Link :
https://drive.google.com/file/d/1afoNrRGNemnfBMM2jv7P8MjntsXEhcHP/view?usp=sharing

Anggita Ghozali
M0719015

Tugas

Lakukan beberapa langkah berikut pada dataset yang sudah diberikan.

Missing Values Checking
Categorical Data Encoding
Anomalies and Outlier Handling

Dataset :

Telco Churn

Langkah-langkah :

1. **Langkah Pertama yang dilakukan yaitu import Library yang akan digunakan**

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
pd.set_option('display.max_columns', None)
```

2. Memasukkan dataset yang akan digunakan

```
data = pd.read_csv('Telco.csv')
```

Melihat apakah data sudah berhasil di import atau belum

```
data.head().transpose()
```

	0	1	2	3	4
customerID	7590-VHVEG	5575-GNVDE	3668-QPYBK	7795-CFOCW	9237-HQITU
gender	Female	Male	Male	Male	Female
SeniorCitizen	0	0	0	0	0
Partner	Yes	No	No	No	No
Dependents	No	No	No	No	No
tenure	1	34	2	45	2
PhoneService	No	Yes	Yes	No	Yes
MultipleLines	No phone service	No	No	No phone service	No
InternetService	DSL	DSL	DSL	DSL	Fiber optic
OnlineSecurity	No	Yes	Yes	Yes	No
OnlineBackup	Yes	No	Yes	No	No
DeviceProtection	No	Yes	No	Yes	No
TechSupport	No	No	No	Yes	No
StreamingTV	No	No	No	No	No
StreamingMovies	No	No	No	No	No
Contract	Month-to-month	One year	Month-to-month	One year	Month-to-month
PaperlessBilling	Yes	No	Yes	No	Yes
PaymentMethod	Electronic check	Mailed check	Mailed check	Bank transfer (automatic)	Electronic check
MonthlyCharges	29.85	56.95	53.85	42.3	70.7
TotalCharges	29.85	1889.5	108.15	1840.75	151.65
Churn	No	No	Yes	No	Yes

3. Melihat info menggunakan .info() untuk melihat type data yang ada

```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 7043 entries, 0 to 7042
Data columns (total 21 columns):
 #   Column            Non-Null Count  Dtype  
---  ------            --------------  -----  
 0   customerID        7043 non-null   object 
 1   gender            7043 non-null   object 
 2   SeniorCitizen     7043 non-null   int64  
 3   Partner           7043 non-null   object 
 4   Dependents        7043 non-null   object 
 5   tenure            7043 non-null   int64  
 6   PhoneService      7043 non-null   object 
 7   MultipleLines     7043 non-null   object 
 8   InternetService   7043 non-null   object 
 9   OnlineSecurity    7043 non-null   object 
 10  OnlineBackup      7043 non-null   object 
 11  DeviceProtection  7043 non-null   object 
 12  TechSupport       7043 non-null   object 
 13  StreamingTV       7043 non-null   object 
 14  StreamingMovies   7043 non-null   object 
 15  Contract          7043 non-null   object 
 16  PaperlessBilling  7043 non-null   object 
 17  PaymentMethod     7043 non-null   object 
 18  MonthlyCharges    7043 non-null   float64
 19  TotalCharges      7043 non-null   object 
 20  Churn             7043 non-null   object 
dtypes: float64(1), int64(2), object(18)
memory usage: 1.1+ MB
```

Dari Output diketahui bahwa TotalCharges type-nya object sehingga masih terbaca salah. Jika dicek missing value pasti tidak akan terdeteksi. Oleh karena itu perlu diubah ke float64.

MISSING VALUE CHECKING

Tidak terbaca adanya missing value

```
data.isna().sum()
```

```
customerID          0
gender              0
SeniorCitizen       0
Partner             0
Dependents          0
tenure              0
PhoneService        0
MultipleLines       0
InternetService     0
OnlineSecurity      0
OnlineBackup        0
DeviceProtection    0
TechSupport         0
StreamingTV         0
StreamingMovies     0
Contract            0
PaperlessBilling    0
PaymentMethod       0
MonthlyCharges      0
TotalCharges        0
Churn               0
dtype: int64
```

Karena hanya TotalCharges yang salah tipe datanya, maka cek nilai yang ada di dalamnya. Dari output diketahui bahwa ada 11 nilai yang kosong dan tidak terdeteksi karena tipe datanya masih object.

```
data["TotalCharges"].value_counts()
```

```
          11
20.2      11
19.75      9
20.05      8
19.9       8
          ..
6849.4     1
692.35     1
130.15     1
3211.9     1
6844.5     1
Name: TotalCharges, Length: 6531, dtype: int64
```

Untuk melihat TotalCharges yang kosong menggunakan syntax dibawah ini. Terlihat ada 11 data yang kosong

```
# melihat TotalCharges yang kosong
datakosong = data[data["TotalCharges"]== " "]
datakosong
```

PaymentMethod	MonthlyCharges	TotalCharges	Churn
Bank transfer (automatic)	52.55		No
Mailed check	20.25		No
Mailed check	80.85		No
Mailed check	25.75		No
Credit card (automatic)	56.05		No
Mailed check	19.85		No
Mailed check	25.35		No
Mailed check	20.00		No
Mailed check	19.70		No
Mailed check	73.35		No
Bank transfer (automatic)	61.90		No

Mengubah Tipe Data TotalCharges ke float64

Karena tipe data masih salah, perlu dilakukan pengubahan tipe data menggunakan syntax seperti pada gambar. Dari output dihasilkan tipe data TotalCharges sudah float64.

```
# mengubah tipe data Total Charges ke Float
data["TotalCharges"] = pd.to_numeric(data.TotalCharges, errors='coerce')
data["TotalCharges"].dtype
```

```
dtype('float64')
```

Setelah tipe datanya sudah benar, dicek apakah ada missing value atau tidak. Dari output terdapat 11 missing value. Hal ini terbaca missing value karena tipe datanya sudah benar yaitu float64. Untuk melihat data yang missing value bisa juga menggunakan syntax dibawah ini.

```
data[data.TotalCharges.isna()]
```

```
# cek missing value lagi
data.isna().sum()
```

```
customerID          0
gender              0
SeniorCitizen       0
Partner             0
Dependents          0
tenure              0
PhoneService        0
MultipleLines       0
InternetService     0
OnlineSecurity      0
OnlineBackup        0
DeviceProtection    0
TechSupport         0
StreamingTV         0
StreamingMovies     0
Contract            0
PaperlessBilling    0
PaymentMethod       0
MonthlyCharges      0
TotalCharges       11
Churn               0
dtype: int64
```

IMPUTASI MENGGUNAKAN MEAN

Untuk mengisi nilai yang kosong, diising dengan mean dari kolom tersebut.

```
# imputasi menggunakan mean
mean = data["TotalCharges"].mean()
data["TotalCharges"] = data["TotalCharges"].replace(np.nan, mean)
```

Lalu dicek apakah masih terdapat missing value, dan hasilnya sudah tidak ada lagi missing value.

```
data.isna().any()
```

```
customerID          False
gender              False
SeniorCitizen       False
Partner             False
Dependents          False
tenure              False
PhoneService        False
MultipleLines       False
InternetService     False
OnlineSecurity      False
OnlineBackup        False
DeviceProtection    False
TechSupport         False
StreamingTV         False
StreamingMovies     False
Contract            False
PaperlessBilling    False
PaymentMethod       False
MonthlyCharges      False
TotalCharges        False
Churn               False
dtype: bool
```

CATEGORICAL DATA ENCODING

Delete variabel customer karena tidak terlalu berpengaruh

```
# delete variabel Customer karena tidak berpengaruh
data.drop(columns="customerID", inplace=True)
```

Setelah itu dilakukan encoding menggunakan syntax dibawah ini.

```
databaru = pd.get_dummies(data, columns=["gender", "Partner", "Dependents",
                                         "PhoneService", "MultipleLines",
                                         "InternetService", "OnlineSecurity",
                                         "OnlineBackup", "DeviceProtection",
                                         "TechSupport", 'StreamingTV', "StreamingMovies",
                                         "Contract", "PaperlessBilling", "PaymentMethod",
                                         "Churn"])
```

Output yang dihasilkan yaitu seperti gambar dibawah ini.

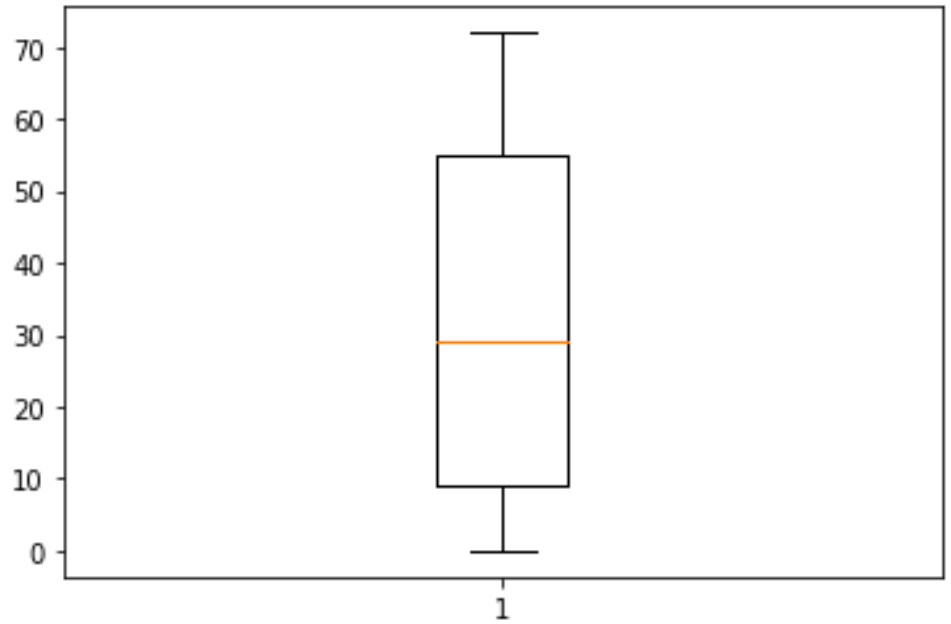
databaru

	SeniorCitizen	tenure	MonthlyCharges	TotalCharges	gender_Female	gender_Male	Partner_No	Partner_Yes	Dependents_No	Dependents_Yes	PhoneService_No	PhoneService_Yes	Multip
0	0	1	29.85	29.85	1	0	0	1	1	0	1	0	
1	0	34	56.95	1889.50	0	1	1	0	1	0	0	1	
2	0	2	53.85	108.15	0	1	1	0	1	0	0	1	
3	0	45	42.30	1840.75	0	1	1	0	1	0	1	0	
4	0	2	70.70	151.65	1	0	1	0	1	0	0	1	
...	...	...	...	...	...	...	...	...	...	...	...	...	
7038	0	24	84.80	1990.50	0	1	0	1	0	1	0	1	
7039	0	72	103.20	7362.90	1	0	0	1	0	1	0	1	
7040	0	11	29.60	346.45	1	0	0	1	0	1	1	0	
7041	1	4	74.40	306.60	0	1	0	1	1	0	0	1	
7042	0	66	105.65	6844.50	0	1	1	0	1	0	0	1	

7043 rows × 47 columns

ANOMALIES AN OUTLIERS HANDLING

```
# anomalies and outliers handling
plt.boxplot(databaru["tenure"])
plt.show()
```




```
Q1 = databaru["tenure"].quantile(0.25)
Q3 = databaru["tenure"].quantile(0.75)
IQR = Q3 - Q1
lower_bound = Q1 - 1.5*IQR
upper_bound = Q3 + 1.5*IQR
```

Dari boxplot tenure terlihat tidak ada outliers yang terdeteksi

```
outliers = databaru[databaru["tenure"]>upper_bound]
outliers
```

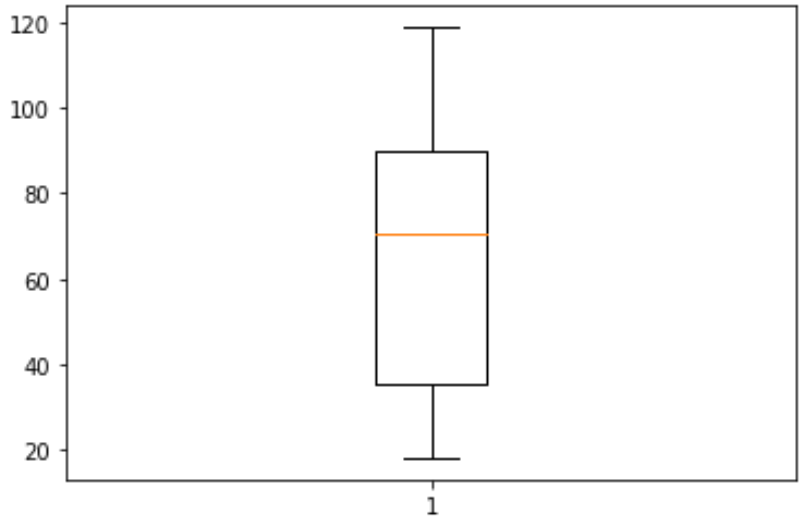
SeniorCitizen	tenure	MonthlyCharges	TotalCharges	gender_Female	gender_Male	Partner_No	Partner_Yes	Dependents_No	Dependents_Yes	PhoneService_No	PhoneService_Yes	MultipleLi

```
outliers1 = databaru[databaru["tenure"]<lower_bound]
outliers1
```

SeniorCitizen	tenure	MonthlyCharges	TotalCharges	gender_Female	gender_Male	Partner_No	Partner_Yes	Dependents_No	Dependents_Yes	PhoneService_No	PhoneService_Yes	MultipleLin

Dari output akan dihasilkan kosong karena tidak ada outlier pada data

```
# anomalies and outliers handling
plt.boxplot(databaru["MonthlyCharges"])
plt.show()
```




```
Q1 = databaru["MonthlyCharges"].quantile(0.25)
Q3 = databaru["MonthlyCharges"].quantile(0.75)
IQR = Q3 - Q1
lower_bound = Q1 - 1.5*IQR
upper_bound = Q3 + 1.5*IQR
```

Dari boxplot MonthlyCharges juga tidak ada outliers yang terdeteksi

```
outliers = databaru[databaru["MonthlyCharges"]>upper_bound]
outliers
```

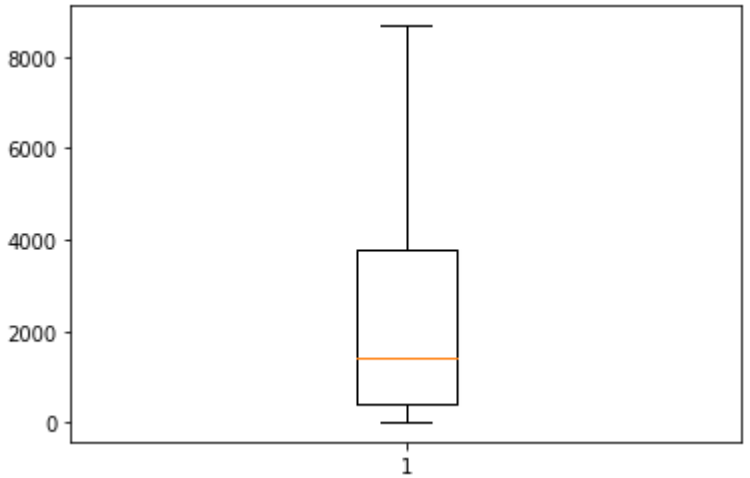
SeniorCitizen	tenure	MonthlyCharges	TotalCharges	gender_Female	gender_Male	Partner_No	Partner_Yes	Dependents_No	Dependents_Yes	PhoneService_No	PhoneService_Yes	MultipleLin

```
outliers1 = databaru[databaru["MonthlyCharges"] < lower_bound]
outliers1
```

SeniorCitizen	tenure	MonthlyCharges	TotalCharges	gender_Female	gender_Male	Partner_No	Partner_Yes	Dependents_No	Dependents_Yes	PhoneService_No	PhoneService_Yes	MultipleLir

Dari output akan dihasilkan kosong karena tidak ada outlier pada data

```
# anomalies and outliers handling
plt.boxplot(databaru["TotalCharges"])
plt.show()
```




```
Q1 = databaru["TotalCharges"].quantile(0.25)
Q3 = databaru["TotalCharges"].quantile(0.75)
IQR = Q3 - Q1
lower_bound = Q1 - 1.5*IQR
upper_bound = Q3 + 1.5*IQR
```

Dari boxplot TotalCharges juga tidak ada outliers yang terdeteksi

```
outliers = databaru[databaru["TotalCharges"]>upper_bound]
outliers
```

SeniorCitizen	tenure	MonthlyCharges	TotalCharges	gender_Female	gender_Male	Partner_No	Partner_Yes	Dependents_No	Dependents_Yes	PhoneService_No	PhoneService_Yes	MultipleLin

```
outliers1 = databaru[databaru["TotalCharges"]<lower_bound]
outliers1
```

SeniorCitizen	tenure	MonthlyCharges	TotalCharges	gender_Female	gender_Male	Partner_No	Partner_Yes	Dependents_No	Dependents_Yes	PhoneService_No	PhoneService_Yes	MultipleLin

Dari output akan dihasilkan kosong karena tidak ada outlier pada data