

در ابتدا کتابخانه های لازم را import می کنیم :

```
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.ensemble import RandomForestClassifier
from sklearn import svm
from sklearn.preprocessing import StandardScaler
from sklearn.model_selection import train_test_split
from sklearn.naive_bayes import GaussianNB
from sklearn.linear_model import LogisticRegression
from sklearn.neighbors import KNeighborsClassifier
```

ابتدا داده ها را که فرمت csv است را از ورودی خواند و پس از انجام هر الگوریتم امتیاز آن را نمایش می دهیم :

برای مثال :

Gaussian Naive Bayes : Training score - 0.7947625969984234 - Test score - 0.7958356366316566

LogisticRegression : Training score - 0.796891955529166 - Test score - 0.7996437565260119

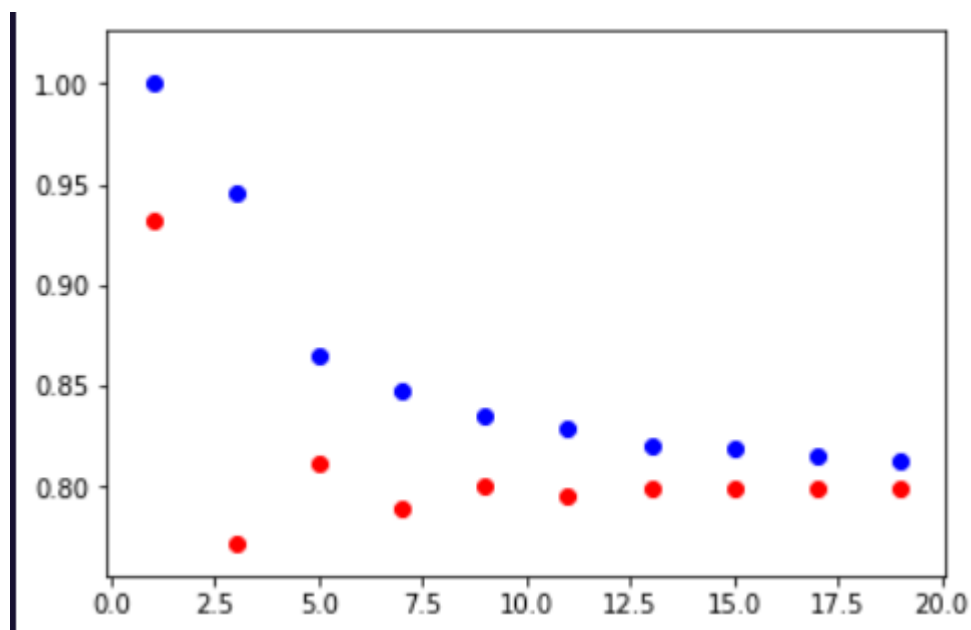
در ادامه با توجه به حلقه نوشته شده با پارامتر های (1,10,2) for دو خروجی k امین نزدیک ترین همسایه های داده ها را چاپ می کنیم(می توان حلقه را گسترش داد):

```
for n in range(1,10,2):
    knn = KNeighborsClassifier(n_neighbors=n)
    knn.fit(x_train,y_train)
    train_score = knn.score(x_train,y_train)
    test_score = knn.score(x_test,y_test)
    train_scores.append(train_score)
    test_scores.append(test_score)
    print(f'KNN : Training score - {train_score} -- Test score - {test_score}')
    knn_scores.append({'algorithm': 'KNN', 'training_score': train_score})
```

خروجی:

```
KNN : Training score - 0.9999795253987429 -- Test score - 0.9323751612308826
KNN : Training score - 0.946233697098749 -- Test score - 0.7712671211842025
KNN : Training score - 0.8647652586965869 -- Test score - 0.8119894355383576
KNN : Training score - 0.847730390450646 -- Test score - 0.7886493458632762
KNN : Training score - 0.8347085440511046 -- Test score - 0.7997051778146306
KNN : Training score - 0.8288528080915624 -- Test score - 0.7950371598796143
KNN : Training score - 0.8205196453799062 -- Test score - 0.7985381733308765
KNN : Training score - 0.8186769312667636 -- Test score - 0.7991523862170629
KNN : Training score - 0.815093876046764 -- Test score - 0.7985995946194951
KNN : Training score - 0.8123502794783072 -- Test score - 0.7995823352373933
```

چنانچه مشهود است این نمودار شباهت زیادی به نمودار نمایان شده در مقاله داشته است.



امتیاز الگوریتم k امین نزدیک ترین همسایه برای داده ها را چاپ می کنیم:

K Neighbors : Training score - 0.8629020699821871 - Test score - 0.8200356243473989

و در نهایت روش مشابه اینبار امتیاز الگوریتم جنگل تصادفی را برای داده ها چاپ می کنیم:

Random Forests : Training score - 0.9999795253987429 - Test score - 0.9623487500767766

-مشاهده میکنیم که روش MLP با دقت 0.947 بیشترین دقت را دارد و GaussianNB با 0.903 کمترین دقت را دارد.

-در روش KNeighborsClassifier تعداد همسایه ها 10 در نظر گرفته شده است و الگوریتم محاسبه نزدیک ترین همسایه auto است

