

Tugas 4 Data Sains dan Analisis

1. Seleksi Univariate

Source Code

```
# import libraries
import pandas as pd
import numpy as np
from sklearn.feature_selection import SelectKBest
from sklearn.feature_selection import chi2

# load data
data = pd.read_csv("train.csv")
print(data.head())

# memilih data yang dibutuhkan
X = data.iloc[:,0:20] #independent columns
y = data.iloc[:,-1]   # target colum i.e price range

# menerapkan SelectKBest untuk melakukan ekstraksi
bestfeatures = SelectKBest(score_func=chi2, k=10)
fit = bestfeatures.fit(X,y)
dfscores = pd.DataFrame(fit.scores_)
dfcolumns = pd.DataFrame(X.columns)

# menggabungkan 2 dataframe
featureScores = pd.concat([dfcolumns,dfscores],axis=1)
featureScores.columns = ['Specs','Score'] #naming the dataframe columns
print(featureScores.nlargest(10,'Score')) #print 10 best features
```

Output

The screenshot displays the Visual Studio Code interface with a Jupyter Notebook open. The Explorer sidebar on the left shows the file structure, including a folder named 'TUGAS4DS' and several CSV files. The main editor area shows the code in 'tugas4datasains.py', which is a Jupyter Notebook file. The code is as follows:

```

1 # Importing Libraries
2 import pandas as pd
3 import sklearn
4
5 # Importing Data
6 data = pd.read_csv('university_towns.txt')
7
8 # Selecting Features
9 X = data.iloc[:,0:20] #independent columns
10 y = data.iloc[:, -1] # target colum i.e price range
11
12 # menerapkan SelectKBest untuk melakukan ekstraksi
13 bestfeatures = SelectKBest(score_func=chi2, k=10)
14 fit = bestfeatures.fit(X,y)
15 dfcores = pd.DataFrame(fit.scores_)
16 dfcolumns = pd.DataFrame(X.columns)
17
18 # menggabungkan 2 dataframe
19 featureScores = pd.concat([dfcolumns,dfcores],axis=1)
20 featureScores.columns = ['Specs', 'Score'] #naming the dataframe columns

```

The bottom of the image shows the terminal output, which is the first 10 rows of the DataFrame loaded from 'university_towns.txt':

```

PS C:\Users\alifi\Documents\TUGAS4DS> C:/Users/alifi/anaconda3/Scripts/activate
PS C:\Users\alifi\Documents\TUGAS4DS> conda activate base
PS C:\Users\alifi\Documents\TUGAS4DS> & C:/Users/alifi/anaconda3/python.exe c:/Users/alifi/Documents/TUGAS4DS/tugas4datasains.py
11 battery_power 17263.569536
12 px_height 14129.866576
13 px_width 9810.586750
14 mobile_wt 95.972863
15 int_memory 89.839124
16 sc_w 16.488319
17 talk_time 13.236480
18 fc 10.135166
19 sc_h 9.614878
20

```

2. Matriks Kolerasi dengan Heatmap (train.csv)

Souce Code

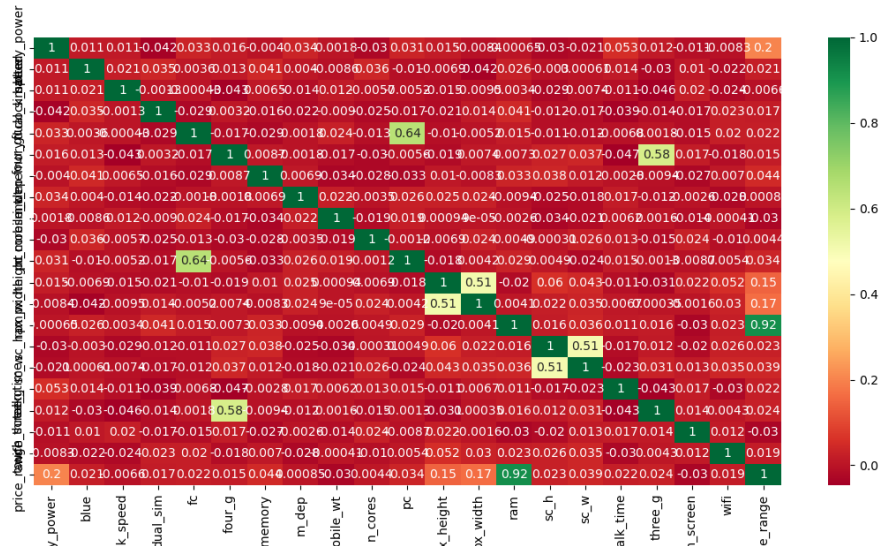
```
# import library
import pandas as pd
import numpy as np
import seaborn as sns
import plotly.graph_objects as go
import matplotlib.pyplot as plt

# memuat data
data = pd.read_csv("train.csv")
X = data.iloc[:,0:20] #independent columns
y = data.iloc[:,-1]   #target column i.e price range

# mendapatkan correlations dari setiap fitur dalam dataset
corrmat = data.corr()
top_corr_features = corrmat.index

# plot heatmap
plt.figure(figsize=(20,20))
g=sns.heatmap(data[top_corr_features].corr(),annot=True,cmap="RdYlGn")
plt.show()
```

Output



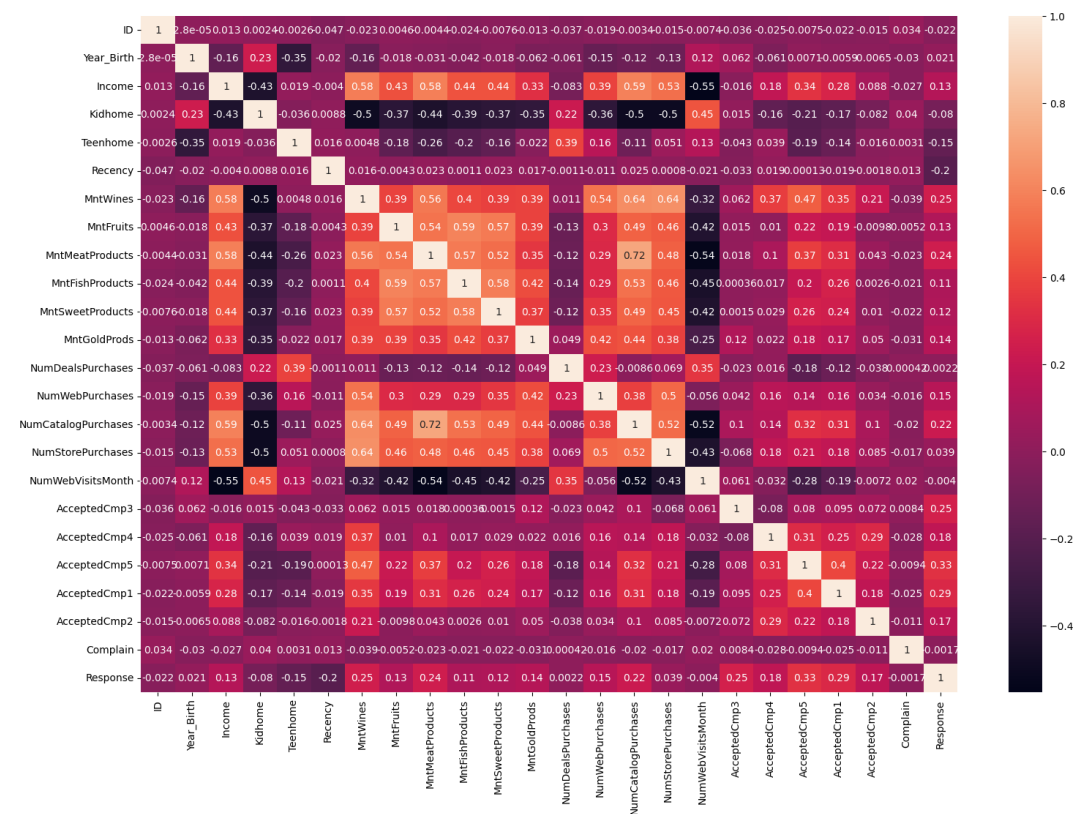
Source Code (marketing_campaign.csv)

```
# import necessary libraries
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt

# load data
data = pd.read_csv("marketing_campaign.csv")

# plot heatmap of correlation matrix
plt.figure(figsize=(18,22)) # set the size of the plot
g = sns.heatmap(data.corr(), annot=True) # plot the heatmap with annotations
plt.show() # show the plot
```

Output



Data Cleanning

Source Code

```
# import library
import pandas as pd
import numpy as np

# load data
df = pd.read_csv("laptop_price.csv")
df.head()

# periksa tipe datanya utk masing2 varibael/kolom/fitur dengan .dtypes
print(df.dtypes)

# melihat statistik dasar
df.describe()

# melihat pada lima entri pertama
print("Setelah Data di Drop\n")
print(df.head())

# memilih kolom-kolom yang akan di drop
to_drop = ['Processor', 'Warranty', 'RAM']

# melakukan drop pada data yang dipilih
df.drop(to_drop, inplace=True, axis=1)
print("Setelah Data di Drop\n")
print(df.head())
```

Output

The screenshot shows a Visual Studio Code editor with a Python script named `tugas4datacleaning.py` and its output in the terminal. The script reads a CSV file named `laptop_price.csv` and performs data cleaning operations. The terminal output displays the data before and after the drop operation.

Before Drop:

Unnamed: 0	Name	Processor	Warranty	Price rating
0	Lenovo IdeaPad S145 Core i5 10th Gen - (8 GB/1...	Intel Core i5 Processor (10th Gen)	1 Year Onsite Warranty	3.9
1	Lenovo IdeaPad Core i3 11th Gen - (8 GB/256 GB...	Intel Core i3 Processor (11th Gen)	1 Year Onsite Warranty	4.2
2	HP Pentium Quad Core - (8 GB/256 GB SSD/Window...	Intel Pentium Quad Core Processor	1 Year Onsite Warranty	4.6
3	HP 14s Core i3 11th Gen - (8 GB/256 GB SSD/Win...	Intel Core i3 Processor (11th Gen)	1 Year Onsite Warranty	4.1
4	HP 15s Athlon Dual Core - (4 GB/1 TB HDD/Win...	AMD Athlon Dual Core Processor	1 Year Onsite Warranty	4.1

After Drop:

Unnamed: 0	Name	Price rating
0	Lenovo IdeaPad S145 Core i5 10th Gen - (8 GB/1...	3.9
1	Lenovo IdeaPad Core i3 11th Gen - (8 GB/256 GB...	4.2
2	HP Pentium Quad Core - (8 GB/256 GB SSD/Window...	4.6
3	HP 14s Core i3 11th Gen - (8 GB/256 GB SSD/Win...	4.1
4	HP 15s Athlon Dual Core - (4 GB/1 TB HDD/Win...	4.1

The terminal output also shows the dimensions of the data frames: [5 rows x 10 columns] before the drop and [5 rows x 7 columns] after the drop.