# Final Project

Zirui Wang (Colin)

3/9/2021

## Introduction

This R Markdown assignment is for Math189 Final Project. In this assignment, I will perform exploratory data analysis (EDA) on the Swiss Bank Notes dataset[1], which contains 200 samples of swiss banknotes with their dimension information. From these 200 samples, 100 were genuine and the other 100 were counterfeit. This dataset is available from the course website[2]. Based on EDA, I will also split our data into training and validation set using K-Fold validation method. Then, I will perform a classification test with both LDA and Logistic Regression and see their performance with and without factor models that are used to reduce the dimension of my feature matrix. In the end, I will analyze the classification results and make conclusions on the project.

### Data Description [3]

`Length`: Length of bill (mm)
`Left`: Width of left edge (mm)
`Right`: Width of right edge (mm)
`Bottom`: Bottom margin width (mm)
`Top`: Top margin width (mm)
`Diagonal`: Length of diagonal (mm)
`Label`(derived): 0 is counterfeit; 1 is genuine

## Body

### Load Data

```
set.seed(42)
df <- read.table("SBN.txt")
df$Label <- append(rep(0, 100), rep(1, 100))
formattable(df[sample(nrow(df), 3), ], format='pandoc')
```

| | Length | Left | Right | Bottom | Top | Diagonal | Label |
|---|---|---|---|---|---|---|---|
| BN49 | 214.6 | 129.7 | 129.8 | 7.9 | 10.3 | 141.1 | 0 |
| BN65 | 215.0 | 130.0 | 129.8 | 8.6 | 10.6 | 141.5 | 0 |
| BN153 | 214.6 | 129.7 | 129.3 | 10.4 | 11.0 | 139.3 | 1 |

[1] Flury, B. and Riedwyl, H. (1988). Multivariate Statistics: A practical approach. London: Chapman & Hall, Tables 1.1 and 1.2, pp. 5-8 [Accessed: 9-Mar-2021]

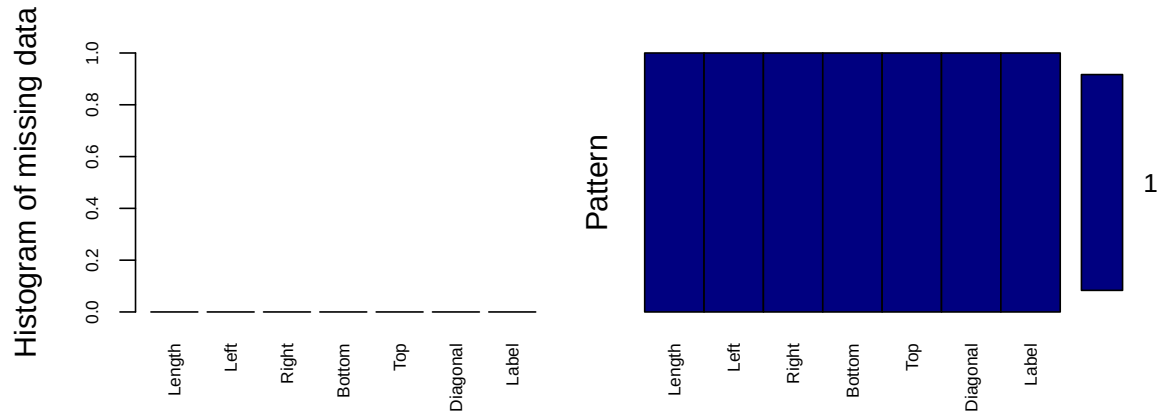
[2] T. McElroy, "ma189" [Online]. Available: https://github.com/tuckermcelroy/ma189/tree/main [Accessed: 9-Mar-2021]

[3] Chris Fraley, banknote: Swiss banknotes data in mclust: Gaussian Mixture Modelling for Model-Based Clustering, Classification, and Density Estimation, 20-Nov-2020. [Online]. Available: https://rdrr.io/cran/mclust/man/banknote.html. [Accessed: 09-Mar-2021].

## Exploratory Data Analysis & Visualization

First, let's check if our dataset contains any missing values by making an aggregated plot for patterns of missing values.

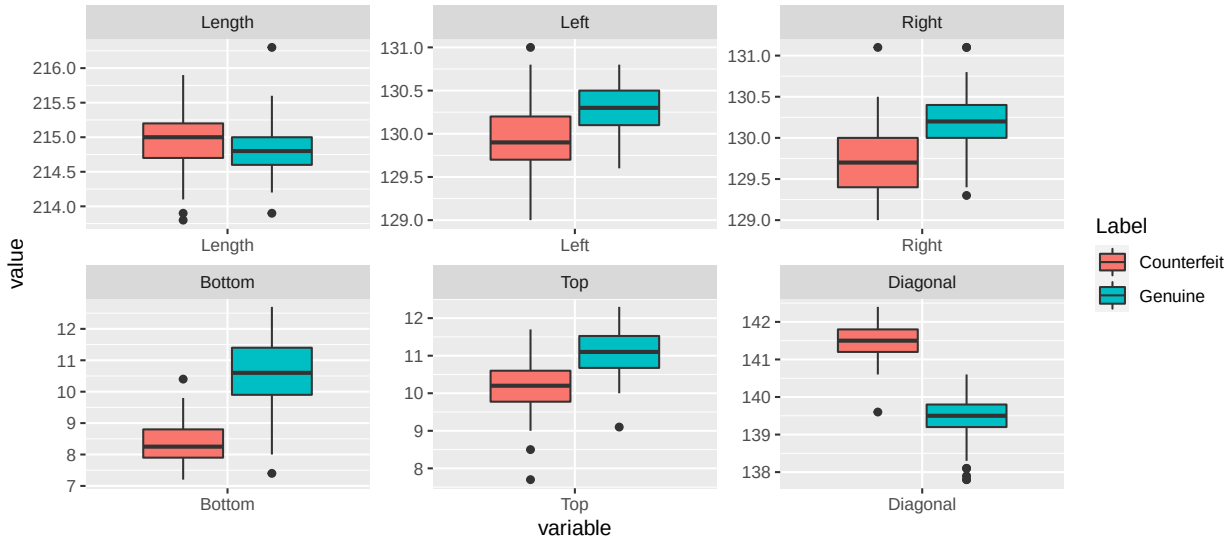
```
aggr_plot <- aggr(df, col=c('navyblue','red'), numbers=TRUE, sortVars=TRUE,
                  labels=names(data), cex.axis=.7, gap=3,
ylab=c("Histogram of missing data","Pattern"))
```




From the result, we saw that our dataset contains no missing values and therefore we don't need to perform any dropping or imputation procedures before we further visualize our data and fit the data into different models.

Now, Let's make some boxplots to see the distribution of variables grouped by real and fake banknotes.

```
df_temp <- df
df_temp$Label <- mapvalues(df_temp$Label, from=c(0, 1),
                           to=c("Counterfeit", "Genuine"))
dfm <- melt(df_temp, id.var = "Label")
p <- ggplot(data = dfm, aes(x=variable, y=value))+geom_boxplot(aes(fill=Label))
p + facet_wrap( ~ variable, scales="free")
```



From the above pairwise boxplots, we see that the dimension between counterfeit and genuine money has significant difference. For most variables (i.e. `Left`, `Right`, `Bottom`, `Top`, `Diagonal`), their 25% quantile and 75% quantile don't overlap, which can be used as good features to fit into our classification models. The only variable that shows some overlapping in distribution is `Length`, which we can further examine by checking the fitting parameters of our classification model later.

Then, let's visualize the relationship between different variables grouped by different Labels with scatterplot, kernel density estimation, as well as correlations (here, correlation plot performs the same as the level plot covered in the lecture). In particular, we are trying to see if there are any variables that are strongly correlated, which can be some potential good candidates to perform dimension reduction upon.

```
ggpairs(df_temp, aes(color = Label), columns = 1:6,
upper = list(continuous = wrap("cor", size=3)))
```


From the above plots, we can also gain insights in correlations between different variables. Seeing them as a whole, we can see that the most significant correlation appears to be between variable `Left` and `Right`, with a correlation coefficient of 0.743. There are also variables with moderate correlations, such as `Top`, `Bottom`, `Right`, `Left` vs `Diagonal` (with correlation magnitude >= 0.5), as well as among `Right`, `Left`, and `Bottom`

(with correlation magnitude >= 0.35). The length `variable` generally has insignificant correlations with other variables. Since most variables have moderate to strong correlations with other variables, we believe that this dataset is suitale for dimension reduction later. However, this may not be helpful in prediction since we can also observe differences in correlations grouped by `Labels`. For instance, real banknotes have very insignificant correlations between `Top` and `Right` whereas fake banknotes have more correlations (0.001 vs 0.133). On the other hand, fake banknotes have very insignificant correlations between `Borrom` and `Diagonal` whereas real banknotes have moderate correlations (-0.001 vs 0.378).

If we look deeper into the above plots, we can also discover that the `Diagonal` variable as well as all scatterplots associated with `Diagonal` exist most significant differences between genuine banknotes and counterfeit banknotes. We can almost always draw a line in the scatterplot that differentiate them into separate classes with high accuracy, and we also see from the kernel density estimation that their distribution in `Diagonal` value almost has no overlap. We also see that the shape of density distribution for `Bottom` variable is different between conterfeit and genuine banknotes, with counterfeit banknotes being much spiky and genuine banknotes being relatively flat.

## K-Fold Validation

Now, after we explored the distribution of our data as well as correlations between variables, let's prepare our data into training and validation set using K-Fold validation procedure. Since we have 200 observations, it might be a good choice to perform 4-Fold cross validation, where in each process there are 150 observations for training and 50 observations for validation

```
fitControl <- trainControl(method = "cv", number = 4)
```

## Classification with LDA & Logistic Regression

We try to make some predictions by fitting our dataset into a logistic regression model and an LDA model and compare the performance fold-wise.

```
fit_model <- function(df, mdl, fc){
set.seed(42)
df$Label <- as.factor(df$Label)
model <- train(Label ~ ., data = df, trControl = fc, method = mdl)
return (model)
}
```

```
logit_6 <- fit_model(df, 'glm', fitControl)
lda_6 <- fit_model(df, 'lda', fitControl)
fold <-logit_6$resample$Resample
performance_6 <- data.frame(fold, logit_6$resample$Accuracy, lda_6$resample$Accuracy)
colnames(performance_6) <- c("Fold", "Logit Accuracy", "LDA Accuracy")
formattable(performance_6, format='pandoc')
```

| Fold | Logit Accuracy | LDA Accuracy |
|---|---|---|
| Fold1 | 0.98 | 0.98 |
| Fold2 | 1.00 | 1.00 |
| Fold3 | 0.98 | 1.00 |
| Fold4 | 0.98 | 1.00 |

From the above results, we see that a logistic regression performs very well on the dataset. Our model's accuracy is above 0.98 for all validation sets over 4 folds. There are only 3 misclassifications. At the same time, we see that an LDA model also performs very well on this dataset. Our model's accuracy is above 0.98 for all validation sets over 4 folds. There is only 1 misclassification in the first fold.

## Factor Model

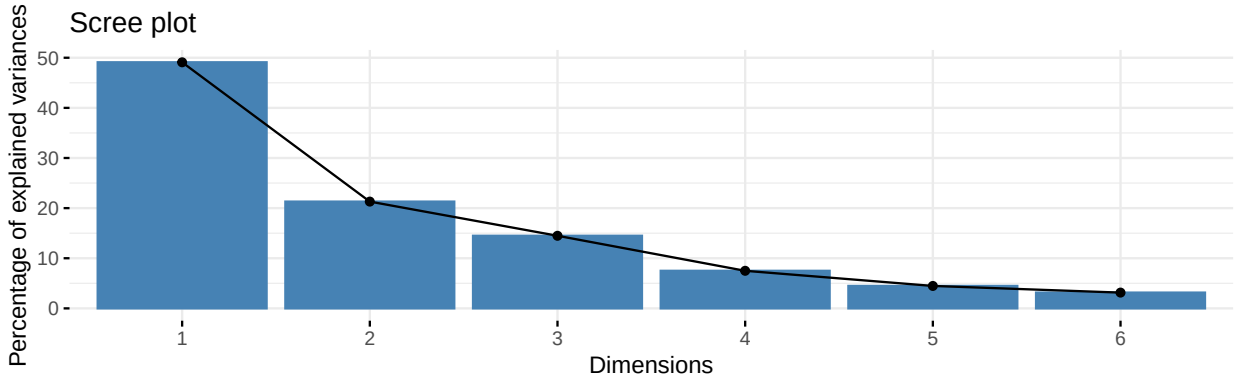
Now that we know both LDA and Logistic Regression models perform very well on our dataset, we want to see if our classifiers can still perform well after we apply dimensionality reduction to our dataset. From the EDA section, we observed that there are many variables that are moderately to strongly correlated with each other. Here we'll use a PCA model to realize that.

```
X <- scale(df[c('Length', 'Left', 'Right', 'Bottom', 'Top', 'Diagonal')])
pca_result <- prcomp(X)
pca_var <- pca_result$sdev^2
pve <- pca_var/sum(pca_var)
out2 <- cbind(pca_var,pve,cumsum(pve))
colnames(out2) <- c("Eigenvalue","Proportion","Cumulative")
rownames(out2) <- c("PC1","PC2","PC3","PC4","PC5","PC6")
formattable(out2, format='f')
```

```
##     Eigenvalue Proportion Cumulative
## PC1 2.9456     0.4909     0.4909
## PC2 1.2781     0.2130     0.7039
## PC3 0.8690     0.1448     0.8488
## PC4 0.4498     0.0750     0.9237
## PC5 0.2687     0.0448     0.9685
## PC6 0.1889     0.0315     1.0000
```

From the above statistics, we see that the PCA model performed well on our dataset, with top 2 principal components explaining more than 70% of variance and top 4 principal components explaining more than 90% of variance. Now, let's use a scree plot to see if there are any apparent elbows that can be used as an indicator as components used.

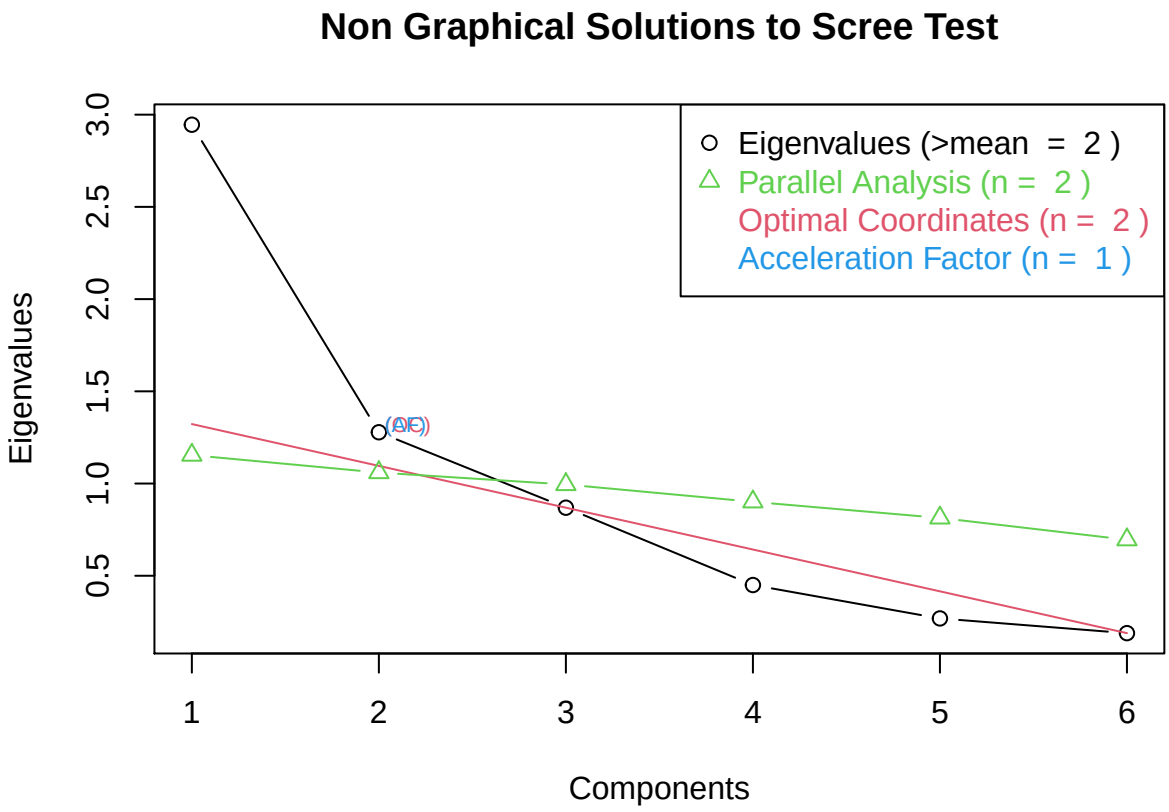
```
fviz_eig(pca_result)
```




From the above plot, we cannot specify a clear elbow. We can use some additional tools to analyze an optimal number of factors. Here, we will be techniques covered in paper Non graphical solutions for the Cattell's scree test[4].

```
ev <- eigen(cor(X)) # get eigenvalues
ap <- parallel(subject=nrow(X),var=ncol(X), rep=100,cent=.05)
plotnScree(nScree(x=ev$values, aparallel=ap$eigen$qevpea))
```

[4]Raiche, G., Riopel, M. and Blais, J.-G. (2006). Non graphical solutions for the Cattell's scree test. Paper presented at the International Annual meeting of the Psychometric Society, Montreal. [http://www.er.uqam.ca/nobel/r17165/RECHERCHE/COMMUNICATIONS/]




So, based on the result of such analysis, we choose to reduce the dimension of our dataset to two based on our first two principal components. Now, let's try to interpret these principal components.

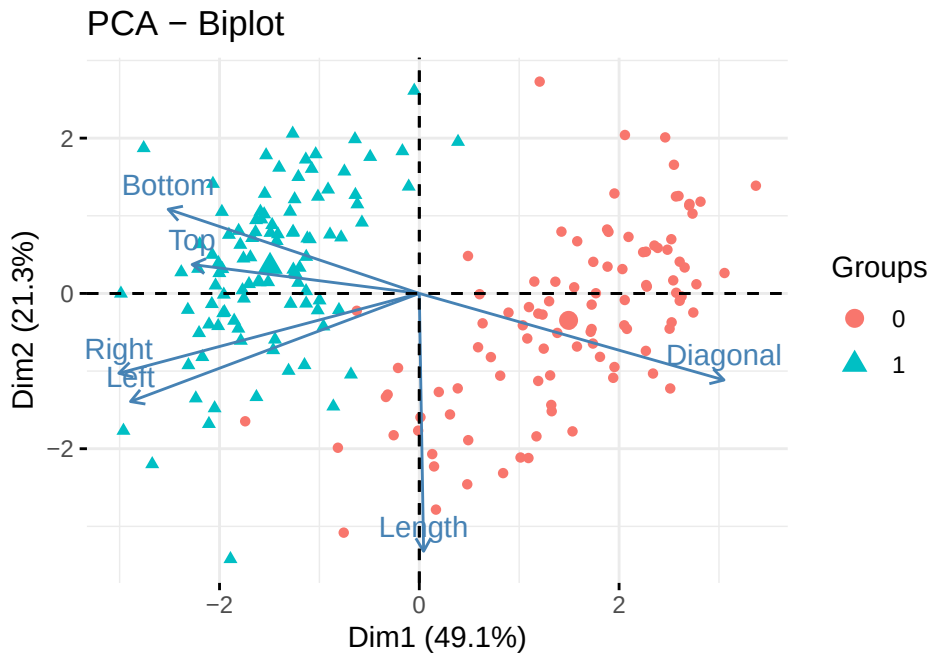
```
formattable(t(pca_result$rotation), format='f')
```

```
##     Length  Left    Right   Bottom  Top     Diagonal
## PC1 0.0070  -0.4678 -0.4867 -0.4068 -0.3679 0.4935
## PC2 -0.8155 -0.3420 -0.2525 0.2662  0.0915  -0.2739
## PC3 0.0177  -0.1034 -0.1235 -0.5835 0.7876  -0.1139
## PC4 0.5746  -0.3949 -0.4303 0.4037  0.1102  -0.3919
## PC5 -0.0588 0.6395  -0.6141 -0.2155 -0.2198 -0.3402
## PC6 0.0311  -0.2977 0.3492  -0.4624 -0.4190 -0.6318
```

From the above table, we can see that our first loading vector places approximately equal weights to `Left`, `Right`, and `Diagonal`. Hence, this component roughly corresponds to a measure of the banknote's horizontal edges' width with respect to its diagonal length of the image; we can see that our second loading vector places most weights to `Length`. Hence, this components roughly corresponds to a measure of the banknote's length in general.

Further, we can also visualize the effectiveness of dimensionality reduction by drawing a biplot for our first two principal components.

```
fviz_pca_biplot(pca_result,label ="var", habillage=df$Label)
```




Based on the above biplot, we can see that our first two principal components can clearly separate our variables into roughly four directions, with `Bottom` and `Top` at one direction, `Right` and `Left` at one direction, `Length` at one direction, and `Diagonal` at one direction. We can also see that our dataset is relatively well separated into two clusters, with genuine (labeled as 1) on the left side and counterfeit (labeled as 0) on the right side.

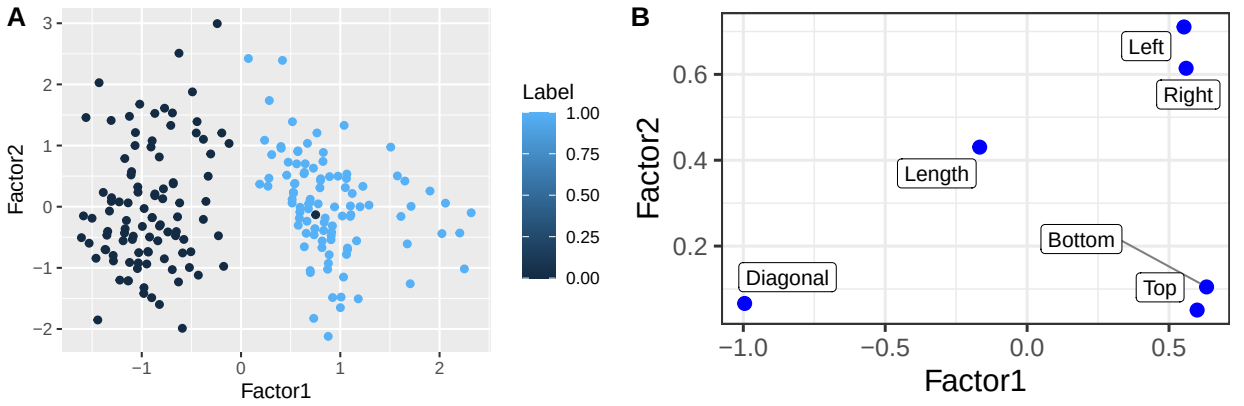
Now after we have performed PCA on our dataset and decide to use 2 principal components, we can also apply a factor model with maximum likelyhood estimation (MLE) with 2 factors to examine its factor loadings and factor scores.

```
MLE <- factanal(factors = 2, x = X, score = "regression", rotation="varimax")
MLE$loadings
```

```
##
## Loadings:
##          Factor1 Factor2
## Length   -0.167   0.431
## Left      0.553   0.711
## Right     0.560   0.614
## Bottom    0.632   0.105
## Top       0.599
## Diagonal -0.995
##
##                Factor1 Factor2
## SS loadings      2.397   1.085
## Proportion Var   0.399   0.181
## Cumulative Var   0.399   0.580
```

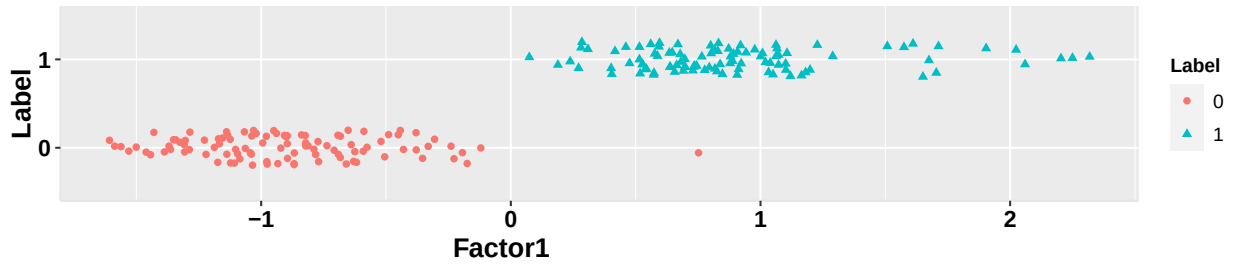
Based on the above factor loadings, we can see that our factor 1 puts most emphasis on the `Diagonal` variable, with a magnitude of 0.995. This indicates the importance of that variable, which is consistent with our analysis in the EDA part. Factor 2 puts most emphasis on the `Left` and `Right` variable. The aggregation of both factors also roughly match the loadings statistic from the above PCA analysis. Now, let's examine the factor scores by visualizing the coordinates of each observation colored by each label (plot A) as well as factor loadings based on our result of MLE model (plot B).

```
load <- as.data.frame(MLE$loadings[,1:2])
scores <- as.data.frame(MLE$scores)
Label <- df$Label
fig1 <- ggplot(scores, aes(x=Factor1, y=Factor2, color=Label)) + geom_point()
fig2 <- ggplot(load, aes(x=Factor1, y=Factor2))+ geom_point(color = "blue",
        size = 3) + geom_label_repel(aes(label = rownames(load)),
        box.padding = 0.35, point.padding = 0.5,segment.color = 'grey50') +
        theme_bw(base_size = 16)
cowplot::plot_grid(fig1, fig2, labels = "AUTO")
```




As we can see from the above factor score visualization, banknote observations with label 0 (counterfeit) and banknote observation with label 1 (genuine) clearly split into two distinct clusters, with only one counterfeit banknote dislocated into the genuine cluster. More concretely, we can observe that except for the dislocated observation, all genuine banknotes have a positive value for factor 1 whereas all counterfeit banknotes have a negative value for factor 1. That being said, even a weak learner that makes prediction solely on the value on factor 1 can still perform very well. On the loadings visualization, the pattern is consistent with the figure in the biplot for PCA illustrated above, where `Bottom` and `Top`, `Left` and `Right`, `Diagonal`, and `Length` are grouped into 4 parts. Therefore, even 1 factor would be sufficient to train a classification model to make correct predictions on most of the labels. Let's visualize our results based on only 1 factor using a strip chart.

```
scores$Label <- df$Label
ggplot2.stripchart(data=scores, xName='Label', yName='Factor1',
        groupName='Label') + coord_flip()
```




Comparing the results between PCA and MLE, we believe that our factor analysis based on MLE performed better. Therefore, our final dimension reduction of variables for our dataset will be based on MLE. Specifically, our new feature matrix will only have a dimension of 1 (i.e. `Factor1`) instead of 6 (i.e. `Length`, `Left`, `Right`, `Bottom`, `Top`, `Diagonal`)

## Discussion of Assumptions, Analysis, and Final Results

### Assumptions for PCA

Although we **did not** use PCA in the end, we can discuss its assumptions and justify **why it doesn't perform as robust as the MLE model**. Since there are no mentioning for assumptions necessary for PCA in PCA lectures (i.e. lecture 19, 20), we found its assumptions from an external website [5], and we'll discuss these assumptions here:

1. Variables are continuous. **This assumption holds** because our banknotes dataset has all variables that are numerically continuous (i.e. dimensions, lengths, widths, etc).
2. There needs to be a linear relationship between all variables. **This assumption doesn't hold** well since the scatterplot illustrated in EDA section showed no significant linear relationship among most variables.
3. We should have sampling adaquacy, which means that we should have a relatively large sample size. Since we have 200 observations for 6 variables, we believe we have sampling adaquacy, but **this assumption holds** only at the borderline level.
4. Our data should be suitable for dimension reduction. We believe that **this assumption holds** because we have seen some high correlations between different variables in the level plot above. Also, regarding our objective, which is to differentiate between genuine and counterfeit banknotes, we have seen significant difference in their variable distribution. Therefore, we believe that our dataset is very suitable for dimension reduction in terms of our objective.
5. There should be no significant outliers. **This assumption doesn't hold** as outliers can be easily identified on the scatterplot above.

To sum up, we believe that the main reasons that caused PCA to not work so well compared to MLA are due to lack of linear relationship between variables as well as existence of outliers for multiple varibles in our dataset.

### Assumptions for MLE

Since we'll be using results from a maximum likelihood estimation model, we also need to justify whether the assumption for such a model has been satisfied. According to lecture slides, we must assume that the data are independently sampled from a **multivariate normal distribution** with mean vector $\underline{\mu}$ and variance-covariance matrix $\mathbf{\Sigma}$.

Here, we'll be using multiple tests for multivariate normality to assess our dataset.

```
# Let's define a wrapper function to reduce repetition of code chucks
test_mvn <- function(mat){
result <- rbind.fill(
  mvn(mat, mvnTest = "energy", desc = TRUE)$multivariateNormality,
  mvn(mat, mvnTest = "hz", desc = TRUE)$multivariateNormality,
  mvn(mat, mvnTest = "mardia", desc = TRUE)$multivariateNormality,
  mvn(mat, mvnTest = "dh", desc = TRUE)$multivariateNormality,
  mvn(mat, mvnTest = "royston", desc = TRUE)$multivariateNormality)
result$Conclusion <- ifelse(!is.na(result$MVN), result$MVN, result$Result)
return (result)
}
```

```
counterfeit <- test_mvn(X[1:100,])
genuine <- test_mvn(X[101:200,])
mixed <- test_mvn(X)
mvn_results <- data.frame(counterfeit$Test, counterfeit['p value'],
```

[5] "Principal Components Analysis (PCA) using SPSS Statistics," How to perform a principal components analysis (PCA) in SPSS Statistics | Laerd Statistics. [Online]. Available: https://statistics.laerd.com/spss-tutorials/principal-components-analysis-pca-using-spss-statistics.php. [Accessed: 06-Mar-2021].

```
                genuine['p value'], mixed['p value'],
                counterfeit$Conclusion, genuine$Conclusion, mixed$Conclusion)
colnames(mvn_results) <- c("Test", "C-P-value",  "G-P-Value",
                           "M-P-Value", "C-MVN",
                           "G-MVN", "M-MVN")
formattable(mvn_results ,format='pandoc')
```

| Test | C-P-value | G-P-Value | M-P-Value | C-MVN | G-MVN | M-MVN |
|---|---|---|---|---|---|---|
| E-statistic | 6.500000e-02 | 0.000000e+00 | 0.000000e+00 | YES | NO | NO |
| Henze-Zirkler | 6.915838e-01 | 2.234157e-11 | 0.000000e+00 | YES | NO | NO |
| Mardia Skewness | 1.000000e+00 | 1.000000e+00 | 1.000000e+00 | NO | NO | NO |
| Mardia Kurtosis | 2.000000e+00 | 2.000000e+00 | 2.000000e+00 | NO | NO | NO |
| MVN | NA | NA | NA | NO | NO | NO |
| Doornik-Hansen | 4.258836e-04 | 2.116117e-02 | 1.605023e-08 | NO | NO | NO |
| Royston | 5.500670e-06 | 1.225216e-08 | 5.820549e-13 | NO | NO | NO |

From the above result for counterfeit banknotes (i.e. denoted with prefix `C`), we see that out of 5 tests (i.e., `energy` test, `Henze-Zirkler` test, `Mardia` test, `Doornik-Hansen` test and `Royston` test) and a significant value of 0.05, only `energy` test and `Henze-Zirkler` test concluded that this subset of our data followed a multivariate normal distribution. However, since the other 3 tests made the opposite conclusion, we cannot conclude that this subset follows a multivariate normal distribution. From the above result for genuine banknotes (i.e. denoted with prefix `G`), we see that all 5 tests concluded that this subset of data don't follow a multivariate normal distribution. This consistency in conclusion is also supported by our visualization in EDA part, where the `Bottom` variable for genuine subset clearly doesn't follow a normal distribution. The mixed dataset (i.e. denoted with predix `M`) follows the same conclusion with the subset with only genuine banknotes.

*In a theoretical perspective*, such a factor analysis might be a waste of time because all our subsets as well as our mixed dataset violate the assumption necessary for an MLE model. However, *from an empirical perspective*, as can be seen from the strip chart in a previous section, our MLE model performs quite well on our dataset in terms of reducing the dimension while differentiating between genuine and counterfeit banknotes. To further our discussion and make a conclusion on the effectiveness of dimension reduction in terms of classification problems, we need to see if the same models can perform equally or better after we reduce the dimensions from 6 to 1. Before we start, let's display some rows of our new data frame.

```
set.seed(42)
scores <- scores[c('Factor1', 'Label')]
formattable(scores[sample(nrow(scores), 3), ], format='pandoc')
```

| | Factor1 | Label |
|---|---|---|
| BN49 | -0.5885789 | 0 |
| BN65 | -0.8680085 | 0 |
| BN153 | 0.8783762 | 1 |

Before we fit both models, we also need to reevaluate the assumptions necessarily needed for both Logistic Regression and LDA. Let's start with LDA.

### Assumptions for LDA

Since LDA was developed from ANOVA and MANOVA, the assumptions for LDA is very similar to MANOVA. In specific:

1. The data from group $k$ has common mean vector $\underline{\mu}^{(k)}$.
2. Homoskedasticity: The data from all groups have common covariance matrix $\mathbf{\Sigma}$.
3. Independence: The observations are independently sampled.
4. Normality: The data are multivariate normally distributed.

While the data we will be using to fit will not be the same with the dataset we have originally, we can still discuss the assumptions against our original dataset since the factor scores of MLE are essentially some linear combinations of variables we had in our original dataset. One can also view this as a latent representation of our dataset processed by a linear regressor.

We can observe from the density plots in EDA section that all of our variables are numerically continuous and all variable distributions from both genuine and counterfeit banknotes do have a peek somewhere, indicating a possible expected mean. Therefore, **the first assumption holds**. We have also justified that our dataset doesn't follow a multivariate normal distribution in discussing the assumptions for MLE, so **the fourth assumption doesn't hold**. We can also sufficiently justify that **the third assumption holds** since each banknots is individually collected. Finally, we determined that **the second assumption doesn't hold**, since homoskedasticity can be visually seen as a cone shaped scatterplot distribution between different variables across each varible. That is, the larger the value, the sparser the distribution. However, this trend cannot be clearly identified from the scatterplots in our EDA section.

### Assumptions for Logistic Regression

Since logistic regression is a likelyhood-based method for classification, there aren't as many constraints as we discussed above for the LDA model. From lecture 15, we know that every likelyhood-based model needs to have $\mathbf{P}[y_i = 1|x_i] = p(x_i)$ and $\mathbf{P}[y_i = 0|x_i] = 1 - p(x_i)$ in order to perform well. **This assumption perfectly holds** since we can see from both the strip plot in the factor analysis results above and the scatterplot grouped by different labels in our EDA analysis that there are variables that can clearly used as indicators (i.e. `Diagonal`, `Factor1`) to differentiate genuine banknotes from counterfeit banknotes, and for the `Factor1` variable that we'll be using, a linear separability is also guaranteed at 0, where all except one ourlier will be correctly classified based on this threshold.

### Final Results across Each Fold

Now, let's start to fit the same models again.

```
logit_1 <- fit_model(scores, 'glm', fitControl)
lda_1 <- fit_model(scores, 'lda', fitControl)
fold <-logit_1$resample$Resample
performance_1 <- data.frame(fold, logit_1$resample$Accuracy, lda_1$resample$Accuracy)
colnames(performance_1) <- c("Fold", "Logit Accuracy", "LDA Accuracy")
formattable(performance_1, format='pandoc')
```

| Fold | Logit Accuracy | LDA Accuracy |
|---|---|---|
| Fold1 | 0.98 | 0.98 |
| Fold2 | 1.00 | 1.00 |
| Fold3 | 1.00 | 1.00 |
| Fold4 | 1.00 | 1.00 |

The above result shows that both Logistic Regression and LDA model still perform very well even if we reduced the dimension of our dataset from 6 to 1 using factor analysis. Both models only have 1 misclassifications across all folds, which we believe is the outlier illustrated in the factor analysis section.

## Aggregation of Final Results, Analysis, and Summary

Now, let's combine the results from models fit with 6 dimensions and models fit with only 1 dimension after MLE.

```
combined_results <- data.frame(fold, performance_6[c('Logit Accuracy')],
                               performance_1[c('Logit Accuracy')],
                               performance_6[c('LDA Accuracy')],
                               performance_1[c('LDA Accuracy')])
colnames(combined_results) <- c("Fold", "Logit Acc", "Logit Acc w/ MLE",
                            "LDA Acc", "LDA Acc w/ MLE")
formattable(combined_results, format='pandoc')
```

| Fold | Logit Acc | Logit Acc w/ MLE | LDA Acc | LDA Acc w/ MLE |
|---|---|---|---|---|
| Fold1 | 0.98 | 0.98 | 0.98 | 0.98 |
| Fold2 | 1.00 | 1.00 | 1.00 | 1.00 |
| Fold3 | 0.98 | 1.00 | 1.00 | 1.00 |
| Fold4 | 0.98 | 1.00 | 1.00 | 1.00 |

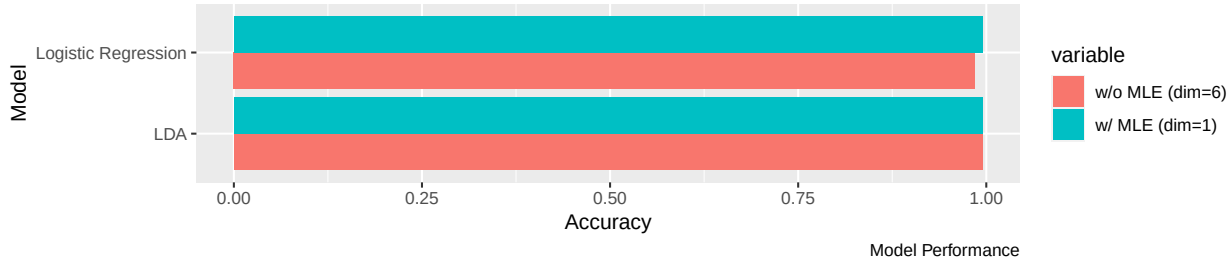
From the combined results, we see that while there's no change in LDA model's accuracy with and without performing data pre-processing with MLE for factor analysis, however, there are two fewer misclassifications for the Logistic Regression model after performing dimension reduction.

Let's aggregate across folds and see the final performance for two models with and without factor analysis.

```
Acc_6 <- c(logit_6$results$Accuracy,lda_6$results$Accuracy)
Acc_1 <- c(logit_1$results$Accuracy,lda_1$results$Accuracy)
Acc <- data.frame(c('Logistic Regression', 'LDA'), Acc_6, Acc_1)
colnames(Acc) <- c('Model', 'w/o MLE (dim=6)', 'w/ MLE (dim=1)')
formattable(Acc, format='pandoc')
```

| Model | w/o MLE (dim=6) | w/ MLE (dim=1) |
|---|---|---|
| Logistic Regression | 0.985 | 0.995 |
| LDA | 0.995 | 0.995 |

```
## Using Model as id variables
```




Based on our results in terms of accuracy, we do conclude that our dimensionality reduction is effective, which not only takes less computational resource, but also reduced the noise in our dataset, which, in turn, resulted in higher prediction accuracy. Here, we believe that the only misclassification comes from the outlier we identified, which we can further study in the future. For instance, we can try to remove that outlier from out dataset and compare the models' performance with and without a factor analysis model to see if there are any further improvements in terms of accuracy.

# Conclusion

In this work, we analyzed the Swiss Bank Notes dataset, which contains 200 observations of genuine and counterfeit banknotes (100 observations for each). We performed exploratory data analysis on variables as well as examine their correlation and distribution with respect to different labels. Then, we used K-Fold validation to split our data into training and validation set with 4 folds, and used both logistic and LDA model to achieve some baseline results. Then, we performed some factor analysis on our dataset using both PCA and MLE models, and we determined that our dataset is very suitable for dimension reduction. We justified the assumptions needed for the MLE model both theoretically and empirically, and subsequently reduced the dimension of our data from 6 to 1 with it. Our pre-processed data only contain one observable outlier lying in the strip chart based on factor 1 grouped by labels. Then, we used the pre-processed the data to fit both classification models again, and compare their performance with respect to their baseline results. Finally, we concluded that our factor analysis achieved positive effects in helping classifiers make correct predictions, which make both of them achieve an overall accuracy of 0.995 for our entire dataset.

# Appendix

## External Library Usage

There're various external libraries used in this project for formatting, visualization, analysis, and modeling. Here is the complete list of libraries that are used in this project: `formattable`, `plyr`, `ggplot2`, `eastGgplot2`, `ggrepel`, `GGally`, `cowplot`, `MVN`, `nFactors`, `factoextra`. `reshape2`, `VIM`, `caret`.