# PupilBio Challenge

```
library(readr)
library(ggplot2)
library(caTools)
library(xgboost)
library(caret)
library(tibble)
```

Presets and reading input

```
pmp <- read_csv("../../Downloads/PupilBioTest_PMP_revA.csv", col_names = T)
cfdna <- pmp[pmp$Tissue == "cfDNA",]
islet <- pmp[pmp$Tissue == "Islet",]
```

Function to calculate coverage of both tissues.

```
calculate_coverage <- function(df){
  to_return <- data.frame()
  for (coord in unique(df$CpG_Coordinates)) {
    each_cpg <- df[df$CpG_Coordinates == coord,]
    per_sample_rep <- rowSums(each_cpg[,c(3:10)])
    #Coverage
    coverage <- sum(per_sample_rep)
    #Coefficient of Variance
    cv <- sd(per_sample_rep)/mean(per_sample_rep)*100
    #Median
    median <- median(per_sample_rep)
    to_return <- rbind(to_return,data.frame(CpG_Coordinates = coord,
                                            Coverage = coverage,
                                            Tissue = unique(each_cpg$Tissue),
                                            Median = median, CV=cv))
  }
  return(to_return)
}
```

Calculaion of said coverage

```
#This is the actual code but my R keeps crashing so I'm reading the files directly
#islet_coverage <- calculate_coverage(df = islet)
#summary(islet_coverage)
#write_csv(islet_coverage,"projects/side_project/islet_coverage.csv", col_names = T)
#cfdna_coverage <- calculate_coverage(df = cfdna)
#summary(cfdna_coverage)
#write_csv(cfdna_coverage,"projects/side_project/cfDNA_coverage.csv", col_names = T)
```

```
islet_coverage <- read_csv("../../islet_coverage.csv", col_names = T)
cfdna_coverage <- read_csv("../../cfDNA_coverage.csv", col_names = T)
summary(islet_coverage)
```

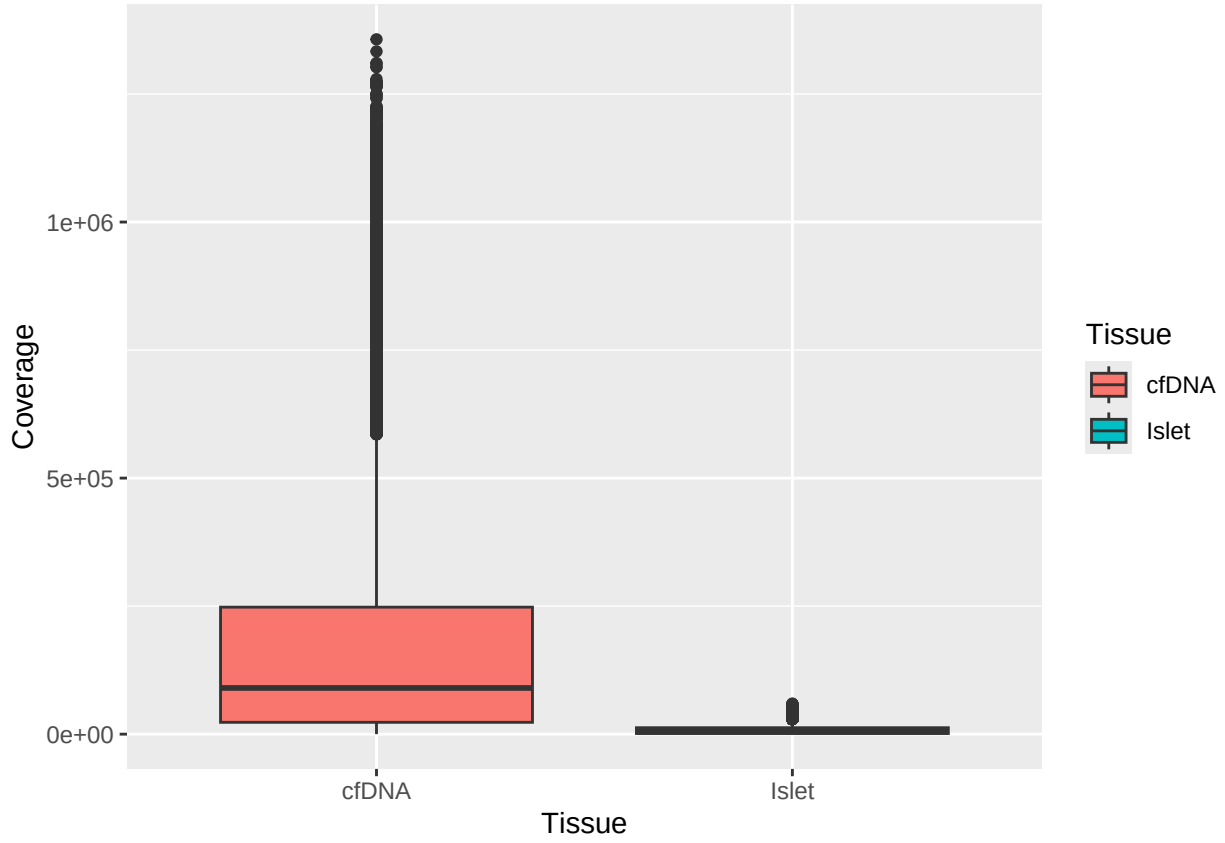
```
##  CpG_Coordinates       Coverage        Tissue              Median
##  Length:65444       Min.   :    1   Length:65444       Min.   :  1.0
##  Class :character   1st Qu.: 1334   Class :character   1st Qu.: 20.0
##  Mode  :character   Median : 4762   Mode  :character   Median : 66.0
##                     Mean   : 8534                      Mean   :126.8
##                     3rd Qu.:12540                      3rd Qu.:185.5
##                     Max.   :59249                      Max.   :899.5
##
##        CV
##  Min.   :  0.00
##  1st Qu.: 29.63
##  Median : 47.43
##  Mean   : 52.81
##  3rd Qu.: 75.21
##  Max.   :156.95
##  NA's   :93
```

```
summary(cfdna_coverage)
```

```
##  CpG_Coordinates       Coverage          Tissue              Median
##  Length:65976       Min.   :      1   Length:65976       Min.   :   1.0
##  Class :character   1st Qu.:  23114   Class :character   1st Qu.:  92.0
##  Mode  :character   Median :  90208   Mode  :character   Median : 341.0
##                     Mean   : 178228                      Mean   : 841.2
##                     3rd Qu.: 248039                      3rd Qu.:1158.0
##                     Max.   :1356771                      Max.   :6775.0
##
##        CV
##  Min.   :  0.00
##  1st Qu.: 39.65
##  Median : 58.80
##  Mean   : 63.85
##  3rd Qu.: 88.15
##  Max.   :227.30
##  NA's   :30
```

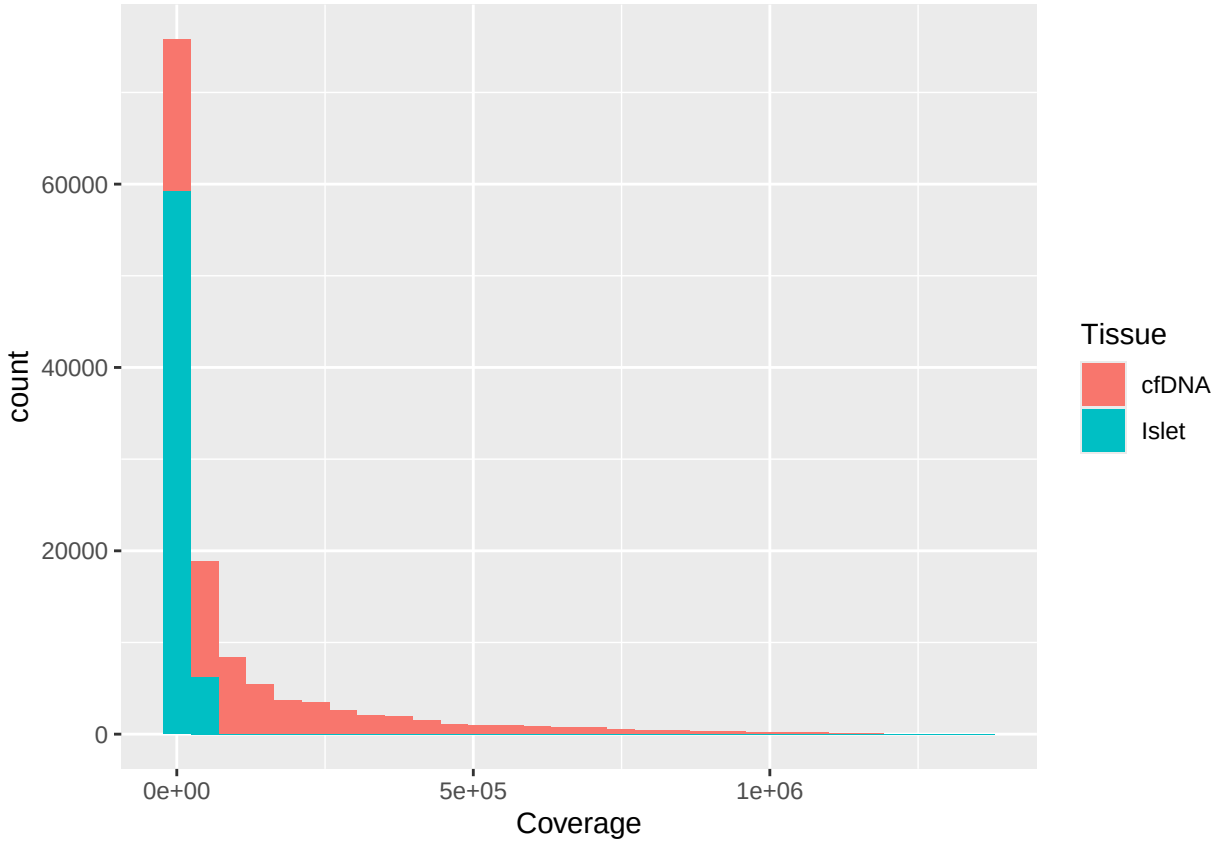
Based on the box plot, it is quite apparent that the cfDNA has much higher coverage than Islet cell.

```
plot_df <- rbind(islet_coverage,cfdna_coverage)
ggplot(plot_df, aes(x=Tissue, y=Coverage, fill=Tissue)) + geom_boxplot()
```



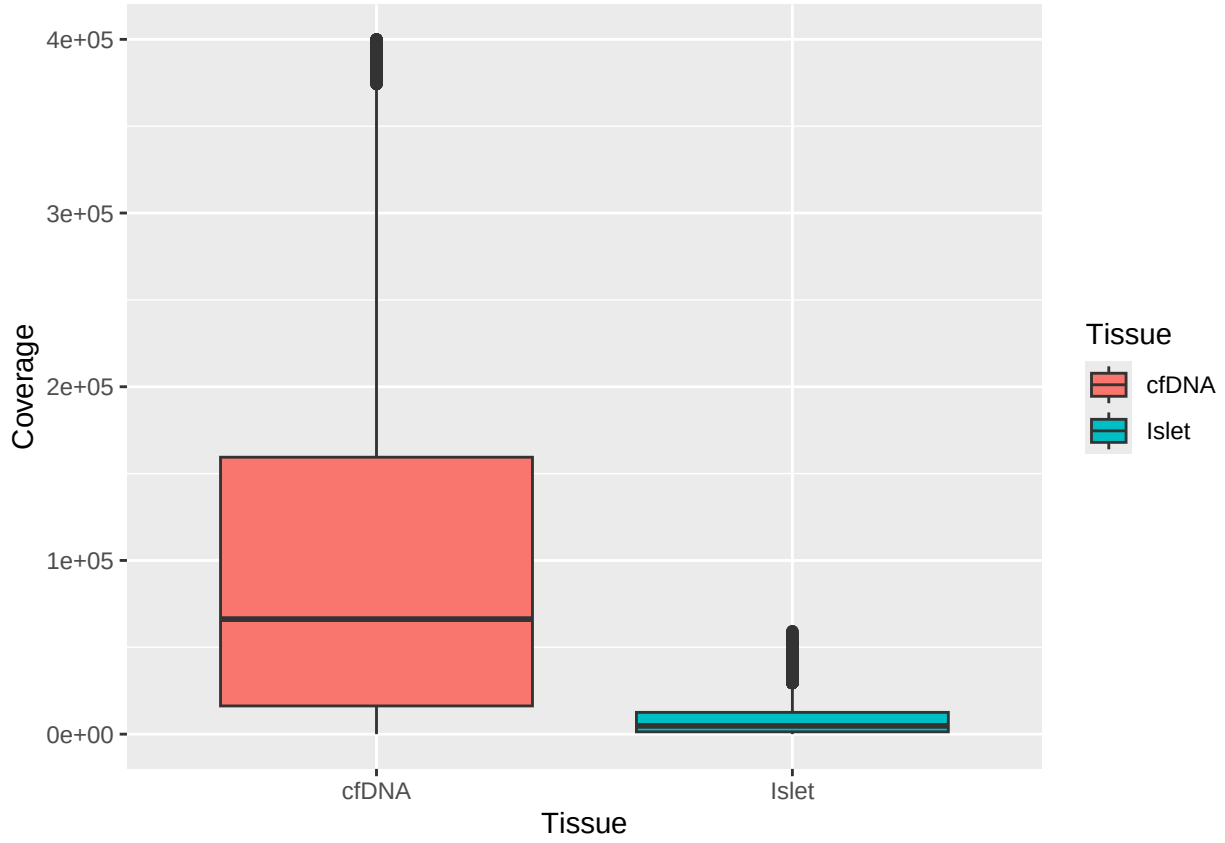

```
ggplot(plot_df, aes(x=Coverage, fill = Tissue)) + geom_histogram()
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

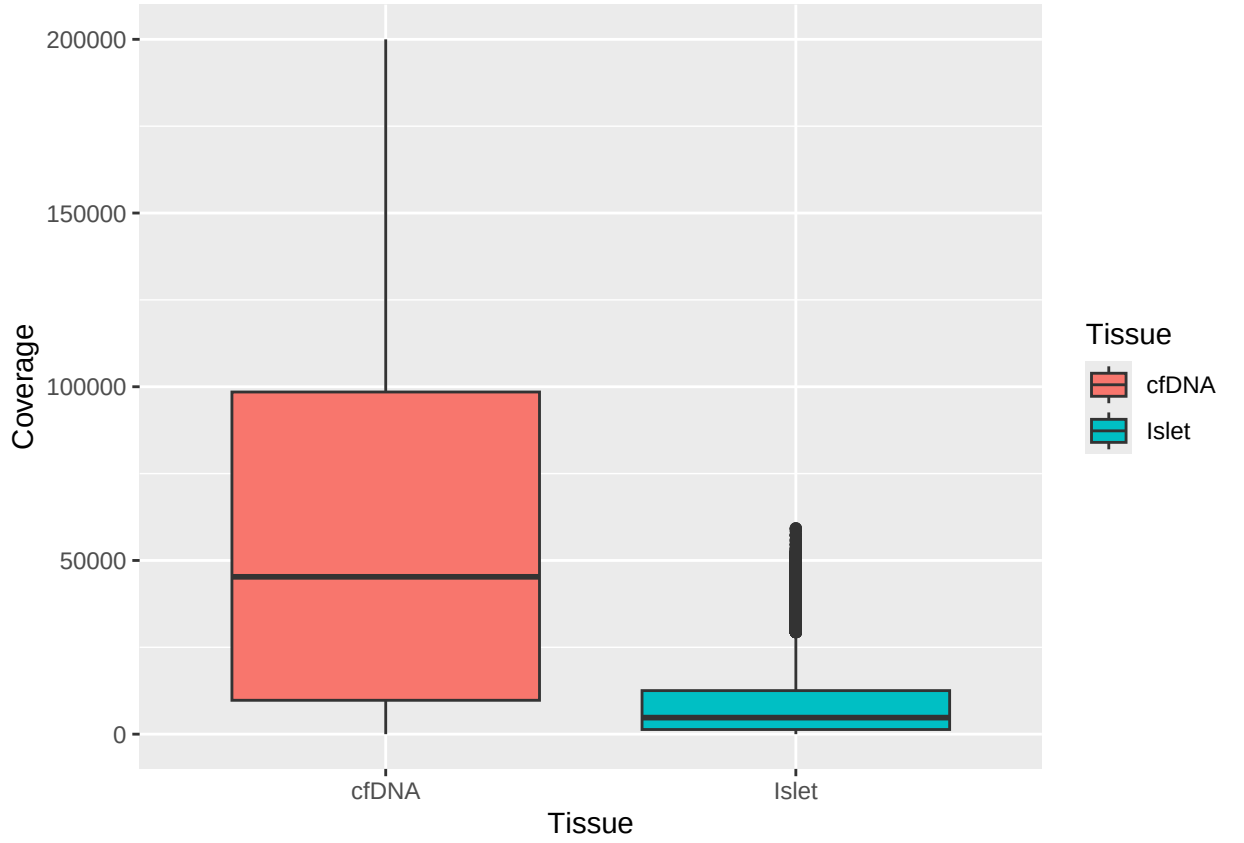



Removing PMP with coverage above 400000 to better visualize the data (by removing the extremes)

```
cfdna_coverage <- cfdna_coverage[cfdna_coverage$Coverage <= 400000,]
plot_df <- rbind(islet_coverage,cfdna_coverage)
ggplot(plot_df, aes(x=Tissue, y=Coverage, fill=Tissue)) + geom_boxplot()
```




```
#Then further decreasing to 200000 to get a granular read.
cfdna_coverage <- cfdna_coverage[cfdna_coverage$Coverage <= 200000,]
plot_df <- rbind(islet_coverage,cfdna_coverage)
ggplot(plot_df, aes(x=Tissue, y=Coverage, fill=Tissue)) + geom_boxplot()
```




A gradient boosting model was used to predict the classification of this dataset using xgboost. Because the dataset is heavily skewed towards cfDNA(0) with 75% of data to Islet cells' 25%, the imbalance is corrected by heavily penalizing errors in the minority class (Islet cell) xgboost (gradient boosting) and thus setting scale_pos_weight to 3.

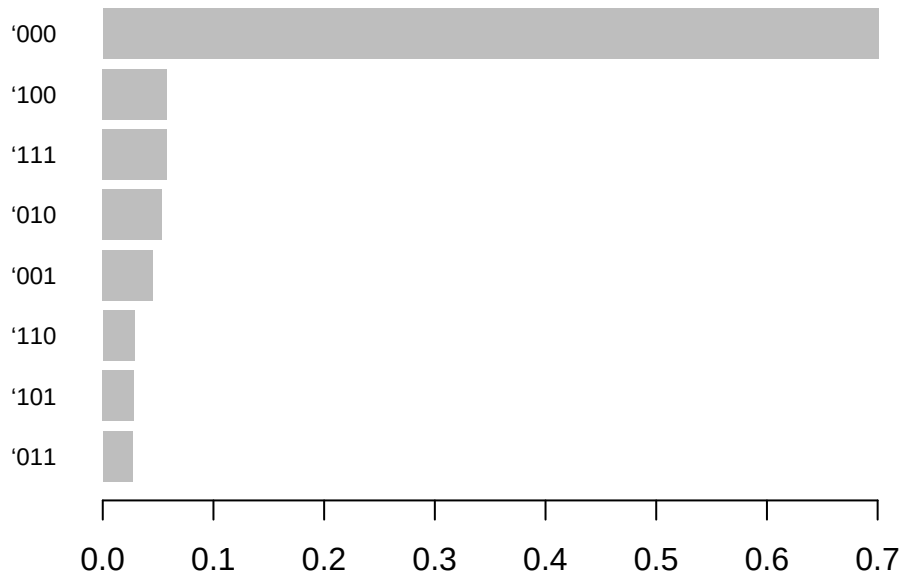
If I had the computational bandwidth, I would have done a k-fold cross validation.

```
#Splitting dataset into test:training in 25:75 ratio at random (and thus setting seed)
pmp$Tissue <- ifelse(pmp$Tissue == "Islet",1,0)
set.seed(234)
split <- sample.split(pmp$Tissue, SplitRatio = 0.75)
training_set <- subset(pmp[,c(3:10,13)], split == TRUE)
training_set_pmp <- subset(pmp[,2], split == TRUE)
test_set <- subset(pmp[,c(3:10,13)], split == FALSE)
test_set_pmp <- subset(pmp[,2], split == FALSE)

# Prepare data matrices
train_matrix <- as.matrix(training_set[,-9])
test_matrix <- as.matrix(test_set[,-9])
train_label <- training_set$Tissue
test_label <- test_set$Tissue


#Gradient Boosting model, prediction and associated statistics
xgb_model <- xgboost(verbose = F,data = train_matrix,
                     label = train_label, objective = "binary:logistic",
                     nrounds = 100, scale_pos_weight = 3)
```

```r
xgb_predictions <- predict(xgb_model, test_matrix)
xgb_rmse <- sqrt(mean((xgb_predictions - test_label)^2))
print(paste("RMSE (XGBoost):", round(xgb_rmse, 2)))
```

```
## [1] "RMSE (XGBoost): 0.4"
```

```r
importance <- xgb.importance(feature_names = colnames(train_matrix), model = xgb_model)
xgb.plot.importance(importance_matrix = importance)
```




```r
saveRDS(xgb_model,"xgb_n100_scaleposweight3.rds")
```

```r
test_set$probability <- xgb_predictions
test_set$prediction <- ifelse(test_set$probability > 0.5,1,0)
test_set$Tissue <- factor(test_set$Tissue)
test_set$prediction <- factor(test_set$prediction)
test_set_pmp <- cbind(test_set_pmp,test_set)
test_set_pmp$coverage <- rowSums(test_set_pmp[,c(2:9)])
write_csv(test_set_pmp,"entire_test_set.csv", col_names = T)
```

```r
confusion_mat <- confusionMatrix(test_set$Tissue,test_set$prediction)
print(confusion_mat)
```

```
## Confusion Matrix and Statistics
```

```
##
##           Reference
## Prediction       0       1
##          0 1886853 1013668
##          1   86350  861175
##
##                Accuracy : 0.7141
##                  95% CI : (0.7137, 0.7146)
##     No Information Rate : 0.5128
##     P-Value [Acc > NIR] : < 2.2e-16
##
##                   Kappa : 0.4208
##
##  Mcnemar's Test P-Value : < 2.2e-16
##
##             Sensitivity : 0.9562
##             Specificity : 0.4593
##          Pos Pred Value : 0.6505
##          Neg Pred Value : 0.9089
##              Prevalence : 0.5128
##          Detection Rate : 0.4903
##    Detection Prevalence : 0.7538
##       Balanced Accuracy : 0.7078
##
##        'Positive' Class : 0
##
```

```
tocsv <- data.frame(cbind(t(confusion_mat$overall),t(confusion_mat$byClass)))
temp <- data.frame(t(tocsv))
temp <- rownames_to_column(temp,"Statistic")
names(temp)[2] <- "Value"
write_csv(temp,file="Statistics_scaleweightpos3_n100.csv",col_names = T)
write_csv(as.data.frame(confusion_mat$table),file="confusionMatrix_scaleweightpos3_n100.csv",col_names =
rm(temp,tocsv)
```

Conclusion: As intended, the FP was minimized while allowing some FN.

```
#This is the actual code, but because of memory issues I'm reading directly from the file
#mean_variant_fraction <- data.frame()
#for (coord in unique(pmp$CpG_Coordinates)) {
#  each_pmp <- pmp[pmp$CpG_Coordinates == coord,]
#  total_reads_per_variant <- colSums(each_pmp[,c(3:10)])
#  total_reads <- sum(total_reads_per_variant)
#  muf <- total_reads_per_variant/total_reads
#  each_pmp <- c(CpG_Coordinates = coord,muf)
#  mean_variant_fraction <- rbind(mean_variant_fraction,each_pmp)
#}
#colnames(mean_variant_fraction) <- c("CpG_Coordinates","X000","X001","X010","X011","X100","X101","X110
#mean_variant_fraction[,c(2:9)] <- apply(mean_variant_fraction[,c(2:9)],2,as.numeric)
#write_csv(mean_variant_fraction,"mean_variant_fraction.csv", col_names = T)
mean_variant_fraction <- read_csv("mean_variant_fraction.csv", col_names = T)
```

```
## Rows: 66023 Columns: 9
```

```
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (1): CpG_Coordinates
## dbl (8): X000, X001, X010, X011, X100, X101, X110, X111
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
summary(mean_variant_fraction)
```

```
##  CpG_Coordinates         X000              X001                X010
##  Length:66023       Min.   :0.0000   Min.   :0.000000   Min.   :0.000000
##  Class :character   1st Qu.:0.9339   1st Qu.:0.002882   1st Qu.:0.002430
##  Mode  :character   Median :0.9611   Median :0.006996   Median :0.006555
##                     Mean   :0.9099   Mean   :0.014557   Mean   :0.013299
##                     3rd Qu.:0.9802   3rd Qu.:0.015710   3rd Qu.:0.015376
##                     Max.   :1.0000   Max.   :0.711809   Max.   :1.000000
##       X011                X100               X101                X110
##  Min.   :0.0000000   Min.   :0.000000   Min.   :0.0000000   Min.   :0.0000000
##  1st Qu.:0.0001647   1st Qu.:0.002981   1st Qu.:0.0000244   1st Qu.:0.0002078
##  Median :0.0008596   Median :0.007526   Median :0.0004146   Median :0.0009967
##  Mean   :0.0071307   Mean   :0.016238   Mean   :0.0042757   Mean   :0.0079000
##  3rd Qu.:0.0026599   3rd Qu.:0.018452   3rd Qu.:0.0014495   3rd Qu.:0.0027933
##  Max.   :0.5938251   Max.   :1.000000   Max.   :0.5448113   Max.   :0.8750000
##       X111
##  Min.   :0.000000
##  1st Qu.:0.001375
##  Median :0.003285
##  Mean   :0.026744
##  3rd Qu.:0.009384
##  Max.   :0.838115
```

3a) Specificity is a measure of true negative (prediction accuracy of Tissue #2). In this case, since Tissue 2 has significantly low coverage, our ability to get a more accurate specificity of the minor class (at 25%) was far lower at 0.4

3b) There are a total of 66023 unique CpG coordinates in the dataset. Since Tissue #2 (Islet cells) is the minority class at 25% of data, 25% of that is 16505 CpG coordinates. 1 million reads would mean Tissue #2 has approximately 250,000 reads.

Coverage_per_target = Total Reads/NUmber of Targets These 250,000 reads are distributed to across only 16505 PMP's would mean a depth of 15 per CpG.

Therefore the threshold is 15 reads per biomarker.

3c) The specificity hypothesis

```
index <- head(order(test_set_pmp$probability,decreasing = T),n = 10)
top10_cfdna <- test_set_pmp[index,]
index <- head(order(test_set_pmp$probability,decreasing = F),n = 10)
top10_islet <- test_set_pmp[index,]
summary(top10_cfdna[,c(11,13)])
```

```
##   probability        coverage
```

```
##  Min.   :0.9969   Min.   : 39.0
##  1st Qu.:0.9971   1st Qu.:113.0
##  Median :0.9972   Median :133.0
##  Mean   :0.9972   Mean   :197.1
##  3rd Qu.:0.9972   3rd Qu.:314.8
##  Max.   :0.9975   Max.   :425.0
```

```
summary(top10_islet[,c(11,13)])
```

```
##   probability           coverage
##  Min.   :1.774e-10   Min.   :3437
##  1st Qu.:4.554e-10   1st Qu.:5397
##  Median :5.483e-10   Median :5504
##  Mean   :4.974e-10   Mean   :5412
##  3rd Qu.:5.878e-10   3rd Qu.:5706
##  Max.   :6.073e-10   Max.   :6980
```

As depicted by the boxplot earlier and the relative coverage of Islet cells and cfDNA, higher coverage leads to more certainity and thus higher specificity.