

```

import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn import svm
from sklearn.svm import SVC
from sklearn.model_selection import train_test_split, cross_val_score, GridSearchCV
from sklearn.preprocessing import StandardScaler
from sklearn.feature_extraction.text import CountVectorizer, TfidfVectorizer
from sklearn.metrics import accuracy_score, confusion_matrix

```

```

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[:,0:7]
y = data.iloc[:,7]

```

```

x_train,x_test,y_train,y_test = train_test_split(x,y,test_size=0.3)

```

```

x_train.shape,x_test.shape,y_train.shape,y_test.shape

```

((536, 7), (231, 7), (536,), (231,))

```

clf=SVC()
param_grid=[{'kernel':['rbf'],'gamma':[10],'C':[10]}]
gsv=GridSearchCV(clf,param_grid,cv=10)
gsv.fit(x_train,y_train)

```

---

```
GridSearchCV(cv=10, estimator=SVC(),
             param_grid=[{'C': [10], 'gamma': [10], 'kernel': ['rbf']}])
```

---

```
gsv.best_params_,gsv.best_score_  
({'C': 10, 'gamma': 10, 'kernel': 'rbf'}, 0.09143955276030748)
```

```
clf=SVC(gamma=10,C=10)  
clf.fit(x_train,y_train)  
predict=clf.predict(x_test)  
acc=accuracy_score(predict,y_test)*100  
print('Accuracy is',acc)  
confusion_matrix(predict,y_test)
```

---

Accuracy is 9.956709956709958

```
array([[ 0,  0,  0, ...,  0,  0,  0],  
       [20, 23, 14, ...,  2,  1,  1],  
       [ 0,  0,  0, ...,  0,  0,  0],  
       ...,  
       [ 0,  0,  0, ...,  0,  0,  0],  
       [ 0,  0,  0, ...,  0,  0,  0],  
       [ 0,  0,  0, ...,  0,  0,  0]], dtype=int64)
```

---

```
plt.hist(data['6'],bins=3,color='red')  
plt.title('Histogram',fontsize=16,fontweight='bold')  
Text(0.5, 1.0, 'Histogram')
```

