Rakamin Academy

Machine Learning Preparation

Exploratory Data Analysis

Outline Pembelajaran

Rakamin Academy

Exploratory Data Analysis

- Exploratory Data Analysis (EDA) itu apa?
- How to EDA
- Hands-on: studi kasus Pengiriman Makanan

Hands-On Required :

Hands - On :

Hands-On - Exploratory Data Analysis.ipynb

Dataset :

1. Food_Delivery_Dataset.csv

Klik disini untuk mengakses folder Hands-On dan Dataset

Apa itu Exploratory Data Analysis (EDA)?

Data juga ingin dipahami

Rakamin Academy

“Exploratory Data Analysis (EDA) adalah proses analisis untuk memahami karakteristik data, dan hal-hal yang perlu kita lakukan agar data tersebut dapat digunakan untuk proses pembelajaran model ”

Tahap	Masuk	Proses	Keluar
Data collection	-	• Survey/Labelling • ETL	• Data mentah
Data understanding	• **Data mentah**	• **Exploratory Data Analysis**	• **Data mentah** • **Insight (?)**
Data preparation	• Data mentah • **Insight (?)**	• Pre-processing • Feature processing	• Data *training* • Data *test/validation*
Modelling	• Data *training*	• Model training • Hyperparameter tuning	• ML Model
Evaluation	• Data *test/ validation*	• Validation	• Performance measure
Deployment	• Data baru	• Prediction	• Prediksi

Insight apa yang kita inginkan dari EDA? Insight yang akan membantu kita melatih model ML yang lebih baik

Pertanyaan yang perlu kita jawab:

- Bagaimana **sebaran nilai** dalam feature dan label kita?
- Apakah kira-kira feature yang kita miliki **cukup baik untuk memprediksi** target?
- **'Persiapan' macam apa** yang harus kita lakukan sebelum dataset kita dapat digunakan dalam proses pelatihan model ML?

Insight apa yang kita inginkan dari EDA? Insight yang akan membantu kita melatih model ML yang lebih baik

Hal yang *tidak* kita lakukan dalam EDA:

- Mencari ‘business insight’ atau ‘story’ dari data yang dimiliki
- ‘Mengubah’ atau ‘membersihkan’ data; dipahami saja jangan disentuh dulu

Outline Pembelajaran

Rakamin Academy

Exploratory Data Analysis

- [x] Exploratory Data Analysis (EDA) itu apa?
- [ ] How to EDA
- [ ] Hands-on: studi kasus House Rent

Bagaimana cara melakukan EDA?

Dataset

Titanic

- **Deskripsi**:

  Memprediksi *survival* dari kecelakaan Titanic berdasarkan data-data penumpang.

- **Data**:

  Setiap baris mewakili penumpang, setiap kolom berisi atribut penumpang.

  Data dipisah menjadi 2, ambil `train.csv`

- **Link Kaggle**: https://www.kaggle.com/c/titanic/data

Load Data

```
1 import numpy as np
2 import pandas as pd
3 import matplotlib.pyplot as plt
4 import seaborn as sns
5 from matplotlib import rcParams
6
7 print('Numpy version: ', np.__version__)
8 print('Pandas version: ',pd.__version__)
9 print('Seaborn version: ',sns.__version__)
```

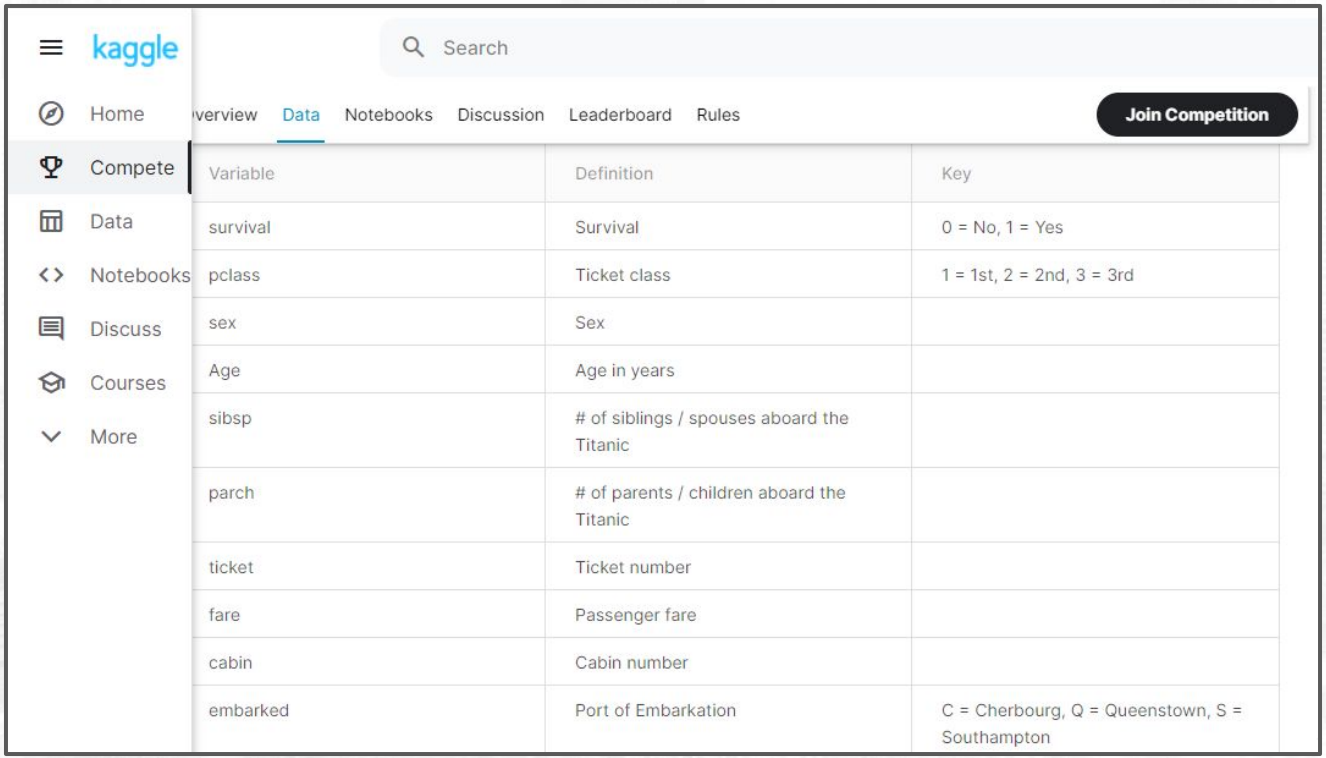
```
Numpy version:  1.18.5
Pandas version:  1.0.5
Seaborn version:  0.10.1
```

```
1 rcParams['figure.figsize'] = (10,7)
2 rcParams['lines.linewidth'] = 2.5
3 rcParams['xtick.labelsize'] = 'x-large'
4 rcParams['ytick.labelsize'] = 'x-large'
```

```
1 df = pd.read_csv('train.csv')
```

`rcParams` mengubah pengaturan *default* `matplotlib`

Understand The Data



Variable	Definition	Key
survival	Survival	0 = No, 1 = Yes
pclass	Ticket class	1 = 1st, 2 = 2nd, 3 = 3rd
sex	Sex	
Age	Age in years	
sibsp	# of siblings / spouses aboard the Titanic	
parch	# of parents / children aboard the Titanic	
ticket	Ticket number	
fare	Passenger fare	
cabin	Cabin number	
embarked	Port of Embarkation	C = Cherbourg, Q = Queenstown, S = Southampton

Bagaimana cara melakukan EDA?

#1: Descriptive Statistics

Rakamin Academy

***Descriptive Statistics*:**

Ringkasan statistik dari setiap kolom di dataset yang dapat memberikan gambaran besar keadaan data.

Column, NULL Values, and Dtypes

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

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 891 entries, 0 to 890
Data columns (total 12 columns):
 #   Column       Non-Null Count  Dtype
---  ------       --------------  -----
 0   PassengerId  891 non-null    int64
 1   Survived     891 non-null    int64
 2   Pclass       891 non-null    int64
 3   Name         891 non-null    object
 4   Sex          891 non-null    object
 5   Age          714 non-null    float64
 6   SibSp        891 non-null    int64
 7   Parch        891 non-null    int64
 8   Ticket       891 non-null    object
 9   Fare         891 non-null    float64
 10  Cabin        204 non-null    object
 11  Embarked     889 non-null    object
dtypes: float64(2), int64(5), object(5)
memory usage: 83.7+ KB
```

`df.info()` akan memunculkan informasi umum mengenai dataframe

Yang perlu diperhatikan:

- **Apakah ada data dengan tipe yang kurang sesuai?**
  Misal, `Name` dengan tipe `int64`
- **Apakah ada data yang hilang?**
  Cari kolom dengan count < jumlah row

Pick + Separate Columns

```
1 numericals = ['Survived', 'Pclass', 'Age', 'SibSp', 'Parch', 'Fare']
2 categoricals = ['Sex', 'Embarked']
```

Pisahkan kolom2 yang ingin dianalisis

Sample

```
1 df.sample(5)
```

`df.sample()`, `df.head()`, atau `df.tail()` akan menampilkan beberapa baris data secara langsung

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
464	465	0	3	Maisner, Mr. Simon	male	NaN	0	0	A/S 2816	8.0500	NaN	S
838	839	1	3	Chip, Mr. Chang	male	32.0	0	0	1601	56.4958	NaN	S
93	94	0	3	Dean, Mr. Bertram Frank	male	26.0	1	2	C.A. 2315	20.5750	NaN	S
442	443	0	3	Petterson, Mr. Johan Emil	male	25.0	1	0	347076	7.7750	NaN	S
386	387	0	3	Goodwin, Master. Sidney Leonard	male	1.0	5	2	CA 2144	46.9000	NaN	S

Yang perlu diperhatikan:

- **Apakah ada kolom dengan nilai yang tidak sesuai dengan nama kolom?**

PROTIP: Memisahkan Berdasarkan Data Types

```
1 num_dtypes = ['int16', 'int32', 'int64', 'float16', 'float32', 'float64']
2
3 # Langsung subset df
4 num_df = df.select_dtypes(include=num_dtypes)
5 numericals = num_df.columns
6
7 print(numericals)
```

```
Index(['PassengerId', 'Survived', 'Pclass', 'Age', 'SibSp', 'Parch', 'Fare'], dtype='object')
```

Kita bisa langsung mengambil hanya kolom-kolom dengan tipe data tertentu!

Pandas dtype	Python type	NumPy type	Usage
object	str or mixed	string_, unicode_, mixed types	Text or mixed numeric and non-numeric values
int64	int	int_, int8, int16, int32, int64, uint8, uint16, uint32, uint64	Integer numbers
float64	float	float_, float16, float32, float64	Floating point numbers
bool	bool	bool_	True/False values
datetime64	NA	datetime64[ns]	Date and time values
timedelta[ns]	NA	NA	Differences between two datetimes
category	NA	NA	Finite list of text values

Statistical Summary (1)

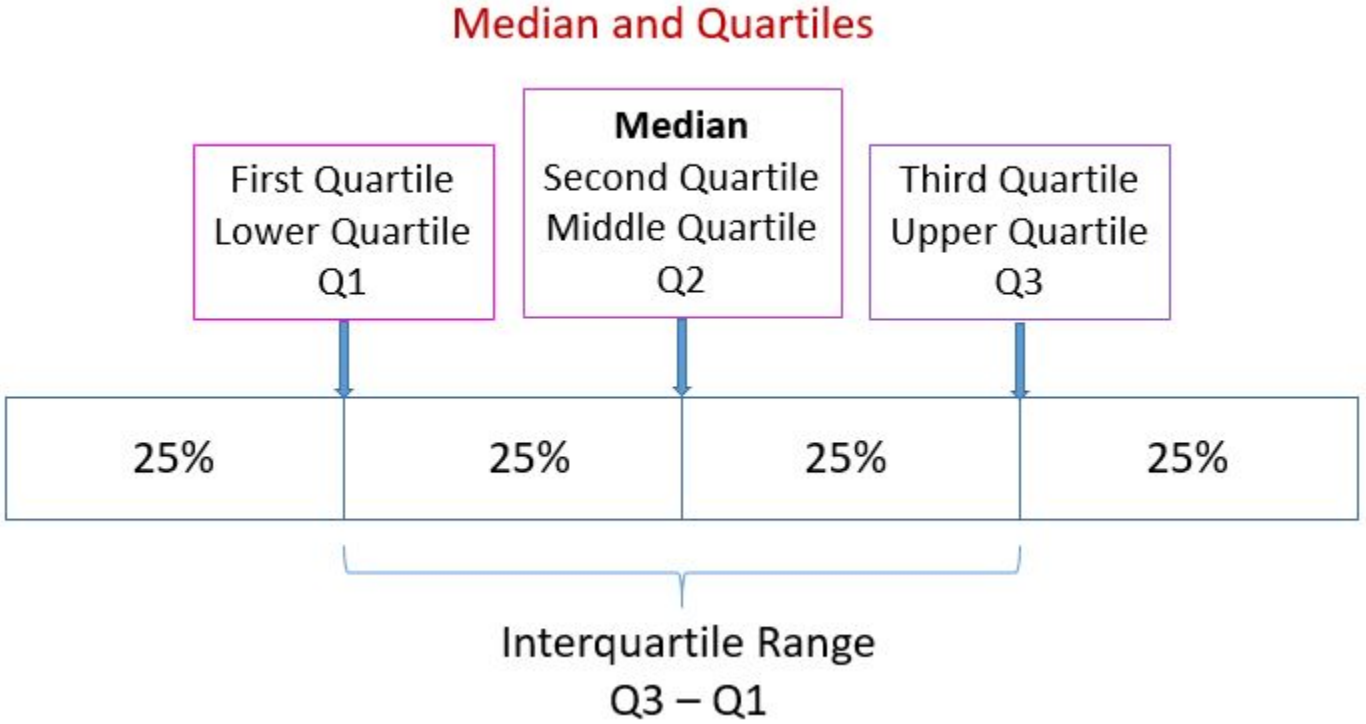
```
1 df.describe()
```

	PassengerId	Survived	Pclass	Age	SibSp	Parch	Fare
count	891.000000	891.000000	891.000000	714.000000	891.000000	891.000000	891.000000
mean	446.000000	0.383838	2.308642	29.699118	0.523008	0.381594	32.204208
std	257.353842	0.486592	0.836071	14.526497	1.102743	0.806057	49.693429
min	1.000000	0.000000	1.000000	0.420000	0.000000	0.000000	0.000000
25%	223.500000	0.000000	2.000000	20.125000	0.000000	0.000000	7.910400
50%	446.000000	0.000000	3.000000	28.000000	0.000000	0.000000	14.454200
75%	668.500000	1.000000	3.000000	38.000000	1.000000	0.000000	31.000000
max	891.000000	1.000000	3.000000	80.000000	8.000000	6.000000	512.329200

`df.describe()` akan menampilkan beberapa statistik dari setiap kolom dalam dataframe. Secara *default* hanya kolom numerik saja yang akan ditampilkan.

Rakamin Academy

Statistika Deskriptif



Statistical Summary (2)

Yang perlu diperhatikan:

- **Apakah nilai yang tertera pada setiap kolom masuk akal?**
- **Apakah nilai maksimal/minimal masih berada di batas wajar?**
  Min/max yang terlalu jauh dari *mean/median* bisa jadi indikasi kesalahan input data
- **Apakah ada kolom dengan perbedaan yang signifikan antara *mean* dan *median*?**
  Perbedaan antara mean/median mengindikasikan *outlier* atau *skewed distribution*

```
1 df[categoricals].describe()
```

	Sex	Embarked
count	891	889
unique	2	3
top	male	S
freq	577	644

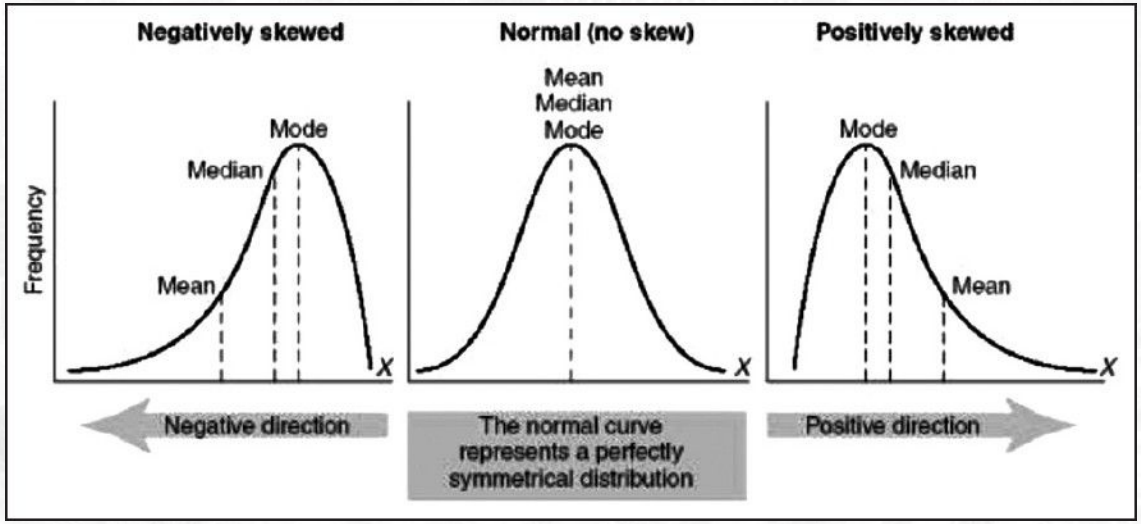
Apabila semua kolom di dataframe bertipe categorical, `df.describe()` akan memunculkan statistik kategori.

Yang perlu diperhatikan:

- **Apakah jumlah *unique values* masuk akal?**
- **Apakah frekuensi dari nilai yang paling umum terlalu timpang?**
  Apabila distribusi nilai terlalu timpang, *feature* tidak akan terlalu berguna dalam klasifikasi

Statistika Deskriptif

Tipe-tipe Distribusi



Value Counting

```
1 df['Sex'].value_counts()
```

```
male      577
female    314
Name: Sex, dtype: int64
```

See Unique Values

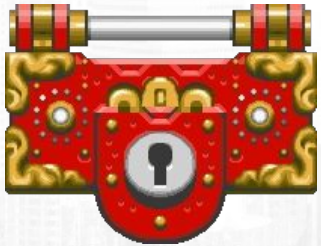
```
1 df['Sex'].unique()
```

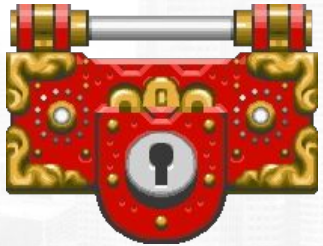
```
array(['male', 'female'], dtype=object)
```

`df['nama_kolom'].value_counts()` akan menampilkan jumlah dari setiap nilai unik di kolom tersebut.

`df['nama_kolom'].unique()` menampilkan semua nilai unik di kolom.

INI PENTING TERUTAMA UNTUK TARGET DI SUPERVISED LEARNING!





Masalah apa yang bisa diidentifikasi hanya dengan statistika deskriptif?

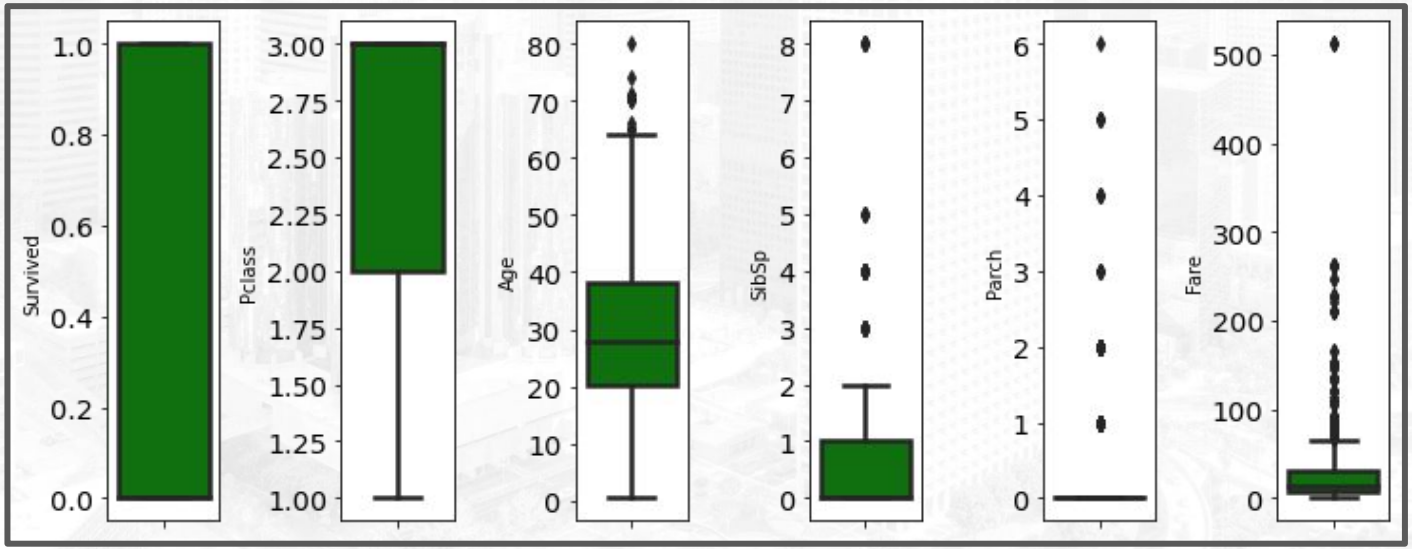
Bagaimana cara melakukan EDA?

#2: Univariate Analysis

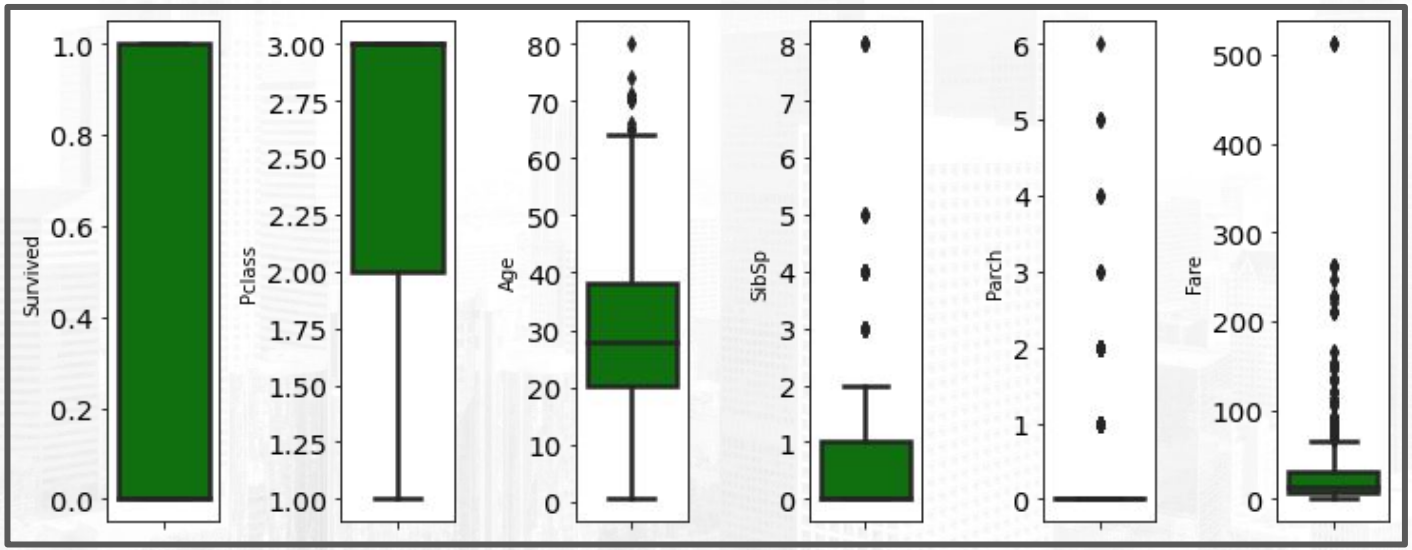
Univariate Analysis:
Analisis setiap kolom secara terpisah, melihat distribusi nilainya secara detail

Individual Boxplots (Numeric)

```
features = numericals
for i in range(0, len(features)):
    plt.subplot(1, len(features), i+1)
    sns.boxplot(y=df[features[i]], color='green', orient='v')
    plt.tight_layout()
```



Individual Boxplots (Numeric)

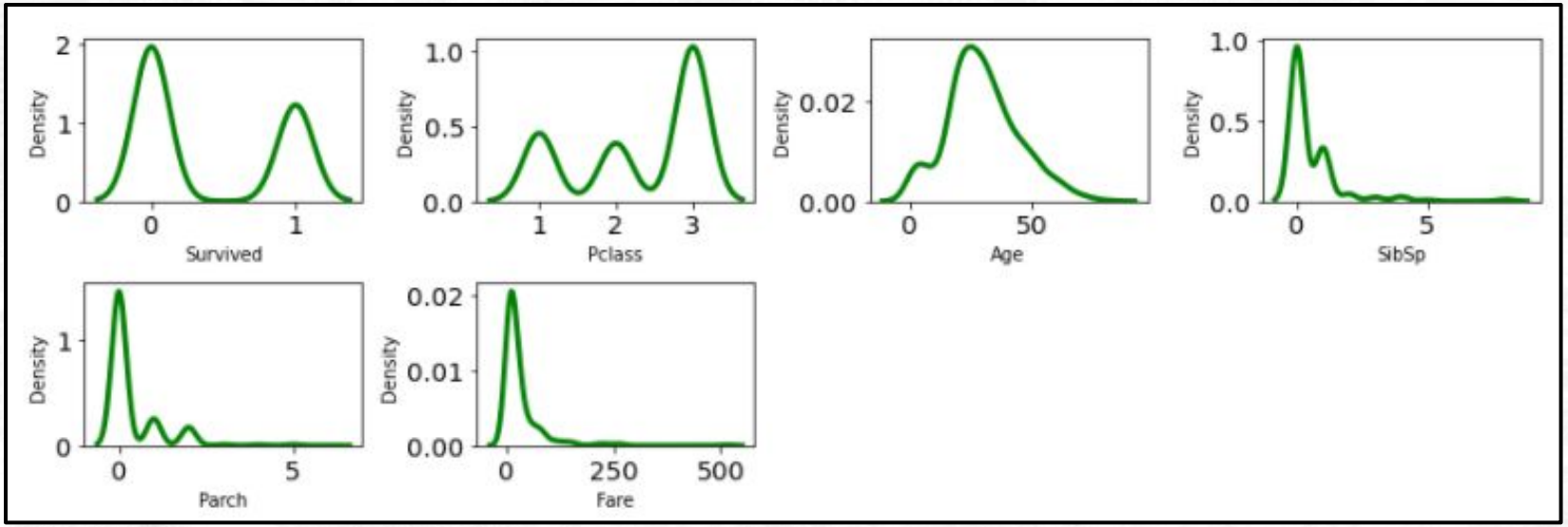


Yang perlu diperhatikan:

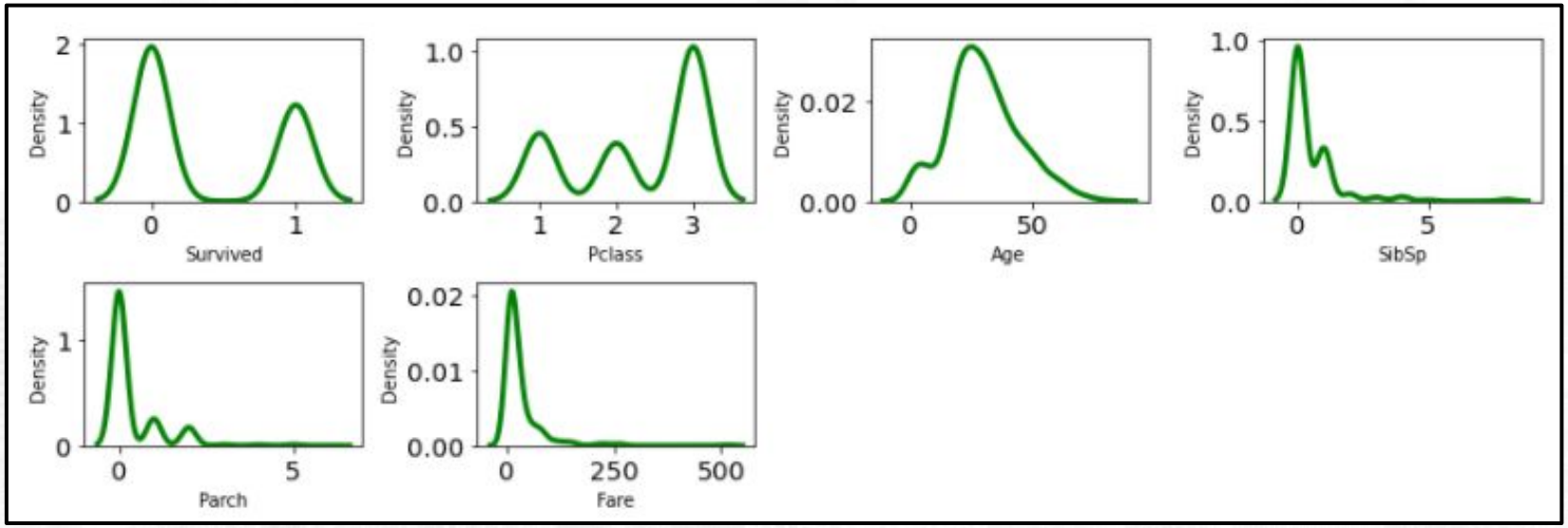
- **Berapa banyak dan berapa jauh *outlier* di setiap kolom?**
  Definisi umum *outlier*: berjarak 1.5x IQR dari Q1/Q3

Individual Distplot (Numerical)

```
features = numericals
for i in range(0, len(features)):
    plt.subplot(2, 4, i+1) # 2x4=8 > len(numericals)=7
    sns.kdeplot(x=df[features[i]], color='green')
    plt.xlabel(features[i])
    plt.tight_layout()
```



Individual Distplot (Numerical)

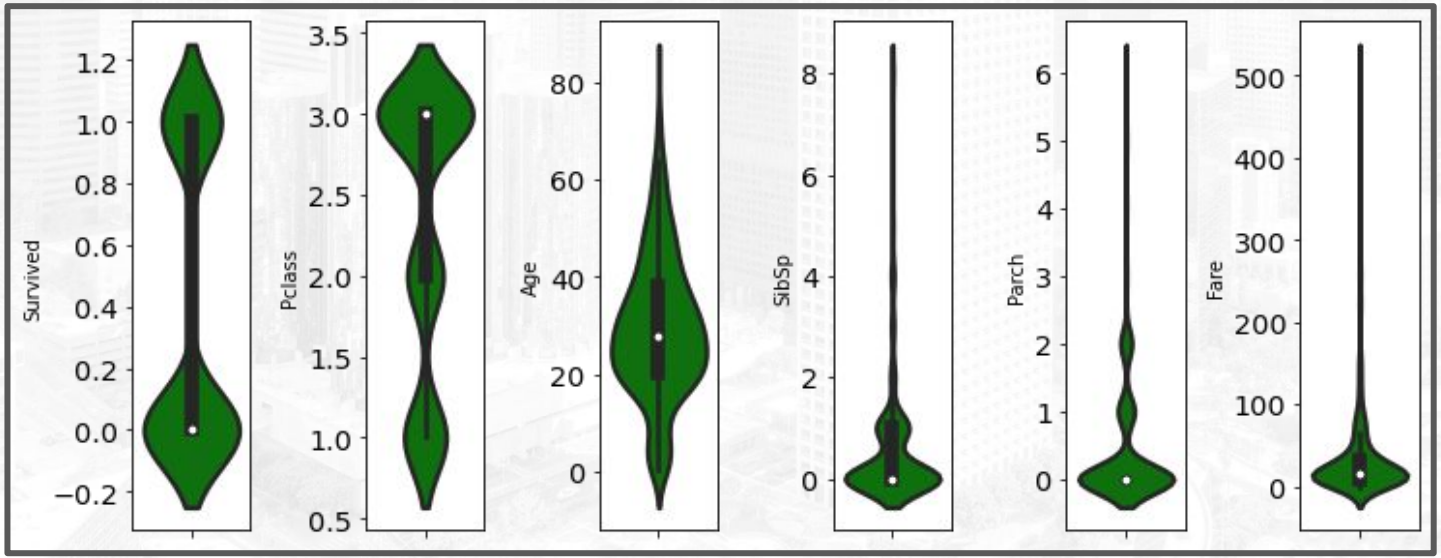


Yang perlu diperhatikan:

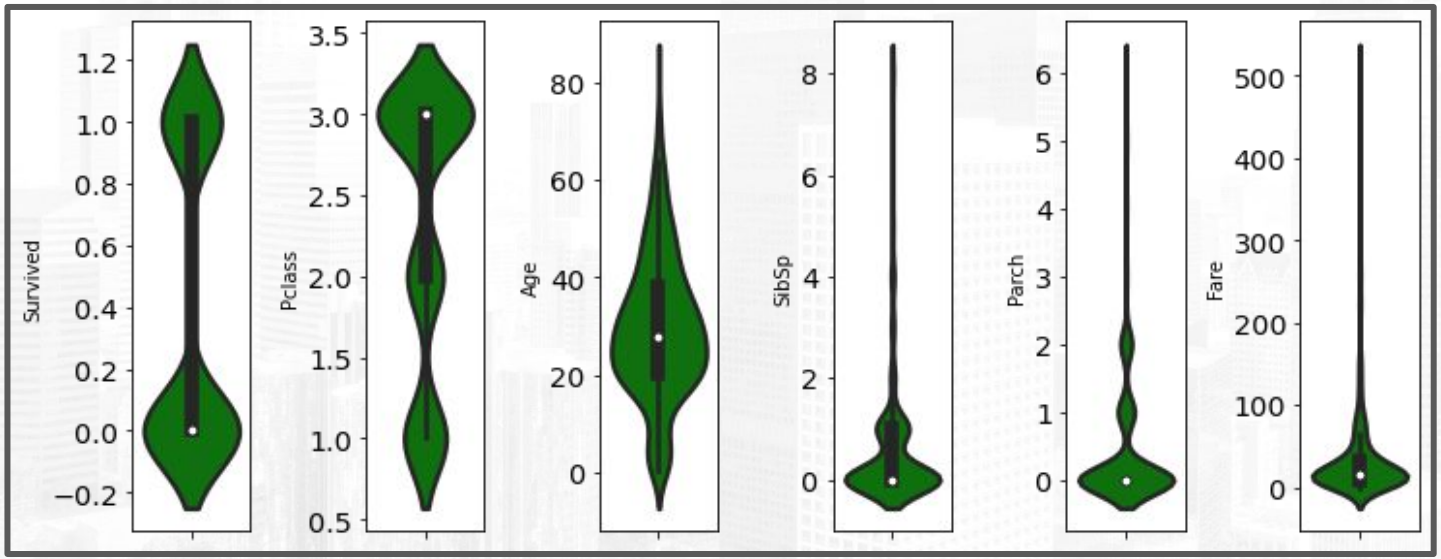
- **Bagaimana bentuk distribusi setiap kolom?**
  Apakah normal? *Positive skewed? Negative skewed*? *Bimodal*?
- **Apakah ada nilai-nilai tertentu yang umum?**
  Lonjakan pada distribution plot mungkin memiliki makna tertentu

Alternatif: Individual Violin Plots (Numeric)

```
features = numericals
for i in range(0, len(features)):
    plt.subplot(1, len(features), i+1)
    sns.violinplot(y=df[features[i]], color='green')
    plt.tight_layout()
```



Individual Violin Plots (Numeric)



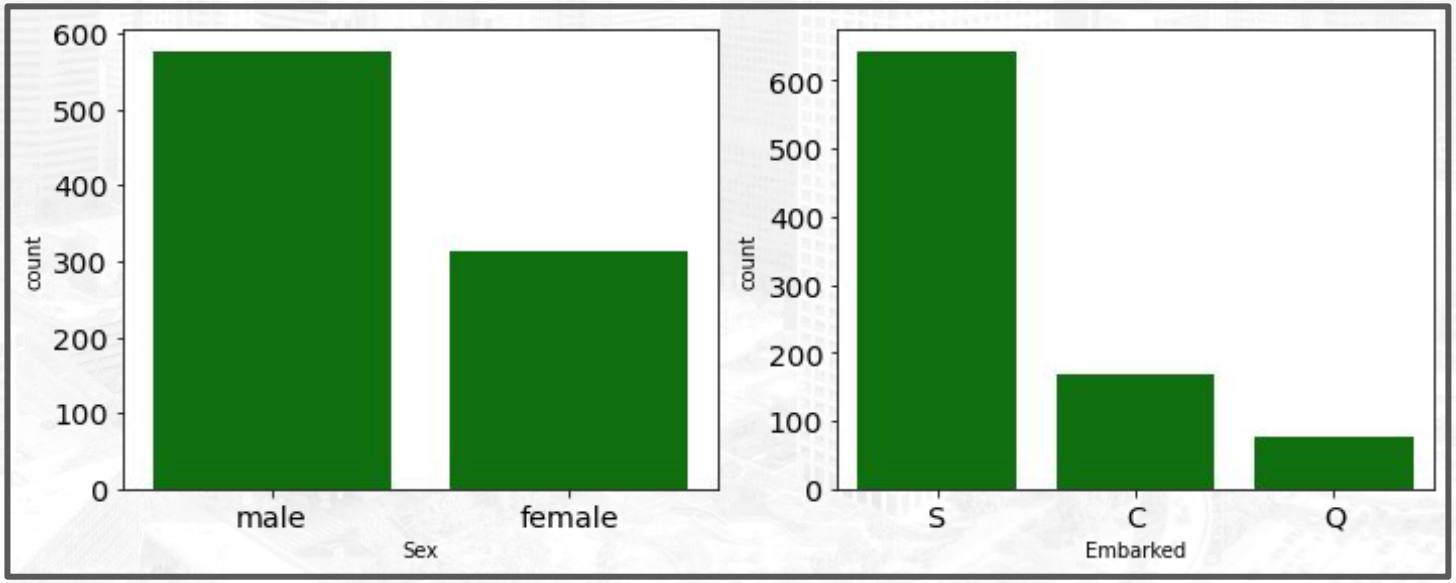
Violin plot merupakan gabungan antara box plot dan distribution plot

Yang perlu diperhatikan:

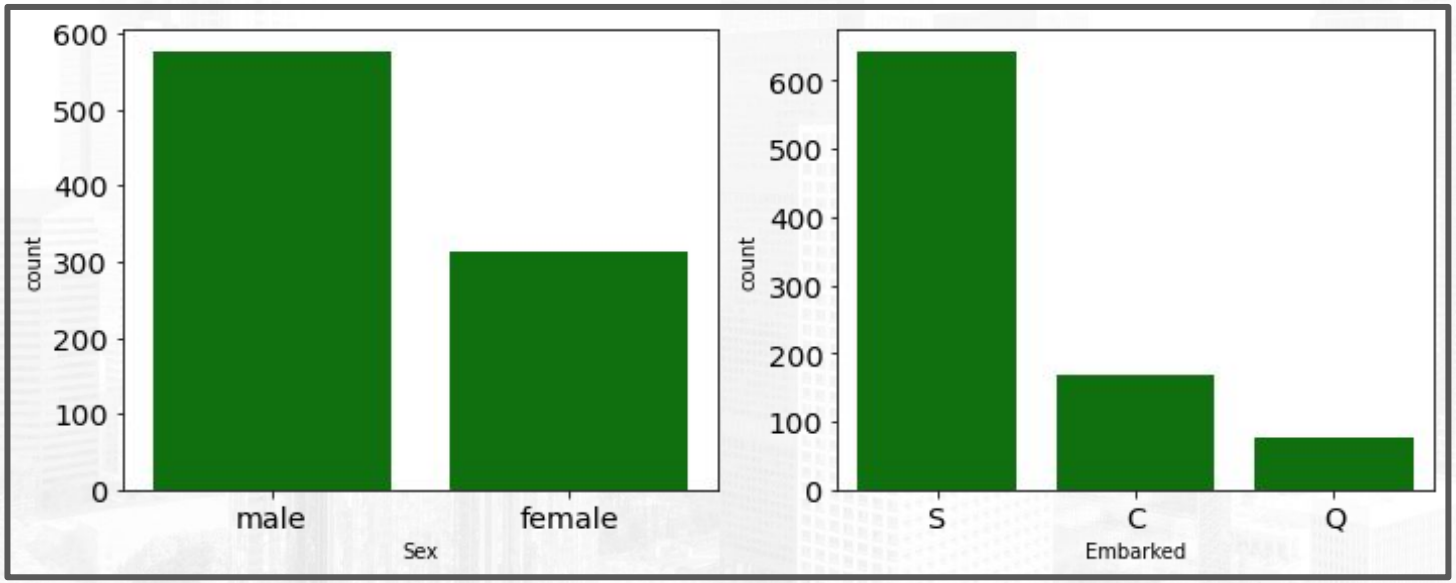
- **Sama dengan box plot + distribution plot**

Individual Countplot (Categorical)

```
features = categoricals
for i in range(0, len(features)):
    plt.subplot(1, len(features), i+1)
    sns.countplot(x=df[features[i]], color='green')
    plt.tight_layout()
```



Individual Countplot (Categorical)



Untuk data kategori, kita bisa menggunakan countplot biasa

Yang perlu diperhatikan:

- **Ketimpangan antar kategori, terutama pada *target***
  Sebaran kategori yang timpang pada feature merupakan indikasi ketidakgunaan feature. Pada target, sebaran yang timpang dapat membuat proses *learning* gagal.

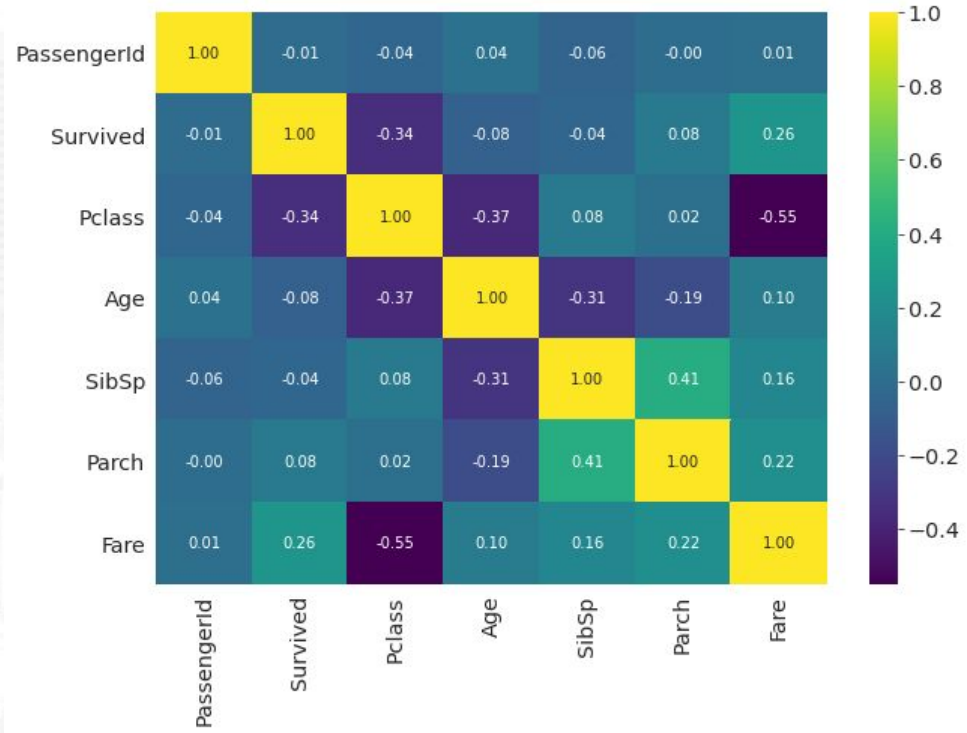
Bagaimana cara melakukan EDA?

#3: Multivariate Analysis

Multivariate Analysis:
Analisis beberapa kolom sekaligus untuk mencari hubungan antar kolom

Correlation Heatmap (Numeric)

```
1 sns.heatmap(df.corr(), cmap='Blues', annot=True, fmt='.2f')
```

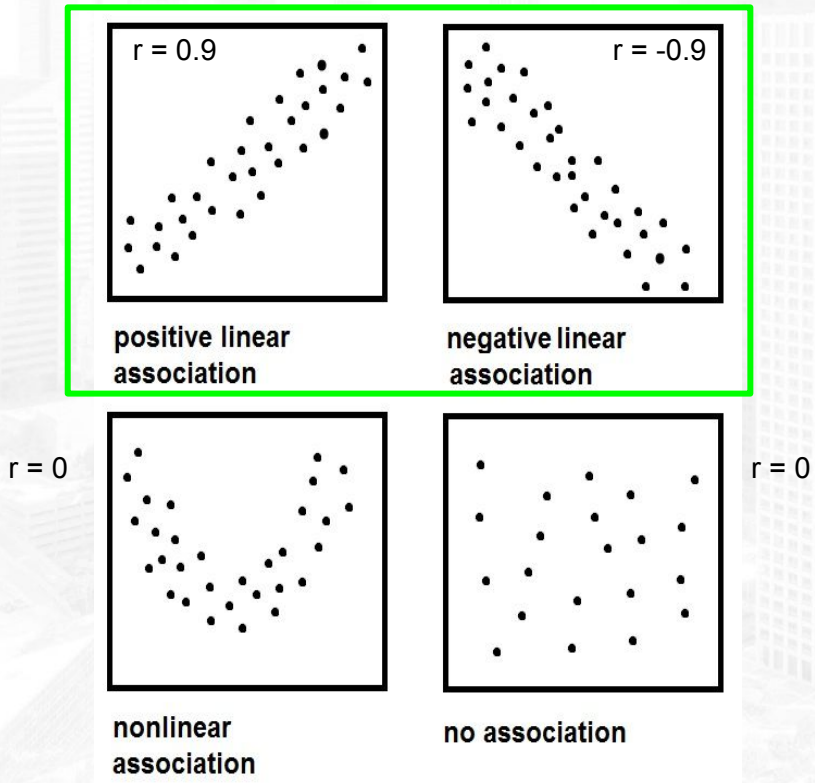


`df.corr()` akan mengembalikan matriks korelasi; `sns.heatmap()` membuat heatmap berdasarkan matriks

Yang perlu diperhatikan:

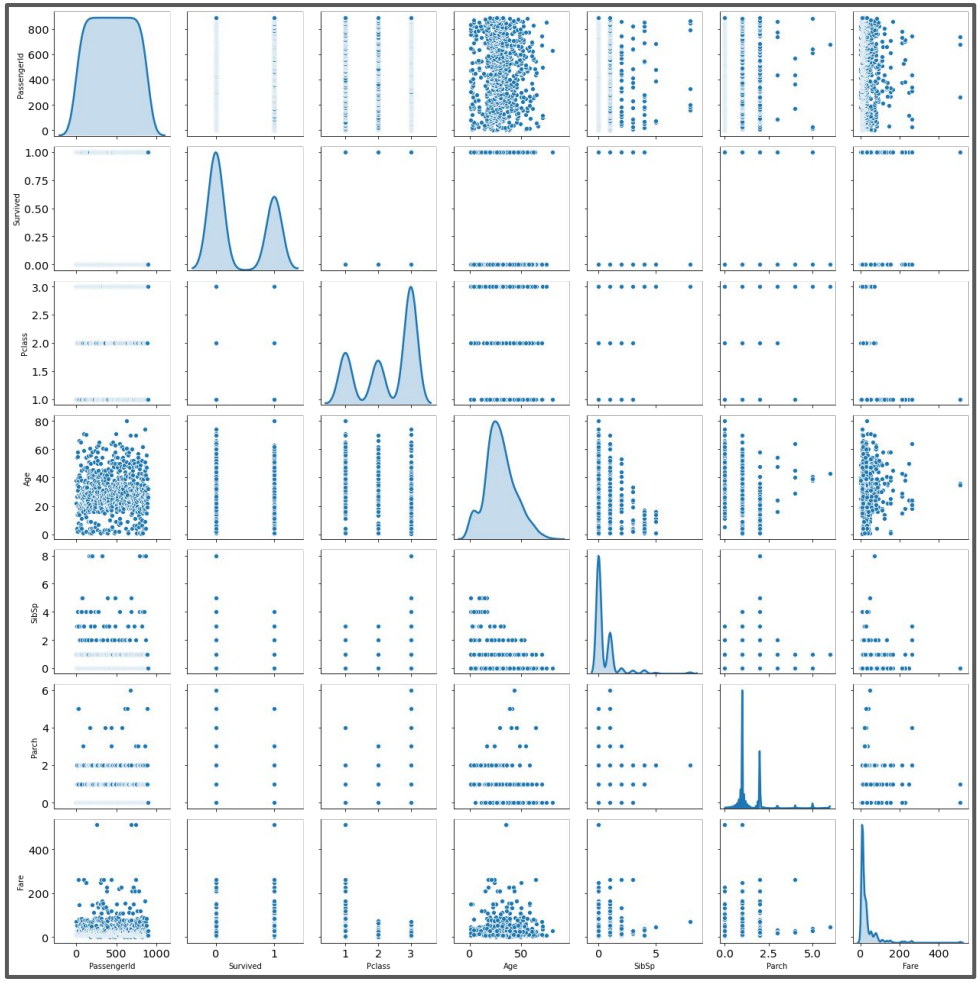
- **Apakah feature memiliki korelasi dengan target?**
  Bila tidak, maka model linear tidak dapat digunakan
- **Apakah ada 2 feature yang berkorelasi kuat (>0.7)?**
  Bila ya, ada kemungkinan besar kedua feature tersebut redundan

Korelasi Linear (Pearson correlation)



- **Pola hubungan antara X dan Y membentuk pola garis lurus**
  Semakin besar X, semakin besar pula Y ATAU semakin besar X, semakin kecil Y, dengan *rate* konstan (linear) (kotak hijau di samping)

- **Nilai korelasi berkisar dari -1 s.d. 1**
  1: hubungan linear sempurna, searah
  -1 : hubungan linear sempurna namun berlawanan arah
  0: pola hubungan BUKAN linear



Pair Plots (Numeric)

```
1 sns.pairplot(df, diag_kind='kde')
```

`sns.pairplot()` secara otomatis menggambar scatter plot untuk setiap pasangan kolom

Yang perlu diperhatikan:

- **Apakah ada scatter plot yang menunjukkan *cluster/pattern* yang cukup jelas?**
  Pola pada scatter plot bisa menjadi petunjuk untuk memilih fitur yang baik

Pair Plots + Hue (Numeric)

```
1 sns.pairplot(df, diag_kind='kde',
2                  hue='Survived')
```

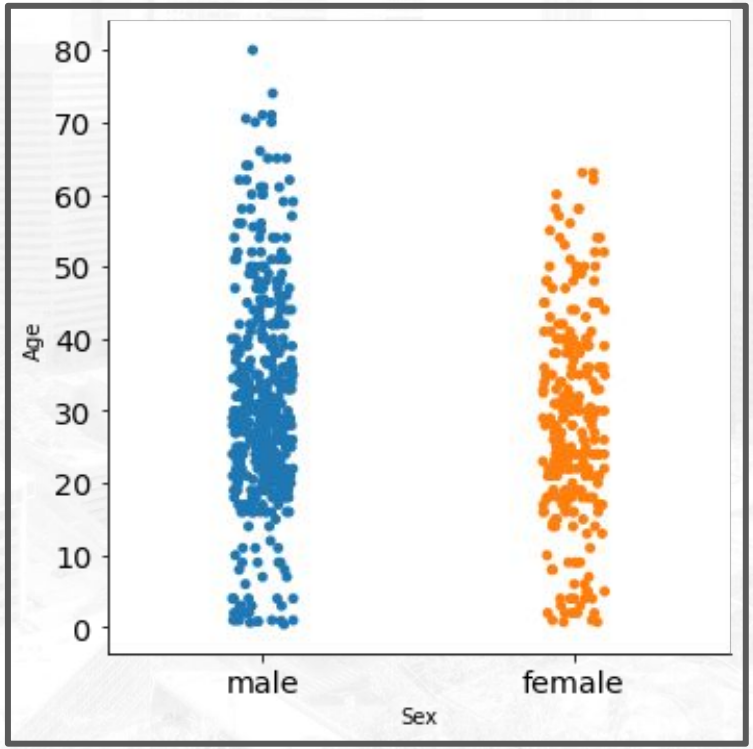
parameter `hue` memberikan warna yang berbeda untuk setiap kemungkinan nilai target

Yang perlu diperhatikan:

- **Apakah ada scatter plot dimana kedua warna terpisah dengan baik?**
  Scatter plot dimana target terpisah dapat menjadi indikasi kombinasi fitur yang baik

Category Plots (Categorical - Numeric)

```
1 sns.catplot(data=df, x='Sex', y='Age')
```



Tidak ada fungsi otomatis untuk menggambarkan semua kombinasi categorical vs numeric value. Harus dilakukan manual dengan `sns.stripplot()`.

Parameter `hue` juga tersedia disini.

Yang perlu diperhatikan:

- **Apakah feature categorical memiliki korelasi dengan target?**
- **Apakah ada 2 feature categorical yang terlihat jelas berkorelasi?**

EDA: Tips + Trik

1. Fokus pada kecepatan dan kelengkapan eksplorasi, bukan estetika visualisasi
2. Jangan menggali business insight - EDA bertujuan untuk mempermudah proses modeling

Outline Pembelajaran

Rakamin Academy

Exploratory Data Analysis

- [x] Exploratory Data Analysis (EDA) itu apa?
- [x] How to EDA
- [ ] Hands-on: studi kasus Pengiriman Makanan

Hands-on: EDA Prediksi Waktu Kedatangan Pesanan

Dataset

Food_Delivery_Dataset.csv

- **Deskripsi**:

  Sebuah perusahaan pengiriman makanan ingin meningkatkan sistemnya dalam menghitung waktu kedatangan pesanan. Manajemen telah memutuskan untuk mengembangkan perangkat lunak cerdas yang dapat memprediksi waktu kedatangan pesanan.

  Dataset. memprediksi waktu kedatangan pesanan.

- **Data**:

  Setiap baris mewakili satu pesanan, setiap kolom berisi atribut pesanan tersebut.

Dataset

Column Name	Description
ID	Represents a unique identification of an entry
Delivery person ID	Represents a unique identification of a delivery person.
Delivery person Age	Represents the age of a delivery person.
Delivery person Ratings	Represents the average ratings given to the delivery person. (1to5)
Restaurant latitude	Represents the latitude of the restaurant.
Restaurant longitude	Represents the longitude of the restaurant.
Delivery location latitude	Represents the latitude of the Delivery location.
Delivery location longitude	Represents the longitude of the Delivery location.
Order Date	Represents the date when the order was placed.
Time Orderd	Represents the time when the order was placed.
Time Order picked	Represents the time when the order was picked from the restaurant.

Dataset

Column Name	Description
Weather conditions	Represent the weather conditions (Windy, Sunny, Cloudy, Stormy, Fog, Sandstorms, etc)
Road traffic density	Represents the road traffic density (Jam, High, Medium and Low)
Vehicle condition	Represents the condition of the vehicle. (Smooth, good or average)
Type of order	Represents the type of order (Snack, Meal, Buffet, Drinks, etc)
Type of vehicle	Represents the type of vehicle one is using (motorbike, bicycle etc.)
multiple deliveries	Represents the number of orders to be delivered in one attempt
Festival	Represents whether day is festive or not
City	Represents the city
Time taken	Represents the time taken by the delivery person to deliver the order.

Sudah.

Sesi tanya-jawab