#load the required libraries

library(readxl)

library(dplyr)

library(ggplot2)

library(tidyverse)

library(corrplot)

#load the Dataset

data\_ds = read.csv('Insurance\_factor\_identification.csv')

#check the top 6 rows of dataset

head(data\_ds)

#check the bottom 6 rows of dataset

tail(data\_ds)

#check the type of of dataset ensure that it is a data.frame

class(data\_ds)

#check the shape/dimension of the dataset

dim(data\_ds)

#check the structure of the dataset and analyse the columns data type

str(data\_ds)

#check allt he column names in there dataset

names\_ds = names(data\_ds)

names\_ds

#check number of rows in the data set

print(nrow(data\_ds))

rown=nrow(data\_ds)

coln = ncol(data\_ds)

#check number of columns in the dataset

print(ncol(data\_ds))

#TASK 1

#The committee is interested to know each field of the data collected through

#descriptive analysis to gain basic insights into the data set and

#to prepare for further analysis

#SOLUTION

#check if the dataset has any null values

print(is.na(data\_ds))

total\_null\_value = sum(is.na(data\_ds))

message=paste("number of null values in dataset is ",total\_null\_value)

cat(message)

summary(data\_ds)

#so we have got 2182x7 data set which has information about

#kilometer traveled by the vehicle that year, Area or zone the the vehicle is registered to

#Bonus - how long its been since last claim

# make type of the vehicle, No of policy years insured

# Claims made by the vehicle owner

# and the insurance paid by the insurance company

#to check a comparison plot between all the columns fo the dataset

pairs(data\_ds, col = data\_ds$Payment,

main = "Relational analysis of all the columns in dataset")

#based on the pair plot result we can see that Payment is increasing when Claims and Insured amount is increasing

#Payment does not have much effect based on Kilometers driven, Zone, and bonus

#but another observation is - payment touches its maximum when Make is at maximum that is 9

ggplot(data\_ds, aes(x=Make, y=Payment, col=Make))+

labs(x="Make", y="Total Payment", title = "Correlation of Payment and Vehicle Model")+

geom\_point()

#This tells us that amongst all the make, the one represented by 9, causes a lot of claims and leads to maximum payment

#this gives me an idea that model type - 9 has some faulty component or perhaps poor build which causes more insurance claims

#let's also analyze which model is the best one with maximum number of 0 insurance claims

data\_no\_claim = data\_ds[data\_ds$Payment==0,]

data\_no\_claim$Make <- factor(data\_no\_claim$Make)

ggplot(data\_no\_claim, aes(x=Make, fill=Make))+

geom\_bar()+

labs(x="Make", y="Count")+

geom\_text(stat = 'count', aes(label = ..count..), vjust = -0.5)+

ggtitle("Countplot of Make in 0 Payment data subset")

#Let's do the same analysis for Maximum number of insurance claims for each make type

data\_max\_claims = data\_ds[data\_ds$Payment >=50000,]

data\_max\_claims$Make <- factor(data\_max\_claims$Make)

ggplot(data\_max\_claims, aes(x=Make, fill=Make))+

geom\_bar()+

labs(x="Make", y="Count")+

geom\_text(stat = "Count", aes(label = ..count..), vjust=-0.05)+

ggtitle("Countplot of Make in maxmimum Payment(>50000) Data subset")

#Lets try to visualize it individually

#first i want to check if there are outliers in Kilometer, Zone and Bonus using boxplot

title = "Box plot for kilometer, Zone, Bonus, Make"

boxplot\_colors <- c("red", "blue", "green", "purple")

boxplot\_labels <- names(data\_ds)

boxplot(data\_ds$Kilometres, data\_ds$Zone, data\_ds$Bonus, data\_ds$Make, main=title, col = boxplot\_colors)

medians <- sapply(list(data\_ds$Kilometres, data\_ds$Zone, data\_ds$Bonus, data\_ds$Make),

function(x) quantile(x, c(0.25, 0.5, 0.75)))

# Add text labels for median, Q1, and Q3

text(x = 1:4, y = medians["50%", ], labels = paste("Median: ", round(medians["50%", ], 2)), pos = 3)

text(x = 1:4, y = medians["25%", ], labels = paste("Q1: ", round(medians["25%", ], 2)), pos = 1)

text(x = 1:4, y = medians["75%", ], labels = paste("Q3: ", round(medians["75%", ], 2)), pos = 1)

for (i in 1:4) {

text(x = i, y = par("usr")[3] - 0.15, labels = boxplot\_labels[i], srt = 45, pos = 2, xpd = TRUE)

}

#Analysis of data based on box plot

# Observation1 - There are no outliers in all 4 columns Kilometer, Zone, Bonus and Make

# Observation2 - Data is normally distributed because median is at the center of all 4 boxes,

#its symmetrical, so no skewness

# Observation3 - Kilometers has a shorter spread or IQR spread and Make has largest spread as compared to others

#Lets plot the histogram for numerical data Insured and Claims

hist(data\_ds$Insured ,col='red',main="Histogram of Count Insured", breaks=20, xlab= "Insured")

hist(data\_ds$Claims,col='blue',breaks=20, main="Histogram of Count Claims", xlab="Claims")

hist(data\_ds$Payment, col = 'green',breaks=20, main="Histogram of Payment", xlab = "Payment")

#just to be sure let's plot the density of Insurance, Claims and payment

plot(density(data\_ds$Insured), col = 'red',main = "Density Plot of Insured", xlab= "Insured")

plot(density(data\_ds$Claims), col = 'blue',main = "Density Plot of Claims", xlab="Claims")

plot(density(data\_ds$Payment), col = 'green', main = "Density plot of Payment",xlab = "Payment" )

#as a result we see that the Insured, Claims and Payment have similar Distribution

#lets study the top 3 and bottom 3 based on Insured, Claims and Payment

#sort the dataset based on Insured amount and check top 3

sort\_Insured = data\_ds[order(data\_ds$Insured),]

tail(sort\_Insured, 3)

head(sort\_Insured,3)

#sort the dataset based on Claims and check top 3

sort\_Claims = data\_ds[order(data\_ds$Claims),]

tail(sort\_Claims, 3)

head(sort\_Claims,3)

#sort the dataset based on Payment and check top 3

sort\_payment = data\_ds[order(data\_ds$Payment),]

tail(sort\_payment, 3)

head(sort\_payment, 3)

#Based on the results it is interesting to see that the top

#3 Payment is also Top 3 Insured and top 3 Claims

#Bottom 3 payments is also bottom 3 claims but interestingly not the same bottom 3 insured

#moving on to our next task - TASK2

#The total value of payment by an insurance company is an important factor to be monitored.

#So the committee has decided to find whether this payment is related to the

#number of claims and the number of insured policy years.

#They also want to visualize the results for better understanding.

#to understand if the "Payment" is related to to the "Claims" and insured policy years "Insured"

# above based on density plot and histogram we saw that our data is skewed for Insured, Claims and Payement

#dim(data\_ds)

#data\_ds <- data\_ds[data\_ds$Insured <= 6000, ] #remove all rows which has insured more than 6000 to remove skewness

#dim(data\_ds)

#SOLUTION

#we should first calculate the correlation between the dependent and independent variables

#for visualization we can plot it as scatter plot

sprintf("The correlation of Distance and payment is: %.3f",cor(data\_ds$Kilometres, data\_ds$Payment))

sprintf("The correlation of Zone and PAyment is: %.3f",cor(data\_ds$Zone, data\_ds$Payment))

sprintf("The correlation of Bonus with Payment is: %.3f",cor(data\_ds$Bonus, data\_ds$Payment))

sprintf("The correlation of Make with Payment is: %.3f",cor(data\_ds$Make, data\_ds$Payment))

sprintf("The correlation of Insured vs Payment is: %.3f",cor(data\_ds$Insured, data\_ds$Payment))

sprintf("The correlation of Claims Vs Payment is %.3f",cor(data\_ds$Claims, data\_ds$Payment))

#based on the outcome we can see that Distance and Zone has negative correlation that too is very negligible

#Bonus and Make has positive correlation but that too very feeble

#there is significant positive correlation between Insured policy year and Payment also Claims and Payment

#plot the correlation (scatter plot - using geom\_point) between Insured years and Payment

ggplot(data\_ds, aes(x=Insured, y=Payment, col = Insured))+

labs(x="insured years", y="Total Payment", title="correlation of payment and insured years")+

geom\_point()

#plot the correlation between Claims and Payment using Scatter plot

ggplot(data\_ds, aes(x=Claims, y=Payment, col=Claims))+

labs(x="Number of Claims", y="Total Payment", title = "Correlation of Payment and No pf claims")+

geom\_point()

#the scatter plot depicts that we have a very strong dependency of payment

#on Insured years and Number of claims

#moving on to next task - TASK3

#The committee wants to figure out the reasons for insurance payment increase and decrease.

#So they have decided to find whether distance, location, bonus, make,

#and insured amount or claims are affecting the payment or all or some of these are affecting it.

#SOLUTION

#to understand the impact of any of these variables on payment we should do a multivariate analysis

#first let's build a model only for KM, Zone, Bonus and MAke, we know this has very less correlation

# just for understanding purpose , let's see how these variables impact payment

regression\_payment<-lm(Payment~Kilometres+Zone+Bonus+Make,data=data\_ds)

summary(regression\_payment)

#output

#Residual standard error: 966700 on 2177 degrees of freedom

#Multiple R-squared: 0.09871, Adjusted R-squared: 0.09705

#F-statistic: 59.61 on 4 and 2177 DF, p-value: < 2.2e-16

#studying the outcome

#A high residual standard error indicates that the model's predictions is not very close to actual values.

#R-square of 9.87 tells us that KM, Zone, Bonus and Make all together only account for 9.87% variation in payment

# Adjusted R-Square of 9.70% tell us that these 4 variables do not give good explanation for the variation in Payment

# F score of 59.61 with 4 and 2177 degree of freedom tells us that the model is statistically significant

# F score is high which tells us that model's prediction ability is not very good

# very small p-value indicates that at the minimum one of the variable has significant affect on Payment

#now we see that when we built a model excluding the main correlated variable (with Payment), the model's prediction is not very good

#however we can also see that model tells us that there is some unknown factor which is impacting the "payment"

# it is very much possible that "Insured" and "Claims" are those factors

#let's build another model with insured and claims'

reg\_model = lm(Payment~Insured+Claims+Kilometres+Zone+Bonus+Make, data= data\_ds)

summary(reg\_model)

# Studying the outcome

#Residual standard error: 70830 on 2175 degrees of freedom

#Multiple R-squared: 0.9952, Adjusted R-squared: 0.9952

#F-statistic: 7.462e+04 on 6 and 2175 DF, p-value: < 2.2e-16

#with this model we see that we have 99.51% of r-squared and same Adjusted R-Squared value

# so our model is able to explain the variation in Payment very well

# a lower value of F-score with degree of freedom as 6 and 2175 model is statistically significant and

#model's prediction ability is very good

#moving on to next task - TASK4

#The insurance company is planning to establish a new branch office,

#so they are interested to find at what location, kilometre, and bonus level their

#insured amount, claims, and payment gets increased. (Hint: Aggregate Dataset)

#SOLUTION

#To achieve this we need to aggregate our dataset for Insured, claims and Payment

#So we create another dataset which has a column for aggregated/binded value of insured, claims and payment along with other variables

aggregate\_data = aggregate(cbind(Insured, Claims, Payment)~Kilometres+Zone+Bonus, data=data\_ds, FUN=mean)

ggplot(aggregate\_data, aes(x=Kilometres, y=Payment))+

geom\_point(aes(color=Zone, size=Bonus))+

#geom\_text( aes(label = Zone), hjust=2)+

geom\_text(aes(label = Zone), hjust = -1, vjust = 0.5, position = position\_nudge(x = -0.1)) +

geom\_text(aes(label = Bonus), hjust = 2, vjust = 0.5, position = position\_nudge(x = 0.1)) +

labs(title="Payment vs. Kilometer by Location and Bonus", x = "Kilometer", y = "Payment")

#Plot Analysis

#with the given outcome we can see that:

# Kilometer - at kilometer value 5 i.e. more than 25000km driven vehicles have lowest payment, followed by 4

# Zone - At zone 7 the payment is lowest followed by 6 and 5, and Zone 4 payment is maximum

# Bonus - At Bonus 7 the payment is lowest followed by 6 and 5, and Bonus 7 payment is maximum

#let's start with next task - Task5

#The committee wants to understand what affects their claim rates

#so as to decide the right premiums for a certain set of situations.

#Hence, they need to find whether the insured amount, zone, kilometre,

#bonus, or make affects the claim rates and to what extent.

#SOLUTION

#Since Claim is a count type of variable

#for us to determine the whether Kilometre, bonus, Make and insured affects the claim

# we need to use regreesion algorithm

#To my knowledge the most suitable one for count type of data is "Poisson Regression"

#so let's build poisson regression model

pois\_model = glm(Claims~Kilometres+Zone+Bonus+Make+Insured, data= data\_ds, family=poisson)

summary(pois\_model)

#Outcome

#Coefficients:

# Estimate Std. Error z value Pr(>|z|)

#(Intercept) 2.656e+00 1.638e-02 162.17 <2e-16 \*\*\*

# Kilometres -2.692e-01 2.415e-03 -111.47 <2e-16 \*\*\*

# Zone -2.516e-01 1.744e-03 -144.27 <2e-16 \*\*\*

# Bonus 7.856e-02 1.656e-03 47.46 <2e-16 \*\*\*

# Make 3.765e-01 1.659e-03 226.95 <2e-16 \*\*\*

# Insured 2.837e-05 9.527e-08 297.82 <2e-16 \*\*\*

#Outcome Study

# Kilometres: For unit increase in "Kilometres," the log of the expected claims decreases by 0.2692. -> -ve affect

# Zone: for unit increase in Zone, the log of the claims decreased by 0.2516 -> -ve affect

# Bonus: for unit increase in bonus, the log of claims increases by 0.0786-> +ve but low affect

# Make: For unit increase in make, the log of claims increases by 0.3765 -> +ve and medium affect

# Insured: For unit increase in Insured, the log of claims increased by 0.00002837 -> +ve but extremely low affect

#Conclusion: "Kilometres" has low -ve affect on Claims,

# "Make" has Medium +ve affect on Claims

plot(pois\_model)

|  |
| --- |
| ***Console Window***  Hit <Return> to see next plot: #load the required libraries  Hit <Return> to see next plot: library(readxl)  Hit <Return> to see next plot: library(dplyr)  > library(ggplot2)  > library(tidyverse)  > library(corrplot)  > #load the Dataset  > data\_ds = read.csv('Insurance\_factor\_identification.csv')  > #check the top 6 rows of dataset  > head(data\_ds)  Kilometres Zone Bonus Make Insured Claims Payment  1 1 1 1 1 455.13 108 392491  2 1 1 1 2 69.17 19 46221  3 1 1 1 3 72.88 13 15694  4 1 1 1 4 1292.39 124 422201  5 1 1 1 5 191.01 40 119373  6 1 1 1 6 477.66 57 170913  > #check the bottom 6 rows of dataset  > tail(data\_ds)  Kilometres Zone Bonus Make Insured Claims Payment  2177 5 7 7 4 2.35 0 0  2178 5 7 7 5 8.74 0 0  2179 5 7 7 6 16.61 0 0  2180 5 7 7 7 2.83 1 966  2181 5 7 7 8 13.06 0 0  2182 5 7 7 9 384.87 16 112252  > #check the type of of dataset ensure that it is a data.frame  > class(data\_ds)  [1] "data.frame"  > #check the shape/dimension of the dataset  > dim(data\_ds)  [1] 2182 7  > #check the structure of the dataset and analyse the columns data type  > str(data\_ds)  'data.frame': 2182 obs. of 7 variables:  $ Kilometres: int 1 1 1 1 1 1 1 1 1 1 ...  $ Zone : int 1 1 1 1 1 1 1 1 1 1 ...  $ Bonus : int 1 1 1 1 1 1 1 1 1 2 ...  $ Make : int 1 2 3 4 5 6 7 8 9 1 ...  $ Insured : num 455.1 69.2 72.9 1292.4 191 ...  $ Claims : int 108 19 13 124 40 57 23 14 1704 45 ...  $ Payment : int 392491 46221 15694 422201 119373 170913 56940 77487 6805992 214011 ...  > #check allt he column names in there dataset  > names\_ds = names(data\_ds)  > names\_ds  [1] "Kilometres" "Zone" "Bonus" "Make" "Insured" "Claims" "Payment"  > #check number of rows in the data set  > print(nrow(data\_ds))  [1] 2182  > rown=nrow(data\_ds)  > coln = ncol(data\_ds)  > #check number of columns in the dataset  > print(ncol(data\_ds))  [1] 7  > #SOLUTION  > #check if the dataset has any null values  > print(is.na(data\_ds))  Kilometres Zone Bonus Make Insured Claims Payment  [1,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [2,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [3,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [4,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [5,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [6,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [7,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [8,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [9,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [10,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [11,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [12,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [13,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [14,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [15,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [16,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [17,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [18,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [19,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [20,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [21,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [22,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [23,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [24,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [25,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [26,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [27,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [28,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [29,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [30,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [31,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [32,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [33,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [34,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [35,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [36,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [37,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [38,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [39,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [40,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [41,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [42,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [43,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [44,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [45,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [46,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [47,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [48,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [49,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [50,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [51,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [52,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [53,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [54,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [55,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [56,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [57,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [58,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [59,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [60,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [61,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [62,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [63,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [64,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [65,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [66,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [67,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [68,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [69,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [70,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [71,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [72,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [73,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [74,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [75,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [76,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [77,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [78,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [79,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [80,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [81,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [82,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [83,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [84,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [85,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [86,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [87,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [88,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [89,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [90,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [91,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [92,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [93,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [94,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [95,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [96,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [97,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [98,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [99,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [100,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [101,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [102,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [103,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [104,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [105,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [106,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [107,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [108,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [109,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [110,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [111,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [112,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [113,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [114,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [115,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [116,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [117,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [118,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [119,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [120,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [121,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [122,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [123,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [124,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [125,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [126,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [127,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [128,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [129,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [130,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [131,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [132,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [133,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [134,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [135,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [136,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [137,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [138,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [139,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [140,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [141,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [142,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [ reached getOption("max.print") -- omitted 2040 rows ]  > total\_null\_value = sum(is.na(data\_ds))  > message=paste("number of null values in dataset is ",total\_null\_value)  > cat(message)  number of null values in dataset is 0  > summary(data\_ds)  Kilometres Zone Bonus Make Insured Claims  Min. :1.000 Min. :1.00 Min. :1.000 Min. :1.000 Min. : 0.01 Min. : 0.00  1st Qu.:2.000 1st Qu.:2.00 1st Qu.:2.000 1st Qu.:3.000 1st Qu.: 21.61 1st Qu.: 1.00  Median :3.000 Median :4.00 Median :4.000 Median :5.000 Median : 81.53 Median : 5.00  Mean :2.986 Mean :3.97 Mean :4.015 Mean :4.992 Mean : 1092.20 Mean : 51.87  3rd Qu.:4.000 3rd Qu.:6.00 3rd Qu.:6.000 3rd Qu.:7.000 3rd Qu.: 389.78 3rd Qu.: 21.00  Max. :5.000 Max. :7.00 Max. :7.000 Max. :9.000 Max. :127687.27 Max. :3338.00  Payment  Min. : 0  1st Qu.: 2989  Median : 27404  Mean : 257008  3rd Qu.: 111954  Max. :18245026  > #to check a comparison plot between all the columns fo the dataset  > pairs(data\_ds, col = data\_ds$Payment,  + main = "Relational analysis of all the columns in dataset")  > #based on the pair plot result we can see that Payment is increasing when Claims and Insured amount is increasing  > #Payment does not have much effect based on Kilometers driven, Zone, and bonus  > #but another observation is - payment touches its maximum when Make is at maximum that is 9  > ggplot(data\_ds, aes(x=Make, y=Payment, col=Make))+  + labs(x="Make", y="Total Payment", title = "Correlation of Payment and Vehicle Model")+  + geom\_point()  > #This tells us that amongst all the make, the one represented by 9, causes a lot of claims and leads to maximum payment  > #this gives me an idea that model type - 9 has some faulty component or perhaps poor build which causes more insurance claims  > #let's also analyze which model is the best one with maximum number of 0 insurance claims  > data\_no\_claim = data\_ds[data\_ds$Payment==0,]  > data\_no\_claim$Make <- factor(data\_no\_claim$Make)  > ggplot(data\_no\_claim, aes(x=Make, fill=Make))+  + geom\_bar()+  + labs(x="Make", y="Count")+  + geom\_text(stat = 'count', aes(label = ..count..), vjust = -0.5)+  + ggtitle("Countplot of Make in 0 Payment data subset")  > #Let's do the same analysis for Maximum number of insurance claims for each make type  > data\_max\_claims = data\_ds[data\_ds$Payment >=50000,]  > data\_max\_claims$Make <- factor(data\_max\_claims$Make)  > ggplot(data\_max\_claims, aes(x=Make, fill=Make))+  + geom\_bar()+  + labs(x="Make", y="Count")+  + geom\_text(stat = "Count", aes(label = ..count..), vjust=-0.05)+  + ggtitle("Countplot of Make in maxmimum Payment(>50000) Data subset")  > #first i want to check if there are outliers in Kilometer, Zone and Bonus using boxplot  > title = "Box plot for kilometer, Zone, Bonus, Make"  > boxplot\_colors <- c("red", "blue", "green", "purple")  > boxplot\_labels <- names(data\_ds)  > boxplot(data\_ds$Kilometres, data\_ds$Zone, data\_ds$Bonus, data\_ds$Make, main=title, col = boxplot\_colors)  > medians <- sapply(list(data\_ds$Kilometres, data\_ds$Zone, data\_ds$Bonus, data\_ds$Make),  + function(x) quantile(x, c(0.25, 0.5, 0.75)))  > # Add text labels for median, Q1, and Q3  > text(x = 1:4, y = medians["50%", ], labels = paste("Median: ", round(medians["50%", ], 2)), pos = 3)  > text(x = 1:4, y = medians["25%", ], labels = paste("Q1: ", round(medians["25%", ], 2)), pos = 1)  > text(x = 1:4, y = medians["75%", ], labels = paste("Q3: ", round(medians["75%", ], 2)), pos = 1)  > for (i in 1:4) {  + text(x = i, y = par("usr")[3] - 0.15, labels = boxplot\_labels[i], srt = 45, pos = 2, xpd = TRUE)  + }  > hist(data\_ds$Insured ,col='red',main="Histogram of Count Insured", breaks=20, xlab= "Insured")  > hist(data\_ds$Claims,col='blue',breaks=20, main="Histogram of Count Claims", xlab="Claims")  > hist(data\_ds$Payment, col = 'green',breaks=20, main="Histogram of Payment", xlab = "Payment")  > plot(density(data\_ds$Insured), col = 'red',main = "Density Plot of Insured", xlab= "Insured")  > plot(density(data\_ds$Claims), col = 'blue',main = "Density Plot of Claims", xlab="Claims")  > plot(density(data\_ds$Payment), col = 'green', main = "Density plot of Payment",xlab = "Payment" )  > #lets study the top 3 and bottom 3 based on Insured, Claims and Payment  > #sort the dataset based on Insured amount and check top 3  > sort\_Insured = data\_ds[order(data\_ds$Insured),]  > tail(sort\_Insured, 3)  Kilometres Zone Bonus Make Insured Claims Payment  1132 3 4 7 9 79614.96 2548 13203616  691 2 4 7 9 121293.07 3338 18245026  252 1 4 7 9 127687.27 2894 15540162  > head(sort\_Insured,3)  Kilometres Zone Bonus Make Insured Claims Payment  1724 4 7 4 4 0.01 0 0  2154 5 7 4 7 0.03 0 0  2133 5 7 1 2 0.04 0 0  > #sort the dataset based on Claims and check top 3  > sort\_Claims = data\_ds[order(data\_ds$Claims),]  > tail(sort\_Claims, 3)  Kilometres Zone Bonus Make Insured Claims Payment  1132 3 4 7 9 79614.96 2548 13203616  252 1 4 7 9 127687.27 2894 15540162  691 2 4 7 9 121293.07 3338 18245026  > head(sort\_Claims,3)  Kilometres Zone Bonus Make Insured Claims Payment  35 1 1 4 8 12.74 0 0  44 1 1 5 8 18.21 0 0  98 1 2 4 8 18.35 0 0  > #sort the dataset based on Payment and check top 3  > sort\_payment = data\_ds[order(data\_ds$Payment),]  > tail(sort\_payment, 3)  Kilometres Zone Bonus Make Insured Claims Payment  1132 3 4 7 9 79614.96 2548 13203616  252 1 4 7 9 127687.27 2894 15540162  691 2 4 7 9 121293.07 3338 18245026  > head(sort\_payment, 3)  Kilometres Zone Bonus Make Insured Claims Payment  35 1 1 4 8 12.74 0 0  44 1 1 5 8 18.21 0 0  98 1 2 4 8 18.35 0 0  > sprintf("The correlation of Distance and payment is: %.3f",cor(data\_ds$Kilometres, data\_ds$Payment))  [1] "The correlation of Distance and payment is: -0.121"  > sprintf("The correlation of Zone and PAyment is: %.3f",cor(data\_ds$Zone, data\_ds$Payment))  [1] "The correlation of Zone and PAyment is: -0.103"  > sprintf("The correlation of Bonus with Payment is: %.3f",cor(data\_ds$Bonus, data\_ds$Payment))  [1] "The correlation of Bonus with Payment is: 0.118"  > sprintf("The correlation of Make with Payment is: %.3f",cor(data\_ds$Make, data\_ds$Payment))  [1] "The correlation of Make with Payment is: 0.244"  > sprintf("The correlation of Insured vs Payment is: %.3f",cor(data\_ds$Insured, data\_ds$Payment))  [1] "The correlation of Insured vs Payment is: 0.933"  > sprintf("The correlation of Claims Vs Payment is %.3f",cor(data\_ds$Claims, data\_ds$Payment))  [1] "The correlation of Claims Vs Payment is 0.995"  > #plot the correlation (scatter plot - using geom\_point) between Insured years and Payment  > ggplot(data\_ds, aes(x=Insured, y=Payment, col = Insured))+  + labs(x="insured years", y="Total Payment", title="correlation of payment and insured years")+  + geom\_point()  > #plot the correlation between Claims and Payment using Scatter plot  > ggplot(data\_ds, aes(x=Claims, y=Payment, col=Claims))+  + labs(x="Number of Claims", y="Total Payment", title = "Correlation of Payment and No pf claims")+  + geom\_point()  > #SOLUTION  > #to understand the impact of any of these variables on payment we should do a multivariate analysis  > #first let's build a model only for KM, Zone, Bonus and MAke, we know this has very less correlation  > # just for understanding purpose , let's see how these variables impact payment  > regression\_payment<-lm(Payment~Kilometres+Zone+Bonus+Make,data=data\_ds)  > summary(regression\_payment)  Call:  lm(formula = Payment ~ Kilometres + Zone + Bonus + Make, data = data\_ds)  Residuals:  Min 1Q Median 3Q Max  -1045662 -348581 -134493 110070 17338804  Coefficients:  Estimate Std. Error t value Pr(>|t|)  (Intercept) 13221 85915 0.154 0.878  Kilometres -88407 14678 -6.023 2.00e-09 \*\*\*  Zone -53480 10409 -5.138 3.03e-07 \*\*\*  Bonus 60831 10348 5.879 4.77e-09 \*\*\*  Make 95324 8001 11.913 < 2e-16 \*\*\*  ---  Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1  Residual standard error: 966700 on 2177 degrees of freedom  Multiple R-squared: 0.09871, Adjusted R-squared: 0.09705  F-statistic: 59.61 on 4 and 2177 DF, p-value: < 2.2e-16  > #let's build another model with insured and claims'  > reg\_model = lm(Payment~Insured+Claims+Kilometres+Zone+Bonus+Make, data= data\_ds)  > summary(reg\_model)  Call:  lm(formula = Payment ~ Insured + Claims + Kilometres + Zone +  Bonus + Make, data = data\_ds)  Residuals:  Min 1Q Median 3Q Max  -806775 -16943 -6321 11528 847015  Coefficients:  Estimate Std. Error t value Pr(>|t|)  (Intercept) -2.173e+04 6.338e+03 -3.429 0.000617 \*\*\*  Insured 2.788e+01 6.652e-01 41.913 < 2e-16 \*\*\*  Claims 4.316e+03 1.895e+01 227.793 < 2e-16 \*\*\*  Kilometres 4.769e+03 1.086e+03 4.392 1.18e-05 \*\*\*  Zone 2.323e+03 7.735e+02 3.003 0.002703 \*\*  Bonus 1.183e+03 7.737e+02 1.529 0.126462  Make -7.543e+02 6.107e+02 -1.235 0.216917  ---  Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1  Residual standard error: 70830 on 2175 degrees of freedom  Multiple R-squared: 0.9952, Adjusted R-squared: 0.9952  F-statistic: 7.462e+04 on 6 and 2175 DF, p-value: < 2.2e-16  > aggregate\_data = aggregate(cbind(Insured, Claims, Payment)~Kilometres+Zone+Bonus, data=data\_ds, FUN=mean)  > ggplot(aggregate\_data, aes(x=Kilometres, y=Payment))+  + geom\_point(aes(color=Zone, size=Bonus))+  + #geom\_text( aes(label = Zone), hjust=2)+  + geom\_text(aes(label = Zone), hjust = -1, vjust = 0.5, position = position\_nudge(x = -0.1)) +  + geom\_text(aes(label = Bonus), hjust = 2, vjust = 0.5, position = position\_nudge(x = 0.1)) +  + labs(title="Payment vs. Kilometer by Location and Bonus", x = "Kilometer", y = "Payment")  > pois\_model = glm(Claims~Kilometres+Zone+Bonus+Make+Insured, data= data\_ds, family=poisson)  > summary(pois\_model)  Call:  glm(formula = Claims ~ Kilometres + Zone + Bonus + Make + Insured,  family = poisson, data = data\_ds)  Coefficients:  Estimate Std. Error z value Pr(>|z|)  (Intercept) 2.656e+00 1.638e-02 162.17 <2e-16 \*\*\*  Kilometres -2.692e-01 2.415e-03 -111.47 <2e-16 \*\*\*  Zone -2.516e-01 1.744e-03 -144.27 <2e-16 \*\*\*  Bonus 7.856e-02 1.656e-03 47.46 <2e-16 \*\*\*  Make 3.765e-01 1.659e-03 226.95 <2e-16 \*\*\*  Insured 2.837e-05 9.527e-08 297.82 <2e-16 \*\*\*  ---  Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1  (Dispersion parameter for poisson family taken to be 1)  Null deviance: 435505 on 2181 degrees of freedom  Residual deviance: 178602 on 2176 degrees of freedom  AIC: 186252  Number of Fisher Scoring iterations: 6  > plot(pois\_model)  Hit <Return> to see next plot: |
|  |
| |  | | --- | | Hit <Return> to see next plot: | |