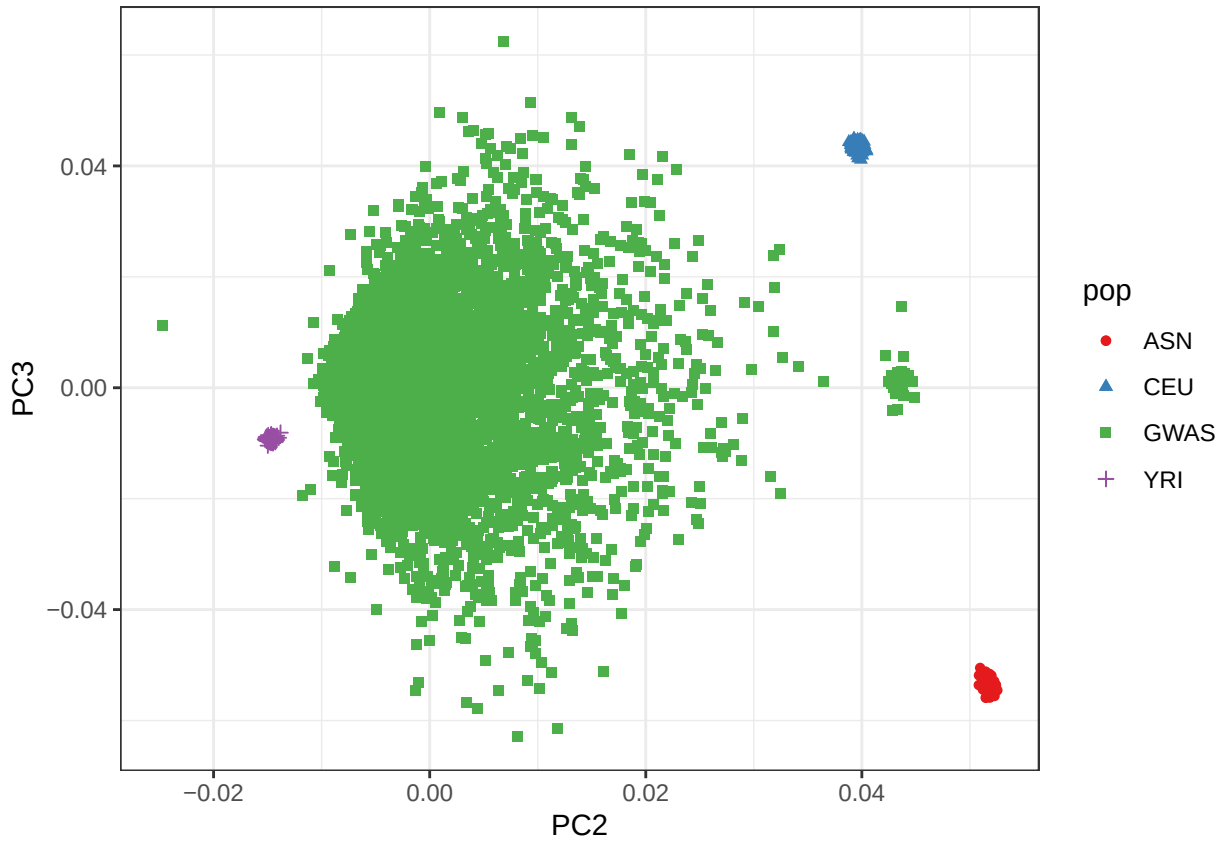
```
pcs <- read.table(my.dir %&% "step6/QCStep6D_PCA.eigenvec")
pcdf <- data.frame(popinfo, pcs[, 3:12]) %>% rename(PC1 = V3,
    PC2 = V4, PC3 = V5, PC4 = V6, PC5 = V7, PC6 = V8, PC7 = V9,
    PC8 = V10, PC9 = V11, PC10 = V12)
gwas <- filter(pcdf, pop == "GWAS")
hm3 <- filter(pcdf, grepl("NA", IID))
eval <- scan(my.dir %&% "step6/QCStep6D_PCA.eigenval")[1:10]
pve <- eval/sum(eval)  #Calculate the percent explained by each PC
PCs <- c(1:10)
PVE <- data.table(PCs, pve)

ggplot() + geom_point(data = gwas, aes(x = PC1, y = PC2, col = pop,
    shape = pop)) + geom_point(data = hm3, aes(x = PC1, y = PC2,
    col = pop, shape = pop)) + theme_bw() + scale_colour_brewer(palette = "Set1")
```



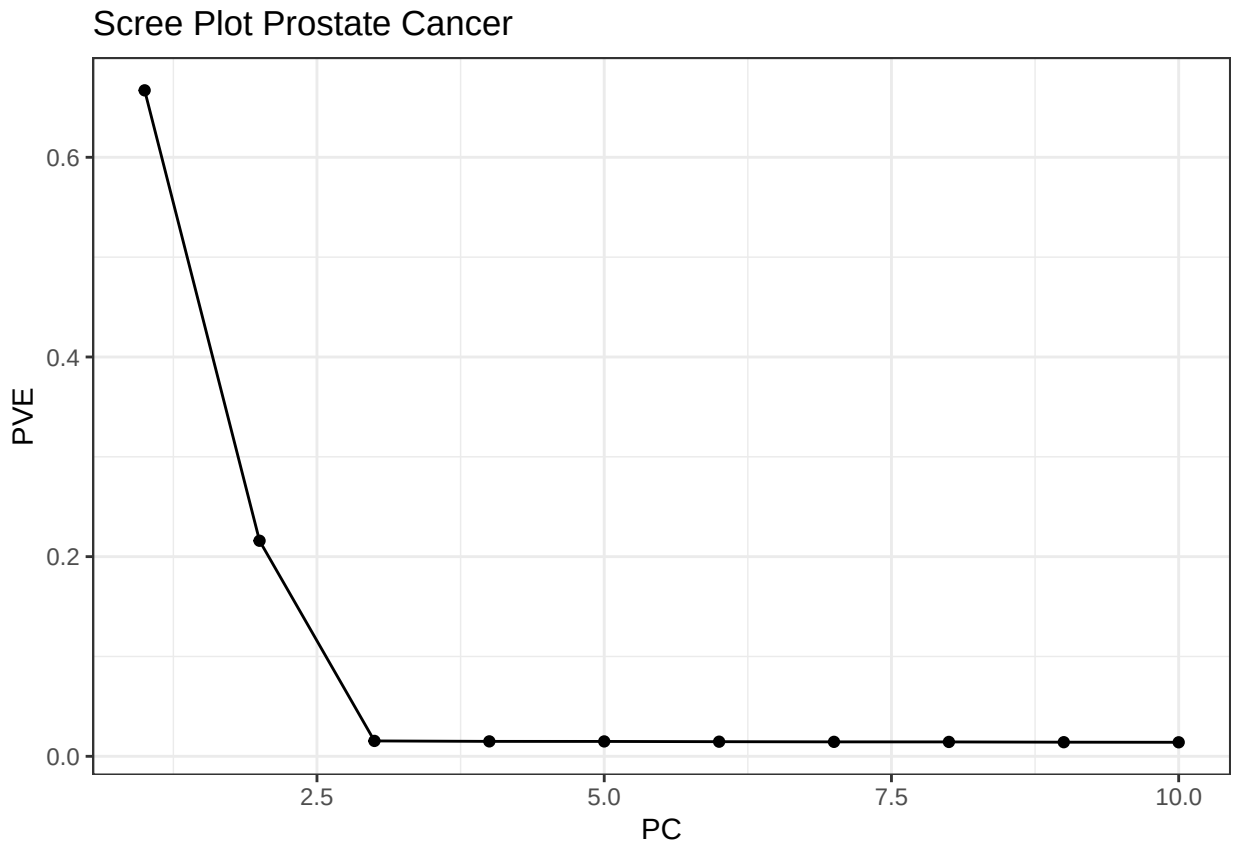

```
ggplot() + geom_point(data = gwas, aes(x = PC1, y = PC3, col = pop,
    shape = pop)) + geom_point(data = hm3, aes(x = PC1, y = PC2,
    col = pop, shape = pop)) + theme_bw() + scale_colour_brewer(palette = "Set1")
```




```
ggplot() + geom_point(data = gwas, aes(x = PC2, y = PC3, col = pop,
    shape = pop)) + geom_point(data = hm3, aes(x = PC1, y = PC2,
    col = pop, shape = pop)) + theme_bw() + scale_colour_brewer(palette = "Set1")
```




```
ggplot(data = PVE, aes(y = pve, x = PCs)) + geom_point() + geom_line() +
    xlab("PC") + ylab("PVE") + ggtitle("Scree Plot Prostate Cancer") +
    theme_bw()
```



Next Steps

Lift Over

Right now, the data is in genome build hg18. We need to lift it over to hg19. A good example of the liftover process can be found at https://github.com/WheelerLab/Neuropsychiatric-Phenotypes/blob/master/SCZ-BD_Px/1_hg18tohg19liftover.md. When we perform the liftover, we will use `home/peter/prostate_cancer/QC_Steps/step4/qcstep4b` since this set of files includes unpruned data with HWE outliers removed.

Imputation

After liftover, we will upload the data to the University of Michigan Imputation Server. The imputed data will then be filtered to remove SNPs with $r^2 < 0.8$ and MAF < 0.01