Exploratory Data Analysis (EDA)

系统分析数据的方式，找到数据中隐藏模式和见解

- 制定问题：
  - 找到数据中感兴趣的变量，诸如利润、存活率等
  - 了解相关性，对不同变量之间的关系有猜想

EDA Check List：

- **明确目标**：你想要解决（或证明是错的）什么问题？清楚地说明你打算用数据解决的问题或问题。有一个明确的目标来指导分析并确保重点。
- **理解数据类型**：您拥有哪些类型的数据？您如何处理不同类型的数据？识别数据集中存在的数据类型（例如，数字，分类），并了解如何适当地处理每种类型。
- **处理丢失的数据**：数据中丢失了什么，您如何处理它（例如，插入、删除）？
- **发现异常值**：异常值在哪里？为什么要关注它们？异常值可能会扭曲结果，需要对其相关性和潜在处理进行评估。
- **特征工程**：如何添加、更改或删除特征以从数据中获得更多？

```
# List all columns
df.columns
```

```
# datatypes of each column
df.dtypes

# Generate statistics for individual columns
df.describe()

# Check missing values
df.info()

# Box and whisker chart
fig, ax = plt.subplots()
ax.boxplot(df['Age'].dropna())
plt.show()

# histogram
df.Age.plot.hist()
# plt.hist(df['Age'].dropna(), bins=20, edgecolor='k')

# scatter
df.plot.scatter("col1", "col2", alpha=.5)

# Correlation
df.corr()

# Cross-tabulation tables: categorical variables
pd.crosstab(df["col1"], df["col2"])

# Functions
df["col"].function()
df["col"].apply(func)

# Filter
df[df_condition]["col"].function()

# Group and Aggregating
df.groupby('condition')["col"].function()
```

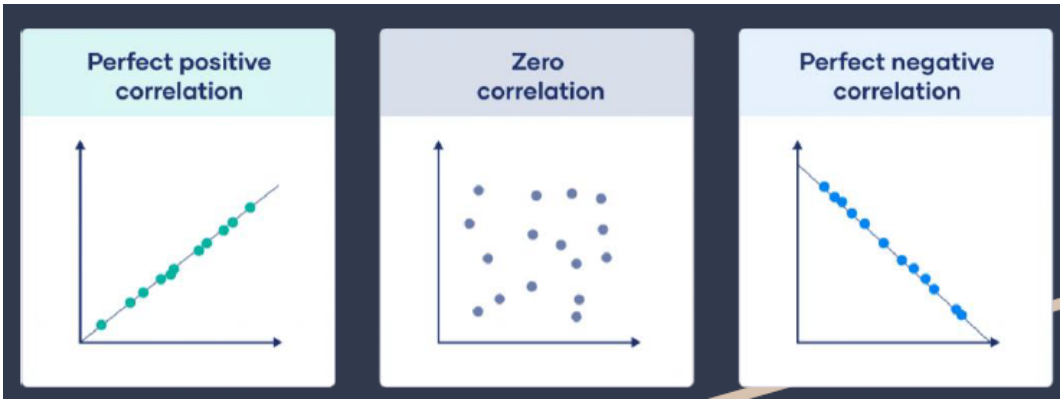
```
# In-place operation
df["col"].function(value, inplace=True)

# Non-in-place operation
df_copy = df["col"].function(value, inplace=False)
```

步骤：

- **Basic Data Manipulation**：对数据类型或值进行更改，以方便处理。比如将日期修改成从某一天算起的天数，时间转换。
- **Understand individual column**：通过describe或画图的方式理解数据的统计状况
- **Distributions**：了解数据的分布
- **事件频率**：用于观察事件发生的次数，通常用散点图表示，且基于一个诸如时间、季节性的连续变量。
- **Pearson相关性**：表示变量之间的相似性，值域为$[-1,1]$，公式为

$$r = \frac{\sum(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum(x_i - \bar{x})^2 \sum(y_i - \bar{y})^2}}$$



- **统计重要性**：如果两个数据集有不同的均值，我们不能假设它们来自同一个数据集，通常要做ttest，如果p值小于0.05，则两个group的均值有显著的区别

```
t_statistic, p_value = stats.ttest_ind(group1, group2, equal_
```

- **错误检测**：
  - 重复值：比较len(df)和len(df.drop_duplicates())的值

    - 缺失值：查看sum(df[″column″].isna())
    - 离群值

目标：

- 理解所有数据的范围：找到离群值，异常，极值，并进一步研究
- 了解变量的相关和独立性
- 知道错误所在：解决缺失值，数据输入错误，离群值
- 了解是否需要其它外部数据