

Assignment

Topic 5 & 6

Oleh: Rosyida Ishma Mardhiyyah

Mentor: Erwin Fernanda



Topic 5

[Link Google Colab](#)

OUTLINE

1.

**Import library &
Understanding Dataset**

2.


**Missing Values
Handling**

3.

**Categorical Data
Encoding**

4.

**Anomalies and
Outlier Handling**



01

Import Library & Understanding Dataset

Import Library

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
```

Load Dataset

```
from google.colab import drive
drive.mount('/content/drive')
```

```
churn=pd.read_csv("drive/MyDrive/WA_Fn-UseC_-Telco-Customer-Churn.csv")
```

Understanding Dataset

Syntax `churn.head()` → Menampilkan beberapa baris teratas dari dataset

Output

	customerID	gender	SeniorCitizen	Partner	Dependents	tenure	PhoneService	MultipleLines	InternetService	Onli
0	7590-VHVEG	Female	0	Yes	No	1	No	No phone service	DSL	
1	5575-GNVDE	Male	0	No	No	34	Yes	No	DSL	
2	3668-QPYBK	Male	0	No	No	2	Yes	No	DSL	
3	7795-CFOCW	Male	0	No	No	45	No	No phone service	DSL	
4	9237-HQITU	Female	0	No	No	2	Yes	No	Fiber optic	

5 rows × 21 columns

Syntax `churn.shape` → Menampilkan jumlah baris dan kolom dari dataset

Output `(7043, 21)` → Jumlah baris: 7043; Jumlah kolom: 21

Understanding Dataset

Syntax

```
churn.info()
```

↓

Menampilkan informasi seperti variabel, jumlah non-null value, type data, dan memory usage

Output

```
<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
```



02

Missing Values Handling

Missing Values Checking

Syntax

```
churn.isnull().sum()
```

Menampilkan
jumlah missing
value pada
masing-masing
variabel.

Output

```
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
```

Tidak terdeteksi
adanya missing
value sehingga
tidak perlu
dilakukan
penanganan lebih
lanjut.



03

Categorical Data Encoding

Data Type Checking

Syntax

```
churn.dtypes
```

Menampilkan tipe data pada masing-masing variabel.

Output

```
churn.dtypes
customerID    object
gender        object
SeniorCitizen int64
Partner       object
Dependents    object
tenure        int64
PhoneService  object
MultipleLines object
InternetService object
OnlineSecurity object
OnlineBackup  object
DeviceProtection object
TechSupport   object
StreamingTV   object
StreamingMovies object
Contract      object
PaperlessBilling object
PaymentMethod object
MonthlyCharges float64
TotalCharges  object
Churn         object
dtype: object
```

Terdapat tipe data yang tidak sesuai dengan variabelnya, yaitu:

- Variabel Senior citizen seharusnya bertipe object
- Variabel Total Charges seharusnya bertipe float

Converting Data Type (Senior Citizen)

Syntax yang digunakan untuk mengubah tipe data “int” menjadi “object” adalah `.astype("str")`

Syntax

```
churn["SeniorCitizen"] = churn["SeniorCitizen"].astype("str")  
churn["SeniorCitizen"].dtype
```

Output

```
dtype('O')
```

→ Tipe data variabel Senior Citizen sudah berubah menjadi “object”

Converting Data Type (Total Charges)

Tipe data “object” dalam variabel Total Charges tidak bisa langsung diubah menjadi “float”, sehingga perlu dilakukan modifikasi dengan mengosongkan value Total Charges, lalu mengganti tipe datanya menjadi float dan mengisi value ulang dengan mengalikan tenure dan Monthly Charges.

Syntax

```
churn['TotalCharges'] = '0'  
churn['TotalCharges'] = churn['TotalCharges'].astype('float')  
churn['TotalCharges'] = churn['tenure']*churn['MonthlyCharges']  
churn['TotalCharges'].dtype
```

Output

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

→ Tipe data variabel Total Charges
sudah berubah menjadi “float”

Categorical Encoding

- Setelah melakukan converting data type, maka dapat dilakukan categorical encoding.
- Categorical encoding dapat dilakukan pada data yang bertipe "object" dan memuat values berupa kategori.
- Dalam categorical encoding, syntax yang digunakan adalah `pd.get_dummies()`

Syntax

```
churn_new = churn.drop(columns = ['customerID'])
churn_dummies = pd.get_dummies(churn_new)
churn_dummies.head()
```

Menghapus customer ID karena customer ID bertipe “object”, tetapi values nya bukan kategorik. Apabila tidak dihapus maka customer ID akan ikut di-encoding.

Output

	tenure	MonthlyCharges	TotalCharges	gender_Female	gender_Male	SeniorCitizen_0	SeniorCitizen_1	Partner_No	Paid_In_Arrears
0	1	29.85	29.85	1	0	1	0	0	0
1	34	56.95	1936.30	0	1	1	0	1	0
2	2	53.85	107.70	0	1	1	0	1	0
3	45	42.30	1903.50	0	1	1	0	1	0
4	2	70.70	141.40	1	0	1	0	1	0



04

Anomalies & Outliers Handling

Anomalies & Outliers Checking

Anomalies dan outliers hanya dapat dideteksi pada data numerik.

Syntax `.describe()` digunakan untuk menampilkan statistik dari data numerik.

Syntax `churn.describe()`

Output

	tenure	MonthlyCharges	TotalCharges
count	7043.000000	7043.000000	7043.000000
mean	32.371149	64.761692	2279.581350
std	24.559481	30.090047	2264.729447
min	0.000000	18.250000	0.000000
25%	9.000000	35.500000	394.000000
50%	29.000000	70.350000	1393.600000
75%	55.000000	89.850000	3786.100000
max	72.000000	118.750000	8550.000000

Variabel **Tenure**, **Monthly Charges**, dan **Total Chargers** merupakan variabel yang bertipe **numerik** sehingga selanjutnya pengecekan serta penanganan anomalies dan outliers dilakukan pada ketiga variabel ini.

Outliers Checking for Tenure

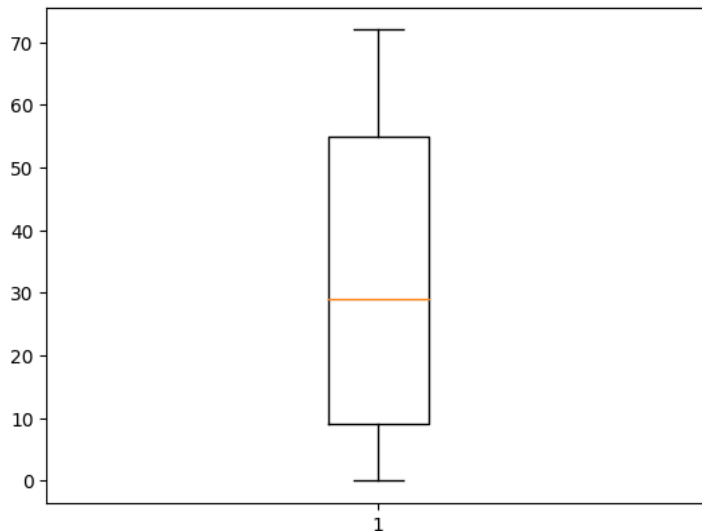
Cara yang paling umum untuk mendeteksi outlier adalah menggunakan boxplot

Import library `import matplotlib.pyplot as plt`

Syntax

```
plt.boxplot(churn["tenure"])  
plt.show
```

Output



→ Tidak terdeteksi adanya outlier dalam variabel Tenure

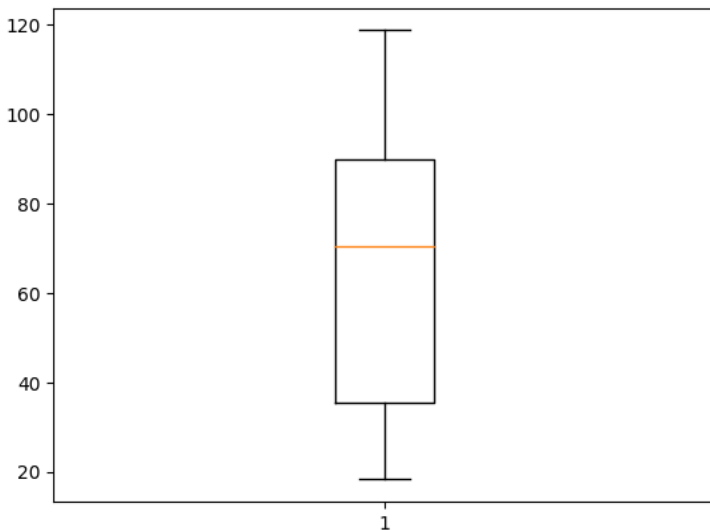
Outliers Checking for Monthly Charges

Cara yang paling umum untuk mendeteksi outlier adalah menggunakan boxplot

Syntax

```
plt.boxplot(churn["MonthlyCharges"])  
plt.show
```

Output



→ Tidak terdeteksi adanya outlier dalam variabel Monthly Charges

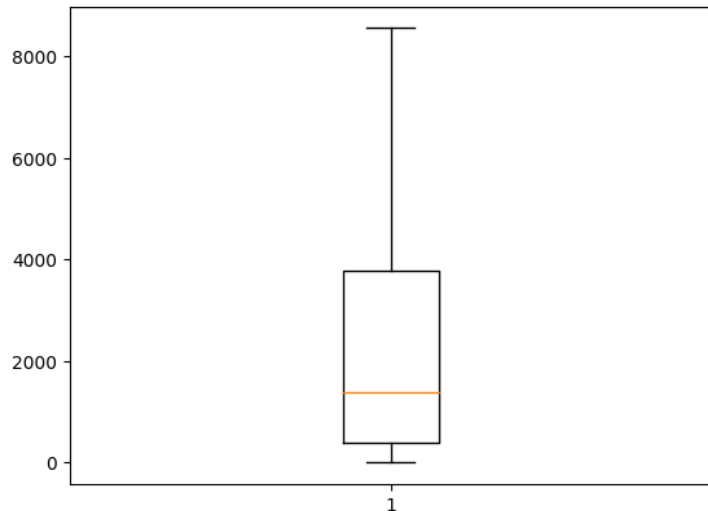
Outliers Checking for Total Charges

Cara yang paling umum untuk mendeteksi outlier adalah menggunakan boxplot

Syntax

```
plt.boxplot(churn["TotalCharges"])  
plt.show
```

Output



→ Tidak terdeteksi adanya outlier dalam variabel Total Charges



Topic 6

[Link Google Colab](#)