SKLearn is an open-source Python library for constructing machine learning and statistical models. Rather than data loading and data manipulation and summarization, it focuses on data modeling.
It is built on numpy, scipy and matplotlib.
Integrates with other python modules like

- Pandas for dataframes
- Numpy for vectorization, high-performance linear algebra and array operations
- Matplotlib for graphs and charts

It offers a range of submodules for

- Supervised learning like linear regression and classification.
    - Linear regression models are used for predicting continuous valued attribute. Ex: house price prediction based on the characteristics of the house
      From sklearn.liner_model import LinearRegression

    - Classification to predict the category to which the object belongs. Ex: classification of shapes in triangle, square, rectangle etc. logistic regression, svm, dt, naive bayes, knn

```
from sklearn.tree import DecisionTreeClassifier
from sklearn.svm import SVC
from sklearn.linear_model import LogisticRegression
from sklearn.neighbors import KNeighborsClassifier
from sklearn.naive_bayes import GaussianNB
```

- Unsupervised learning tasks like
    - Clustering to group the similar objects. Kmeans, hierarchical
      From sklearn.cluster import KMeans, DBSCAN, OPTICS
      From sklearn.mixture import GaussianMixture
    - Dimensionality reduction to reduce the number of attributes for summarization and visualizations.
      From sklearn.decompose import PCA

- Ensemble methods combines predictions from multiple supervised models

```
klearn.ensemble import RandomForestClassifier, BaggingClassifier, AdaBoostClassifier, GradientBoostingClassifier
```

- Model selection : comparing the models to choose a good model, choosing parameters of the model etc
  - Manual tuning
  - Automated tuning
    - Grid search:
    - Randamised search

```
from sklearn.model_selection import GridSearchCV, RandomizedSearchCV
```

- data preprocessing:
  - feature extraction to define the attributes in image & text

```
from sklearn.feature_extraction.text import CountVectorizer, tfidf_vectorizer
```

  - Feature selection to identify useful features to create a supervised learning model
  - Data normalization to bring the features into similar ranges
    - MinMaxScaler
    - StanderdScaler
  - Identifying and handling missing values
    isna(), isnull(), dropna(), fillna(), replace(), info()
  - From sklearn.model_selection import train_test_split
  - As many of the ML models work on numeric data, text and images have to be converted to numeric called data or word embeddings
    From sklearn.preprocessing import LabelEncoder, OneHotEncoding

- Built-in datasets for
  - Regression: Boston housing , California housing, diabetis, wine quality, synthetic dataset -make-regression etc
  - Classification : iris, wine quality, breast cancer wisconsin, digits, make-classification
  - Clustering: make-blobs, make-circles,iris, wine, digits

```
from sklearn.datasets import load_iris
```

```
from sklearn.datasets import make_circles
from sklearn.datasets import fetch_california_housing
```