

Diabetes Predictor using K Nearest Neighbours

Consider below dataset

Pregnancies	Glucose	BloodPressure	SkinThickness	Insulin	BMI	DiabetesPedigreeFunction	Age	Outcome
6	148	72	35	0	33.6	0.627	50	1
1	85	66	29	0	26.6	0.351	31	0
8	183	64	0	0	23.3	0.672	32	1
1	89	66	23	94	28.1	0.167	21	0
0	137	40	35	168	43.1	2.288	33	1
5	116	74	0	0	25.6	0.201	30	0
3	78	50	32	88	31	0.248	26	1
10	115	0	0	0	35.3	0.134	29	0
2	197	70	45	543	30.5	0.158	53	1
8	125	96	0	0	0	0.232	54	1

Diabetes predictor application using K nearest neighbour algorithm

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from sklearn.neighbors import KNeighborsClassifier
from sklearn.model_selection import train_test_split

print("---- Marvellous Infosystems by Piyush Khairnar----")

print("---- Diabetes predictor using K Nearest neighbour ----")

diabetes = pd.read_csv('diabetes.csv')

print("Columns of Dataset")
print(diabetes.columns)

print("First 5 records of dataset")
print(diabetes.head())

print("Dimension of diabetes data: {}".format(diabetes.shape))

X_train, X_test, y_train, y_test = train_test_split(diabetes.loc[:, diabetes.columns
!= 'Outcome'], diabetes['Outcome'], stratify=diabetes['Outcome'],
random_state=66)
```

```

training_accuracy = []
test_accuracy = []

# try n_neighbors from 1 to 10
neighbors_settings = range(1, 11)

for n_neighbors in neighbors_settings:
    # build the model
    knn = KNeighborsClassifier(n_neighbors=n_neighbors)
    knn.fit(X_train, y_train)
    # record training set accuracy
    training_accuracy.append(knn.score(X_train, y_train))
    # record test set accuracy
    test_accuracy.append(knn.score(X_test, y_test))

plt.plot(neighbors_settings, training_accuracy, label="training accuracy")
plt.plot(neighbors_settings, test_accuracy, label="test accuracy")
plt.ylabel("Accuracy")
plt.xlabel("n_neighbors")
plt.legend()
plt.savefig('knn_compare_model')
plt.show()

knn = KNeighborsClassifier(n_neighbors=9)

knn.fit(X_train, y_train)

print('Accuracy of K-NN classifier on training set: {:.2f}'.format(knn.score(X_train, y_train)))

print('Accuracy of K-NN classifier on test set: {:.2f}'.format(knn.score(X_test, y_test)))

```

Output of above application

---- Marvellous Infosystems by Piyush Khairnar-----

---- Diabetes predictor using K Nearest neighbour -----

Columns of Dataset

```

Index(['Pregnancies', 'Glucose', 'BloodPressure', 'SkinThickness', 'Insulin',
       'BMI', 'DiabetesPedigreeFunction', 'Age', 'Outcome'],
      dtype='object')

```

First 5 records of dataset

	Pregnancies	Glucose	BloodPressure	...	DiabetesPedigreeFunction	Age	Outcome
0	6	148	72 ...		0.627	50	1
1	1	85	66 ...		0.351	31	0
2	8	183	64 ...		0.672	32	1
3	1	89	66 ...		0.167	21	0
4	0	137	40 ...		2.288	33	1

[5 rows x 9 columns]

Dimension of diabetes data: (768, 9)

Accuracy of K-NN classifier on training set: 0.79

Accuracy of K-NN classifier on test set: 0.78

