****

**Prepared By:** Sandeep Singh

**Student ID:** C0727422

**Subject:** 2018S-T2 BDM 1053 - Big Data Algorithms and Statistics 01

**Title:** Assignment 8

**Week:** 23 July - 29 July

**Phone:** +1 705-977-1868

**Email (Internal):** c0727422@mylambton.ca

**Email (External):** official.sandeep93@gmail.com

**Submitted To:** Mr. [Pedram Habibi](https://moodle.queenscollege.ca/moodle/user/view.php?id=370&course=1590)

**Submission Date:** 02 Aug, 2018

**Contents**

[Explanation 3](#_Toc520983145)

[Pre-requisites 3](#_Toc520983146)

[Question 1 3](#_Toc520983147)

# Explanation

# Pre-requisites

Before performing this exercise, the following libraries must be installed in R:

* corrgram
* cluster

# Question 1

***Analyze the information given in the following ‘Insurance Policy dataset’ to create clusters of persons falling in the same type***

***The description of the attributes in the dataset are as follows:***

***1.    Premium Paid –Amount paid by the person***

***2.    Age –Indicates the age of the person***

***3.    Days to renew –Days remaining to renew the policy***

***4.    Claims made –Indicates the claims already made by the person***

***5.    Income –Net income of the person***

We begin with setting the working directory and importing the csv data into dataframe.

setwd("D:/Docs/Sandeep/Colleges/Lambton/Term 2/2018S-T2 BDM 1053 - Big Data Algorithms and Statistics 01/Assignments/08 Assignment 8/")

df = read.csv("Insurance Policy Dataset.csv")

#View(df)

We find the fields that are closely related using corrgram() method.

library(corrgram)

corrgram(df)

After finding the closely related fields, we plot the graph.

#plot(df['Premiums.Paid'],df['Income'])

plot(Premiums.Paid ~ Income, df , main = "Premiums Paid vs Income")

We create 5 clusters for closely related points and plot the graph.

kc = kmeans(df,5)

plot(Premiums.Paid ~ Income, df, col= kc$cluster, main = "Premiums Paid vs Income")

Next we plot cluster for complete dataframe

wss = NULL

for(i in 1:15)

{

set.seed(140)

kmeans\_temp =kmeans(df,centers = i)

wss[i]= sum(kmeans\_temp$withinss)

}

We plot elbow graph to check the value of K.

plot(1:15,wss,type="b",xlab="Number of clusters",ylab = "wss")

set.seed(140)

kmeans\_Model =kmeans(df,5)

str(kmeans\_temp)

Lastly, we end with plotting the cluster for our dataframe.

library(cluster)

clusplot(x =df,clus =kmeans\_Model$cluster,color = T ,labels =2,lines =0,

main="Clustering",sub = "4 Cluster Model")

\*\*\*\*\*