INT303 W3 Note

The basic EDA Workflow

1. Build a DataFrame from the data (ideally, put all data in this object)

2. Clean the DataFrame. It should have the following properties:

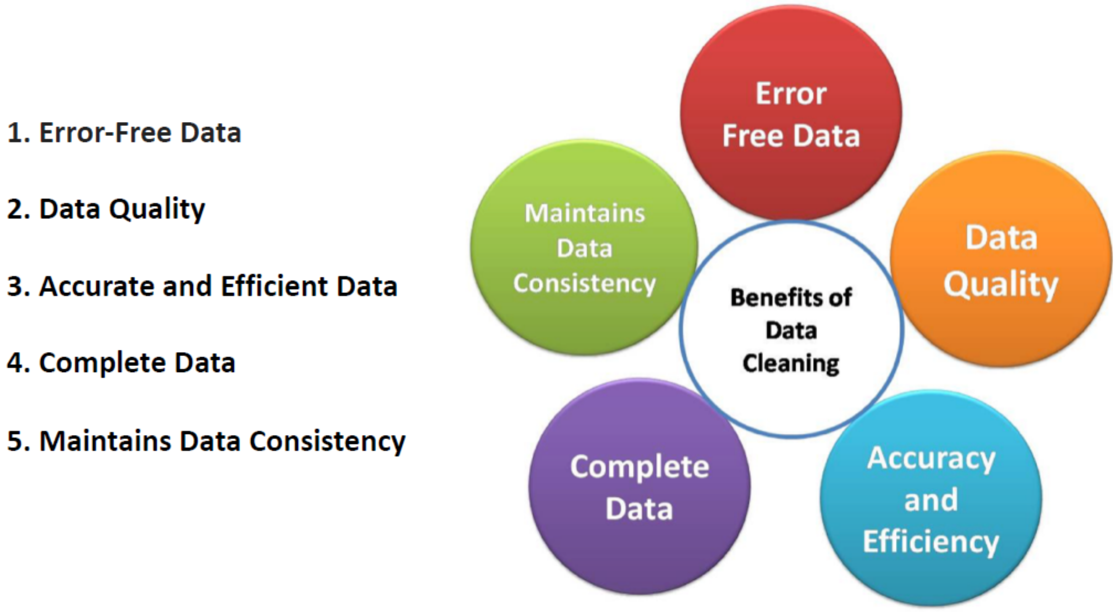
    - Each row describes a single object
    - Each column describes a property of that object
    - Columns are numeric whenever appropriate
    - Columns contain atomic properties that cannot be further decomposed

3. Explore global properties. Use histograms, scatter plots, and aggregation functions to summarize the data.

4. Explore group properties. Use groupby, queries, and small multiples to compare subsets of the data.

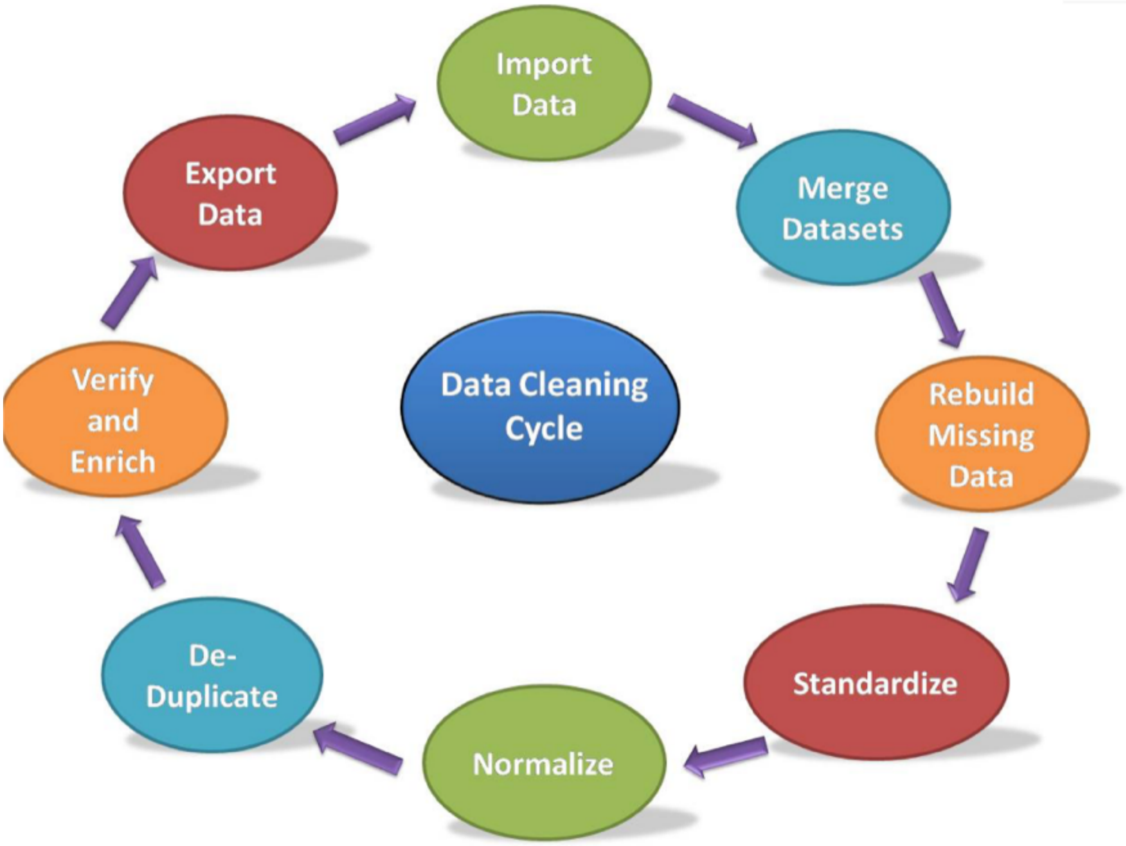
Data cleaning

- why essential?



1. **Error-Free Data**
2. **Data Quality**
3. **Accurate and Efficient Data**
4. **Complete Data**
5. **Maintains Data Consistency**

- data cleaning cycle



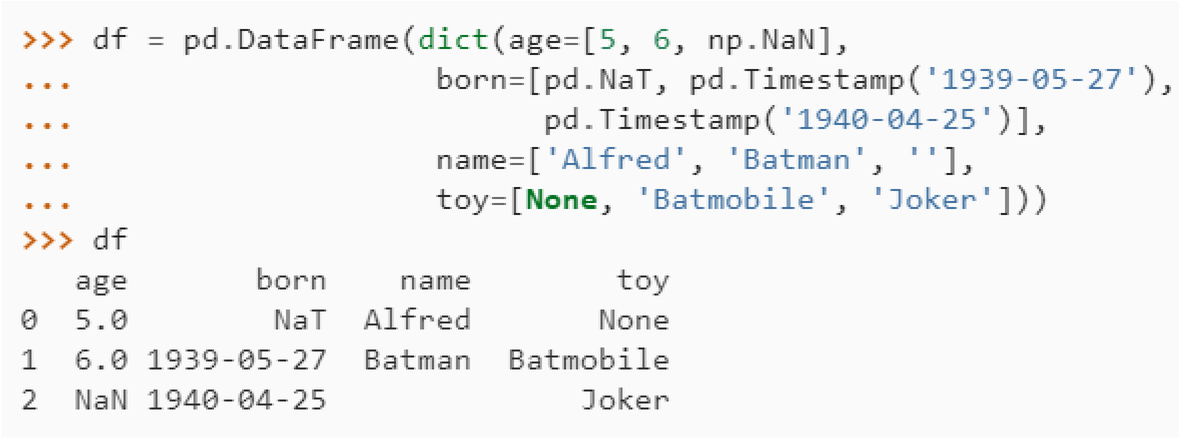
Merge dataset

Using pandas.DataFrame.merge to merge dataset.

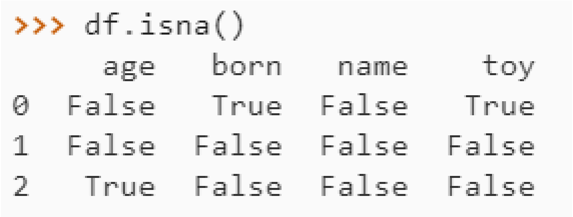
Rebuild missing data

Find the missing data

Using isnull() or isna() function, for example:

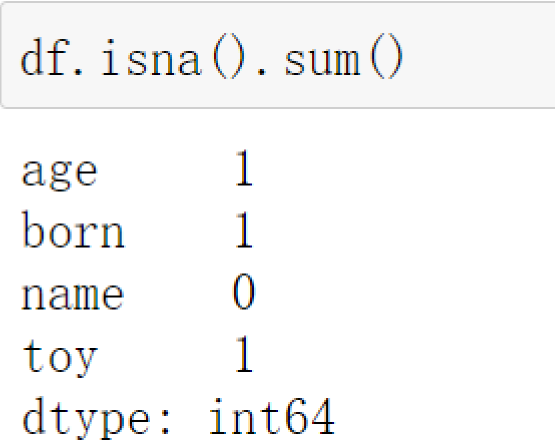


```
>>> df = pd.DataFrame(dict(age=[5, 6, np.NaN],
...                    born=[pd.NaT, pd.Timestamp('1939-05-27'),
...                          pd.Timestamp('1940-04-25')],
...                    name=['Alfred', 'Batman', ''],
...                    toy=[None, 'Batmobile', 'Joker']))
>>> df
   age       born    name        toy
0  5.0        NaT  Alfred       None
1  6.0 1939-05-27  Batman  Batmobile
2  NaN 1940-04-25              Joker
```



```
>>> df.isna()
     age   born   name    toy
0  False   True  False   True
1  False  False  False  False
2   True  False  False  False
```

同时使用 sum()，会更清晰地显示（行标题下有几个 nan）：



```
df.isna().sum()
```

```
age       1
born      1
name      0
toy       1
dtype: int64
```

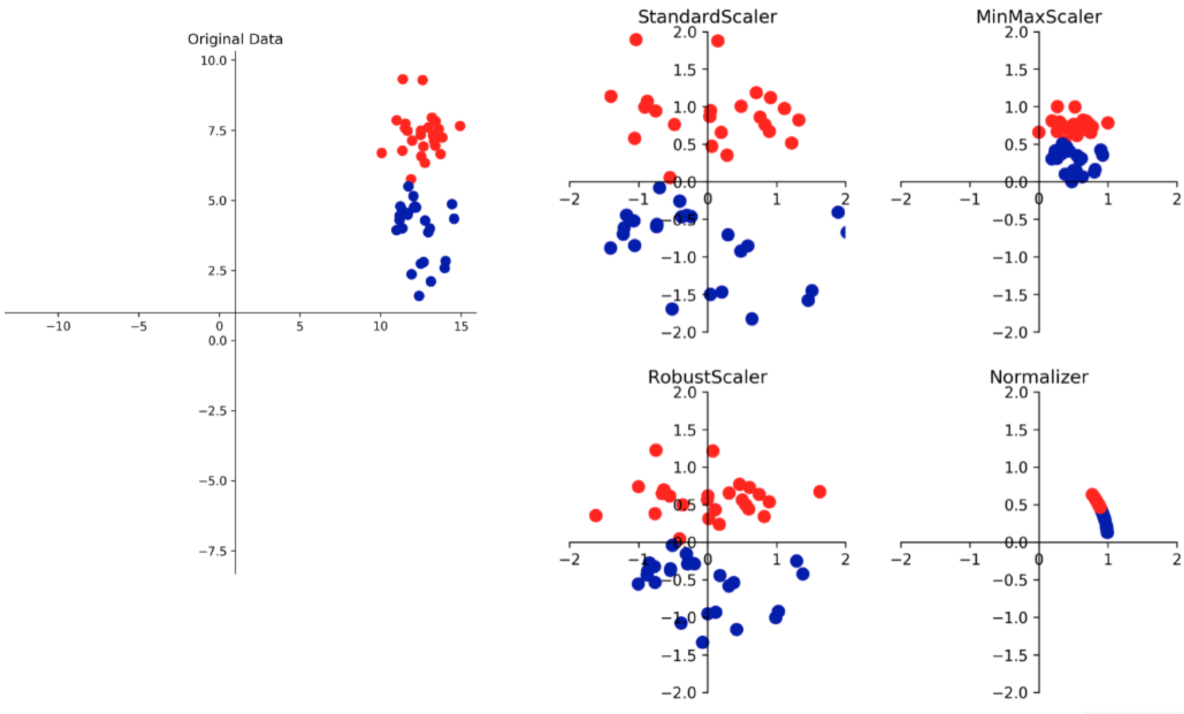
Fill the messing data

使用 fillna() 函数来填充 NA/NaN 数据。

De-duplicate

De-Duplicate 意为删除所有重复的数据，使用 duplicated() 函数来找到重复的数据，使用 drop_duplicates() 函数来删除重复的数据。

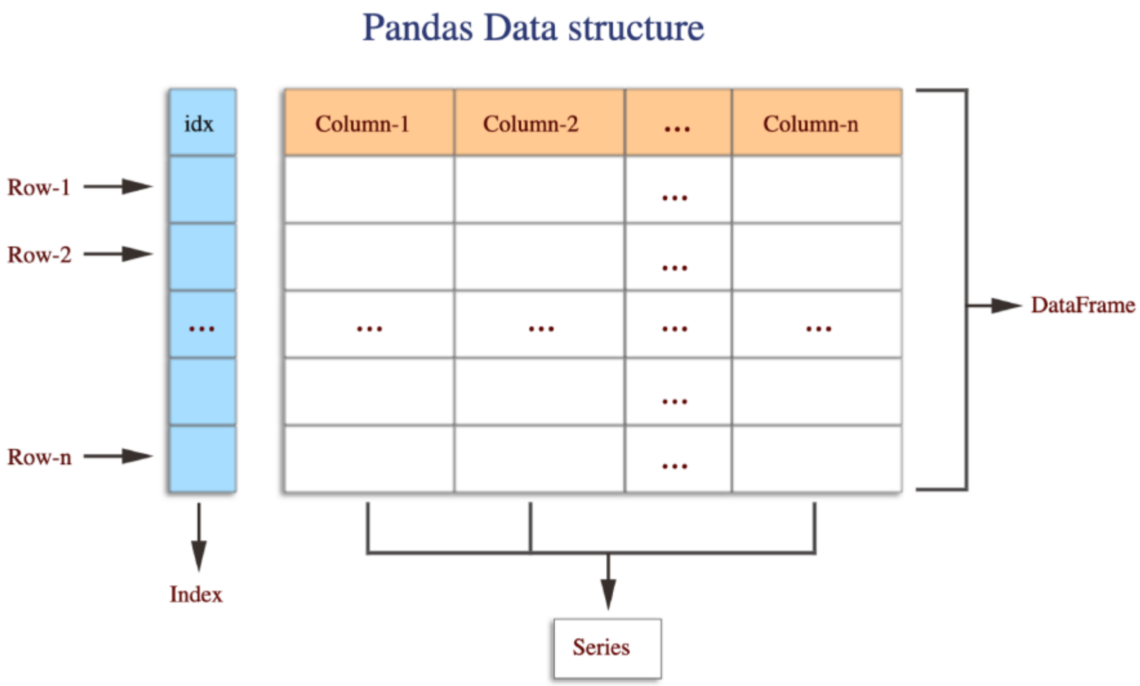
Standardization and Normalization



对数据进行标准化和归一化：

- StandardScaler：通过删除平均值并缩放到单位方差来标准化数据
- MinMaxScaler：通过将每个数据缩放到给定范围来转换数据
- RobustScaler：对离群值进行缩放，来标准化数据
- Normalizer：将 sample 单独归一化为 unit norm，公式为 x_ = (x - min) / (max - min)

Pandas

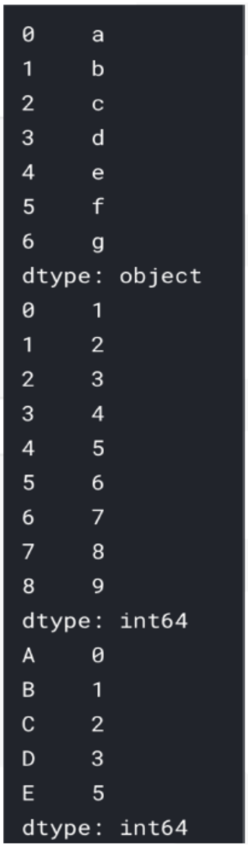


Grammar

HOW TO CREATE A SERIES FROM A LIST, NUMPY ARRAY AND DICT?

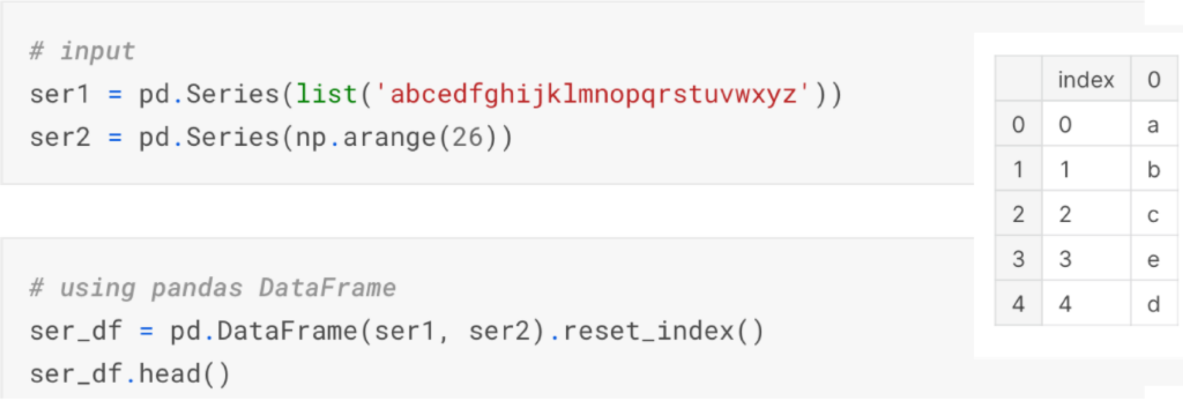
```
# Input
import numpy as np
a_list = list("abcdefg")
numpy_array = np.arange(1, 10)
dictionary = {"A":  0, "B":1, "C":2, "D":3, "E":5}
```

```
series1 = pd.Series(a_list)
print(series1)
series2 = pd.Series(numpy_array)
print(series2)
series3 = pd.Series(dictionary)
print(series3)
```



```
0     a
1     b
2     c
3     d
4     e
5     f
6     g
dtype: object
0     1
1     2
2     3
3     4
4     5
5     6
6     7
7     8
8     9
dtype: int64
A     0
B     1
C     2
D     3
E     5
dtype: int64
```

HOW TO COMBINE MANY SERIES TO FORM A DATAFRAME?

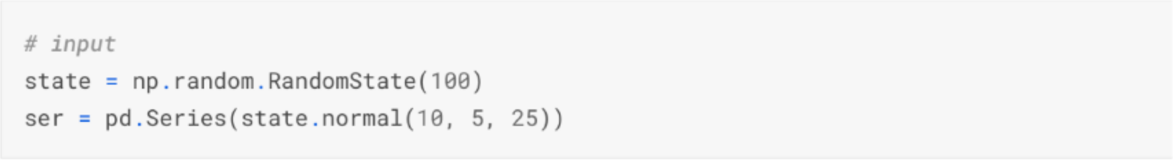


```
# input
ser1 = pd.Series(list('abcedfghijklmnopqrstuvwxyz'))
ser2 = pd.Series(np.arange(26))
```

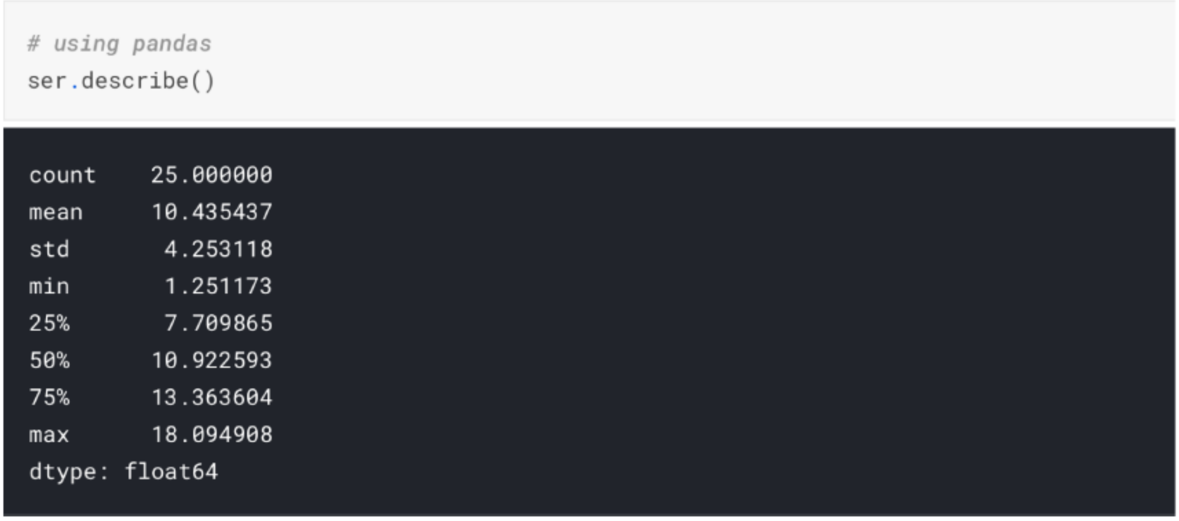
```
# using pandas DataFrame
ser_df = pd.DataFrame(ser1, ser2).reset_index()
ser_df.head()
```

	index	0
0	0	a
1	1	b
2	2	c
3	3	e
4	4	d

HOW TO GET USEFUL INFOS



```
# input
state = np.random.RandomState(100)
ser = pd.Series(state.normal(10, 5, 25))
```



```
# using pandas
ser.describe()
```

```
count    25.000000
mean     10.435437
std       4.253118
min       1.251173
25%       7.709865
50%      10.922593
75%      13.363604
max      18.094908
dtype: float64
```

groupby

根据某些标准将数据拆分为组，使用 groupby() 函数。

merge

将数据进行合并，使用 merge() 函数。