Nama : Firman Ramadhan Saputra
NIM : 231011400891
Kelas : 05TPLE015
Mata Kuliah : Machine Learning

LAPORAN DATA PREPARATION – PERTEMUAN 4

1. **Buat Dataset dan Data Collection**

   Dataset yang digunakan adalah 'kelulusan_mahasiswa.csv', yang berisi data IPK, Jumlah_Absensi, Waktu_Belajar_Jam, Lulus, yang seluruhnya adalah numerik atau angka. Dataset tersebut dimuat dengan menggunakan **Pandas** dan data diperiksa menggunakan **df.info()** untuk melihat tipe data dan menggunakan **df.head()** untuk melihat data dari baris paling atas. Berikut contoh codenya:

```
# collection
import pandas as pd
df = pd.read_csv("kelulusan_mahasiswa.csv")
print(df.info())
print(df.head())
```

   Dan akan menampilkan output berikut:

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 110 entries, 0 to 109
Data columns (total 4 columns):
 #   Column            Non-Null Count  Dtype
---  ------            --------------  -----
 0   IPK               110 non-null    float64
 1   Jumlah_Absensi    110 non-null    int64
 2   Waktu_Belajar_Jam 110 non-null    int64
 3   Lulus             110 non-null    int64
dtypes: float64(1), int64(3)
memory usage: 3.6 KB
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
```

2. Data Cleaning

Lakukan Pengecekan nilai kosong pada data dengan menggunakan **df.isnull().sum()** yang dilakukan untuk memastikan tidak ada missing value pada data. Untuk data duplikat dihapus dengan **df.drop_duplicates()** agar analisis lebih akurat. Serta identifikasi outlier dengan boxplot.

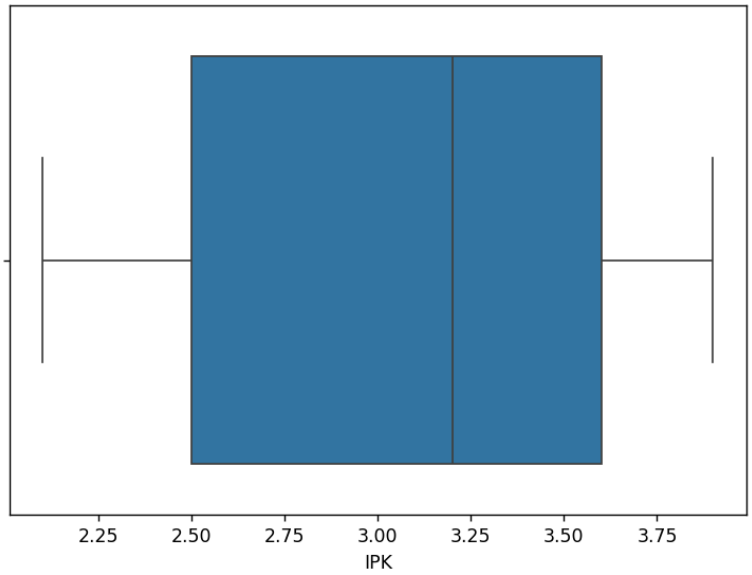
Berikut contoh codenya:

```
# cleaning
print(df.isnull().sum())
df = df.drop_duplicates()

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

Dengan output berikut:

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



3. **Exploratory Data Analysis (EDA)**

Melakukan perhitungan deskriptif dengan **df.describe()**, lalu melakukan import library **seaborn** dan **matplotlib.pyplot** untuk pembuatan histogram dan menampilkan hasil berupa visualisasi grafik dari perhitungan data. Buat visualisasi histogram dengan menggunakan **sns.histplot**, visualisasi scatterplot antara IPK dan waktu belajar dengan menggunakan **sns.scatterplot**, serta sajikan visualisasi heatmap dengan **sns.heatmap** untuk melihat korelasi antar variabel. Berikut codenya:

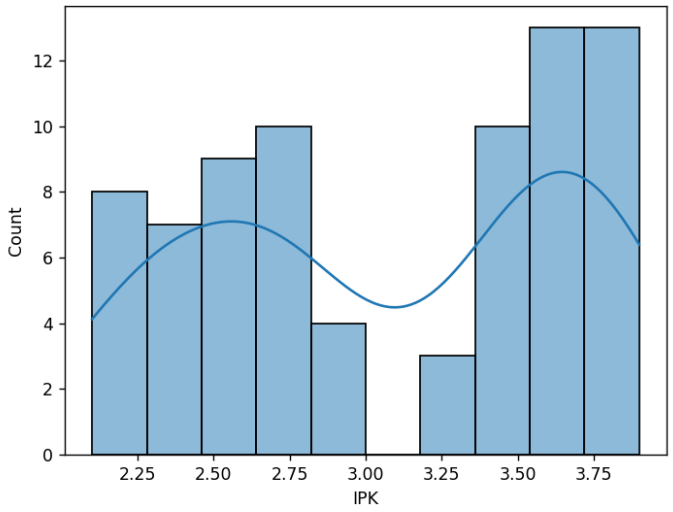
```
# Exploratory Data Analysis (EDA)
print(df.describe())
sns.histplot(df['IPK'], bins=10, kde=True)
plt.show()
sns.scatterplot(x='IPK', y='Waktu_Belajar_Jam', data=df, hue='Lulus')
plt.show()
sns.heatmap(df.corr(), annot=True, cmap='coolwarm')
plt.show()
```

Lalu berikut output yang dihasilkan:

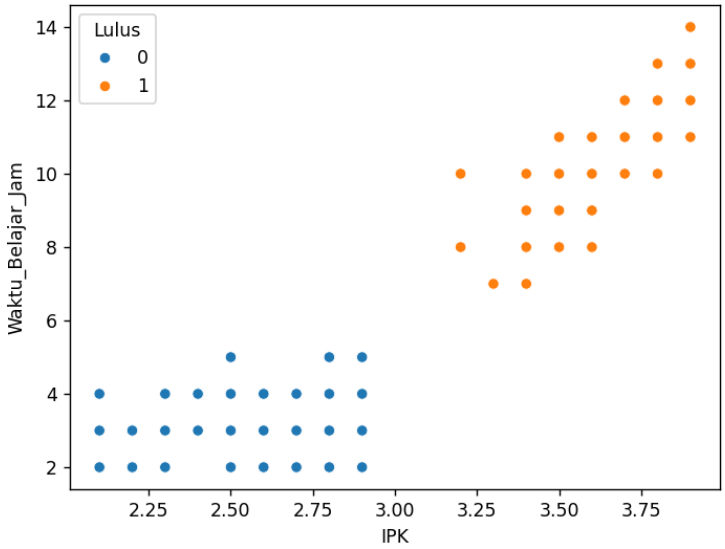
Hasil perhitungan deskriptif:

	IPK	Jumlah_Absensi	Waktu_Belajar_Jam	Lulus
count	77.000000	77.000000	77.000000	77.000000
mean	3.081818	6.311688	6.896104	0.506494
std	0.607568	2.987957	3.988749	0.503236
min	2.100000	2.000000	2.000000	0.000000
25%	2.500000	4.000000	3.000000	0.000000
50%	3.200000	5.000000	7.000000	1.000000
75%	3.600000	9.000000	11.000000	1.000000
max	3.900000	12.000000	14.000000	1.000000

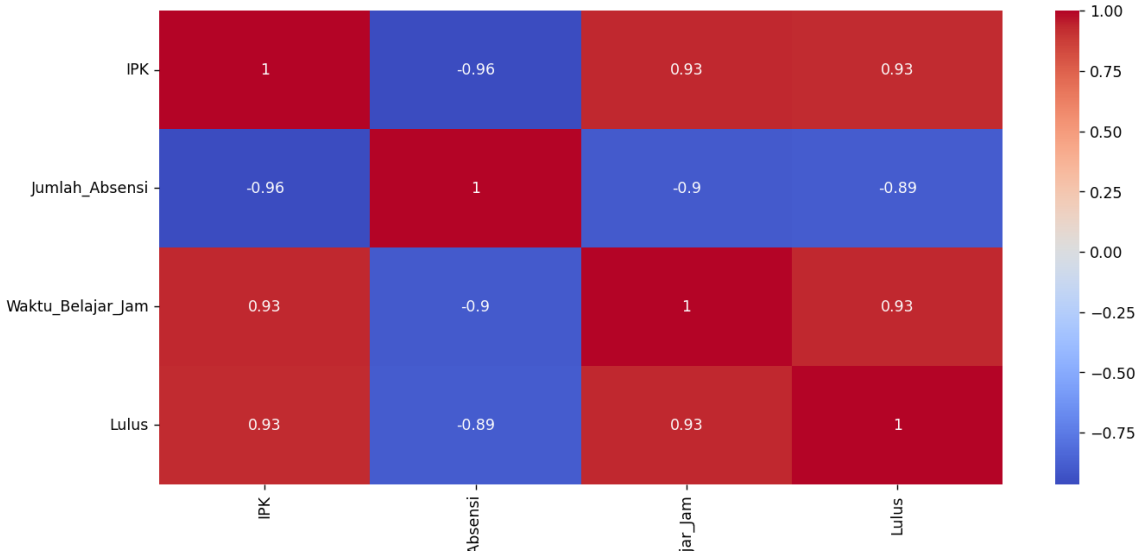
Hasil dari Histplot:



Hasil dari Scatterplot:



Hasil dari Heatmap:



4. **Feature Engineering**

Pada tahap *feature engineering*, dibuat fitur turunan baru berupa **Rasio_Absensi** yang dihitung dari pembagian jumlah absensi dengan total 14 pertemuan, serta **IPK_x_Study** yang merupakan hasil perkalian antara IPK dan waktu belajar, kemudian dataset hasil transformasi tersebut disimpan ke dalam file **processed_kelulusan.csv**.

Berikut dengan contoh codenya:

```
# Feature Engineering
df['Rasio_Absensi'] = df['Jumlah_Absensi'] / 14
df['IPK_x_Study'] = df['IPK'] * df['Waktu_Belajar_Jam']
df.to_csv("processed_kelulusan", index=False)
```

Berikut adalah hasilnya dengan format file .csv:

processed_kelulusan.csv

```
IPK,Jumlah_Absensi,Waktu_Belajar_Jam,Lulus,Rasio_Absensi,IPK_x_Study
3.8,3,10,1,0.21428571428571427,38.0
2.5,8,5,0,0.5714285714285714,12.5
3.4,4,7,1,0.2857142857142857,23.8
2.1,12,2,0,0.8571428571428571,4.2
3.9,2,12,1,0.14285714285714285,46.8
2.8,6,4,0,0.42857142857142855,11.2
3.2,5,8,1,0.35714285714285715,25.6
2.7,7,3,0,0.5,8.100000000000001
3.6,4,9,1,0.2857142857142857,32.4
2.3,9,4,0,0.6428571428571429,9.2
3.5,3,9,1,0.21428571428571427,31.5
2.6,7,4,0,0.5,10.4
3.7,4,10,1,0.2857142857142857,37.0
2.2,11,3,0,0.7857142857142857,6.6000000000000005
3.9,2,11,1,0.14285714285714285,42.9
2.9,6,5,0,0.42857142857142855,14.5
3.3,5,7,1,0.35714285714285715,23.099999999999998
2.4,8,4,0,0.5714285714285714,9.6
3.8,3,12,1,0.21428571428571427,45.599999999999994
2.5,9,3,0,0.6428571428571429,7.5
3.6,4,8,1,0.2857142857142857,28.8
2.7,10,2,0,0.7142857142857143,5.4
3.4,5,9,1,0.35714285714285715,30.599999999999998
2.1,11,4,0,0.7857142857142857,8.4
3.7,3,11,1,0.21428571428571427,40.7
2.8,7,5,0,0.5,14.0
3.2,4,10,1,0.2857142857142857,32.0
2.6,9,3,0,0.6428571428571429,7.800000000000001
3.9,2,13,1,0.14285714285714285,50.699999999999996
2.3,10,2,0,0.7142857142857143,4.6
3.5,4,8,1,0.2857142857142857,28.0
2.4,9,4,0,0.6428571428571429,9.6
```

5. **Splitting Dataset**

Dataset dibagi menjadi tiga bagian menggunakan **stratified split**, yaitu **70% untuk data training**, **15% untuk data validation**, dan **15% untuk data testing**, dengan memastikan distribusi kelas pada variabel target tetap seimbang.

Berikut contoh codenya:

```python
# splitting Dataset
from sklearn.model_selection import train_test_split

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

x_train, x_temp, y_train, y_temp = train_test_split(
    x, y, test_size=0.3, stratify=y, random_state=42
)

x_val, x_test, y_val, y_test = train_test_split(
    x_temp, y_temp, test_size=0.5, stratify=y_temp, random_state=42
)

print(x_train.shape, x_val.shape, x_test.shape)
```

Berikut dengan output yang dihasilkan:

```
(53, 5) (12, 5) (12, 5)
```

Hasil yang ditampilkan (7, 5) (1, 5) (2, 5) itu adalah:

- (7, 5) → **Data Training** (7 baris, 5 fitur).
- (1, 5) → **Data Validation** (1 baris, 5 fitur).
- (2, 5) → **Data Testing** (2 baris, 5 fitur).

Kesimpulan

Berdasarkan seluruh rangkaian langkah yang dilakukan mulai dari pembuatan dataset, proses pembersihan, analisis eksploratif, rekayasa fitur, hingga pembagian data, dapat disimpulkan bahwa dataset **kelulusan_mahasiswa** berhasil dipersiapkan dengan baik: data bersih dari duplikasi dan *missing value*, pola distribusi IPK serta hubungan antarvariabel dapat dipahami melalui EDA, fitur turunan baru berhasil dibuat untuk memperkuat analisis, dan dataset telah terbagi seimbang menjadi **data training, validation, dan testing** sesuai proporsi 70%-15%-15%, sehingga siap digunakan untuk tahap pembangunan serta evaluasi model prediksi kelulusan mahasiswa.