

```
In [1]: import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as snb

In [2]: var=pd.read_csv('C://Users/Gopi/Desktop/titanic/train.csv')

In [3]: var['Age'] = var['Age'].fillna(var['Age'].mean())

var['Cabin'] = var.Cabin.fillna(0)

varun=var
varun.drop(['Name','Ticket','Cabin'],axis=1,inplace = True)

a=pd.get_dummies(varun['Sex'])
b=pd.get_dummies(varun['Embarked'])

varun=pd.concat([varun,a,b],axis='columns')

varun.drop(['Sex','Embarked'],axis=1,inplace = True)

varun.drop(['PassengerId','Survived'],axis=1,inplace = True)

In [4]: varun.head()
```

Out[4]:

	Pclass	Age	SibSp	Parch	Fare	female	male	C	Q	S
0	3	22.0	1	0	7.2500	0	1	0	0	1
1	1	38.0	1	0	71.2833	1	0	1	0	0
2	3	26.0	0	0	7.9250	1	0	0	0	1
3	1	35.0	1	0	53.1000	1	0	0	0	1
4	3	35.0	0	0	8.0500	0	1	0	0	1

```
In [5]: q=var.Survived

In [6]: train_df=pd.concat([varun,q],axis='columns')

In [7]: train_df.head()
```

Out[7]:

	Pclass	Age	SibSp	Parch	Fare	female	male	C	Q	S	Survived
0	3	22.0	1	0	7.2500	0	1	0	0	1	0
1	1	38.0	1	0	71.2833	1	0	1	0	0	1
2	3	26.0	0	0	7.9250	1	0	0	0	1	1
3	1	35.0	1	0	53.1000	1	0	0	0	1	1
4	3	35.0	0	0	8.0500	0	1	0	0	1	0

```
In [8]: test_df=pd.read_csv('testdata.csv')

In [9]: final_df=pd.concat([train_df,test_df],axis=0)

C:\Users\Gopi\Anaconda3\lib\site-packages\ipykernel_launcher.py:1: FutureWarning: Sorting becaus
e non-concatenation axis is not aligned. A future version
of pandas will change to not sort by default.

To accept the future behavior, pass 'sort=False'.

To retain the current behavior and silence the warning, pass 'sort=True'.

    """Entry point for launching an IPython kernel.
```

```
In [10]: print(train_df.shape)
print(test_df.shape)
print(final_df.shape)

(891, 11)
(418, 10)
(1309, 11)
```

```
In [11]: final_df =final_df.loc[:,~final_df.columns.duplicated()]
```

```
In [12]: print(final_df.shape)

(1309, 11)
```

```
In [13]: df_Train=final_df.iloc[:891,:]
df_Test=final_df.iloc[891:,:]
```

```
In [14]: print(df_Train.shape)
print(df_Test.shape)

(891, 11)
(418, 11)
```

```
In [15]: df_Test.drop(['Survived'],axis=1,inplace=True)

C:\Users\Gopi\Anaconda3\lib\site-packages\pandas\core\frame.py:4102: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/user_guide/ind
exing.html#returning-a-view-versus-a-copy
    errors=errors,
```

```
In [16]: X_train=df_Train.drop(['Survived'],axis=1)
y_train=df_Train['Survived']
```

MODEL BUILDING

```
In [24]: from sklearn.tree import DecisionTreeClassifier
tree = DecisionTreeClassifier(criterion = 'entropy', random_state = 0)
poi=tree.fit(X_train,y_train)
```

```
In [52]: df_Test['Fare']=df_Test['Fare'].fillna(df_Test['Fare'].mean())

C:\Users\Gopi\Anaconda3\lib\site-packages\ipykernel_launcher.py:1: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/user_guide/ind
exing.html#returning-a-view-versus-a-copy
    """Entry point for launching an IPython kernel.
```

```
In [53]: y_pred=tree.predict(df_Test)
print(y_pred)

[0. 0. 1. 1. 1. 0. 0. 0. 0. 0. 0. 0. 0. 0. 1. 0. 1. 1. 0. 1. 1. 0. 1. 1. 0.
 1. 0. 1. 1. 1. 0. 0. 0. 1. 0. 1. 1. 0. 0. 1. 0. 0. 0. 1. 1. 0. 0. 0. 1. 1. 0.
 1. 1. 0. 0. 1. 0. 1. 0. 0. 0. 0. 1. 0. 0. 0. 1. 1. 1. 0. 0. 0. 1. 1. 0. 0.
 0. 0. 1. 0. 0. 1. 0. 1. 1. 0. 0. 0. 0. 0. 1. 0. 0. 1. 1. 0. 0. 0. 0. 0. 0.
 1. 1. 1. 1. 1. 0. 0. 0. 1. 0. 0. 0. 1. 0. 0. 0. 1. 0. 1. 0. 0. 1. 1. 1.
 1. 0. 1. 0. 0. 1. 0. 1. 0. 0. 1. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 1. 1. 0.
 0. 0. 0. 0. 0. 0. 1. 0. 0. 1. 0. 0. 1. 0. 0. 1. 0. 1. 1. 0. 0. 1. 1. 0.
 1. 0. 0. 0. 0. 0. 1. 1. 1. 1. 1. 0. 1. 0. 1. 0. 1. 0. 1. 0. 0. 0. 0. 0.
 1. 1. 1. 0. 1. 0. 0. 0. 1. 1. 0. 1. 0. 1. 0. 0. 0. 1. 0. 0. 0. 0. 0. 0. 0.
 0. 0. 1. 0. 1. 0. 1. 1. 1. 0. 0. 0. 0. 0. 0. 1. 1. 0. 1. 0. 1. 1. 1. 1.
 1. 1. 0. 0. 0. 1. 0. 1. 0. 1. 0. 1. 0. 0. 0. 0. 0. 1. 0. 0. 0. 1. 0. 0. 1. 1.
 0. 0. 0. 0. 0. 1. 0. 1. 0. 0. 0. 0. 1. 1. 0. 1. 0. 0. 0. 1. 0. 0. 1. 0. 0.
 0. 0. 1. 1. 1. 0. 0. 0. 0. 0. 0. 1. 1. 0. 1. 0. 0. 0. 1. 0. 0. 1. 0. 0.
 0. 0. 0. 0. 0. 0. 1. 0. 1. 0. 0. 0. 1. 1. 0. 0. 0. 0. 0. 0. 1. 0. 0. 1.
 0. 1. 1. 1. 0. 0. 1. 1. 1. 0. 1. 0. 1. 0. 0. 0. 0. 0. 0. 0. 0. 1. 1.
 0. 1. 0. 0. 1. 0. 1. 1. 1. 0. 0. 1. 0. 1. 0. 0. 1. 0. 1. 0. 0. 1. 0. 0.
 0. 1. 0. 1. 0. 0. 1. 0. 0. 0.]
```

```
In [56]: pred=pd.DataFrame(y_pred)
sub_df=pd.read_csv('C://Users/Gopi/Desktop/titanic/final_sub.csv')
datasets=pd.concat([sub_df['PassengerId'],pred],axis=1)
datasets.columns=['PassengerId','Survived']
datasets.to_csv('sample_submission.csv',index=False)
```