Quantitative Genomics and Genetics 2018 Project

Darya Akimova

May 8, 2018

```
# is there any missing data?
anyNA(list(geno, pheno, covars, snp_info, gene_info))
```

```
[1] FALSE
```

```
# are there approximately equal numbers of people in each covariate group?
table(covars$Population)
```

```
CEU FIN GBR TSI
 78  89  85  92
```

```
table(covars$Sex)
```

```
FEMALE   MALE
   181    163
```

```
# does the coding of the genotypes makes sense across all the data?
max(as.matrix(geno))
```

```
[1] 2
```

```
min(as.matrix(geno))
```

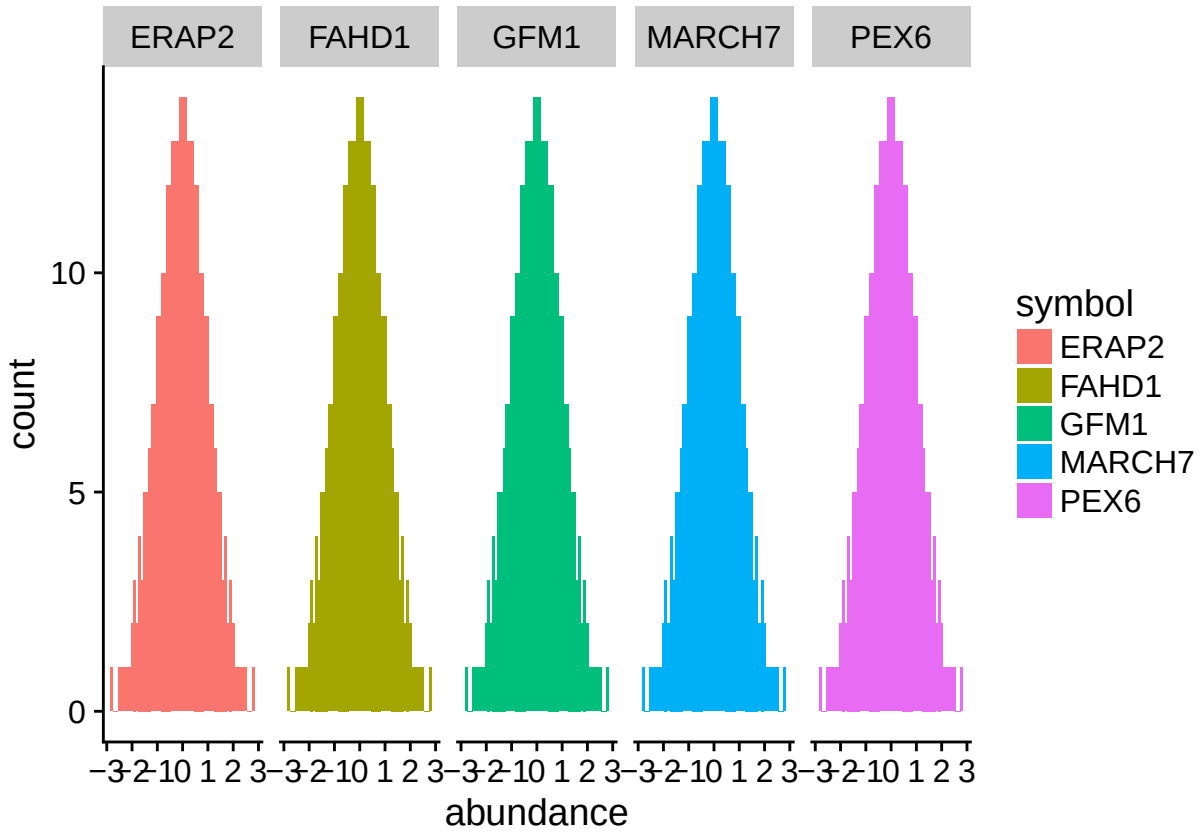
```
[1] 0
```

```
# calculate allele frequency
geno_sums <- map_dbl(geno, function(x) sum(x) / (nrow(geno) * 2))
# any genotypes need to be removed because of MAF < 5%?
any(geno_sums < 0.05 | geno_sums > 0.95)
```

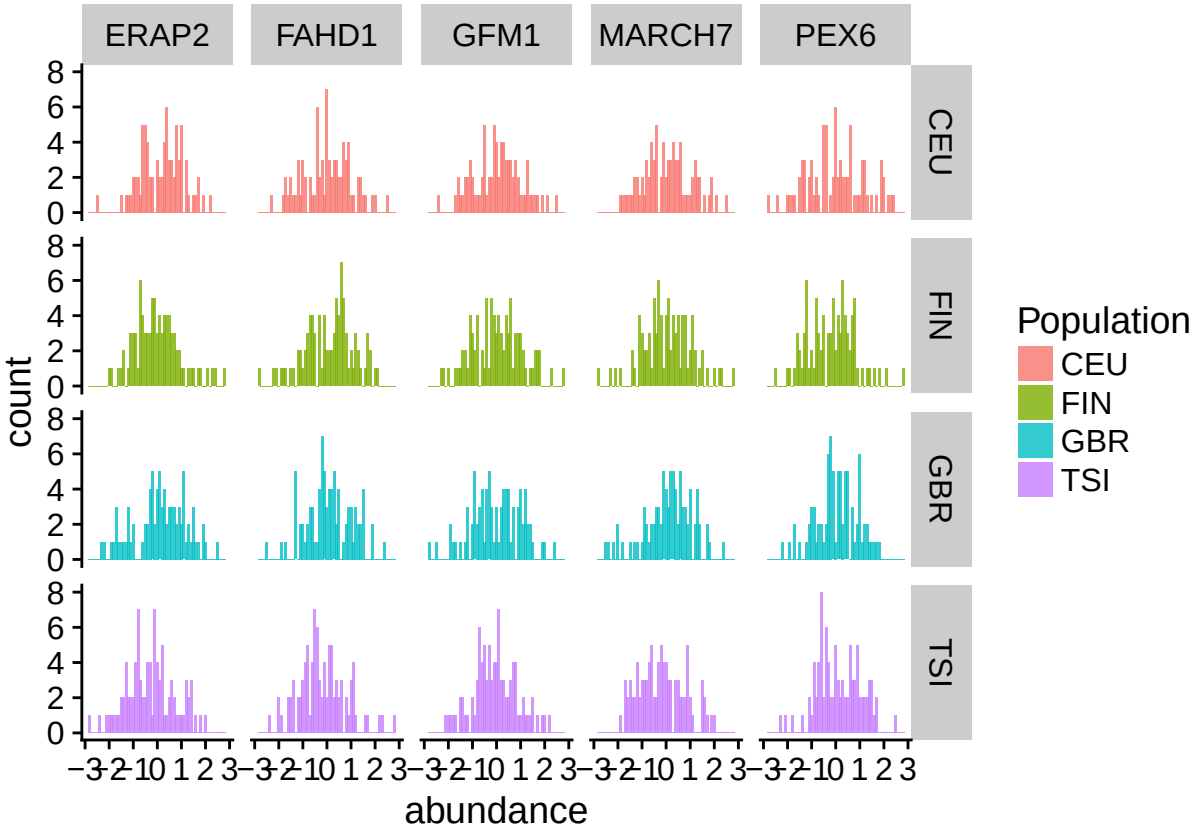
```
[1] FALSE
```

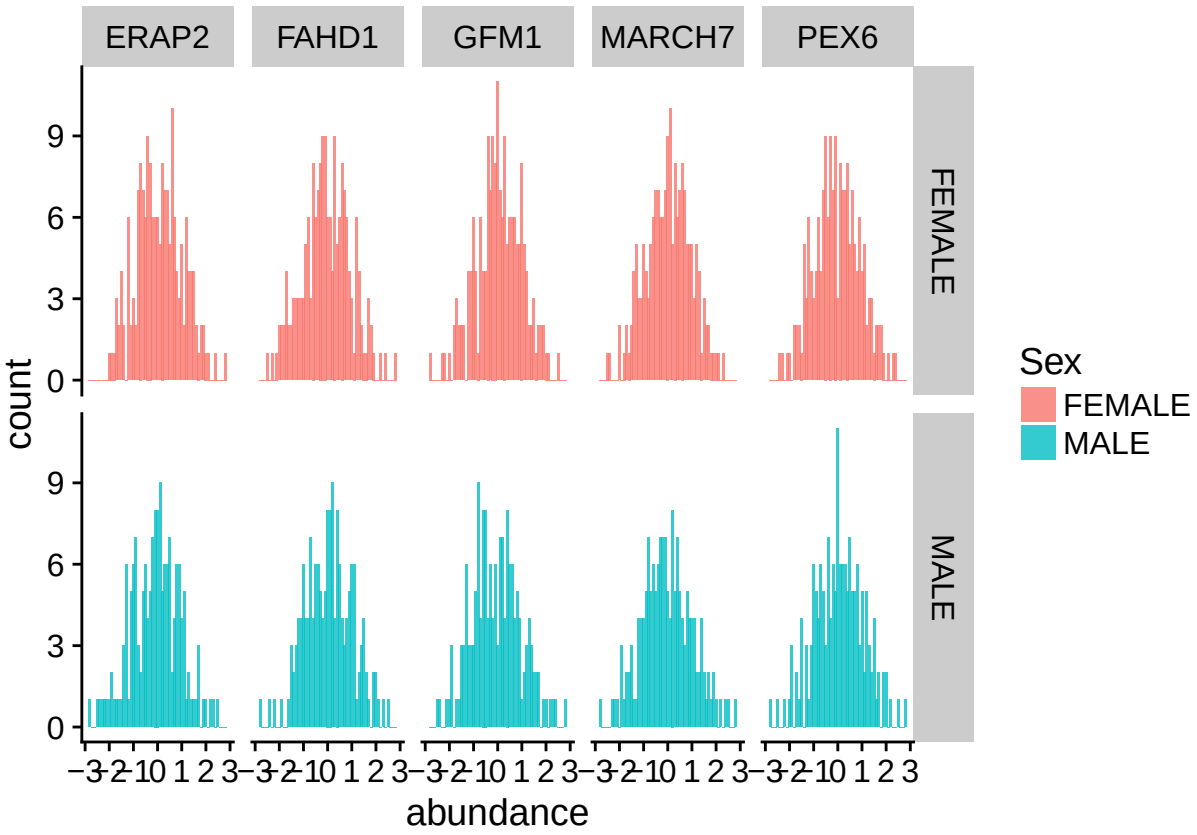
```
# no genotypes need to be removed because of MAF < 5%
# are there any genotypes without associated phenotypes, or vice versa?
all.equal(rownames(geno), rownames(pheno))
```

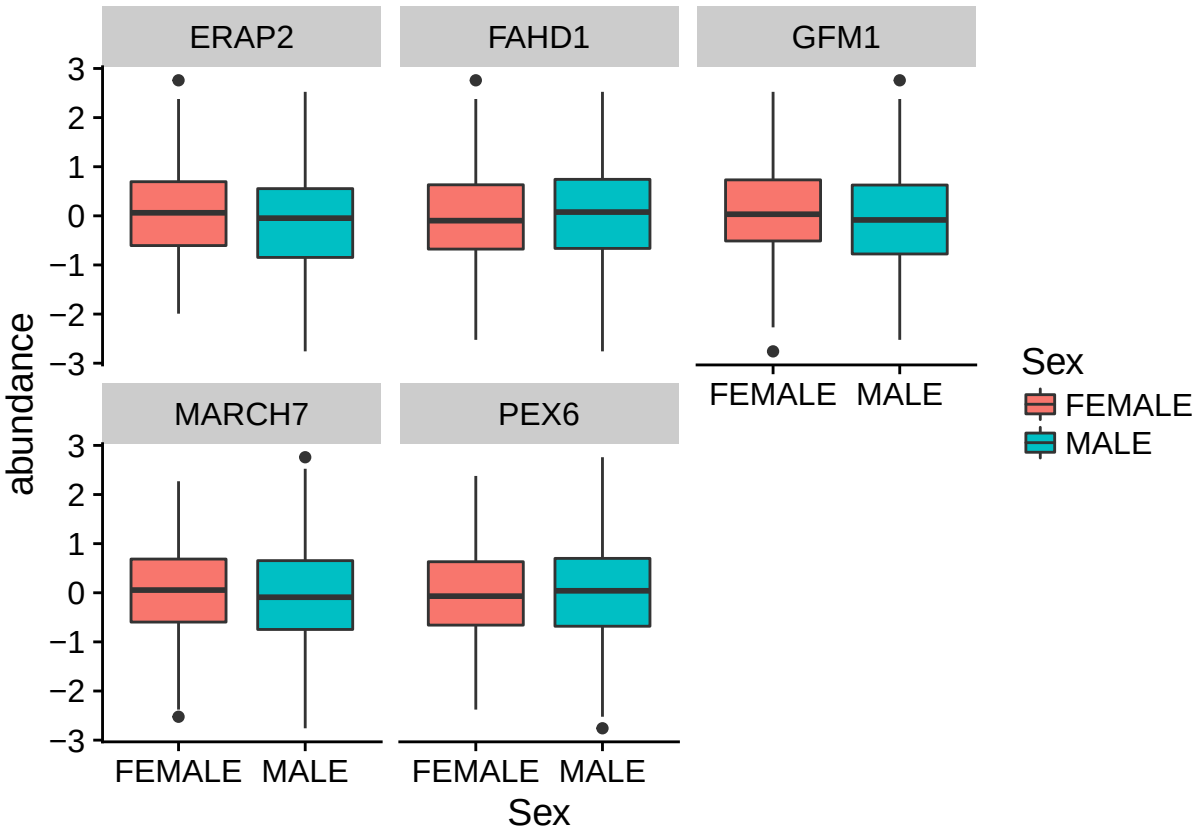
```
[1] TRUE
```

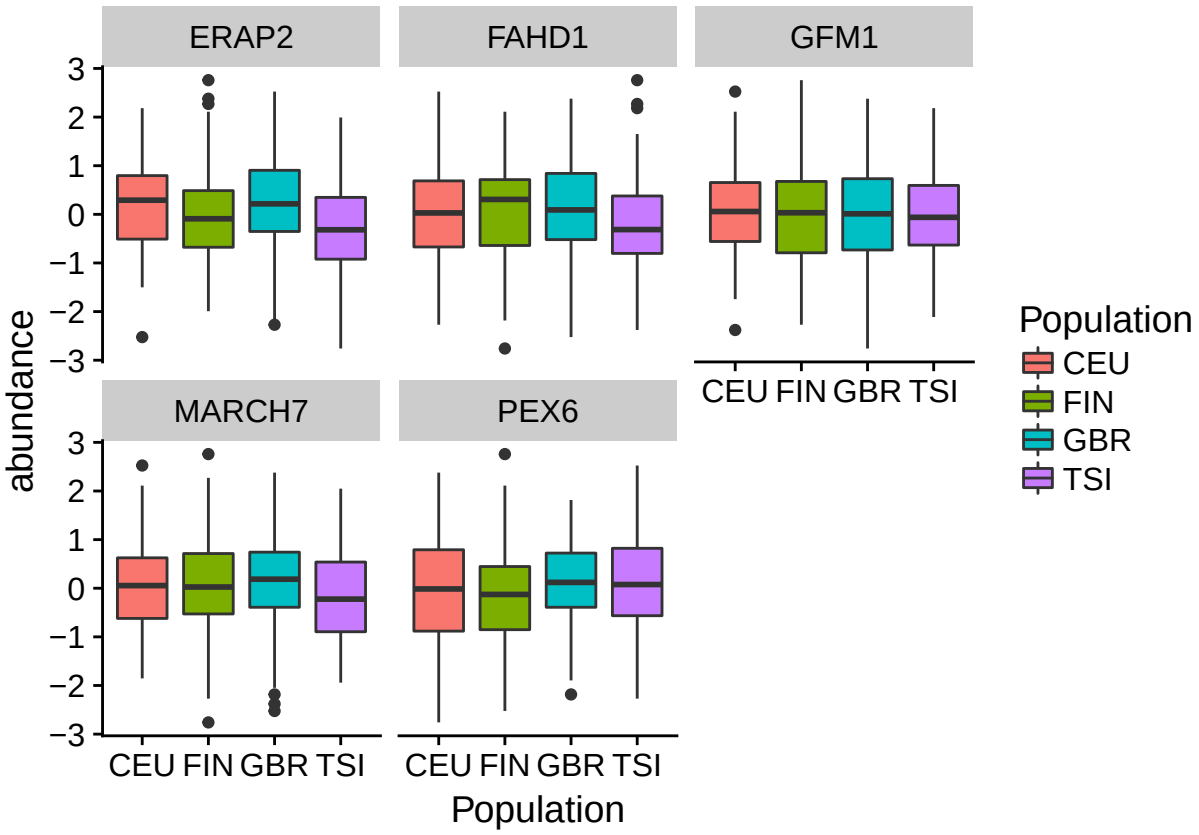


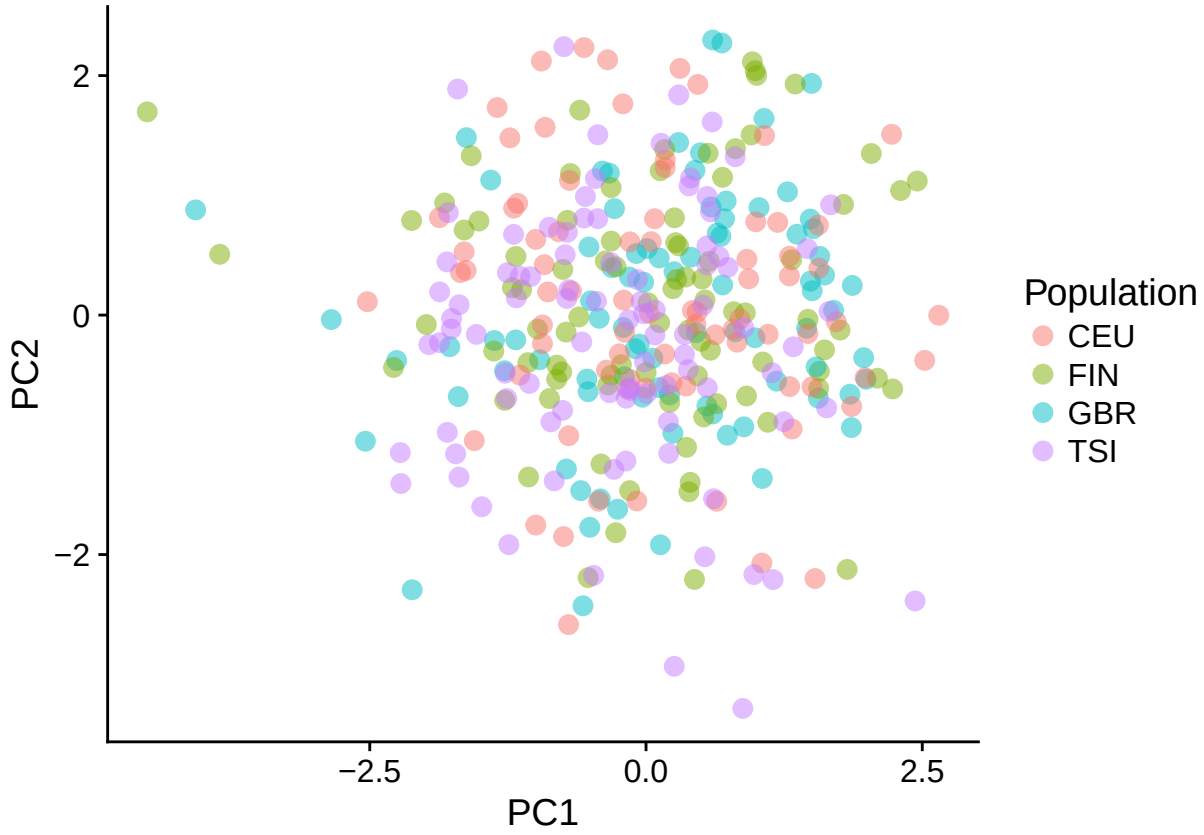
Phenotype plots:

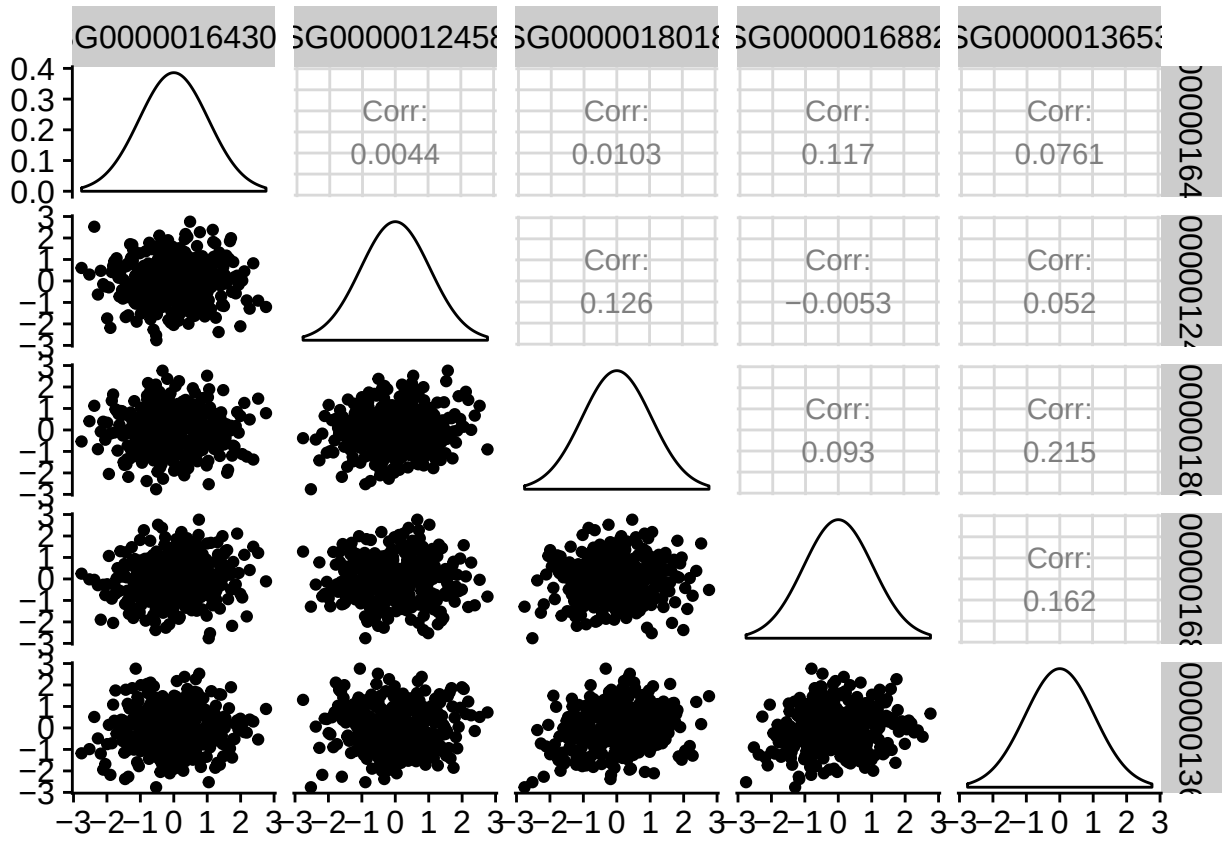

ERAP2
FAHD1
GFM1
MARCH7
PEX6
CEU
FIN
GBR
TSI
count
abundance
Population
CEU
FIN
GBR
TSI


ERAP2
FAHD1
GFM1
MARCH7
PEX6
FEMALE
MALE
count
abundance
Sex
FEMALE
MALE


ERAP2
FAHD1
GFM1
MARCH7
PEX6
abundance
3
2
1
0
−1
−2
−3
FEMALE
MALE
Sex
Sex
FEMALE
MALE

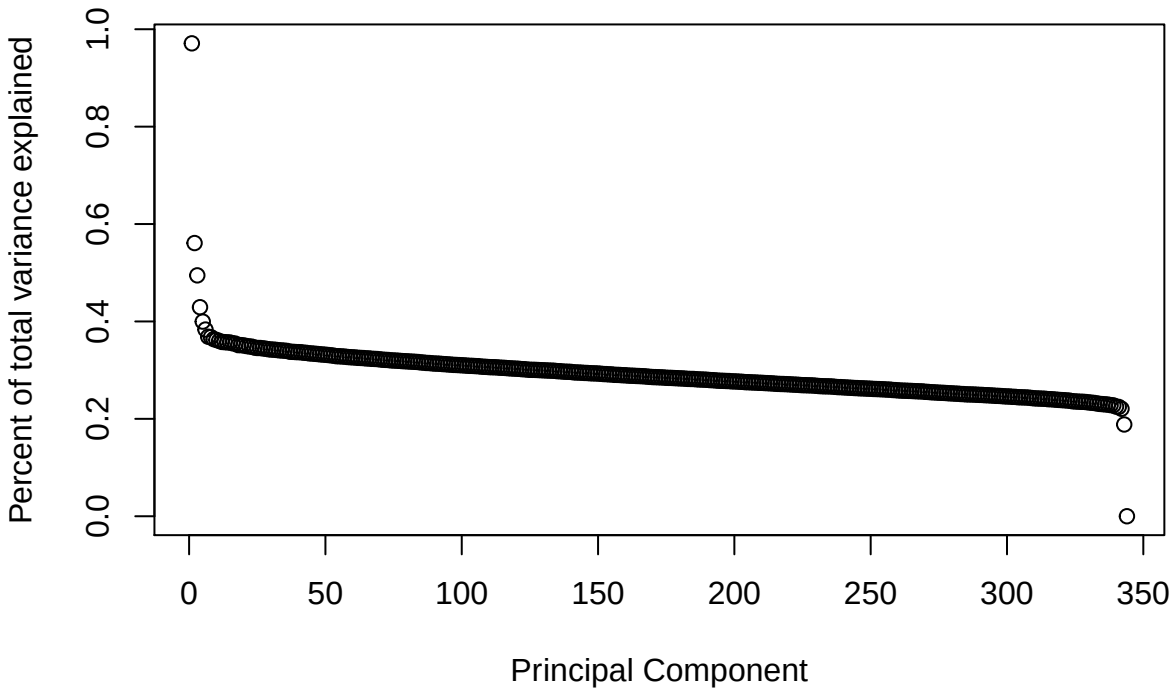

ERAP2
FAHD1
GFM1
MARCH7
PEX6
abundance
3
2
1
0
−1
−2
−3
CEU FIN GBR TSI
Population
Population
CEU
FIN
GBR
TSI


2
0
−2
PC2
−2.5
0.0
2.5
PC1
Population
CEU
FIN
GBR
TSI

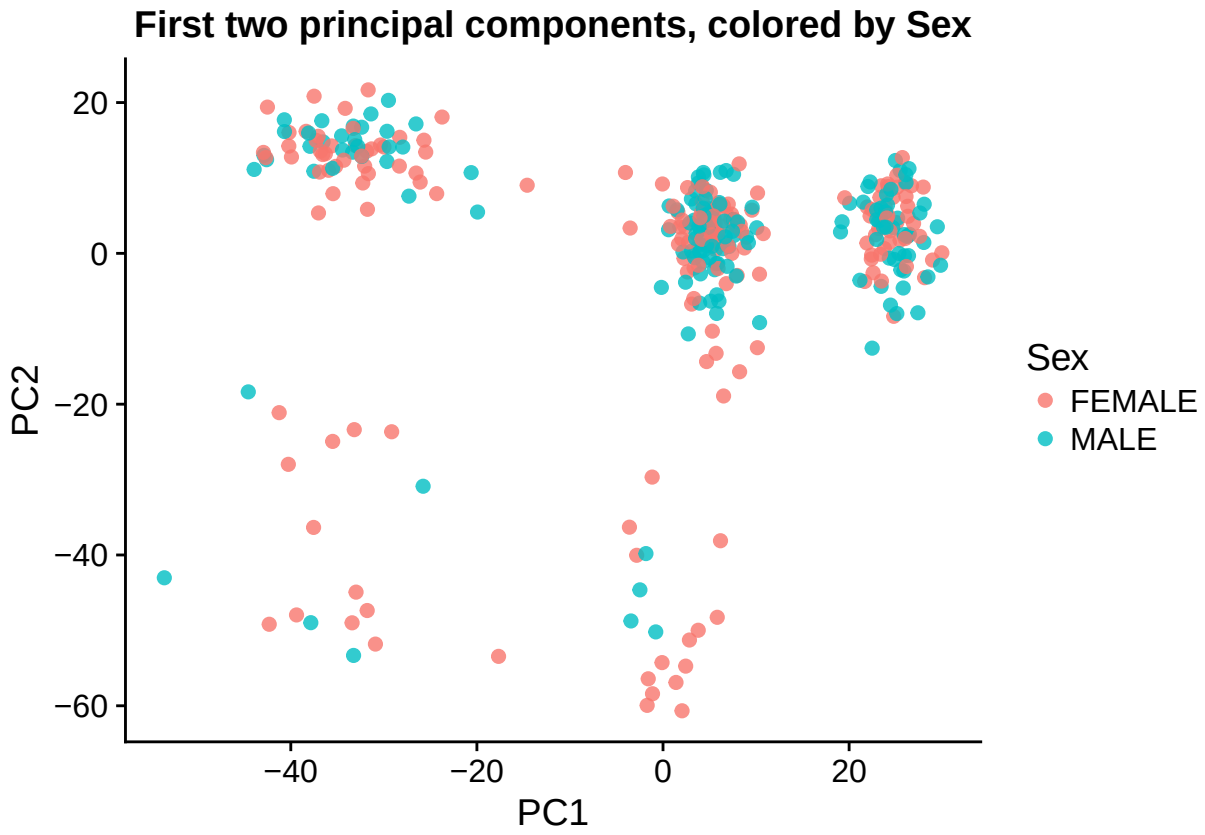


Genotype plots:

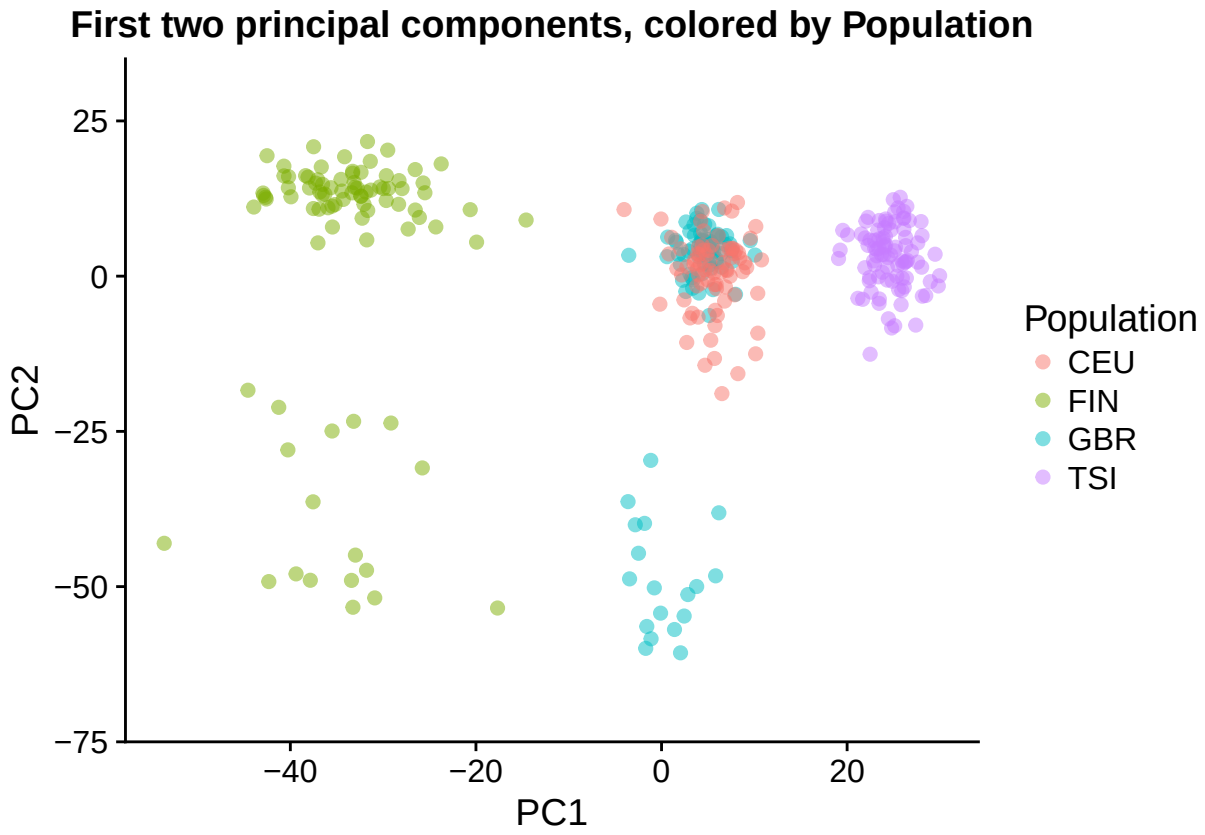
```
# PCA analysis of genetic data
geno_pca <- prcomp(geno, center = TRUE, scale. = TRUE)
plot((geno_pca$sdev^2/ sum(geno_pca$sdev^2)) * 100, xlab = "Principal Component", ylab = "Percent of to
```



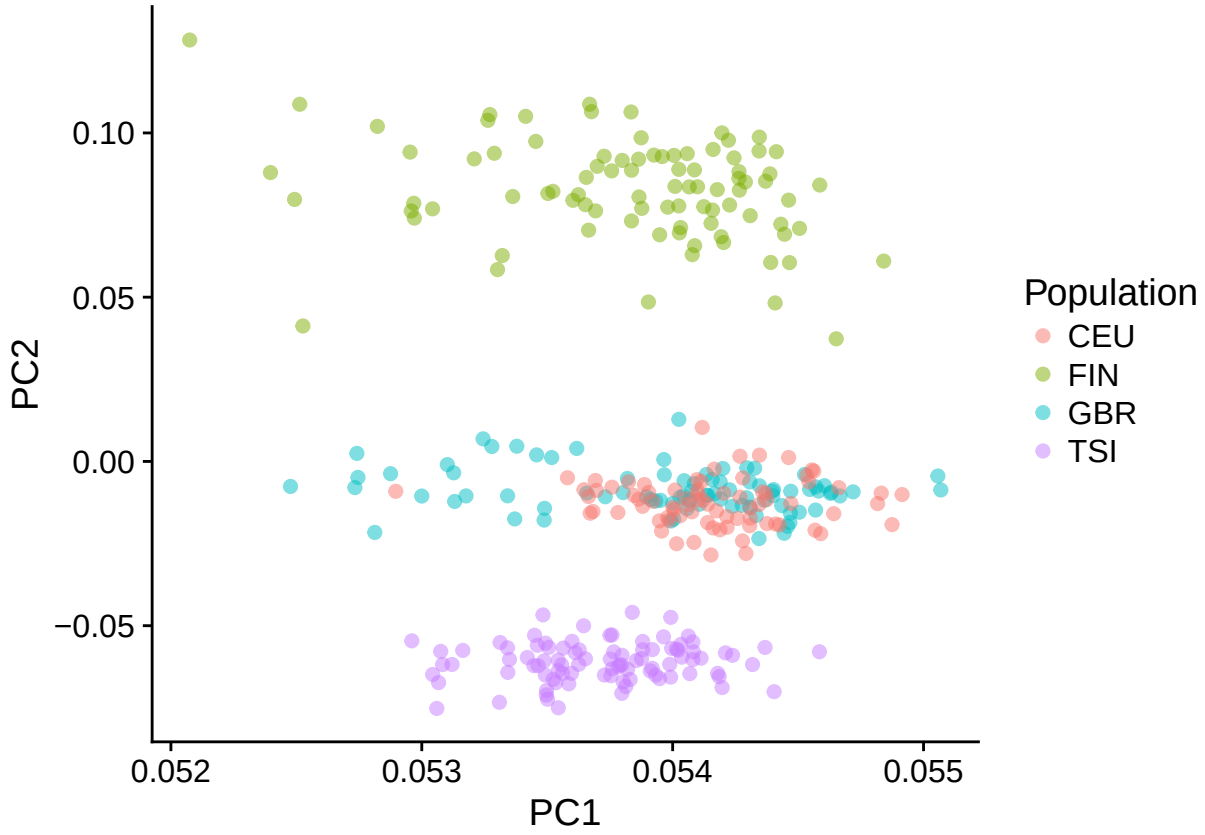

```
geno_pca_x <- data.frame(geno_pca$x)
geno_pca_x %>%
  rownames_to_column("sample") %>%
  left_join(covars %>% rownames_to_column("sample"), by = "sample") %>%
  ggplot(aes(x = PC1, y = PC2, color = Sex)) +
    geom_point(size = 2, alpha = 0.8) +
    ggtitle("First two principal components, colored by Sex")
```




```
geno_pca_x %>%
  rownames_to_column("sample") %>%
  left_join(covars %>% rownames_to_column("sample"), by = "sample") %>%
  ggplot(aes(x = PC1, y = PC2, color = Population)) +
    geom_point(size = 2, alpha = 0.5) +
    ggtitle("First two principal components, colored by Population") +
  ylim(-70, 30)
```

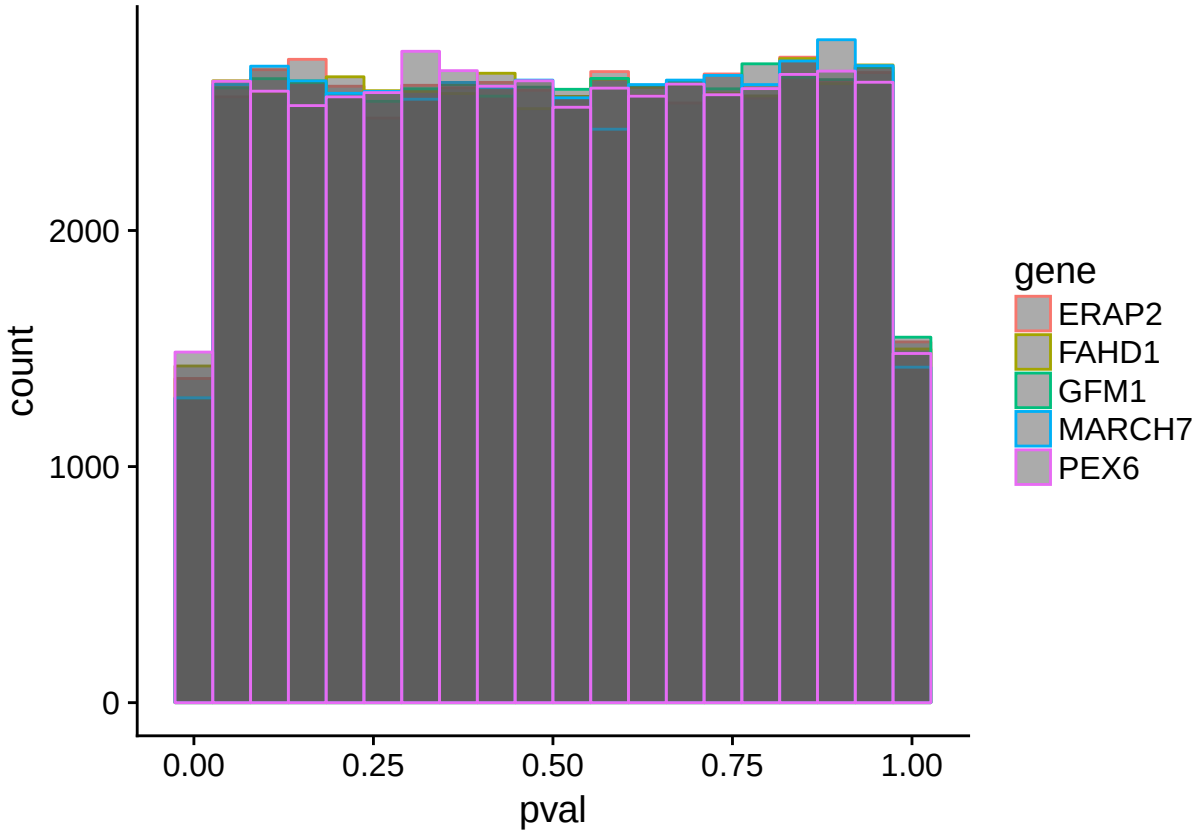


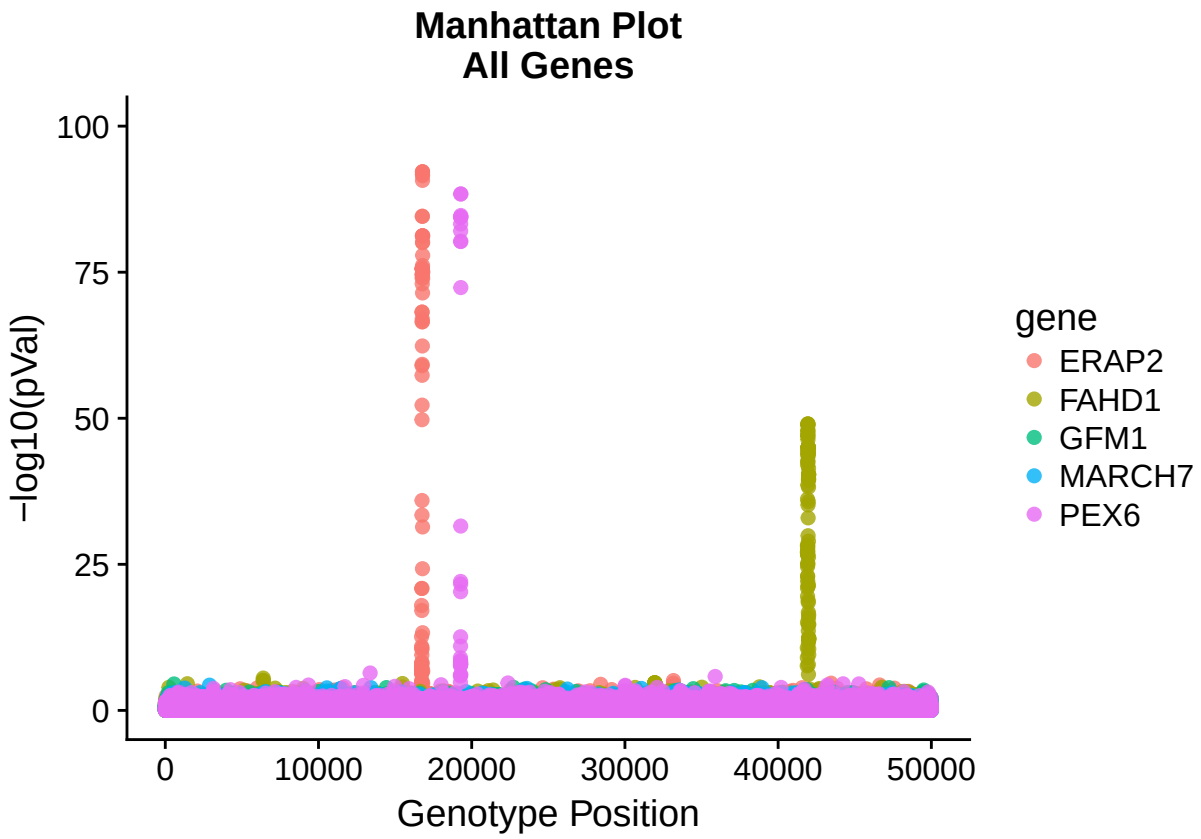

```
indv_pca <- prcomp(t(geno), center = TRUE, scale. = TRUE)
indv_pca_x <- as.data.frame(indv_pca$rotation) %>%
  rownames_to_column("sample") %>%
  left_join(covars %>% rownames_to_column("sample"), by = "sample")
indv_pca_x %>%
  ggplot(aes(x = PC1, y = PC2, color = Population)) +
  geom_point(size = 2, alpha = 0.5)
```



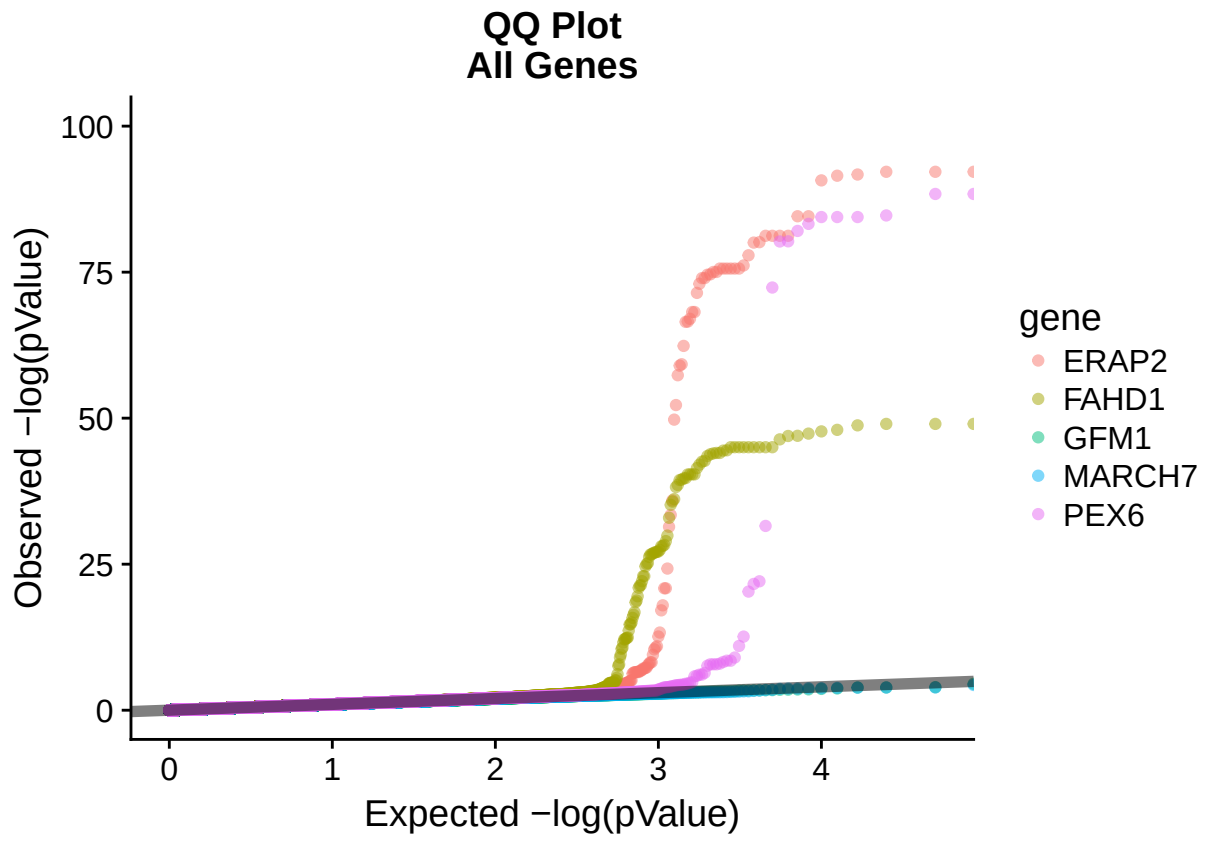
Based on on the principal component analysis of the genomes, colored by the population of origin, it is possible to see that there is clearly population structure.

Test each covariate individually.


count
2000
1000
0
0.00
0.25
0.50
0.75
1.00
pval
gene
ERAP2
FAHD1
GFM1
MARCH7
PEX6




```
# A tibble: 3 x 2
# Groups:   gene [3]
  gene      n
  <chr> <int>
1 ERAP2    75
2 FAHD1    93
3 PEX6     31
```



Seems like there is a problem with just the base model. Based of the PCA analysis, it appears that the Population may be an important covariate to include in the models.

But first, let's test the relationship of each phenotype with each covariate:

```
gene1_pop_lm <- lm(pheno[, 1] ~ as.numeric(factor(covars$Population)))  # significant
gene2_pop_lm <- lm(pheno[, 2] ~ as.numeric(factor(covars$Population)))  # ns
gene3_pop_lm <- lm(pheno[, 3] ~ as.numeric(factor(covars$Population)))  # ns
gene4_pop_lm <- lm(pheno[, 4] ~ as.numeric(factor(covars$Population)))  # ns
gene5_pop_lm <- lm(pheno[, 5] ~ as.numeric(factor(covars$Population)))  # ns

nested_pheno <- pheno %>%
  rownames_to_column("samples") %>%
  left_join(
    covars %>%
      rownames_to_column("samples"),
    by = "samples"
  ) %>%
  gather("gene", "abundance", 2:6) %>%
  nest(-gene)

num_factor_pop_lm <- nested_pheno %>%
  mutate(model = map(data, ~lm(abundance ~ as.numeric(factor(Population)), data = .))) %>%
  mutate(glance_model = map(model, glance)) %>%
  unnest(glance_model) %>%
  dplyr::select(gene, p.value)
num_factor_pop_lm[which(num_factor_pop_lm$p.value < 0.05), ]
```

```r
# what if Population was treated as a factor and R was allowed to code it
cat_factor_pop_lm <- nested_pheno %>%
  mutate(model = map(data, ~lm(abundance ~ factor(Population), data = .))) %>%
  mutate(glance_model = map(model, glance)) %>%
  unnest(glance_model) %>%
  dplyr::select(gene, p.value)
cat_factor_pop_lm[which(cat_factor_pop_lm$p.value < 0.05), ]
# what about the Sex covariable?
covar_sex_lm <- nested_pheno %>%
  mutate(model = map(data, ~lm(abundance ~ factor(Sex), data = .))) %>%
  mutate(glance_model = map(model, glance)) %>%
  unnest(glance_model) %>%
  dplyr::select(gene, p.value)
any(covar_sex_lm < 0.05)
# none are significant on the sex covariable
# what about a combination?
covars_sex_num_pop_lm <- nested_pheno %>%
  mutate(model = map(data, ~lm(abundance ~ as.numeric(factor(Population)) + factor(Sex), data = .))) %>%
  mutate(glance_model = map(model, glance)) %>%
  unnest(glance_model) %>%
  dplyr::select(gene, p.value)
covars_sex_num_pop_lm[which(covars_sex_num_pop_lm$p.value < 0.05), ]
nested_pheno %>%
  mutate(model = map(data, ~lm(abundance ~ as.numeric(factor(Population)) + factor(Sex), data = .))) %>%
  mutate(tidy_model = map(model, tidy)) %>%
  unnest(tidy_model) %>%
  filter(gene %in% covars_sex_num_pop_lm[which(covars_sex_num_pop_lm$p.value < 0.05), 1])
# indicates that the relationship between population only is significant
covars_sex_cat_pop_lm <- nested_pheno %>%
  mutate(model = map(data, ~lm(abundance ~ factor(Population) + factor(Sex), data = .))) %>%
  mutate(glance_model = map(model, glance)) %>%
  unnest(glance_model) %>%
  dplyr::select(gene, p.value)
covars_sex_cat_pop_lm[which(covars_sex_cat_pop_lm$p.value < 0.05), ]
```

Including covariates:

```r
library(lmtest)

lr_likelihood <- function(y, x_input = NULL){
    n_samples <- length(y)
    X_mx <- cbind(matrix(1, nrow = n_samples, ncol = 1), x_input)
    MLE_beta <- ginv(t(X_mx) %*% X_mx) %*% t(X_mx) %*% y
    y_hat <- X_mx %*% MLE_beta
    var_hat <- sum((y - (y_hat))^2) / (n_samples - 1)
    log_likelihood <- -((n_samples / 2) * log(2 * pi * var_hat) ) - ((1/ (2*var_hat)) * sum((y - (y_hat
    return(log_likelihood)
}

LRT_test <- function(logl_H0, logl_HA, df_test){

    LRT<-2*logl_HA-2*logl_H0 #likelihood ratio test statistic
    #likelihood ratio test statistic for every genotype
    pval <- pchisq(LRT, df_test, lower.tail = F)
```

```
    return(pval)
}

set.seed(2018)
x = sample(c(-1,0,1), 100, replace = TRUE)
y = 0.9 * x + rnorm(100)
h0_nocovar <- lr_likelihood(y)
h1_nocovar <- lr_likelihood(y, x)
LRT_test(h0_nocovar, h1_nocovar, df_test = 1)
x_c = sample(c(0,1), 100, replace = TRUE)
y2 = y + 0.8 * x_c

h0_withcovar <- lr_likelihood(y2)
h1_withcovar <- lr_likelihood(y2, x)
LRT_test(h0_withcovar, h1_withcovar, df_test = 1)

h0_includecovar <- lr_likelihood(y2, x_c)
ha_includecovar <- lr_likelihood(y2, cbind(x,x_c))
LRT_test(h0_includecovar, ha_includecovar, df_test = 1)

# y2 = y, x_c = covar, x = x

X_mat_null <- cbind(1, x_c)
beta_hat_null <- ginv(t(X_mat_null) %*% X_mat_null) %*% t(X_mat_null) %*% y2
X_mat_alt <- cbind(1, x, x_c)
beta_hat_alt <- ginv(t(X_mat_alt) %*% X_mat_alt) %*% t(X_mat_alt) %*% y2
y_hat_null <- X_mat_null %*% beta_hat_null
y_hat_alt <- X_mat_alt %*% beta_hat_alt
SSE_null <- sum((y2 - y_hat_null) ^ 2)
SSE_alt <- sum((y2 - y_hat_alt) ^ 2)
fstat <- ((SSE_null - SSE_alt) / 2) / (SSE_alt / (length(x) - 3))

pf(fstat, 2, length(x) - 3, lower.tail = FALSE)

# testing with real data

X_mat_null <- cbind(matrix(1, nrow = nrow(x_a), ncol = 1), NULL)
beta_hat_null <- ginv(t(X_mat_null) %*% X_mat_null) %*% t(X_mat_null) %*% pheno[, 2]
X_mat_alt <- cbind(1, x_a[, 2], x_d[, 2])
beta_hat_alt <- ginv(t(X_mat_alt) %*% X_mat_alt) %*% t(X_mat_alt) %*% pheno[, 2]
y_hat_null <- X_mat_null %*% beta_hat_null
y_hat_alt <- X_mat_alt %*% beta_hat_alt
SSE_null <- sum((pheno[, 2] - y_hat_null) ^ 2)
SSE_alt <- sum((pheno[, 2] - y_hat_alt) ^ 2)
fstat <- ((SSE_null - SSE_alt) / 2) / (SSE_alt / (nrow(geno) - 3))
pf(fstat, 2, nrow(geno) - 3, lower.tail = FALSE)
```

```
lm_test <- lm(pheno$ENSG00000164308.12 ~ x_a[, 1000] + x_d[, 1000] + factor(covars$Population))
lm_tidy <- tidy(lm_test)
fstat <- summary(lm_test)$fstatistic
fstat_2 <- glance(lm_test)$statistic
pf(fstat[1], fstat[2], fstat[3], lower.tail = FALSE)
```