

Session 10 – Correlations

Assignment - 1



Contents

[Introduction](#page3) [2](#page3)



[Objective](#page3) [2](#page3)



[Prerequisites](#page3) [2](#page3)



[Associated Data Files](#page3) [2](#page3)



[Problem Statement](#page3) [2](#page3)



[Expected Output](#page4) [3](#page4)



|  |  |
| --- | --- |
| Copyrights© 2017, AcadGild. All Rights Reserved | 1 |
|  |  |



**Introduction**



This assignment will help you to understand the key concepts learnt in this session.

**Objective**



This assignment will test your skills on Correlations in R.

**Prerequisites**



Not Applicable

**Associated Data Files**



Not Applicable

**Problem Statement**



Import dataset from the following link: [AirQuality Data Set](https://archive.ics.uci.edu/ml/datasets/Air+quality)

Perform the following written operations:

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

#Answer1

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

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

View(d)

**#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 the following command too

summary(airquality)

describe(airquality)

#or by visually

library(purrr)

library(tidyr)

library(ggplot2)

airquality

keep(is.numeric)

gather()

ggplot(aes(value)) +

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

geom\_histogram()

#or we can plot univariate individually for each variable

#hence plotting histogram

hist(airquality$Ozone ,xlab = "ozone", ylab = "Frequency",main="Histogram of ozone",col="red")

hist(airquality$Solar.R ,xlab = "solar.r", ylab = "Frequency",main="Histogram of solar.r",col="blue")

hist(airquality$Wind ,xlab = "wind", ylab = "Frequency",main="Histogram of wind",col="yellow")

hist(airquality$Temp ,xlab = "temp", ylab = "Frequency",main="Histogram of temp",col="darkblue")

hist(airquality$Month ,xlab = "month", ylab = "Frequency",main="Histogram of month",col="pink")

hist(airquality$Day ,xlab = "day", ylab = "Frequency",main="Histogram of day",col="purple")

**#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 ozone and solar.r has missing values

**#4)Impute the missing values using appropriate methods.**

#first lets see the structure of airquality

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)

completeData

#or we an alternate way

#in another way do it for variable Solar.R in airquality dataset

newair =airquality

dim(newair)

str(newair)

summary(newair)

#before imputing

hist(newair$Solar.R ,xlab = "Solar.R", ylab = "frequency",main="histogram of Solar.R",col="red")

mean(newair$Solar.R)

mean(newair$Solar.R,na.rm = T)

#imputed my mean

newair$Solar.R[is.na(newair$Solar.R)]<- mean(newair$Solar.R,na.rm = T)

#check summary after done with imputing

summary(newair)

newair$Solar.R

#visualize after imputing the variable Solar.R with the mean

#lets visualize through histogram

#after imputing

hist(newair$Solar.R ,xlab = "Solar.R", ylab = "frequency",main="histogram of Solar.R",col="red")

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

#bivariate analysis between our variables

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.**

#lets find out the structure

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.**

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)

#or

table(data$a,data$b)

**#8)Check for trends and patterns in time series.**

table(data$a,data$b)

plot.ts(datasetname)

souvenir\_decomp=decompose(souvenir\_ts)

plot(souvenir\_decomp)

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

#https://archive.ics.uci.edu/ml/datasets/Air+quality

univariateTable()

library(devtools)

## with the command

install\_github("TagTeam/Publish")

## then whenever you want to use the package and the data

library(Publish)

univariateTable(~date,time,data(airqu))

univariateTable(~age +gender + height + weight,data=Diabetes)

|  |  |
| --- | --- |
| Copyrights© 2017, AcadGild. All Rights Reserved | 2 |
|  |  |



**Expected Output**



Not Applicable

|  |  |
| --- | --- |
| Copyrights© 2017, AcadGild. All Rights Reserved | 3 |
|  |  |