

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.model_selection import KFold,cross_val_score,GridSearchCV
from sklearn.neighbors import KNeighborsClassifier
```

```
data=pd.read_csv('Downloads/pima-indians-diabetes.data.csv')
data
```

	6	148	72	35	0	33.6	0.627	50	1
0	1	85	66	29	0	26.6	0.351	31	0
1	8	183	64	0	0	23.3	0.672	32	1
2	1	89	66	23	94	28.1	0.167	21	0
3	0	137	40	35	168	43.1	2.288	33	1
4	5	116	74	0	0	25.6	0.201	30	0
...
762	10	101	76	48	180	32.9	0.171	63	0
763	2	122	70	27	0	36.8	0.340	27	0
764	5	121	72	23	112	26.2	0.245	30	0
765	1	126	60	0	0	30.1	0.349	47	1
766	1	93	70	31	0	30.4	0.315	23	0

767 rows × 9 columns

```
x=data.iloc[:,0:8]
y=data.iloc[:,8]
```

```
kfold=KFold(n_splits=10)
model=KNeighborsClassifier()
result=cross_val_score(model,x,y)
print(result.mean())
```

```
0.7248960190136661
```

```
n_neighbors=np.array(range(1,40))
param_grid=dict(n_neighbors=n_neighbors)
model=KNeighborsClassifier()
gsv=GridSearchCV(estimator=model,param_grid=param_grid)
gsv.fit(x,y)
```

```
GridSearchCV(estimator=KNeighborsClassifier(),
              param_grid={'n_neighbors': array([ 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17,
18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34,
35, 36, 37, 38, 39])}))
```

```
krange=range(1,41)
```

```
kscore=[]
```

```
for k in krange:
```

```
    knn=KNeighborsClassifier()
```

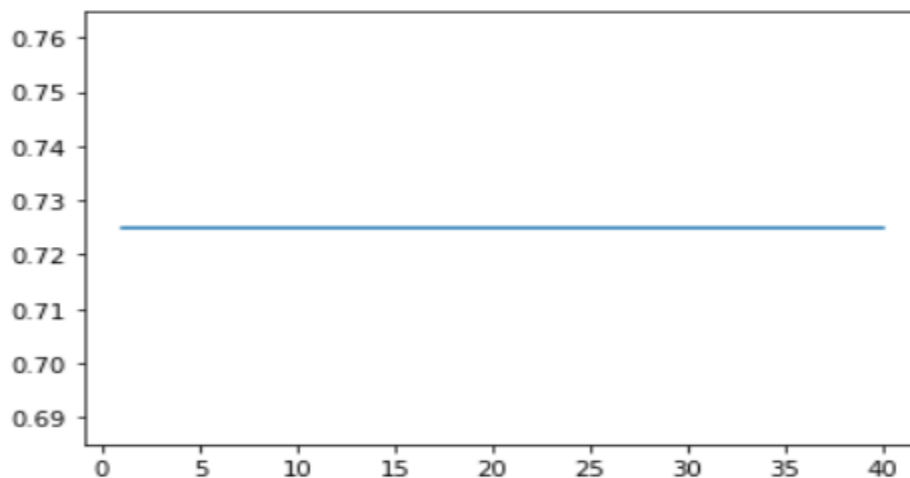
```
    score=cross_val_score(knn,x,y)
```

```
    kscore.append(score.mean())
```

```
import matplotlib.pyplot as plt
```

```
plt.plot(krange,kscore)
```

```
[<matplotlib.lines.Line2D at 0x1651864c760>]
```



```
data.describe()
```

	6	148	72	35	0	33.6	0.627	50	1
count	767.000000	767.000000	767.000000	767.000000	767.000000	767.000000	767.000000	767.000000	767.000000
mean	3.842243	120.859192	69.101695	20.517601	79.903520	31.990482	0.471674	33.219035	0.348110
std	3.370877	31.978468	19.368155	15.954059	115.283105	7.889091	0.331497	11.752296	0.476682
min	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.078000	21.000000	0.000000
25%	1.000000	99.000000	62.000000	0.000000	0.000000	27.300000	0.243500	24.000000	0.000000
50%	3.000000	117.000000	72.000000	23.000000	32.000000	32.000000	0.371000	29.000000	0.000000
75%	6.000000	140.000000	80.000000	32.000000	127.500000	36.600000	0.625000	41.000000	1.000000
max	17.000000	199.000000	122.000000	99.000000	846.000000	67.100000	2.420000	81.000000	1.000000