CHANDAN KUMAR

ID: GO_STP_13267

```
# libraries
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
```

```
df= pd.read_csv('/content/HR_comma_sep.csv')
df
```

	satisfaction_level	last_evaluation	number_project	average_montly_hours	ti
0	0.38	0.53	2	157	
1	0.80	0.86	5	262	
2	0.11	0.88	7	272	
3	0.72	0.87	5	223	
4	0.37	0.52	2	159	
...	...	...	...	...	
14994	0.40	0.57	2	151	
14995	0.37	0.48	2	160	
14996	0.37	0.53	2	143	
14997	0.11	0.96	6	280	
14998	0.37	0.52	2	158	

14999 rows × 10 columns

```
# laber encoder of data
from sklearn.preprocessing import LabelEncoder
col=['Department','salary']
label_encoder =LabelEncoder()
df['Department']= label_encoder.fit_transform(df['Department'])
df['salary']= label_encoder.fit_transform(df['salary'])
print("after the laber encoder : \n",df)
```

```
after the laber encoder :
        satisfaction_level  last_evaluation  ...  Department  salary
0                    0.38             0.53  ...           7       1
1                    0.80             0.86  ...           7       2
2                    0.11             0.88  ...           7       2
3                    0.72             0.87  ...           7       1
4                    0.37             0.52  ...           7       1
```

```
...                    ...             ...  ...          ...      ...
14994                 0.40            0.57  ...            8        1
14995                 0.37            0.48  ...            8        1
14996                 0.37            0.53  ...            8        1
14997                 0.11            0.96  ...            8        1
14998                 0.37            0.52  ...            8        1

[14999 rows x 10 columns]
```

```
# LogisticRegression of data
from sklearn.linear_model import LogisticRegression
from sklearn.model_selection import train_test_split
from sklearn.metrics import confusion_matrix,accuracy_score
```

training and prediction

```
#prepare model
X_train, X_test, y_train, y_test = train_test_split(df[['Department','satisfaction_leve
X_train
```

	Department	satisfaction_level	salary
9465	7	0.96	2
12556	7	0.46	1
10199	8	0.61	2
8248	9	0.97	1
3522	7	0.74	1
...	...	...	...
4030	0	0.64	1
11132	3	0.43	2
9526	0	0.45	1
9771	5	0.63	2
10967	8	0.20	1

10499 rows × 3 columns

```
#fit model
lr=LogisticRegression()
lr.fit(X_train, y_train)
```

```
LogisticRegression(C=1.0, class_weight=None, dual=False, fit_intercept=True,
                   intercept_scaling=1, l1_ratio=None, max_iter=100,
                   multi_class='auto', n_jobs=None, penalty='l2',
                   random_state=None, solver='lbfgs', tol=0.0001, verbose=0,
                   warm_start=False)
```

```
#prediction
y_test_pred =lr.predict(X_test)
y_test_pred
```

```
array([0, 0, 0, ..., 0, 0, 0])
```

```
y_test
```

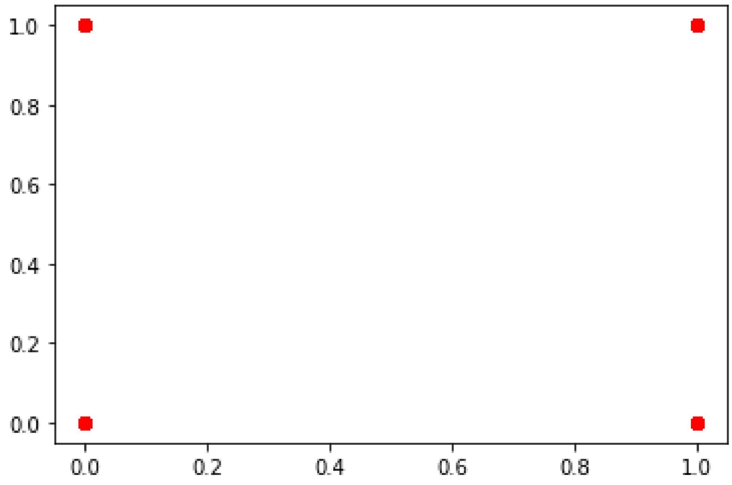
```
10759    0
8888     0
6185     0
3003     0
14254    1
        ..
5813     0
6992     0
2505     0
970      1
3500     0
Name: left, Length: 4500, dtype: int64
```

```
#accuracy
accuracy =accuracy_score(y_test,y_test_pred)
accuracy
```

```
0.7757777777777778
```

model visualisation

```
plt.scatter(y_test,y_test_pred,color='red')
plt.show()
```



✓ 0s completed at 4:56 PM ● ×