BANK CUSTOMER SEGMENTATION

DENNIS MULUMBI KYALO

2020-11-21

1. INTRODUCTION

Customer segmentation is the process of dividing and grouping customers into different categories based on similar traits such as gender, age, interests, and location. This process, in turn, helps businesses make effective decisions in their marketing and advertising strategies. Companies can therefore target each category efficiently and effectively. This project aims to examine the various customers in the German bank's dataset using both supervised and unsupervised learning techniques. The dataset is comprised of different individuals who take credit from a bank.

2. METHOD

Using the supervised learning technique, we shall apply the logistic, K-Nearest-Neighbor (KNN), and random forest classification algorithms. In contrast, in unsupervised methods, we shall use the K-Means and Hierarchical clustering techniques. For precision, we shall determine which methodology accurately classifies and predicts gender. Lastly, we shall apply the best-unsupervised machine learning method on the entire dataset and then see the various customer clusters.

2.1 Getting Started.

We first load the dataset and skim through it to have a better understanding.

Table 1: Credit Dataset

X	Age	Sex	Job	Housing	Saving.accounts	Checking.account	Credit.amount	Duration	Purpose
0	67	male	2	own	NA	little	1169	6	radio/TV
1	22	female	2	own	little	moderate	5951	48	radio/TV
2	49	male	1	own	little	NA	2096	12	education
3	45	male	2	free	little	little	7882	42	furniture/equipment
4	53	male	2	free	little	little	4870	24	car
5	35	male	1	free	NA	NA	9055	36	education

The dataset consists of ten columns; therefore, we omit the first column since its an index set.

```r
## We can go ahead and do some data cleaning on our data.
# First we remove the first column which is simply an index set.

credit_data <- credit_data %>% select(-X) # remove the x column
```

Table 2: Data summary

Name	credit_data
Number of rows	1000
Number of columns	9
Column type frequency:	
character	5
numeric	4
Group variables	None

Variable type: character

skim_variable	n_missing	complete_rate	min	max	empty	n_unique	whitespace
Sex	0	1.00	4	6	0	2	0
Housing	0	1.00	3	4	0	3	0
Saving.accounts	183	0.82	4	10	0	4	0
Checking.account	394	0.61	4	8	0	3	0
Purpose	0	1.00	3	19	0	8	0

Variable type: numeric

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100
Age	0	1	35.55	11.38	19	27.0	33.0	42.00	75
Job	0	1	1.90	0.65	0	2.0	2.0	2.00	3
Credit.amount	0	1	3271.26	2822.74	250	1365.5	2319.5	3972.25	18424
Duration	0	1	20.90	12.06	4	12.0	18.0	24.00	72

The dataset now comprises 1000 rows and nine columns. In which five variables are characters, and four are numeric.

- The five character variables include:

  1. Sex (male or female).
  2. Housing (own, rent, or free) .
  3. Saving accounts (little, moderate, quite rich, rich).
  4. Checking account (little, moderate, rich).
  5. Purpose (car, furniture/equipment, radio/TV, domestic appliances, repairs, education, business, vacation/others).

- The four numeric variable includes:

  1. Age.
  2. Job (0 - unskilled and non-resident, 1 - unskilled and resident, 2 - skilled, 3 - highly skilled).
  3. Credit Amount (in DM – Deutsche Mark currency).
  4. Duration (in months).

There are 183 and 394 missing values from the saving accounts and checking accounts, respectively, which could probably mean that perhaps the customers did not have those accounts.

```r
## We can now proceed to check the number of unique features in each column

library(dplyr, warn.conflicts = FALSE)
options(dplyr.summarise.inform = FALSE)

credit_data %>% gather(key = features, value = Value) %>%
group_by(features) %>% summarise(Unique_features_tally = n_distinct(Value)) %>%
  arrange(Unique_features_tally) %>% knitr::kable()
```

features	Unique_features_tally
Sex	2
Housing	3
Checking.account	4
Job	4
Saving.accounts	5
Purpose	8
Duration	33
Age	53
Credit.amount	921

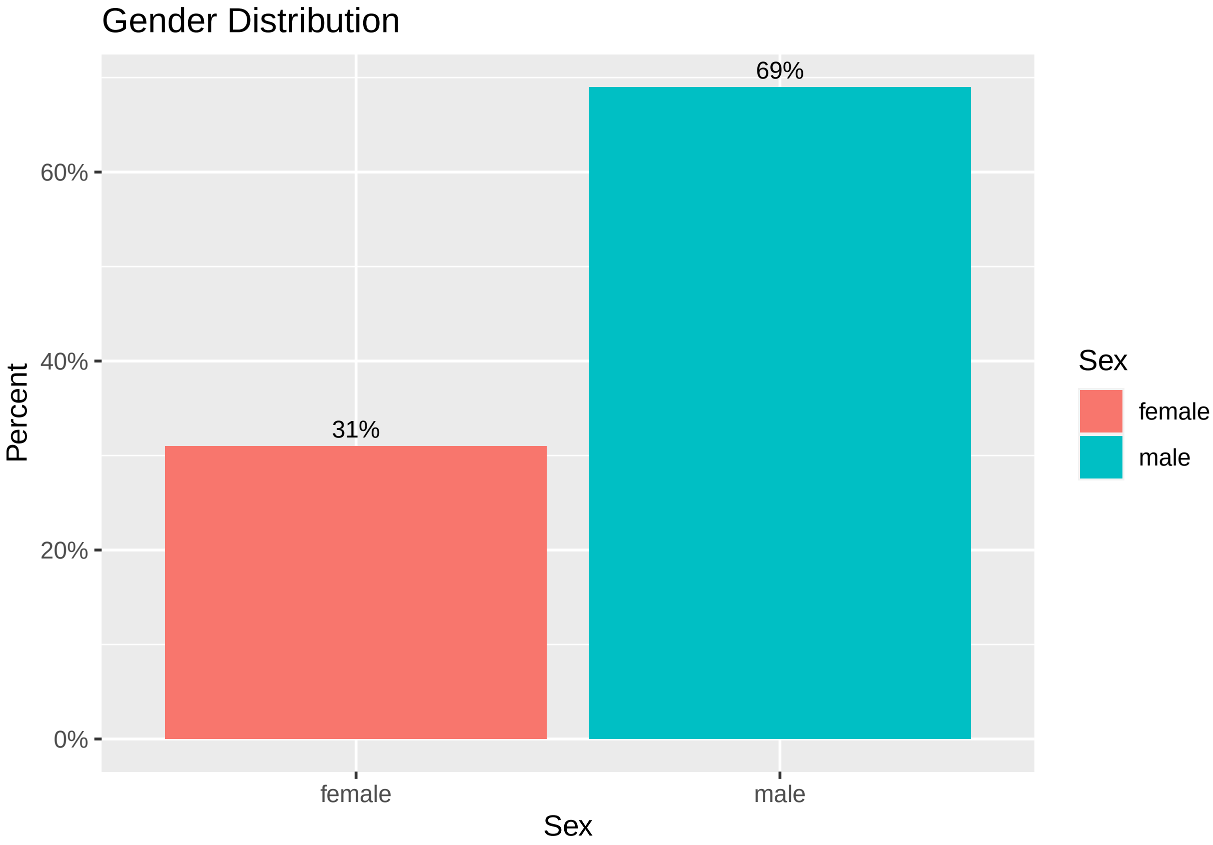
The above table shows the different types of unique features in the different variables.

2.2 Exploratory Data Analysis.

We perform exploratory data analysis on the dataset.

2.2.1 Gender Distribution.

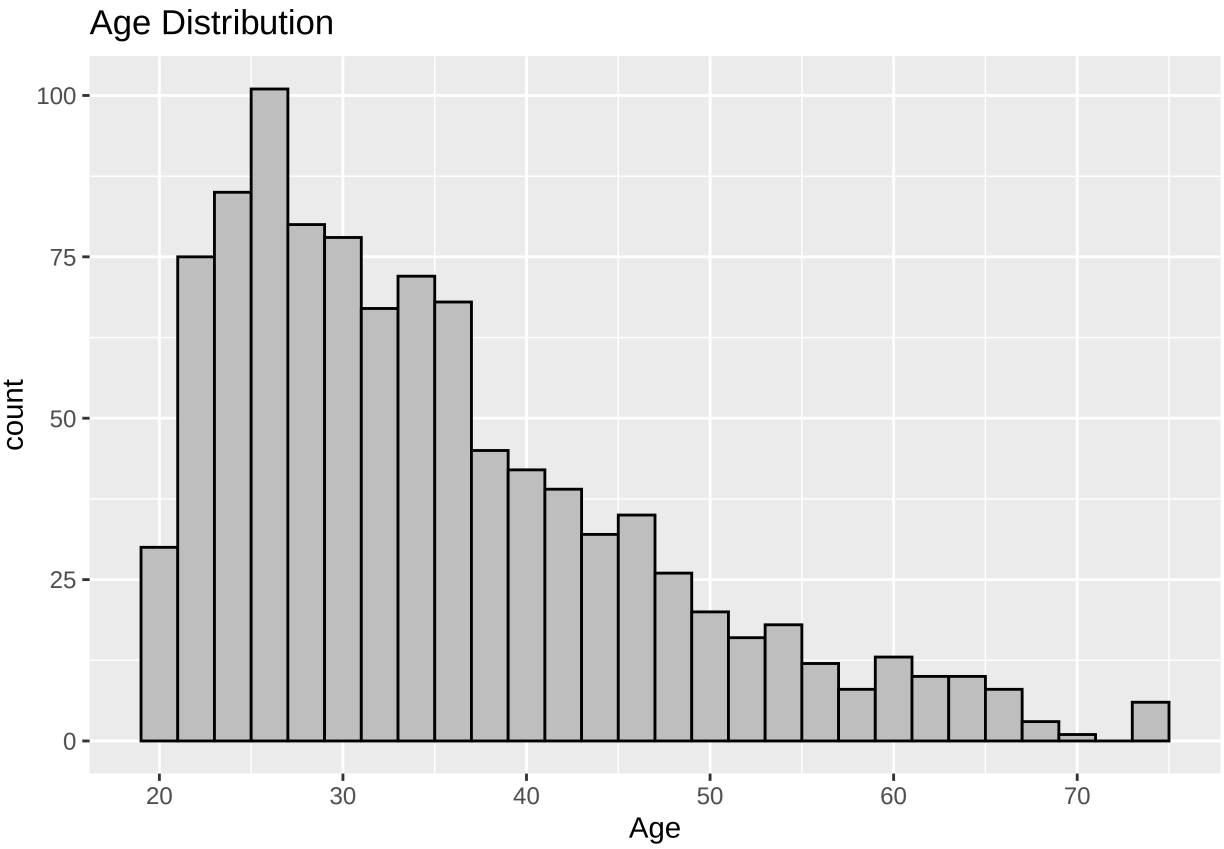
```
## Let's have a look at the sex distribution

credit_data %>% dplyr::count(Sex = factor(Sex)) %>%
   mutate(Percent = prop.table(n)) %>%
   ggplot(aes(x = Sex, y = Percent, fill = Sex, label = scales::percent(Percent))) +
   geom_col(position = 'dodge') +
   geom_text(position = position_dodge(width = .9),     # move to center of bars
             vjust = -0.5,     # nudge above top of bar
             size = 3) +
   scale_y_continuous(labels = scales::percent) +
   labs(title = "Gender Distribution")
```



The males are 69% of the distribution while the females are 31%, simply 690 male and 310 females.

2.2.2 Age Distribution

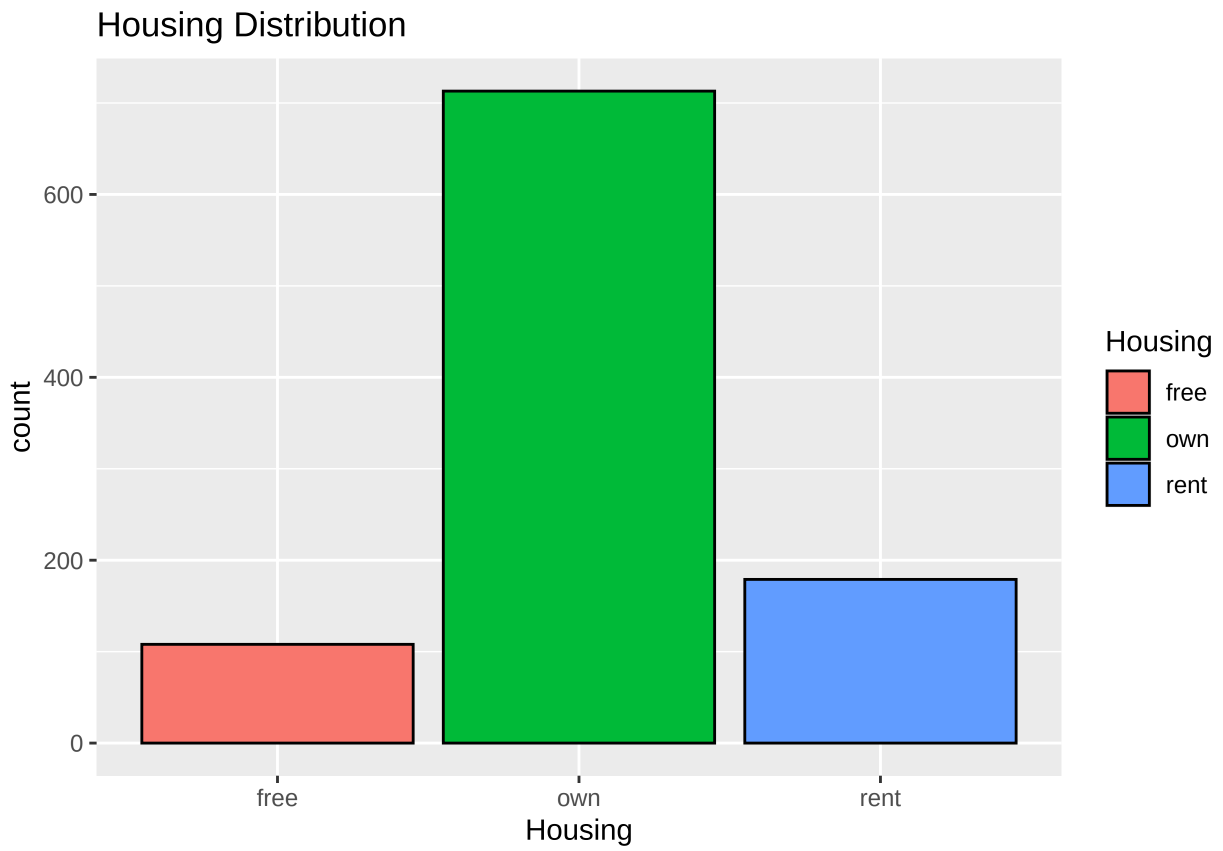
```
## Let's have a look at the age distribution

credit_data %>% ggplot(aes(x = Age, y = ..count.., label = Age)) +
  geom_histogram(binwidth = 2, colour = "black", fill = "grey") +
  scale_x_continuous(breaks = seq(0, 80, by = 10)) +
  labs(title = "Age Distribution")
```



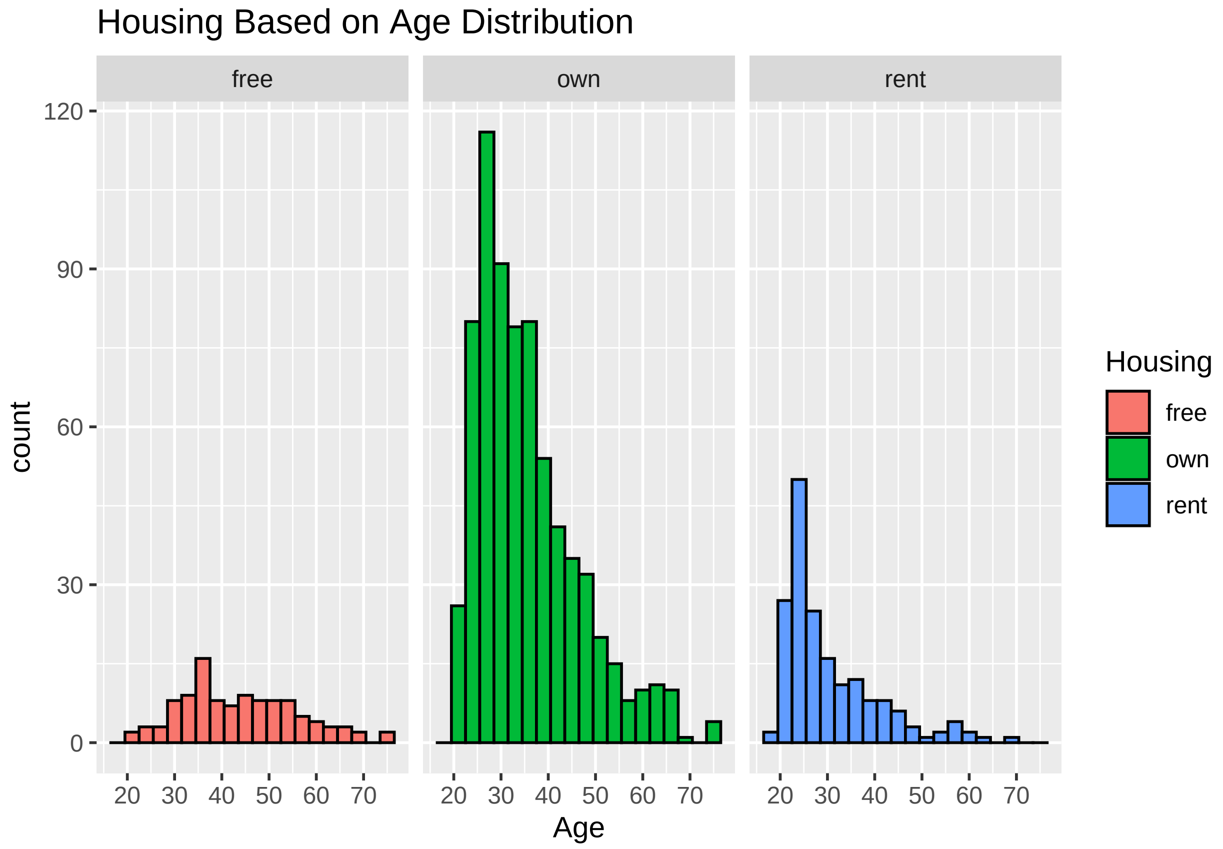
The age graph is positively skewed, indicating that most of the customers in the dataset are between the ages of 20 and 40.

2.2.3 Housing Distribution.

```
## Let's have a look at the house distribution

credit_data %>% ggplot(aes(Housing, fill = Housing)) +
  geom_histogram(stat = "count" , color = "black") +
  labs(title = "Housing Distribution")
```



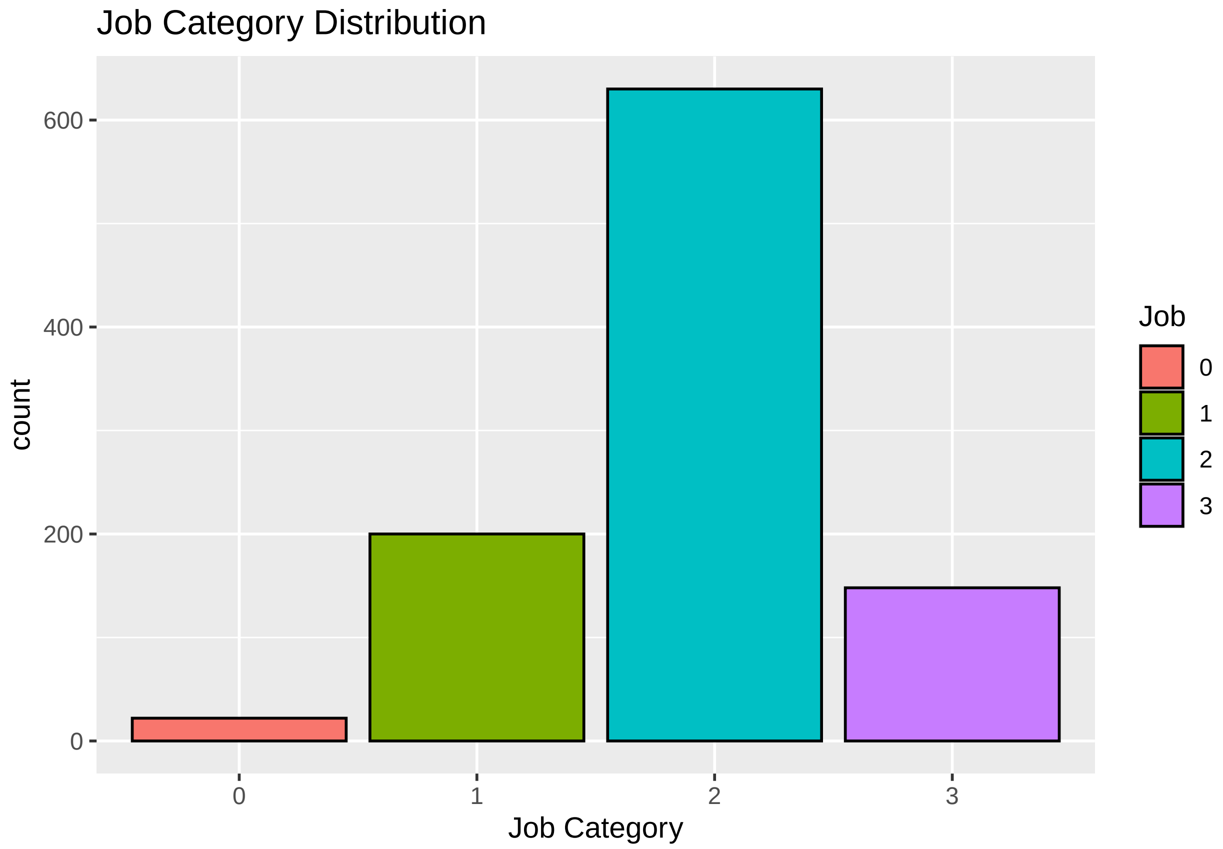
```
## Let's have a look at the Housing Based on Age Distribution

credit_data %>% ggplot(aes(Age, fill = Housing)) +
  geom_histogram(binwidth = 3, color = "black")+
  facet_wrap(~ Housing, ncol = 3) +
  scale_x_continuous(breaks = seq(0, 80, by = 10)) +
  labs(title = "Housing Based on Age Distribution")
```



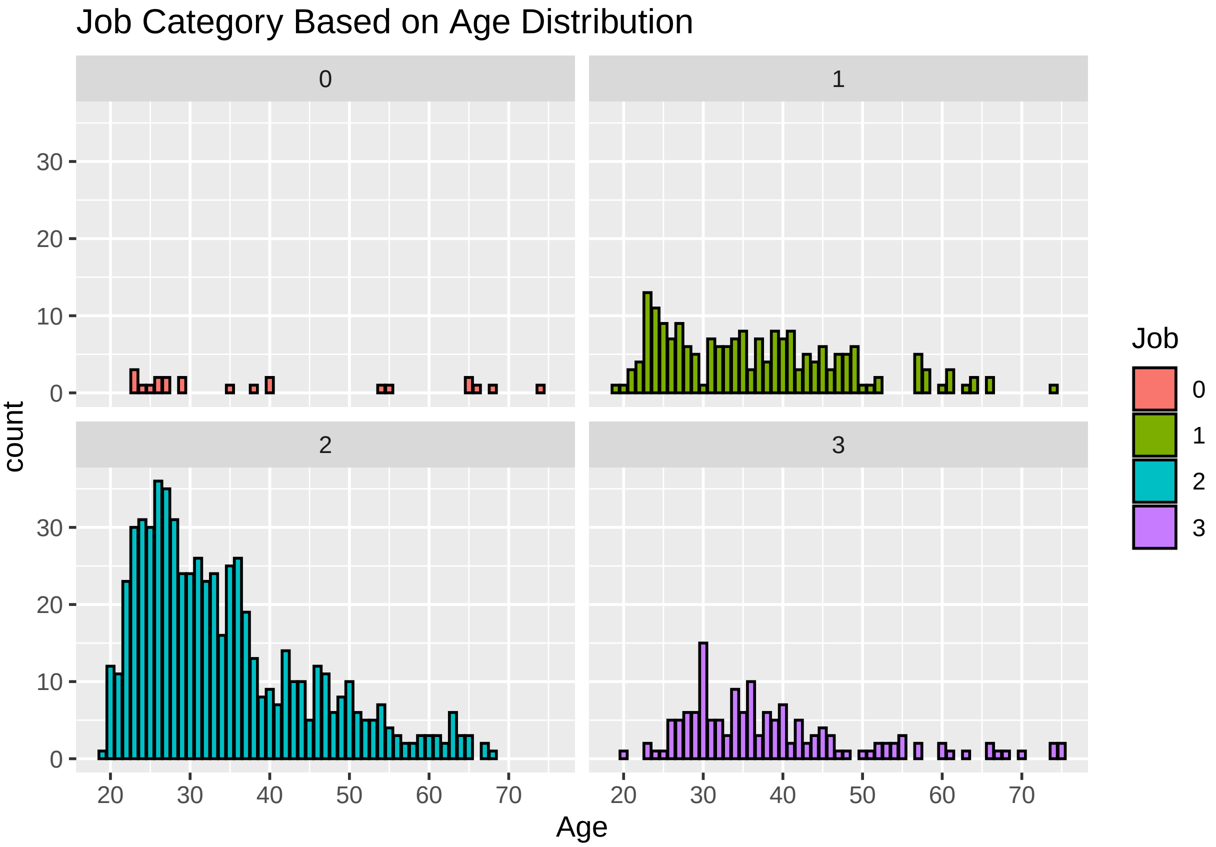
Most of the customers are house owners, with the majority being young.

2.2.4 Job Category Distribution.

```
## Let's have a look at the Job Category Distribution

credit_data %>% mutate(Job = factor(Job)) %>% ggplot(aes(Job , fill = Job)) +
  geom_histogram(stat = "count" , color = "black") +
  labs(title = "Job Category Distribution",
       x = "Job Category")
```



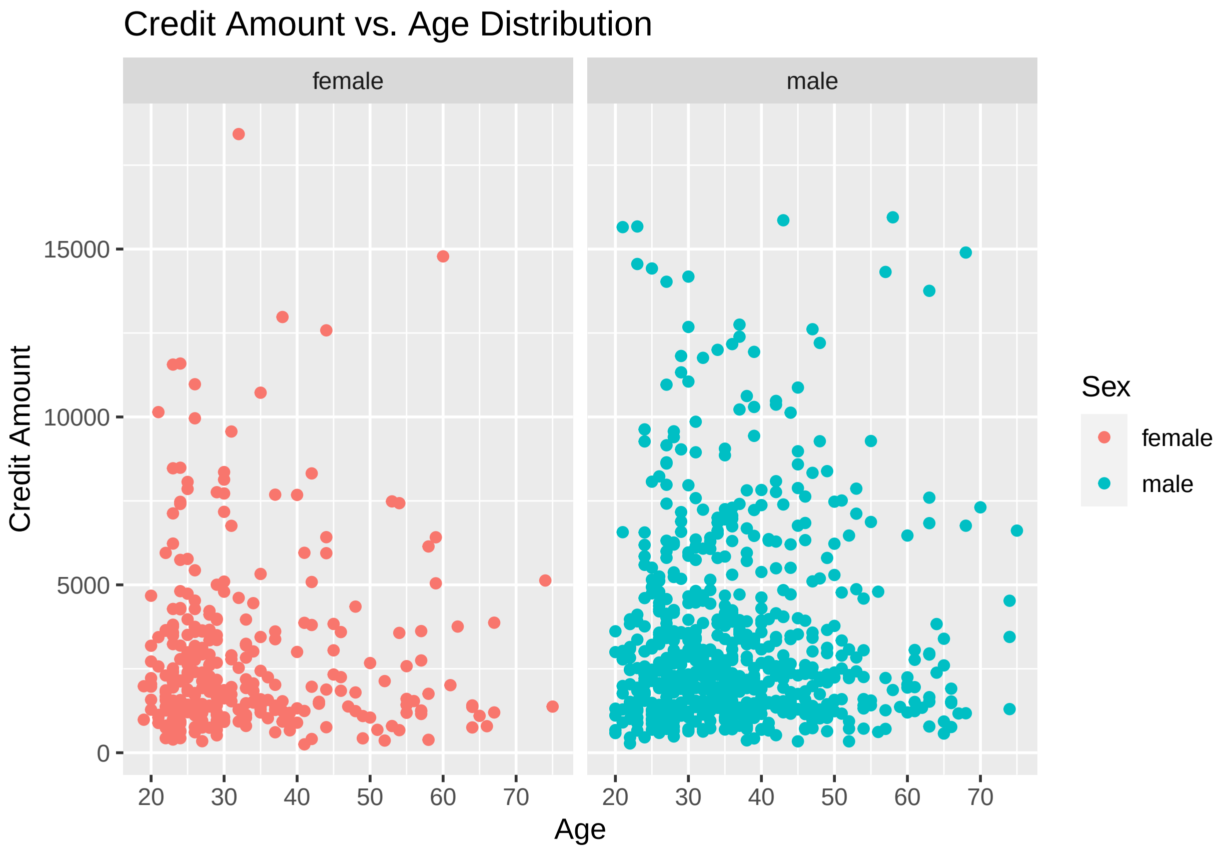
```
## Let's have a look at the Job Category Based on Age Distribution

credit_data %>% mutate(Job = factor(Job)) %>% ggplot(aes(Age, fill = Job)) +
  geom_histogram(stat = "count" , binwidth = 3, color = "black") +
  facet_wrap(~ Job, ncol = 2) +
  scale_x_continuous(breaks = seq(0, 80, by = 10)) +
  labs(title = "Job Category Based on Age Distribution",
       x = "Age")
```



The majority of customers are in job category 2 (skilled) are aged between 20 and 40. We can also see that many customers in job category 3 (highly skilled) are within the same age bracket.

2.2.5 Credit Amount Distribution.

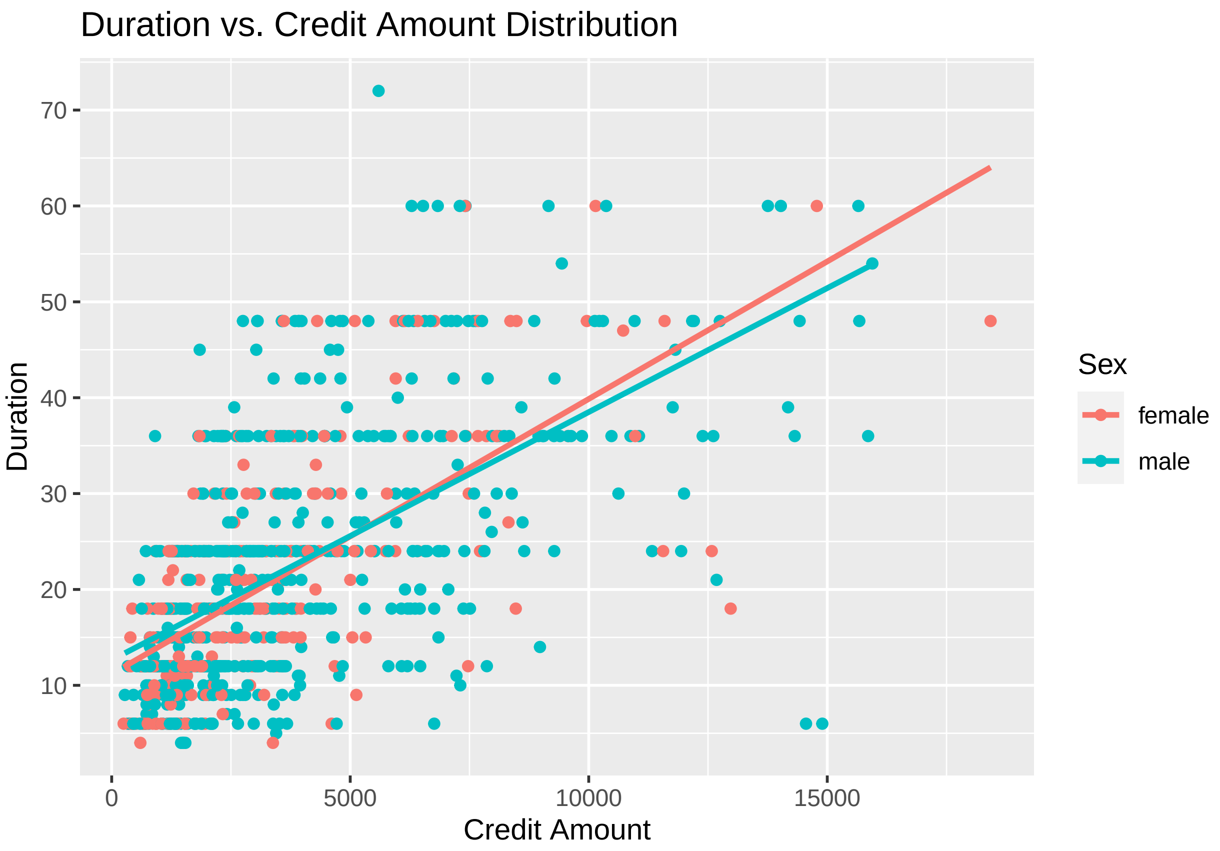
```
## Let's Look at the Credit Amount vs. Age Distribution based on Gender

credit_data %>% ggplot(aes(Age, Credit.amount, color = Sex)) +
  geom_point()+
  scale_x_continuous(breaks = seq(0, 80, by = 10)) +
  facet_wrap(~ Sex, ncol = 2) +
  labs(title = "Credit Amount vs. Age Distribution",
       y = "Credit Amount")
```



Most individuals between the ages of 20 and 40 take loans worth 5,000 and less.

2.2.6 Duration vs. Credit Amount Distribution.

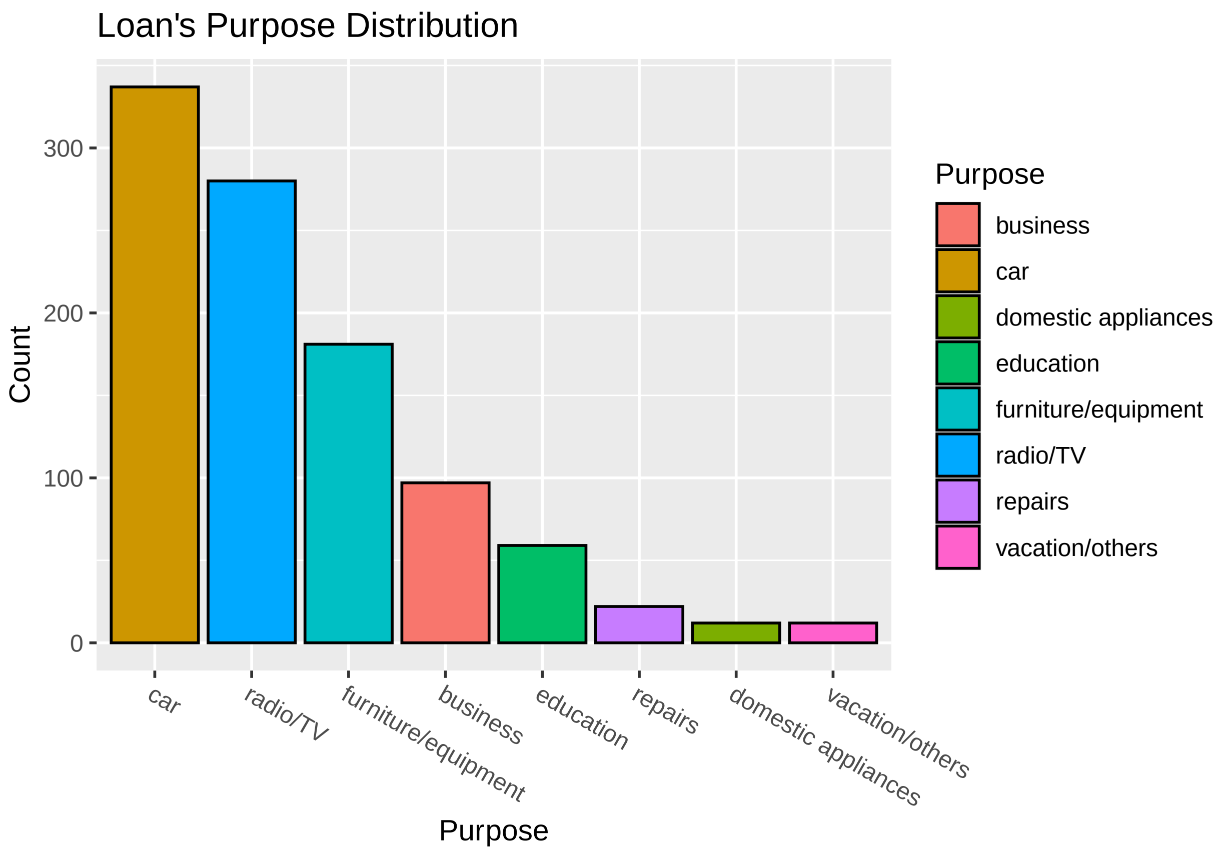
```
## Let's look at Duration vs. Credit Amount Distribution

credit_data %>% ggplot(aes(Credit.amount, Duration, colour = Sex)) +
  geom_point() +
  stat_smooth(formula = "y ~ x", method = "lm", se = FALSE, span = 10) +
  scale_y_continuous(breaks = c(10, 20, 30, 40, 50, 60, 70)) +
  labs(title = "Duration vs. Credit Amount Distribution",
       x = "Credit Amount")
```



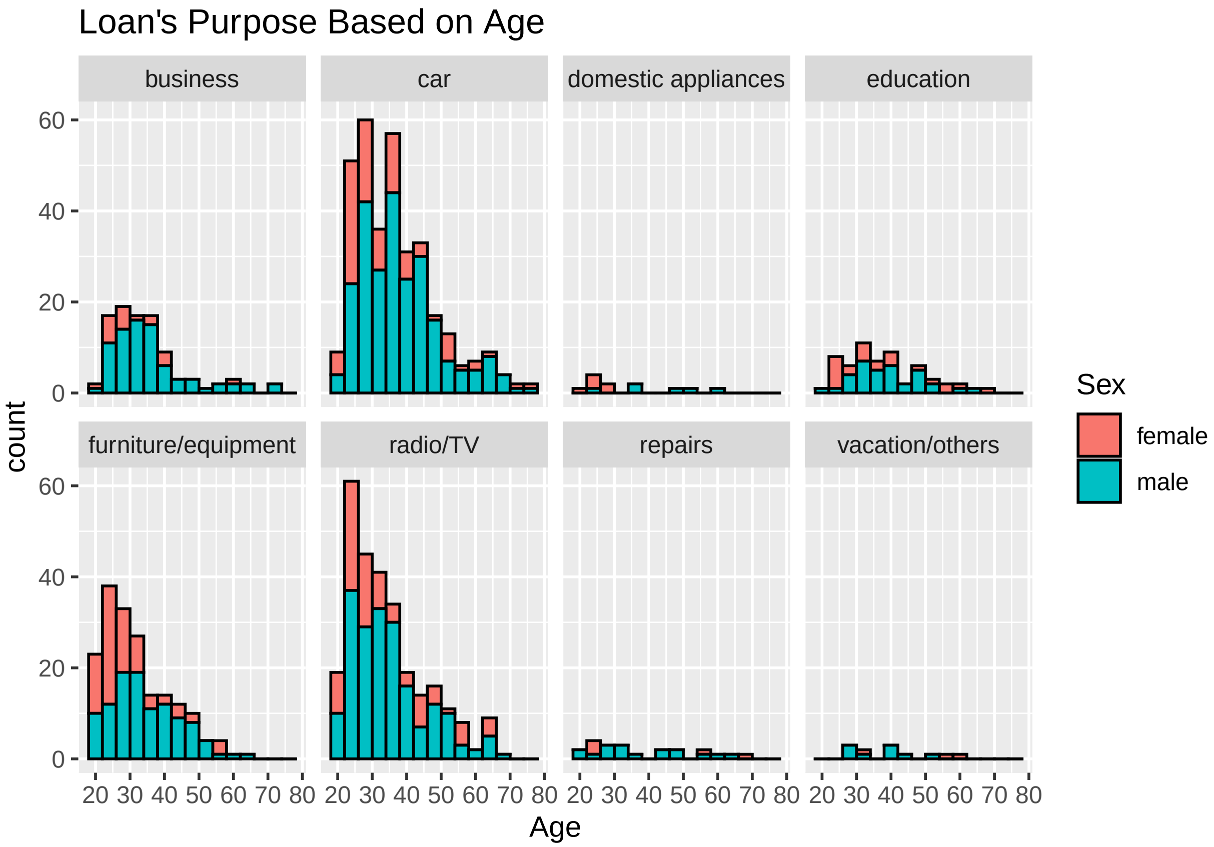
The plot shows that there is a positive linear relationship on the duration and credit amount distribution. Explaining that as the credit amount increases, the loan duration increases as well. There is also no substantial distinction between males and females.

2.2.7 Loan's Purpose Distribution.

```
## Let's look at Which purpose got more loans

credit_data %>% group_by(Purpose) %>%
  summarise(Count = n()) %>%
  arrange(desc(Count)) %>%
  data.frame(.) %>%
  ggplot(aes(x = reorder(Purpose, -Count),  y = Count , fill = Purpose)) +
  geom_bar(stat = "identity", colour = "black") +
  labs(x = "Purpose") +
  theme(axis.text.x = element_text(angle = -30, vjust = 1, hjust = 0)) +
  labs(title = "Loan's Purpose Distribution")
```



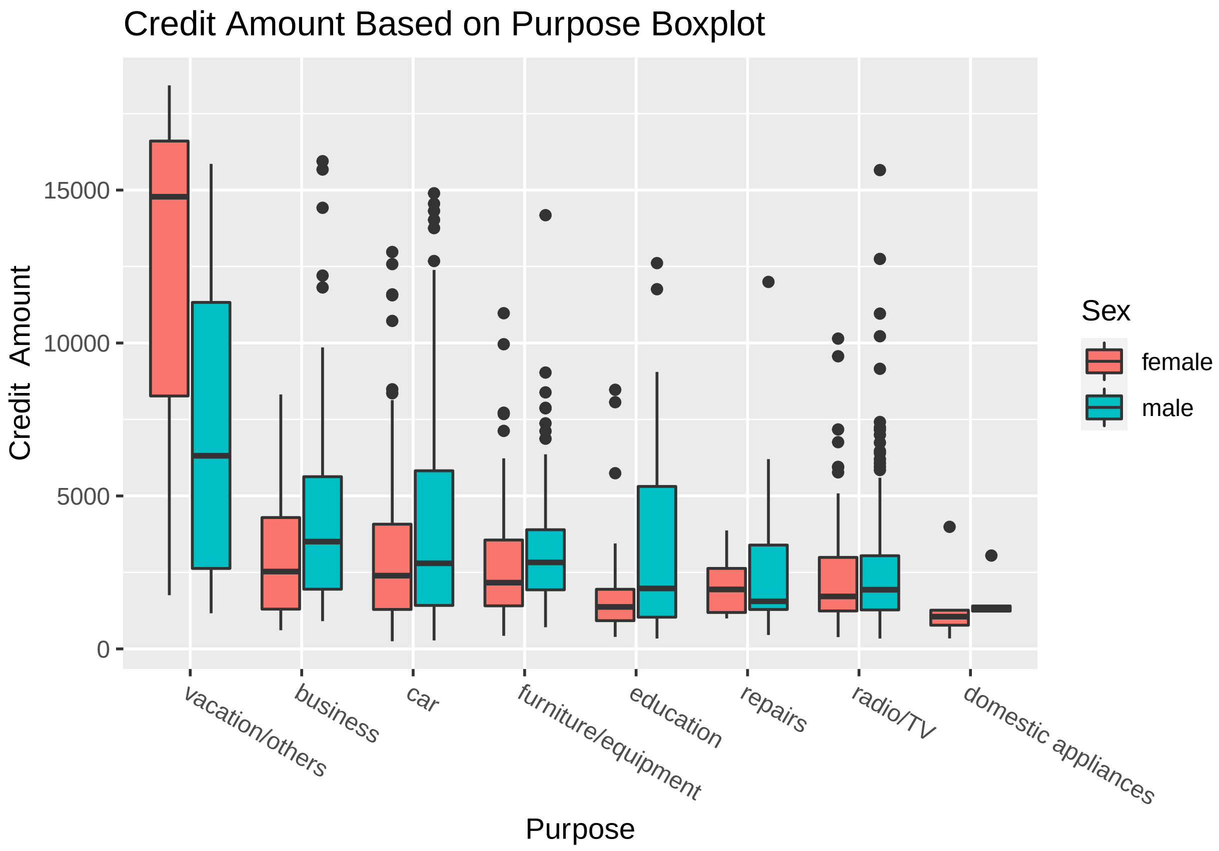
```r
## Let's look at the Loan's Purpose Based on Age

credit_data %>% ggplot(aes(Age, fill = Sex)) +
  geom_histogram(binwidth = 4, colour = "black") +
  facet_wrap(~ Purpose, ncol = 4) +
  scale_x_continuous(breaks = seq(0, 80, by = 10)) +
  labs(title = "Loan's Purpose Based on Age")
```



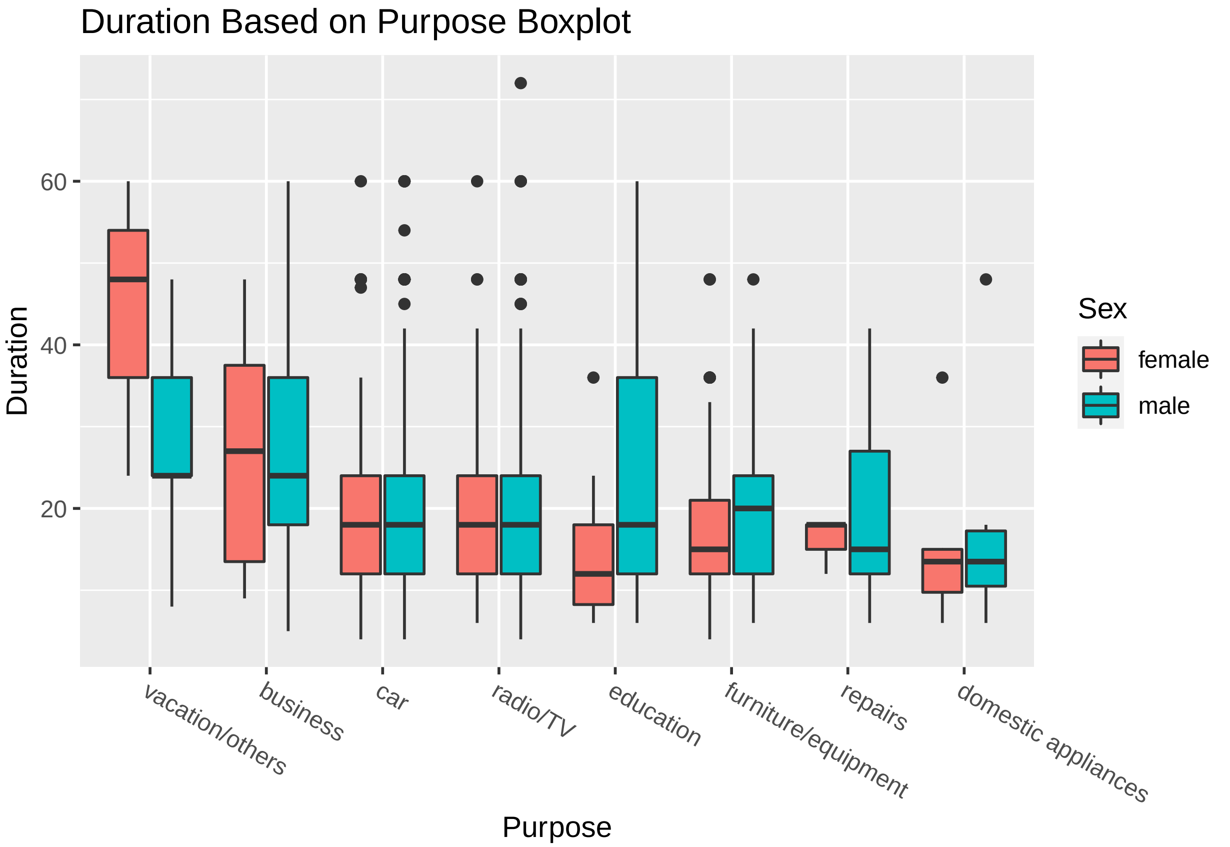
It is clear that most of the loans are taken by the young age group, in which the bulk of the loans applied are for cars, followed by radio/tv then furniture/equipment, with vacation/others attracting the least number of creditors.

2.2.8 Credit Amount and Duration Based on Purpose Boxplots.

```r
## Boxplot of Credit Amount Based on Purpose

credit_data %>%
  ggplot(aes(x = reorder(Purpose, -Credit.amount), Credit.amount, fill = Sex)) +
  geom_boxplot() +
  theme(axis.text.x = element_text(angle = -30, vjust = 1, hjust = 0)) +
  labs(title = "Credit Amount Based on Purpose Boxplot",
       x = "Purpose" ,
       y = "Credit  Amount")
```



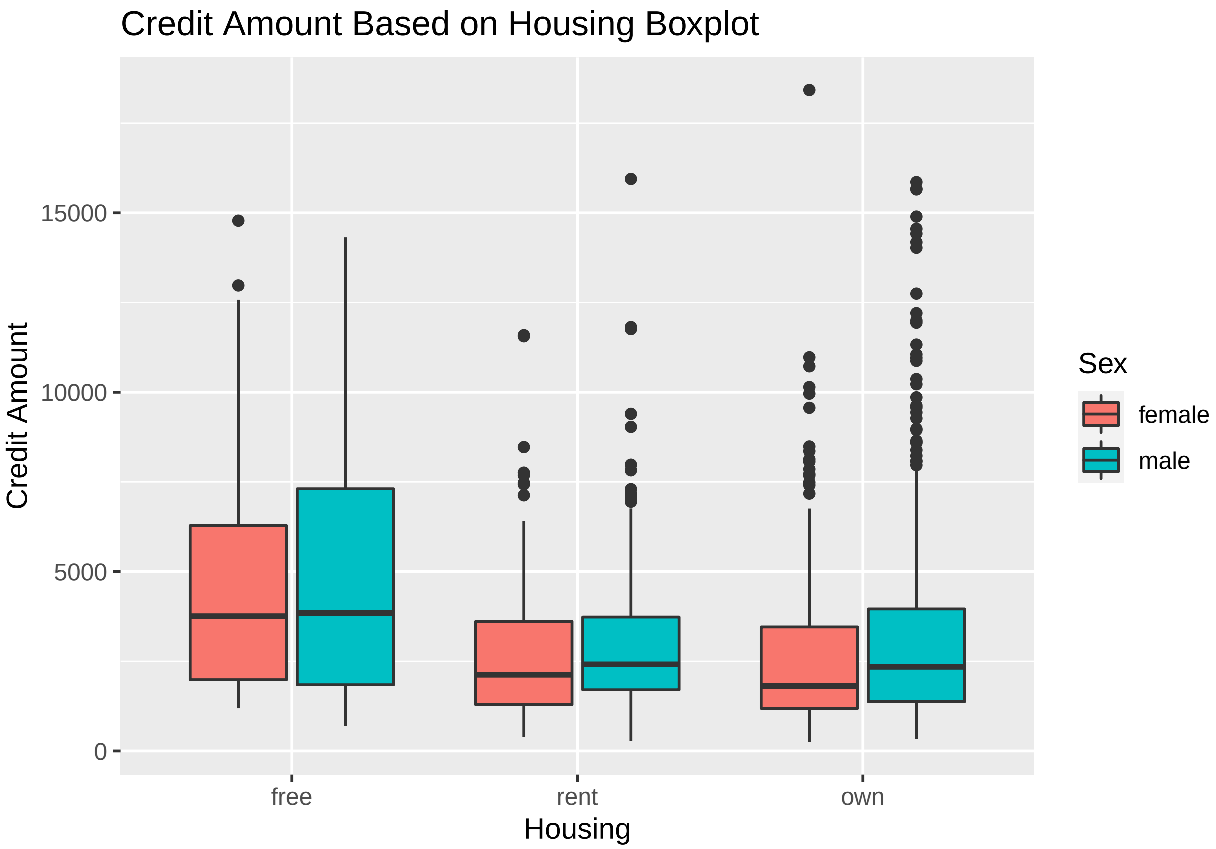
```r
## Boxplot of Duration Based on Purpose Boxplot

credit_data %>% ggplot(aes(x = reorder(Purpose, -Duration), Duration, fill = Sex)) +
  geom_boxplot() +
  theme(axis.text.x = element_text(angle = -30, vjust = 1, hjust = 0)) +
  labs(title = "Duration Based on Purpose Boxplot",
       x = "Purpose" ,
       y = "Duration")
```



The box plots above reveal that large sums of credit are provided for vacation/others, with females dominating this sector, while domestic appliances received less. The large sums of credit also explain the reason as to why a longer duration of loan payments was given to vacation/others while domestic appliances were receiving less. There are also outliers in different purposes, indicating huge amounts and longer duration. Generally, notwithstanding the vacation/others purpose, there is no gender disparity.

2.2.9 Credit Amount Based on Housing Boxplot.

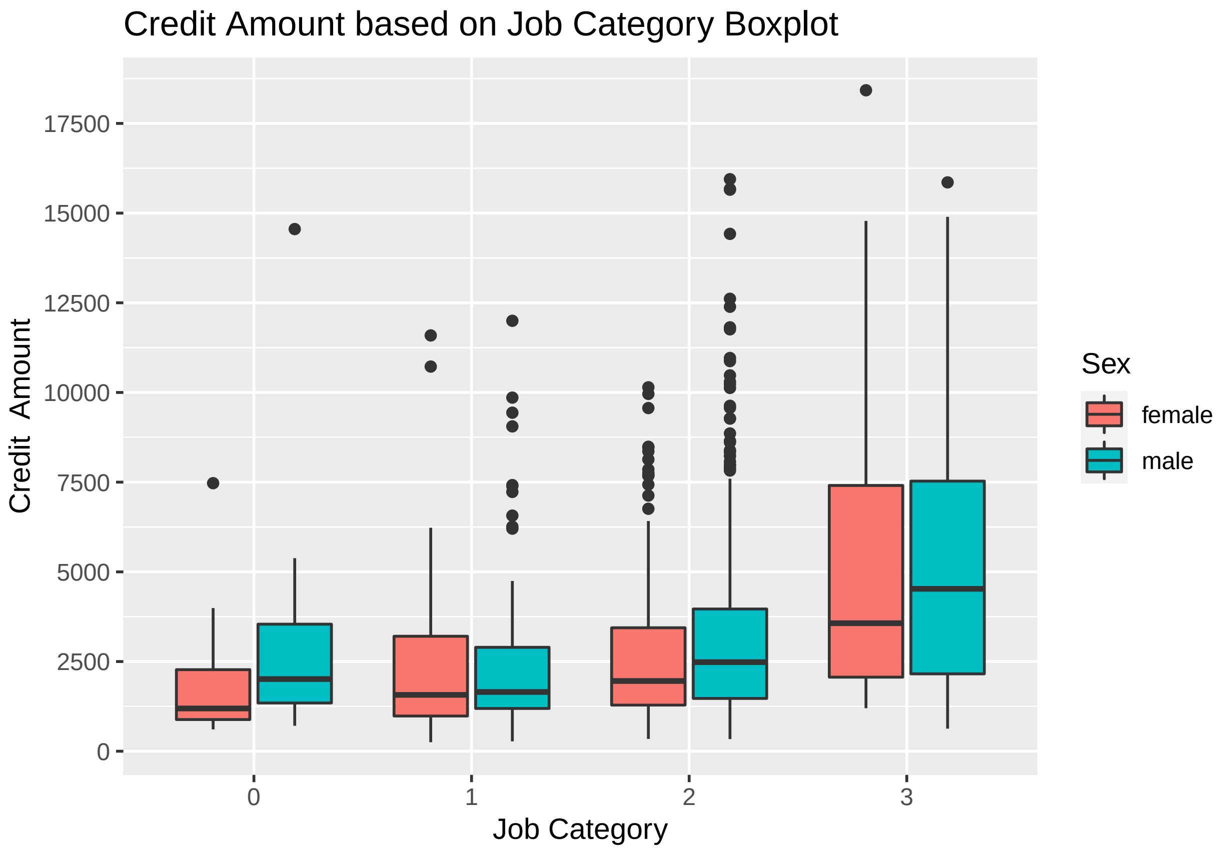
```
## Boxplot of Credit Amount Based on Housing

credit_data %>%
  ggplot(aes(x = reorder(Housing, -Credit.amount), Credit.amount, fill = Sex)) +
  geom_boxplot() +
  labs(title = "Credit Amount Based on Housing Boxplot",
       x = "Housing",
       y = "Credit Amount")
```



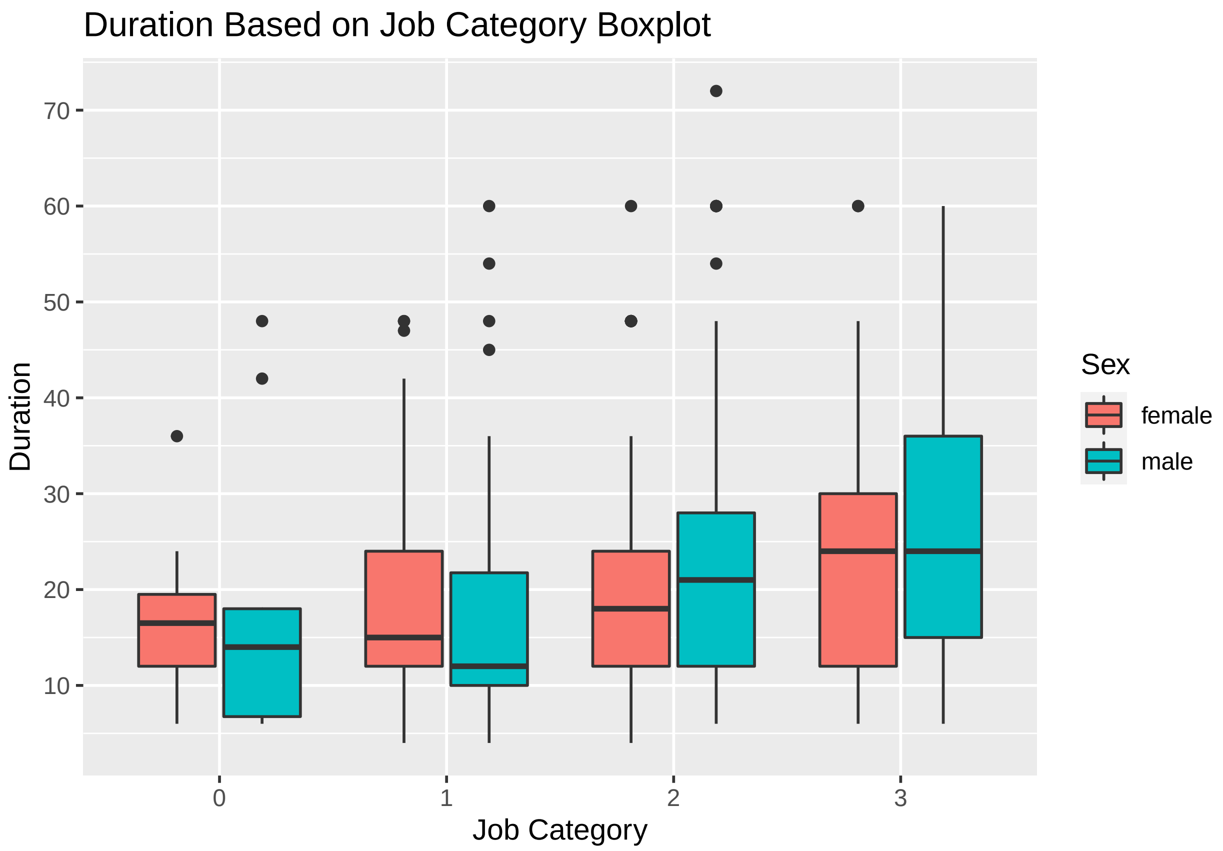
In terms of housing, we can see that there is no much variation on average despite customers with free housing having taken marginally higher amounts of loans. Homeowners have the largest number of outliers, indicating huge amounts of loans given to some customers in this category.

2.2.10 Credit Amount and Duration Based on Job Category Boxplots.

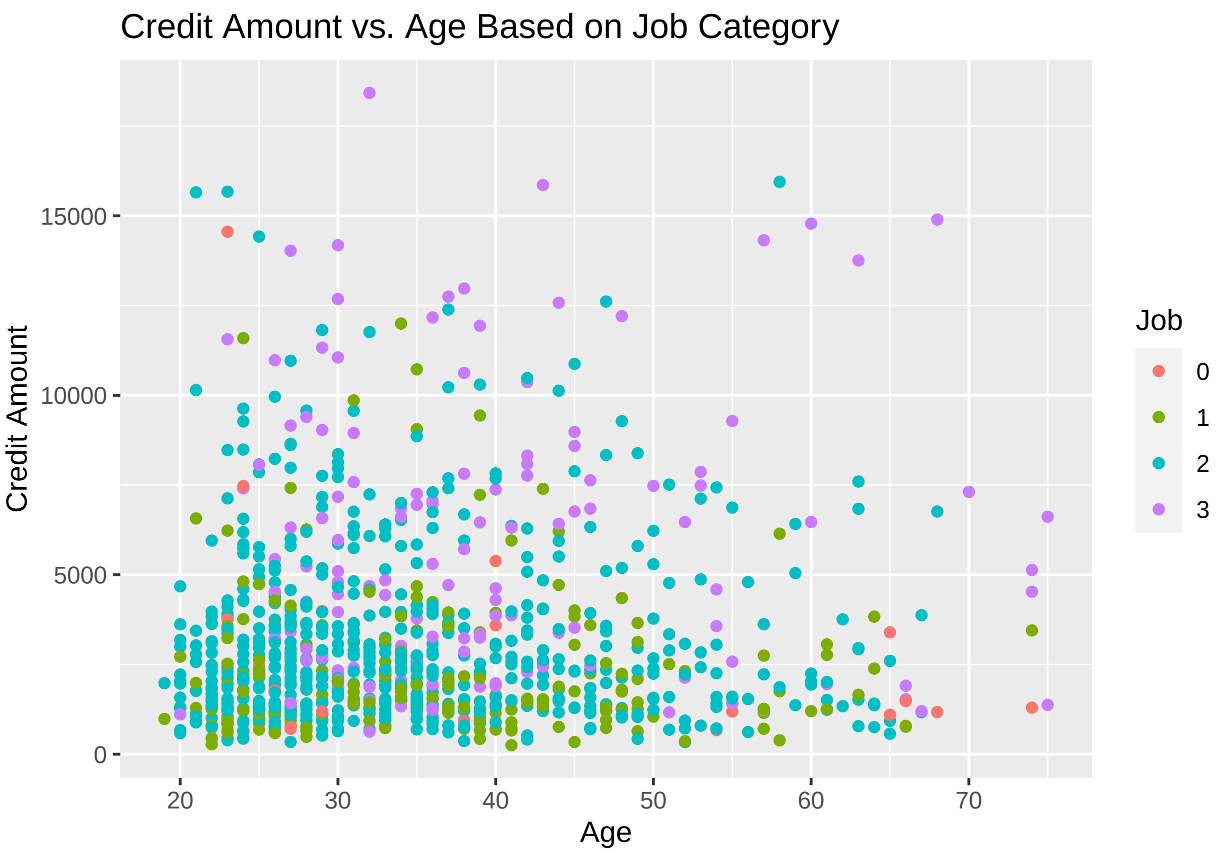
```
## Boxplot of Credit Amount based on Job Category

credit_data %>% ggplot(aes(as.factor(Job), Credit.amount, fill = Sex)) +
  scale_y_continuous(breaks = seq(0, 18000, by = 2500)) +
  geom_boxplot() +
  labs(title = "Credit Amount based on Job Category Boxplot",
       x = "Job Category",
       y = "Credit  Amount")
```



```
## Boxplot of Duration Based on Job Category

credit_data %>% ggplot(aes(as.factor(Job), Duration, fill = Sex)) +
  scale_y_continuous(breaks = c(10, 20, 30, 40, 50, 60, 70))  +
  geom_boxplot() +
  labs(title = "Duration Based on Job Category Boxplot",
       x = "Job Category",
       y = "Duration")
```



```
## Let's look at Credit Amount vs. Age Based on Job Category

credit_data %>% mutate(Job = as.factor(Job)) %>%
  ggplot(aes(Age, Credit.amount, color = Job)) +
  geom_point() +
  scale_x_continuous(breaks = seq(0, 80, by = 10)) +
  labs(title = "Credit Amount vs. Age Based on Job Category",
       y = "Credit Amount")
```



As far as the job category is concerned, customers in job category three (highly skilled) took the highest amount of credit and had the longest loan repayment duration. At the same time, customers in job category zero (unskilled and non-resident) took the least amount of loans and had the shortest period for the loan repayment. Again, there is no gender disparity.

```
## Let's check for correlation in the data

credit_data %>% select(Job, Age, Credit.amount, Duration) %>% cor(.) %>%
  knitr::kable(caption = "Correlation.")
```

Table 6: Correlation.

	Job	Age	Credit.amount	Duration
Job	1.0000000	0.0156732	0.2853853	0.2109097
Age	0.0156732	1.0000000	0.0327164	-0.0361364
Credit.amount	0.2853853	0.0327164	1.0000000	0.6249842
Duration	0.2109097	-0.0361364	0.6249842	1.0000000

The table above shows the different type of correlations between the variables, with credit amount and duration having the strongest positive correlation of 0.62498.

2.3 Analysis.

We proceed to analyze the dataset using both the supervised and unsupervised learning algorithms.

2.3.1 Supervised Machine Learning Classification Techniques.

In supervised machine learning, algorithms learn from labeled data. In this case, we shall apply the classification algorithms: logistic, K-Nearest-Neighbor (KNN), and random forest. This analysis will be concentrating on age, job category, credit amount, and duration to predict the categorical class, sex. We shall then proceed to determine their accuracies on how well they classify and predict the gender.

We first begin by selecting the required variables then scale them. After that, we proceed to split the data set into a train and test set.

```r
## Let's see how the supervised classification algorithm and unsupervised clustering
## algorithms work.

set.seed(1234)
credit_analysis_scaled <- credit_data %>% select(Age, Job, Credit.amount, Duration) %>%
  scale() %>%
  data.frame() # Selecting the columns to be used and scaling them

## Splitting the dataset

credit_analysis_scaled$Sex <- credit_data$Sex # Adding Sex column after scaling the data
y <- credit_analysis_scaled$Sex
train_index <- createDataPartition(y , times = 1, p = 0.2, list = FALSE) # splitting
train_set <- credit_analysis_scaled[-train_index,]
test_set <- credit_analysis_scaled[train_index,]
```

We begin our prediction using the Logistic regression.

```r
## We begin our prediction using the Logistic regression.
options(digits = 3)
set.seed(1234)
train_glm <- train(Sex ~ ., method = "glm", data = train_set) # logistic regression
glm_preds <- predict(train_glm, test_set, type = "raw") # predict gender on the test set
glm_acc <- mean(glm_preds == test_set$Sex) # checking accuracy

accuracy <- tibble(Method = "Logistic Regression", Accuracy = glm_acc)

accuracy
```

```
## # A tibble: 1 x 2
##   Method              Accuracy
##   <chr>                  <dbl>
## 1 Logistic Regression     0.69
```

We apply the K-Nearest Neighbor method (KNN).

```
## We use the K-Nearest-Neighbor.

##KNN
set.seed(1234)
tuning <- data.frame(k = seq(3, 50, 2)) # setting values of k for optimization
train_knn <- train(train_set[,-5],train_set[,5],
                   method = "knn", # knn method
                   tuneGrid = tuning)

knn_preds <- predict(train_knn, test_set) # predicting the gender on the test set
knn_acc <- mean(knn_preds == test_set$Sex) # checking accuracy

accuracy <- bind_rows(accuracy,
                      tibble(Method = "K-Nearest Neighbor", Accuracy = knn_acc))


accuracy
```

```
## # A tibble: 2 x 2
##   Method               Accuracy
##   <chr>                   <dbl>
## 1 Logistic Regression     0.69
## 2 K-Nearest Neighbor      0.7
```

We apply the random forest algorithm.

```
## We use the random forest algorithm.

library(randomForest)
set.seed(1234)
train_rf <- train(train_set[,-5],train_set[,5],
                  method = "rf", # random forest method
                  tuneGrid = data.frame(mtry = c(1,2,3,4)))

rf_preds <- predict(train_rf, test_set) # predicting the gender on the test set
rf_acc <- mean(rf_preds == test_set$Sex)

accuracy <- bind_rows(accuracy,
                      tibble(Method = "Random Forest", Accuracy = rf_acc))

accuracy
```

```
## # A tibble: 3 x 2
##   Method               Accuracy
##   <chr>                   <dbl>
## 1 Logistic Regression    0.69
## 2 K-Nearest Neighbor     0.7
## 3 Random Forest          0.705
```

2.3.2 Supervised Machine Learning Classification Techniques

We now proceed to the unsupervised machine learning clustering algorithms. In this method, the dataset is clustered into clusters that have not been labeled, classified, or categorized. We shall apply both the K-Means and hierarchical clustering techniques. To test for accuracy, we shall extract the clusters and calculate the features' distance to the cluster centers and then select the cluster with minimum distance.

We begin with the K-Means technique.

```
## We now proceed to the unsupervised techniques


set.seed(1234)
predict_kmeans <- function(x, k) {
  centers <- k$centers    # extract cluster centers
  # calculate distance to cluster centers
  distances <- sapply(1:nrow(x), function(i){
    apply(centers, 1, function(y) dist(rbind(x[i,], y)))
  })
  max.col(-t(distances))  # select cluster with min distance to center
}

## We use the K-Means algorithm.

set.seed(1234)
k <- kmeans(train_set[,-5], centers = 4) # setting number of clusters
kmeans_preds <- ifelse(predict_kmeans(test_set[-5], k) == 1, "female", "male")
kmeans_acc <- mean(kmeans_preds == test_set$Sex)

accuracy <- bind_rows(accuracy,
                      tibble(Method = "K-Means Clustering", Accuracy = kmeans_acc))

accuracy
```

```
## # A tibble: 4 x 2
##   Method              Accuracy
##   <chr>                  <dbl>
## 1 Logistic Regression    0.69
## 2 K-Nearest Neighbor     0.7
## 3 Random Forest          0.705
## 4 K-Means Clustering     0.615
```

We then use the hierarchical clustering algorithm.

```
## We use the Hierarchical  Clustering algorithm.

set.seed(1234)
predict_hier <- function(x, c) {
  merge <- c$merge    # extract cluster centers
  # calculate distance to cluster centers
  distances <- sapply(1:nrow(x), function(i){
    apply(merge, 1, function(y) dist(rbind(x[i,], y)))
  })
  max.col(-t(distances))  # select cluster with min distance to center
}

c <- hclust (dist(train_set[,-5]), method = "ward.D")
hierach_preds <- ifelse(predict_hier(test_set[-5], c) == 1, "female", "male")
hierach_acc <- mean(hierach_preds == test_set$Sex)

accuracy <- bind_rows(accuracy,
             tibble(Method = "Hierarchical  Clustering", Accuracy = hierach_acc)) %>%
             knitr::kable(caption = "Algorithms' Accuracy")

accuracy
```

Table 7: Algorithms' Accuracy

Method	Accuracy
Logistic Regression	0.690
K-Nearest Neighbor	0.700
Random Forest	0.705
K-Means Clustering	0.615
Hierarchical Clustering	0.690

3. RESULTS

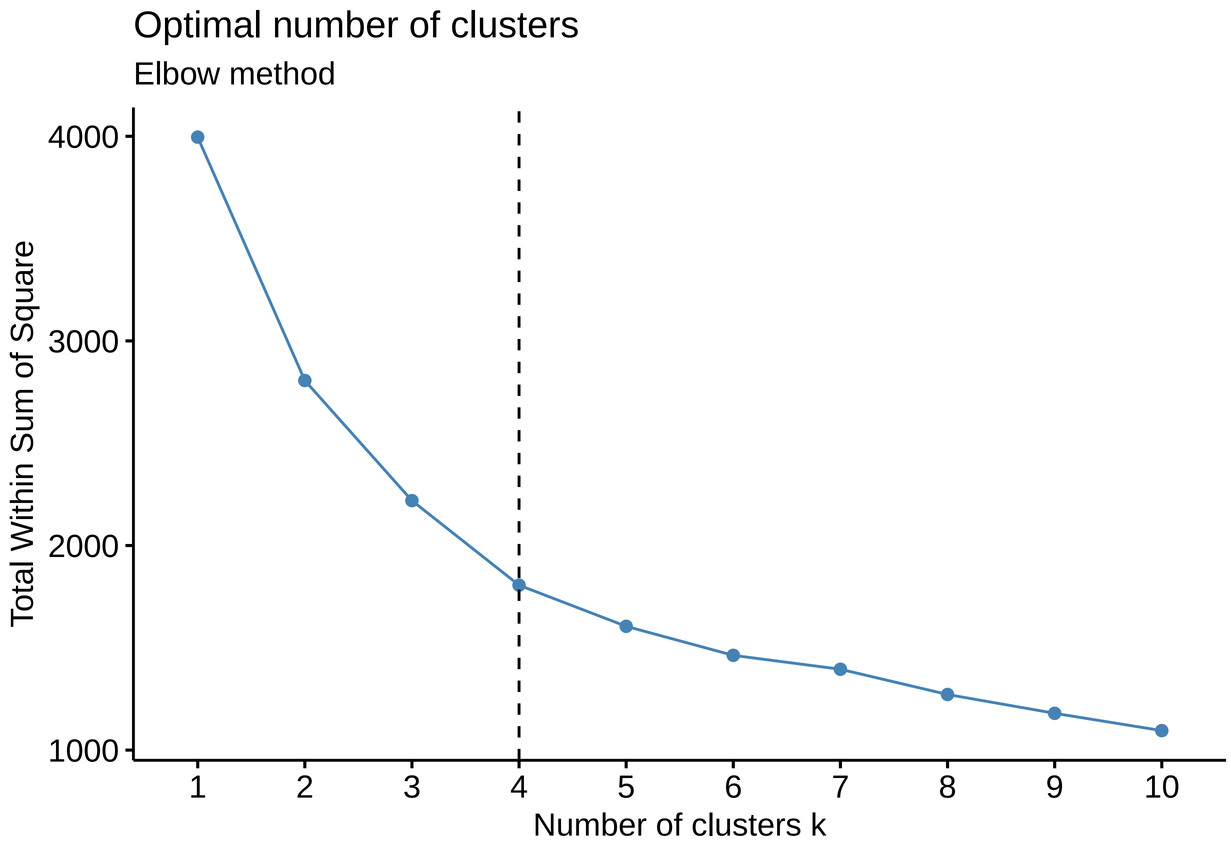
From the analysis, the random forest algorithm, a supervised learning method, performed the best generally in classifying and predicting the gender, with an accuracy of 0.705. Down to the unsupervised learning category, the hierarchical clustering algorithm did the best compared to the K-Means algorithm.

Therefore, we now proceed to apply the hierarchical clustering technique to the entire dataset to determine how different customers are clustered.

3.1 Number of Clusters.

First, we use the K-Means elbow method to determine the optimal number of clusters, applying it to the hierarchical clustering technique. In this method, we calculate the Within-Cluster-Sum of squared errors (WSS) for various values of k, then select the value of k for which WSS begins to decline. The optimal value k is evident as an elbow in the WSS-versus-k plot, which looks like an arm.

```r
## We apply the elbow method

credit_analysis <- credit_data %>%
  select(Age, Job, Credit.amount, Duration)

credit_scaled <- as.data.frame(lapply(credit_analysis, scale)) # Scale the data
fviz_nbclust(credit_scaled, kmeans, method = "wss") +
  geom_vline(xintercept = 4, linetype = 2) +
  labs(subtitle = "Elbow method") ## Check Elbow
```



From the plot, k = 4 is the optimal number since it's at the "elbow" part.

```
## Applying hierarchical clustering using ward.D method and setting clusters to 4

hierach_ward <- eclust(credit_scaled, "hclust", hc_method = "ward.D", k = 4)

## We make a table on how the gender was distributed in the clusters

hierach_clusters <- hierach_ward$cluster
table(hierach_clusters, credit_data$Sex)
```

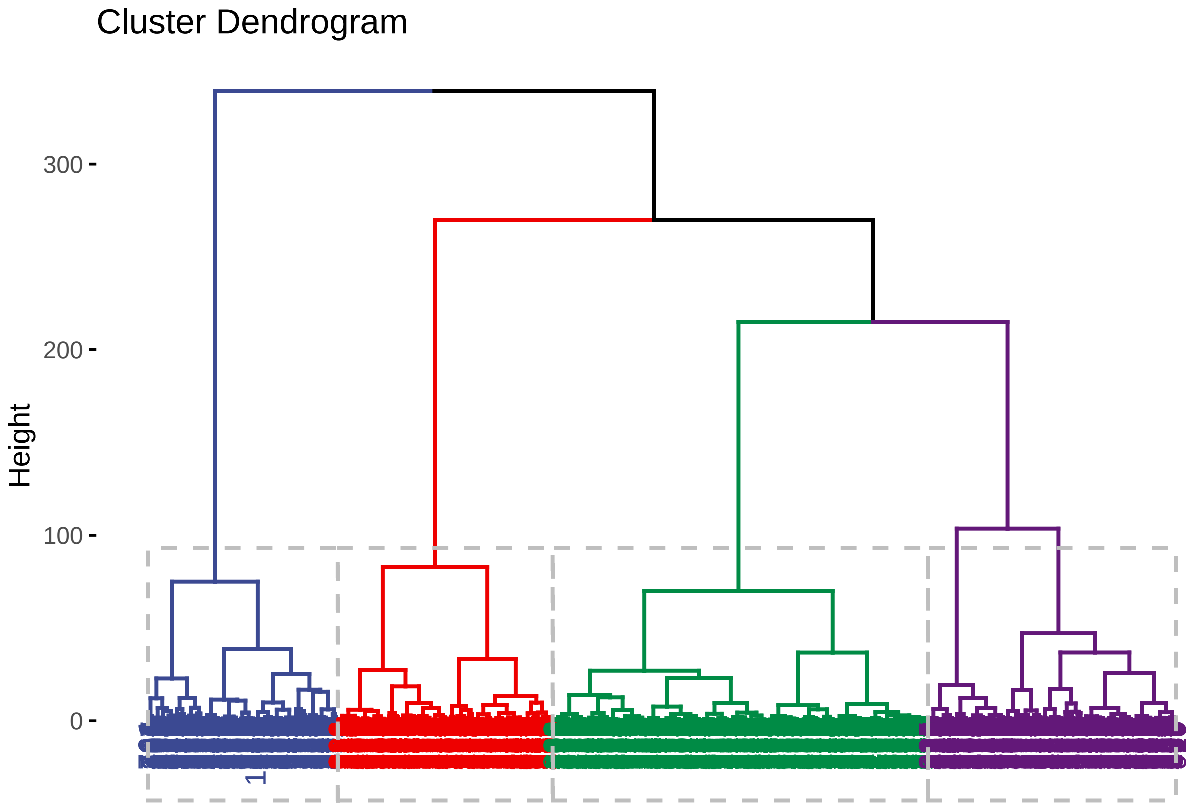
```
##
## hierach_clusters female male
##                1     51  190
##                2     45  140
##                3     72  137
##                4    142  223
```

The table above shows how gender was distributed among the four clusters.

3.2 Hierarchical Clustering.

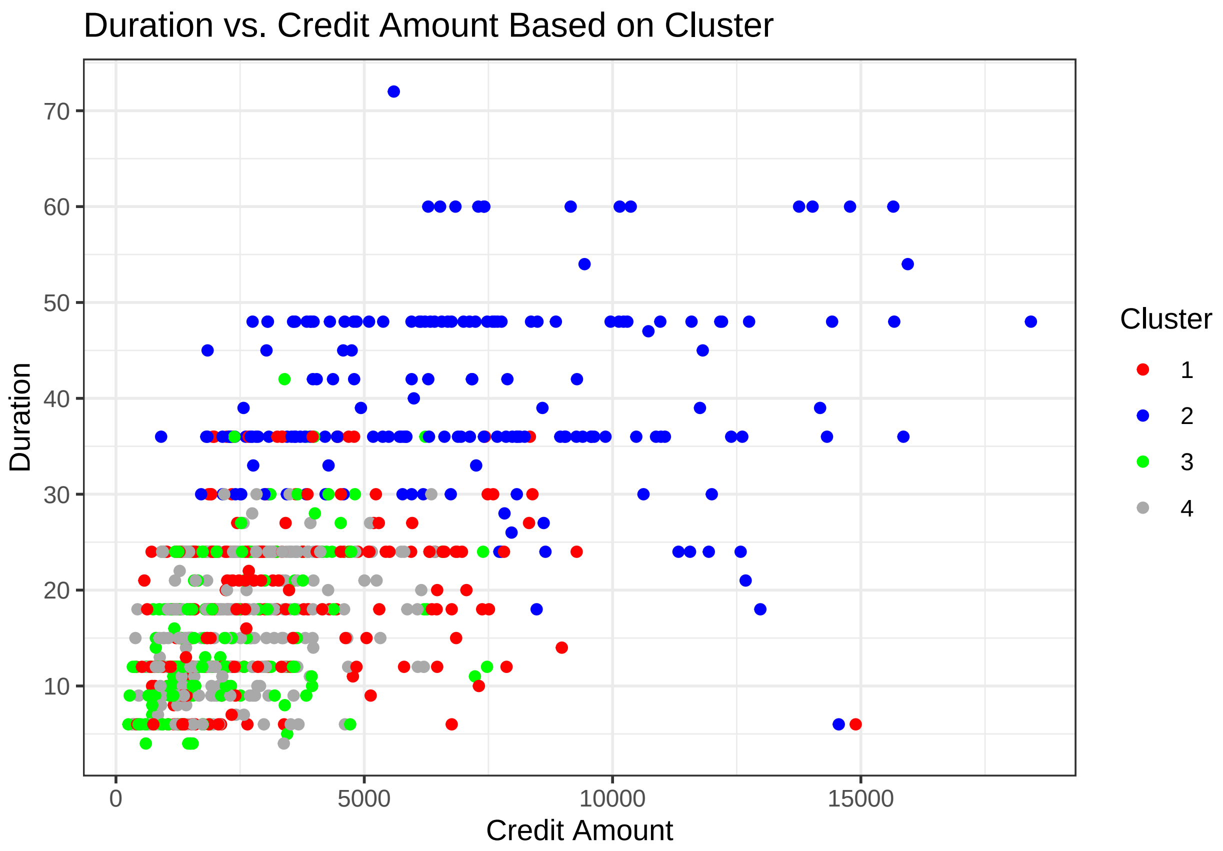
```
table(hierach_ward$cluster)
```

```
##
##   1   2   3   4
## 241 185 209 365
## We create the Tree diagram i.e. Dendrogram graph

fviz_dend(hierach_ward, show_labels = T, palette = "aaas", color_labels_by_k = TRUE,
          rect = TRUE)
```

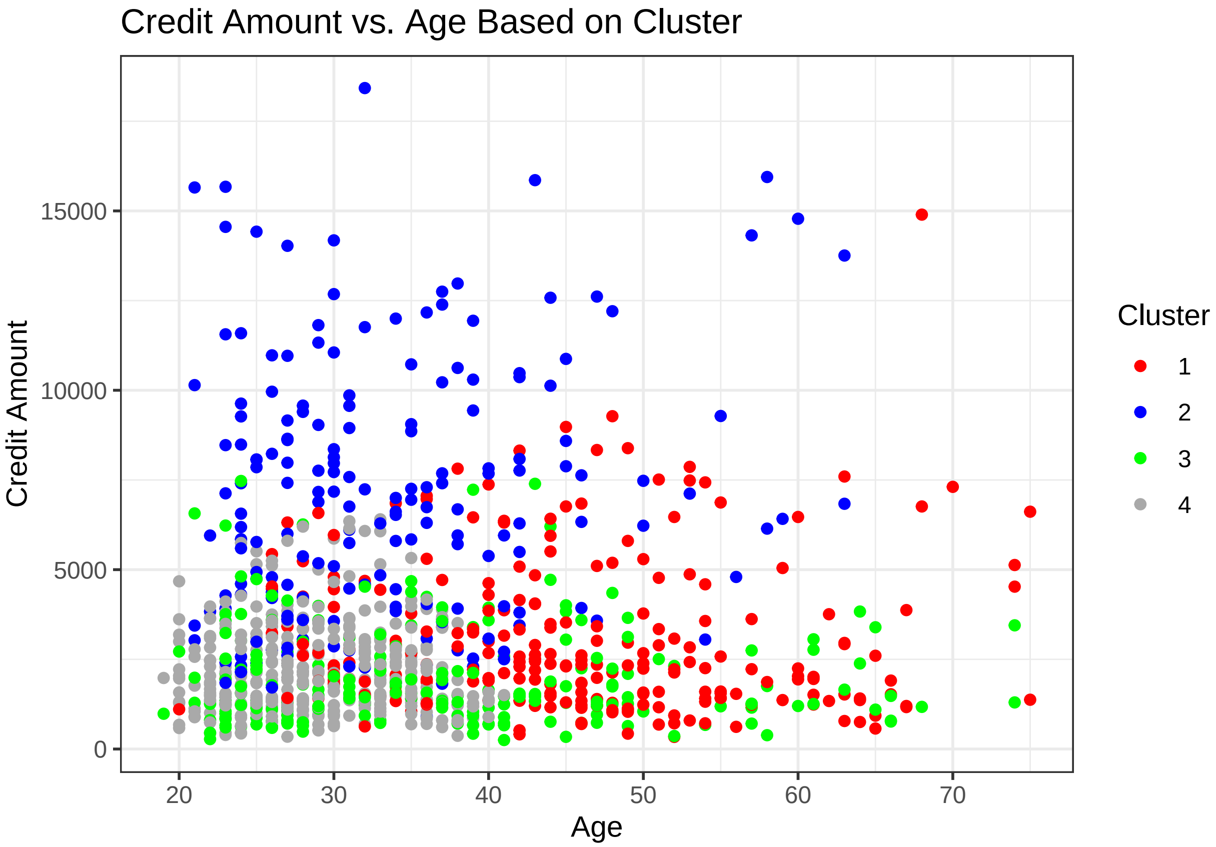


There are four clusters, with most customers being in the fourth cluster and the least being in the second cluster. However, the clusters seem to be evenly distributed with little variation. The cluster dendrogram shows how the clusters are distributed.

```
## Plot showing Duration vs. Credit Amount Based on Cluster

credit_analysis$Cluster <- as.factor(hierach_ward$cluster)

credit_analysis %>%
  ggplot(aes(Credit.amount, Duration, col = Cluster)) +
  geom_point() +
  theme_bw() +
  scale_y_continuous(breaks = c(10, 20, 30, 40, 50, 60, 70)) +
  scale_color_manual(breaks = c("1", "2", "3", "4"),
                     values=c("red", "blue", "green", "darkgrey")) +
  labs(title = "Duration vs. Credit Amount Based on Cluster",
       x = "Credit Amount")
```



```
## Plot showing Credit Amount vs. Age Based on Cluster

credit_analysis %>%
  ggplot(aes( Age, Credit.amount, col = Cluster)) +
  geom_point() +
  theme_bw() +
  scale_x_continuous(breaks = seq(0, 80, by = 10)) +
  scale_color_manual(breaks = c("1", "2", "3", "4"),
                     values = c("red", "blue", "green", "darkgrey")) +
  labs(title = "Credit Amount vs. Age Based on Cluster",
       y = "Credit Amount",
       x = "Age")
```



The duration versus credit amount and credit amount versus age scatterplots show how the customers' different clusters were placed.

```
## Table showing the mean of different attributes in each cluster

credit_analysis$Cluster <- as.numeric(hierach_ward$cluster)

credit_analysis %>%
  group_by(Cluster) %>%
  summarise_all(.funs = "mean") %>%
  round(.,digits = 0) %>%
  knitr::kable("pipe")
```

Cluster	Age	Job	Credit.amount	Duration
1	46	2	3056	18
2	34	2	7086	40
3	37	1	2016	15
4	29	2	2199	16

We finally performed the average on the age, job, credit amount, and duration in the four clusters. The above analysis, therefore, shows that:

- Cluster 1 – Older customers, skilled workers, low mean credit amount, shorter duration.
- Cluster 2 – Younger customers, skilled workers, high mean credit amount, longer duration.
- Cluster 3 – Middle-aged customers, unskilled and resident workers, low mean credit amount, shorter duration.
- Cluster 4 – Younger customers, skilled workers, low mean credit amount, shorter duration.

4. CONCLUSION

Customer segmentation is an important process that helps companies identify different types of customers in the market. This process has helped many firms in their decision-making strategies by identifying which category of customers to advertise their products to, thus increasing their revenues. We have seen how different algorithms can help us predict gender based on the given attributes from the analysis. The hierarchical clustering method used in the analysis shows the different types of customer clusters the bank lends its loans. Hierarchical clustering was the only unsupervised learning algorithm that we applied on the dataset, which might not be optimal in this situation. For this cause, other clustering methods such as the fuzzy K-means clustering should be tested and compared to the hierarchical implementation.

5. REFERENCES

Ziafat, Hasan, and Majid Shakeri. “Using data mining techniques in customer segmentation.” Journal of Engineering Research and Applications 4.9 (2014): 70-79.

Kotsiantis, Sotiris B., I. Zaharakis, and P. Pintelas. “Supervised machine learning: A review of classification techniques.” Emerging artificial intelligence applications in computer engineering 160.1 (2007): 3-24.

Syakur, M. A., et al. “Integration k-means clustering method and elbow method for identification of the best customer profile cluster.” IOP Conference Series: Materials Science and Engineering. Vol. 336. No. 1. IOP Publishing, 2018.

Murtagh, Fionn, and Pierre Legendre. “Ward’s hierarchical agglomerative clustering method: which algorithms implement Ward’s criterion?.” Journal of classification 31.3 (2014): 274-295.