

Nama Dosen : Teguh Iman Hermanto, M.Kom  
 Mata Kuliah : Machine Learning 1  
 Pembahasan : Exploratory Data Analysis (EDA)  
 Pokok Pemb : - Mengenal Library ML  
 - Mengenal Statistik Deskriptif  
 - Mengenal EDA Data Numerik  
 - Mengenal EDA Data Kategori  
 - Mengenal EDA data Multivariabel  
 - Mengenal Univariate Analysis  
 - Mengenal Bivariate Analysis

## 1. Mengenal Library Machine Learning



```
1 import pandas as pd
2 import matplotlib.pyplot as plt
3 import seaborn as sns
```



```
1 df = pd.read_csv("adult.data.csv")
```



```
1 df.head()
```



```
1 df.info()
```

## 2. Mengenal Statistik Deskriptif



```
1 # Menghitung statistik deskriptif untuk kolom numerik
2 numerical_stats = df.describe()
3 print("Statistik Deskriptif untuk Kolom Numerik:\n", numerical_stats)
```



```
1 # Menghitung frekuensi untuk kolom kategorikal
2 categorical_stats = df.select_dtypes(include=['category']).apply(lambda x: x.value_counts(normalize=True) * 100)
3 print("\nPersentase Frekuensi untuk Kolom Kategorikal:\n", categorical_stats)
```

```
1 # Menampilkan jumlah missing values pada tiap kolom
2 missing_values = df.isnull().sum()
3 print("\nJumlah Missing Values pada tiap kolom:\n", missing_values)
```

### 3. Mengenal Bentuk EDA dengan Data Numerik

```
1 # Histogram
2 plt.figure(figsize=(8, 6))
3 sns.histplot(df['age'], kde=True)
4 plt.title('Histogram of Age')
5 plt.xlabel('Age')
6 plt.ylabel('Frequency')
7 plt.show()
```

```
1 # Box Plot
2 plt.figure(figsize=(8, 6))
3 sns.boxplot(x='sex', y='hours-per-week', data=df)
4 plt.title('Box Plot of Hours per Week by Sex')
5 plt.xlabel('Sex')
6 plt.ylabel('Hours per Week')
7 plt.show()
```

```
1 # Scatter Plot
2 plt.figure(figsize=(8, 6))
3 sns.scatterplot(x='age', y='capital-gain', data=df)
4 plt.title('Scatter Plot of Age vs Capital Gain')
5 plt.xlabel('Age')
6 plt.ylabel('Capital Gain')
7 plt.show()
```

## 4. Mengenal Bentuk EDA dengan Data Kategori

```

1 # Line Plot (Example: Average capital gain by age)
2 avg_capital_gain_by_age = df.groupby('age')['capital-gain'].mean()
3 plt.figure(figsize=(8, 6))
4 plt.plot(avg_capital_gain_by_age.index, avg_capital_gain_by_age.values)
5 plt.title('Line Plot of Average Capital Gain by Age')
6 plt.xlabel('Age')
7 plt.ylabel('Average Capital Gain')
8 plt.show()

```

```

1 # Bar Chart
2 # Contoh: Melihat jumlah orang berdasarkan tingkat pendidikan
3 education_counts = df['education'].value_counts()
4 plt.figure(figsize=(10, 5))
5 plt.bar(education_counts.index, education_counts.values)
6 plt.xlabel('Tingkat Pendidikan')
7 plt.ylabel('Jumlah Orang')
8 plt.title('Jumlah Orang Berdasarkan Tingkat Pendidikan')
9 plt.xticks(rotation=45, ha='right')
10 plt.show()

```

```

1 # Pie Chart
2 # Contoh: Melihat persentase jenis kelamin
3 sex_counts = df['sex'].value_counts()
4 plt.figure(figsize=(6, 6))
5 plt.pie(sex_counts.values, labels=sex_counts.index, autopct='%1.1f%%', startangle=90)
6 plt.title('Persentase Jenis Kelamin')
7 plt.show()

```

```

1 # Donut Chart
2 # Contoh: Melihat persentase ras
3 race_counts = df['race'].value_counts()
4 plt.figure(figsize=(6, 6))
5 plt.pie(race_counts.values, labels=race_counts.index, autopct='%1.1f%%', startangle=90, wedgeprops=dict(width=0.4))
6 plt.title('Persentase Ras')
7 plt.show()

```

## 5. Mengenal Bentuk EDA dengan Data Multivariabel

```

1 # Create a heatmap
2 # Select numerical features for the heatmap
3 numerical_features = ['age', 'fnlwgt', 'education-num', 'capital-gain', 'capital-loss', 'hours-per-week']
4 heatmap_data = df[numerical_features].corr()
5
6 plt.figure(figsize=(10, 8))
7 sns.heatmap(heatmap_data, annot=True, cmap='coolwarm', fmt=".2f")
8 plt.title('Correlation Heatmap')
9 plt.show()

```

```

1 # Create a pair plot
2 sns.pairplot(df[['age', 'education-num', 'hours-per-week', 'capital-gain', 'capital-loss']])
3 plt.show()

```

```

1 # Create a bubble chart
2 # Select features for the bubble chart
3 bubble_data = df[['age', 'hours-per-week', 'capital-gain']]
4
5 plt.figure(figsize=(10, 8))
6 plt.scatter(x='age', y='hours-per-week', s='capital-gain', data=bubble_data, alpha=0.5)
7 plt.xlabel('Age')
8 plt.ylabel('Hours per Week')
9 plt.title('Bubble Chart: Age, Hours per Week, and Capital Gain')
10 plt.show()

```

## 6. Mengenal Univariate Analysis

```

1 # Univariate Analysis for Age
2 plt.figure(figsize=(8, 6))
3 sns.histplot(df['age'], kde=True)
4 plt.title('Distribution of Age')
5 plt.xlabel('Age')
6 plt.ylabel('Frequency')
7 plt.show()

```

```

1 # Univariate Analysis for Education (Categorical)
2 plt.figure(figsize=(10, 6))
3 sns.countplot(x='education', data=df)
4 plt.title('Count of Individuals by Education Level')
5 plt.xlabel('Education Level')
6 plt.ylabel('Count')
7 plt.xticks(rotation=45, ha='right')
8 plt.show()

```

```

1 # Univariate Analysis for Salary (Categorical)
2 plt.figure(figsize=(6, 6))
3 sns.countplot(x='salary', data=df)
4 plt.title('Count of Individuals by Salary')
5 plt.xlabel('Salary')
6 plt.ylabel('Count')
7 plt.show()

```

## 7. Mengenal Bivariate Analysis

```

1 # 1. Age vs. Salary
2 # Deskripsi: Melihat hubungan antara umur dan tingkat pendapatan.
3 # Kita mengharapkan bahwa orang yang lebih tua mungkin memiliki pendapatan yang lebih tinggi.
4 plt.figure(figsize=(8, 6))
5 sns.boxplot(x='salary', y='age', data=df)
6 plt.title('Hubungan antara Umur dan Tingkat Pendapatan')
7 plt.show()

```

```

1 # 2. Education vs. Salary
2 # Deskripsi: Melihat hubungan antara tingkat pendidikan dan pendapatan.
3 # Kita mengharapkan bahwa orang dengan pendidikan lebih tinggi cenderung memiliki pendapatan lebih tinggi.
4 plt.figure(figsize=(12, 6))
5 sns.countplot(x='education', hue='salary', data=df)
6 plt.title('Hubungan antara Tingkat Pendidikan dan Tingkat Pendapatan')
7 plt.xticks(rotation=45, ha='right')
8 plt.show()

```

```

1 # 3. Hours-per-week vs. Salary
2 # Deskripsi: Melihat hubungan antara jumlah jam kerja per minggu dan pendapatan.
3 # Kita mengharapkan bahwa orang yang bekerja lebih banyak jam cenderung memiliki pendapatan lebih tinggi.
4 plt.figure(figsize=(8, 6))
5 sns.boxplot(x='salary', y='hours-per-week', data=df)
6 plt.title('Hubungan antara Jam Kerja per Minggu dan Tingkat Pendapatan')
7 plt.show()

```

```

1 # 4. Sex vs. Salary
2 # Deskripsi: Melihat perbedaan pendapatan antara pria dan wanita.
3 plt.figure(figsize=(8, 6))
4 sns.countplot(x='sex', hue='salary', data=df)
5 plt.title('Perbedaan Tingkat Pendapatan Berdasarkan Jenis Kelamin')
6 plt.show()

```

```

1 # 5. Occupation vs. Salary
2 # Deskripsi: Melihat hubungan antara jenis pekerjaan dan pendapatan.
3 # Kita mengharapkan bahwa beberapa jenis pekerjaan memiliki pendapatan yang lebih tinggi daripada yang lain.
4 plt.figure(figsize=(12, 6))
5 sns.countplot(x='occupation', hue='salary', data=df)
6 plt.title('Hubungan antara Jenis Pekerjaan dan Tingkat Pendapatan')
7 plt.xticks(rotation=45, ha='right')
8 plt.show()

```

```

1 # 6. Correlation Matrix
2 # Deskripsi: Melihat korelasi antara variabel-variabel numerik.
3 # Kita dapat melihat variabel mana yang memiliki korelasi kuat dengan variabel lainnya.
4 numeric_cols = df.select_dtypes(include=['number']).columns
5 correlation_matrix = df[numeric_cols].corr()
6 plt.figure(figsize=(10, 8))
7 sns.heatmap(correlation_matrix, annot=True, cmap='coolwarm', fmt=".2f")
8 plt.title('Correlation Matrix')
9 plt.show()

```

**Latihan**

1. Tampilkan Plot Education berdasarkan Salarynya
2. Tampilkan jumlah pendidikan Bachelors dengan gaji  $\leq 50k$
3. Tampilkan data occupation dengan gaji diatas 50k
4. Tampilkan Plot occupation dengan gaji diatas 50k
5. Tampilkan rata-rata hours per week untuk occupation sales?
6. Tampilkan rata-rata hours per week untuk occupation Prof-Speciality?