QG Final Exam

Yuchen Sun

2023-05-20

Note: All codes can compile, but some needs longer time to run.

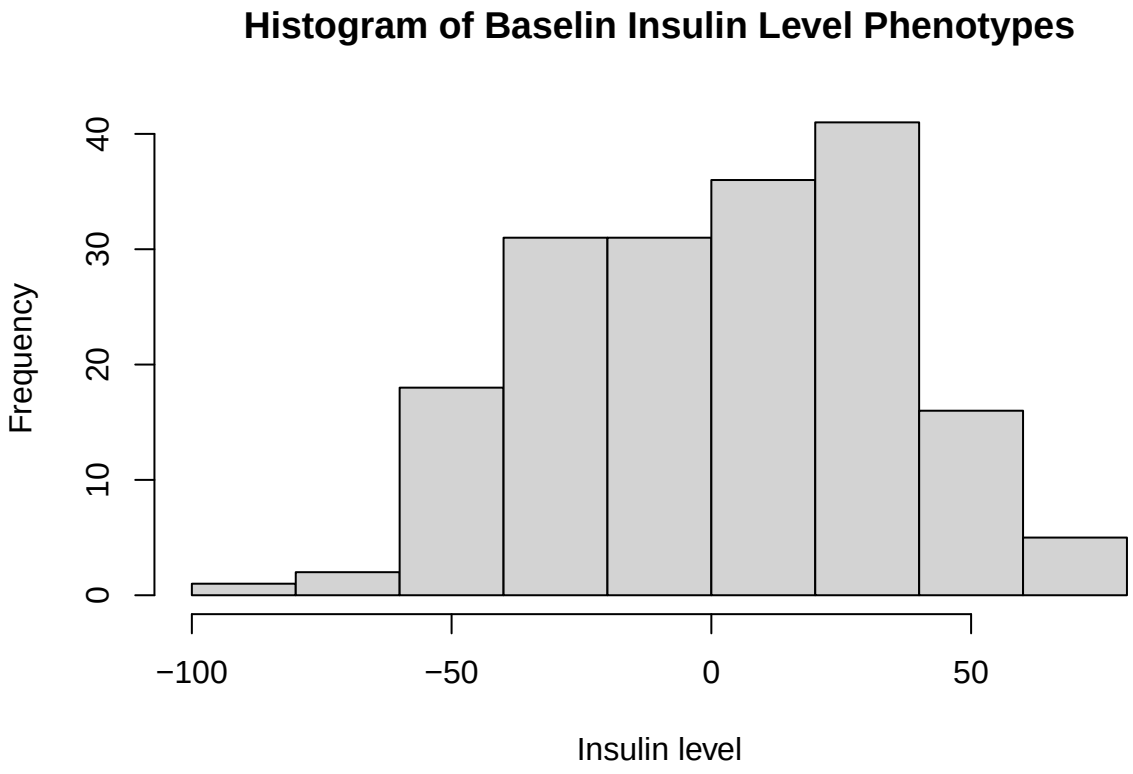
1.(a) Import the Baseline Insulin Level phenotypes.

```
pheno_insulin <- read.csv(paste0(wd, "2023QG_finalexam_insulin.txt"),
    header = F)$V1
n <- length(pheno_insulin)
cat("Total sample size n =", n)
```

```
## Total sample size n = 181
```

1.(b) Plot a histogram of the Baseline Insulin Level phenotypes.

```
hist(pheno_insulin, breaks = 10, xlab = "Insulin level",
    main = "Histogram of Baselin Insulin Level Phenotypes")
```



1.(c) Using no more than two sentences, explain why a linear regression would be an appropriate model for a GWAS analysis of these phenotype data given this observation?

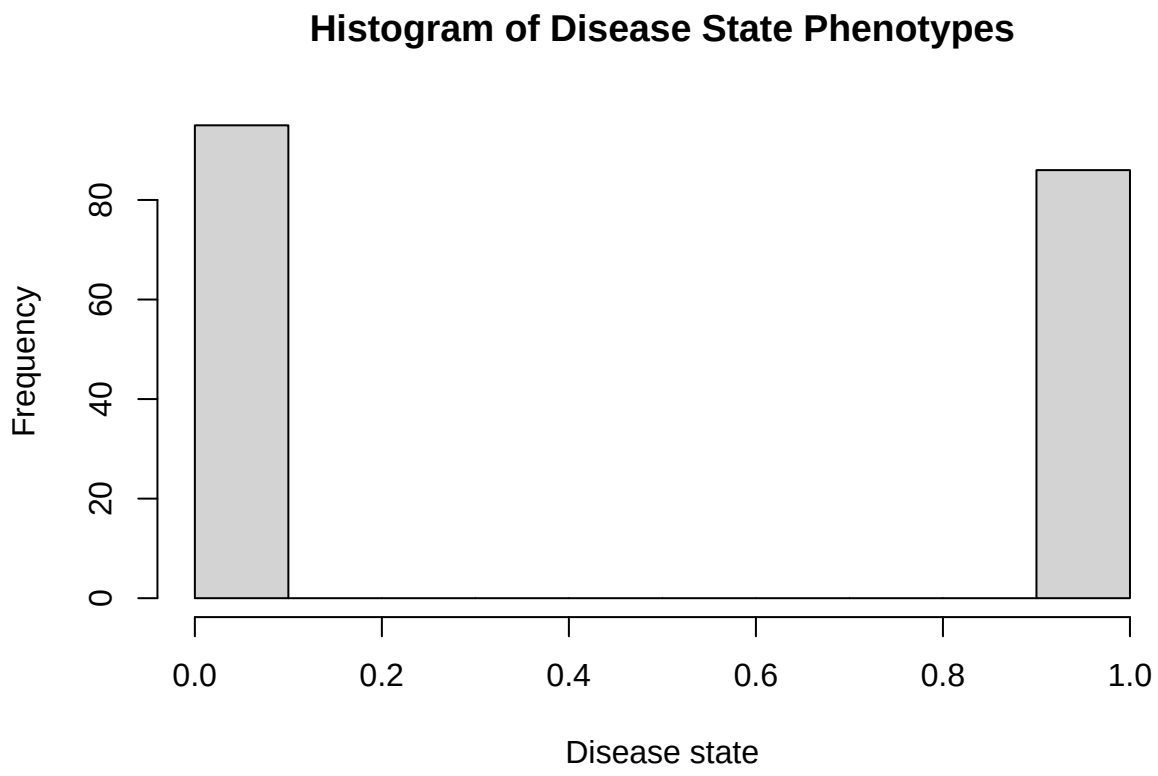
Because these phenotype data is continuous and follows a normal distribution, therefore can be best described by a linear regression.

2.(a) Import the Disease State phenotypes.

```
pheno_diabetes <- read.csv(paste0(wd, "2023QG_finalexam_diabetes.txt"),
    header = F)$V1
```

2.(b) Plot a histogram or bar plot of the Disease State phenotypes.

```
hist(pheno_diabetes, xlab = "Disease state", main = "Histogram of Disease State Phenotypes")
```



2.(c) Using no more than two sentences, explain why a logistic regression would be an appropriate model for a GWAS analysis of these phenotype data?

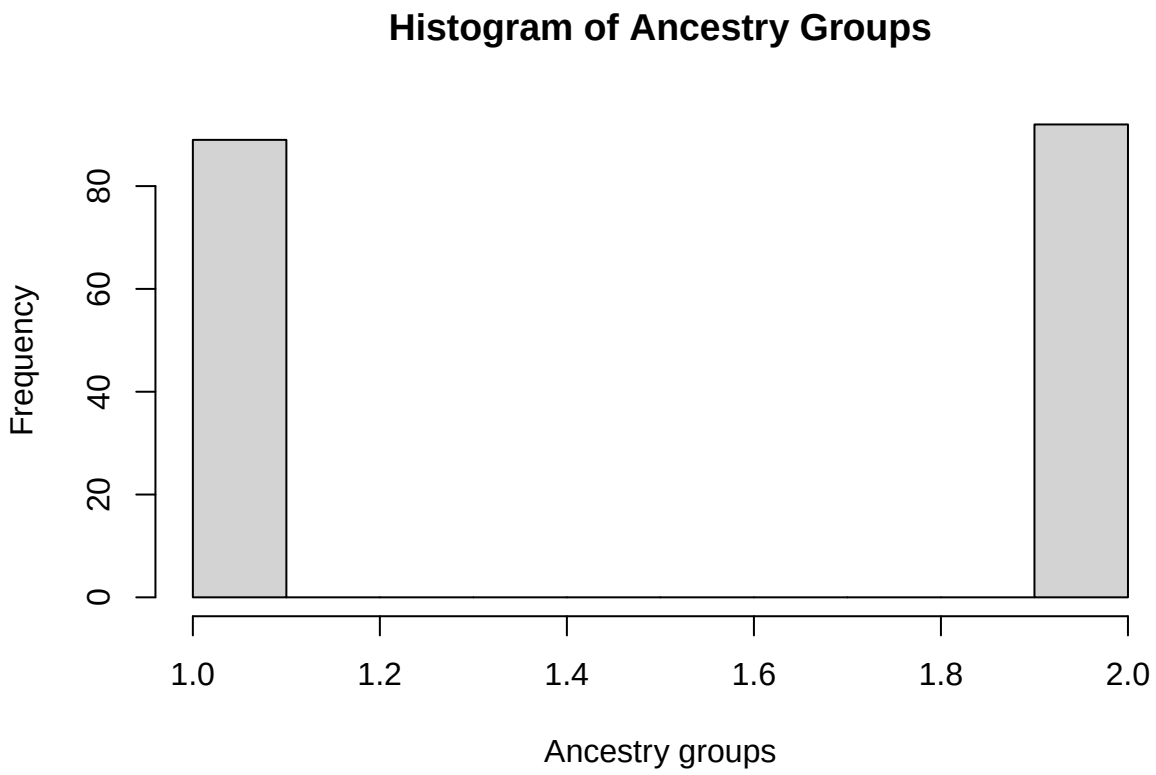
Because these phenotype data is not continuous, but discrete with only two states. Therefore, it is best to be described by a logistic regression.

3.(a) Import the Ancestry Group data.

```
ancestry <- read.csv(paste0(wd, "2023QG_finalexam_ancestry.txt"),
    header = F)$V1
```

3.(b) Plot a histogram or bar plot of the Ancestry Groups

```
hist(ancestry, xlab = "Ancestry groups", main = "Histogram of Ancestry Groups")
```



3.(c) Using no more than two sentences, provide the two conditions under which these ancestry groups could produce false positives in your GWAS analyses if you DO NOT incorporate a covariate to account for ancestry?

If (1) the marker genotype is not correlated with a causal polymorphism but (2) the ancestry groups is correlated with both the marker genotype and the phenotype, then not incorporating the ancestry as a covariate would produce false positives.

4.(a) Import the genotype data.

```
genotypes <- read.csv(paste0(wd, "2023QG_finalexam_genotypes.txt"),
    header = F)
```

4.(b) Report the number of genotypes N.

```
N <- dim(genotypes)[2]
cat("Number of genotypes N =", N)
```

```
## Number of genotypes N = 22001
```

5.(a) genetic linear regression model WITH NO COVARIATES

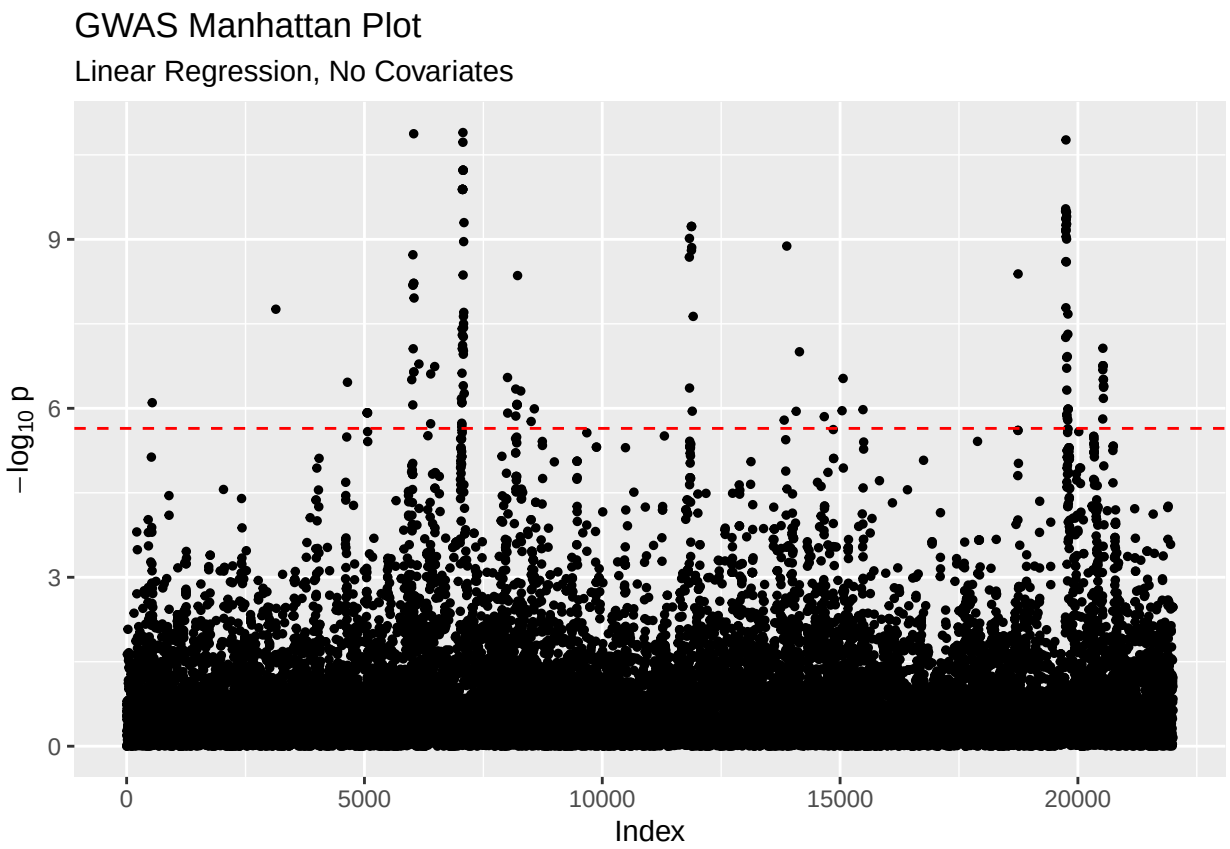
```r
xa <- as.matrix(genotypes - 1)
xd <- 1 - 2 * abs(xa)
```

```r
# GWAS function for without covariates
GWAS_linear_no_cov <- function(xa_input, xd_input, pheno_input) {
    n_samples <- length(xa_input)
    X_mx <- cbind(1, xa_input, xd_input)
    MLE_beta <- ginv(t(X_mx) %*% X_mx) %*% t(X_mx) %*% pheno_input
    y_hat <- X_mx %*% MLE_beta
    SSM <- sum((y_hat - mean(pheno_input))^2)
    SSE <- sum((pheno_input - y_hat)^2)
    df_M <- ncol(X_mx) - 1
    df_E <- n_samples - ncol(X_mx)
    MSM <- SSM/df_M
    MSE <- SSE/df_E
    Fstatistic <- MSM/MSE
    pval <- pf(Fstatistic, df_M, df_E, lower.tail = F)
    return(data.table(f_statistic = Fstatistic, p = pval,
        model = "No Covariate"))
}
```

```r
results.linear.no_cov <- lapply(1:ncol(xa), function(column.counter) {
    GWAS_linear_no_cov(xa[, column.counter], xd[, column.counter],
        pheno_insulin)
}) %>%
    rbindlist() %>%
    mutate(index = 1:ncol(xa))
```

5.(b) Produce a Manhattan plot for these p-values

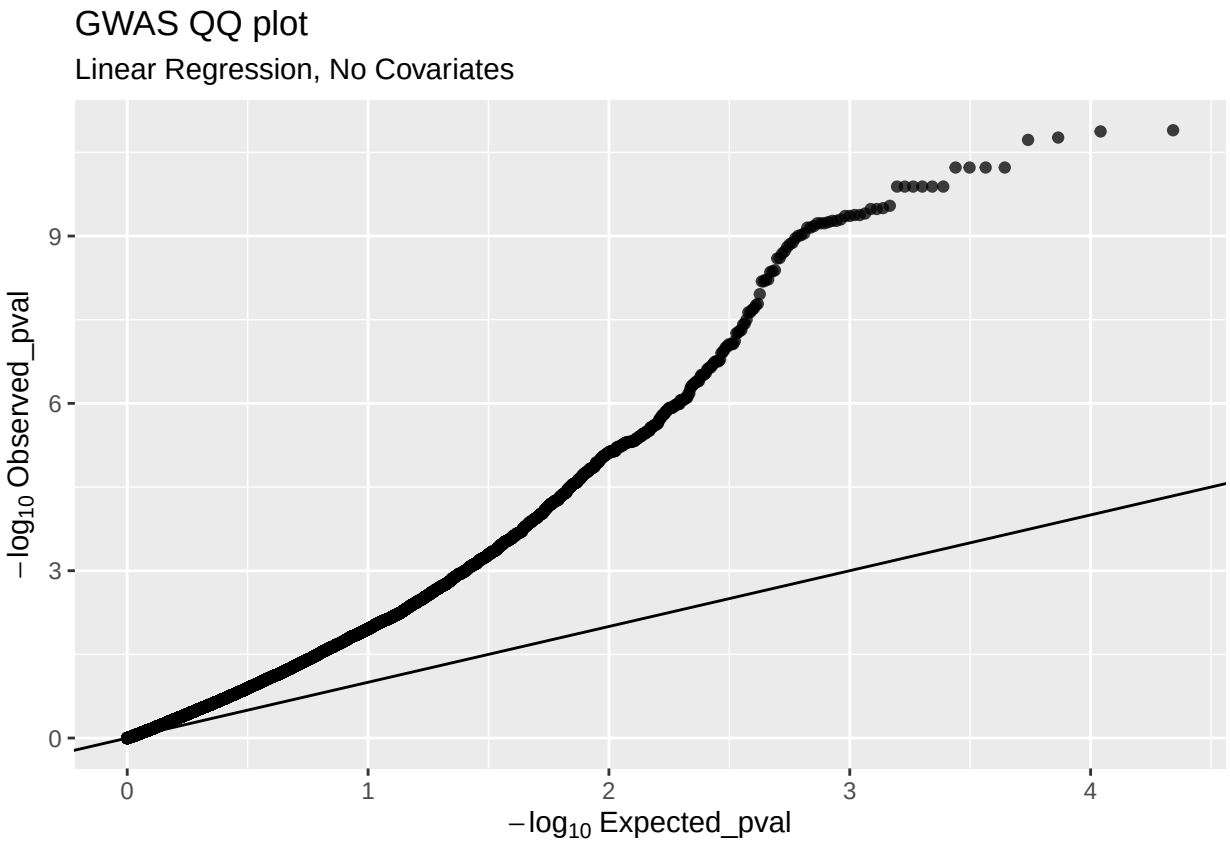
```r
ggplot(results.linear.no_cov, aes(x = index, y = -log10(p))) +
    geom_point(size = 1) + geom_hline(yintercept = -log10(alpha/N),
    color = "red", lty = 2) + labs(x = "Index", y = expression(-log[10] ~
    p), title = "GWAS Manhattan Plot", subtitle = "Linear Regression, No Covariates")
```



5.(c) Produce a QQ plot for these same p-values

```
GWAS_qqplot <- function(res) {
    obs_pval <- sort(res$p)
    exp_pval <- qunif(seq(0, 1, length.out = length(obs_pval) +
        2), min = 0, max = 1)
    exp_pval <- exp_pval[exp_pval != 0 & exp_pval != 1]
    p_df <- data.frame(observed = -log10(obs_pval), expected = -log10(exp_pval))
    p <- ggplot(p_df, aes(x = expected, y = observed)) +
        geom_point(alpha = 0.75) + geom_abline(intercept = 0,
        slope = 1) + labs(x = expression(-log[10] ~ Expected_pval),
        y = expression(-log[10] ~ Observed_pval), title = "GWAS QQ plot")
    return(p)
}
```

```
GWAS_qqplot(results.linear.no_cov) + labs(subtitle = "Linear Regression, No Covariates")
```



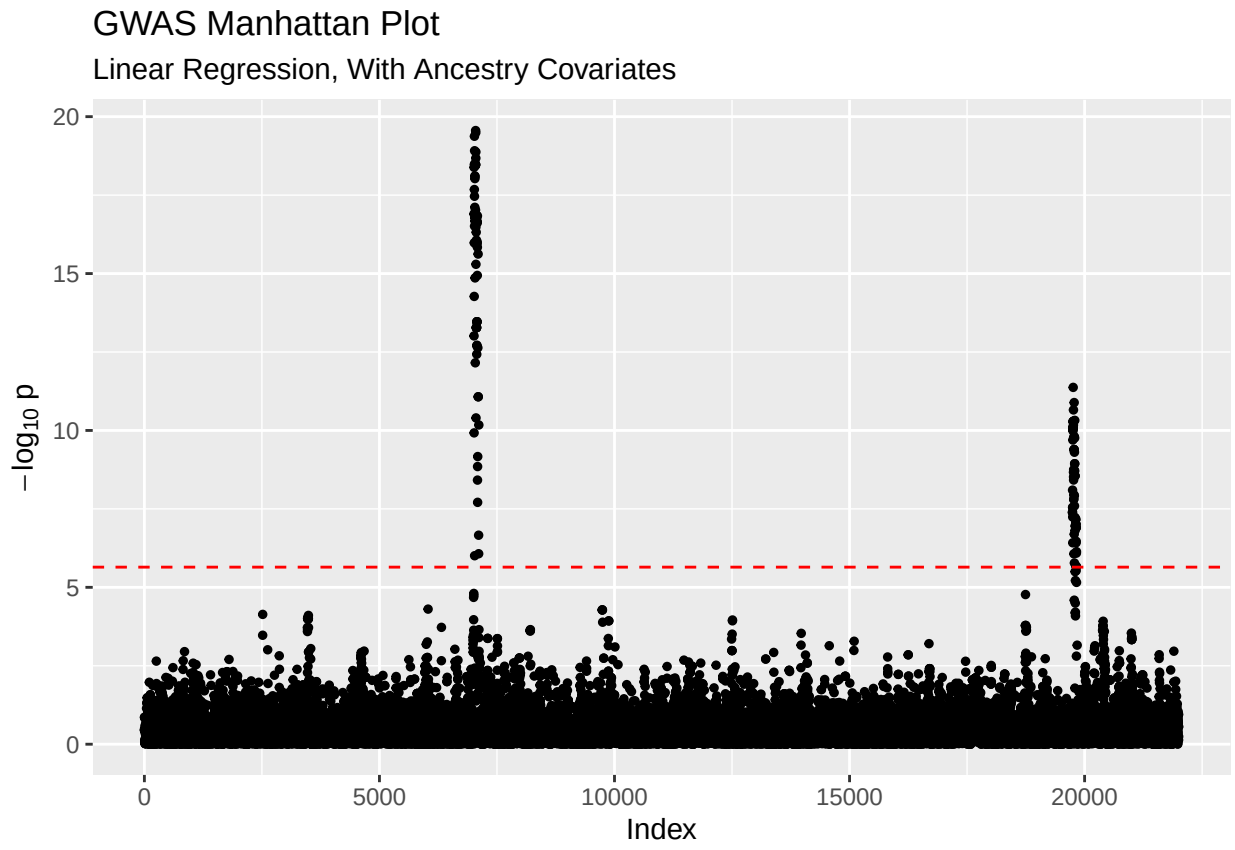
6.(a) genetic linear regression model WITH THE ANCESTRY indicators calculated in question [3] as a (single) covariate

```
# GWAS function for with covariates
GWAS_linear_cov <- function(xa_input, xd_input, xz_input,
    pheno_input) {
    n_samples <- length(xa_input)
    x_h1 <- cbind(1, xa_input, xd_input, xz_input)
    MLE_h1 <- ginv(t(x_h1) %*% x_h1) %*% t(x_h1) %*% pheno_input
    x_h0 <- cbind(1, xz_input)
    MLE_h0 <- ginv(t(x_h0) %*% x_h0) %*% t(x_h0) %*% pheno_input
    y_hat_0 <- x_h0 %*% MLE_h0
    y_hat_1 <- x_h1 %*% MLE_h1
    SSE_theta_0 <- sum((pheno_input - y_hat_0)^2)
    SSE_theta_1 <- sum((pheno_input - y_hat_1)^2)
    df_M <- ncol(x_h1) - ncol(x_h0)
    df_E <- n_samples - ncol(x_h1)
    numertator <- (SSE_theta_0 - SSE_theta_1)/df_M
    denom <- SSE_theta_1/df_E
    Fstatistic <- numertator/denom
    pval <- pf(Fstatistic, df_M, df_E, lower.tail = F)
    return(data.table(f_statistic = Fstatistic, p = pval,
        model = "Covariate"))
}
```

```
results.linear.cov <- lapply(1:ncol(xa), function(column.counter) {
    GWAS_linear_cov(xa[, column.counter], xd[, column.counter],
        ancestry, pheno_insulin)
}) %>%
    rbindlist() %>%
    mutate(index = 1:ncol(xa))
```

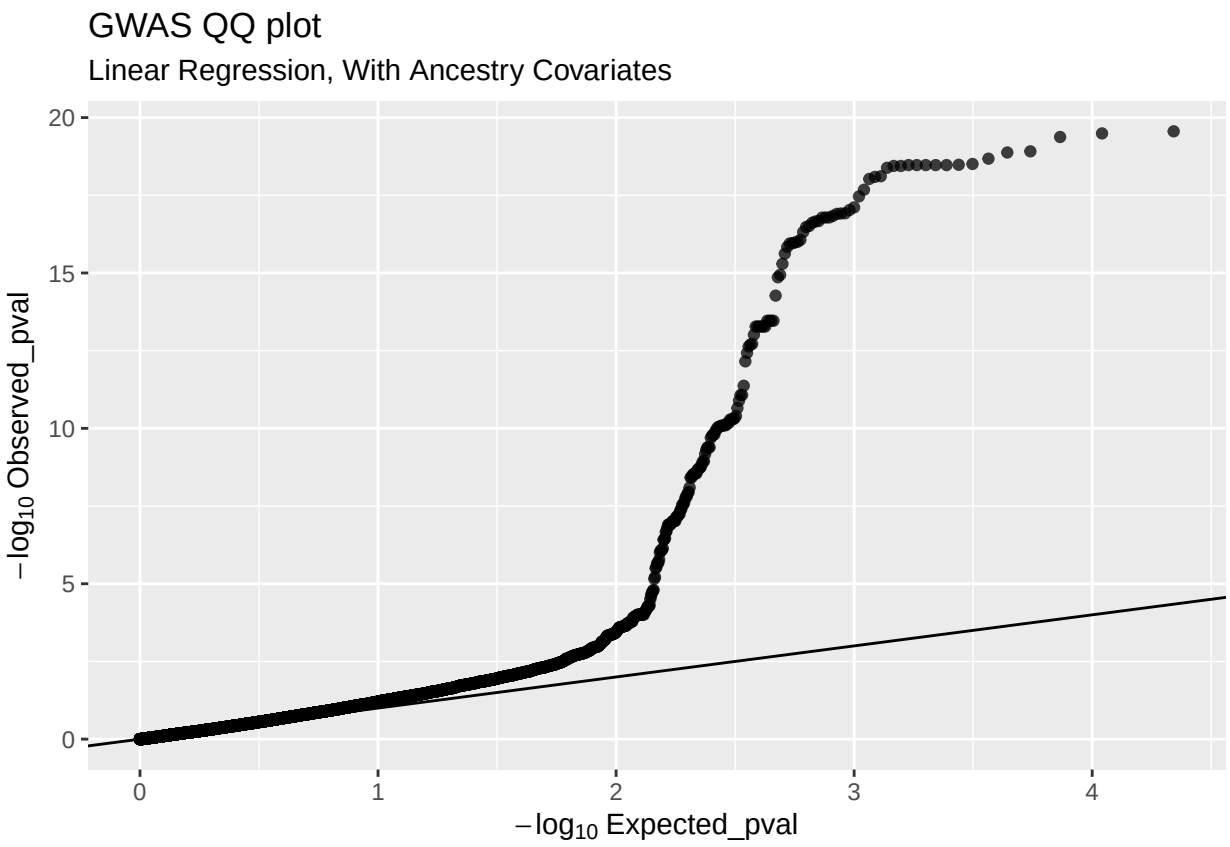
6.(b) Produce a Manhattan plot for these p-values

```
ggplot(results.linear.cov, aes(x = index, y = -log10(p))) +
    geom_point(size = 1) + geom_hline(yintercept = -log10(alpha/N),
    color = "red", lty = 2) + labs(x = "Index", y = expression(-log[10] ~
    p), title = "GWAS Manhattan Plot", subtitle = "Linear Regression, With Ancestry Covariates")
```



6.(c) Produce a QQ plot for these same p-values

```
GWAS_qqplot(results.linear.cov) + labs(subtitle = "Linear Regression, With Ancestry Covariates")
```



7.(a) Using no more than two sentences, explain whether you think the analysis in question [5] or in question [6] resulted in appropriate model fit to the data and explain the reasoning behind your answer based on the shapes of the QQ plots.

The analysis in question [6] resulted in the appropriate model fit, because the QQ plot for the analysis mostly complies to the null hypothesis trend, with only a tail with high deviations from the trend at the end indicating high significance (low p-value) matches.

7.(b) When controlling the study-wide Type 1 error to 0.05, what is the appropriate p-value cutoff for assessing which genetic markers are significant when using a Bonferroni correction?

The corrected p value cutoff would be 0.05 divided by the sample size (`alpha/N`), therefore

```
cat("Adjusted p-value cutoff =", alpha/N)
```

```
## Adjusted p-value cutoff = 2.272624e-06
```

7.(c) For the results of question [6], note how many separate peaks you observed that were greater than the Bonferroni correction level and for each of these separate peaks, list the p-value of the most significant SNP in the peak.

There are two separate peaks greater than the Bonferroni correction level.

```
results.linear.cov$index[results.linear.cov$p < alpha/N]
```

```
##   [1]  7011  7012  7013  7014  7016  7017  7019  7020  7021  7022  7023  7024
##  [13]  7025  7028  7029  7030  7031  7032  7033  7034  7035  7037  7038  7039
##  [25]  7040  7041  7042  7043  7044  7045  7046  7047  7048  7049  7050  7051
##  [37]  7052  7053  7054  7055  7057  7058  7059  7060  7063  7064  7065  7066
##  [49]  7067  7069  7070  7071  7072  7073  7074  7075  7076  7078  7079  7080
##  [61]  7081  7082  7083  7084  7086  7088  7089  7091  7093  7099  7101  7102
##  [73]  7110  7111  7114 19741 19743 19744 19745 19746 19747 19748 19749 19750
##  [85] 19751 19752 19753 19754 19755 19756 19757 19758 19759 19760 19761 19762
##  [97] 19763 19764 19765 19766 19767 19768 19769 19770 19772 19773 19774 19775
## [109] 19776 19777 19778 19779 19781 19782 19783 19784 19785 19786 19787 19788
## [121] 19789 19790 19791 19792 19794 19795 19796 19797 19799 19807 19808 19809
## [133] 19810 19811 19812 19813 19814 19815 19816 19817 19818 19819 19821 19822
## [145] 19823 19824 19825 19827
```

The first peak is between index 7011~7114, and the second peak is between index 19741~19827.

```
cat("Most significnat p-value for first peak =", min(results.linear.cov$p[7011:7114]))
```

```
## Most significnat p-value for first peak = 2.767575e-20
```

```
cat("Most significnat p-value for second peak =", min(results.linear.cov$p[19741:19827]))
```

```
## Most significnat p-value for second peak = 4.242741e-12
```

7.(d) Assuming each peak indicates the position of a causal genotype, is the most significant SNP in each peak necessarily closer to the causal genotype than either of the SNPs on each side of the most significant SNP? Use no more than two sentences in your answer.

The most significant SNP is not necessarily closer to the causal genotype, because the result is restricted by the resolution of GWAS, which is more determined by the level of linkage disequilibrium. The result from GWAS would give a range (corresponding to the tower), but not necessarily the tip of the tower.

8.(a) genetic logistic regression model WITH NO COVARIATES.

```
gamma_inv_calc <- function(X_mx, beta_t) {
    K <- X_mx %*% beta_t
    gamma_inv <- exp(K)/(1 + exp(K))
    return(gamma_inv)
}

W_calc <- function(gamma_inv) {
    W <- diag(as.vector(gamma_inv * (1 - gamma_inv)))
    return(W)
}

beta_update <- function(X_mx, W, Y, gamma_inv, beta) {
    beta_up <- beta + ginv(t(X_mx) %*% W %*% X_mx) %*% t(X_mx) %*%
        (Y - gamma_inv)
    return(beta_up)
}
```

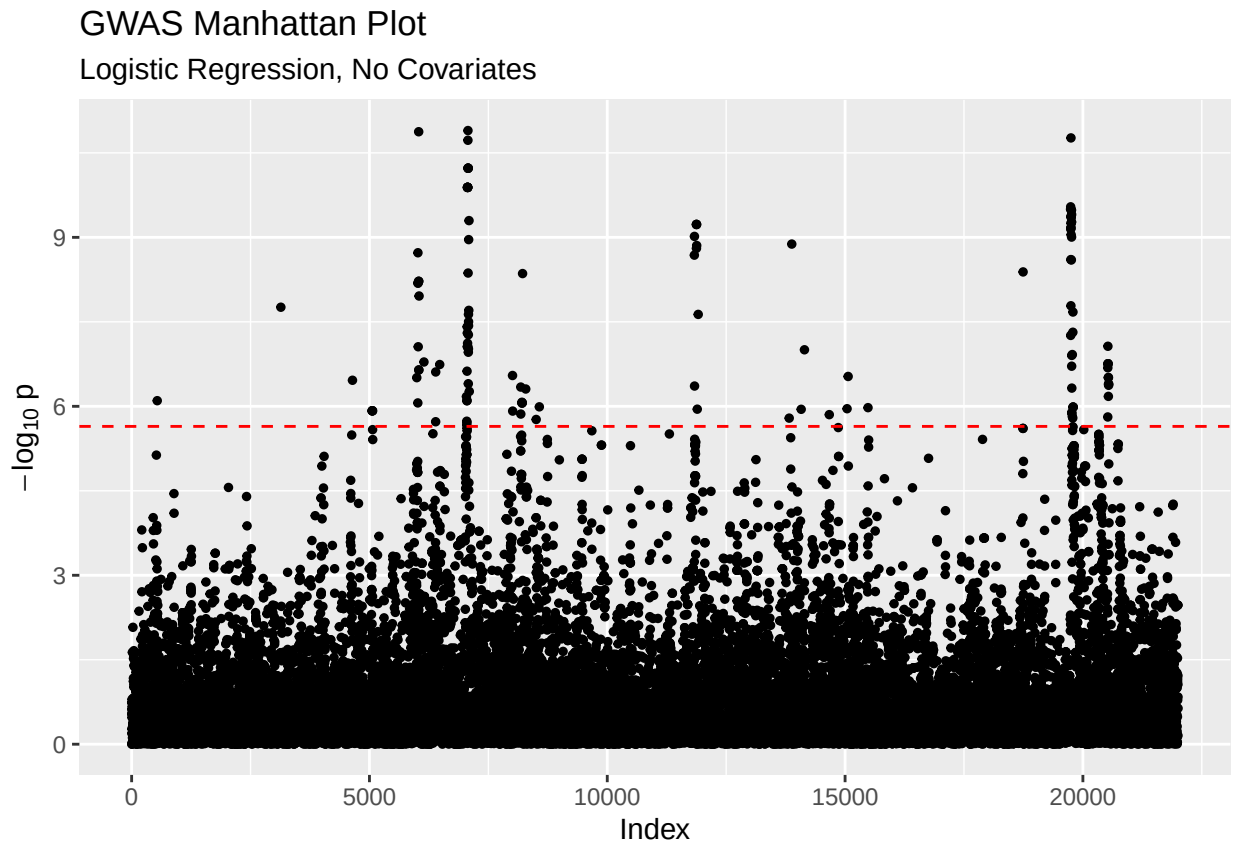
```
dev_calc <- function(Y, gamma_inv) {
    deviance <- 2 * (sum(Y[Y == 1] * log(Y[Y == 1]/gamma_inv[Y ==
        1])) + sum((1 - Y[Y == 0]) * log((1 - Y[Y == 0])/(1 -
        gamma_inv[Y == 0]))))
    return(deviance)
}

loglik_calc <- function(Y, gamma_inv) {
    loglik <- sum(Y * log(gamma_inv) + (1 - Y) * log(1 -
        gamma_inv))
    return(loglik)
}
```

```
logistic.IRLS <- function(X_mx, Y = pheno_diabetes, beta.initial.vec = c(0,
    0, 0), d.stop.th = 1e-06, it.max = 100) {
    beta_t <- beta.initial.vec
    dt <- 0
    gamma_inv <- gamma_inv_calc(X_mx, beta_t)
    for (i in 1:it.max) {
        dpt1 <- dt
        W <- W_calc(gamma_inv)
        beta_t <- beta_update(X_mx, W, Y, gamma_inv, beta_t)
        gamma_inv <- gamma_inv_calc(X_mx, beta_t)
        dt <- dev_calc(Y, gamma_inv)
        absD <- abs(dt - dpt1)

        if (absD < d.stop.th) {
            logl <- loglik_calc(Y, gamma_inv)
            return(list(beta_t, logl))
        }
    }
    return(list(beta_t = c(NA, NA, NA), logl = NA))
}

GWAS_logistic_no_cov <- function(Xa, Xd, Y, beta.initial.vec = c(0,
    0, 0), d.stop.th = 1e-06, it.max = 100) {
    beta_t <- beta.initial.vec
    dt <- 0
    X_mx <- cbind(rep(1, length(Y)), Xa, Xd)
    gamma_inv <- gamma_inv_calc(X_mx, beta_t)
    h1 <- logistic.IRLS(X_mx, Y = Y)
    X_mx <- cbind(rep(1, length(Y)), rep(0, length(Y)),
        rep(0, length(Y)))
    gamma_inv <- gamma_inv_calc(X_mx, beta_t)
    h0 <- logistic.IRLS(X_mx, Y = Y, beta_t)

    LRT <- 2 * h1[[2]] - 2 * h0[[2]]
    pval <- pchisq(LRT, 2, lower.tail = F)
    # return(pval)
    return(data.table(p = pval, model = "No Covariate"))
}
```

```
results.logistic.no_cov <- lapply(1:ncol(xa), function(column.counter) {
    GWAS_logistic_no_cov(xa[, column.counter], xd[, column.counter],
        pheno_diabetes)
}) %>%
    rbindlist() %>%
    mutate(index = 1:ncol(xa))
```

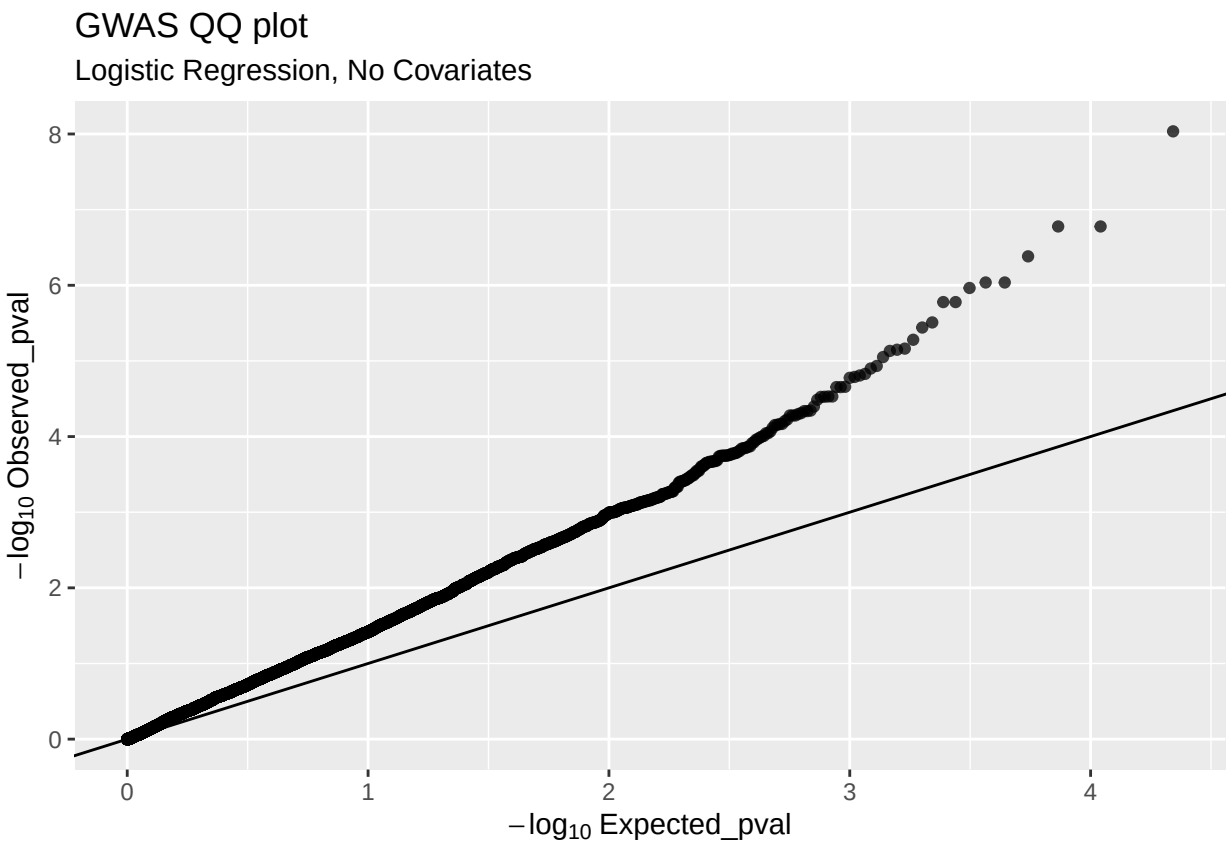
8.(b) Produce a Manhattan plot for these p-values.

```
ggplot(results.linear.no_cov, aes(x = index, y = -log10(p))) +
    geom_point(size = 1) + geom_hline(yintercept = -log10(alpha/N),
    color = "red", lty = 2) + labs(x = "Index", y = expression(-log[10] ~
    p), title = "GWAS Manhattan Plot", subtitle = "Logistic Regression, No Covariates")
```



8.(c) Produce a QQ plot for these same p-values.

```
GWAS_qqplot(results.logistic.no_cov) + labs(subtitle = "Logistic Regression, No Covariates")
```



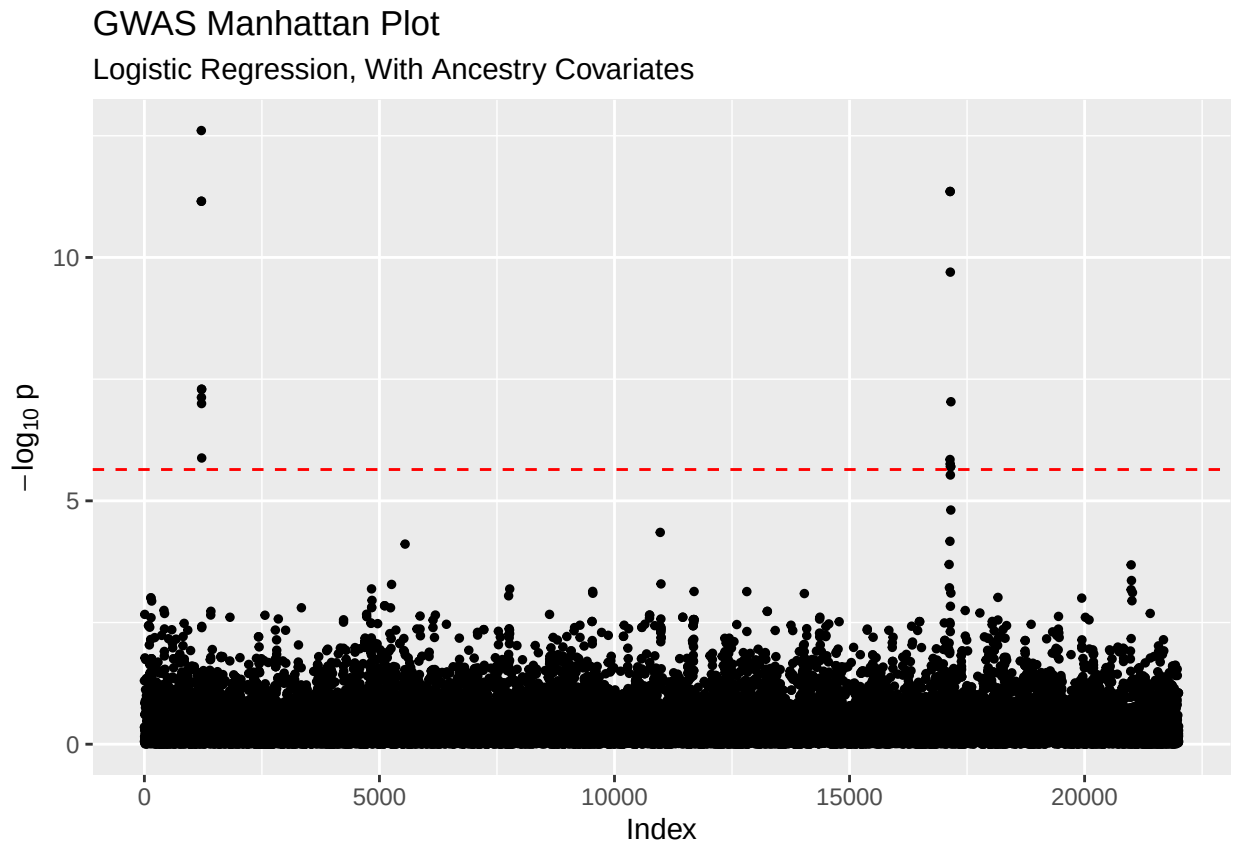
9.(a) genetic logistic regression model WITH THE ANCESTRY indicators calculated in question [3] as a (single) covariate.

```
GWAS_logistic_cov <- function(Xa, Xd, Xz, Y, beta.initial.vec = c(0,
    0, 0, 0), d.stop.th = 1e-06, it.max = 100) {
    beta_t <- beta.initial.vec
    dt <- 0
    X_mx <- cbind(rep(1, length(Y)), Xa, Xd, Xz)
    gamma_inv <- gamma_inv_calc(X_mx, beta_t)
    h1 <- logistic.IRLS(X_mx, Y = Y, beta.initial.vec = c(0,
        0, 0, 0))
    X_mx <- cbind(rep(1, length(Y)), rep(0, length(Y)),
        rep(0, length(Y)), Xz)
    gamma_inv <- gamma_inv_calc(X_mx, beta_t)
    h0 <- logistic.IRLS(X_mx, Y = Y, beta_t)

    LRT <- 2 * h1[[2]] - 2 * h0[[2]]
    pval <- pchisq(LRT, 2, lower.tail = F)
    return(data.table(p = pval, model = "No Covariate"))
}
```

```
results.logistic.cov <- lapply(1:ncol(xa), function(column.counter) {
    GWAS_logistic_cov(xa[, column.counter], xd[, column.counter],
        ancestry, pheno_diabetes)
}) %>%
    rbindlist() %>%
    mutate(index = 1:ncol(xa))
```

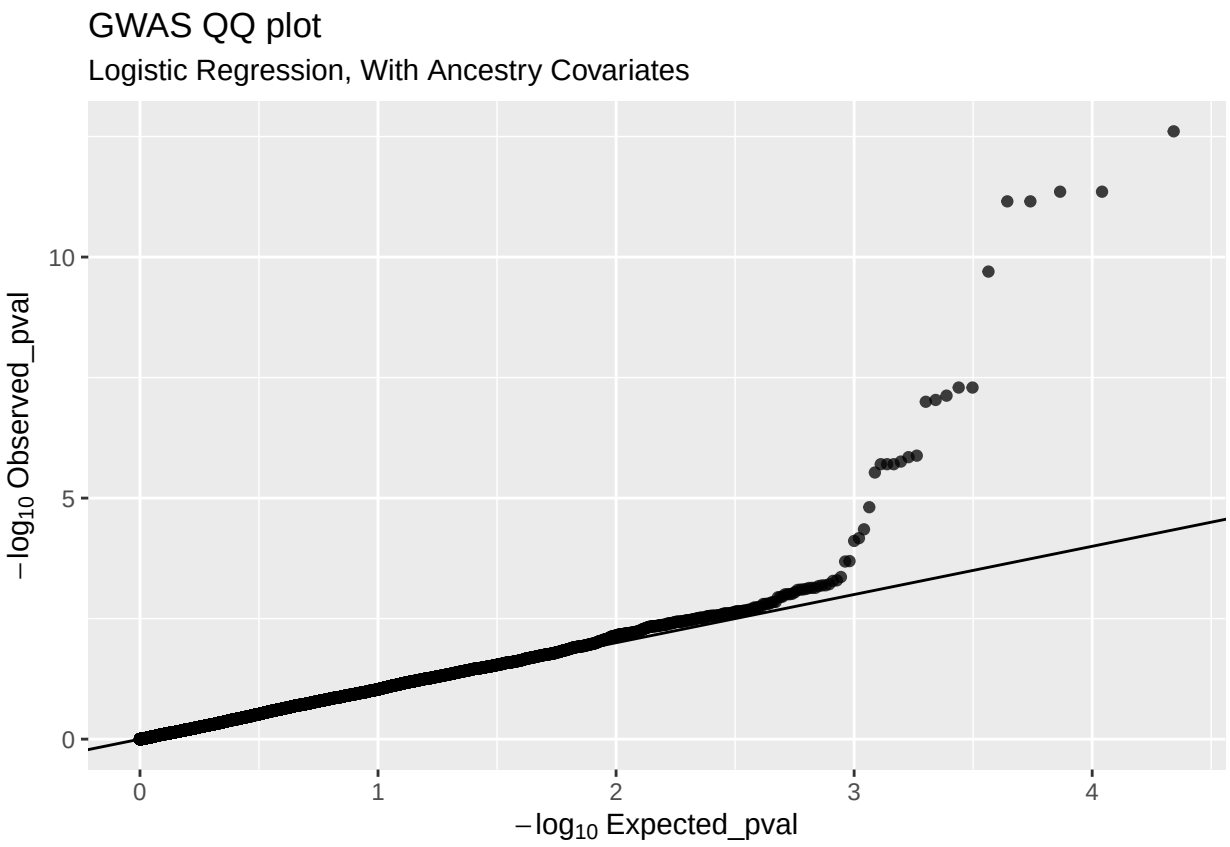
9.(b) Produce a Manhattan plot for these p-values.

```
ggplot(results.logistic.cov, aes(x = index, y = -log10(p))) +
    geom_point(size = 1) + geom_hline(yintercept = -log10(alpha/N),
    color = "red", lty = 2) + labs(x = "Index", y = expression(-log[10] ~
    p), title = "GWAS Manhattan Plot", subtitle = "Logistic Regression, With Ancestry Covariates")
```



9.(c) Produce a QQ plot for these same p-values.

```
GWAS_qqplot(results.logistic.cov) + labs(subtitle = "Logistic Regression, With Ancestry Covariates")
```



10.(a) For the results of question [9], note how many separate peaks you observed that were greater than the Bonferroni correction level and for each of these separate peaks, list the p-value of the most significant SNP in the peak.

There are two separate peaks greater than the Bonferroni correction level.

```
results.logistic.cov$index[results.logistic.cov$p < alpha/N]
```

```
##  [1]  1211  1212  1213  1214  1216  1217  1218  1219 17137 17139 17141 17142
## [13] 17144 17147 17149 17151 17157
```

The first peak is between index 1211~1219, and the second peak is between index 17137~17157.

```
cat("Most significnat p-value for first peak =", min(results.logistic.cov$p[1211:1219]))
```

```
## Most significnat p-value for first peak = 2.468504e-13
```

```
cat("Most significnat p-value for second peak =", min(results.logistic.cov$p[17137:17157]))
```

```
## Most significnat p-value for second peak = 4.409516e-12
```

10.(b) Using no more than two sentences, explain whether you believe the peaks in question [9] are indicating different causal genotypes than in question [6].

I believe the peaks in question [9] indicates different causal genotypes than question [6] because their range of significant hits does not match: they have a difference of about 5000 index and 2000 index, respectively.

10.(c) Using no more than two sentences, explain to your collaborator - by providing at least one possible reason - why there could well be many causal genotypes that impact BOTH Baseline Insulin Level and Disease State (healthy / diabetes) that were not identified with your GWAS analyses?

It might be that these causal genotypes are located at the bottom (two sides) of the tower so they are eliminated by the adjusted alpha threshold. However, if we plot the linkage disequilibrium we might find that the resolution is not specifically for the tip of the tower, but it spans through a wide range, including the specific causal genotype which is below the significance threshold.