INSURANCE CLAIM DATASET

In [1]:

```
library(ggplot2)
```

```
Warning message:
"package 'ggplot2' was built under R version 3.6.2"
```

importing the required libraries for svm classification

In [2]:

```
library(e1071)
```

```
Warning message:
"package 'e1071' was built under R version 3.6.3"
```

In [3]:

```
library(dplyr)
```

```
Warning message:
"package 'dplyr' was built under R version 3.6.2"
Attaching package: 'dplyr'

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

    filter, lag

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

    intersect, setdiff, setequal, union
```

reading the csv data and storing it in insurance

In [5]:

```
insurance = read.csv("C:/Users/LENOVO/Desktop/insurance3r2.csv")
```

In [6]:

```
head(insurance)
```

age	sex	bmi	steps	children	smoker	region	charges	insuranceclaim
19	0	27.900	3009	0	1	3	16884.924	1
18	1	33.770	3008	1	0	2	1725.552	1
28	1	33.000	3009	3	0	2	4449.462	0
33	1	22.705	10009	0	0	1	21984.471	0
32	1	28.880	8010	0	0	1	3866.855	1
31	0	25.740	8005	0	0	2	3756.622	0

In [7]:

```
str(insurance)
```

```
'data.frame':   1338 obs. of  9 variables:
 $ age           : int  19 18 28 33 32 31 46 37 37 60 ...
 $ sex           : int  0 1 1 1 1 0 0 0 1 0 ...
 $ bmi           : num  27.9 33.8 33 22.7 28.9 ...
 $ steps         : int  3009 3008 3009 10009 8010 8005 3002 8007 8002 5008
...
 $ children      : int  0 1 3 0 0 0 1 3 2 0 ...
 $ smoker        : int  1 0 0 0 0 0 0 0 0 0 ...
 $ region        : int  3 2 2 1 1 2 2 1 0 1 ...
 $ charges       : num  16885 1726 4449 21984 3867 ...
 $ insuranceclaim: int  1 1 0 0 1 0 1 0 0 0 ...
```

There are some variables present as integer, we need to change them in to factors such as sex,children, smoker, region, insurance claim.

In [8]:

```
sum(is.na(insurance))
```

0

checking for missing values using is.na()..No missing values found

In [9]:

```
summary(insurance)
```

```
      age             sex              bmi             steps       
 Min.   :18.00   Min.   :0.0000   Min.   :15.96   Min.   : 3000  
 1st Qu.:27.00   1st Qu.:0.0000   1st Qu.:26.30   1st Qu.: 3008  
 Median :39.00   Median :1.0000   Median :30.40   Median : 4007  
 Mean   :39.21   Mean   :0.5052   Mean   :30.66   Mean   : 5329  
 3rd Qu.:51.00   3rd Qu.:1.0000   3rd Qu.:34.69   3rd Qu.: 8004  
 Max.   :64.00   Max.   :1.0000   Max.   :53.13   Max.   :10010  
    children         smoker           region         charges     
 Min.   :0.000   Min.   :0.0000   Min.   :0.000   Min.   : 1122  
 1st Qu.:0.000   1st Qu.:0.0000   1st Qu.:1.000   1st Qu.: 4740  
 Median :1.000   Median :0.0000   Median :2.000   Median : 9382  
 Mean   :1.095   Mean   :0.2048   Mean   :1.516   Mean   :13270  
 3rd Qu.:2.000   3rd Qu.:0.0000   3rd Qu.:2.000   3rd Qu.:16640  
 Max.   :5.000   Max.   :1.0000   Max.   :3.000   Max.   :63770  
 insuranceclaim  
 Min.   :0.0000  
 1st Qu.:0.0000  
 Median :1.0000  
 Mean   :0.5852  
 3rd Qu.:1.0000  
 Max.   :1.0000  
```

we can see the summary of data minimum charge is 1122 and maximum charges 63770

converting into factors from int datatype and replacing them. using factor function to convert it into factors

In [10]:

```
insurance$insuranceclaim = factor(insurance$insuranceclaim)
```

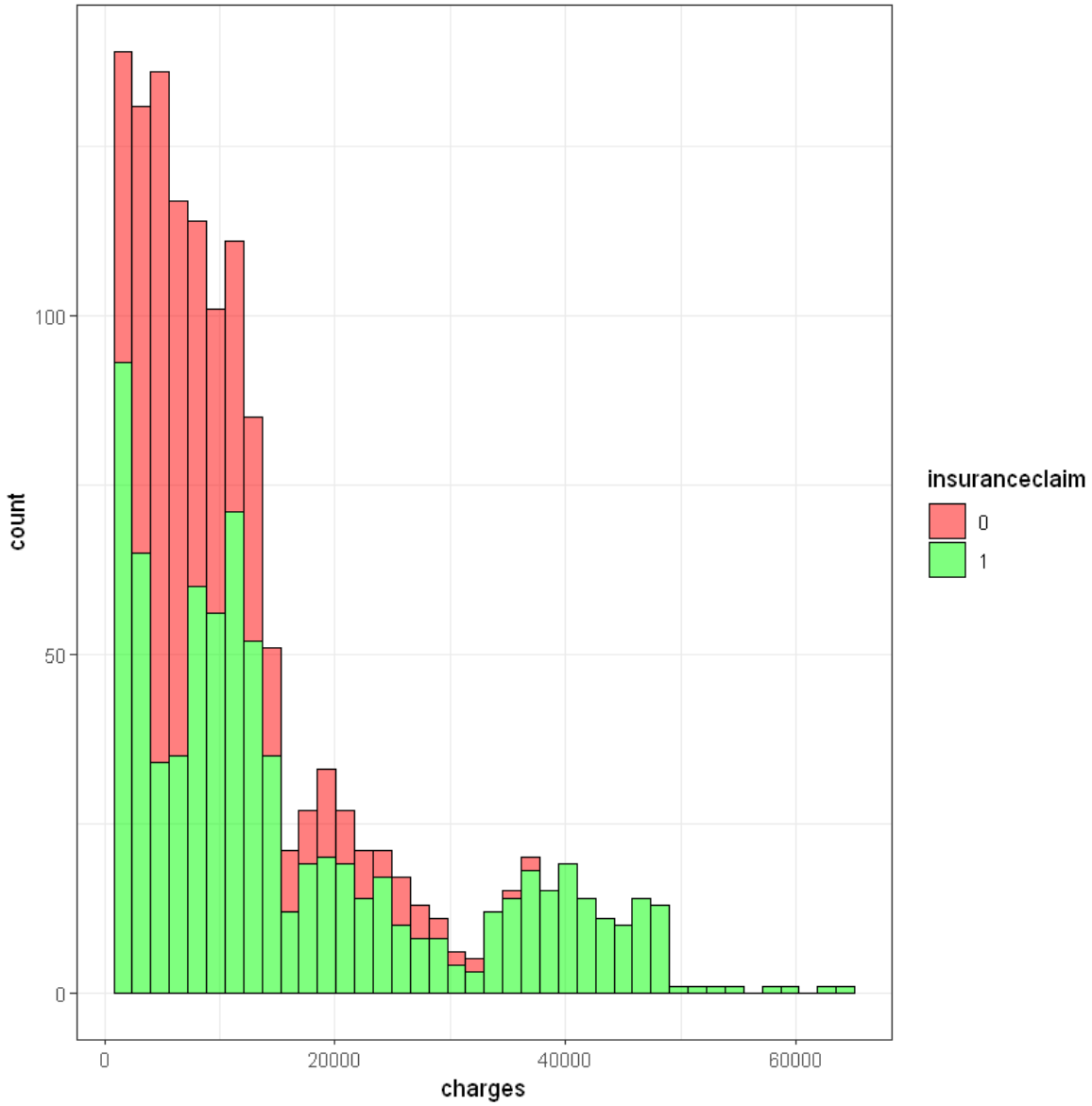
In [11]:

```
insurance$ sex = factor(insurance$ sex)
insurance$children = factor(insurance$children)
insurance$ smoker = factor(insurance$ smoker)
insurance$ region = factor(insurance$ region)
```

EDA

In [13]:

```
pl = ggplot(insurance, aes(x=charges))
pl = pl+geom_histogram(aes(fill = insuranceclaim), color = 'black', bins = 40, alpha = 0.5)
pl + scale_fill_manual(values = c('red','green')) + theme_bw()
```



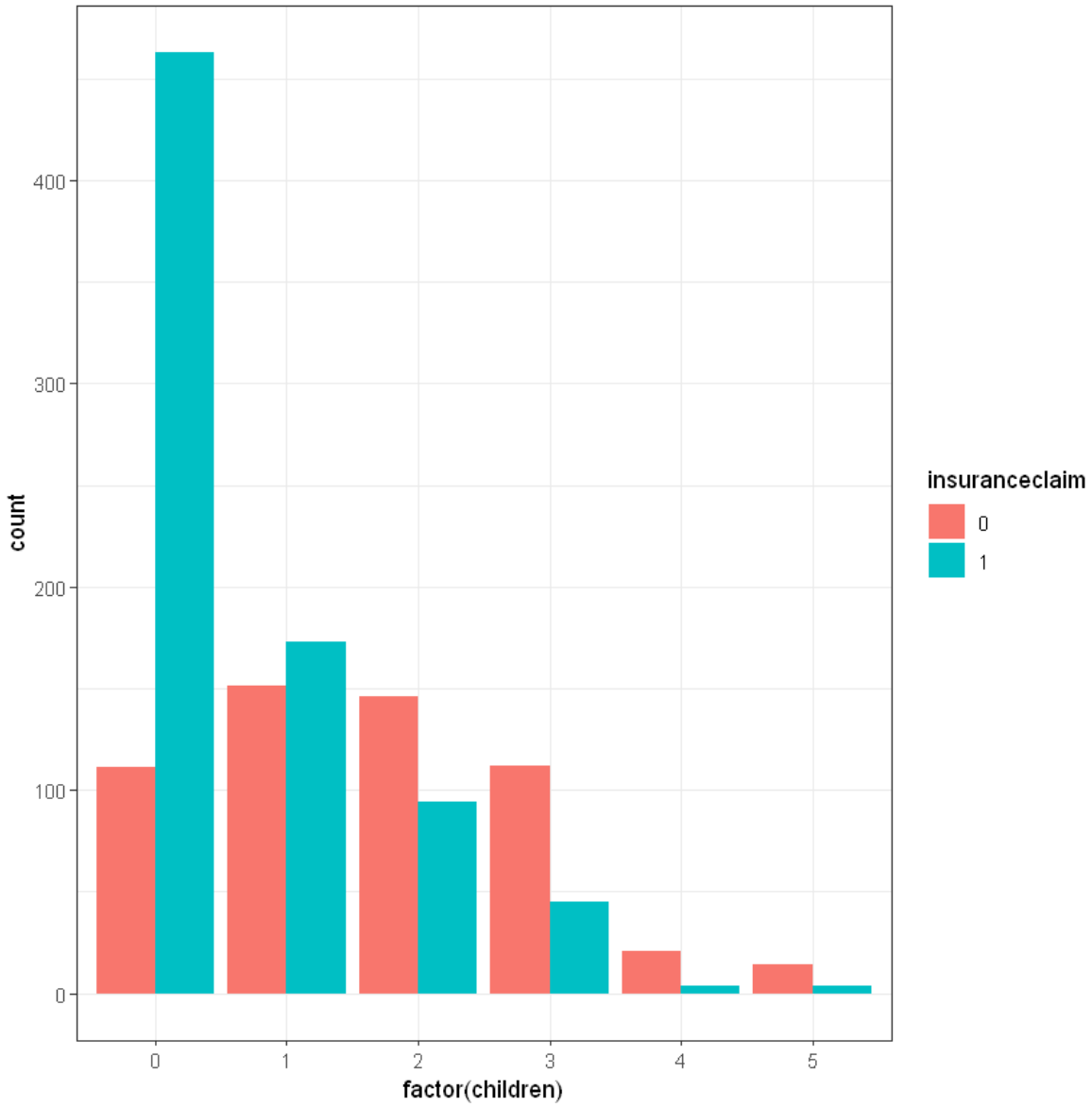
Most of the people are opting for a low charge insurance premium.

From the graph we can analyze that the customers with high charges are more likely to make their insurance claim effectively.

It can also be seen that customers who have thier insurance charges of over 4000 will definitely claim thier insurance.

In [17]:

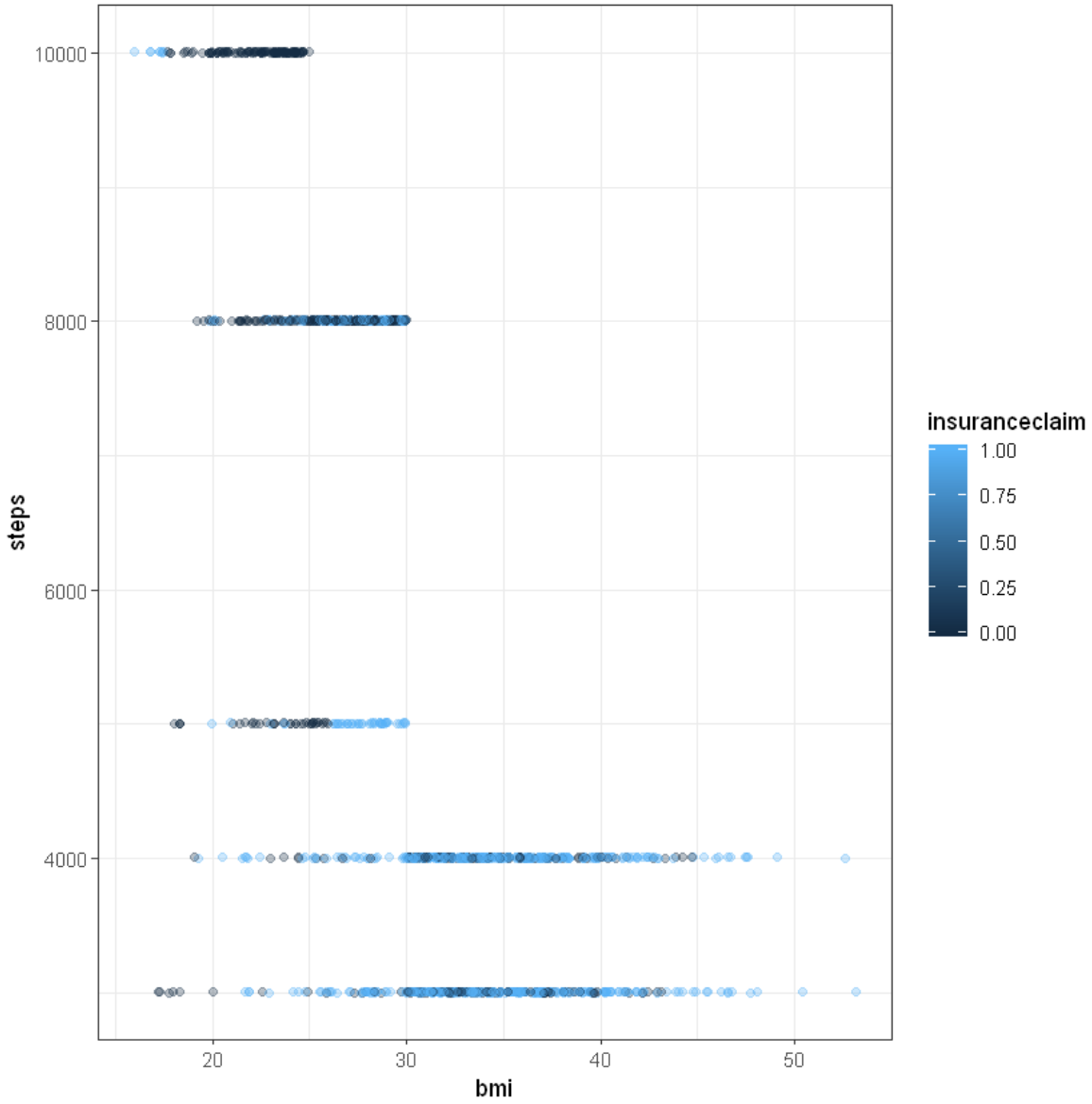
```
pl = ggplot(insurance, aes(x = factor(children)))
pl = pl + geom_bar(aes(fill = insuranceclaim), position = 'dodge')
pl + theme_bw()
```



It can be seen that customers with no (0) or 1 children are more likely to make their insurance claim sucessful

In [18]:

```
ggplot(loans, aes(bmi,steps)) + geom_point(aes(color = insuranceclaim), alpha = 0.3) + them
```



In [19]:

```
library(caTools)
```

```
Warning message:
"package 'caTools' was built under R version 3.6.2"
```

Importing required library for svm to execute

In [20]:

```
set.seed(101)
```

In [21]:

```
spl = sample.split(insurance$insuranceclaim, 0.7)
```

creating the split based on target variable and split ratio is 70:30

In [22]:

```
train = subset(insurance, spl = TRUE)
```

70% of data storing in train by giving spl equal to true

In [23]:

```
test = subset(insurance, spl = FALSE)
```

30% of data storing in test by giving spl equal to false

TRAINING THE MODEL USING SVM FUNCTION WITHOUT TUNING THE COST AND GAMMA

In [24]:

```
model = svm(insuranceclaim ~ ., data = train)
```

In [25]:

```
summary(model)
```

```
Call:
svm(formula = insuranceclaim ~ ., data = train)


Parameters:
   SVM-Type:  C-classification
 SVM-Kernel:  radial
       cost:  1

Number of Support Vectors:  631

 ( 317 314 )


Number of Classes:  2

Levels:
 0 1
```

From the summary support vectors on margin are 631 used to divide the margins

In [27]:

```
predicted.values = predict(model, test[0:9])
```

Removing the label column from test and passing model into predict function to predict the data without labels

In [28]:

```
table(predicted.values, test$insuranceclaim)
```

```
predicted.values   0   1
               0 488  73
               1  67 710
```

From the confusion matrix we can see we got bad results as cost and gamma values are not properly defined. It incorrectly classified the customers making their insurance claim column.

The Accuracy now is 89.5%

gamma and cost parameters are bad so we need to tune the results

In [29]:

```
model = svm(insuranceclaim ~ ., data = train, gamma = 1, cost = 10)
```

Training the model with better gamma and cost function

In [31]:

```
predicted.values1 = predict(model, test[0:9])
```

predicting using predict function after tuning the model

printing confusion matrix

In [32]:

```
table(predicted.values1, test$insuranceclaim)
```

```
predicted.values1   0   1
                0 552   1
                1   3 782
```

We have got good results by tuning the model. svm correctly classified as 552 customers didn't make their insurance claim and 782 customers made their insurance claim.

After tuning the model the achieved accuracy is 99.7%

training the model with different gamma and cost function

In [33]:

```
model = svm(insuranceclaim ~ ., data = train, gamma = 0.1, cost = 200)
```

In [34]:

```
predicted.values2 = predict(model, test[0:9])
```

In [35]:

```
table(predicted.values2, test$insuranceclaim)
```

```
predicted.values2   0   1
                0 545   9
                1  10 774
```

Here we can see that when we increase the cost function further the model's accuracy started to decline. so now the accuracy has fallen from 99.7% for cost=100 to 98.5% for cost=200.

In [ ]:

```

```