**Computer Engineering Department**

**National University of Technology Islamabad, Pakistan**

**Introduction to Data Mining**

**Practice Exercise 07**

****

Name: Muhammad Sami Uddin Rafay

Roll Number: F18604013

Submitted To: Dr. Kamran Javed

Date: 07 December 2020

**Practice Exercise 07**

**Statistics for Data Mining**

**Objective:**

* Given temperature data record for five working days at NUTECH. The hourly temperature measurements for a day are stored in file. That is, for five working days we have 5 files and each file has 24 temperature measurements (that are normalized).

File1= [0.1, 0.2, 0.2, 0.2, 0.2 ,0.3, 0.4, 0.5, 0.5, 0.5, 0.5, 0.5, 0.5, 0.5, 0.6, 0.7,0.8, 0.8, 0.8, 0.9, 0.9, 0.8, 0.8, 0.9]

File2= [0.5, 0.5, 0.2, 0.5, 0.5, 0.6, 0.5, 0.6, 0.7,0.8, 0.8, 0.1, 0.2, 0.2, 0.6, 0.2 ,0.3, 0.4, 0.8, 0.9, 0.9, 0.9, 0.9, 0.9]

File3= [0.1, 0.1, 0.1, 0.5, 0.5, 0.6, 0.5, 0.6, 0.7,0.1, 0.7, 0.1, 0.2, 0.3, 0.6, 0.2 ,0.3, 0.4, 0.8, 0.1, 0.9, 0.4, 0.4, 0.4]

File4= [0.1, 0.4, 0.4, 0.4, 0.4 ,0.3, 0.4, 0.5, 0.5, 0.5, 0.5, 0.4, 0.5, 0.5, 0.6, 0.7,0.8, 0.8, 0.8, 0.4, 0.4, 0.8, 0.8, 0.9]

File5= [0.1, 0.4, 0.4, 0.4, 0.4 ,0.3, 0.4, 0.5, 0.5, 0.5, 0.5, 0.4, 0.5, 0.5, 0.6, 0.7,0.8, 0.8, 0.8, 0.4, 0.4, 0.8, 0.8, 0.9]

**Equipment/Software Required:**

* Python (Spyder 4.0 Anaconda Distribution)

**Tasks:**

a) Feature Extraction

1. Compute mean value feature from the given daily records.

2. Compute mode value feature from the given daily records.

3. Compute minimum value feature from the given daily records.

4. Compute maximum value feature from the given daily records.

5. Note: all four feature values must be written in tabular form.

b) Feature Reduction

1. Reduce the feature set to two features using low variance filter technique.

2. Write reduced feature set in tabular form and the variance of each feature at the end of the table.

c) Perform visualization and determine feature similarity

1. Using the reduced data set of two features compute correlation coefficient.

2. Draw the reduced feature set in 2-D and label each dimension.

**Code:**

**#importing necessarily Libraries**

import numpy as np

import matplotlib.pyplot as plt

import pandas as pd

import statistics

from scipy.stats import pearsonr

**# Given That**

**###############################################**

File1=np.array([0.1, 0.2, 0.2, 0.2, 0.2 ,0.3, 0.4, 0.5, 0.5, 0.5, 0.5, 0.5, 0.5, 0.5, 0.6, 0.7,0.8, 0.8, 0.8, 0.9, 0.9, 0.8, 0.8, 0.9])

File2=np.array([0.5, 0.5, 0.2, 0.5, 0.5, 0.6, 0.5, 0.6, 0.7, 0.8, 0.8, 0.1, 0.2, 0.2, 0.6, 0.2 ,0.3, 0.4, 0.8, 0.9, 0.9, 0.9, 0.9, 0.9])

File3=np.array([0.1, 0.1, 0.1, 0.5, 0.5, 0.6, 0.5, 0.6, 0.7, 0.1, 0.7, 0.1, 0.2, 0.3, 0.6, 0.2 ,0.3, 0.4, 0.8, 0.1, 0.9, 0.4, 0.4, 0.4])

File4=np.array([0.1, 0.4, 0.4, 0.4, 0.4, 0.3, 0.4, 0.5, 0.5, 0.5, 0.5, 0.4, 0.5, 0.5, 0.6, 0.7, 0.8, 0.8, 0.8, 0.4, 0.4, 0.8, 0.8, 0.9])

File5=np.array([0.1, 0.4, 0.4, 0.4, 0.4, 0.3, 0.4, 0.5, 0.5, 0.5, 0.5, 0.4, 0.5, 0.5, 0.6, 0.7, 0.8, 0.8, 0.8, 0.4, 0.4, 0.8, 0.8, 0.9])

**#a**

File= [File1, File2, File3, File4, File5]

**# Mean**

**###############################################**

def Mean(n\_num):

n = len(n\_num)

get\_sum = sum(n\_num)

mean = get\_sum / n

return mean

def MEAN(File):

l=len (File)

m=0

all\_means=np.array([])

for i in range(l):

m=Mean(File[i])

all\_means=np.append(all\_means,m)

return all\_means

All\_MEAN\_Values=MEAN(File)

print("Mean of all Features :")

print(All\_MEAN\_Values)

print("\n")

**# Mode**

**###############################################**

def Mode(n\_num):

mode=statistics.mode(n\_num)

return mode

def MODE(File):

l=len(File)

m=0

all\_modes=np.array([])

for i in range(l):

m=Mode(File[i])

all\_modes=np.append(all\_modes,m)

return all\_modes

All\_Mode\_Values=MODE(File)

print("Mode of all Features :")

print(All\_Mode\_Values)

print("\n")

**# Minimum Value**

**###############################################**

def Min(array):

Min=np.array([])

m=0

a=len(array)

for i in range(a):

m=min(array[i])

Min=np.append(Min,m)

return Min

All\_Minimum\_Values=Min(File)

print("Minimum Value of all Features :")

print(All\_Minimum\_Values)

print("\n")

**# Maximum Value**

**###############################################**

def Max(array):

Max=np.array([])

m=0

a=len(array)

for i in range(a):

m=max(array[i])

Max=np.append(Max,m)

return Max

All\_Miximum\_Values=Max(File)

print("Maximum Value of all Features :")

print(All\_Miximum\_Values)

print("\n")

# Printing all Files in Tabular Form

**###############################################**

Tabular\_Table=pd.DataFrame()

Tabular\_Table["Mean"]=All\_MEAN\_Values

Tabular\_Table["Mode"]=All\_Mode\_Values

Tabular\_Table["Min"]=All\_Minimum\_Values

Tabular\_Table["Max"]=All\_Miximum\_Values

print(Tabular\_Table)

**##############################################**

**#B**

**# Low Variance Filter**

**###############################################**

Reduced\_Features=pd.DataFrame()

def Low\_Variance\_Filter(DataFrame):

for feature in DataFrame.columns:

print(np.var(DataFrame[feature]))

print("\n")

print("Variance :")

Low\_Variance\_Filter(Tabular\_Table)

# Correlation

##############################################

print("\n")

print("Correlation of Reduced Features")

def Correlation(a,b):

Correlation,\_=pearsonr(a,b)

return Correlation

print(Correlation(Tabular\_Table["Mean"],Tabular\_Table["Mode"]))

# Reduced Features

###############################################

Reduced\_Features=pd.DataFrame()

Reduced\_Features["Mean"]=Tabular\_Table["Mean"]

Reduced\_Features["Mode"]=Tabular\_Table["Mode"]

print("\n")

print("Reduced Features :")

print(Reduced\_Features)

# Plotting

###############################################

def Plot\_Reduced\_Features(a,b):

plt.figure(1)

plt.plot(a,'g')

plt.plot(b,'r')

plt.title("Reduced Features")

plt.grid()

Plot\_Reduced\_Features(Tabular\_Table["Mode"],Tabular\_Table["Mean"])

**Output:**

**Mean of all Features :**

[0.54583333 0.5625 0.4 0.53333333 0.53333333]

**Mode of all Features :**

[0.5 0.5 0.1 0.4 0.4]

**Minimum Value of all Features :**

[0.1 0.1 0.1 0.1 0.1]

**Maximum Value of all Features :**

[0.9 0.9 0.9 0.9 0.9]

**Mean Mode Min Max**

0 0.545833 0.5 0.1 0.9

1 0.562500 0.5 0.1 0.9

2 0.400000 0.1 0.1 0.9

3 0.533333 0.4 0.1 0.9

4 0.533333 0.4 0.1 0.9

**Variance :**

0.003420833333333341

0.0216

0.0

0.0

**Correlation of Reduced Features :**

**0.9849623468943504**

**Reduced Features :**

**Mean Mode**

**0 0.545833 0.5**

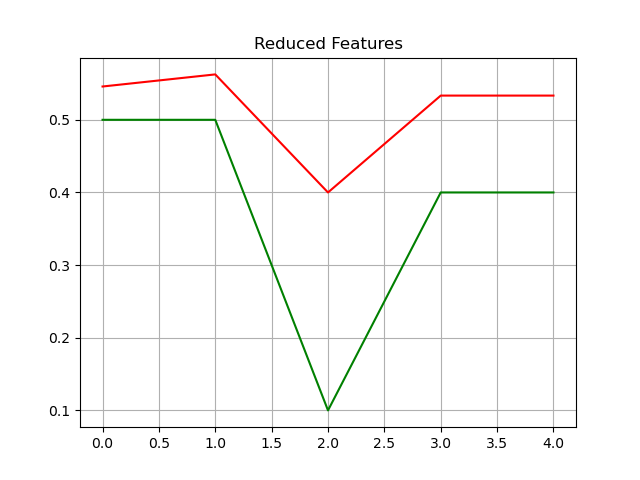
**1 0.562500 0.5**

**2 0.400000 0.1**

**3 0.533333 0.4**

**4 0.533333 0.4**

**Graphs:**

****

**Results and Discussions:**

This exercise is totally composed of Dimensionality Reduction Techniques using Low Variance Filter and High Correlated Filter and is used for the reduce the dimension of our Tabular Dataset.

**Conclusion:**

We can easily avoid curse of dimensionality with dimensionality reduction techniques.