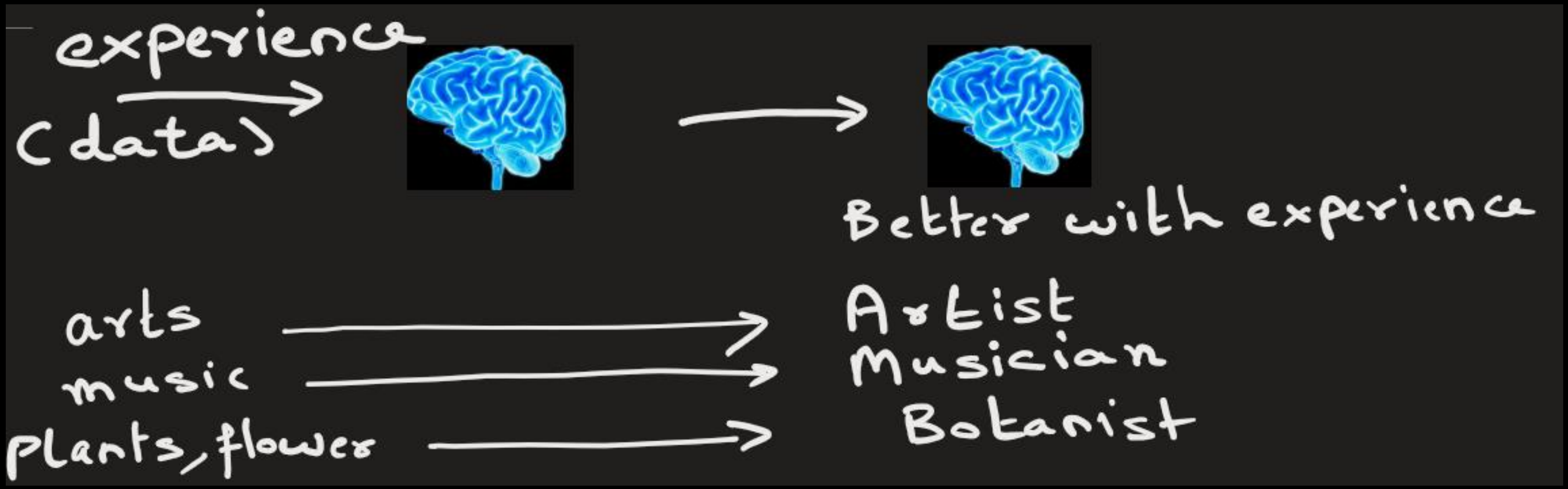
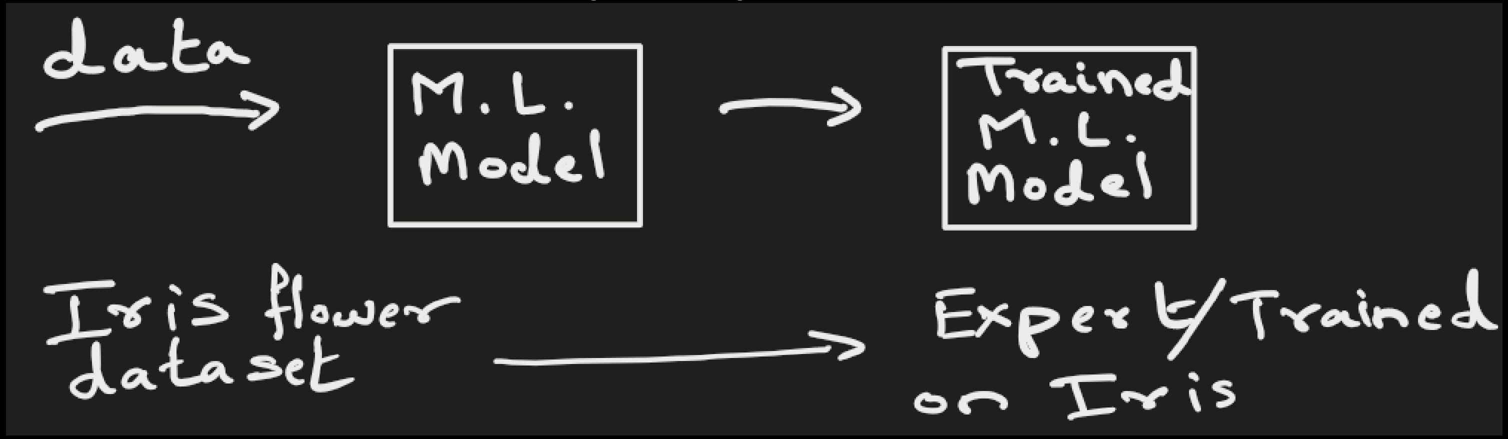
Machine Learning – General Idea

Human brain learns by experience.

This experience comes in the form of data. The brain becomes better with more data.

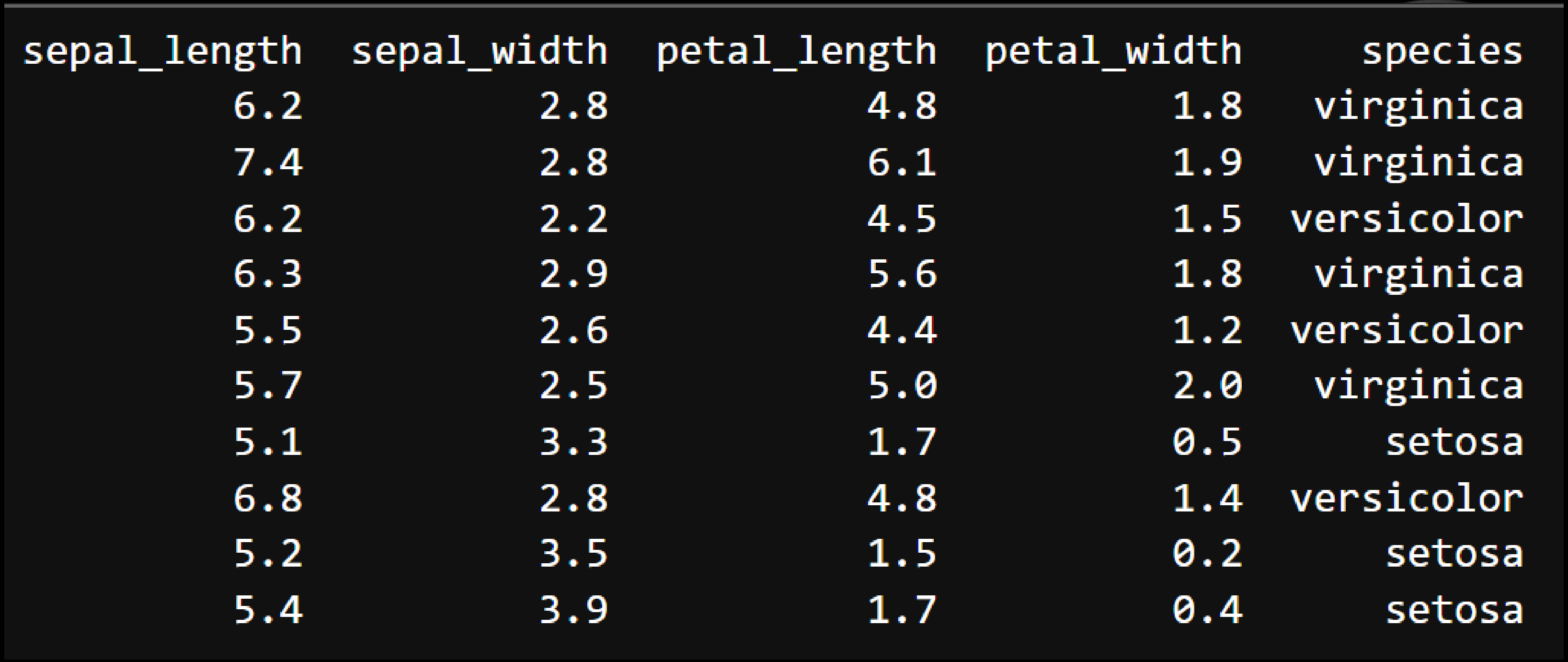


ML Model works on the same principle.

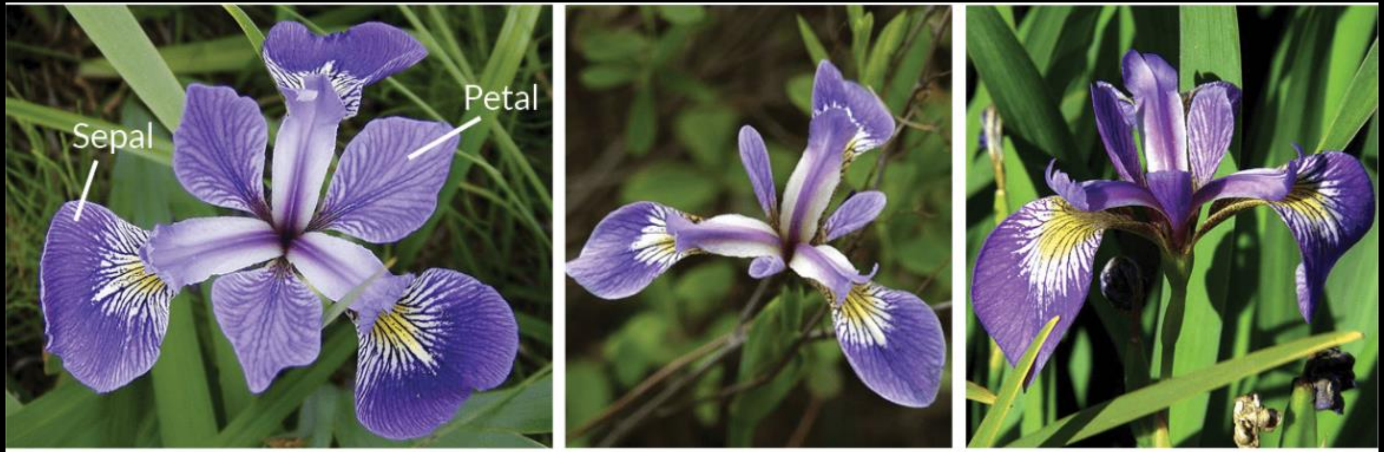


Task: You are given data of Iris flower. Make a ML model that would become expert on Iris flower.

The ML model should be able to recognize the type of flower (Setosa, Versicolor or Virginica) based on input:
sepal and petal dimensions

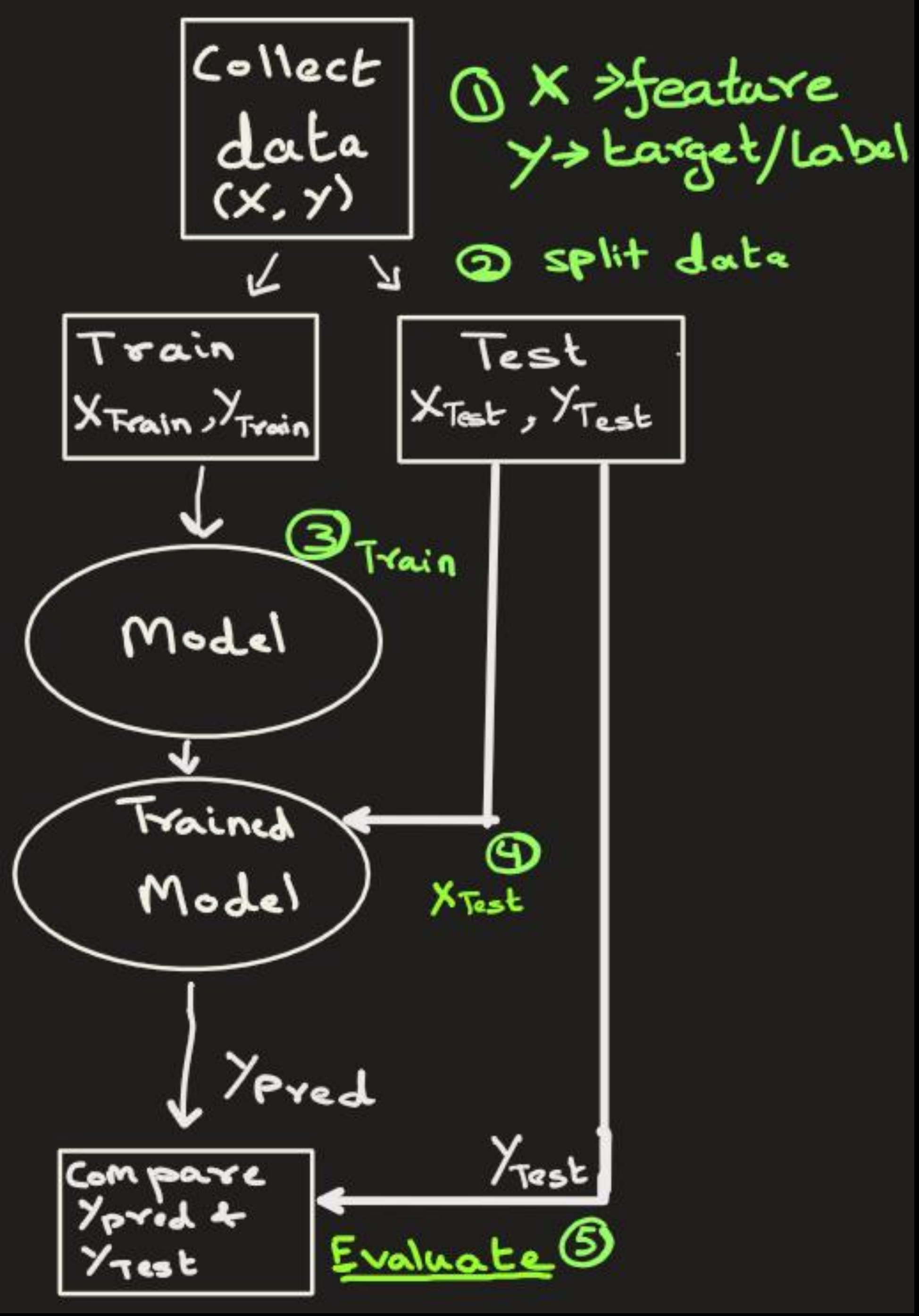


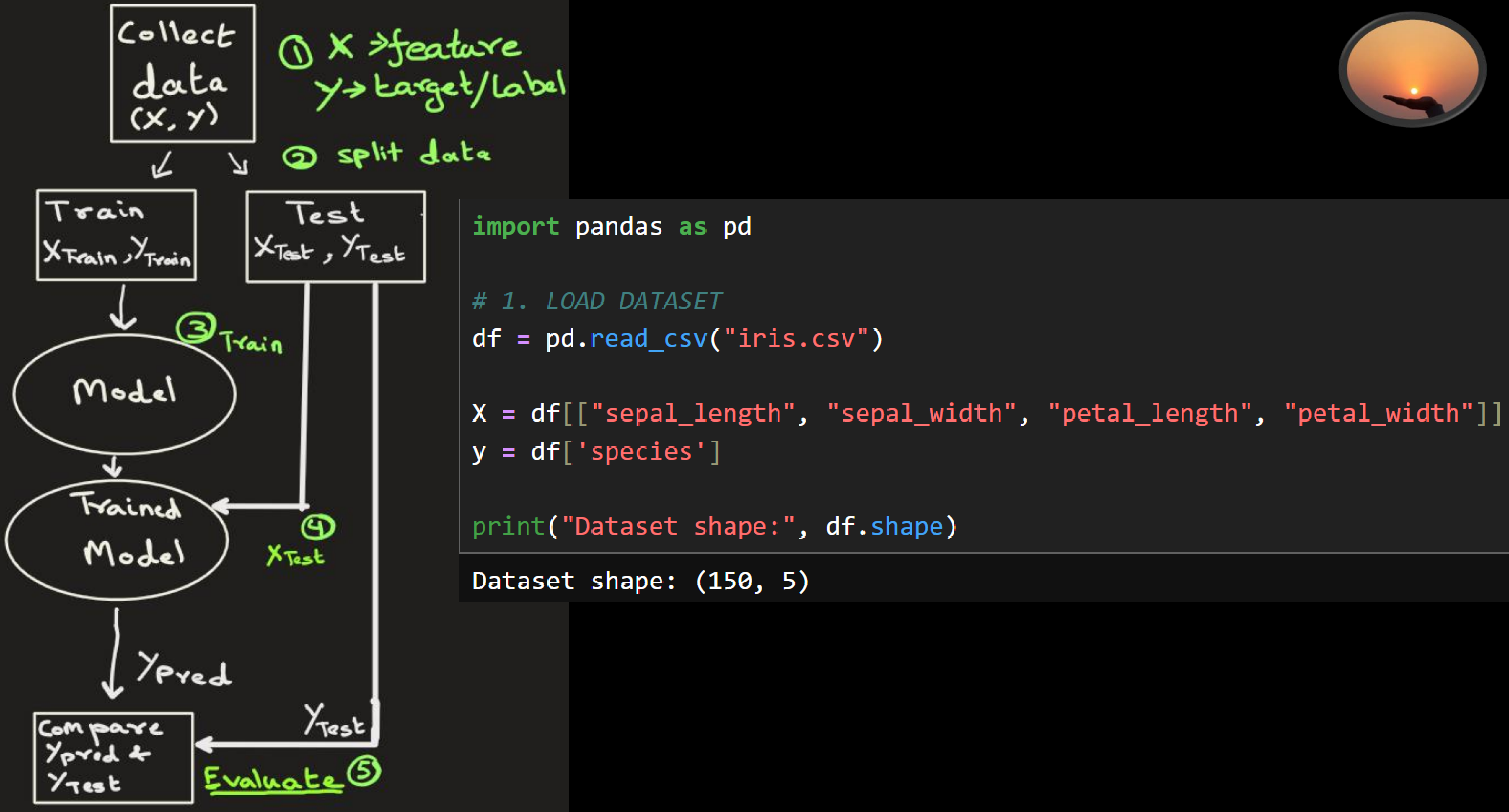
sepal_length	sepal_width	petal_length	petal_width	species
6.2	2.8	4.8	1.8	virginica
7.4	2.8	6.1	1.9	virginica
6.2	2.2	4.5	1.5	versicolor
6.3	2.9	5.6	1.8	virginica
5.5	2.6	4.4	1.2	versicolor
5.7	2.5	5.0	2.0	virginica
5.1	3.3	1.7	0.5	setosa
6.8	2.8	4.8	1.4	versicolor
5.2	3.5	1.5	0.2	setosa
5.4	3.9	1.7	0.4	setosa



Iris Versicolor Iris Setosa Iris Virginica

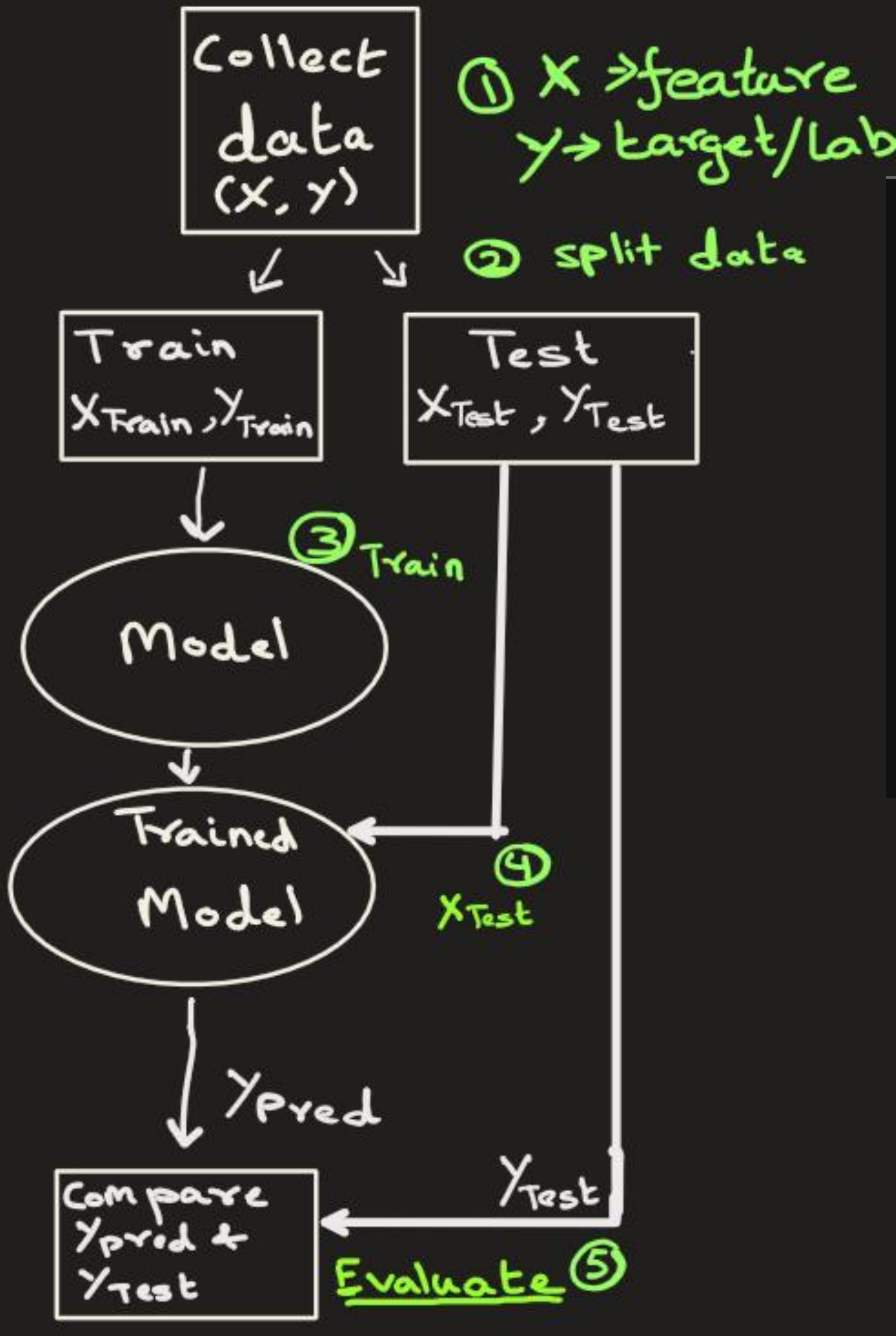
General Idea of ML

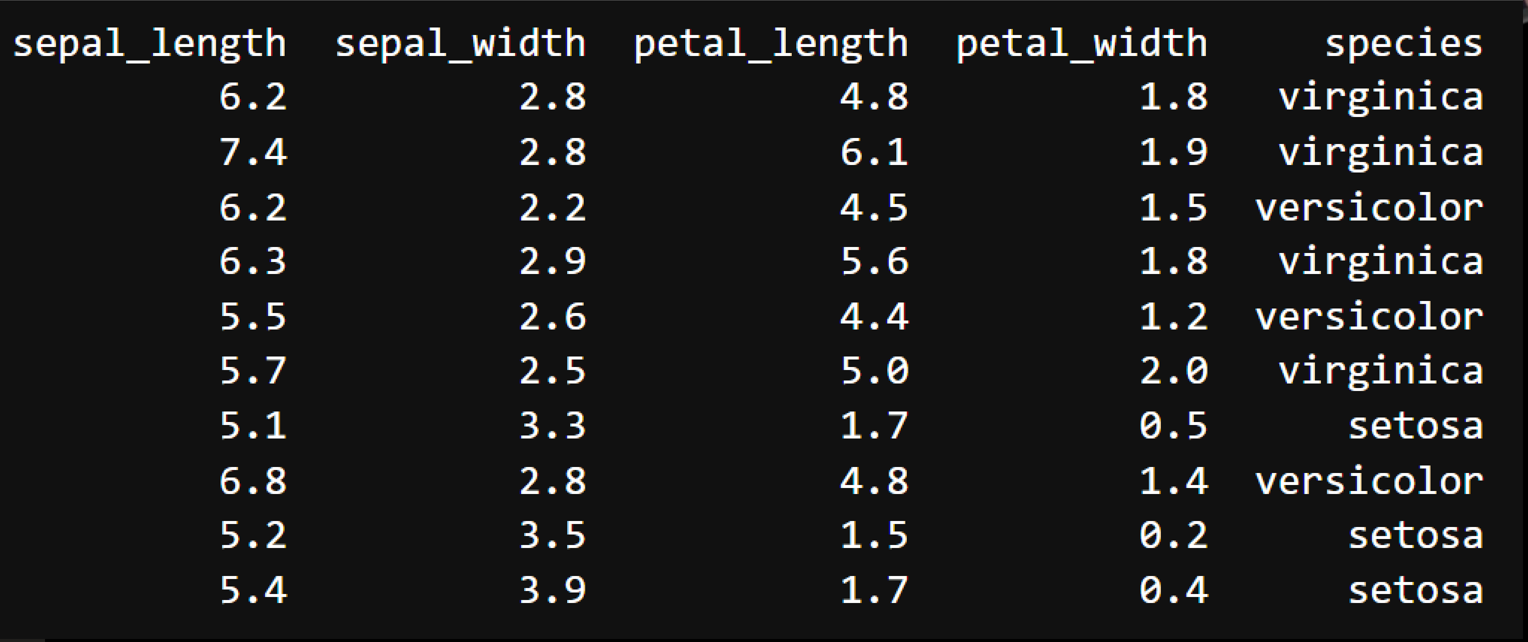




```
import pandas as pd

# 1. LOAD DATASET
df = pd.read_csv("iris.csv")

X = df[["sepal_length", "sepal_width", "petal_length", "petal_width"]]
y = df['species']

print("Dataset shape:", df.shape)
```

```
Dataset shape: (150, 5)
```

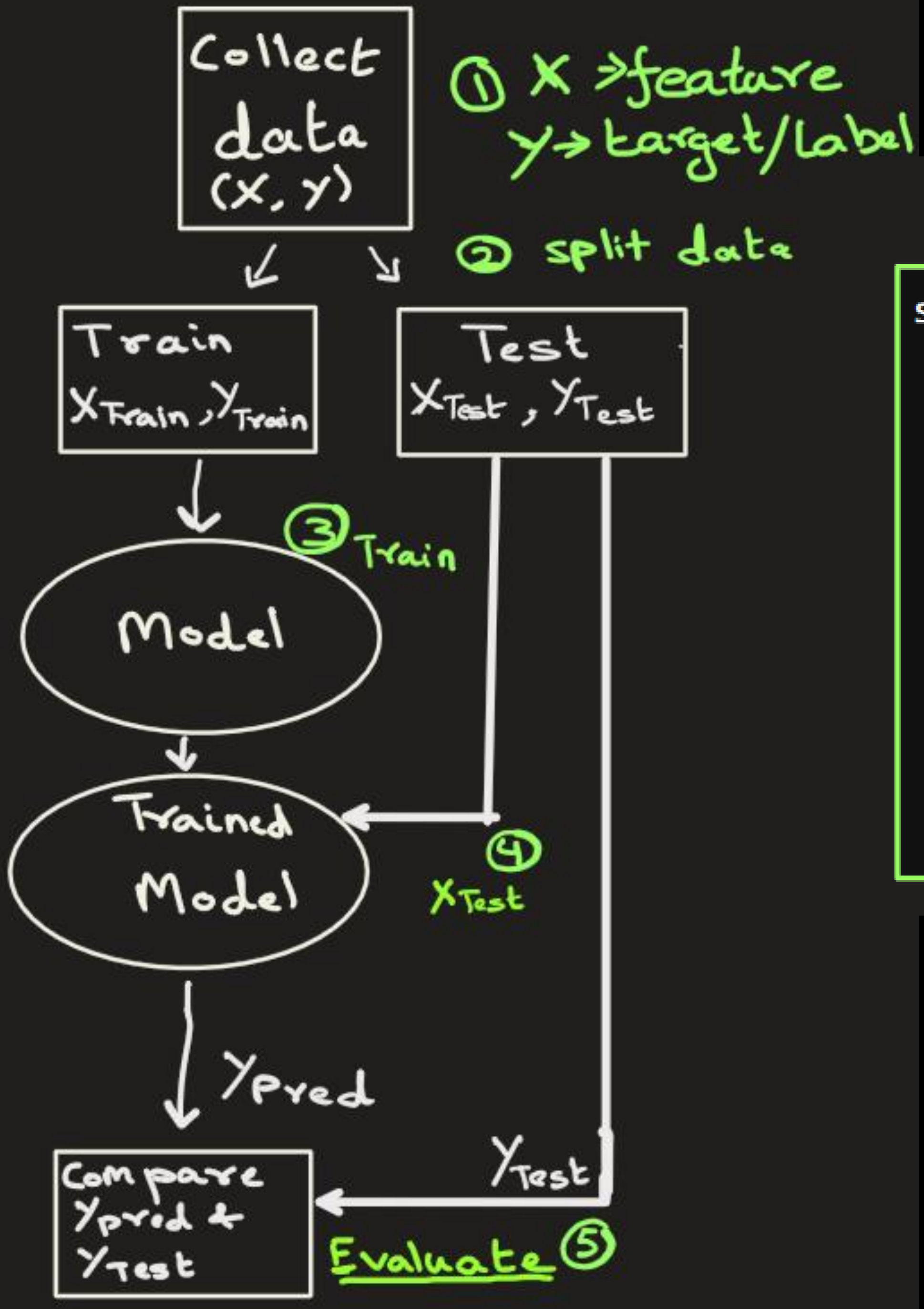


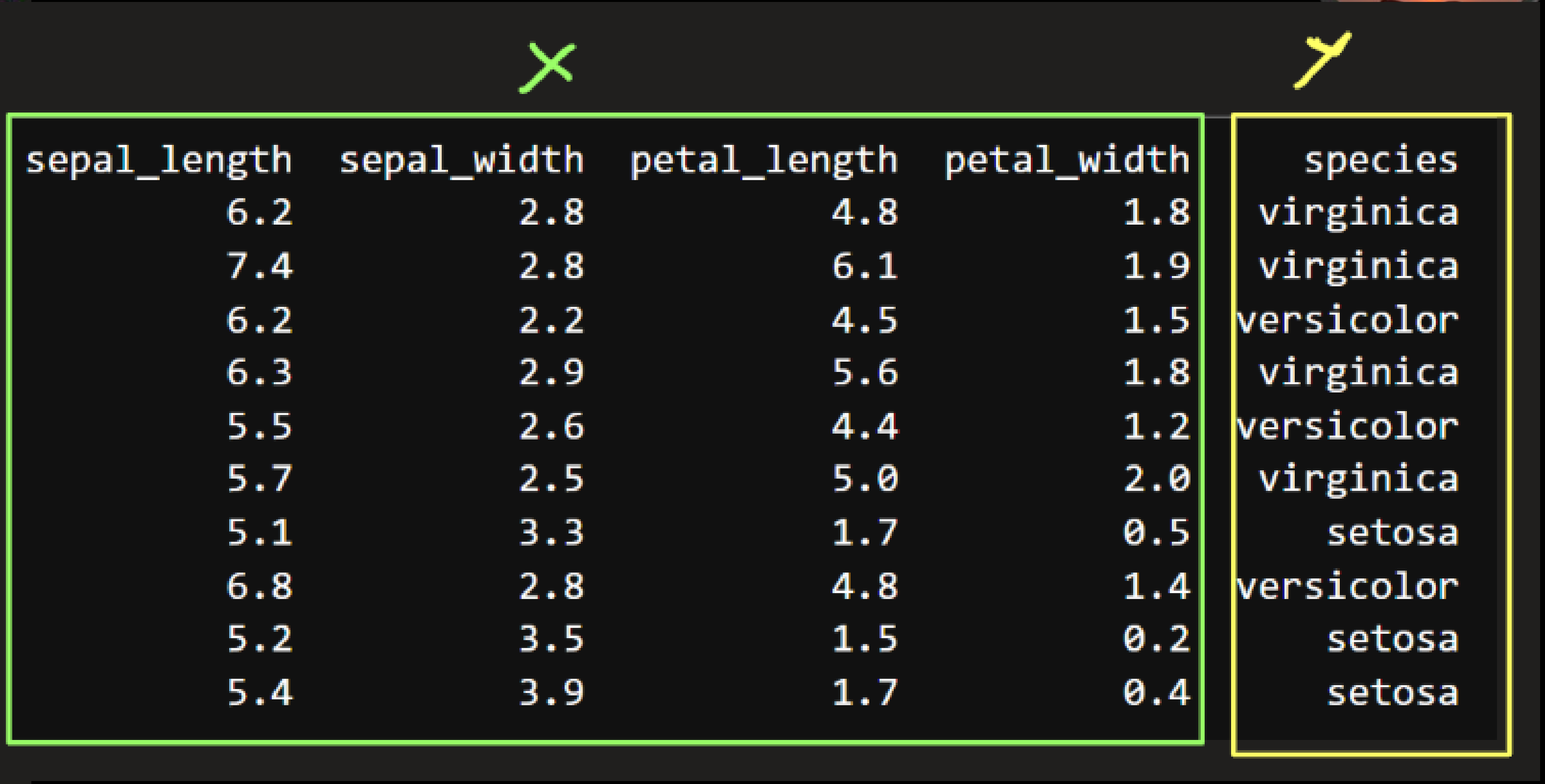


```
sepal_length  sepal_width  petal_length  petal_width     species
         6.2          2.8           4.8          1.8   virginica
         7.4          2.8           6.1          1.9   virginica
         6.2          2.2           4.5          1.5  versicolor
         6.3          2.9           5.6          1.8   virginica
         5.5          2.6           4.4          1.2  versicolor
         5.7          2.5           5.0          2.0   virginica
         5.1          3.3           1.7          0.5      setosa
         6.8          2.8           4.8          1.4  versicolor
         5.2          3.5           1.5          0.2      setosa
         5.4          3.9           1.7          0.4      setosa
```

Following is a sample taken from Iris flower dataset.

- This dataset has total of 150 records
- It contains records for 3 types of flowers:
  Setosa, Versicolor and Virginica
- Each type of flower has 50 records.



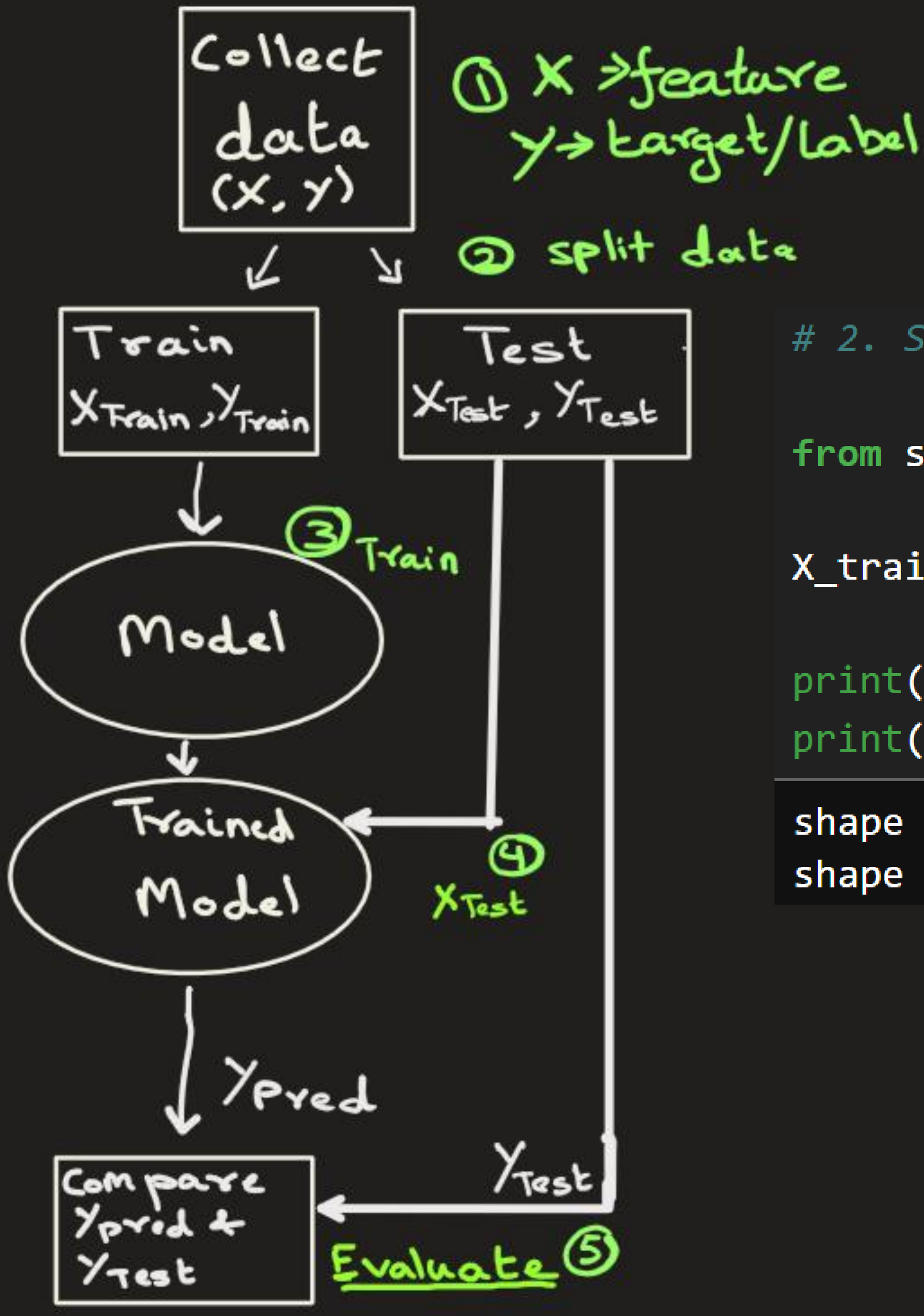


```
sepal_length  sepal_width  petal_length  petal_width     species
         6.2          2.8           4.8          1.8   virginica
         7.4          2.8           6.1          1.9   virginica
         6.2          2.2           4.5          1.5  versicolor
         6.3          2.9           5.6          1.8   virginica
         5.5          2.6           4.4          1.2  versicolor
         5.7          2.5           5.0          2.0   virginica
         5.1          3.3           1.7          0.5      setosa
         6.8          2.8           4.8          1.4  versicolor
         5.2          3.5           1.5          0.2      setosa
         5.4          3.9           1.7          0.4      setosa
```

X is called **features** (independent)

y is called **target/response/label** (dependent)

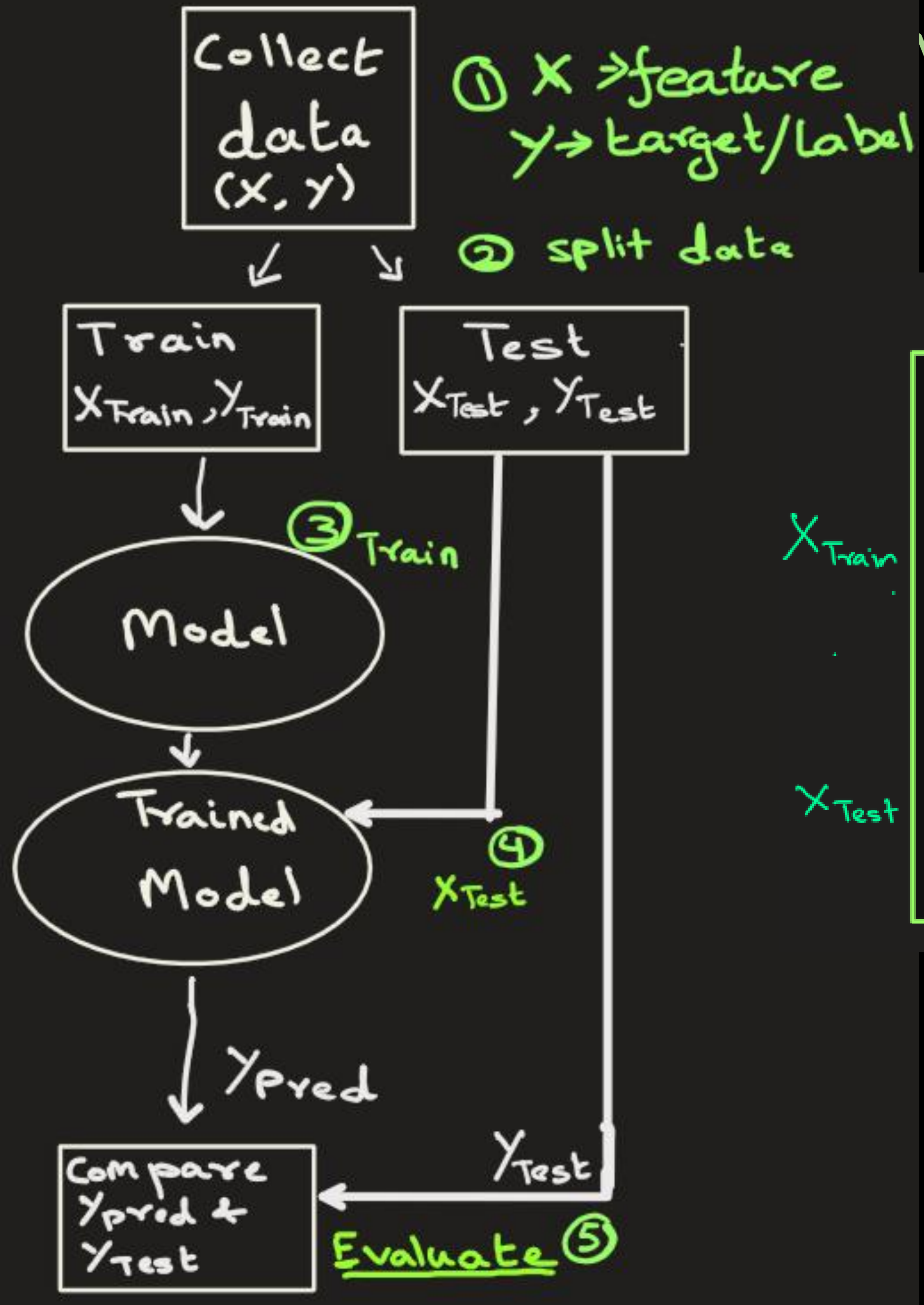
y is function of X.

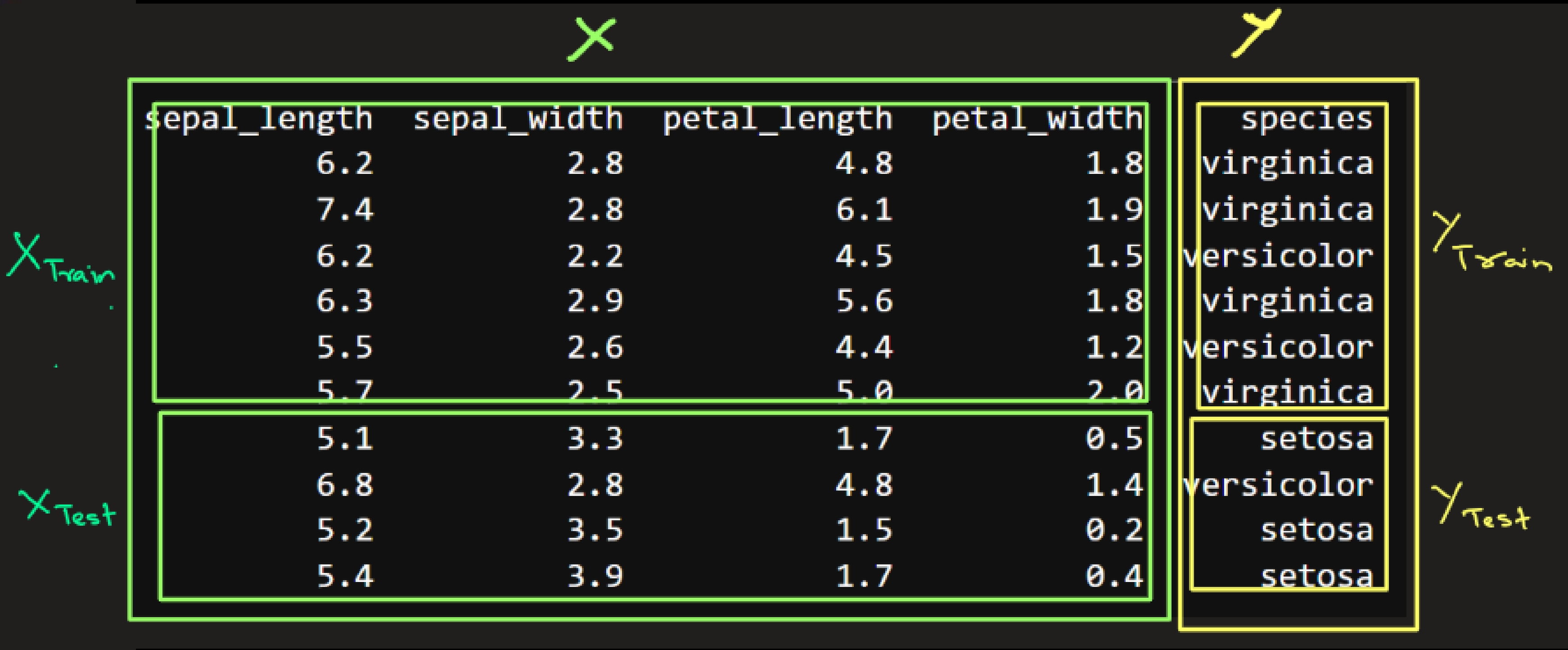


```
# 2. SPLIT DATA INTO TRAINING AND TESTING

from sklearn.model_selection import train_test_split

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2)

print("shape of X_train:", X_train.shape)
print("shape of X_test :", X_test.shape)
```

```
shape of X_train: (120, 4)
shape of X_test : (30, 4)
```

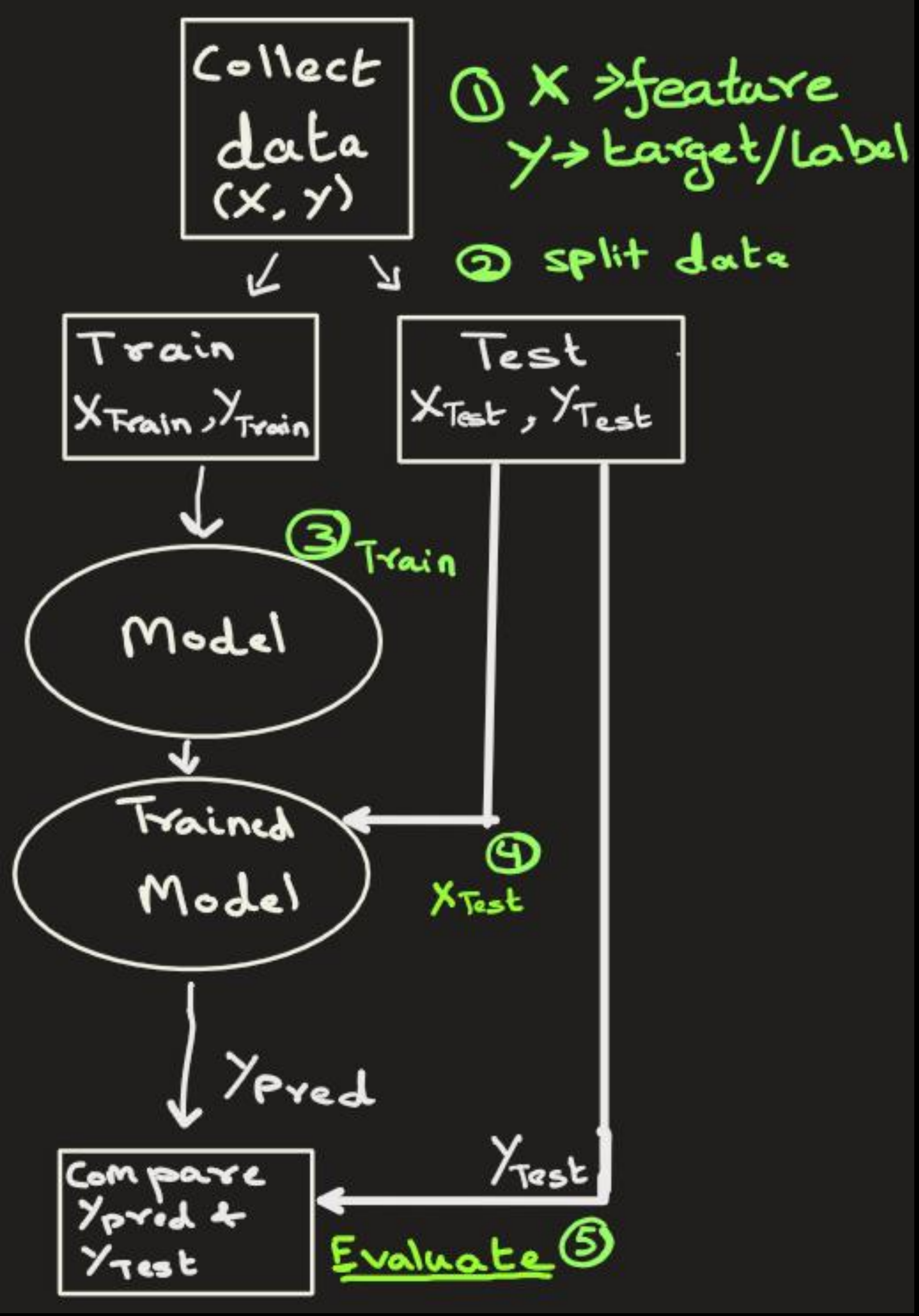
ML Overview

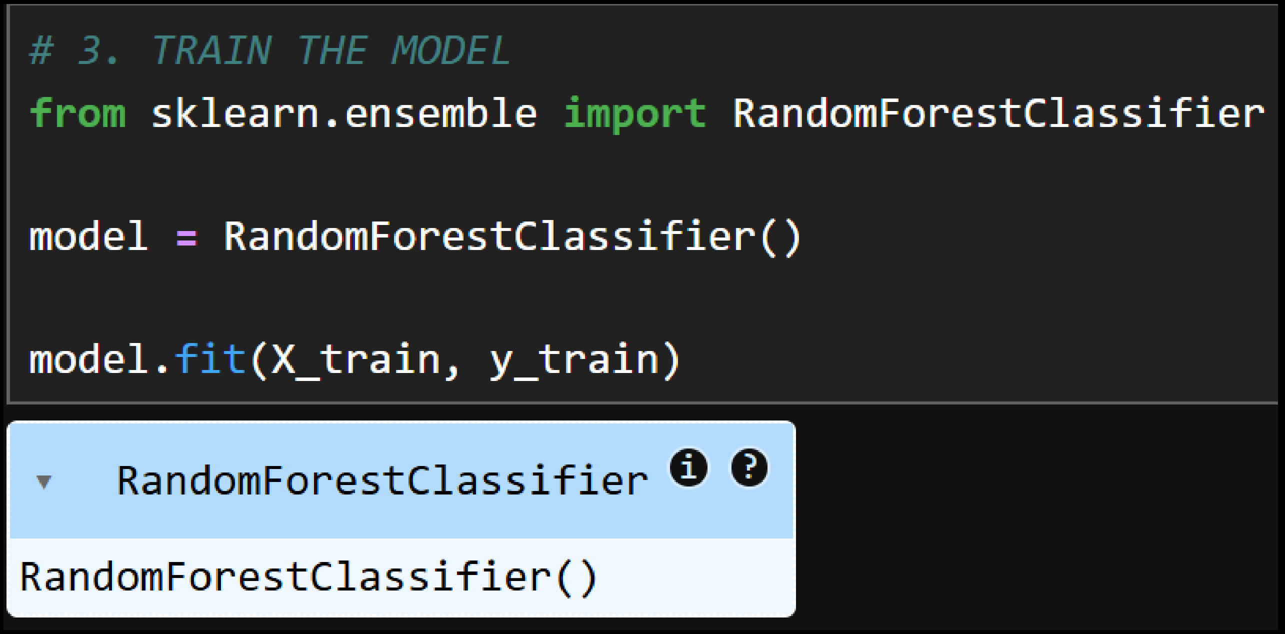




```
sepal_length  sepal_width  petal_length  petal_width     species
         6.2          2.8           4.8          1.8   virginica
         7.4          2.8           6.1          1.9   virginica
         6.2          2.2           4.5          1.5  versicolor
         6.3          2.9           5.6          1.8   virginica
         5.5          2.6           4.4          1.2  versicolor
         5.7          2.5           5.0          2.0   virginica
         5.1          3.3           1.7          0.5      setosa
         6.8          2.8           4.8          1.4  versicolor
         5.2          3.5           1.5          0.2      setosa
         5.4          3.9           1.7          0.4      setosa
```

X and y are split into training and testing part.
There would be total 4 parts now.

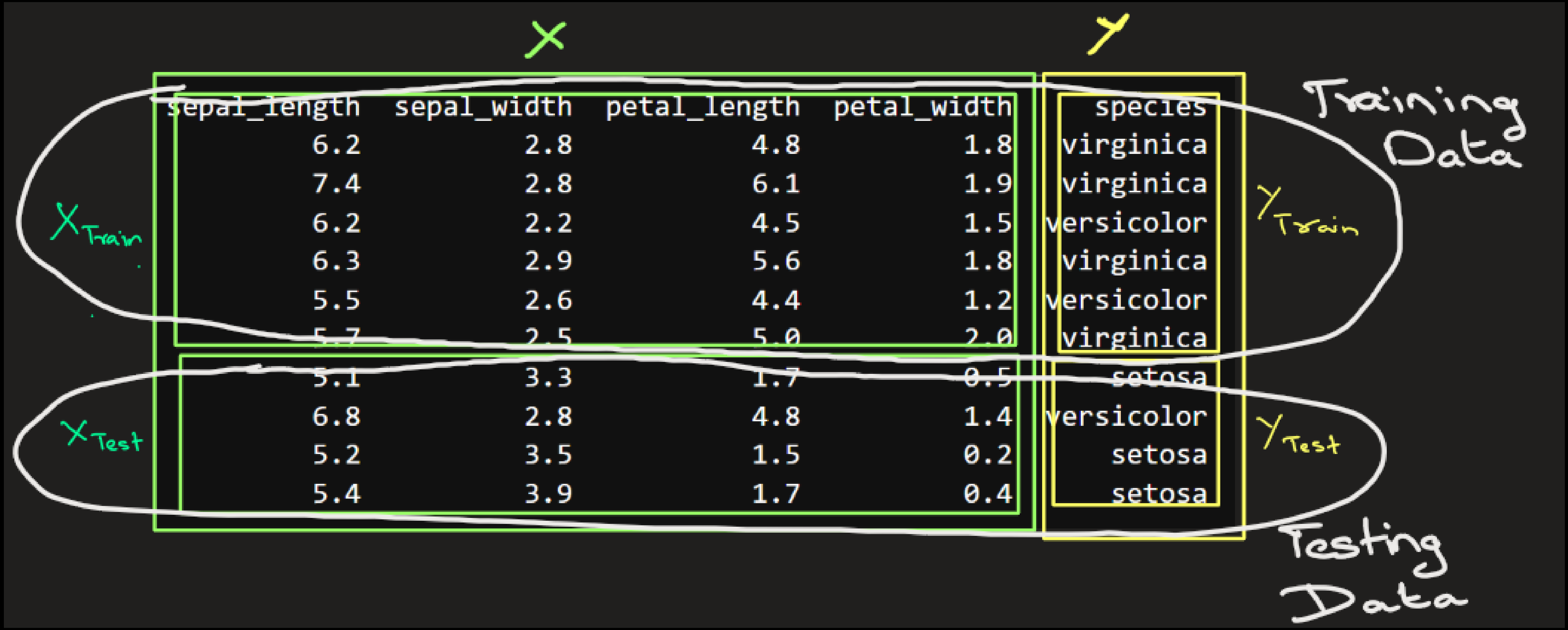




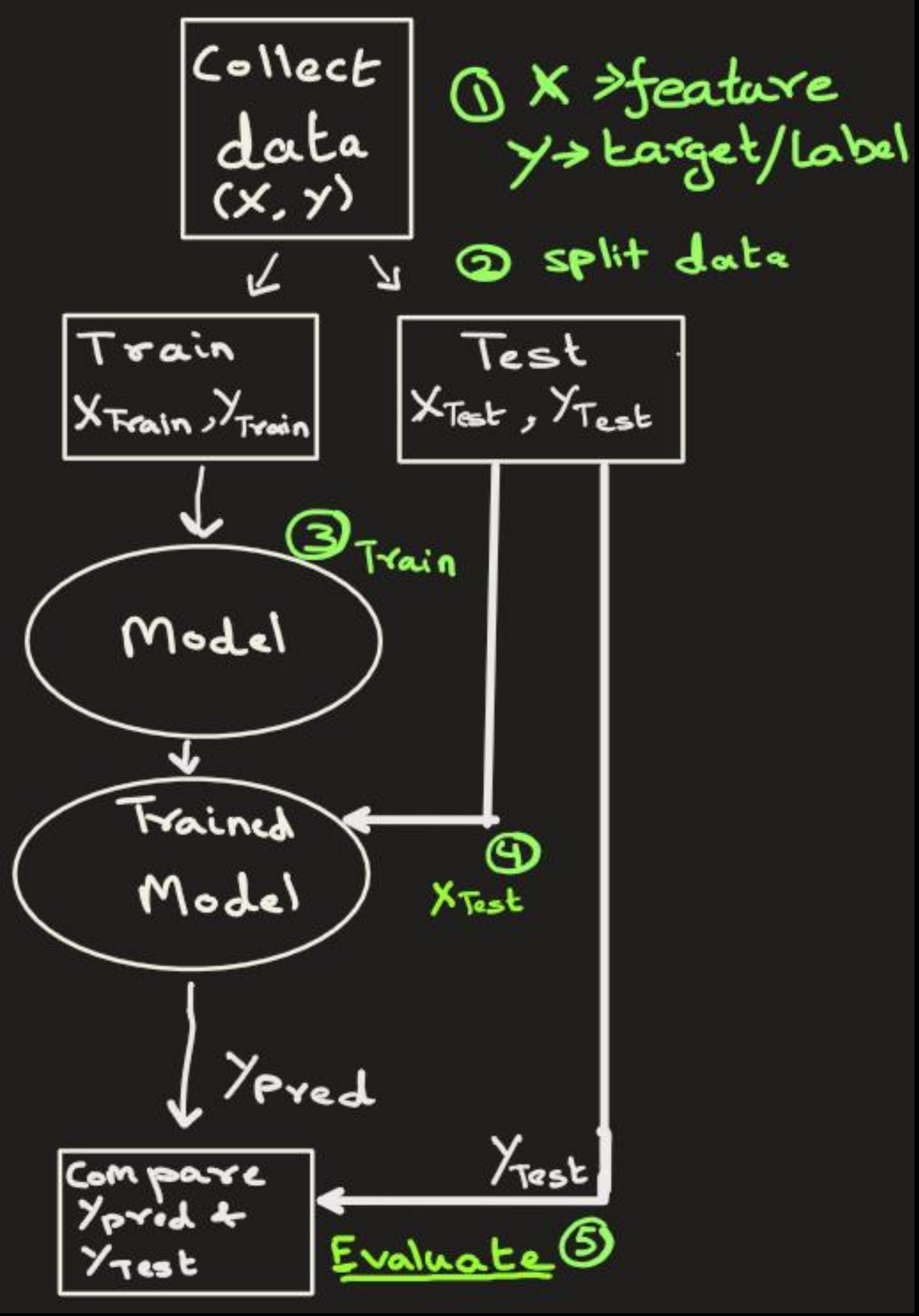
```
# 3. TRAIN THE MODEL
from sklearn.ensemble import RandomForestClassifier

model = RandomForestClassifier()

model.fit(X_train, y_train)
```

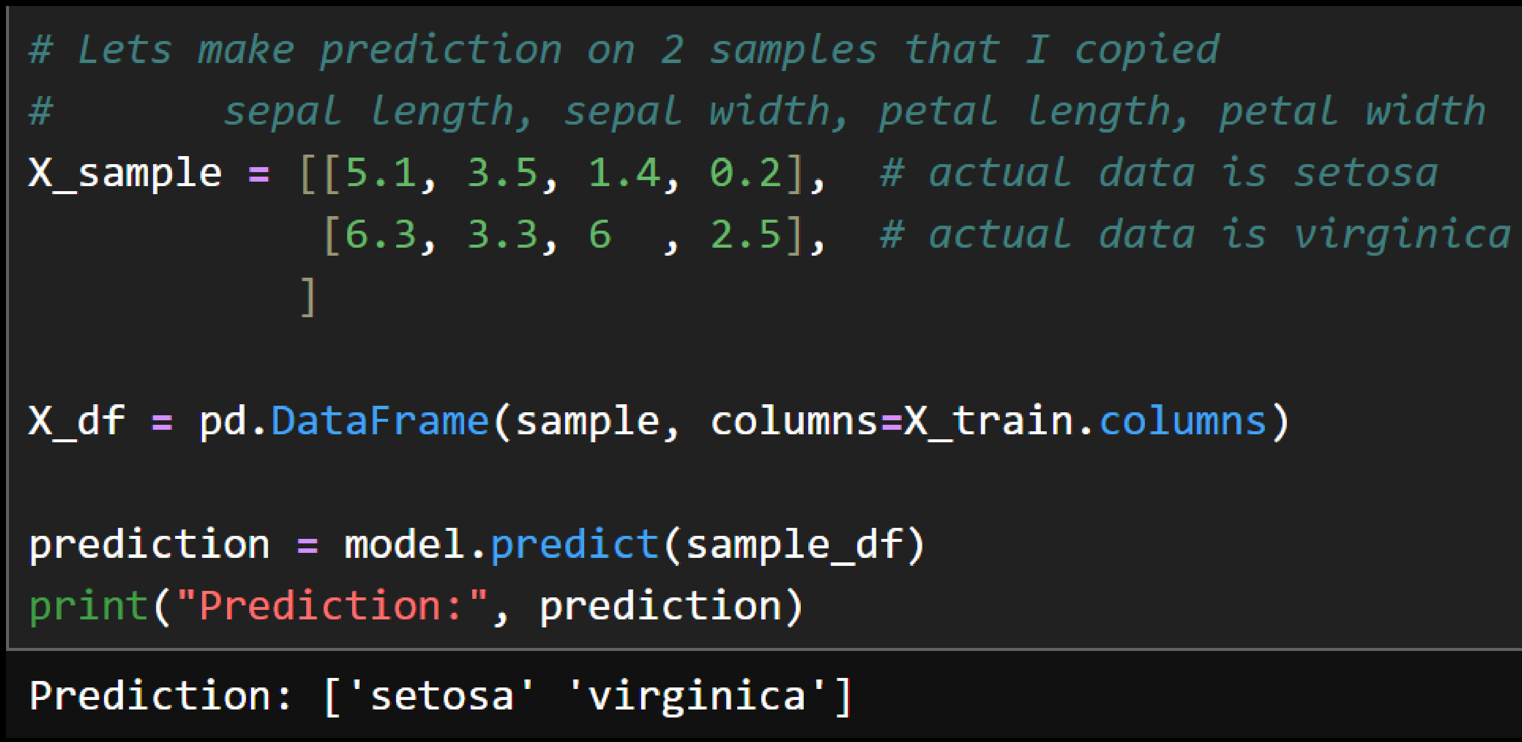
```
▾ RandomForestClassifier
RandomForestClassifier()
```

Training data is fed to the ML Model.
It may take some time for model to learn/train.



```
sepal_length  sepal_width  petal_length  petal_width     species
         6.2          2.8           4.8          1.8   virginica
         7.4          2.8           6.1          1.9   virginica
         6.2          2.2           4.5          1.5  versicolor
         6.3          2.9           5.6          1.8   virginica
         5.5          2.6           4.4          1.2  versicolor
         5.7          2.5           5.0          2.0   virginica
         5.1          3.3           1.7          0.5      setosa
         6.8          2.8           4.8          1.4  versicolor
         5.2          3.5           1.5          0.2      setosa
         5.4          3.9           1.7          0.4      setosa
```

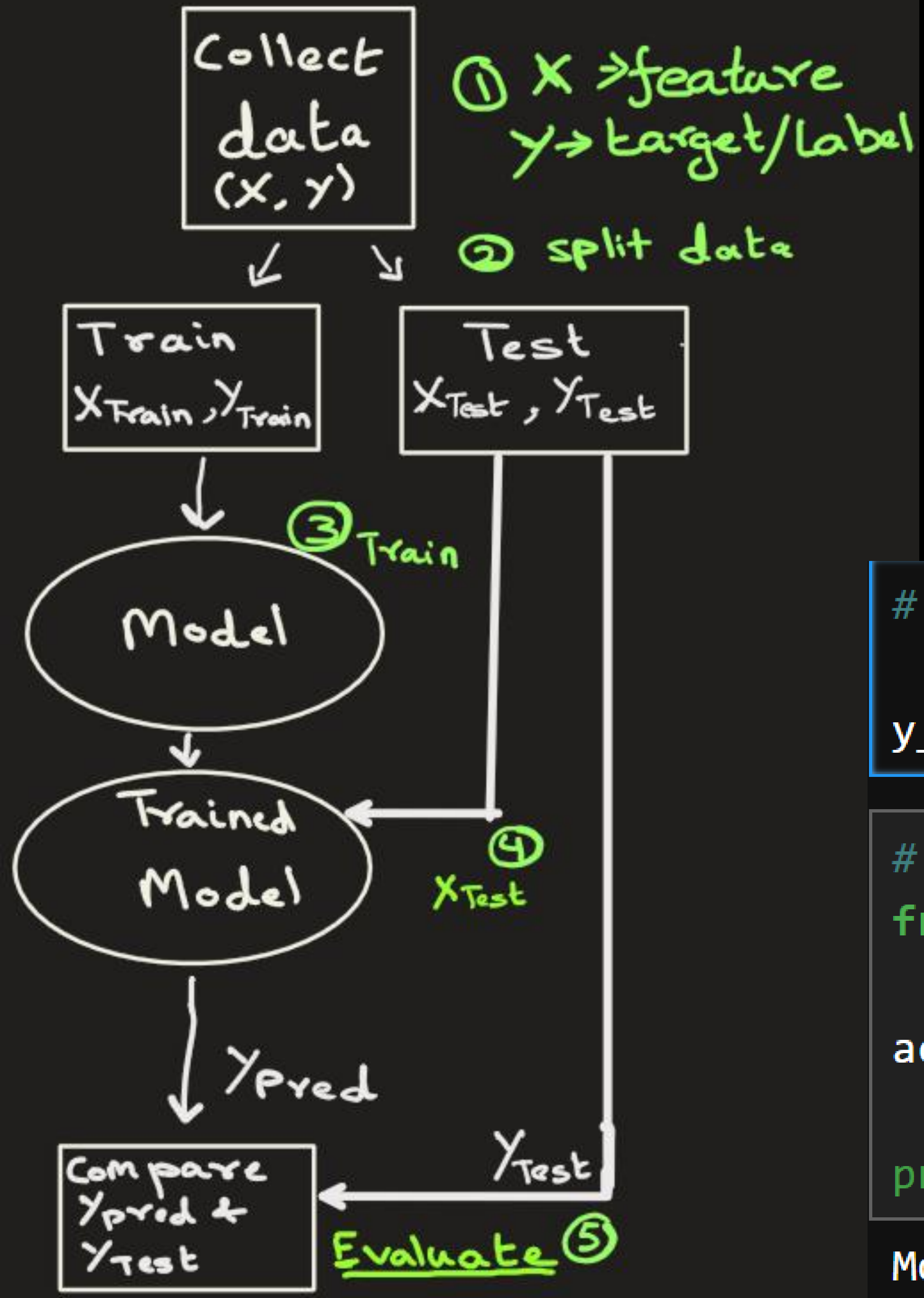


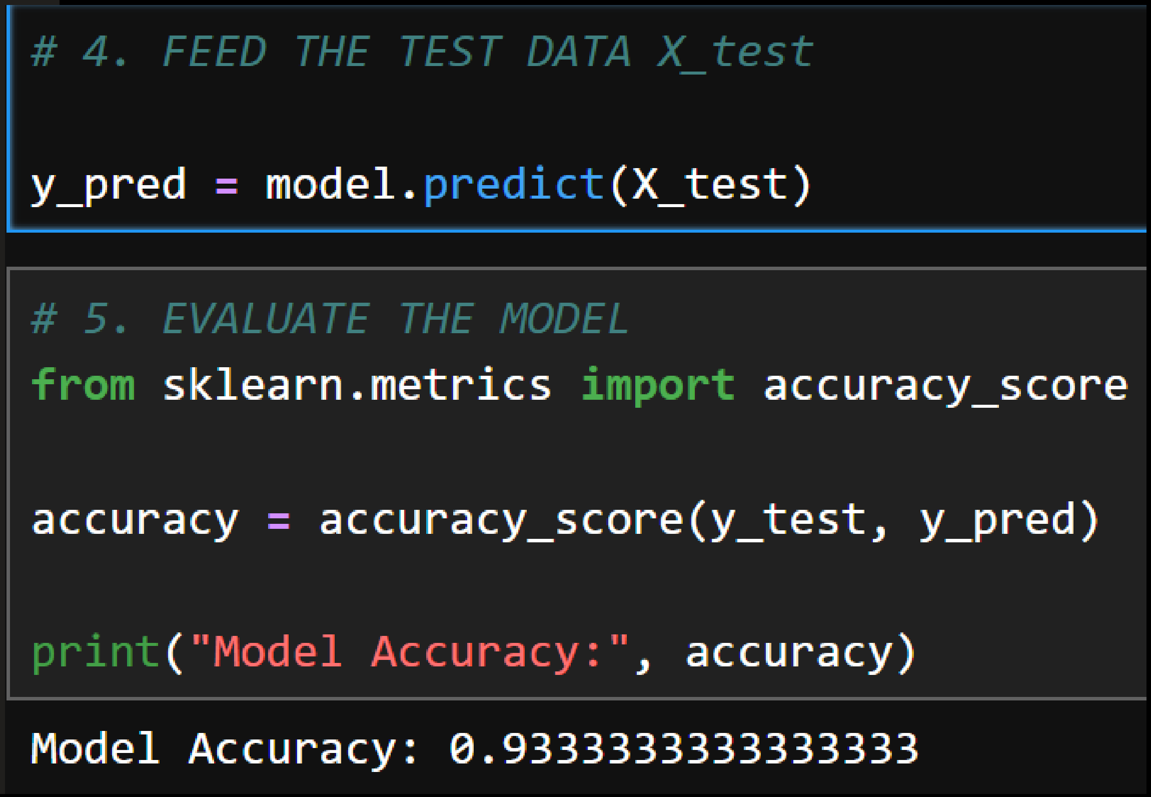


```
# Lets make prediction on 2 samples that I copied
#       sepal length, sepal width, petal length, petal width
X_sample = [[5.1, 3.5, 1.4, 0.2],  # actual data is setosa
             [6.3, 3.3, 6  , 2.5],  # actual data is virginica
            ]

X_df = pd.DataFrame(sample, columns=X_train.columns)

prediction = model.predict(sample_df)
print("Prediction:", prediction)
```

```
Prediction: ['setosa' 'virginica']
```

Iris Versicolor Iris Setosa Iris Virginica





```
# 4. FEED THE TEST DATA X_test

y_pred = model.predict(X_test)
```

```
# 5. EVALUATE THE MODEL
from sklearn.metrics import accuracy_score

accuracy = accuracy_score(y_test, y_pred)

print("Model Accuracy:", accuracy)
```

```
Model Accuracy: 0.9333333333333333
```

ML Overview