

Rauen William Final

Will Rauen

2024-04-23

Setup

For this investigation, we will be using a multitude of packages to answer a research question related to logistic regression.

Data

```
bill_df <- read.csv("/Users/williamrauen/Desktop/DS 3100/Final/Data/bill_data.csv")  
  
dim(bill_df)
```

```
## [1] 1500    7
```

The data for this investigation comes from Kaggle.com and can be accessed at this link:

https://www.kaggle.com/datasets/alexandrepetit881234/fake-bills?select=fake_bills.csv

This data set contains 1500 rows and 7 columns of data pertaining to whether or not a bill/legal tender is real or counterfeit. The first column gives TRUE/FALSE values to indicate whether bills are real (TRUE) or counterfeit (FALSE) and will be the dependent variable of our investigation. The other 6 columns are quantitative values of the bill in question that are the measurements of its diagonal, left side height, right side height, length of the lower margin of bill, length of the upper margin of bill, and length of the base of the bill. These are all measured in millimeters in the data set. Furthermore, there is some missingness in the data set so we will need to wrangle for missing data. Thus, we have a dataset with 1 variable that is categorical, and 6 variables that are quantitative, making this the perfect candidate for logistic regression analysis.

Variables of Interest

The dependent variable of interest as aforementioned is whether or not the bills are legal tender or counterfeit. This is recorded in the data set as a categorical logical operator in the data where a 'TRUE' indicates the bill is legal tender and 'FALSE' indicates the bill is counterfeit. The independent variables of interest for the investigation are continuous variables measured in millimeters that correspond to physical attributes of the bills in question. These measurements pertain to the bills diagonal lengths, side lengths, base lengths, and margin lengths. Thus, we have 1 dependent variable and 6 independent variables.

Research Question

In this assignment, I aim to determine if there are any statistically significant features of a bill that can help classify it as legal tender or counterfeit. If so, I then also wish to determine what qualities of these bills indicate the classification.

Data Wrangling

Firstly, to move forward with a logistic regression, we need to code the dependent variable as zeros and ones. In this case, we will code “TRUE” (legal tender) as 1 and “FALSE” (counterfeit) as 0.

```
# Coding dependent variable
bill_df <- bill_df %>%
  mutate(is_genuine = ifelse(is_genuine == "TRUE", 1, 0))
```

Next, we will convert the dependent variables into a factor variable.

```
# Make Dependent Variable a Factor
bill_df <- bill_df %>%
  mutate(is_genuine = as.factor(is_genuine))
```

Lastly, we will get rid of rows with missing data values. We find that there were 37 rows of missing data.

```
# remove missingness from data set
bill_df <- bill_df[complete.cases(bill_df), ]

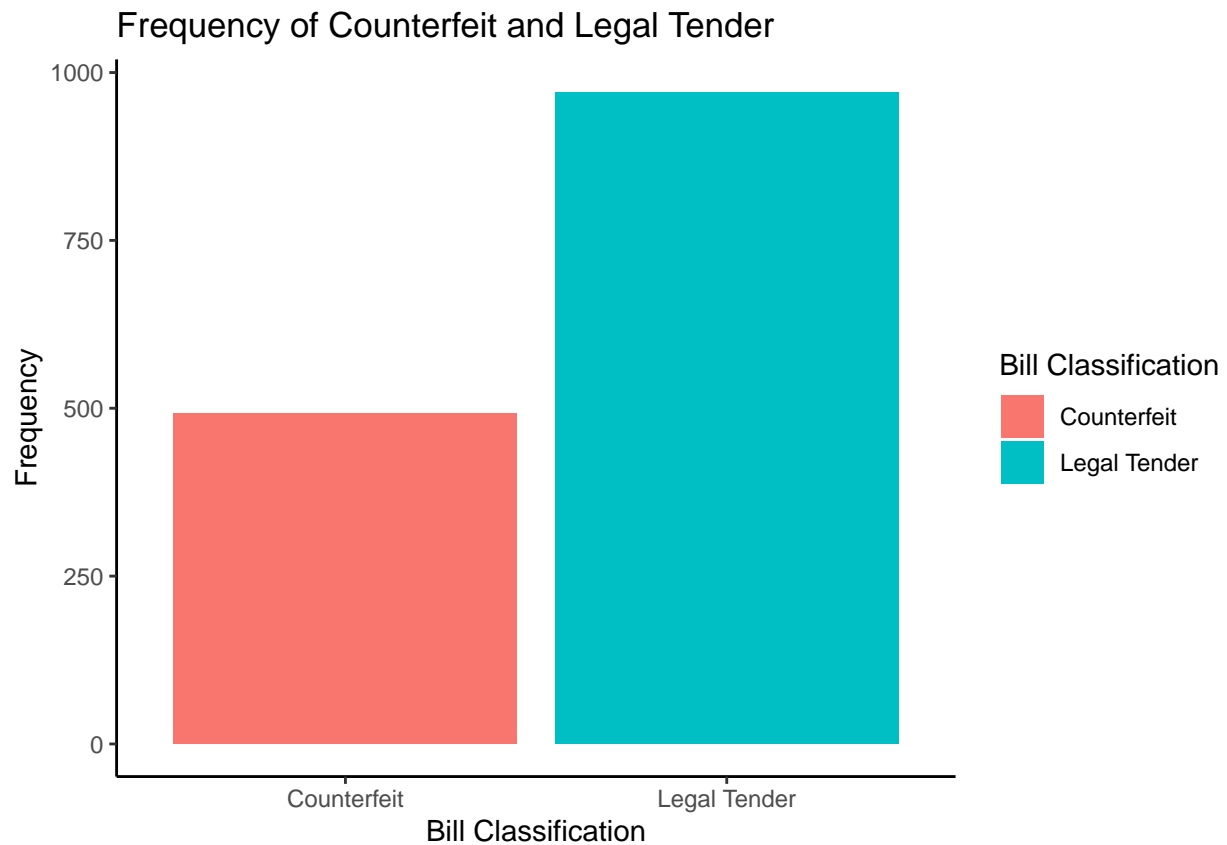
dim(bill_df)
```

```
## [1] 1463    7
```

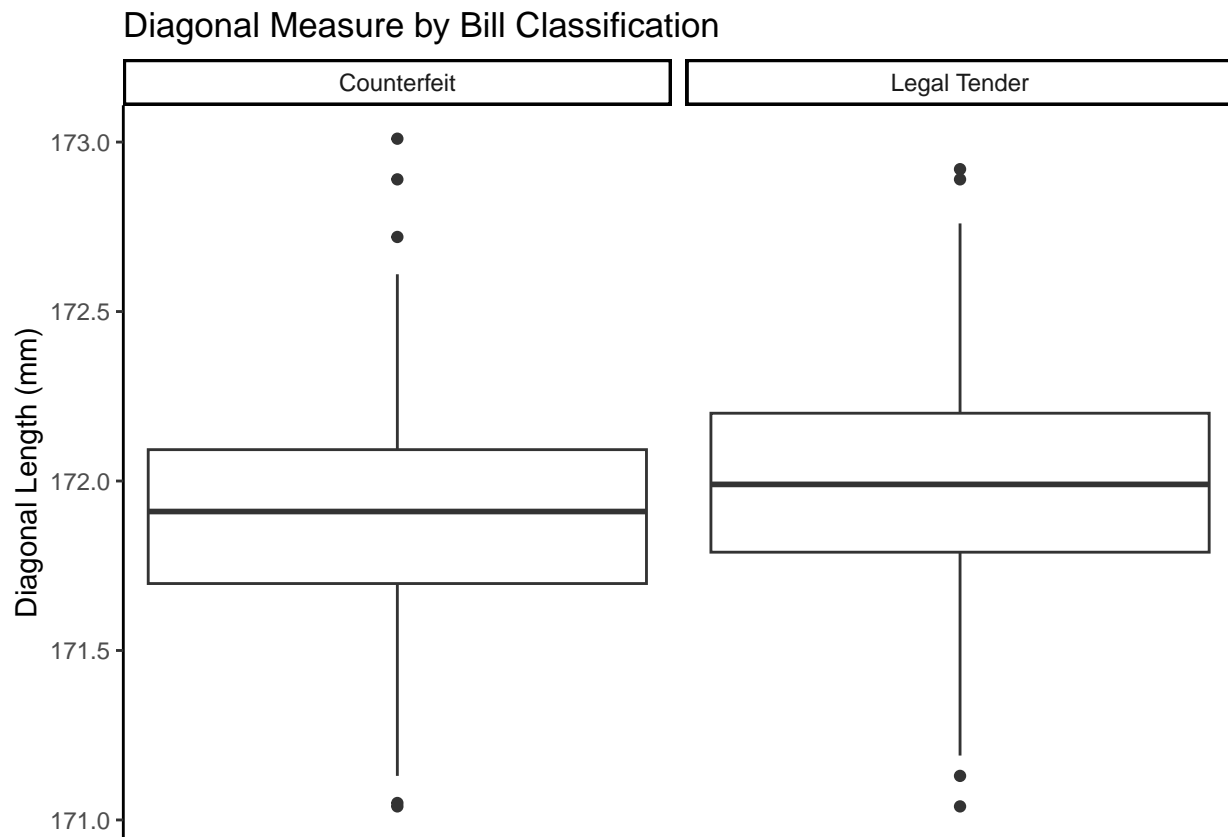
Now we can create some data visualizations to see if there are any blatant and apparent relationships between the dependent and independent variables.

```
#Labels for data
custom_labels <- labeller(is_genuine = c("0" = "Counterfeit",
                                           "1" = "Legal Tender"))

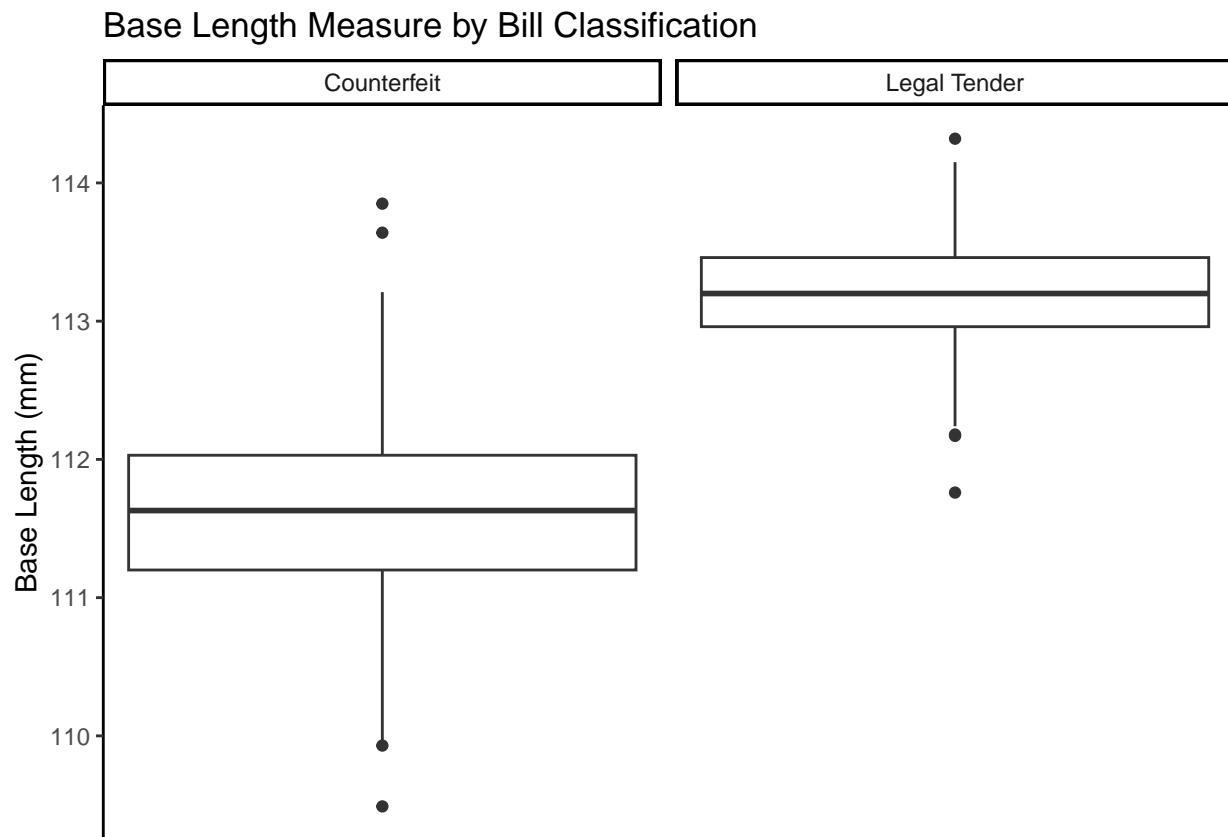
# Plot of count of each Dependent Variable
bill_df %>%
  mutate(is_genuine = ifelse(is_genuine == "1", "Legal Tender", "Counterfeit")) %>%
  ggplot(aes(x=is_genuine, fill = factor(is_genuine))) +
  geom_bar() +
  theme_classic() +
  labs(title = "Frequency of Counterfeit and Legal Tender",
       x = "Bill Classification",
       y = "Frequency",
       fill = "Bill Classification")
```



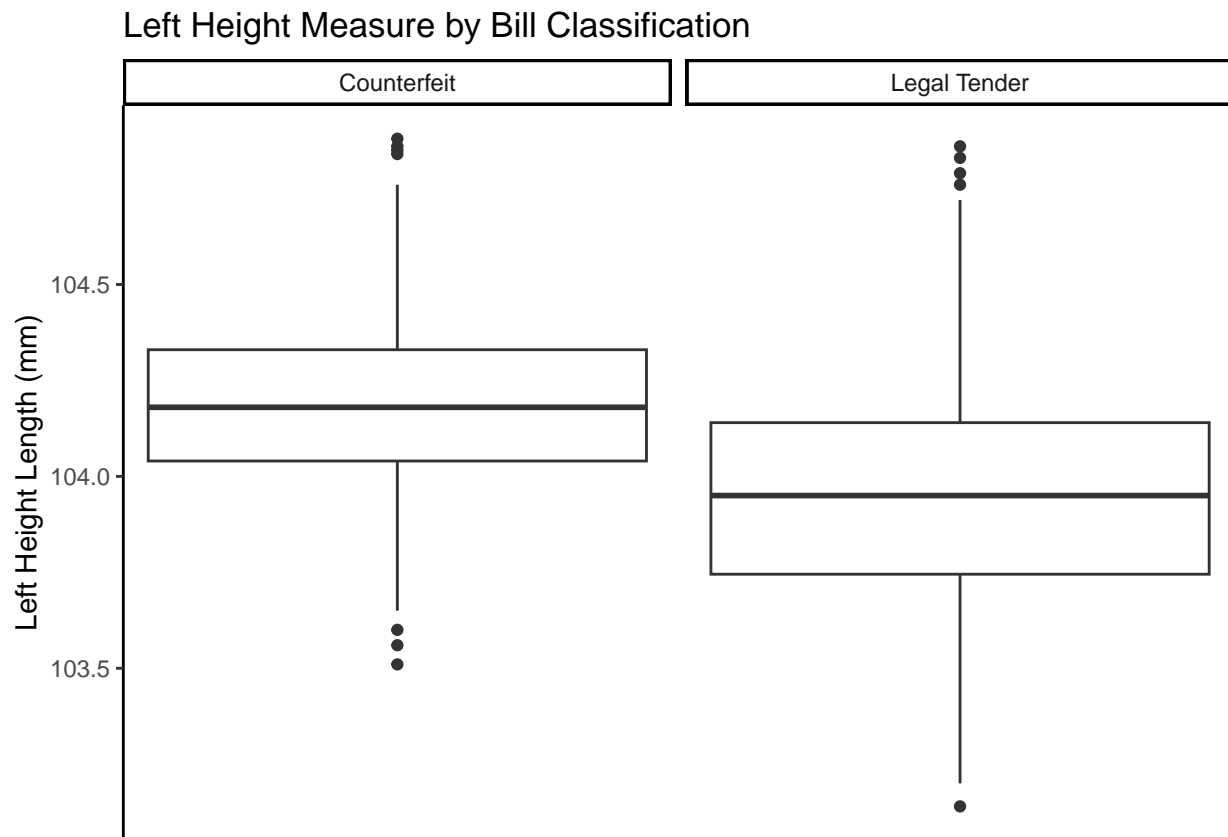
```
# Boxplot of Diagonals by Bill Classification
bill_df %>%
  ggplot(aes(x = diagonal)) +
  geom_boxplot() +
  facet_wrap(~is_genuine, labeller = custom_labels) +
  coord_flip() +
  theme_classic() +
  theme(axis.text.x = element_blank(),
        axis.ticks.x = element_blank()) +
  labs(title = 'Diagonal Measure by Bill Classification',
        x = 'Diagonal Length (mm)')
```



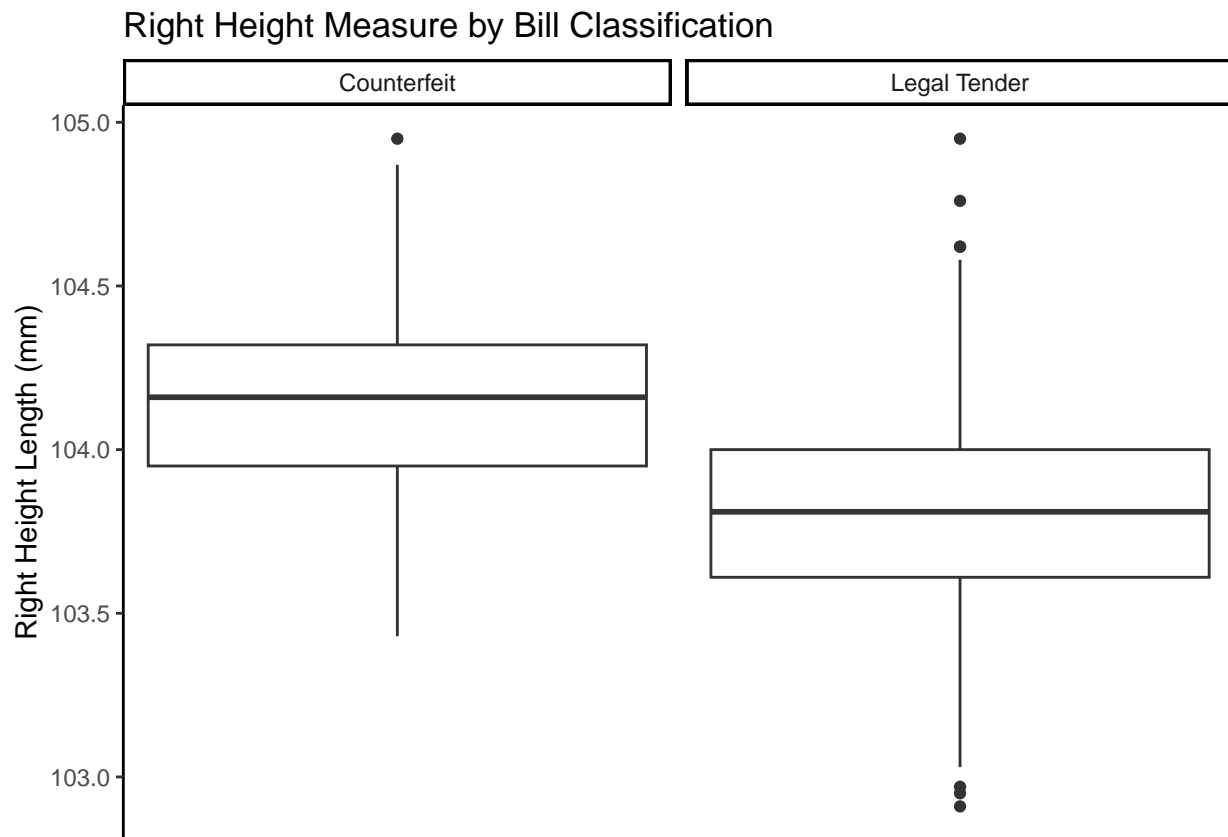
```
# Boxplot of Lengths by Bill Classification
bill_df %>%
  ggplot(aes(x = length)) +
  geom_boxplot() +
  facet_wrap(~is_genuine, labeller = custom_labels) +
  coord_flip() +
  theme_classic() +
  theme(axis.text.x = element_blank(),
        axis.ticks.x = element_blank()) +
  labs(title = 'Base Length Measure by Bill Classification',
       x = 'Base Length (mm)')
```



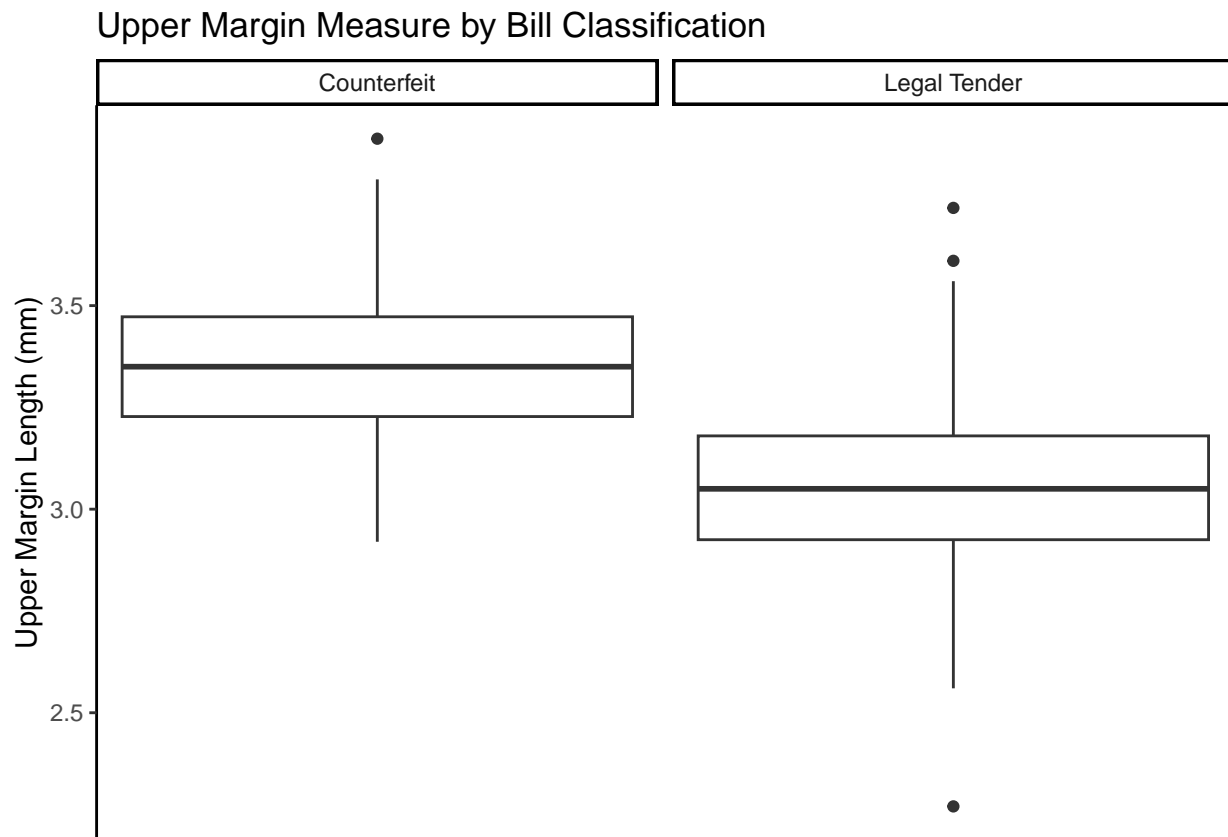
```
# Boxplot of Left Height by Bill Classification
bill_df %>%
  ggplot(aes(x = height_left)) +
  geom_boxplot() +
  facet_wrap(~is_genuine, labeller = custom_labels) +
  coord_flip() +
  theme_classic() +
  theme(axis.text.x = element_blank(),
        axis.ticks.x = element_blank()) +
  labs(title = 'Left Height Measure by Bill Classification',
       x = 'Left Height Length (mm)')
```



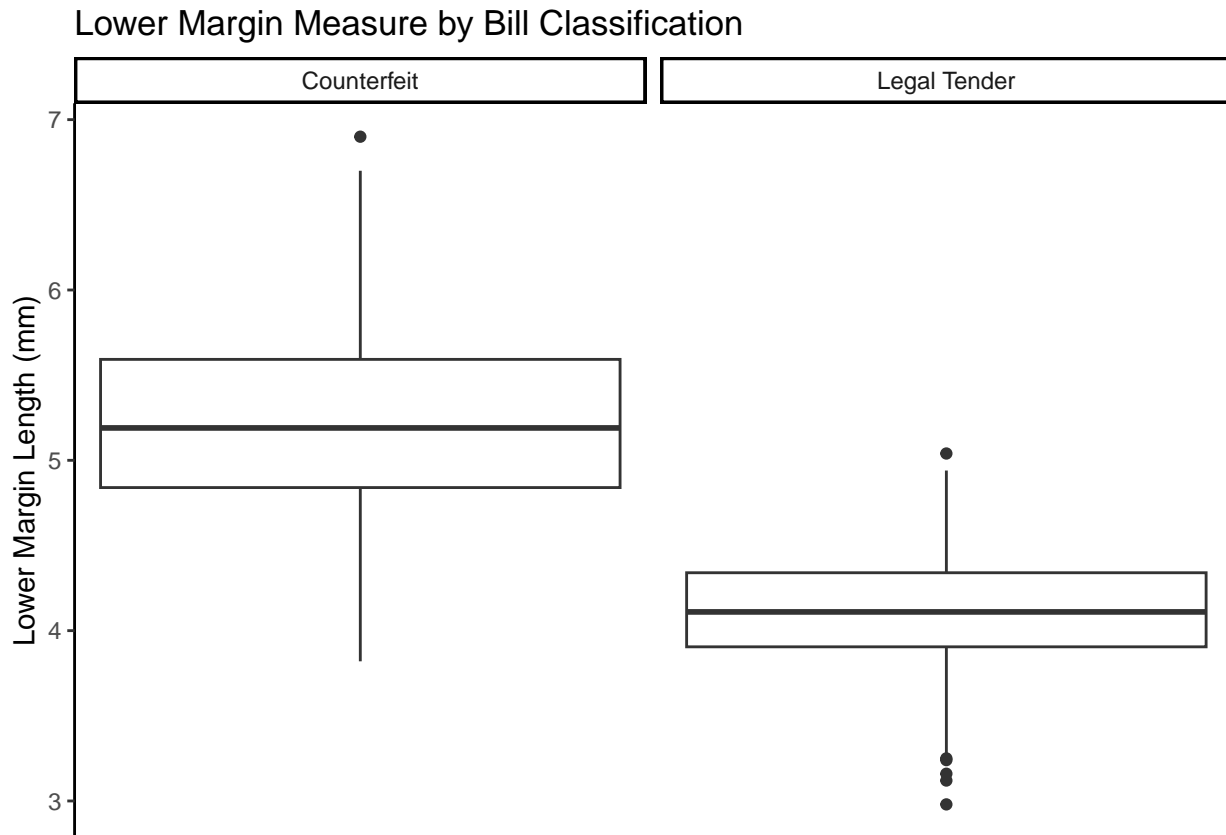
```
# Boxplot of Right Height by Bill Classification
bill_df %>%
  ggplot(aes(x = height_right)) +
  geom_boxplot() +
  facet_wrap(~is_genuine, labeller = custom_labels) +
  coord_flip() +
  theme_classic() +
  theme(axis.text.x = element_blank(),
        axis.ticks.x = element_blank()) +
  labs(title = 'Right Height Measure by Bill Classification',
        x = 'Right Height Length (mm)')
```



```
# Boxplot of Upper Margin Length by Bill Classification
bill_df %>%
  ggplot(aes(x = margin_up)) +
  geom_boxplot() +
  facet_wrap(~is_genuine, labeller = custom_labels) +
  coord_flip() +
  theme_classic() +
  theme(axis.text.x = element_blank(),
        axis.ticks.x = element_blank()) +
  labs(title = 'Upper Margin Measure by Bill Classification',
        x = 'Upper Margin Length (mm)')
```



```
# Boxplot of Lower Margin Length by Bill Classification
bill_df %>%
  ggplot(aes(x = margin_low)) +
  geom_boxplot() +
  facet_wrap(~is_genuine, labeller = custom_labels) +
  coord_flip() +
  theme_classic() +
  theme(axis.text.x = element_blank(),
        axis.ticks.x = element_blank()) +
  labs(title = 'Lower Margin Measure by Bill Classification',
        x = 'Lower Margin Length (mm)')
```

Our visualizations indicate that the main differentiators of bill classification might be found in the bills base length measurement and lower margin measurement, so we should look out for these variables as we conduct our analysis. Moreover, the boxplots revealed that there are outliers in the dataset so we should move forward with some outlier detection to see if those rows of data are worth keeping. Nonetheless, since there were some visual differences by bill classification, we can be more confident in assuming logistic regression is an appropriate model for analysis.

Data Analysis

An initial logistic regression model would look something like this:

```
# Creating Logistic Model
glm1_bill <- glm(is_genuine ~ diagonal + height_left + height_right +
  margin_low + margin_up + length,
  data = bill_df,
  family = binomial(link='logit'))

summary(glm1_bill)

##
## Call:
## glm(formula = is_genuine ~ diagonal + height_left + height_right +
##     margin_low + margin_up + length, family = binomial(link = "logit"),
##     data = bill_df)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
```

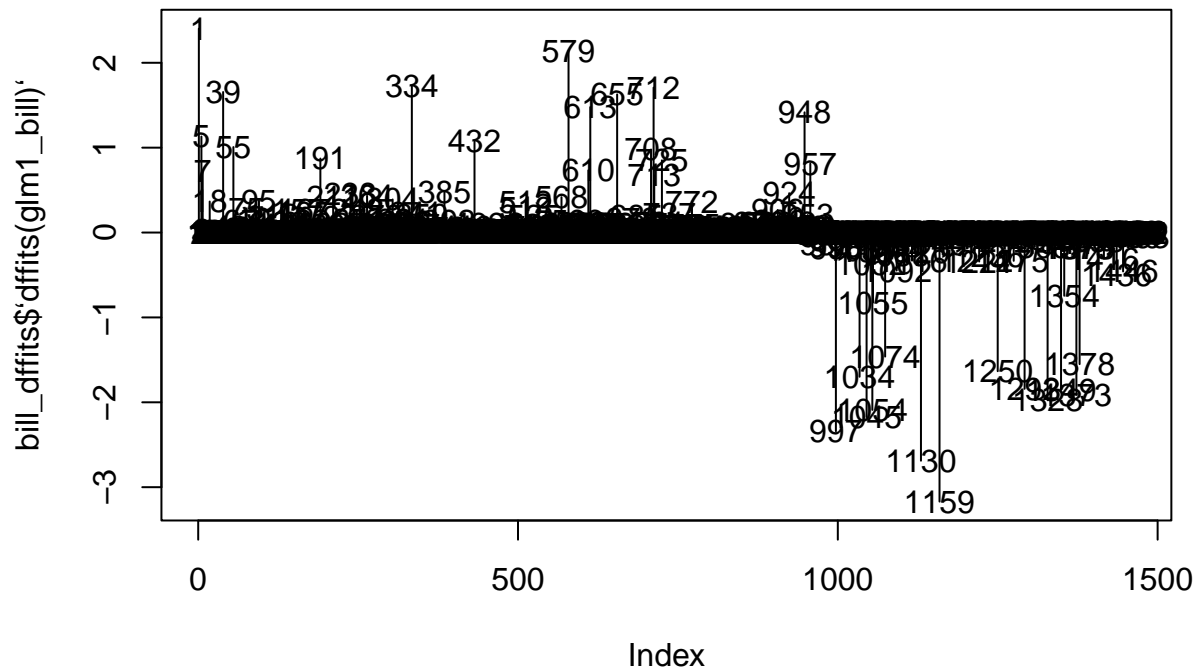
```
## -3.9476 -0.0009 0.0047 0.0273 2.7849
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -243.4159   249.3696  -0.976  0.3290
## diagonal      0.4029    1.1594   0.348  0.7282
## height_left  -1.1602    1.1247  -1.032  0.3023
## height_right -3.1087    1.2241  -2.540  0.0111 *
## margin_low   -5.9078    0.9830  -6.010 1.86e-09 ***
## margin_up   -10.2367    2.1987  -4.656 3.23e-06 ***
## length        6.0242    0.8925   6.750 1.48e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 1868.390  on 1462  degrees of freedom
## Residual deviance:   78.315  on 1456  degrees of freedom
## AIC: 92.315
##
## Number of Fisher Scoring iterations: 10
```

At the moment, our initial logistic regression model reveals 4 significant predictor variables in classifying bills as either counterfeit or legal tender. These variables are: height_right, margin_low, margin_up, and length.

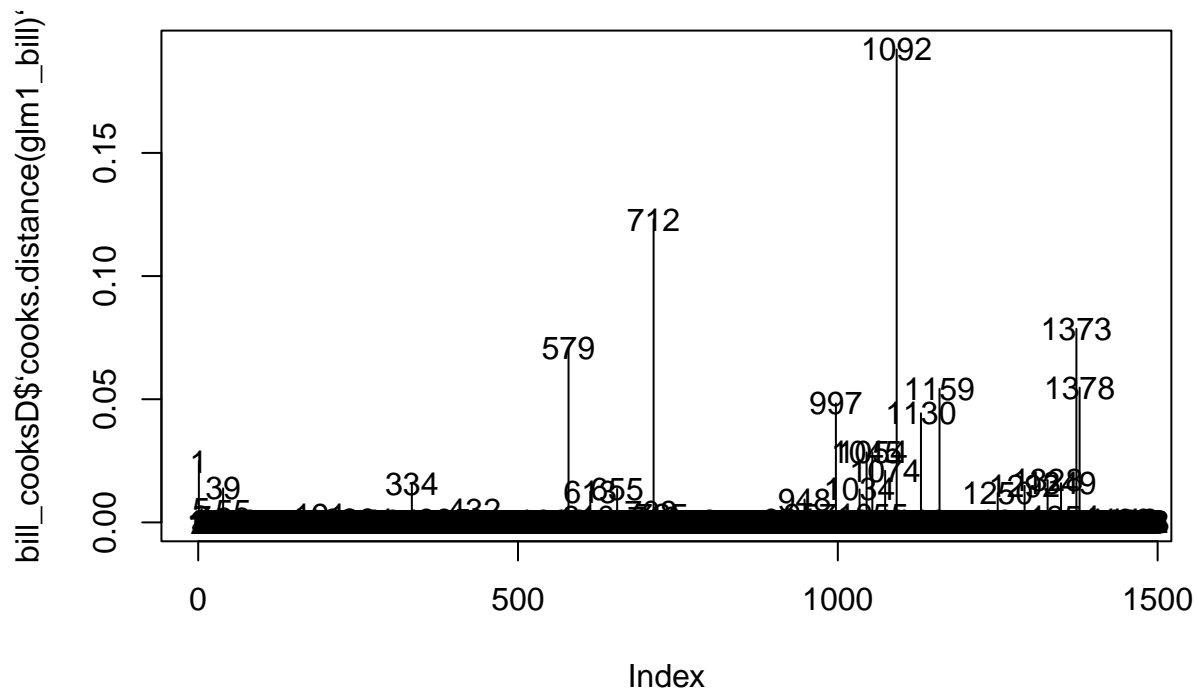
Outlier Detection

Before we move forward with feature selection, we must make sure to remove outliers that might significantly influence our data. Using DFBETAS, Cook's Distance, and DFFITS, we can try to identify possible outliers in the data set. When using DFBETAS, I will only consider the variables that have been shown to be statistically significant in the initial logistic regression model.

```
#DFFITS
bill_dffits <- as.data.frame(dffits(glm1_bill))
plot(bill_dffits$dffits(glm1_bill)~, type='h')
text(bill_dffits$dffits(glm1_bill)~,
      row.names(bill_dffits$dffits(glm1_bill)~))
```



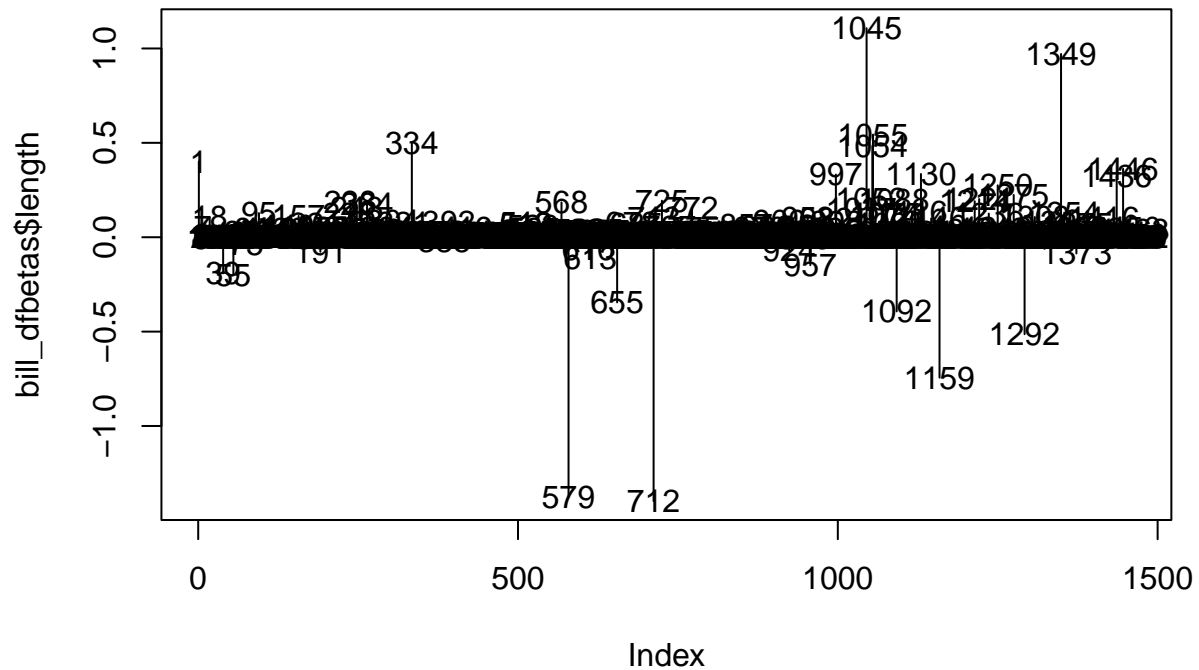
```
#COOKS DISTANCE
bill_cooksD <- as.data.frame(cooks.distance(glm1_bill))
plot(bill_cooksD$`cooks.distance(glm1_bill)`, type='h')
text(bill_cooksD$`cooks.distance(glm1_bill)`,
      row.names(bill_cooksD$`cooks.distance(glm1_bill)`))
```



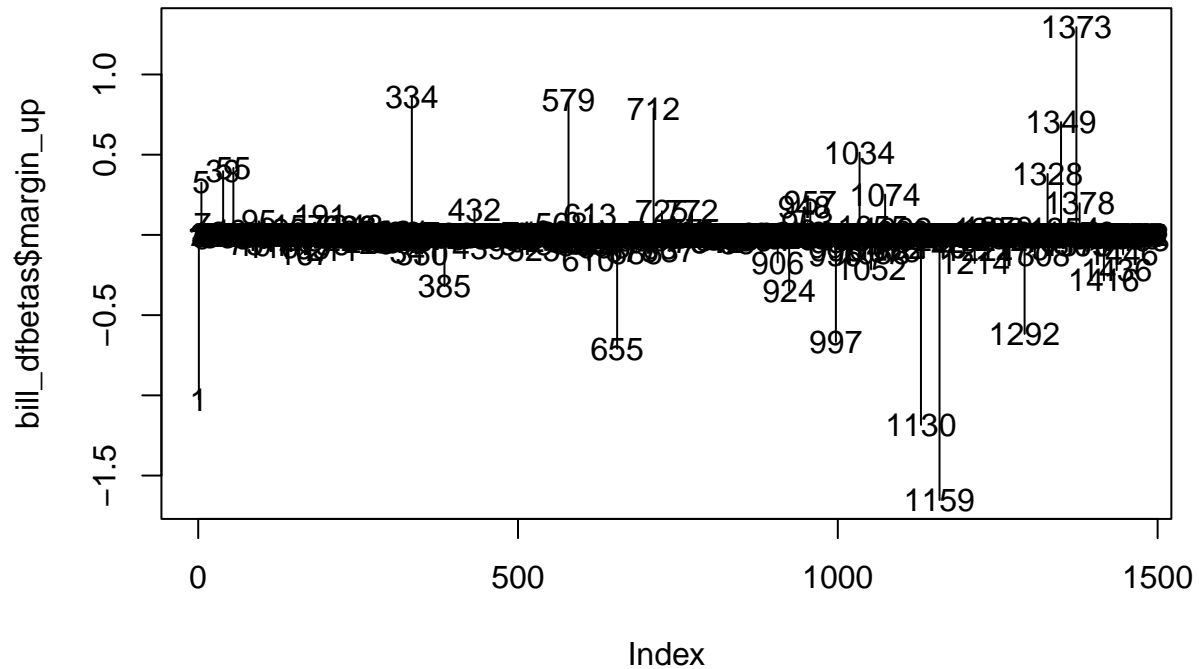
```
#DFBETAS
bill_dfbetas <- as.data.frame(dfbetas(glm1_bill))

plot(bill_dfbetas$`length`, type='h')
text(bill_dfbetas$`length`,
```

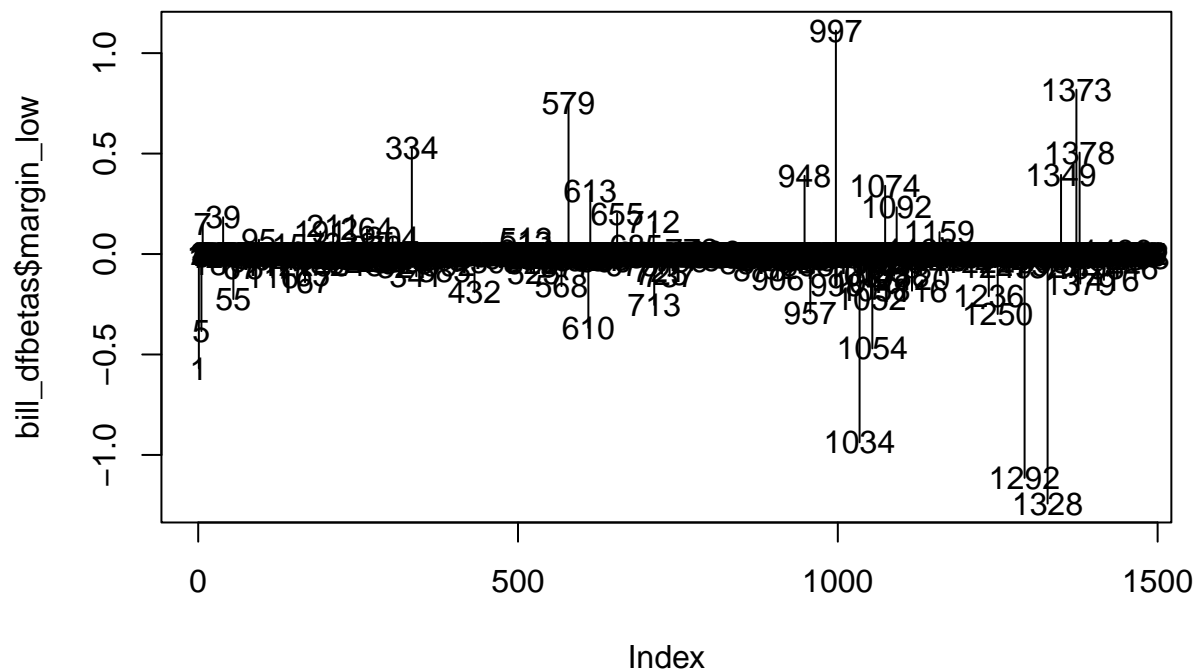
```
row.names(bill_dfbetas$`length`))
```



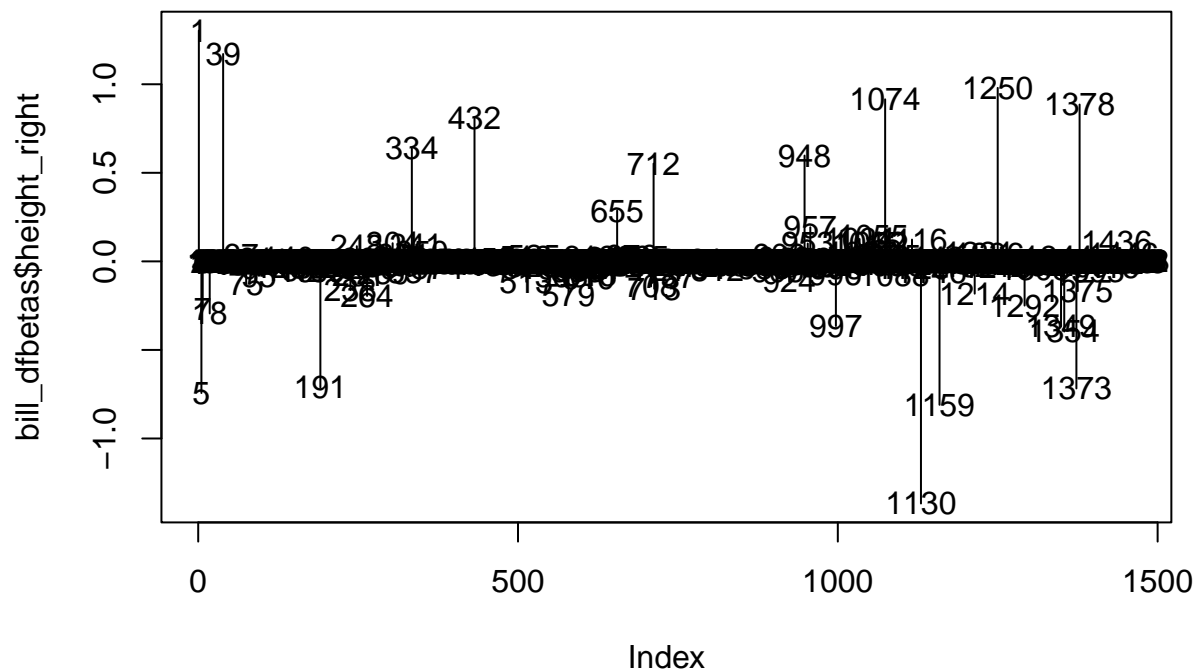
```
plot(bill_dfbetas$`margin_up`, type = 'h')
text(bill_dfbetas$`margin_up`,
      row.names(bill_dfbetas$`margin_up`))
```



```
plot(bill_dfbetas$`margin_low`, type = 'h')
text(bill_dfbetas$`margin_low`,
      row.names(bill_dfbetas$`margin_low`))
```



```
plot(bill_dfbetas$`height_right`, type = 'h')
text(bill_dfbetas$`height_right`,
     row.names(bill_dfbetas$`height_right`))
```



From the visualizations of outlier measurements, it appears that significant and consistent outliers might be found at row 1159, 712, and 1. Since it is only 3 outliers, it would be safe to remove them from our data set without influencing our data too much considering the sample is greater than 1000.

```
#Subset data to remove outliers
bill_df <- bill_df[c(-1,-712,-1159), ]
```

Generalizability

To move forward with our model prediction and investigation, we will create training and testing data to see the generalizability and accuracy of our model.

```
set.seed(051204) # set seed for randomness

train_index <- sample(1:nrow(bill_df), round(nrow(bill_df) * 0.70), # 70%
                     replace = FALSE)

#Using train_index to denote rows to retain
#We create training and testing datasets
bill_train <- bill_df[train_index,]
bill_test  <- bill_df[-train_index,]
```

Feature Selection

```
# Creating new logistic model with training data
glm_bill_train1 <- glm(is_genuine ~ diagonal + height_left + height_right +
                      margin_low + margin_up + length,
                      data = bill_train,
                      family = binomial(link='logit'))

summary(glm_bill_train1)
```

```
##
## Call:
## glm(formula = is_genuine ~ diagonal + height_left + height_right +
##      margin_low + margin_up + length, family = binomial(link = "logit"),
##      data = bill_train)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -4.3564  -0.0008   0.0037   0.0258   1.0476
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -619.6994   364.2310  -1.701  0.08887 .
## diagonal       0.9701     1.6176   0.600  0.54870
## height_left   -0.4468     1.4314  -0.312  0.75493
## height_right  -2.4018     1.5144  -1.586  0.11275
## margin_low    -5.4751     1.2187  -4.493 7.04e-06 ***
## margin_up     -8.9604     2.8294  -3.167  0.00154 **
## length         7.1336     1.3564   5.259 1.45e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 1320.007  on 1021  degrees of freedom
## Residual deviance:   49.729  on 1015  degrees of freedom
## AIC: 63.729
##
## Number of Fisher Scoring iterations: 10
```

Once again, this model tells us that we have the same four significant features as found before: height_right, margin_low, margin_up, and length. From here, we can create another new logistic model with those variables, and once again check that the same variables are significant.

```
# Creating Another logistic model with training data
glm_bill_train2 <- glm(is_genuine ~ height_right + margin_low + margin_up + length,
                      data = bill_train,
                      family = binomial(link='logit'))

summary(glm_bill_train2)

##
## Call:
## glm(formula = is_genuine ~ height_right + margin_low + margin_up +
##      length, family = binomial(link = "logit"), data = bill_train)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -4.3805  -0.0007   0.0034   0.0246   1.0941
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -484.354    205.438  -2.358   0.0184 *
## height_right   -2.649     1.388  -1.908   0.0564 .
## margin_low     -5.689     1.192  -4.772 1.83e-06 ***
## margin_up      -8.953     2.743  -3.264  0.0011 **
## length         7.237     1.360   5.321 1.03e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 1320.007  on 1021  degrees of freedom
## Residual deviance:   50.176  on 1017  degrees of freedom
## AIC: 60.176
##
## Number of Fisher Scoring iterations: 10
```

This model shows that all the aforementioned variables are statistically significant, so we can move forward with our analysis.

Another aspect of the prediction variables we can quickly check is if any of them are collinear. Using the car package we can measure variable multicollinearity by finding each of the variable's Variation Inflation Factor (VIF). Any value above 10 means we should be wary of collinearity.

```
#Testing for Multicollinearity using Variation Inflation Factor

vif(glm_bill_train2)
```

```
## height_right  margin_low  margin_up    length
##      1.014891    1.313084    1.150808    1.193161
```

None of the VIF scores indicate that there are variables that are collinear with one another as there are no VIF values above 10. Thus, we cannot immediately remove any features at the moment.

Nonetheless, it is safe to say that our model is appropriate for conducting linear regression.

Odds Interpretations and Marginal Effects

As a reminder, the meaning of our dichotomous variables are as follows:

0: Counterfeit 1: Legal Tender

```
# Odds Ratio
```

```
bill_coef <- coef(glm_bill_train2)
```

```
bill_coef
```

```
## (Intercept) height_right margin_low margin_up length
## -484.353539 -2.648720 -5.689375 -8.953015 7.236856
```

```
# Log Odds Ratio
```

```
exp(bill_coef)
```

```
## (Intercept) height_right margin_low margin_up length
## 4.445605e-211 7.074172e-02 3.381706e-03 1.293466e-04 1.389718e+03
```

The odds ratio for the height_right, margin_low, and margin_up are all below 1 which indicate that an increase in these variables decrease the odds of a bill being legal tender, assuming we hold all other variables constant. For example, for every one unit increase in the height_right variable, the odds of being legal tender decreases by a factor of 0.015. For every one unit increase in the margin_low variable, the odds of being legal tender decreases by a factor of 0.004. For every one unit increase in the margin_up variable, the odds of being legal tender decreases by a factor of 0.00004. The only variable that indicate an increase in the odds of being legal tender when its value increases by a unit of the variable is the length variable. For every one unit increase in the length variable, the odds of the bill being legal tender increases by a factor of ~200. The log odds ratio follows similar logic for the increase and decrease in odds of a bill being classified as legal tender assuming you hold all other variables constant.

Using the margins and DescTools packages, we can also find the McFadden R squared value along with the average marginal effects.

```
# Finds average marginal effects
```

```
marg_bill <- margins(glm_bill_train2)
```

```
summary(marg_bill)
```

```
## factor AME SE z p lower upper
## height_right -0.0171 0.0089 -1.9264 0.0541 -0.0345 0.0003
## length 0.0469 0.0082 5.7220 0.0000 0.0308 0.0630
## margin_low -0.0368 0.0071 -5.1737 0.0000 -0.0507 -0.0228
## margin_up -0.0579 0.0181 -3.2005 0.0014 -0.0933 -0.0224
```

```
# Finds McFadden's' R Squared Score
```

```
PseudoR2(glm_bill_train2)
```

```
## McFadden
```

```
## 0.9619878
```

Once more, the average marginal effects reaffirm what was shown in the odds and log odds ratio and are the closest measures to slope values we have. These values can be interpreted as: For every one unit increase of length, the odds of the bill being legal tender increases by 4.68%, for every one unit increase in height_right, the odds of the bill being legal tender decreases by 3.61%, for every one unit increase in margin_low, the odds of the bill being legal tender decreases by 4.8%, and for every one unit increase in margin_up, the odds of the bill being legal tender decreases by 8.84%. The McFadden R-Square value was found to be 0.95 which indicates that our model does a good job explaining variability and is generally accurate.

Model Prediction

To test the accuracy of our model, we will use the model to predict the dependent variable of our testing data and training data and then create a confusion matrix to depict the accuracy of the model.

```
# Add column of predicted data for test data
bill_test$predicted <- predict(glm_bill_train2, bill_test, type = "response")

# Create new columns for predicted class and make them factors
bill_test <- bill_test %>%
  mutate(predicted = ifelse(predicted > 0.5, '1', '0')) %>%
  mutate(predicted_factor = ifelse(predicted == '1', "Legal Tender",
                                   "Counterfeit")) %>%
  mutate(predicted_factor = factor(predicted_factor,
                                   labels = c("Counterfeit", "Legal Tender")))

# Add column of predicted data for training data
bill_train$predicted <- glm_bill_train2$fitted.values

# Create new columns for predicted class and make them factors
bill_train <- bill_train %>%
  mutate(predicted = ifelse(predicted > 0.5, '1', '0')) %>%
  mutate(predicted_factor = ifelse(predicted == '1', "Legal Tender",
                                   "Counterfeit")) %>%
  mutate(predicted_factor = factor(predicted_factor,
                                   labels = c("Counterfeit", "Legal Tender")))
```

This gives us the predicted probabilities of each data points classification and creates a new column for the predicted classification of each row. Furthermore, we were able to factor these classifications of the predicted variables.

In order to create the confusion matrices, we will also have to revert the `is_genuine` variables back to their original factor classifications of “Legal Tender” and “Counterfeit”.

```
# Revert is_genuine variable to original factor labels for training data
bill_train <- bill_train %>%
  mutate(is_genuine = ifelse(is_genuine == "1", "Legal Tender", "Counterfeit")) %>%
  mutate(is_genuine = factor(is_genuine,
                             labels = c("Counterfeit", "Legal Tender")))

bill_test <- bill_test %>%
  mutate(is_genuine = ifelse(is_genuine == "1", "Legal Tender", "Counterfeit")) %>%
  mutate(is_genuine = factor(is_genuine,
                             labels = c("Counterfeit", "Legal Tender")))
```

Since all labels and factors are in order, we can now create our confusion matrices.

```
# Confusion matrix for our model on the training data
confusionMatrix(bill_train$predicted_factor, bill_train$is_genuine,
                positive = 'Legal Tender')
```

```
## Confusion Matrix and Statistics
##
##              Reference
## Prediction      Counterfeit Legal Tender
##   Counterfeit          351           0
##   Legal Tender           4          667
```

```
##
##           Accuracy : 0.9961
##           95% CI   : (0.99, 0.9989)
##    No Information Rate : 0.6526
##    P-Value [Acc > NIR] : <2e-16
##
##           Kappa : 0.9913
##
##    McNemar's Test P-Value : 0.1336
##
##           Sensitivity : 1.0000
##           Specificity : 0.9887
##           Pos Pred Value : 0.9940
##           Neg Pred Value : 1.0000
##           Prevalence : 0.6526
##           Detection Rate : 0.6526
##           Detection Prevalence : 0.6566
##           Balanced Accuracy : 0.9944
##
##           'Positive' Class : Legal Tender
##
```

```
# Mosaic Table for training data
```

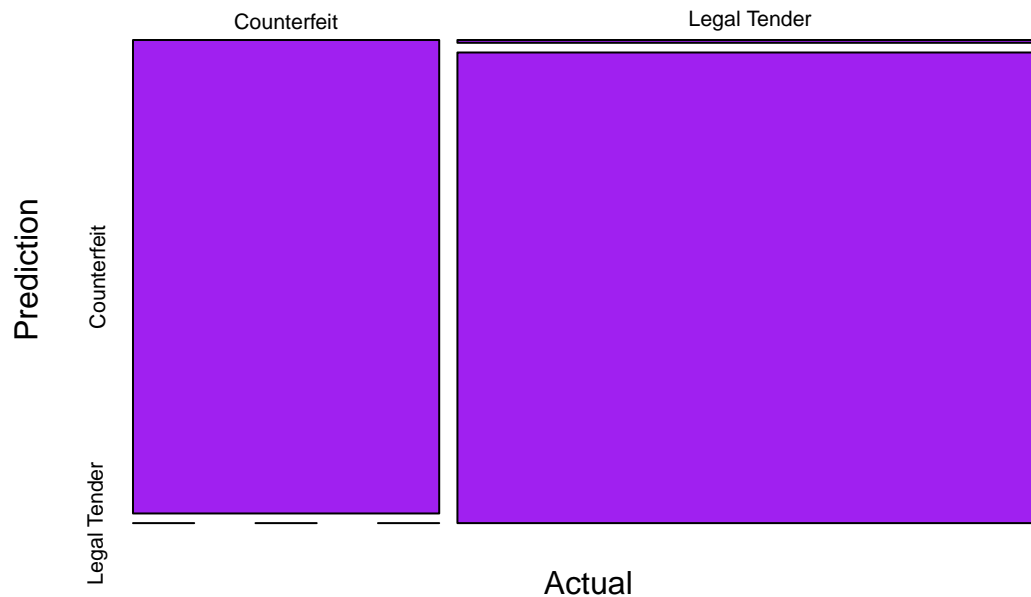
```
bill_mos_tab_train <- table(bill_train$predicted_factor, bill_train$is_genuine)
bill_mos_tab_train
```

```
##
##           Counterfeit Legal Tender
##    Counterfeit           351           0
##    Legal Tender           4           667
```

```
# Mosaic Plot
```

```
mosaicplot(bill_mos_tab_train, main = 'Confusion Matrix', xlab = 'Actual',
           ylab = 'Prediction',
           color = 'purple')
```

Confusion Matrix



The confusion matrix reveals a model accuracy of 99% with our training data which is quite remarkable. Of the nearly 1000 rows of data, only 8 were incorrectly predicted and it was revealed that there is a high Kappa value of 0.98. However, McNemar's Test was unable to be rejected at the 0.05 alpha level of significance, so the reliability of our model may not be considered high.

The next confusion matrix we can make is using our training data model on our test data.

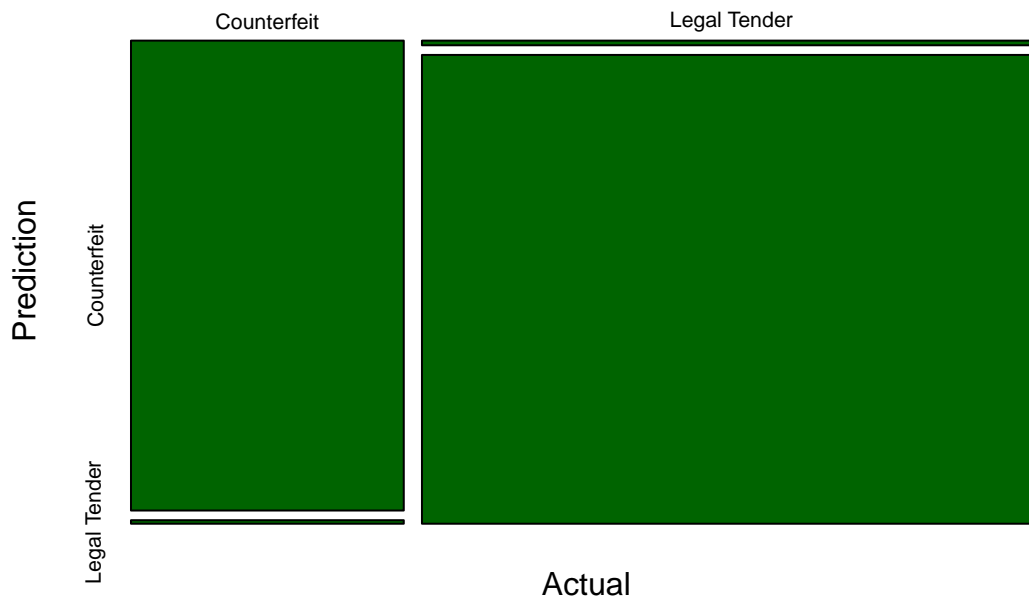
```
# Confusion matrix for our model on the testing data
confusionMatrix(bill_test$predicted_factor, bill_test$is_genuine,
                positive = 'Legal Tender')
```

```
## Confusion Matrix and Statistics
##
##              Reference
## Prediction    Counterfeit Legal Tender
## Counterfeit      133         1
## Legal Tender       3        301
##
##              Accuracy : 0.9909
##              95% CI : (0.9768, 0.9975)
##      No Information Rate : 0.6895
##      P-Value [Acc > NIR] : <2e-16
##
##              Kappa : 0.9786
##
##  McNemar's Test P-Value : 0.6171
##
##              Sensitivity : 0.9967
##              Specificity : 0.9779
##      Pos Pred Value : 0.9901
##      Neg Pred Value : 0.9925
##              Prevalence : 0.6895
##      Detection Rate : 0.6872
##      Detection Prevalence : 0.6941
```

```
##          Balanced Accuracy : 0.9873
##
##          'Positive' Class : Legal Tender
##
# Mosaic Table for testing data
bill_mos_tab_test <- table(bill_test$predicted_factor, bill_test$is_genuine)
bill_mos_tab_test

##
##          Counterfeit Legal Tender
## Counterfeit          133           1
## Legal Tender           3          301
# Mosaic Plot
mosaicplot(bill_mos_tab_test, main = 'Confusion Matrix', xlab = 'Actual',
           ylab = 'Prediction',
           color = 'dark green')
```

Confusion Matrix



The confusion matrix regarding the testing data once again reveal a high level of accuracy of our model at around 99%. There were only 4 misclassifications done by the model in total. Similar to before, there was a high Kappa value of around 0.98 which reveals that our results are statistically significant; however, the McNemar's test P-value was extremely close to one. This means the null hypothesis is unable to be rejected in its entirety which may indicate that our model does not do a good job generalizing.

Discussion

The original objective of this investigation was to determine if there were significant predictors in classifying a bill as legal tender or counterfeit. The initial logistic regression used 6 continuous predictor variables, but not all were found to be considered statistically significant in classifying a bill. So, we created a logistic regression model that found 4 significant predictor variables for classification which included the bill's measurements of its length, margins, and right height.

In order to create a good model, we first used Cook's Distance, DFBETAS, and DFFITS test for outliers,

and determined 3 outliers to remove from our model. Then we split our data into 70% training data and 30% testing data, and used the training data to train our logistic model. Once we had our logistic model, we used the Variance Inflation Factor to make sure our variables were not collinear, and went forward with the investigation.

The odds ratio and average marginal effects revealed that there was a correlation between a unit increase in the `height_right`, `margin_low`, and `margin_up` variables with the likelihood of a bill being classified as counterfeit. The only variable that increased the likelihood of a bill being classified as legal tender with a unit increase in said variable was the `length` variable. This might mean that counterfeit bills often have similar specifications to legal tender, but have difficulty in maintaining the correct mint length.

After running the models and comparing their predictions to the actual training and testing values, we found that it was 99% accurate in both cases. These results were found to be statistically significant according to the Kappa value of the confusion matrices, but were not considered generalizable according to the McNemar's P-value test. Thus, this model may do a good job classifying bills, but may not be applicable to all other data sets.

Ultimately, we were able to create a logistic model that found 4 statistically significant predictors in classifying a bill as legal tender or counterfeit. Future investigations may want to look at other qualities of legal tender besides their measurements to create a more general model.

Link to Data: https://www.kaggle.com/datasets/alexandrepetit881234/fake-bills?select=fake_bills.csv