Exploratory Data Analysis

Home About Contact

EDA

Explatory data analysis

Dataset ini berisi informasi mengenai penumpang kapal Titanic yang tenggelam pada tahun 1912. Dalam dataset ini, kita akan mencoba memahami pola, karakteristik, serta beberapa faktor yang mungkin berpengaruh terhadap kemungkinan seseorang selamat atau tidak dari tragedi tersebut.

Melalui proses EDA, kita akan melihat bagaimana data ini disusun, melakukan inspeksi terhadap data yang hilang, melihat ringkasan statistik, hingga menyusun insight awal yang bisa digunakan untuk analisis lebih lanjut.

Presented by Farhan Azka

Exploratory Data Analysis

WHAT IS EDA?

Exploratory Data Analysis (EDA) is the initial step in analyzing data to understand its structure, patterns, and key characteristics. It helps identify missing values, duplicates, data distribution, and relationships between variables. The main goal is to gain insights and prepare the data for accurate and effective analysis.

Exploratory Data Analysis

PORTOFOLIO GOALS

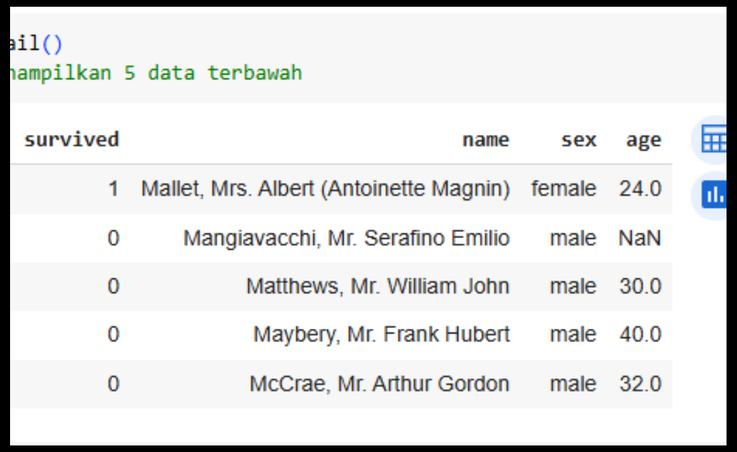
The main objectives of this Exploratory Data Analysis are:

- Duplicate Data
- Missing values
- Data observation

LOAD DATA

At the beginning of the analysis, the Titanic dataset was loaded and explored using functions like head(), tail(), and sample() to get a general idea of the data structure.
Some of the main columns in the dataset include:

- survived: 0 means did not survive, 1 means survived
- name: passenger's full name
- sex: gender (male or female)
- age: age of the passenger in decimal format



```
ail()
ampilkan 5 data terbawah
```

survived	name	sex	age
1	Mallet, Mrs. Albert (Antoinette Magnin)	female	24.0
0	Mangiavacchi, Mr. Serafino Emilio	male	NaN
0	Matthews, Mr. William John	male	30.0
0	Maybery, Mr. Frank Hubert	male	40.0
0	McCrae, Mr. Arthur Gordon	male	32.0

```
ple(5)
pilkan 5 data acak
```

survived	name	sex
1	Hippach, Mrs. Louis Albert (Ida Sophia Fischer)	female
1	Becker, Miss. Ruth Elizabeth	female
1	Taylor, Mrs. Elmer Zebley (Juliet Cummins Wright)	female
0	Blackwell, Mr. Stephen Weart	male
0	Harrington, Mr. Charles H	male

```
port data
= pd.read_excel('titanic.xlsx')
= df.copy()
ead() #Menampilkan 5 teratas
```

survived	name	sex	ag
1	Allen, Miss. Elisabeth Walton	female	29.000
1	Allison, Master. Hudson Trevor	male	0.916
0	Allison, Miss. Helen Loraine	female	2.000
0	Allison, Mr. Hudson Joshua Creighton	male	30.000
0	Allison, Mrs. Hudson J C (Bessie Waldo Daniels)	female	25.000

LOAD DATA

```
df.info()
# Menampilkan informasi umum dataset
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 500 entries, 0 to 499
Data columns (total 4 columns):
 #   Column    Non-Null Count  Dtype
---  ------    --------------  -----
 0   survived  500 non-null    int64
 1   name      500 non-null    object
 2   sex       500 non-null    object
 3   age       451 non-null    float64
dtypes: float64(1), int64(1), object(2)
memory usage: 15.8+ KB
```

Using the data.info() function, we found that the Titanic dataset contains 500 rows and 4 columns:
survived, name, sex, and age.
Key Points:

- All columns except age have complete data (500 entries).
- The age column has only 451 non-null values, meaning there are 49 missing entries.
- Data types for each column:
- survived: integer (int64)
- name and sex: string (object)
- age: float (float64)

Exploratory Data Analysis

STATITICAL SUMMARY

```
df.describe(include='object')
```

	name	sex
count	500	500
unique	499	2
top	Eustis, Miss. Elizabeth Mussey	male
freq	2	288

```
df.describe()
```

	survived	age
count	500.000000	451.000000
mean	0.540000	35.917775
std	0.498897	14.766454
min	0.000000	0.666700
25%	0.000000	24.000000
50%	1.000000	35.000000
75%	1.000000	47.000000
max	1.000000	80.000000

Using the describe() function, we can summarize the key statistics of the Titanic dataset:
For numerical columns:

- The survived column has a mean of 0.54, which suggests that around 54% of passengers survived.
- The age column shows an average age of 35.91 years, ranging from 0.67 to 80 years.
- There are 451 valid age entries, meaning 49 are missing.
- The standard deviation for age is 14.77, showing a wide variation in passenger ages.
- For categorical columns (via describe (include='object')):
- The sex column contains 2 unique values: male and female, with male being the most common (288 passengers).
- The name column has 499 unique names out of 500 entries, indicating there is 1 duplicate name.

DATA DUPLICATE HANDLING

```
[10] # Menampilkan baris duplikat
     duplicates = df[df.duplicated(keep=False)]

     duplicate_counts = duplicates.groupby(list(df.columns)).size().reset_index(name='jumlah_duplikat')

     sorted_duplicates = duplicate_counts.sort_values(by='jumlah_duplikat', ascending=False)

     print("Baris yang terduplikasi:")
     sorted_duplicates
```

```
Baris yang terduplikasi:
```

	survived	name	sex	age	jumlah_duplikat
0	1	Eustis, Miss. Elizabeth Mussey	female	54.0	2

```
[11] # Hapus duplikat
     df = df.drop_duplicates()
```

```
len(df.drop_duplicates()) / len(df)
```

```
1.0
```

To check for duplicate data
Findings:

- Before cleaning, 99.8% of the rows were unique, meaning there was 1 duplicate row.
- The duplicate was from Eustis, Miss. Elizabeth Mussey, aged 54.0, with a survival status of 1.
- This exact row appeared twice in the dataset.

Action Taken:

- The duplicate was removed using df.drop_duplicates().
- After this step, the dataset became 100% unique, ensuring there are no repeated entries.

DATA DUPLICATE HANDLING

```python
# Cek jumlah dan persentase missing value
missing = df.isnull().sum()
missing_percent = (missing / len(df)) * 100

print("Jumlah Missing Value:")
print(missing)
print("\nPersentase Missing Value:")
print(missing_percent)

# Handling: isi missing value pada kolom 'age' dengan median
median_age = df['age'].median()
df['age'].fillna(median_age, inplace=True)

print("\nMissing value pada kolom 'age' telah diisi dengan nilai median:", median_age)
```

```
Jumlah Missing Value:
survived     0
name         0
sex          0
age         49
dtype: int64

Persentase Missing Value:
survived    0.000000
name        0.000000
sex         0.000000
age         9.819639
dtype: float64

Missing value pada kolom 'age' telah diisi dengan nilai median: 35.0
<ipython-input-13-8336c31301e2>:12: FutureWarning: A value is trying to be set on a copy of a DataFrame or Series through chai
The behavior will change in pandas 3.0. This inplace method will never work because the intermediate object on which we are se

For example, when doing 'df[col].method(value, inplace=True)', try using 'df.method({col: value}, inplace=True)' or df[col] =
```

Findings:

- The dataset contains 4 columns: survived, name, sex, and age.

Based on the missing value inspection:

- The age column contains 49 missing values.
- This represents 9.819639 (9.82%) of the total dataset.
- The other columns (survived, name, and sex) do not have any missing values.

Handling:

- To handle the missing values in the age column:
- The median value of the age column was calculated using df['age'].median().
- The missing entries in the age column were then filled with this median value using df['age'].fillna(median_age, inplace=True).

DATA DUPLICATE HANDLING

To check for duplicate data
Findings:

- Before cleaning, 99.8% of the rows were unique, meaning there was 1 duplicate row.
- The duplicate was from Eustis, Miss. Elizabeth Mussey, aged 54.0, with a survival status of 1.
- This exact row appeared twice in the dataset.

Action Taken:

- The duplicate was removed using df.drop_duplicates().
- After this step, the dataset became 100% unique, ensuring there are no repeated entries.

FINAL RESULT MISSING VALUE HANDLING

- The age column no longer contains missing values.
-
- The median value used for imputation was 35.0, as displayed in the final print output.
-
- This process ensures data consistency by replacing missing values with a statistically representative value, minimizing potential bias.