**NAME: SANDEEP PATIL**

**Email: saashasandeep@gmail.com**

**DATA ANALYTICS WITH R, EXCEL and TABLEAU**

**Session 10 – Assignment – 10.1**

#1. Read the file in Zip format and get it into R.

forecasturl = paste('<https://archive.ics.uci.edu/ml/machine-learning-databases/00360/>',

                    'AirQualityUCI.zip', sep='')

# create a temporary directory

td = tempdir()

# create the placeholder file

tf = tempfile(tmpdir=td, fileext=".zip")

# download into the placeholder file

download.file(forecasturl, tf)

# get the name of the first file in the zip archive

fname = unzip(tf, list=TRUE)$Name[1]

fname

# unzip the file to the temporary directory

unzip(tf, files=fname, exdir=td, overwrite=TRUE)

# fpath is the full path to the extracted file

fpath = file.path(td, fname)

fpath

airquality = read.csv(fpath,sep = ";")

View(airquality)

#2. Create Univariate for all the columns.

#Univariate analysis is the simplest form of analyzing data. "Uni" means "one",

#so in other words your data has only one variable

#we can do univariate analysis by this command too

library(psych)

summary(airquality)

describe(airquality)

#or visually

library(purrr)

library(tidyr)

library(ggplot2)

airquality %>%

  keep(is.numeric) %>%

  gather() %>%

  ggplot(aes(value)) +

  facet\_wrap(~ key,scales = "free") +

  geom\_histogram()

#3. Check for missing values in all columns.

#with the help of summary function we can find which variable has how many NA value

#or check for missing values

summary(airquality)

#thus  [PT08.S1.CO](http://pt08.s1.co/" \t "_blank).,[NMHC.GT](http://nmhc.gt/)., PT08.S2.NMHC. , NOx.GT. , ...... NA=114  has missing values

#4. Impute the missing values using appropriate methods.

#lets see the structure of airquality first

str(airquality)

library(mice)

md.pattern(airquality)

#visualizing

library(VIM)

mice\_plot <- aggr(airquality, col=c('navyblue','yellow'),

                  numbers=TRUE, sortVars=TRUE,

                  labels=names(airquality), cex.axis=.7,

                  gap=3, ylab=c("Missing data","Pattern"))

# In this case we are using predictive mean matching as imputation method

imputed\_Data <- mice(airquality, m=5, maxit = 50, method = 'pmm', seed = 500)

summary(imputed\_Data)

completeData <- complete(imputed\_Data)

View(completeData)

#5. Create bi-variate analysis for all relationships.

library(psych)

pairs.panels( airquality[,c(1,2,3,4,5,6)],

              method = "pearson", # correlation method

              hist.col = "red",

              density = TRUE,  # show density plots

              ellipses = TRUE, # show correlation ellipses

              lm=TRUE,

              main ="Bivariate Scatter plots with Pearson Correlation & Histogram"

)

#6. Test relevant hypothesis for valid relations.

#Using inbuilt dataset (airquality)

#lets see the structure first

str(airquality)

#we do paired test for continous variables

#some of test are as follows

#define the null hypothesis

#Ho: Mean of first variable - Mean of 2 variable is equal to 0

#Ha: Mean of first variable - Mean of 2 variable is not equal to 0

t.test(x=airquality$Ozone, y=airquality$Solar.R ,alternative = "two.sided",mu=0 ,paired = TRUE)

t.test(x=airquality$Temp, y=airquality$Wind ,alternative = "two.sided",mu=0 ,paired = TRUE)

t.test(x=airquality$Ozone, y=airquality$Temp ,alternative = "two.sided",mu=0 ,paired = TRUE)

t.test(x=airquality$Day, y=airquality$Solar.R ,alternative = "two.sided",mu=0 ,paired = TRUE)

#as p value of this test is <0.05 we reject the null hypo

#and accept the alternative hypothesis which says there

#Mean of 1 variable - Mean of 2 variable is not equal to 0

#thus this are some test that we performed

#7. Create cross tabulations with derived variables.

#we are using inbuilt data "airquality"

attach(airquality)

unique(Wind)

unique(Temp)

#derived variables of wind and temp

x<- cut(Wind,quantile(Wind))

x<- cut(Wind,breaks = seq(1,21,3),labels = c("wind1","wind2","wind3","wind4","wind5","wind6"))

y<- cut(Temp,quantile(Temp))

y<- cut(Temp,breaks = seq(55,100,9),labels = c("temp1","temp2","temp3","temp4","temp5"))

table(x,y)

#or like this using xtabs function

mytable<- xtabs(~x+y,data = airquality)

mytable

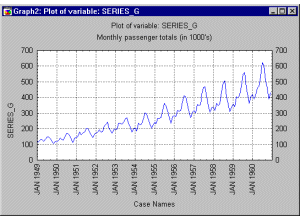
#crosstabulate

library(gmodels)

CrossTable(x,y)

8. Check for trends and patterns in time series.

Answer: Most time series patterns can be described in terms of two basic classes of components: trend and seasonality. The former represents a general systematic linear or (most often) nonlinear component that changes over time and does not repeat or at least does not repeat within the time range captured by our data (e.g., a plateau followed by a period of exponential growth). The latter may have a formally similar nature (e.g., a plateau followed by a period of exponential growth), however, it repeats itself in systematic intervals over time. Those two general classes of time series components may coexist in real-life data. For example, sales of a company can rapidly grow over years but they still follow consistent seasonal patterns (e.g., as much as 25% of yearly sales each year are made in December, whereas only 4% in August).



There are no proven "automatic" techniques to identify trend components in the time series data; however, as long as the trend is monotonous (consistently increasing or decreasing) that part of data analysis is typically not very difficult. If the time series data contain considerable error, then the first step in the process of trend identification is smoothing. Smoothing. Smoothing always involves some form of local averaging of data such that the nonsystematic components of individual observations cancel each other out. The most common technique is moving average smoothing which replaces each element of the series by either the simple or weighted average of n surrounding elements, where n is the width of the smoothing "window" (see Box & Jenkins, 1976; Velleman & Hoaglin, 1981). Medians can be used instead of means. The main advantage of median as compared to moving average smoothing is that its results are less biased by outliers (within the smoothing window). Thus, if there are outliers in the data (e.g., due to measurement errors), median smoothing typically produces smoother or at least more "reliable" curves than moving average based on the same window width. The main disadvantage of median smoothing is that in the absence of clear outliers it may produce more "jagged" curves than moving average and it does not allow for weighting. In the relatively less common cases (in time series data), when the measurement error is very large, the distance weighted least squares smoothing or negative exponentially weighted smoothing techniques can be used. All those methods will filter out the noise and convert the data into a smooth curve that is relatively unbiased by outliers (see the respective sections on each of those methods for more details). Series with relatively few and systematically distributed points can be smoothed with bicubic splines. Fitting a function. Many monotonous time series data can be adequately approximated by a linear function; if there is a clear monotonous nonlinear component, the data first need to be transformed to remove the nonlinearity. Usually a logarithmic, exponential, or (less often) polynomial function can be used.

9. Find out the most polluted time of the day and the name of the chemical compound.