Useful commands for pandas.DataFrame

1. Take a quick look

- df.head(10) : return the first 10 rows of the dataframe
- df.tail(10) : return the last 10 rows of the dataframe
- df.shape : return the dimensions of the dataframe
- df.info() : return a summary of columns, no. of non-null value & data type
- df[ColA] .value_counts : return counts of unique values in ColA
- df.describe : return some statistics for df / df subsets

2. Data Manipulation

- **Drop named columns**
  - df.drop(columns = [[ColA, ColB, …]], inplace = True)
- **Split concatenated columns**
  - df[[cola, colB]] = df[concat_col].str.split(',', expand = True)
- **Sort the order**
  - df[ColA].sort_values(ascending = False)
- **Groupby**
  - df.groupby(ColA)…some more operations here…

Useful commands for matplotlib.pyplot as plt

1. Histogram Graph

```
plt.hist(
        pd.DataFrame[ColA]
)
plt.xlabel( … )
plt.ylabel ( … )
plt.title( … )
```

2. Boxplot Graph

```
plt.box(
        pd.DataFrame[ColA],
)
plt.xlabel( … )
plt.title( … )
```

Useful commands for plotly.express as px

1. Scatter Map Graph

```
fig = px.scatter_map(
        pd.DataFrame,
        lat = colA,
        lon = colB,
        center={"lat": … , "lon": … },
        width=600,
        height=600,
        hover_data=[ColC]
```

```
)
fig.update_layout(mapbox_style="open-street-map")
fig.show()
```

Popular Models

1. **Linear Regression** (sklearn.linear_model.LinearRegression)
   Ordinary least squares Linear Regression
   LinearRegression fits a linear model with coefficients w = (w1, …, wp) to minimize the residual sum of squares
   - coef_
   - rank_
   - intercept_
2. **Auto Regressive** (statsmodels.tsa.ar_model.AutoReg)
   - cooperate with PACF to study the correlation of previous values.
3. **ARMA** (statsmodels.tsa.ar_model.ARIMA)

Preprocessing

1. **SimpleImputer** (sklearn.impute.SimpleImputer)
   Replace missing values with specific value, such as mean, median
2. **OrdinalEncoder** (sklearn.preprocessing.OrdinalEncoder)
   Encode categorical features with integer array (0…n-1), applicable for values with ordering.
3. **OneHotEncoder** (sklearn.preprocessing.OneHotEncoder)
   Encode categorical features with binary format, applicable for values without ordering.

MongoDB (NoSQL)

Running as a non-relational database, it can handle storage for structured, semi-structured & unstructured data.

- Structure: Database → Collection (= table) → Document (= record)

Useful read commands:

1. *List(client.list()_databases()) / List(database.list_collections())*
2. *db.collection.find/findOne(<query>, <projection>, <options>)* **[Ref]**
3. *collection.aggregate([{ … }])* **[Ref]**