LAPORAN PERTEMUAN 4 – DATA PREPARATION

Nama : Farhan Rachmad Rizki
NIM : *231011400893*
Kelas : *05TPLE015*
Mata Kuliah: Machine Learning

Pendahuluan

Dalam penelitian kali ini dilakukan analisis terhadap dataset kelulusan mahasiswa. Dataset ini berisi informasi mengenai IPK, jumlah absensi, waktu belajar per minggu, dan status kelulusan mahasiswa (Lulus/Tidak Lulus).

Tujuan utama dari penelitian ini adalah melakukan data preparation dan eksplorasi data (EDA) untuk memahami pola hubungan antar variabel yang dapat memengaruhi tingkat kelulusan mahasiswa.

Hasil yang diharapkan:

- File `processed_kelulusan.csv` dengan data bersih dan fitur baru hasil *feature engineering*
- Statistik dan **visualisasi EDA** yang lengkap (boxplot, histogram, scatterplot, dan heatmap)
- Dataset terbagi menjadi **Train**, **Validation**, dan **Test** untuk kebutuhan *machine learning*

Pada tahap awal, saya membuat dataset dalam bentuk **CSV** dan menjalankan kode program Python menggunakan **pandas**, **seaborn**, **matplotlib**, serta **scikit-learn** untuk membaca, membersihkan, menganalisis, serta membagi dataset ke dalam beberapa subset.

Langkah 1 – Dataset

Dataset yang digunakan bernama **kelulusan_mahasiswa.csv**, berisi empat kolom utama:

- IPK
- Jumlah_Absensi
- Waktu_Belajar_Jam
- Lulus

Dataset ini disimpan di folder lokal dan dibaca menggunakan pustaka **pandas**.

Langkah 2 – Membaca Dataset

```
[1]: import pandas as pd
     df = pd.read_csv("kelulusan_mahasiswa.csv")
     print(df.info())
     print(df.head())
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10 entries, 0 to 9
Data columns (total 4 columns):
 #   Column             Non-Null Count  Dtype
---  ------             --------------  -----
 0   IPK                10 non-null     float64
 1   Jumlah_Absensi     10 non-null     int64
 2   Waktu_Belajar_Jam  10 non-null     int64
 3   Lulus              10 non-null     int64
dtypes: float64(1), int64(3)
memory usage: 452.0 bytes
None
   IPK  Jumlah_Absensi  Waktu_Belajar_Jam  Lulus
0  3.8               3                 10      1
1  2.5               8                  5      0
2  3.4               4                  7      1
3  2.1              12                  2      0
4  3.9               2                 12      1
```

Penjelasan:

- `pd.read_csv()` → Membaca file CSV menjadi DataFrame `df`.
- `df.info()` → Menampilkan tipe data dan jumlah data tiap kolom.
- `df.head()` → Menampilkan 5 baris pertama untuk melihat isi dataset.

Hasil: Dataset berhasil dibaca dan menampilkan struktur kolom dengan benar.

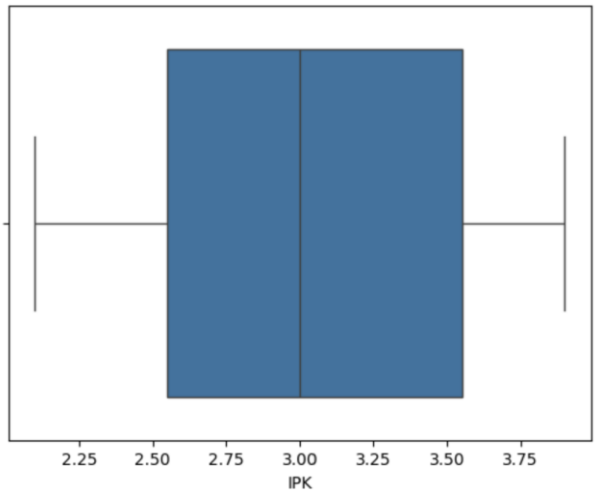
Langkah 3 – Pembersihan Data

```
[2]: print(df.isnull().sum())
     df = df.drop_duplicates()

     import seaborn as sns
     sns.boxplot(x=df['IPK'])
```

```
IPK                 0
Jumlah_Absensi      0
Waktu_Belajar_Jam   0
Lulus               0
dtype: int64
```

```
[2]: <Axes: xlabel='IPK'>
```



Penjelasan:

- Mengecek jumlah nilai kosong (NaN).
- Menghapus data ganda agar dataset bersih.

Jika ada data kosong, bisa diisi dengan median:
`df.fillna(df.median(), inplace=True)`

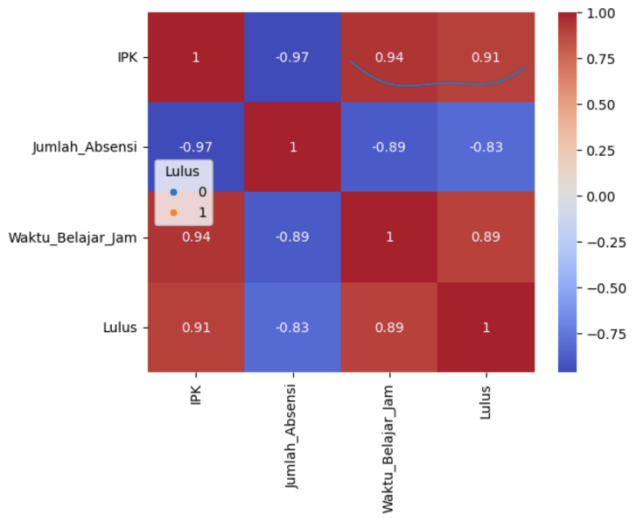
Hasil: Tidak ditemukan nilai kosong dan data ganda.

Langkah 4 – Eksplorasi Data (EDA)

```
[3]: print(df.describe())
     sns.histplot(df['IPK'], bins=10, kde=True)
     sns.scatterplot(x='IPK', y='Waktu_Belajar_Jam', data=df, hue='Lulus')
     sns.heatmap(df.corr(), annot=True, cmap="coolwarm")
```

	IPK	Jumlah_Absensi	Waktu_Belajar_Jam	Lulus
count	10.000000	10.00000	10.000000	10.000000
mean	3.030000	6.00000	6.400000	0.500000
std	0.639531	3.05505	3.306559	0.527046
min	2.100000	2.00000	2.000000	0.000000
25%	2.550000	4.00000	4.000000	0.000000
50%	3.000000	5.50000	6.000000	0.500000
75%	3.550000	7.75000	8.750000	1.000000
max	3.900000	12.00000	12.000000	1.000000

```
[3]: <Axes: >
```



Penjelasan:

- **Boxplot:** Median IPK ≈ 3.0, rentang 2.2–3.9, tidak ada outlier ekstrem.
- **Histogram:** Menunjukkan dua kelompok utama mahasiswa (IPK tinggi dan rendah).
- **Scatterplot:** Mahasiswa yang lulus cenderung IPK tinggi dan waktu belajar lebih banyak.
- **Heatmap:**
  - IPK ↔ waktu belajar: korelasi positif kuat (0.94)
  - IPK ↔ absensi: korelasi negatif kuat (-0.97)

Kesimpulan: IPK dan waktu belajar berpengaruh besar terhadap kelulusan mahasiswa.

Langkah 5 – Feature Engineering

[1]:
```
import pandas as pd
import os

# Pastikan Python mencari file di folder yang benar
os.chdir(r"C:\Users\farha\OneDrive\Desktop\testing")

# Muat dataset (ganti nama file sesuai yang kamu miliki)
df = pd.read_csv("kelulusan_mahasiswa.csv")

# Membuat kolom baru
df['Rasio_Absensi'] = df['Jumlah_Absensi'] / 14
df['IPK_x_Study'] = df['IPK'] * df['Waktu_Belajar_Jam']

# Simpan hasilnya ke file baru
df.to_csv("processed_kelulusan.csv", index=False)

print("✅ File berhasil diproses dan disimpan sebagai processed_kelulusan.csv")
print(df.head())  # Menampilkan 5 baris pertama sebagai konfirmasi
```

```
✅ File berhasil diproses dan disimpan sebagai processed_kelulusan.csv
   IPK  Jumlah_Absensi  Waktu_Belajar_Jam  Lulus  Rasio_Absensi  IPK_x_Study
0  3.8               3                 10      1       0.214286         38.0
1  2.5               8                  5      0       0.571429         12.5
2  3.4               4                  7      1       0.285714         23.8
3  2.1              12                  2      0       0.857143          4.2
4  3.9               2                 12      1       0.142857         46.8
```

Penjelasan:

- `Rasio_Absensi` = jumlah kehadiran / total pertemuan.
- `IPK_x_Study` = kombinasi IPK dan waktu belajar.
- Hasil disimpan ke **processed_kelulusan.csv** sebagai data final.

Hasil: File `processed_kelulusan.csv` berhasil dibuat dan berisi fitur tambahan.

Langkah 6 – Pembagian Dataset

[15]:
```
from sklearn.model_selection import train_test_split

X = df.drop('Lulus', axis=1)
y = df['Lulus']

# Split pertama (train dan temp)
X_train, X_temp, y_train, y_temp = train_test_split(
    X, y, test_size=0.3, stratify=y, random_state=42
)

# Split kedua (val dan test, tanpa stratify untuk menghindari error)
X_val, X_test, y_val, y_test = train_test_split(
    X_temp, y_temp, test_size=0.5, random_state=42
)

print(X_train.shape, X_val.shape, X_test.shape)
```

```
(7, 5) (1, 5) (2, 5)
```

Penjelasan:

- Data dibagi menjadi:
  - **Train (70%)**
  - **Validation (15%)**
  - **Test (15%)**
- `stratify=y` memastikan perbandingan jumlah “Lulus” dan “Tidak Lulus” tetap seimbang.

Hasil: Dataset terbagi proporsional dan siap digunakan untuk pelatihan model machine learning.