Data Analysis - Logistics Dataset - Classification

Amanpreet Kaur

December 09, 2022

PART A

1. Preliminary Data Preparation

```r
#Reading the data set and modifying the variable names with initials
Logistics_Dataset <- read.table("Data Analysis - Logistics Dataset.txt",
                                sep=",",
                                header = TRUE)

Logistics_Dataset <- as.data.frame(Logistics_Dataset)
colnames(Logistics_Dataset) <- paste(colnames(Logistics_Dataset),
                                     "AK",
                                     sep = "_")
head(Logistics_Dataset)
```

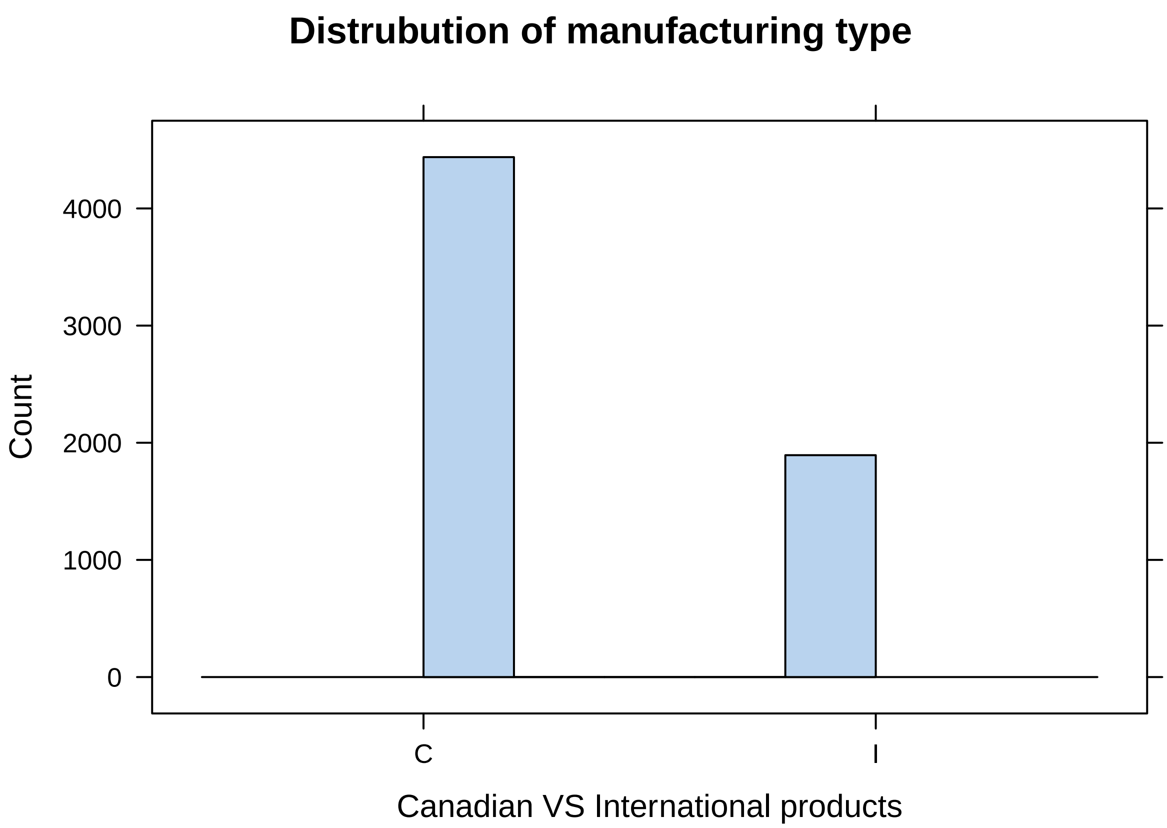
```
##   Del_AK Vin_AK Pkg_AK Cst_AK Mil_AK Dom_AK Haz_AK           Car_AK
## 1    9.5      6      6     13   1447      C      H M-Press Delivery
## 2   11.9     18      7      7   1874      I      N         Fed Post
## 3   14.6      7      7      8   1865      I      N         Fed Post
## 4   17.5     11      5     16   3111      I      H M-Press Delivery
## 5   10.7     12      4     10   1319      C      H         Fed Post
## 6   10.5     12      3      5   1415      C      N M-Press Delivery
```

```r
# Statistics for all the variables
stat.desc(Logistics_Dataset)
```

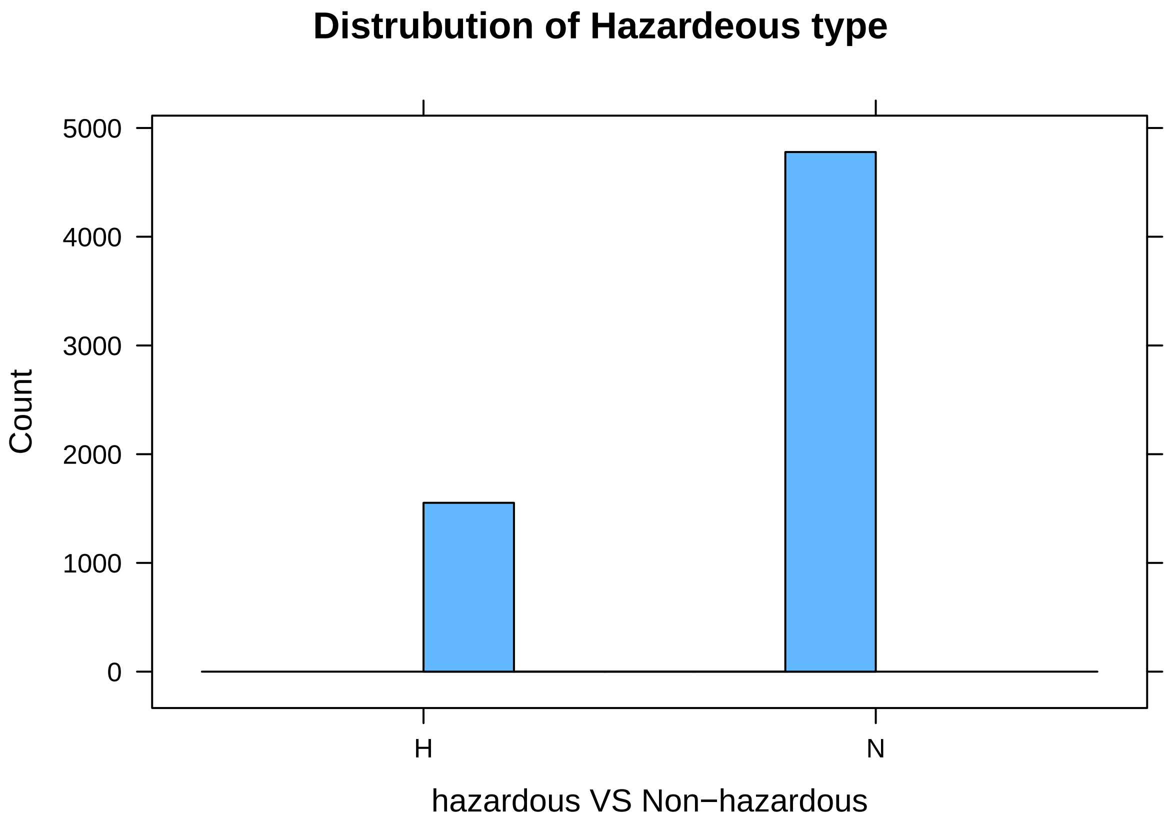
```
##                    Del_AK       Vin_AK       Pkg_AK       Cst_AK        Mil_AK
## nbr.val      6.332000e+03 6.332000e+03 6.332000e+03 6.332000e+03  6.332000e+03
## nbr.null     0.000000e+00 0.000000e+00 0.000000e+00 0.000000e+00  0.000000e+00
## nbr.na       0.000000e+00 0.000000e+00 0.000000e+00 0.000000e+00  0.000000e+00
## min          1.000000e-01 2.000000e+00 1.000000e+00 1.000000e+00 -6.200000e+01
## max          2.260000e+01 2.800000e+01 1.500000e+01 2.100000e+01  3.608000e+03
## range        2.250000e+01 2.600000e+01 1.400000e+01 2.000000e+01  3.670000e+03
## sum          6.688820e+04 8.249200e+04 2.530500e+04 5.668100e+04  1.034416e+07
## median       1.060000e+01 1.300000e+01 4.000000e+00 9.000000e+00  1.630000e+03
## mean         1.056352e+01 1.302780e+01 3.996368e+00 8.951516e+00  1.633633e+03
## SE.mean      3.905475e-02 4.507398e-02 2.472609e-02 3.734352e-02  6.348855e+00
## CI.mean.0.95 7.656054e-02 8.836027e-02 4.847151e-02 7.320595e-02  1.244591e+01
## var          9.658031e+00 1.286449e+01 3.871255e+00 8.830219e+00  2.552300e+05
```

```
## std.dev      3.107737e+00 3.586711e+00 1.967551e+00 2.971568e+00  5.052030e+02
## coef.var     2.941953e-01 2.753122e-01 4.923347e-01 3.319626e-01  3.092513e-01
##              Dom_AK Haz_AK Car_AK
## nbr.val          NA     NA     NA
## nbr.null         NA     NA     NA
## nbr.na           NA     NA     NA
## min              NA     NA     NA
## max              NA     NA     NA
## range            NA     NA     NA
## sum              NA     NA     NA
## median           NA     NA     NA
## mean             NA     NA     NA
## SE.mean          NA     NA     NA
## CI.mean.0.95     NA     NA     NA
## var              NA     NA     NA
## std.dev          NA     NA     NA
## coef.var         NA     NA     NA
```

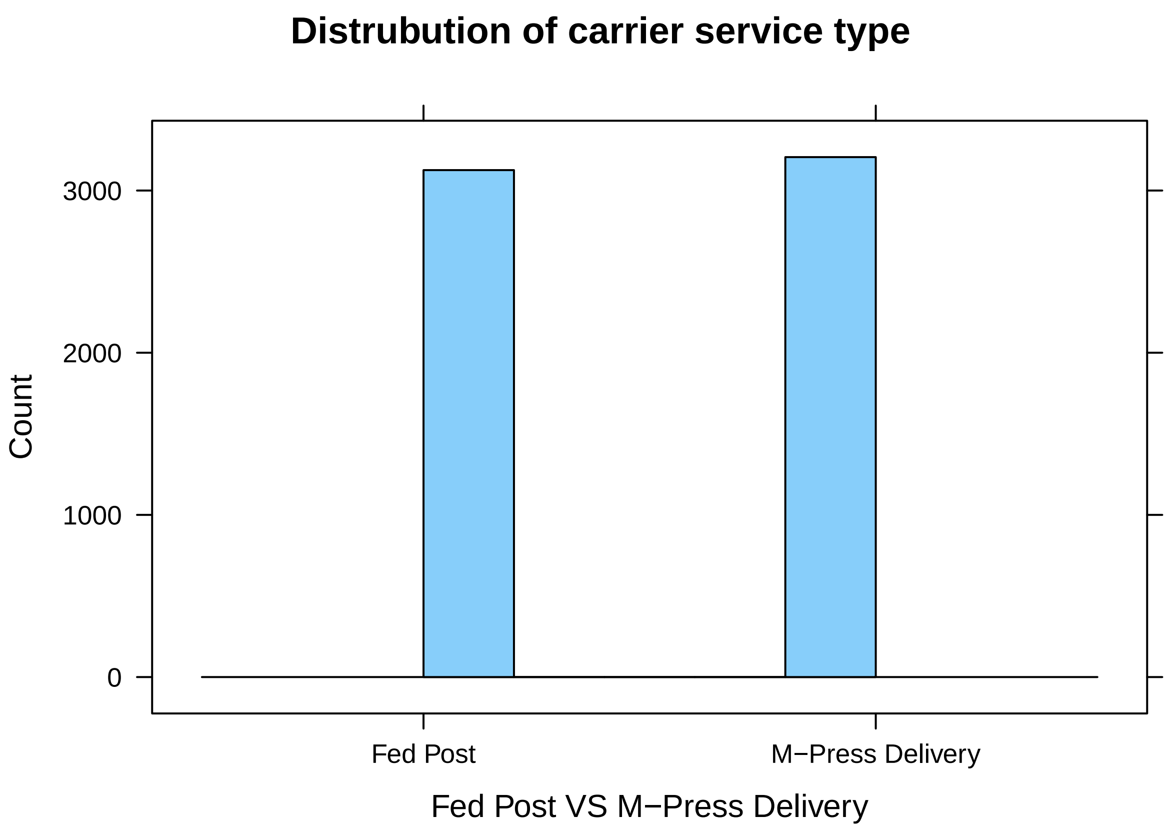
```r
#Converting the categorical variables to factor variables
Logistics_Dataset$Dom_AK <- as.factor(Logistics_Dataset$Dom_AK)
Logistics_Dataset$Haz_AK <- as.factor(Logistics_Dataset$Haz_AK)
Logistics_Dataset$Car_AK <- as.factor(Logistics_Dataset$Car_AK)

histogram( ~ Dom_AK,
          dat = Logistics_Dataset,
          breaks=4,
          col="slategray2",
          type="count",
          main="Distrubution of manufacturing type",
          xlab = "Canadian VS International products")
```



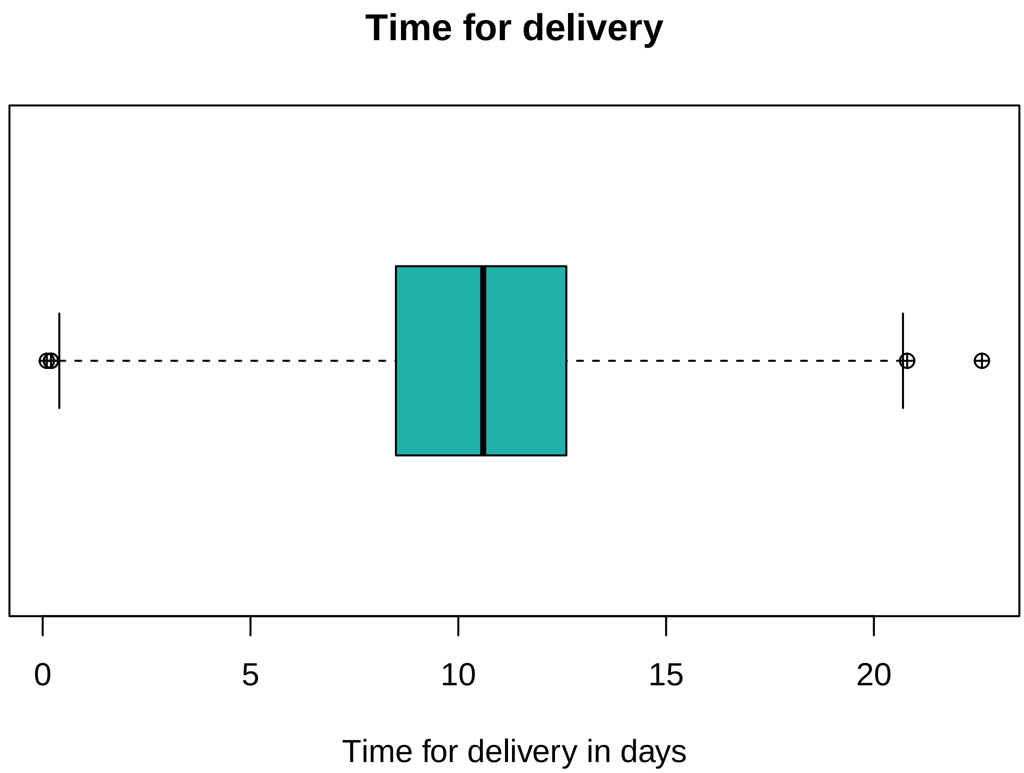

```
histogram( ~ Haz_AK,
          dat = Logistics_Dataset,
          breaks=4,
          col="steelblue1",
          type="count",
          main="Distrubution of Hazardeous type",
          xlab = "hazardous VS Non-hazardous")
```



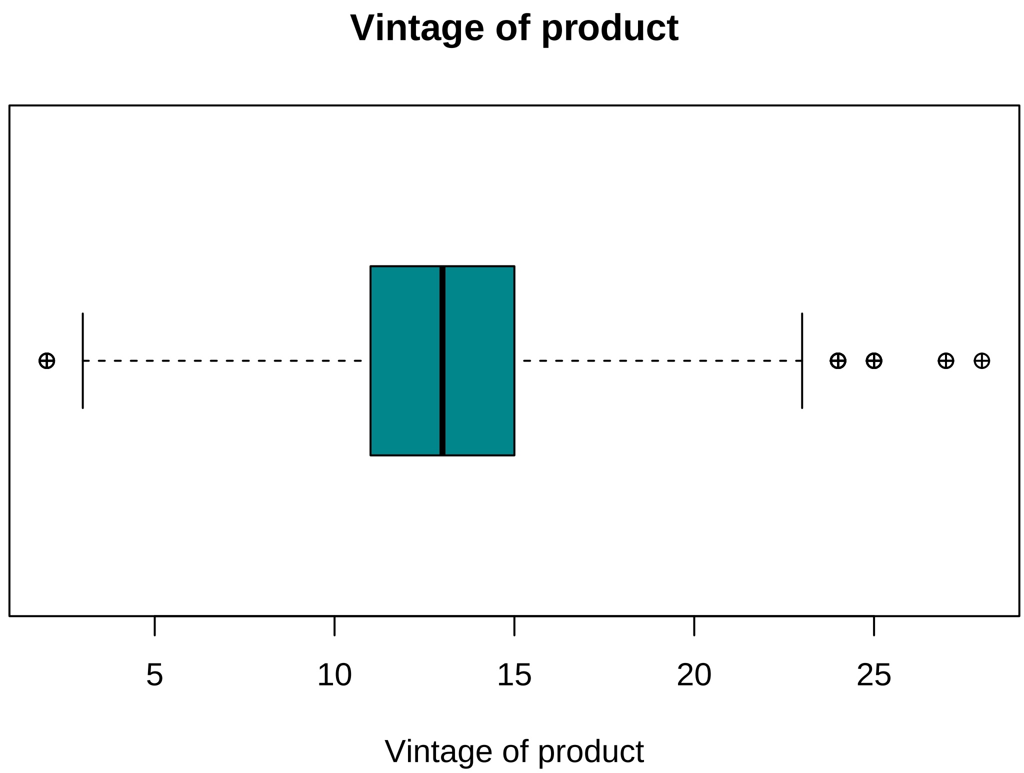

```r
histogram( ~ Car_AK,
          dat = Logistics_Dataset,
          breaks=4,
          col="lightskyblue",
          type="count",
          main="Distrubution of carrier service type",
          xlab = "Fed Post VS M-Press Delivery")
```



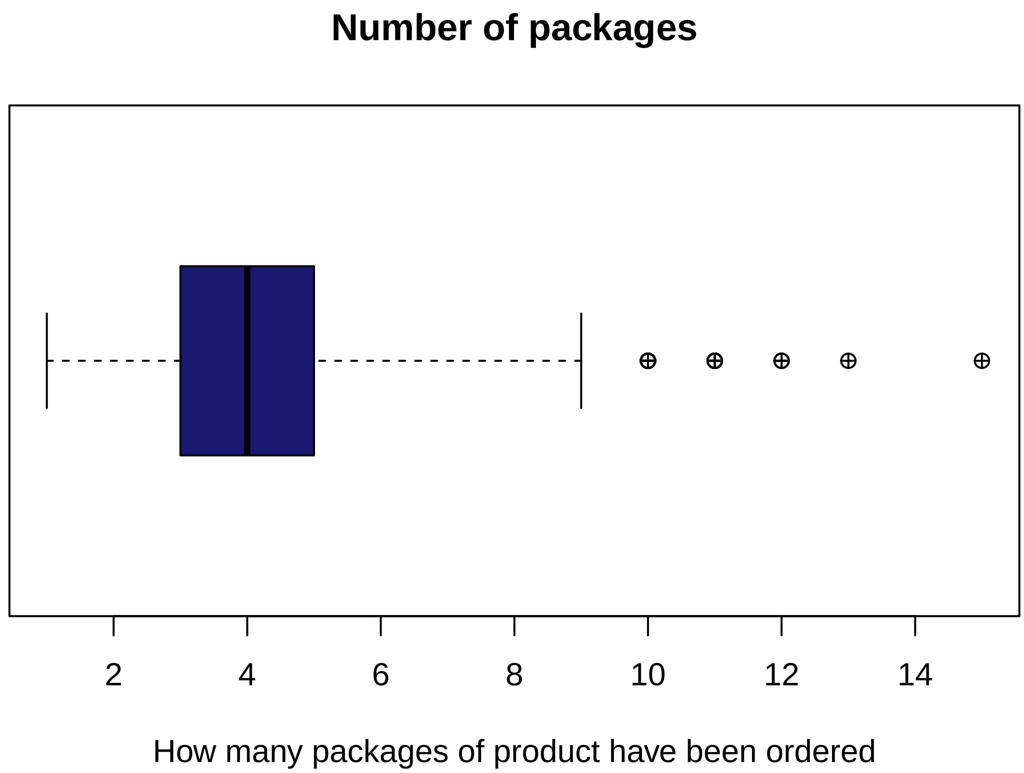

```
boxplot(Logistics_Dataset$Del_AK,
        main="Time for delivery",
        xlab="Time for delivery in days",
        col = "lightseagreen",
        border = "black",
        horizontal = TRUE,
        pch=10,
        range =2)
```



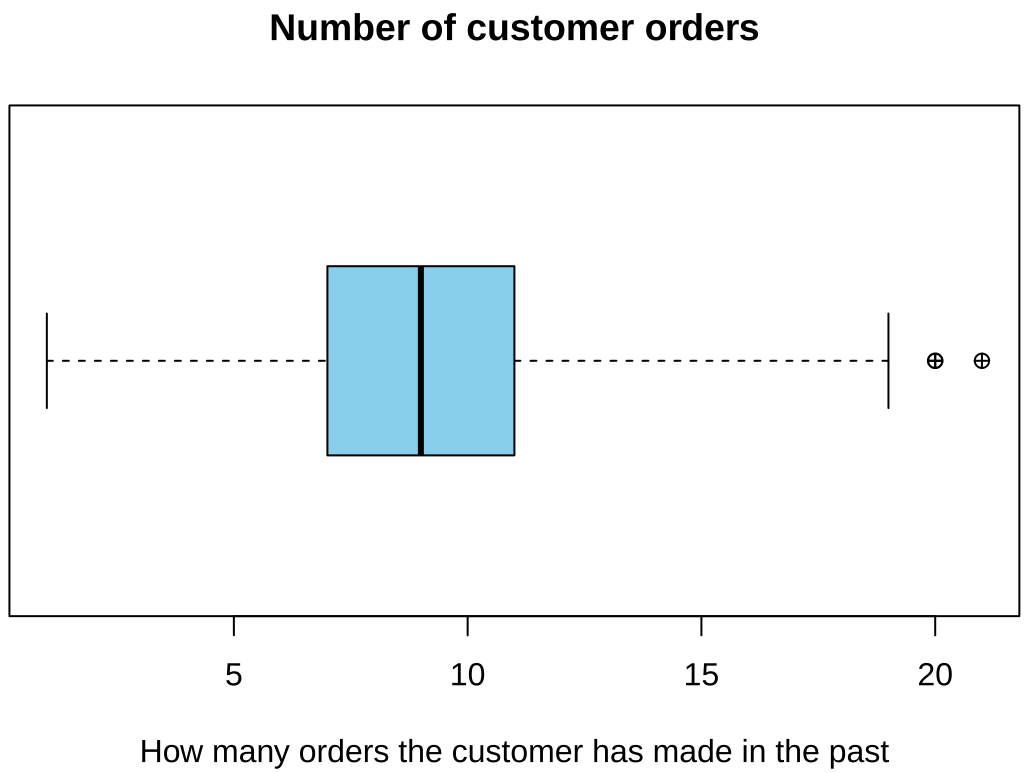

```r
boxplot(Logistics_Dataset$Vin_AK,
        main="Vintage of product",
        xlab="Vintage of product",
        col = "turquoise4",
        border = "black",
        horizontal = TRUE,
        pch=10,
        range =2)
```



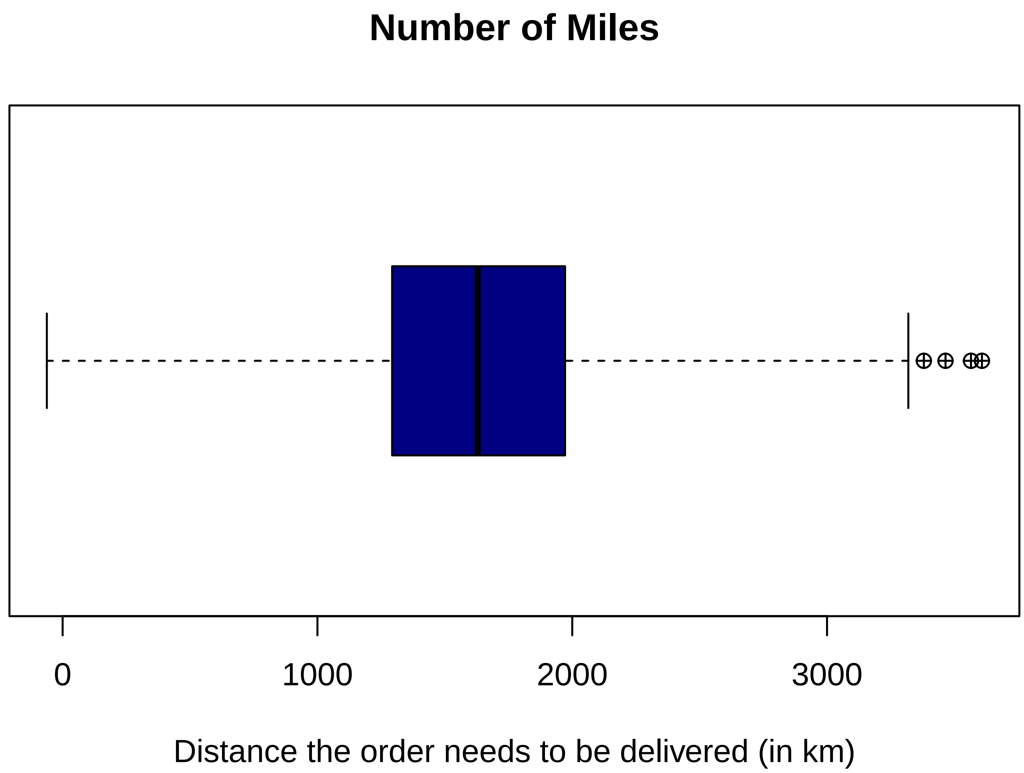

```
boxplot(Logistics_Dataset$Pkg_AK,
        main="Number of packages",
        xlab="How many packages of product have been ordered",
        col = "midnightblue",
        border = "black",
        horizontal = TRUE,
        pch=10,
        range =2)
```



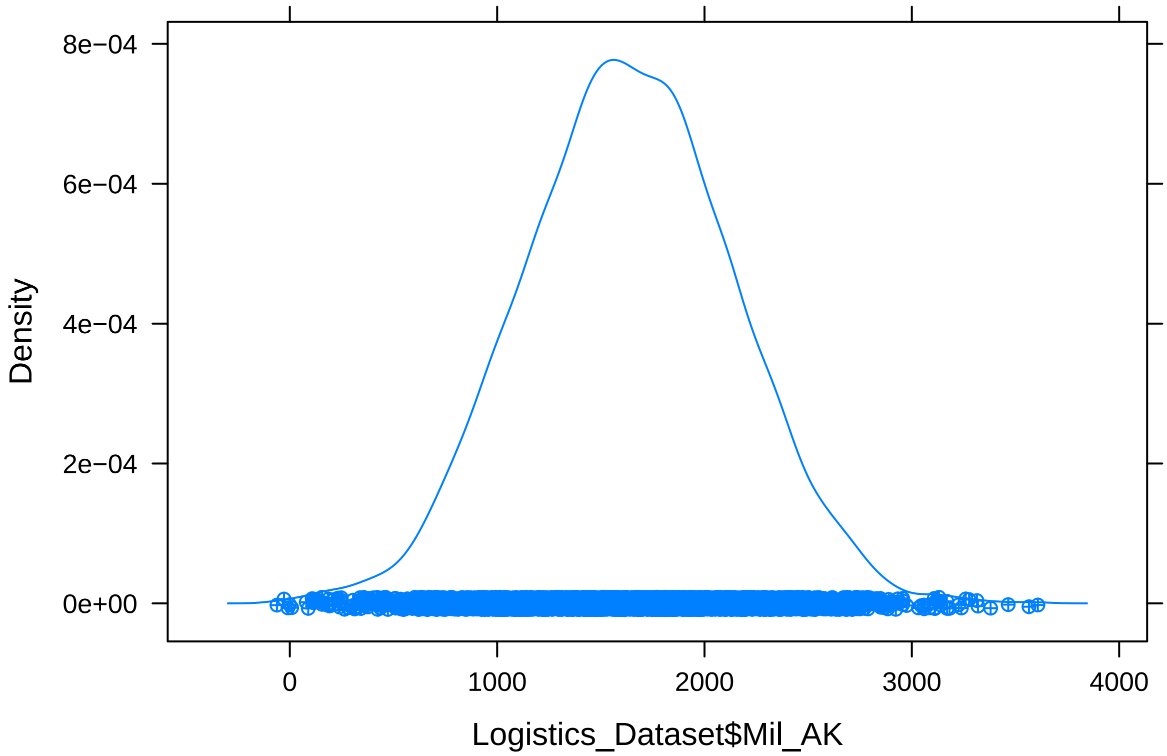

```
boxplot(Logistics_Dataset$Cst_AK,
        main="Number of customer orders",
        xlab="How many orders the customer has made in the past",
        col = "skyblue",
        border = "black",
        horizontal = TRUE,
        pch=10,
        range =2)
```




```r
boxplot(Logistics_Dataset$Mil_AK,
        main="Number of Miles",
        xlab="Distance the order needs to be delivered (in km)",
        col = "navy",
        border = "black",
        horizontal = TRUE,
        pch=10,
        range =2)
```




```
densityplot(Logistics_Dataset$Mil_AK, pch = 10)
```




```
#removing data points with Distance the order needs to be delivered (in km) is negative

Logistics_Dataset <-  subset(Logistics_Dataset , Mil_AK >= 0)
##################################################################################
#
# 1. Dom : The domestic or international indicator for the product have two values
#          and is a categorical data free from any outliers.
#
# 2. Haz : The indicator representing if product is hazardous or not also have
#          two categories and is free from any outliers.
#
# 3. Car : The indicator representing carrier service of the product have
#          two categories and is free from any outliers.
#
# 4. Del : The delivery time has one outlier but it does not have high influence
#          as the value seems high but normal for the dataset.
#
# 5. Vin : As per the box plot, there vintage time has 5 outliers which seems to
#          normal as there is no unusual value for the variable.
#
# 6. Pkg : The number of packages has 5 outliers with no high influence these outliers
#          are normal.
#
# 7. Cst : As per the box plot, the number of orders the customer has made in the past
#          has two outliers with no unusual values.
#
```

```
# 8. Mil : As per the box plot, there are 4 outliers which seems to be normal,
#          as the distance the orders needs to be delivered can be high than the
#          regular data. However there seems to be few values less than 0.
#          The distance can not be negative.
#
#          As per the density plot, there are approximately 4 data points which
#          are below 0.
#          Hence, removing these records.
#
#
###############################################################################
```

2. Exploratory Analysis

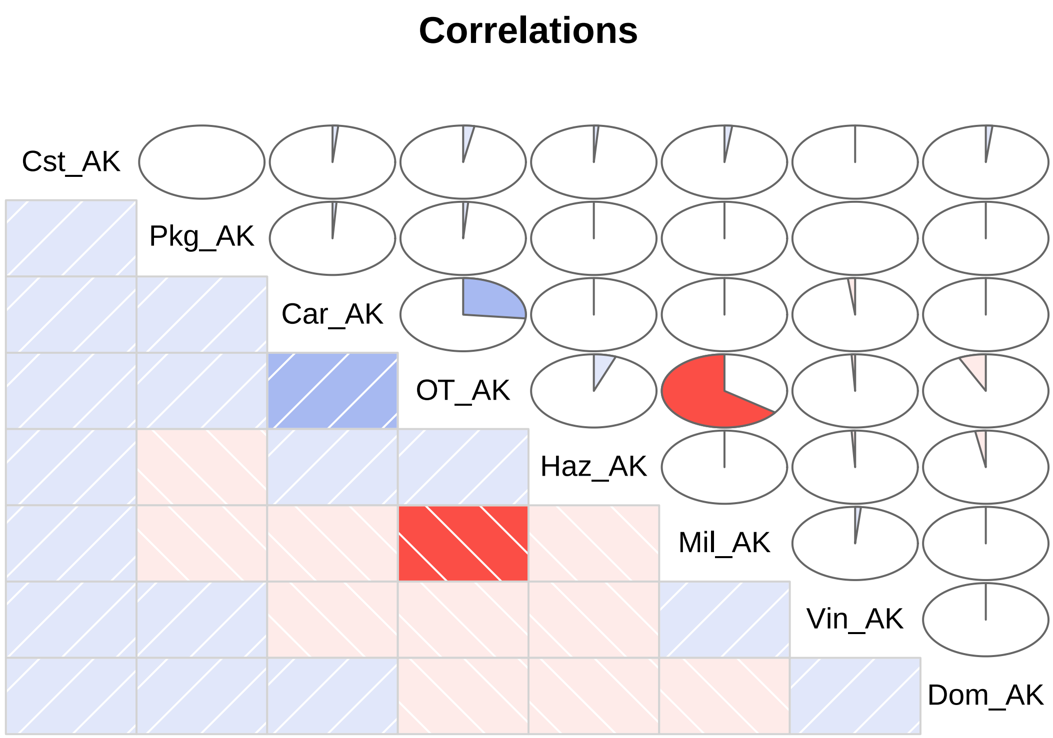
```
Logistics_Dataset$OT_AK <- as.numeric( as.factor( ifelse(Logistics_Dataset$Del_AK < 10.1, 1,0)))
Logistics_Dataset$Dom_AK <- as.numeric(Logistics_Dataset$Dom_AK)
Logistics_Dataset$Haz_AK <- as.numeric(Logistics_Dataset$Haz_AK)
Logistics_Dataset$Car_AK <- as.numeric(Logistics_Dataset$Car_AK)


# Removing the delivery column before checking the correlation with in the variables
# as OT_AK column is computed based on the delivery
Logistics_Dataset <- Logistics_Dataset[,-c(1)]
str(Logistics_Dataset)
```

```
## 'data.frame':    6328 obs. of  8 variables:
##  $ Vin_AK: int  6 18 7 11 12 12 21 12 13 16 ...
##  $ Pkg_AK: int  6 7 7 5 4 3 1 4 6 5 ...
##  $ Cst_AK: int  13 7 8 16 10 5 10 12 8 10 ...
##  $ Mil_AK: int  1447 1874 1865 3111 1319 1415 1599 2361 1394 1121 ...
##  $ Dom_AK: num  1 2 2 2 1 1 1 1 2 2 ...
##  $ Haz_AK: num  1 2 2 1 1 2 1 2 2 1 ...
##  $ Car_AK: num  2 1 1 2 1 2 2 2 1 2 ...
##  $ OT_AK : num  2 1 1 1 1 1 1 1 2 2 ...
```

```
#numerical correlation matrix
round(cor(Logistics_Dataset, method="spearman"),2)
```

```
##        Vin_AK Pkg_AK Cst_AK Mil_AK Dom_AK Haz_AK Car_AK OT_AK
## Vin_AK   1.00   0.00   0.00   0.02   0.00  -0.01  -0.02 -0.01
## Pkg_AK   0.00   1.00   0.00  -0.01   0.01  -0.01   0.01  0.01
## Cst_AK   0.00   0.00   1.00   0.01   0.02   0.01   0.02  0.03
## Mil_AK   0.02  -0.01   0.01   1.00   0.00   0.00  -0.01 -0.68
## Dom_AK   0.00   0.01   0.02   0.00   1.00  -0.03   0.01 -0.07
## Haz_AK  -0.01  -0.01   0.01   0.00  -0.03   1.00   0.01  0.06
## Car_AK  -0.02   0.01   0.02  -0.01   0.01   0.01   1.00  0.27
## OT_AK   -0.01   0.01   0.03  -0.68  -0.07   0.06   0.27  1.00
```

```
#graphical correlation matrix
corrgram(Logistics_Dataset, order=TRUE, lower.panel=panel.shade,
         upper.panel=panel.pie, text.panel=panel.txt,
         main="Correlations")
```




```
chisq_AK <- chisq.test(Logistics_Dataset$OT_AK, Logistics_Dataset$Car_AK, correct=FALSE)
chisq_AK
```

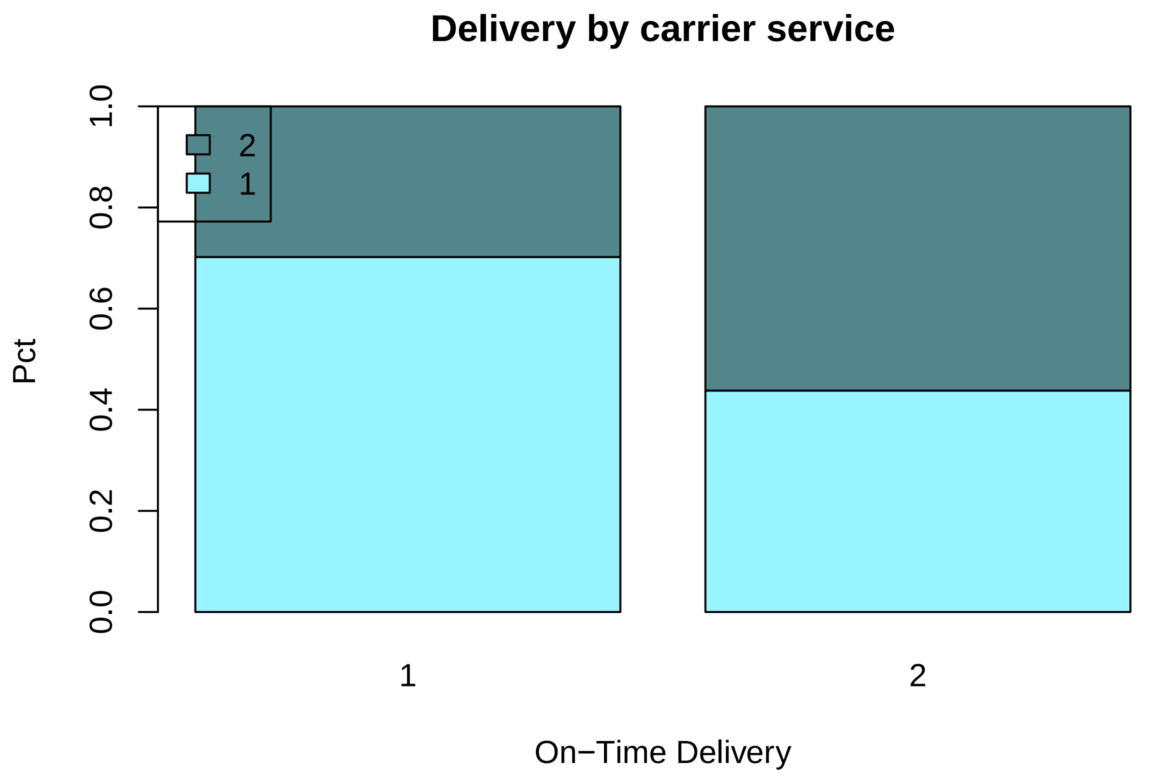
```
##
##  Pearson's Chi-squared test
##
## data:  Logistics_Dataset$OT_AK and Logistics_Dataset$Car_AK
## X-squared = 449.66, df = 1, p-value < 2.2e-16
```

```
table_OT_Car <- table(Logistics_Dataset$OT_AK, Logistics_Dataset$Car_AK,
                   dnn=list("On-Time delivery","Carrier Services"))
table_OT_Car
```

```
##                   Carrier Services
## On-Time delivery    1    2
##                  1 2193 1403
##                  2  931 1801
```

```
#Vertical Bar Chart
barplot(prop.table(table_OT_Car,2),
                   xlab='On-Time Delivery',
                   ylab='Pct',
                   main="Delivery by carrier service",
                   col=c("cadetblue1","cadetblue4"),
```

```
                    legend=rownames(table_OT_Car),
                    args.legend = list(x = "topleft"))
```




```
##############################################################################
# 1. Numerical Correlation:
#
#    The mod of correlation between On-Time Delivery and Mil i.e. distance in
#    kms is approximately 0.68 which represents that there is moderate linear
#    correlation between these two variables.
#    Also, there is a weak linear relation between On-Time Delivery and Carrier
#    Services with correlation of 0.27.
#
#    The delivery time variable is removed from the data set to avoid co-linear
#    variables in the data set, as the new variable i.e. On-Time Delivery is
#    derived from the delivery variable.
#
#    We can also depict the same about the variables mentioned above from the
#    graphical representation of the correlation matrix.
#
# 2. Identifying the most significant predictor for On-Time Delivery:
#
#    We have performed Chi-Squared test to check if there is any relationship
#    between the Carrier services and On-time delivery as both are categorical
#    variables.
#    After observing the p-value (p-value < 2.2e-16) we can say that there is
```

```
#    statistical evidence that there is a relationship between both the
#    variables.
#
################################################################################
```

3. Model Development

```
Logistics_Dataset$OT_AK <-  as.factor(Logistics_Dataset$OT_AK)
Logistics_Dataset$Dom_AK <- as.factor(Logistics_Dataset$Dom_AK)
Logistics_Dataset$Haz_AK <- as.factor(Logistics_Dataset$Haz_AK)
Logistics_Dataset$Car_AK <- as.factor(Logistics_Dataset$Car_AK)

str(Logistics_Dataset)
```

```
## 'data.frame':    6328 obs. of  8 variables:
##  $ Vin_AK: int  6 18 7 11 12 12 21 12 13 16 ...
##  $ Pkg_AK: int  6 7 7 5 4 3 1 4 6 5 ...
##  $ Cst_AK: int  13 7 8 16 10 5 10 12 8 10 ...
##  $ Mil_AK: int  1447 1874 1865 3111 1319 1415 1599 2361 1394 1121 ...
##  $ Dom_AK: Factor w/ 2 levels "1","2": 1 2 2 2 1 1 1 1 2 2 ...
##  $ Haz_AK: Factor w/ 2 levels "1","2": 1 2 2 1 1 2 1 2 2 1 ...
##  $ Car_AK: Factor w/ 2 levels "1","2": 2 1 1 2 1 2 2 2 1 2 ...
##  $ OT_AK : Factor w/ 2 levels "1","2": 2 1 1 1 1 1 1 1 2 2 ...
```

```
#full model
glm.fit <- glm(OT_AK ~ . , data=Logistics_Dataset, family='binomial')
summary(glm.fit)
```

```
##
## Call:
## glm(formula = OT_AK ~ ., family = "binomial", data = Logistics_Dataset)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -3.0579  -0.4647  -0.0800   0.4314   3.3751
##
## Coefficients:
##                 Estimate Std. Error z value Pr(>|z|)
## (Intercept)  7.1141551  0.2991127  23.784  < 2e-16 ***
## Vin_AK       0.0190389  0.0111076   1.714   0.0865 .
## Pkg_AK       0.0231762  0.0201096   1.152   0.2491
## Cst_AK       0.0558557  0.0132618   4.212 2.53e-05 ***
## Mil_AK      -0.0061375  0.0001591 -38.586  < 2e-16 ***
## Dom_AK2     -0.7614948  0.0880635  -8.647  < 2e-16 ***
## Haz_AK2      0.5528396  0.0924725   5.978 2.25e-09 ***
## Car_AK2      2.4106820  0.0921437  26.162  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
```

```
##     Null deviance: 8654.1  on 6327  degrees of freedom
## Residual deviance: 4105.9  on 6320  degrees of freedom
## AIC: 4121.9
##
## Number of Fisher Scoring iterations: 6
```

```r
#backward model
step.fit <- step(glm.fit,direction = "backward", trace = 0)
summary(step.fit)
```

```
##
## Call:
## glm(formula = OT_AK ~ Vin_AK + Cst_AK + Mil_AK + Dom_AK + Haz_AK +
##     Car_AK, family = "binomial", data = Logistics_Dataset)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -3.0412  -0.4669  -0.0807   0.4316   3.3941
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept)  7.2027284  0.2896389  24.868  < 2e-16 ***
## Vin_AK       0.0189733  0.0111066   1.708   0.0876 .
## Cst_AK       0.0555671  0.0132600   4.191 2.78e-05 ***
## Mil_AK      -0.0061327  0.0001589 -38.607  < 2e-16 ***
## Dom_AK2     -0.7605857  0.0880322  -8.640  < 2e-16 ***
## Haz_AK2      0.5526367  0.0924502   5.978 2.26e-09 ***
## Car_AK2      2.4098809  0.0921050  26.165  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 8654.1  on 6327  degrees of freedom
## Residual deviance: 4107.3  on 6321  degrees of freedom
## AIC: 4121.3
##
## Number of Fisher Scoring iterations: 6
```

```r
###############################################################################
# Interpretation:
#
# (1) AIC:
#
#     AIC for Full model is 4121.9 and AIC for backward model is 4121.3,
#     which means there is no significant difference based on the AIC.
#     However as we consider lower AIC value as better, therefore backward
#     is better.
#
# (2) Deviance:
#
#     The difference between null and residual deviance is 4555 for
#     full model and 4553.6 for backward model. As the difference is
```

```
#     more for the full model, full model is better.
#
# (3) Residual symmetry:
#
#     The residuals for both the models seems quite symmetrical
#     Therefore, both models are good in this case.
#
# (4) z-values:
#
#     For the full model, two variables are not statistically significant
#     i.e. , vintage and number of packages.
#     The other variable and intercept is statistically significant as
#     the p-value is less than 0.05.
#
#     For the backward model, there is one variable which is not
#     statistically significant and other variables are as their p-value
#     is less than 0.05.
#
#     After comparing both, backward model has less number of variables
#     and less number of variable which are not statistically significant
#     therefore, backward model is better in this case.
#
# (5) Parameter Co-Efficient:
#
#     The parameter coefficients for both the models are quite same.
#
#
#  Conclusion:
#
#     Overall, the backward model is slightly better than the full model,
#     as there are less number of variable and better based on the main
#     measures interpreted above.
#
###############################################################################
```

PART B

1. Logistic Regression – Backward

```
#Logistic Regression - Backward
starttime <- Sys.time()
step.fit_LR_AK <- step(glm.fit, direction = "backward", trace = 0)
endtime <- Sys.time()
timetaken_M1 <- endtime - starttime
summary(step.fit_LR_AK)
```

```
##
## Call:
## glm(formula = OT_AK ~ Vin_AK + Cst_AK + Mil_AK + Dom_AK + Haz_AK +
##     Car_AK, family = "binomial", data = Logistics_Dataset)
##
```

```
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -3.0412  -0.4669  -0.0807   0.4316   3.3941
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept)  7.2027284  0.2896389  24.868  < 2e-16 ***
## Vin_AK       0.0189733  0.0111066   1.708   0.0876 .
## Cst_AK       0.0555671  0.0132600   4.191 2.78e-05 ***
## Mil_AK      -0.0061327  0.0001589 -38.607  < 2e-16 ***
## Dom_AK2     -0.7605857  0.0880322  -8.640  < 2e-16 ***
## Haz_AK2      0.5526367  0.0924502   5.978 2.26e-09 ***
## Car_AK2      2.4098809  0.0921050  26.165  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 8654.1  on 6327  degrees of freedom
## Residual deviance: 4107.3  on 6321  degrees of freedom
## AIC: 4121.3
##
## Number of Fisher Scoring iterations: 6
```

```
timetaken_M1
```

```
## Time difference of 0.4991231 secs
```

```
responseM1 <- predict(step.fit_LR_AK, type = "response")
head(responseM1,10)
```

```
##           1           2           3           4           5           6
## 0.828460490 0.022588333 0.020524075 0.000108448 0.474273517 0.880054146
##           7           8           9          10
## 0.681457589 0.031686859 0.296700056 0.944615218
```

```
classM1 <- ifelse(responseM1>0.5,2,1)
head(classM1)
```

```
## 1 2 3 4 5 6
## 2 1 1 1 1 2
```

```
CM1 <- table( Logistics_Dataset$OT_AK, classM1, dnn=list("Actual","Predicted"))
CM1
```

```
##       Predicted
## Actual    1    2
##      1 3164  432
##      2  480 2252
```

2. Naive-Bayes Classification

```
#Naive-Bayes Classification
starttime <- Sys.time()
NaiveBayes_AK <- NaiveBayes(OT_AK ~ . , data = Logistics_Dataset, na.action = na.omit)
endtime <- Sys.time()
timetaken_M2 <- endtime - starttime
summary(NaiveBayes_AK)
```

```
##           Length Class      Mode
## apriori   2      table      numeric
## tables    7      -none-     list
## levels    2      -none-     character
## call      3      -none-     call
## x         7      data.frame list
## usekernel 1      -none-     logical
## varnames  7      -none-     character
```

```
timetaken_M2
```

```
## Time difference of 0.008337021 secs
```

```
responseM2 <- predict( NaiveBayes_AK, Logistics_Dataset )
CM2 <- table( Logistics_Dataset$OT_AK, Predicted = responseM2$class, dnn=list("Actual","Predicted"))
CM2
```

```
##       Predicted
## Actual    1    2
##      1 3153  443
##      2  505 2227
```

3. Linear Discriminant Analysis

```
#Linear Discriminant Analysis
start_time <- Sys.time()
LDA_AK <- lda(OT_AK ~ . , data = Logistics_Dataset , na.action=na.omit)
end_time <- Sys.time()
timetaken_M3 <- end_time - start_time
summary(step.fit_LR_AK)
```

```
##
## Call:
## glm(formula = OT_AK ~ Vin_AK + Cst_AK + Mil_AK + Dom_AK + Haz_AK +
##     Car_AK, family = "binomial", data = Logistics_Dataset)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -3.0412  -0.4669  -0.0807   0.4316   3.3941
##
```

```
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept)  7.2027284  0.2896389  24.868  < 2e-16 ***
## Vin_AK       0.0189733  0.0111066   1.708   0.0876 .
## Cst_AK       0.0555671  0.0132600   4.191 2.78e-05 ***
## Mil_AK      -0.0061327  0.0001589 -38.607  < 2e-16 ***
## Dom_AK2     -0.7605857  0.0880322  -8.640  < 2e-16 ***
## Haz_AK2      0.5526367  0.0924502   5.978 2.26e-09 ***
## Car_AK2      2.4098809  0.0921050  26.165  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 8654.1  on 6327  degrees of freedom
## Residual deviance: 4107.3  on 6321  degrees of freedom
## AIC: 4121.3
##
## Number of Fisher Scoring iterations: 6
```

```
timetaken_M3
```

```
## Time difference of 0.01231909 secs
```

```
responseM3<- predict(LDA_AK,Logistics_Dataset)
CM3 <- table (Actual=Logistics_Dataset$OT_AK, Predicted=responseM3$class)
CM3
```

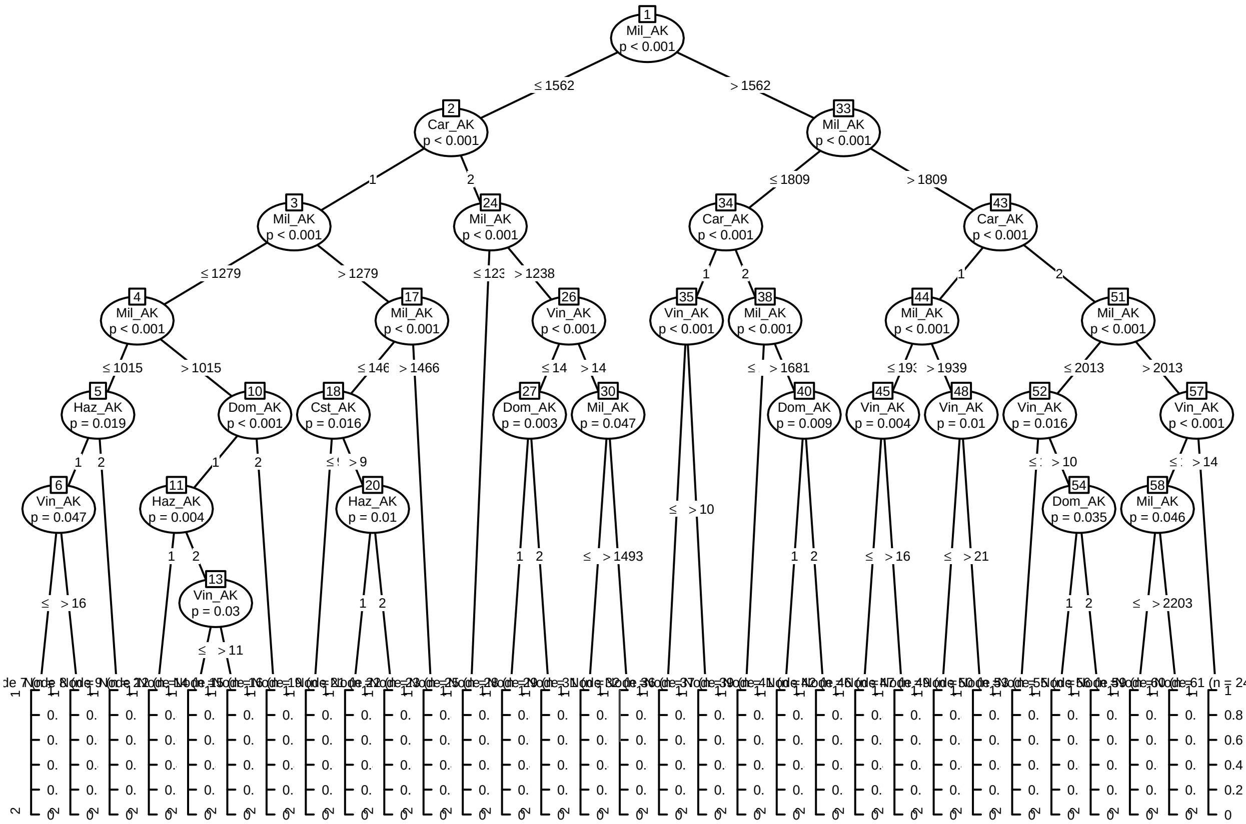
```
##       Predicted
## Actual    1    2
##      1 3156  440
##      2  469 2263
```

4. Decision Tree

```
start_time <- Sys.time()
tree.fit_AK <- ctree(OT_AK ~ . , data=Logistics_Dataset)
end_time <- Sys.time()

timetaken_M4 <- end_time - start_time
timetaken_M4
```

```
## Time difference of 0.139802 secs
```

```
plot(tree.fit_AK, gp=gpar(fontsize=5))
```



```
responseM4 <- predict(tree.fit_AK, Logistics_Dataset)
CM4 <- table(Actual=Logistics_Dataset$OT_AK, Predicted=responseM4)
CM4
```

```
##        Predicted
## Actual    1    2
##      1 3193  403
##      2  504 2228
```

5. Compare All Classifiers

```
# Calculating accuracy for Logistic Regression - Backward classifier
TP_M1<- CM1[2,2]
TN_M1<- CM1[1,1]
AccuracyM1 <- (TP_M1+TN_M1)/sum(CM1)

# Calculating accuracy for Naive-Bayes Classification classifier
TP_M2<- CM2[2,2]
TN_M2<- CM2[1,1]
AccuracyM2 <- (TP_M2+TN_M2)/sum(CM2)

# Calculating accuracy for Linear Discriminant Analysis classifier
TP_M3<- CM3[2,2]
TN_M3<- CM3[1,1]
```

```r
AccuracyM3 <- (TP_M3+TN_M3)/sum(CM3)

# Calculating accuracy for Decision Tree classifier
TP_M4<- CM4[2,2]
TN_M4<- CM4[1,1]
AccuracyM4 <- (TP_M4+TN_M4)/sum(CM4)

AccuracyM1
```

```
## [1] 0.8558786
```

```r
AccuracyM2
```

```
## [1] 0.8501896
```

```r
AccuracyM3
```

```
## [1] 0.8563527
```

```r
AccuracyM4
```

```
## [1] 0.8566688
```

```r
#Time taken for Logistic Regression - Backward classifier
timetaken_M1
```

```
## Time difference of 0.4991231 secs
```

```r
#Time taken for Naive-Bayes Classification classifier
timetaken_M2
```

```
## Time difference of 0.008337021 secs
```

```r
#Time taken for Linear Discriminant Analysis classifier
timetaken_M3
```

```
## Time difference of 0.01231909 secs
```

```r
#Time taken for Decision Tree classifier
timetaken_M4
```

```
## Time difference of 0.139802 secs
```

```r
# Extracting values for false postives for all classifiers to a variable
FP_M1<- CM1[1,2]
FP_M2<- CM2[1,2]
FP_M3<- CM3[1,2]
FP_M4<- CM4[1,2]

#False positives for Logistic Regression - Backward classifier
FP_M1
```

```
## [1] 432
```

```r
#False positives for Naive-Bayes Classification classifier
FP_M2
```

```
## [1] 443
```

```r
#False positives for Linear Discriminant Analysis classifier
FP_M3
```

```
## [1] 440
```

```r
#False positives for Decision Tree classifier
FP_M4
```

```
## [1] 403
```

```r
################################################################################
# Overall comparison of classifiers:
#
# 1. Accuracy:
#
#    The decision tree classifier has the highest accuracy as compared to
#    other models. The Linear Discriminant Analysis classifier has a slightly low
#    accuracy. Naive-Bayes Classification have the least
#    accuracy.
#
# 2. Processing Speed:
#
#    In case the processing speed is a priority, the Naive-Bayes
#    Classification is the best with least processing speed.
#
# 3. Minimize false positives:
#
#    To minimize false positive, the decision tree classifier have the least
#    false positives with value of 403.
#
# 4. Best model overall:
#
#    To conclude the best model overall, it is necessary to consider the main
#    requirements.
#
#    If the accuracy and minimizing false positives is our top
#    priority then decision tree classifier is the best. However, if we need
#    the classification to be fast and processing speed is our priority
#    then decision tree classifier is slower than Naive Bayes.
#    Else, if fast processing is the requirement then  Naive Bayer classifier
#    is fastest. Naive Bayes have the least accuracy and more number of false positives.
#
################################################################################
```