

```
In [ ]: import numpy as np
import pandas as pd
from sklearn.model_selection import train_test_split, KFold, cross_val_score
from sklearn.neighbors import KNeighborsClassifier
from sklearn.metrics import accuracy_score
```

Load Data

```
In [ ]: df = pd.read_csv("iris_data.csv")
df
```

```
Out[ ]:
```

	sepal.length	sepal.width	petal.length	petal.width	variety
0	5.1	3.5	1.4	0.2	Setosa
1	4.9	3.0	1.4	0.2	Setosa
2	4.7	3.2	1.3	0.2	Setosa
3	4.6	3.1	1.5	0.2	Setosa
4	5.0	3.6	1.4	0.2	Setosa
...
145	6.7	3.0	5.2	2.3	Virginica
146	6.3	2.5	5.0	1.9	Virginica
147	6.5	3.0	5.2	2.0	Virginica
148	6.2	3.4	5.4	2.3	Virginica
149	5.9	3.0	5.1	1.8	Virginica

150 rows × 5 columns

```
In [ ]: X = df.iloc[:, :4]
Y = df.iloc[:, 4]
```

```
In [ ]: # helper function to display accuracy with passed training and
# testing data

def find_accuracy(X_train, X_test, Y_train, Y_test):
    model = KNeighborsClassifier(n_neighbors=4)
    model.fit(X_train, Y_train)
    preds = model.predict(X_test)
    accuracy = accuracy_score(preds, Y_test)
    print(f"Accuracy : {accuracy}\n")
```

a) 75% train and 25% test

```
In [ ]: X1_train,X1_test,Y1_train, Y1_test = train_test_split(X, Y,
                                                    random_state=104,
                                                    test_size=0.25,
                                                    shuffle=True)
find_accuracy(X1_train,X1_test,Y1_train, Y1_test)
```

Accuracy : 1.0

b) 66.6% train and 33.3% test

```
In [ ]: X2_train,X2_test,Y2_train, Y2_test = train_test_split(X, Y,
                                                    random_state=104,
                                                    test_size=0.333,
                                                    shuffle=True)
find_accuracy(X2_train,X2_test,Y2_train, Y2_test)
```

Accuracy : 0.98

c) Random Subsampling

```
In [ ]: train_len = int(150 * .75)

times = 6

for i in range(times):
    idx = np.random.choice(df.index, train_len , replace= False)
    rem = pd.Index(set(df.index).difference(set(idx)))

    X_train = df.iloc[idx , :4]
    Y_train = df.iloc[idx , 4]

    X_test = df.iloc[idx , :4]
    Y_test = df.iloc[idx , 4]

    print(f"Subsampling {i + 1}")
    find_accuracy(X_train, X_test , Y_train , Y_test)
```

Subsampling 1
Accuracy : 0.9732142857142857

Subsampling 2
Accuracy : 0.9642857142857143

Subsampling 3
Accuracy : 0.9910714285714286

Subsampling 4
Accuracy : 0.9553571428571429

Subsampling 5
Accuracy : 0.9910714285714286

Subsampling 6
Accuracy : 0.9553571428571429

d) Cross Validation

```
In [ ]: k_folds = KFold(n_splits = 5)

scores = cross_val_score(KNeighborsClassifier(n_neighbors=4), X, Y, cv = k_folds)

print("Cross Validation Scores: ", scores)
print("Average CV Score: ", scores.mean())
print("Number of CV Scores used in Average: ", len(scores))
```

```
Cross Validation Scores: [1.          1.          0.83333333 0.93333333 0.76666667]
Average CV Score: 0.9066666666666666
Number of CV Scores used in Average: 5
```