In [1]:
```
library(tidyverse)
library(lubridate)
library(ggplot2)
library(readxl)
library(openxlsx)
library(dplyr)
library(caTools)
library(pROC)
library(repr)
```

```
Registered S3 methods overwritten by 'ggplot2':
  method         from
  [.quosures     rlang
  c.quosures     rlang
  print.quosures rlang
Registered S3 method overwritten by 'rvest':
  method            from
  read_xml.response xml2
-- Attaching packages --------------------------------------- tidyverse 1.2.1 --
v ggplot2 3.1.1       v purrr   0.3.2
v tibble  2.1.1       v dplyr   0.8.0.1
v tidyr   0.8.3       v stringr 1.4.0
v readr   1.3.1       v forcats 0.4.0
-- Conflicts ------------------------------------------ tidyverse_conflicts() --
x dplyr::filter() masks stats::filter()
x dplyr::lag()    masks stats::lag()

Attaching package: 'lubridate'

The following object is masked from 'package:base':

    date

Warning message:
"package 'openxlsx' was built under R version 3.6.3"Warning message:
"package 'caTools' was built under R version 3.6.3"Warning message:
"package 'pROC' was built under R version 3.6.3"Type 'citation("pROC")' for a citatio
n.

Attaching package: 'pROC'

The following objects are masked from 'package:stats':

    cov, smooth, var
```

In [2]:
```
df = read_excel("Flight_Delay_time_difference.xlsx")
head(df,n=3)
```

schedtime	deptime	carrier	dest	distance	origin	weather	dayweek	daymonth	Difference in Minutes	delay
1899-12-31 14:55:00	1899-12-31 14:55:00	OH	JFK	184	BWI	0	4	1	0	ontime
1899-12-31 16:40:00	1899-12-31 16:40:00	DH	JFK	213	DCA	0	4	1	0	ontime
1899-12-31 12:45:00	1899-12-31 12:45:00	DH	LGA	229	IAD	0	4	1	0	ontime

In [3]:
```
#sapply(df, class)
```

CHECKING OUT NULL VALUES

In [4]:
```
sapply(df, function(x) sum(is.na(x)))
```

schedtime 0
deptime 0
carrier 0
dest 0
distance 0
origin 0
weather 0
dayweek 0
daymonth 0
Difference in Minutes 0
delay 0

In [5]:
```
colnames(df)
```

1. 'schedtime'
2. 'deptime'
3. 'carrier'
4. 'dest'
5. 'distance'
6. 'origin'
7. 'weather'
8. 'dayweek'
9. 'daymonth'
10. 'Difference in Minutes'
11. 'delay'

In [6]:
```
names(df)[10] <- "diff_minutes"
names(df)[1] <- "schedtime"
colnames(df)
```

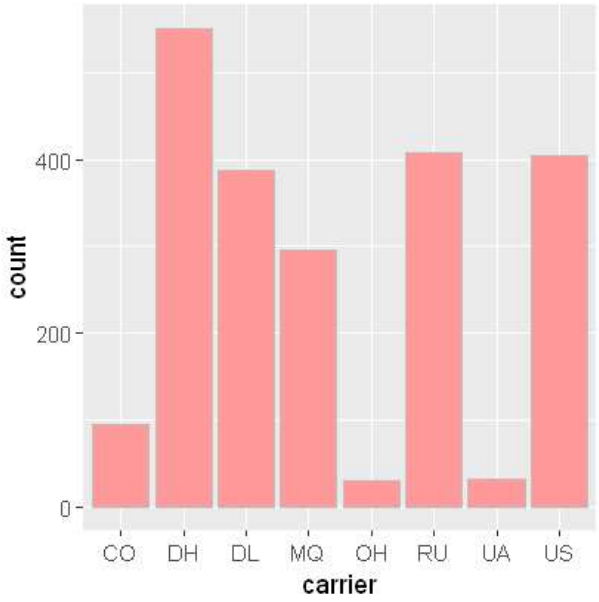
1. 'schedtime'
2. 'deptime'
3. 'carrier'
4. 'dest'
5. 'distance'
6. 'origin'
7. 'weather'
8. 'dayweek'
9. 'daymonth'
10. 'diff_minutes'
11. 'delay'

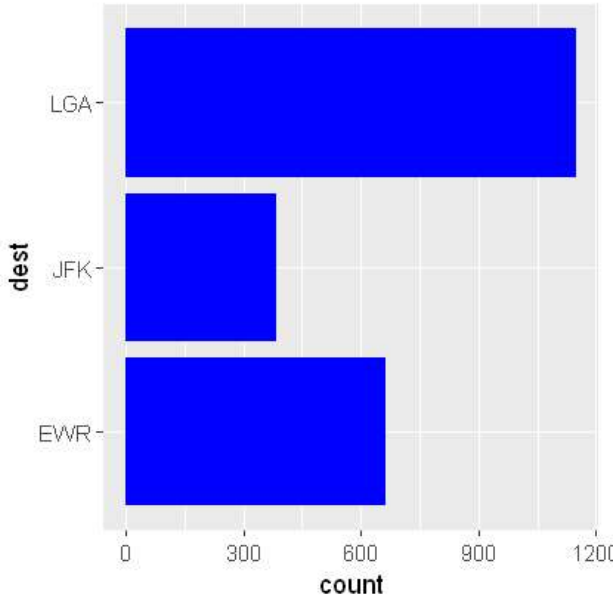
EDA-ANALYIS

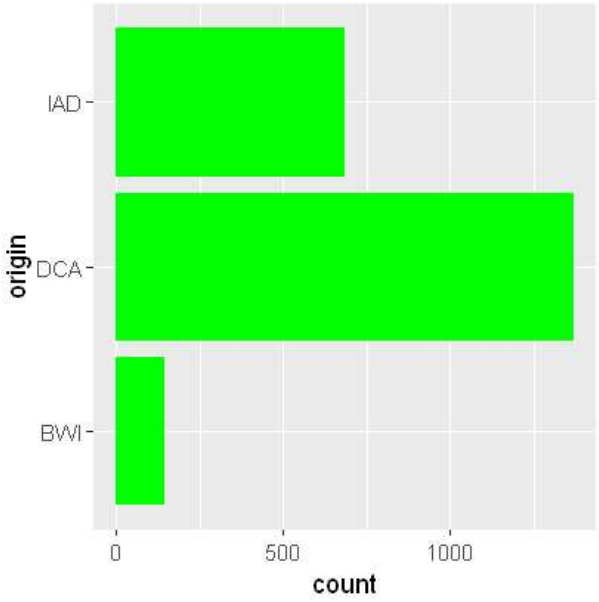
In [7]:
```
library(gridExtra)
```

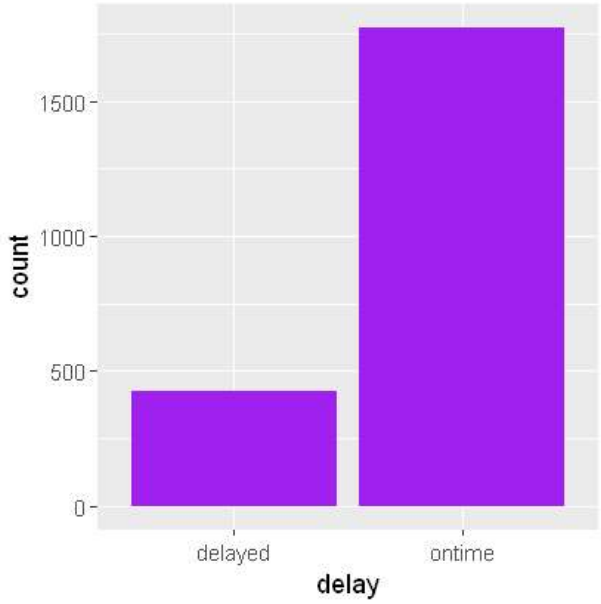
```
Warning message:
"package 'gridExtra' was built under R version 3.6.3"
Attaching package: 'gridExtra'

The following object is masked from 'package:dplyr':

    combine
```

In [8]:
```
par(mfrow = c(2, 2))
car1 = ggplot(data = df) +
  geom_bar(mapping = aes(x = carrier),fill="#FF9999", colour="grey")
des1 = ggplot(data = df) +
  geom_bar(mapping = aes(x = dest),fill = 'blue')
des1 = des1+ coord_flip()
ori1 = ggplot(data = df) +
  geom_bar(mapping = aes(x = origin),fill = 'green')
ori1 = ori1 + coord_flip()
del1 = ggplot(data = df) +
  geom_bar(mapping = aes(x = delay),fill = 'purple')
grid.arrange(car1,des1,ori1,del1,nrow=2 ,ncol=2)
```



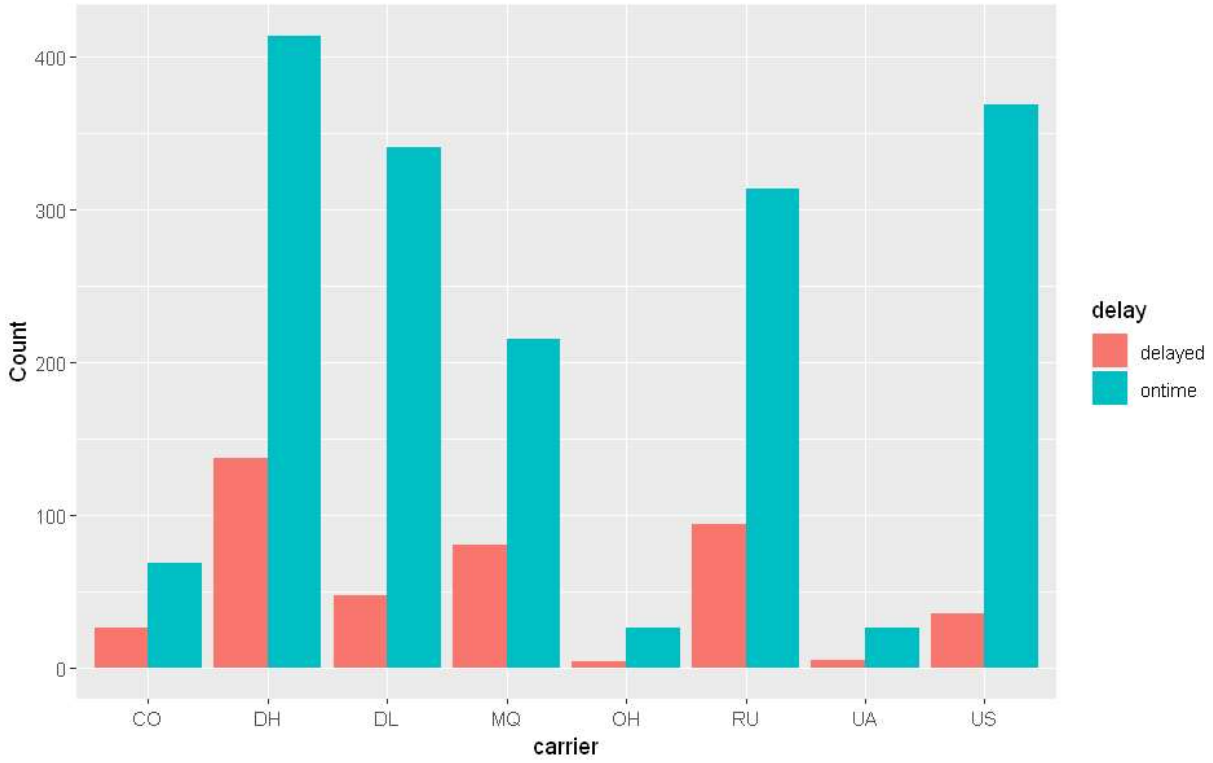






In [9]:
```
options(repr.plot.width=8, repr.plot.height=5)
```

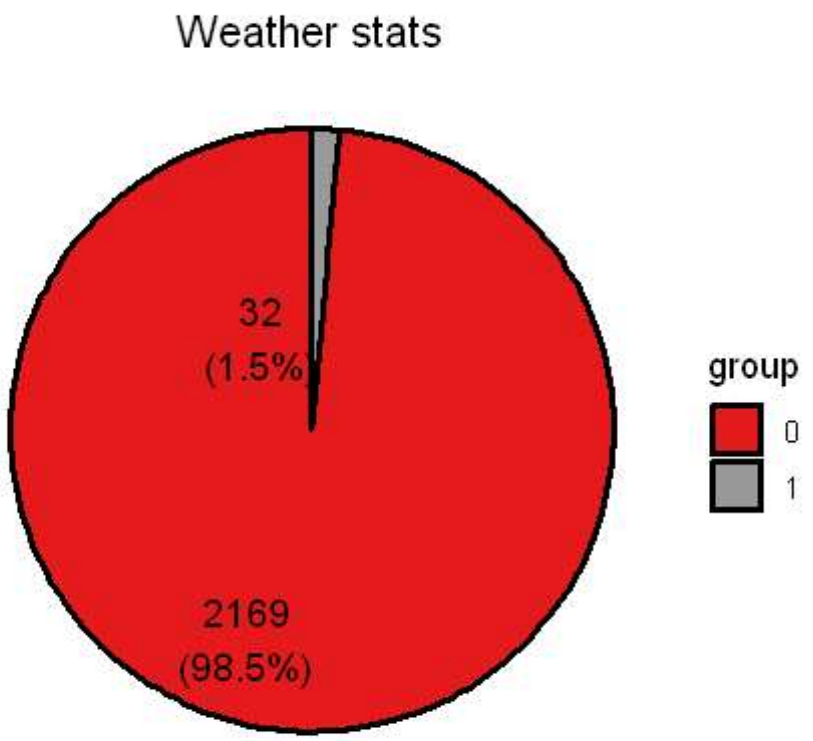
In [10]:
```
df %>%
  group_by(carrier, delay) %>%
  summarize(Count = n()) %>%
  ggplot(aes(x=carrier, y=Count, fill=delay)) +
  geom_bar(stat='identity', position= "dodge")
```



In [11]:
```
library(ggpie)
```

In [12]:
```
options(repr.plot.width=4, repr.plot.height=4)
```

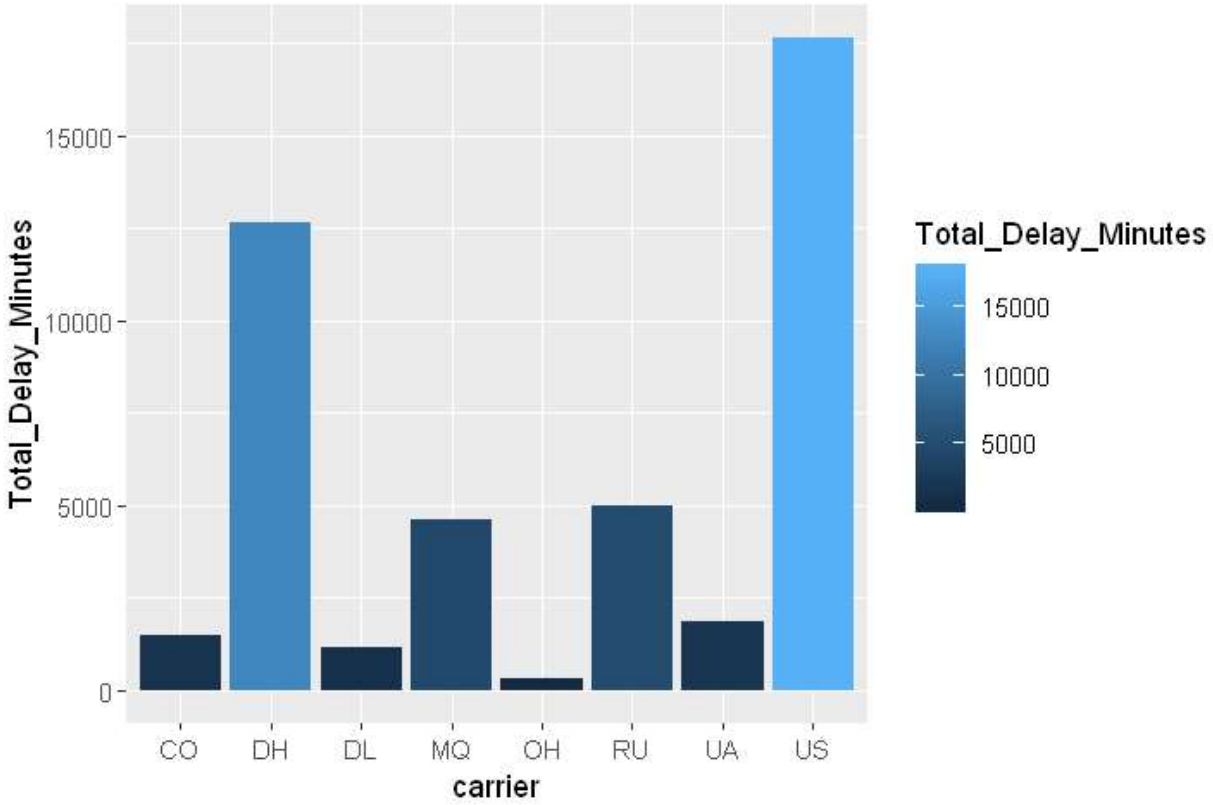
In [13]:
```
ggpie(data = df, group_key = "weather", count_type = "full",label_info = "all",label_t
  theme(plot.title = element_text(hjust = 0.5))
```



In [14]:
```
a = df %>%
  group_by(carrier) %>%
  summarise(Total_Delay_Minutes = sum(diff_minutes))
a
```

carrier	Total_Delay_Minutes
CO	1506
DH	12633
DL	1177
MQ	4601
OH	327
RU	4988
UA	1866
US	17664

In [15]:
```
options(repr.plot.width=6, repr.plot.height=4)
dl5 = ggplot(a, aes(x = carrier, y = Total_Delay_Minutes) ) + geom_bar(aes(fill=Total_
dl5
```



Data Preprocessing and Label encoding

In [16]:
```
df1 = df
head(df1,n=7)
```

schedtime	deptime	carrier	dest	distance	origin	weather	dayweek	daymonth	diff_minutes	del
1899-12-31 14:55:00	1899-12-31 14:55:00	OH	JFK	184	BWI	0	4	1	0	ontir
1899-12-31 16:40:00	1899-12-31 16:40:00	DH	JFK	213	DCA	0	4	1	0	ontir
1899-12-31 12:45:00	1899-12-31 12:45:00	DH	LGA	229	IAD	0	4	1	0	ontir
1899-12-31 17:15:00	1899-12-31 17:09:00	DH	LGA	229	IAD	0	4	1	0	ontir
1899-12-31 10:39:00	1899-12-31 10:35:00	DH	LGA	229	IAD	0	4	1	0	ontir
1899-12-31 12:40:00	1899-12-31 11:39:00	DH	JFK	228	IAD	0	4	1	0	ontir
1899-12-31 12:40:00	1899-12-31 12:43:00	DH	JFK	228	IAD	0	4	1	3	ontir

Dropping unnecessary columns

In [17]:
```
df1$schedtime <- NULL
df1$deptime <- NULL
head(df1,n=2)
```

carrier	dest	distance	origin	weather	dayweek	daymonth	diff_minutes	delay
OH	JFK	184	BWI	0	4	1	0	ontime
DH	JFK	213	DCA	0	4	1	0	ontime

Label Encoding

In [18]:
```
df1$carrier<- as.numeric(factor(df1$carrier))
head(df1,n=2)
```

carrier	dest	distance	origin	weather	dayweek	daymonth	diff_minutes	delay
5	JFK	184	BWI	0	4	1	0	ontime
2	JFK	213	DCA	0	4	1	0	ontime

In [19]:
```
df1$dest<- as.numeric(factor(df1$dest))
df1$origin<- as.numeric(factor(df1$origin))
head(df1,n=7)
```

carrier	dest	distance	origin	weather	dayweek	daymonth	diff_minutes	delay
5	2	184	1	0	4	1	0	ontime
2	2	213	2	0	4	1	0	ontime
2	3	229	3	0	4	1	0	ontime
2	3	229	3	0	4	1	0	ontime
2	3	229	3	0	4	1	0	ontime
2	2	228	3	0	4	1	0	ontime
2	2	228	3	0	4	1	3	ontime

In [20]:
```
df1$delay <- ifelse(df1$delay == "ontime",1,0)
head(df1,n=7)
```

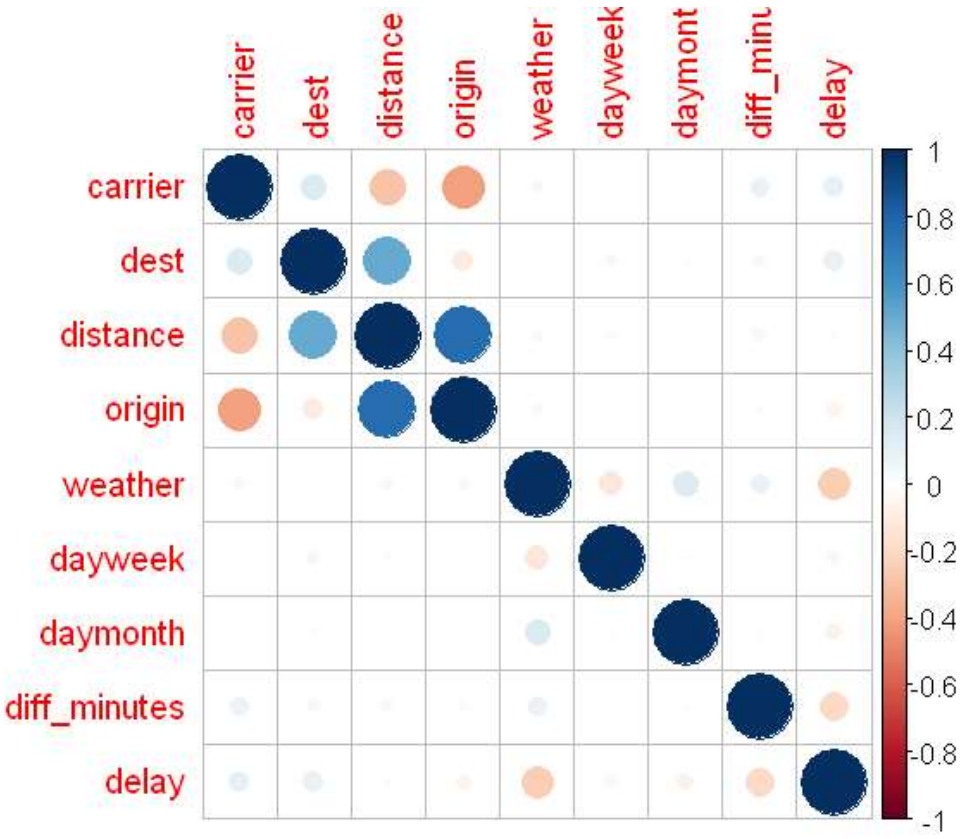
carrier	dest	distance	origin	weather	dayweek	daymonth	diff_minutes	delay
5	2	184	1	0	4	1	0	1
2	2	213	2	0	4	1	0	1
2	3	229	3	0	4	1	0	1
2	3	229	3	0	4	1	0	1
2	3	229	3	0	4	1	0	1
2	2	228	3	0	4	1	0	1
2	2	228	3	0	4	1	3	1

Correlation Matrix

In [21]:
```
correlation_matrix <- round(cor(df1),2)
correlation_matrix
#head(correlation_matrix[, 1:9])
```

	carrier	dest	distance	origin	weather	dayweek	daymonth	diff_minutes	delay
carrier	1.00	0.15	-0.28	-0.40	-0.04	-0.01	0.00	0.09	0.10
dest	0.15	1.00	0.51	-0.10	0.00	-0.05	0.02	0.04	0.09
distance	-0.28	0.51	1.00	0.76	0.03	-0.02	0.01	0.04	0.02
origin	-0.40	-0.10	0.76	1.00	0.03	0.00	0.00	0.02	-0.06
weather	-0.04	0.00	0.03	0.03	1.00	-0.12	0.14	0.08	-0.25
dayweek	-0.01	-0.05	-0.02	0.00	-0.12	1.00	0.02	0.00	0.04
daymonth	0.00	0.02	0.01	0.00	0.14	0.02	1.00	0.02	-0.07
diff_minutes	0.09	0.04	0.04	0.02	0.08	0.00	0.02	1.00	-0.19
delay	0.10	0.09	0.02	-0.06	-0.25	0.04	-0.07	-0.19	1.00

In [22]:
```
corrplot::corrplot(cor(df1))
```



In [23]:
```
colnames(df1)
```

1. 'carrier'
2. 'dest'
3. 'distance'
4. 'origin'
5. 'weather'
6. 'dayweek'
7. 'daymonth'
8. 'diff_minutes'
9. 'delay'

Building a Logistic Regression Model

In [24]:
```
set.seed(123)
```

In [25]:
```
sample <- sample.split(df1$delay, SplitRatio = 0.7)
train  <- subset(df1, sample == TRUE)
test   <- subset(df1, sample == FALSE)
```

In [26]:
```
dim(train)
dim(test)
```

1. 1541
2. 9

1. 660
2. 9

In [27]:
```
#df2 <- train %>% mutate_at(c('carrier','dest', 'distance', 'origin', 'weather', 'dayw
#df2
df2 <- train %>% mutate_all(~(scale(.) %>% as.vector))
head(df2,n=7)
dim(df2)
```

carrier	dest	distance	origin	weather	dayweek	daymonth	diff_minutes	
0.2973115	-0.2528852	-2.12346610	-2.2184007	-0.1146329	0.06525838	-1.723113	-0.2264426	0.491
-1.0204938	0.8865771	1.29154252	1.3333452	-0.1146329	0.06525838	-1.723113	-0.2264426	0.491
-1.0204938	-0.2528852	1.21565344	1.3333452	-0.1146329	0.06525838	-1.723113	-0.2264426	0.491
-1.0204938	-0.2528852	1.21565344	1.3333452	-0.1146329	0.06525838	-1.723113	-0.1926462	0.491
-1.0204938	-0.2528852	1.21565344	1.3333452	-0.1146329	0.06525838	-1.723113	-0.2264426	0.491
-1.0204938	-0.2528852	1.21565344	1.3333452	-0.1146329	0.06525838	-1.723113	-0.1250533	0.491
-0.5812254	-0.2528852	0.07731723	-0.4425277	-0.1146329	0.06525838	-1.723113	-0.1926462	0.491

1. 1541
2. 9

In [28]:
```
model <- glm(delay ~.,family=binomial(link='logit'),data=train)
summary(model)
```

```
Call:
glm(formula = delay ~ ., family = binomial(link = "logit"), data = train)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.2582   0.4131   0.5853   0.6395   1.8453

Coefficients:
              Estimate Std. Error z value Pr(>|z|)
(Intercept)  -3.349e+00  2.865e+00  -1.169    0.242
carrier       1.570e-01  3.692e-02   4.254 2.10e-05 ***
dest          3.458e-02  1.907e-01   0.181    0.856
distance      2.424e-02  1.892e-02   1.281    0.200
origin       -3.476e-01  3.841e-01  -0.905    0.365
weather      -1.663e+01  3.206e+02  -0.052    0.959
dayweek       1.595e-03  3.610e-02   0.044    0.965
daymonth     -7.182e-03  7.931e-03  -0.905    0.365
diff_minutes -4.536e-03  7.166e-04  -6.329 2.46e-10 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 1519.2  on 1540  degrees of freedom
Residual deviance: 1376.3  on 1532  degrees of freedom
AIC: 1394.3

Number of Fisher Scoring iterations: 14
```

In [29]:
```
model1 <- glm(delay ~ carrier + diff_minutes ,family=binomial(link='logit'),data=trair
summary(model1)
```

```
Call:
glm(formula = delay ~ carrier + diff_minutes, family = binomial(link = "logit"),
    data = train)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.1246   0.4702   0.5635   0.6742   2.0445

Coefficients:
              Estimate Std. Error z value Pr(>|z|)
(Intercept)   0.9132120  0.1372087   6.656 2.82e-11 ***
carrier       0.1541550  0.0314464   4.902 9.48e-07 ***
diff_minutes -0.0048978  0.0007904  -6.197 5.77e-10 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 1519.2  on 1540  degrees of freedom
Residual deviance: 1444.9  on 1538  degrees of freedom
AIC: 1450.9

Number of Fisher Scoring iterations: 4
```

In [30]:
```
anova(model1, test="Chisq")
```

	Df	Deviance	Resid. Df	Resid. Dev	Pr(>Chi)
NULL	NA	NA	1540	1519.230	NA
carrier	1	18.28962	1539	1500.941	1.897378e-05
diff_minutes	1	56.03767	1538	1444.903	7.109563e-14

In [31]:
```
fitted.results <- predict(model1,test,type='response')
fitted.results <- ifelse(fitted.results > 0.5,1,0)

misClasificError <- mean(fitted.results != test$delay)
sprintf(misClasificError, fmt = '%#.3f')        # Apply sprintf function
k = 1-misClasificError
r = sprintf(k, fmt = '%#.2f')
print(paste('Accuracy',r))
```

'0.200'

```
[1] "Accuracy 0.80"
```

In [32]:
```
probabilities <- model1 %>% predict(test, type = "response")
predicted.classes <- ifelse(probabilities > 0.5, "Ontime", "Delay")
```

In [33]:
```
df3 = test
head(df3,n=2)
```

carrier	dest	distance	origin	weather	dayweek	daymonth	diff_minutes	delay
2	2	213	2	0	4	1	0	1
2	3	229	3	0	4	1	0	1

In [34]:
```
df3$New_delay <- predict(model1,test, type = "response")
head(df3,n=3)
```

carrier	dest	distance	origin	weather	dayweek	daymonth	diff_minutes	delay	New_delay
2	2	213	2	0	4	1	0	1	0.7723313
2	3	229	3	0	4	1	0	1	0.7723313
2	3	229	3	0	4	1	0	1	0.7723313

In [35]:
```
df4 <- cbind(df3,predicted.classes)
head(df4,n=3)
```

carrier	dest	distance	origin	weather	dayweek	daymonth	diff_minutes	delay	New_delay	predict
2	2	213	2	0	4	1	0	1	0.7723313	
2	3	229	3	0	4	1	0	1	0.7723313	
2	3	229	3	0	4	1	0	1	0.7723313	

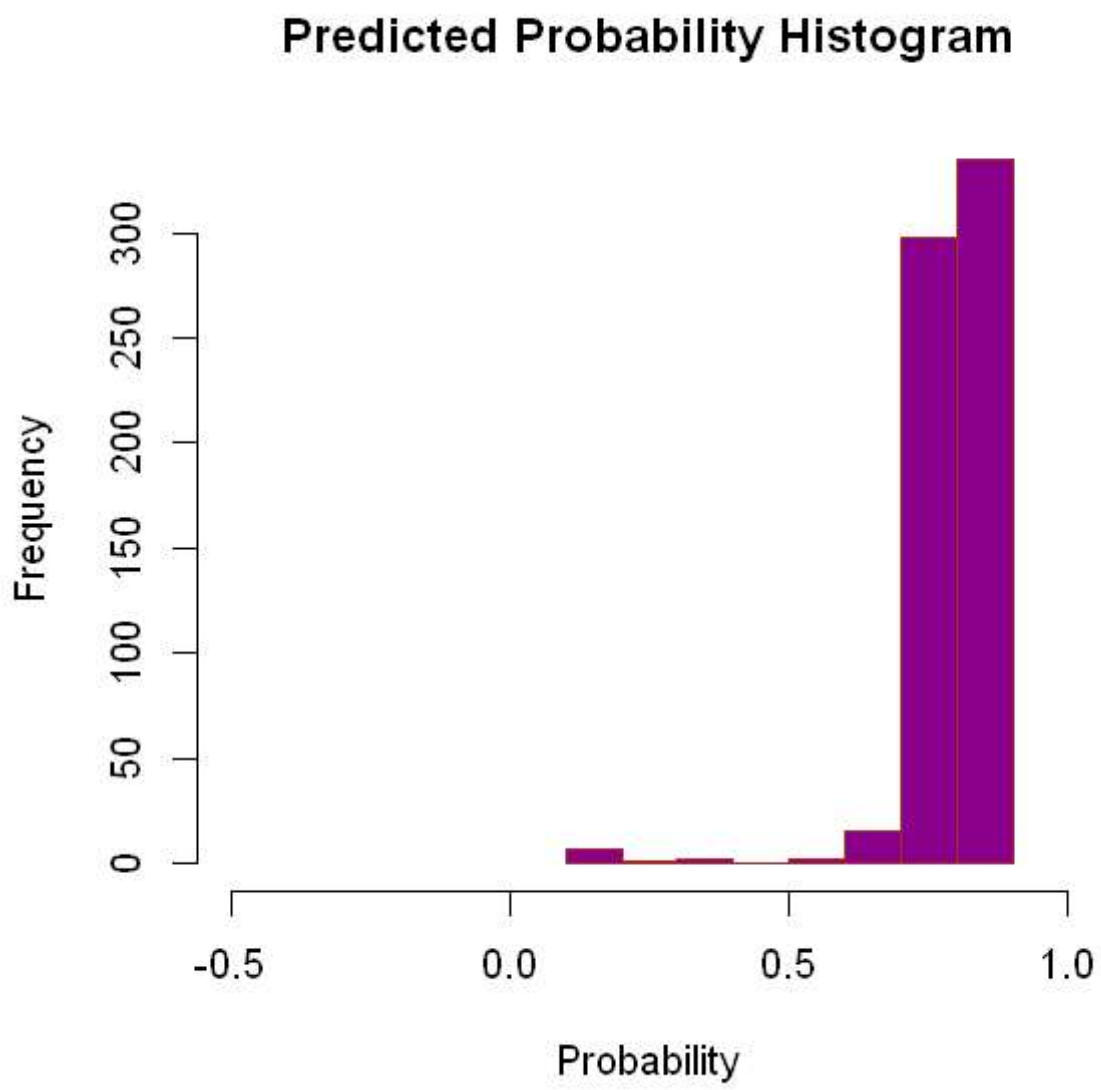
In [36]:
```
names(df4)[11] <- "Final_Prediction"
head(df4,n=3)
```

carrier	dest	distance	origin	weather	dayweek	daymonth	diff_minutes	delay	New_delay	Final_P
2	2	213	2	0	4	1	0	1	0.7723313	
2	3	229	3	0	4	1	0	1	0.7723313	
2	3	229	3	0	4	1	0	1	0.7723313	

In [37]:
```
options(repr.plot.width=5, repr.plot.height=5)
```

In [38]:
```
h1 <- hist(df3$New_delay,
main="Predicted Probability Histogram",
xlab="Probability",
xlim=c(-0.5,1),
col="darkmagenta",
border="brown")
h1
```

```
$breaks
[1] 0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9

$counts
[1]   7   1   2   0   2  15 298 335

$density
[1] 0.10606061 0.01515152 0.03030303 0.00000000 0.03030303 0.22727273 4.51515152
[8] 5.07575758

$mids
[1] 0.15 0.25 0.35 0.45 0.55 0.65 0.75 0.85

$xname
[1] "df3$New_delay"

$equidist
[1] TRUE

attr(,"class")
[1] "histogram"
```



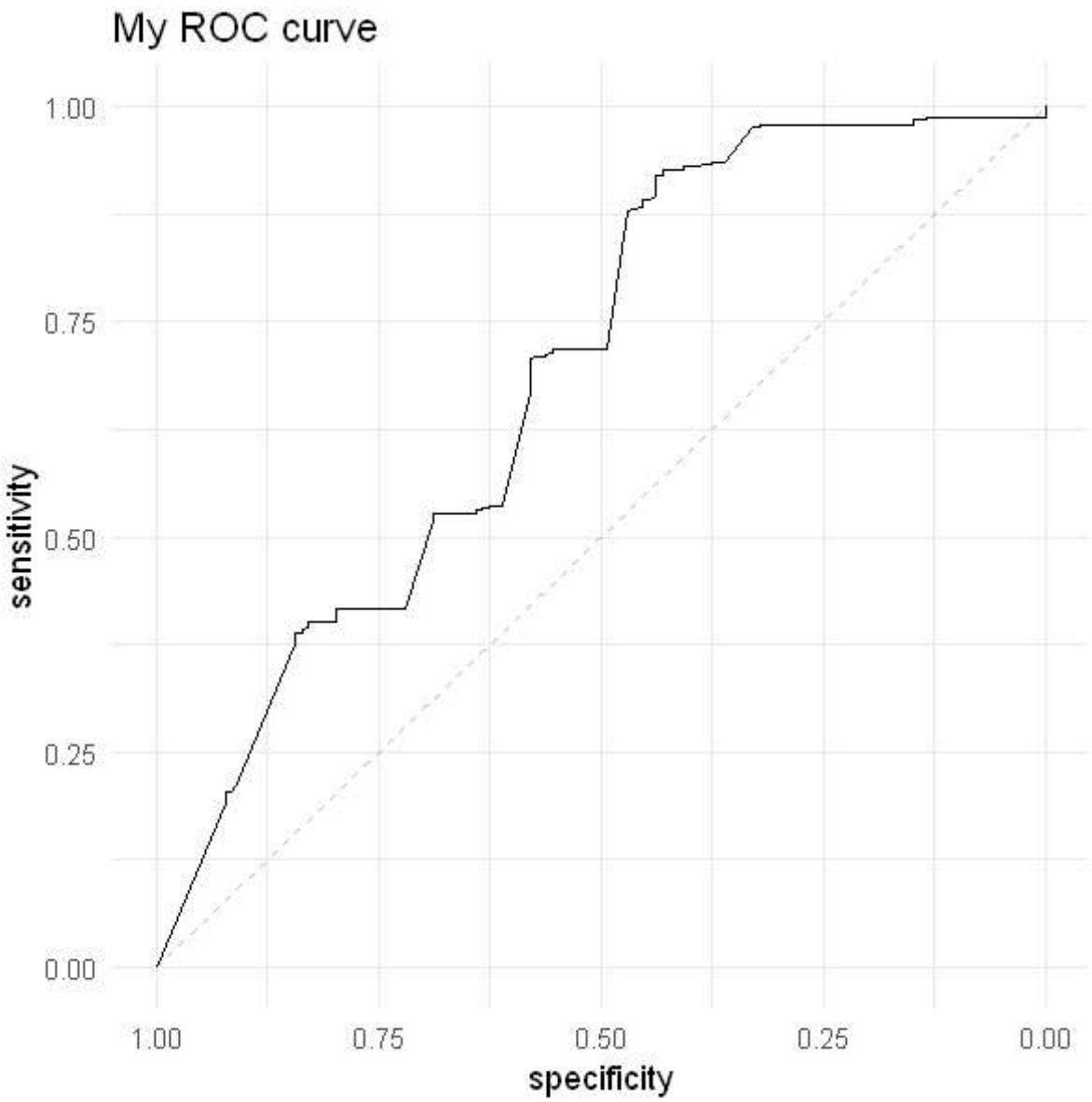
Model evaluation, ROC Plot and AUC

In [39]:
```
roc(df3$delay,df3$New_delay)
```

```
Setting levels: control = 0, case = 1
Setting direction: controls < cases
Call:
roc.default(response = df3$delay, predictor = df3$New_delay)

Data: df3$New_delay in 128 controls (df3$delay 0) < 532 cases (df3$delay 1).
Area under the curve: 0.6882
```

In [40]:
```
g4 = ggroc(roc(df3$delay,df3$New_delay)) +
  theme_minimal() +
  ggtitle("My ROC curve") +
  geom_segment(aes(x = 1, xend = 0, y = 0, yend = 1), color="grey", linetype="dashed")
g4
```

```
Setting levels: control = 0, case = 1
Setting direction: controls < cases
```



In [41]:
```
a1 = auc(df3$delay,df3$New_delay)
b1= round(a1, digits = 2)
print(paste0("The Value of AUC: ", b1))
```

```
Setting levels: control = 0, case = 1
Setting direction: controls < cases
[1] "The Value of AUC: 0.69"
```

Misclassification error is 0.80 which is good for the model

Area Under Curve (AUC) value is 0.69 which is good

AUC close to 0.70 represents a good model

In [ ]: