ARLINGTON ACCIDENT ANALYSIS AT INTERSECTIONS

By

Group # 2

Kritin Yanamala Reddy

Sushma Chowdary Mallampati

Rishika Ravi Kukuttala

This Report has been Submitted

in Partial Fulfillment of the Requirements for

DASC 5302 – Introduction to Probability and Statistics Course

at the University of Texas at Arlington

April 24, 2023

Introduction

Our project is based on the number of accidents occurring at intersections in and around UTA and Arlington. We have collected the data manually by visiting various intersection points at different locations and times of the day, as the data collected would vary at areas around UTA and other locations.

We are focused on calculating the descriptive statistics for the continuous variable and the grouping variables, the continuous variable being the number of vehicles passing through an intersection while taking into consideration various grouping variables like the time of day, the type of intersection, and the number of lanes to determine the number of accidents that happen at each intersection in a specific period of time.

For Dataset 2, we calculate the inter arrival times for each event by using the data which was manually collected. We considered variables such as Start time and the End time by keeping in mind the continuous variable which is the number of pedestrians crossing an intersection at any given minute.

We have documented and subjected both the datasets to cleaning by checking for outliers and null values. We have also summarized the collected data and have performed various descriptive statistics on them. Lastly we have plotted the histogram and have also found if the distribution is normal or exponential.

Data Cleaning and Processing

In Dataset 1, the data has been collected manually by keeping in mind various grouping variables, we considered the continuous random variable as the number of vehicles crossing at a given time in a day with grouping variables being the time of day, number of lanes and intersection type. We have collected the data for the number of accidents occurring at each intersection considering all the various variables.

In Dataset 2, we calculate the inter arrival times for each event by using the data which was manually collected. We considered variables such as Start time and the End time by keeping in mind the continuous variable which is the number of pedestrians crossing an intersection at any given minute.

For both Dataset 1 and Dataset 2, we cleaned the data by first checking for null values in the datasets. Cleaning isn't required as much for the datasets as the data was manually collected. If there were any null values present, we filled them or dropped them accordingly.

We created a boxplot to check for outliers in the initial data frame created, then we dropped all these outliers from the dataframe for both the datasets.
The analysis was done on the new data frame created without the outliers.

We focused on calculating the descriptive statistics such as the mean, median, mode etc., we have also calculated the geometric mean and exponential mean for the grouping variables.

We have done the Goodness of fit for both the datasets using functions like pnorm and pexp to find class probabilities. Now, to perform Hypothesis testing for T- test we use t.test function to find the p-value and for F-test we create a linear model by using lm and anova to find the p-values.

Results

Descriptive Statistics of Dataset 1

Figure 1: Boxplot for Dataset 1 with outliers

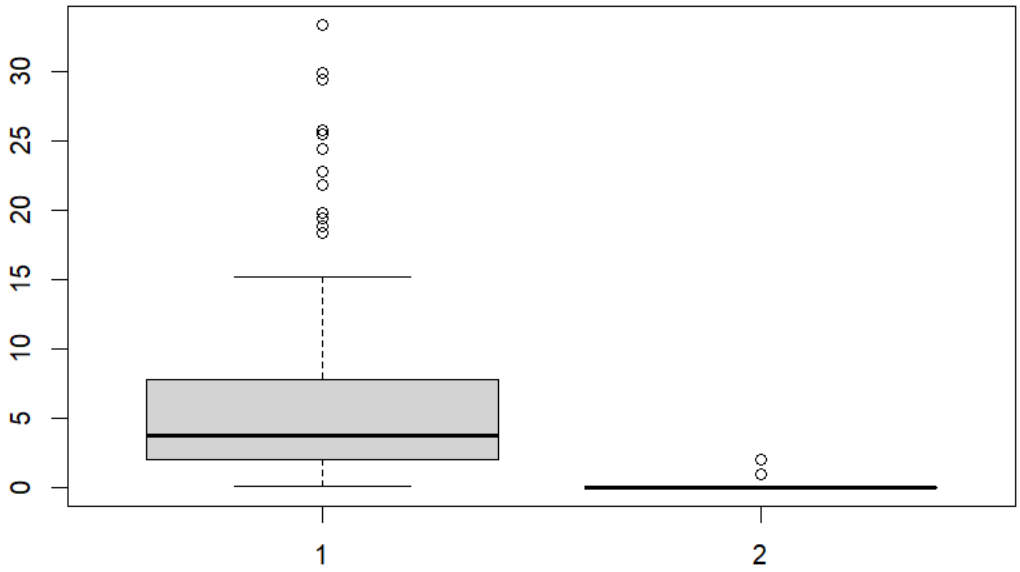


Figure 2: Boxplot for Dataset 1 without outliers

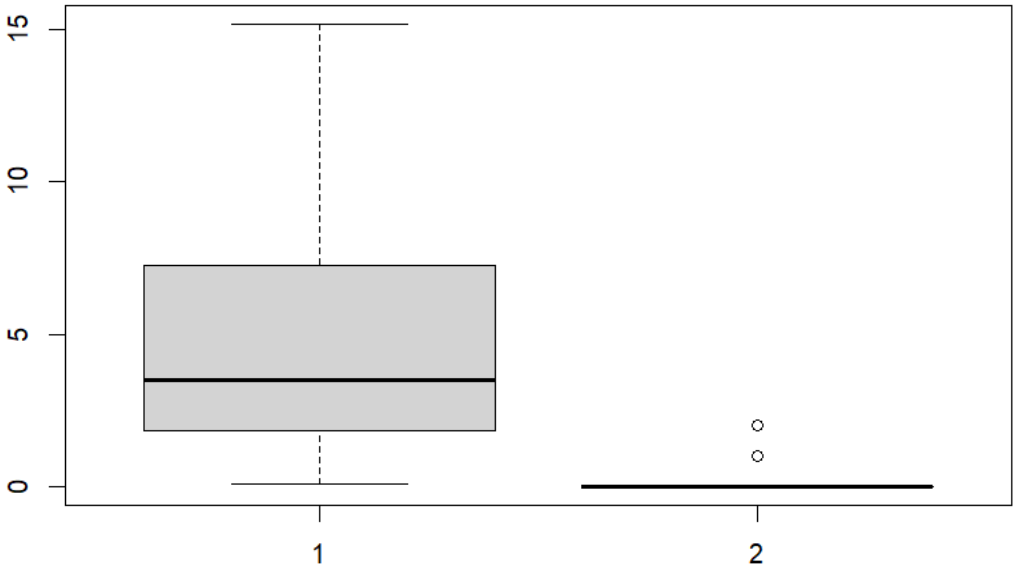
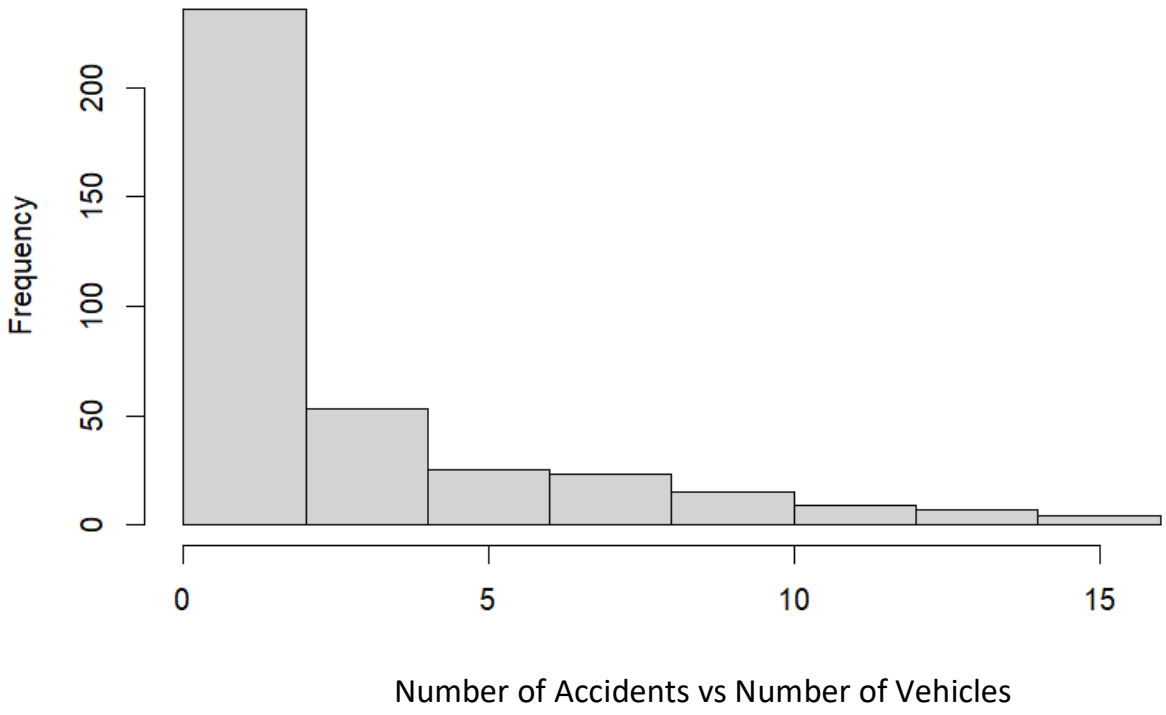


Table 1: Descriptive Statistics of Dataset 1

Statistic	Overall	Number of accidents	Time of the Day			Number of lanes				Intersection Type	
			Morning	Afternoon	Evening	3	4	6	8	All way Stop	Signalised
Sample size	186	186	61	78	47	35	81	46	24	97	89
Mean	4.741	0.1	5.69	4.20	4.40	1.79	3.14	6.87	10.4	2.50	7.18
Median	3.510	0	4.83	3.31	2.95	1.6	2.78	6.96	10.4	2.2	7.27
Mode	3.2	0	4.27,4.89,3.2,12.9,7	1.52,4.58,2.07,3.7,3.2,3,1.85	1.2	3.2	3.6,4.27,1.2,3.7,3,1.85	8.8, 4.89,4.58	0	3.2	9.57,8.8, 4.89,4.58,12.9,7, 2.93,3,7.27
Variance	13.302	0.114	16.2	9.88	141	1.26	4.19	9.31	8.16	2.76	13.4
Standard Deviation	3.65	0.337	4.03	3.14	3.74	1.12	2.05	3.05	2.86	1.66	3.66
Coefficient of Variance	3.65	0.337	4.03	3.14	3.74	1.12	2.05	3.05	2.86	1.66	3.66
Exponential Mean	3.605	0	4.24	3.14	2.79	1.37	2.55	6.16	9.92	1.97	6.02
Range	15.08	2	14.5	14.1	12.6	4.7	9.54	13.1	10.3	8.9	14.6
Geometric mean	3.605	0	4.24	3.14	3.74	1.37	2.55	6.16	9.92	1.97	6.02

Figure 3: Frequency histogram for the overall dataset 1



Descriptive Statistics of Dataset 2

Figure 4: Boxplot for Dataset 2 with outliers

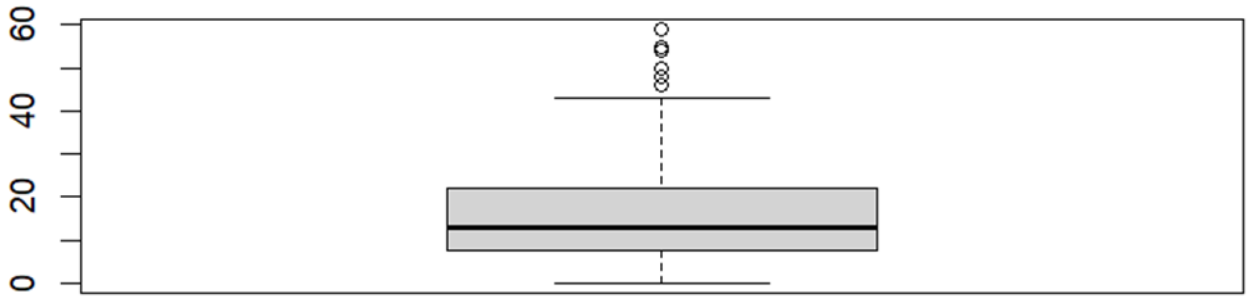


Figure 5: Boxplot for Dataset 2 without outliers

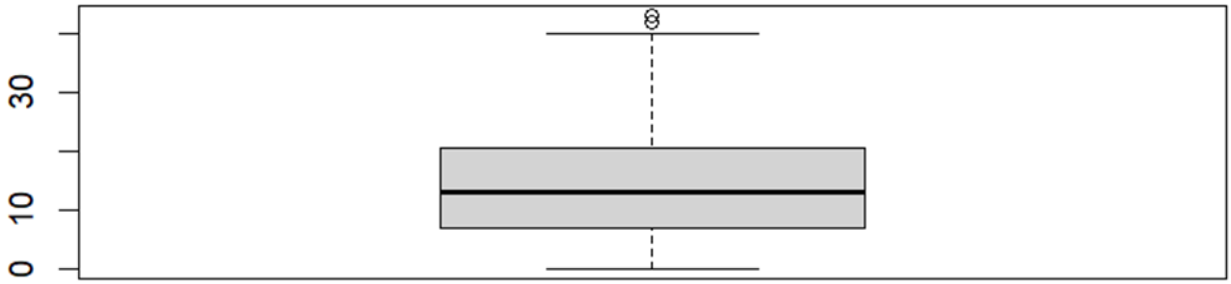


Table 2: Descriptive Statistics of Dataset 2

Statistics	No of Pedestrians	Inter arrival Time
Sample size	170	170
Mean	14.8	3.138
Median	13.0	0
Mode	8	0
Variance	105.3764	243776.3
Standard Deviation	10.2653	493.74
Coefficient of Variance	10.2653	493.74
Range	43	3000 seconds

Figure 6. Frequency histogram for the overall dataset

Goodness of Fit Tests

Table 3: Goodness of fit test for Dataset 1

Interval	Frequency	Class Probability	Expected Value	Chi-Square
0<x<2	236	0.2262017	42.07351	893.8519
2<x<4	53	0.19133467	35.96249	8.07165
4<x<6	25	0.2155234	40.08735	5.678303
6<x<8	23	0.1791822	33.32788	3.200479
8<x<10	15	0.1111015	20.66489	1.552921
10<x<12	9	0.05137177	9.555148	0.03225379
x<16	11	0.01872523	3.4821386	88.488706
Total	**372**	**0.9954484**	**185.1534**	**993.8762**

This Table shows Goodness of fit Test for Dataset 1. Since the frequencies in the intervals x<16 was 4, we merged the previous row (i.e 12<x<14 and x<16) to form one class interval x<16, all the values along these rows have also been merged.

H_0 : The Dataset follows a Normal Distribution

H_1 : The Dataset does not follow a Normal Distribution

From, the above table we can see that,

χ^2 = **993.8762**

$\chi^2_{\alpha,k-1}$ = **12.59159**

Since $\chi^2 > \chi^2_{\alpha,k-1}$, we reject null hypothesis H_0 this also shows this is a strong conclusion.

Hence, we can say that our dataset follows exponential distribution.

Goodness of Fit Test

Table 4: Goodness of fit test for Dataset 2

Class Intervals	Frequency	Class Probabilities	Expected Value	Chi Square
0<x<5	32	0.2709297	46.0585	4.290861
5<x<10	39	0.1975268	33.57955	0.8749737
10<x<15	27	0.1440109	24.48186	0.2590101
15<x<20	27	1.05E-01	17.84899	4.691631
20<x<25	12	0.07654807	13.01317	0.07888298
25<x<30	14	0.05580893	9.487517	2.146241
30<x<40	10	0.0703535	11.960095	0.3373013
x<60	9	0.7144655	12.145912	2.25876061
Total	**170**	**0.99161855**	**168.575594**	**14.93766069**

This Table shows Goodness of fit Test for Dataset 2. Since the frequencies in the intervals x<60 was 1, 50<x<55 was 2, 45<x<50 was 3, 40<x< 45 was 3 so we merged these rows together to form one class interval x<60. Similarly, we merged the class intervals 30<x<35 and 35<x<40 to form a new class interval 30<x<40, all the values along these rows have also been merged accordingly.

H_0 : The Dataset follows a Exponential Distribution

H_1 : The Dataset does not follow a Exponential Distribution

From, the above table we can see that,

χ^2 = **14.9376**

$\chi^2_{\alpha,k-1}$ = **14.06714**

Since $\chi^2 > \chi^2_{\alpha,k-1}$, we reject null hypothesis H_0 this also shows this is a strong conclusion.

Hence, we can say that our dataset does not follow a exponential distribution.

Hypothesis Testing

T-test Statistic:

Here we can define the Null hypotheses first as,

H_0 : μ1 = μ2

H_1 : μ1 ≠ μ2

Looking at the table for t- tests:

Table 5 : t - test statistic table

	Values
μ1	4.20
μ2	5.69
t*	-2.3631
p-value	0.01986273
α	0.05

Here, we can see that both means aren't equal, which is μ1 ≠ μ2, we can also conclude from the table that p -value < α. Hence we are going to reject Null Hypothesis.

F- test:

This is used to check if both the variances in the grouping variable are equal. So, we define our null hypothesis as:

$H_0 : \ \sigma_1^2 = \sigma_2^2$

$H_1 : \ \sigma_1^2 \neq \sigma_2^2$

Table 6 : F- test table

	Values
$\boldsymbol{\sigma_1^2}$	9.31
$\boldsymbol{\sigma_2^2}$	8.16
F*	21.485
p-value	1.667e-05***
α	0.05

Here, we can see that both means aren't equal, which is $\sigma_1^2 \neq \sigma_2^2$, we can also conclude from the table that p -value < α. Hence we are going to reject Null Hypothesis.

Discussions

Dataset 1: From the data analyzed, we have observed that in Figure 3: Histogram of Dataset 1, the distribution is not Normal Distributed. Normally distributed data would peak at the center of the graph. Since the histogram peaks as it starts from the right as observed in the distribution, it doesn't appear to be a normally distributed data.

The **Overall size** of the dataset 1 is 186.
Mean is 144.65
Standard deviation is 131.38
Mode is 48
Median is 90

Goodness of fit in Dataset 1:

H_0 : The Dataset follows a Normal Distribution

H_1 : The Dataset does not follow a Normal Distribution

We have observed that the Chi-Square value was $\chi^2_{\alpha,k-1}$ = **12.59159** when we have a confidence interval of 95% and the critical value we have found is χ^2 = **993.8762 .**

Since $\chi^2 > \chi^2_{\alpha,k-1}$ **, we reject null hypothesis H_0 this also shows this is a weak conclusion.**

Hence, we can say that our dataset does not follow an Exponential Distribution.

Since the value is greater than the critical value **we fail to reject null hypothesis** and indicate that our data follows an Exponential Distribution.

Hypothesis Testing:

T - test Statistics

Null Hypotheses : **Ho :μ1 = μ2, H1: μ1 ≠ μ2**

We have selected our grouping variable to be Time of day and selected two levels which are morning and afternoon. Now, our null hypothesis is to test if both the mean values are

equal, to do this we report t-statistics and p-value.

So, by running the code for the following t-statistics we found out that our

t*= -2.3631

From this we have also found out our p-value to be

p-value=0.01986273

Now, using a 95% confidence interval we have α =0.05, now we check for p < α, so we reject null hypotheses**.**

p-value calculated shows it is lesser than 0.05, so we are going to **Reject Null Hypotheses.**

We can also say that this is a **weak conclusion.**

F-test Statistics:

$\boldsymbol{H_0: \sigma_1^2 = \sigma_2^2}$

$\boldsymbol{H_1: \sigma_1^2 \neq \sigma_2^2}$

We have selected our grouping variable to be the Number of lanes in this case and selected two levels which are 6 and 8 lanes.. Now, our null hypothesis is to test if both the variances are equal, to do this we have to report F-statistics and p-value.

So, by running the code for the following F-statistics we found out that our

F*= 21.485

From this we have also found out our p-value to be

p-value= 1.667e-05 ***

Now, using a 95% confidence interval we have α =0.05, now we check for p < α, then we reject null hypotheses .

Looking at the p-value we can say that it's almost close to 0, which is less than 0.05, so we are going to **Reject Null Hypotheses.**

This also tells us that it is a **weak conclusion.**

For **Dataset 2**, we have observed that in Figure 6: Histogram of Dataset 2, the distribution

is Exponential Distribution. Since the Histogram represents an Exponential distributed data, the Distribution would peak at the start of the graph plots and gradually reduce in a curved manner as the x value increments.

The **Overall size** of the dataset 2 is

Mean is 0.54

Standard deviation is 0.38

Mode is 0.4

Median is 0.43

Goodness of fit in Dataset 2:

H_0 : The Dataset follows a Exponential Distribution

H_1 : The Dataset does not follow a Exponential Distribution

From, the above table we can see that,

We have observed that the Chi-Square value when we have a confidence interval of 95% is $\chi^2_{\alpha,k-1}$ = **14.06714** and the critical value we have found is χ^2 = **14.9376**.

Since $\chi^2 < \chi^2_{\alpha,k-1}$, we fail to reject null hypothesis H_0 , this also shows this is a weak conclusion.

Hence, we can say that our dataset follows exponential distribution.

Since the value is greater than the critical value we fail to reject null hypothesis and indicate that the data is exponential.

References

[1] Saiz, A. Z., González, C. Q., Gil, L. H., & Ruiz, D. M. (2020). *An Introduction to Data Analysis in R: Hands-on Coding, Data Mining, Visualization and Statistics from Scratch*. Springer Nature.

[2] Winter, B. (2019). *Statistics for Linguists: An Introduction Using R*. Routledge.

Appendices

Appendix A: Meeting Minutes

Meeting #1

Date:	03/08/2023
Modality:	[In-person]
Members Attended:	Kritin Yanamala Reddy, Sushma Chowdary Mallampati
Topics Discussed	Data Collection

Meeting #2

Date:	03/12/2023
Modality:	[In-person]
Members Attended:	Kritin, Yanamala Reddy, Sushma Chowdary Mallampati, Rishika Ravi Kukuttala
Topics Discussed	Data Collection

Meeting #3

Date:	03/14/2023
Modality:	[In-person]
Members Attended:	Kritin Yanamala Reddy, Sushma Chowdary Mallampati, Rishika Ravi Kukuttala
Topics Discussed	Data collection, Data cleaning

Meeting #4

Date:	03/16/2023
Modality:	[In-person]
Members Attended:	Kritin Yanamala Reddy, Sushma Chowdary Mallampati, Rishika Ravi Kukuttala
Topics Discussed	Data cleaning, Data processing

Meeting #5

Date:	03/19/2023
Modality:	[In-person]
Members Attended:	Kritin Yanamala Reddy, Sushma Chowdary Mallampati, Rishika Ravi Kukuttala
Topics Discussed	Data processing

Meeting #6

Date:	03/22/2023
Modality:	[In-person]
Members Attended:	Kritin Yanamala Reddy, Sushma Chowdary Mallampati, Rishika Ravi Kukuttala
Topics Discussed	Data analysis, Report

Meeting #7

Date:	03/18/2023
Modality:	[In-person]
Members Attended:	Kritin Yanamala Reddy, Sushma Chowdary Mallampati, Rishika Ravi Kukuttala
Topics Discussed	Data analysis, Report

Meeting #8

Date:	04/20/2023
Modality:	[In-person]
Members Attended:	Kritin Yanamala Reddy, Sushma Chowdary Mallampati, Rishika Ravi Kukuttala
Topics Discussed	Data analysis, Report

Meeting #9

Date:	04/23/2023
Modality:	[In-person]
Members Attended:	Kritin Yanamala Reddy, Sushma Chowdary Mallampati, Rishika Ravi Kukuttala
Topics Discussed	Data analysis, Report, Powerpoint presentation

Appendix B : R code

DATASET 1 - CODE:

```
#installing and importing the required packages
install.packages("psych")
library(psych)
install.packages("tidyr")
library(tidyr)
getwd()
#reading the CSV file into Rstudio and storing it in a variable
data1 <- read.csv("C:/Users/kriti/Downloads/PROJECT DATASETS - Sheet1
(2).csv", header=T, sep =",")

data2 = data.frame(data1 )#Creating a data frame
View(data2) #used to view the data frame
#using is.na function to check for any null values
is.na(data2)
#Using the function summary and describe to represent descriptive statistics
summary(data2)
describe(data2)
#create a boxplot for the entire data frame with outliers
boxplot(data2$Number.of.lanes,data2$Number.of.Vehicles,data2$Accidents)

#dropping the outliers
Q1 <- quantile(data2$Number.of.Vehicles, .25)
Q3 <- quantile(data2$Number.of.Vehicles, .75)
IQR <- IQR(data2$Number.of.Vehicles)
```

```
#only keep rows in dataframe that have values within 1.5*IQR of Q1 and Q3
ndata2 <- subset(data2, data2$Number.of.Vehicles > (Q1 - 1.5*IQR) &
data2$Number.of.Vehicles< (Q3 + 1.5*IQR))
View(ndata2)
#view row and column count of new data frame
dim(ndata2) #Orginal dataframe had 199 values after removing outliers dataframe has
187 values
#create a boxplot for the entire data frame without outliers
boxplot(ndata2$Number.of.lanes,ndata2$Number.of.Vehicles,ndata2$Accidents)

#Using the function summary and describe to represent descriptive statistics
summary(ndata2)
describe(ndata2)
#Continuous random variable (Number of Vehicles)
summary(ndata2)
describe(ndata2$Number.of.Vehicles) #Finding descriptive statistics
# Finding Coefficient of Variance using Standard Deviation and Variance
s <-
var(ndata2$Number.of.Vehicles) s
l <-
sd(ndata2$Number.of.Vehicles) l
cv <- s/l
cv
# Finding mode by creating a function
getmode <- function(v) {
  uniqv <- unique(v)
```

```
  uniqv[which.max(tabulate(match(v, uniqv)))]
}
mo <-
getmode(ndata2$Number.of.Vehicles) mo
#Exponential Mean of the continuous variable
gm1 <-
exp(mean(log(ndata2$Number.of.Vehicles))) gm1
#Geometric mean of the continuous variable
gm2 <-
geometric.mean(ndata2$Number.of.Vehicles) gm2
#install packages to do the groupby function
install.packages("dplyr")
library(dplyr)

#TIME OF DAY
#Calculate descriptive statistics of 'Time of Day' by 'Number of Vehicles'
tod <- ndata2 %>%
group_by(Time.of.day) tod
to <- tod %>%
summarise(describe(Number.of.Vehicles)) to
#calculating Variance
tov <- tod %>%
summarise(var(Number.of.Vehicles)) tov
#Calculating Standard Deviation
tos <- tod %>%
summarise(sd(Number.of.Vehicles)) tos
```

```
cv <- tos/tov # Finding Coefficient of Variance using Standard Deviation and Variance
cv
#calculate mode of 'Time of day' by 'Number of Vehicles'
find_mode <- function(x) {
  u <- unique(x)
  tab <- tabulate(match(x, u))
  u[tab == max(tab)]
}
data2 %>%
  group_by(Time.of.day)
  %>%
  reframe(mode_points = find_mode(Number.of.Vehicles))

#Calculate the Exponential Mean for "Time of day" by 'Number of Vehicles'
ndata2 %>%
  group_by(Time.of.day)
  %>%
  summarize(geometric_mean = exp(mean(log(Number.of.Vehicles))))
  #NUMBER OF LANES
#Calculate descriptive statistics of 'Number of lanes' by 'Number of Vehicles'
nol <- ndata2 %>%
group_by(Number.of.lanes) nol
no <- nol %>%
summarise(describe(Number.of.Vehicles)) no
#Calculate the Geometric mean for "Time of day" by 'Number Of Vehicles'
ndata2 %>%
```

```
  group_by(Time.of.day)
  %>%
  reframe(j = geometric.mean(Number.of.Vehicles))
#calculate mode of 'Number of Lanes' by 'Number of Vehicles'
findmode <- function(b) {
  n <- unique(b)
  tab <- tabulate(match(b, n))
  n[tab == max(tab)]
}
ndata2 %>%
  group_by(Number.of.lanes)
  %>%
  reframe(modepoints = findmode(Number.of.Vehicles))
# Calculate the geometric mean
ndata2 %>%
group_by(Number.of.lanes) %>%
  summarize(geometric_mean = exp(mean(log(Number.of.Vehicles))))
#calculate Exponential mean
ndata2 %>%
group_by(Number.of.lanes) %>%
  summarize(geometric_mean = exp(mean(log(Number.of.Vehicles))))
#calculating Variance
tol <- nol %>%
summarise(var(Number.of.Vehicles)) tol
#Calculating Standard Deviation
tols <- nol %>%
```

```
summarise(sd(Number.of.Vehicles)) tols

cv <- tol/tols # Finding Coefficient of Variance using Standard Deviation and Variance
cv
 #INTERSECTION TYPE
 #Calculate descriptive statistics of 'Intersection Type' by 'Number of Vehicles'
it <- ndata2 %>%
group_by(Intersection.Type) it
t <- it %>%
summarise(describe(Number.of.Vehicles)) t
#Calculate mode of 'Intersection Type' by 'Number of Vehicles'
finddmode <- function(k) {
  ns <- unique(k)
  tab <- tabulate(match(k, ns))
  ns[tab == max(tab)]
}
ndata2 %>%
  group_by(Intersection.Type)
  %>%
  reframe(mode_points = finddmode(Number.of.Vehicles))
# Calculate the geometric mean of 'Intersection Type' by 'Number of Vehicles'
ndata2 %>%
  group_by(Intersection.Type)
  %>%
  reframe(geo_mean = geometric.mean(Number.of.Vehicles))

#calculating Variance
```

```
itv <- it %>%
summarise(var(Number.of.Vehicles)) itv
#Calculating Standard Deviation
its <- it %>%
summarise(sd(Number.of.Vehicles)) its
cv <- itv/its # Finding Coefficient of Variance using Standard Deviation and Variance
cv
#calculate Exponential mean
ndata2 %>%
group_by(Intersection.Type) %>%
  summarize(expo_mean = exp(mean(log(Number.of.Vehicles))))

# Calculate the geometric mean
ndata2 %>%
group_by(Intersection.Type) %>%
  summarize(expo_mean = exp(mean(log(Number.of.Vehicles))))

#ACCIDENTS
#Calculating the Descriptive Statistics for Number of Accidents using Summary and describe
summary(ndata2$Accidents)
describe(ndata2$Accidents)
#Calculating Variance
s <- var(ndata2$Accidents)
#Calculating Standard Deviation
l <- sd(ndata2$Accidents)
l
cv <- s/l # Finding Coefficient of Variance using Standard Deviation and Variance
```

```
cv
# Finding mode by creating a function
getmode <- function(v) {
  hash
  uniqv <- unique(v)
  uniqv[which.max(tabulate(match(v, uniqv)))]
}
mo <- getmode(ndata2$Accidents)
mo
#Exponential Mean of the continuous variable
gm1 <- exp(mean(log(ndata2$Accidents)))
gm1
#Geometric mean of the continuous variable
gm2 <- geometric.mean(ndata2$Accidents)
gm2
# Create the histogram for entire Dataset 1
histo = hist(c(ndata2$Accidents,ndata2$Num.of.Vehicles))
histo
counts <- histo$counts
breaks <- histo$breaks
samplesMean <- mean(ndata2$Num.of.Vehicles)
samplesSd <- sd(ndata2$Num.of.Vehicles)
prev <- 0
cp <- list()
for(i in 1:(length(breaks)-1)){
  low <- breaks[i]
```

```
  high <- breaks[i+1]
  classProb <- pnorm(high,mean = samplesMean, sd =samplesSd)
  if(i==(length(breaks)-1)){
    cp <- append(cp, 1-classProb)
  }
  else
    cp <- append(cp, classProb-prev)
  prev <- classProb
}
cp
sum(unlist(cp))
# perform chi-square goodness-of-fit test
#expected <- rep(length(ndata2$Num.of.Vehicles)/length(cp), length(cp))
exp <- list()
for (i in cp){
  exp <- append(exp, i*length(ndata2$Num.of.Vehicles))
}
exp
chilist <- list()
for (i in 1:length(exp)){
  #print(counts[i])
  #print(exp[i])
  e[i] <- unlist(exp[i])
  chilist <- append(chilist,(counts[i] - e[i])^2/e[i])
}
chilist
```

```
chisq <- sum((counts - e)^2/ e)
df <- length(ndata2$Num.of.Vehicles)-1
value <- qchisq(0.05, df, lower.tail= FALSE)
value
#p-values
result <- t.test(Num.of.Vehicles ~ Time.of.day, data = ndata2 , subset = Time.of.day %in%
c("Morning", "Afternoon"))
p_values <- result$p.value
p_values
full_model <- lm(Num.of.Vehicles ~ Number.of.lanes , data = ndata2, subset = Number.of.lanes %in%
c("6", "8"))
# Fit a reduced model with only the intercept
reduced_model <- lm(Num.of.Vehicles ~ 1, data = ndata2 , subset = Number.of.lanes %in% c("6",
"8"))
# Compare the fit of the two models using an F-test
result2 <- anova(reduced_model, full_model)
result2
# Extract the p-value for the F-test
p_value2 <- result2["Pr(>F)"]
# Print the p-value
p_value2
```

DATASET 2 - CODE:

```
#Installing required
packages
install.packages("psych")
library(psych)
```

```
install.packages("lubridate")
library(lubridate)
install.packages("dplyr")
library(dplyr)
install.packages("hms")
library(hms)
install.packages("tidyr")
library(tidyr)

#reading the CSV file into Rstudio and storing it in a variable
data1 <- read.csv("C:/Users/sushm/Downloads/PROJECT DATASETS 2 - Sheet2
(1).csv", header=T, sep=",")
data2 = data.frame(data1) #Creating a data frame
View(data2) #used to view the data frame
  #Checking for null values
is.na(data3)
#Using the function summary and describe to represent descriptive statistics
summary(data2)
describe(data2)
#create a boxplot for the Number of pedestrians
boxplot(data2$Pedestrians)
#dropping the outliers
Q1 <- quantile(data2$Pedestrians, .25)
Q3 <- quantile(data2$Pedestrians, .75)
IQR <- IQR(data2$Pedestrians)
#only keep rows in dataframe that have values within 1.5*IQR of Q1 and Q3
```

```
ndata2 <- subset(data2, data2$Pedestrians > (Q1 - 1.5*IQR) & data2$Pedestrians< (Q3 +
1.5*IQR))
View(ndata2)
 boxplot(ndata2$Pedestrians)
#view row and column count of new data frame
dim(ndata2) #Orginal dataframe had 175 values after removing outliers dataframe has 170 values
#Finding descriptive statistics for pedestrians
describe(ndata2$Pedestrians)
s <-  var(ndata2$Pedestrians)
s
l <- sd(ndata2$Pedestrians)
l
cv <- s/l # Finding Coefficient of Variance using Standard Deviation and Variance
cv
# Finding mode by creating a function
getmode <- function(v) {
  uniqv <- unique(v)
  uniqv[which.max(tabulate(match(v, uniqv)))]
}
mo <- getmode(ndata2$Pedestrians)
mo

#Boxplot for Number of Pedestrains
boxplot(ndata2$Pedestrians)
#sorting the start time
new <-data3[order(data3$Start.Time),]
```

```
new
df <- new
View(df)
#Calculating Inter arrival time
Inter_arrival_time <- strptime(df$Start.Time, "%H:%M") -
lag(strptime(df$Start.Time, "%H:%M"))
b <- as.numeric(Inter_arrival_time)
b
#Combining the calculated inter arrival column to main dataframe
df1 <- cbind(df, b)
df1
#checking for na values and droping them
df1 <- df1 %>% drop_na()
#Finding descriptive statistics for Inter arrival time
describe(Inter_arrival_time)
mean(df1$b)
median(df1$b)
a <- var(df1$b)
a
b <- sd(df1$b)
b
v <- a/b # Finding Coefficient of Variance using Standard Deviation ad Variance
v
df <- new
View(df)
#Calculating Inter arrival time
```

```
Inter_arrival_time <- strptime(df$Start.Time, "%H:%M") -
lag(strptime(df$Start.Time, "%H:%M"))
b <- as.numeric(Inter_arrival_time)
b
#Combining the calculated inter arrival column to main dataframe
df1 <- cbind(df, b)
df1
#checking for na values and droping them
df1 <- df1 %>% drop_na()
#Finding descriptive statistics for Inter arrival time
describe(Inter_arrival_time)
mean(df1$b)
median(df1$b)
a <- var(df1$b)
a
b <- sd(df1$b)
b
v <- a/b # Finding Coefficient of Variance using Standard Deviation ad Variance
v
# Finding mode by creating a function
getmode <- function(p) {
  uniq <- unique(p)
  uniq[which.max(tabulate(match(p, uniq)))]
}
m <- getmode(df1$b)
```

```
m

#Boxplot for Inter arrival time

boxplot(df1$b)

#Histogram for number of Pedestrians
hist(c(df1$b,df1$Pedestrians))
#Goodness-of-Fit Calculations
samples <- ndata2$Number.of.Pedestrians[1:nrow(ndata2)]
samples
a <- mean(samples)
a
s <- sd(samples)
s
histo = hist(c(ndata2$Number.of.Pedestrians))
histo
count <- histo$counts
# calculate the class probabilities
classpro <- list()
classpro <- append(classpro, pexp(5,rate = 0.06319702))
classpro <- append(classpro, (pexp (10,  rate = 0.06319702) - pexp(5, rate = 0.06319702)))
classpro <- append(classpro, (pexp (15,  rate = 0.06319702) - pexp(10, rate = 0.06319702)))
classpro <- append(classpro, (pexp (20,  rate = 0.06319702) - pexp(15, rate = 0.06319702)))
classpro <- append(classpro, (pexp (25,  rate = 0.06319702) - pexp(20, rate = 0.06319702)))
classpro <- append(classpro, (pexp (30,  rate = 0.06319702) - pexp(25, rate = 0.06319702)))
classpro <- append(classpro, (pexp (35,  rate = 0.06319702) - pexp(30, rate = 0.06319702)))
classpro <- append(classpro, (pexp (40,  rate = 0.06319702) - pexp(35, rate = 0.06319702)))
classpro <- append(classpro, (pexp (45,  rate = 0.06319702) - pexp(40, rate = 0.06319702)))
classpro <- append(classpro, (pexp (50,  rate = 0.06319702) - pexp(45, rate = 0.06319702)))
classpro <- append(classpro, (pexp (55,  rate = 0.06319702) - pexp(50, rate = 0.06319702)))
classpro <- append(classpro, (1 - pexp(60, rate = 0.06319702)))
classpro
#Calculating the expected values
expected_freq <- list()
for (i in classpro){
```

```
  expected_freq <- append(expected_freq,i*length(samples))
}
expected_freq
#Calculating Chi-Squared
chiList <- list()
for(i in 1:length(expected_freq)){
  a[i] <- unlist(expected_freq[i])
  chiList <- append(chiList, (count[i] - a[i])^2 / a[i])
}
chiList
chisq <- sum((count - a)^2 / a)
df <- length(ndata2$Number.of.Pedestrians)-1
df
n <- length(data2$Pedestrians)
n
# Calculate test statistic and p-value
test_statistic <- sum((count - a)^2 / a)
test_statistic
df = length(ndata2$Number.of.Pedestrians) - 1
p_value <- qchisq(0.05, df)
```