

```
In [1]: import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sn
from sklearn.linear_model import LogisticRegression
```

```
In [2]: df=pd.read_csv('test g csv').dropna()
df2 = pd.read_csv("train g csv")
df
```

```
Out[2]:
```

| | PassengerId | Pclass | Name | Sex | Age | SibSp | Parch | Ticket | Fare | Cabin |
|-----|-------------|--------|---|--------|------|-------|-------|------------------|----------|--------------------------|
| 12 | 904 | 1 | Snyder, Mrs. John Pillsbury (Nelle Stevenson) | female | 23.0 | 1 | 0 | 21228 | 82.2667 | B45 |
| 14 | 906 | 1 | Chaffee, Mrs. Herbert Fuller (Carrie Constance... | female | 47.0 | 1 | 0 | W.E.P. 5734 | 61.1750 | E31 |
| 24 | 916 | 1 | Ryerson, Mrs. Arthur Larned (Emily Maria Borie) | female | 48.0 | 1 | 3 | PC 17608 | 262.3750 | B57 B59 B63 B66 |
| 26 | 918 | 1 | Ostby, Miss. Helene Ragnhild | female | 22.0 | 0 | 1 | 113509 | 61.9792 | B36 |
| 28 | 920 | 1 | Brady, Mr. John Bertram | male | 41.0 | 0 | 0 | 113054 | 30.5000 | A21 |
| ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... |
| 404 | 1296 | 1 | Frauenthal, Mr. Isaac Gerald | male | 43.0 | 1 | 0 | 17765 | 27.7208 | D40 |
| 405 | 1297 | 2 | Nourney, Mr. Alfred (Baron von Drachstedt)" | male | 20.0 | 0 | 0 | SC/PARIS 2166 | 13.8625 | D38 |
| 407 | 1299 | 1 | Widener, Mr. George Dunton | male | 50.0 | 1 | 1 | 113503 | 211.5000 | C80 |
| 411 | 1303 | 1 | Minahan, Mrs. William Edward (Lillian E Thorpe) | female | 37.0 | 1 | 0 | 19928 | 90.0000 | C78 |
| 414 | 1306 | 1 | Oliva y Ocana, Dona. Fermina | female | 39.0 | 0 | 0 | PC 17758 | 108.9000 | C105 |

87 rows × 11 columns



```
In [3]: df.dropna(inplace=True)
df2.dropna(inplace=True)
```

In [4]: df2

Out[4]:

| | PassengerId | Survived | Pclass | Name | Sex | Age | SibSp | Parch | Ticket | Fare | Ca | |
|--|-------------|----------|--------|------|---|--------|-------|-------|--------|----------|---------|---|
| | 1 | 2 | 1 | 1 | Cumings, Mrs. John Bradley (Florence Briggs Th... | female | 38.0 | 1 | 0 | PC 17599 | 71.2833 | (|
| | 3 | 4 | 1 | 1 | Futrelle, Mrs. Jacques Heath (Lily May Peel) | female | 35.0 | 1 | 0 | 113803 | 53.1000 | C |
| | 6 | 7 | 0 | 1 | McCarthy, Mr. Timothy J | male | 54.0 | 0 | 0 | 17463 | 51.8625 | I |
| | 10 | 11 | 1 | 3 | Sandstrom, Miss. Marguerite Rut | female | 4.0 | 1 | 1 | PP 9549 | 16.7000 | |
| | 11 | 12 | 1 | 1 | Bonnell, Miss. Elizabeth | female | 58.0 | 0 | 0 | 113783 | 26.5500 | C |
| | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | |
| | 871 | 872 | 1 | 1 | Beckwith, Mrs. Richard Leonard (Sallie Monypeny) | female | 47.0 | 1 | 1 | 11751 | 52.5542 | I |
| | 872 | 873 | 0 | 1 | Carlsson, Mr. Frans Olof | male | 33.0 | 0 | 0 | 695 | 5.0000 | I |
| | 879 | 880 | 1 | 1 | Potter, Mrs. Thomas Jr (Lily Alexenia Wilson) | female | 56.0 | 0 | 1 | 11767 | 83.1583 | (|
| | 887 | 888 | 1 | 1 | Graham, Miss. Margaret Edith | female | 19.0 | 0 | 0 | 112053 | 30.0000 | I |
| | 889 | 890 | 1 | 1 | Behr, Mr. Karl Howell | male | 26.0 | 0 | 0 | 111369 | 30.0000 | C |

183 rows × 12 columns

In [5]: df.info()

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 87 entries, 12 to 414
Data columns (total 11 columns):
 #   Column          Non-Null Count  Dtype
---  -
 0   PassengerId     87 non-null    int64
 1   Pclass         87 non-null    int64
 2   Name           87 non-null    object
 3   Sex            87 non-null    object
 4   Age            87 non-null    float64
 5   SibSp          87 non-null    int64
 6   Parch          87 non-null    int64
 7   Ticket         87 non-null    object
 8   Fare           87 non-null    float64
 9   Cabin          87 non-null    object
10   Embarked       87 non-null    object
dtypes: float64(2), int64(4), object(5)
memory usage: 8.2+ KB
```

In [6]: feature_matrix = df[['PassengerId', 'Pclass', 'Age', 'SibSp', 'Parch', 'Fare']]
target_vector = df['Embarked']

In [7]: feature_matrix.shape

Out[7]: (87, 6)

In [8]: target_vector.shape

Out[8]: (87,)

In [9]: from sklearn.preprocessing import StandardScaler

In [10]: fs = StandardScaler().fit_transform(feature_matrix)

In [11]: logr=LogisticRegression()
logr.fit(fs,target_vector)

Out[11]: LogisticRegression()

In [12]: observation = df2[['PassengerId', 'Pclass', 'Age', 'SibSp', 'Parch', 'Fare']]

```
In [13]: prediction = logr.predict(observation)
prediction
```

```
Out[13]: array(['C', 'C', 'C', 'Q', 'C', 'Q', 'Q', 'C', 'Q', 'Q', 'Q', 'Q', 'Q', 'Q',
                'Q', 'Q', 'Q', 'Q', 'Q', 'Q', 'Q', 'Q', 'Q', 'Q', 'Q', 'Q',
                'Q', 'Q', 'Q', 'Q', 'Q', 'Q', 'Q', 'Q', 'Q', 'Q', 'Q', 'Q', 'Q',
                'Q', 'Q', 'Q', 'Q', 'Q', 'Q', 'Q', 'Q', 'Q', 'Q', 'Q', 'Q', 'Q',
                'Q', 'Q', 'Q', 'Q', 'Q', 'Q', 'Q', 'Q', 'Q', 'Q', 'Q', 'Q', 'Q',
                'Q', 'Q', 'Q', 'Q', 'Q', 'Q', 'Q', 'Q', 'Q', 'Q', 'Q', 'Q', 'Q',
                'Q', 'Q', 'Q', 'Q', 'Q', 'Q', 'Q', 'Q', 'Q', 'Q', 'Q', 'Q', 'Q',
                'Q', 'Q', 'Q', 'Q', 'Q', 'Q', 'Q', 'Q', 'Q', 'Q', 'Q', 'Q', 'Q',
                'Q', 'Q', 'Q', 'Q', 'Q', 'Q', 'Q', 'Q', 'Q', 'Q', 'Q', 'Q', 'Q',
                'Q', 'Q', 'Q', 'Q', 'Q', 'Q', 'Q', 'Q', 'Q', 'Q', 'Q', 'Q', 'Q',
                'Q', 'Q', 'Q', 'Q', 'Q', 'Q', 'Q', 'Q', 'Q', 'Q', 'Q', 'Q', 'Q',
                'Q', 'Q', 'Q', 'Q', 'Q', 'Q', 'Q', 'Q', 'Q', 'Q', 'Q', 'Q', 'Q',
                'Q'], dtype=object)
```

```
In [14]: logr.classes_
```

```
Out[14]: array(['C', 'Q', 'S'], dtype=object)
```

```
In [15]: logr.predict_proba(observation)[0][0]
```

```
Out[15]: 0.99999999999955971
```

```
In [16]: logr.predict_proba(observation)[0][1]
```

```
Out[16]: 4.402884174313719e-12
```

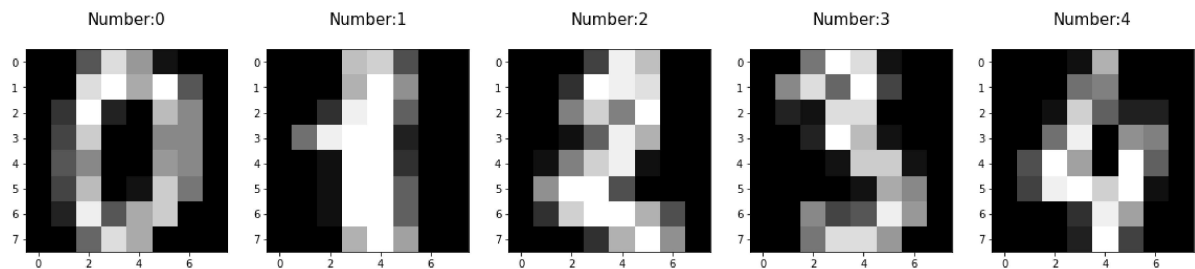
Logistic Regression 2

```
In [17]: import re
from sklearn.datasets import load_digits
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.linear_model import LogisticRegression
from sklearn.model_selection import train_test_split
```

```
In [18]: digits = load_digits()
digits
```

```
Out[18]: {'data': array([[ 0.,  0.,  5., ...,  0.,  0.,  0.],
        [ 0.,  0.,  0., ..., 10.,  0.,  0.],
        [ 0.,  0.,  0., ..., 16.,  9.,  0.],
        ...,
        [ 0.,  0.,  1., ...,  6.,  0.,  0.],
        [ 0.,  0.,  2., ..., 12.,  0.,  0.],
        [ 0.,  0., 10., ..., 12.,  1.,  0.])),
  'target': array([0, 1, 2, ..., 8, 9, 8]),
  'frame': None,
  'feature_names': ['pixel_0_0',
    'pixel_0_1',
    'pixel_0_2',
    'pixel_0_3',
    'pixel_0_4',
    'pixel_0_5',
    'pixel_0_6',
    'pixel_0_7',
    'pixel_1_0',
    'pixel_1_1',
    ...]
```

```
In [19]: plt.figure(figsize=(20,4))
for index,(image,label) in enumerate(zip(digits.data[0:5],digits.target[0:5])):
    plt.subplot(1,5,index+1)
    plt.imshow(np.reshape(image,(8,8)),cmap=plt.cm.gray)
    plt.title("Number:%i\n"%label,fontsize=15)
```



```
In [20]: X_train,X_test,y_train,y_test=train_test_split(digits.data,digits.target,test_
```

```
In [21]: print(X_train.shape)
print(X_test.shape)
print(y_train.shape)
print(y_test.shape)
```

```
(1257, 64)
(540, 64)
(1257,)
(540,)
```

```
In [22]: logre=LogisticRegression(max_iter=10000)
logre.fit(X_train,y_train)
```

```
Out[22]: LogisticRegression(max_iter=10000)
```

Loading [MathJax]/jax/output/HTML-CSS/fonts/STIX-Web/fontdata.js

```
In [23]: print(logre.score(X_test,y_test))
```

```
0.975925925925926
```

Random Forest

```
In [24]: df['Embarked'].value_counts()
```

```
Out[24]: C    47  
         S    39  
         Q     1  
         Name: Embarked, dtype: int64
```

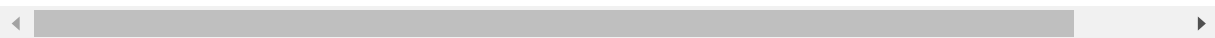
```
In [32]: x=df[['PassengerId','Pclass','Age','SibSp','Parch','Fare']]  
         y=df['Embarked']
```

```
In [33]: g1={'Embarked':{'C':1, 'S':2, 'Q':3}}
df=df.replace(g1)
df
```

```
Out[33]:
```

| | PassengerId | Pclass | Name | Sex | Age | SibSp | Parch | Ticket | Fare | Cabin |
|-----|-------------|--------|---|--------|------|-------|-------|------------------|----------|--------------------------|
| 12 | 904 | 1 | Snyder, Mrs. John Pillsbury (Nelle Stevenson) | female | 23.0 | 1 | 0 | 21228 | 82.2667 | B45 |
| 14 | 906 | 1 | Chaffee, Mrs. Herbert Fuller (Carrie Constance... | female | 47.0 | 1 | 0 | W.E.P. 5734 | 61.1750 | E31 |
| 24 | 916 | 1 | Ryerson, Mrs. Arthur Larned (Emily Maria Borie) | female | 48.0 | 1 | 3 | PC 17608 | 262.3750 | B57 B59 B63 B66 |
| 26 | 918 | 1 | Ostby, Miss. Helene Ragnhild | female | 22.0 | 0 | 1 | 113509 | 61.9792 | B36 |
| 28 | 920 | 1 | Brady, Mr. John Bertram | male | 41.0 | 0 | 0 | 113054 | 30.5000 | A21 |
| ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... |
| 404 | 1296 | 1 | Frauenthal, Mr. Isaac Gerald | male | 43.0 | 1 | 0 | 17765 | 27.7208 | D40 |
| 405 | 1297 | 2 | Nourney, Mr. Alfred (Baron von Drachstedt)" | male | 20.0 | 0 | 0 | SC/PARIS 2166 | 13.8625 | D38 |
| 407 | 1299 | 1 | Widener, Mr. George Dunton | male | 50.0 | 1 | 1 | 113503 | 211.5000 | C80 |
| 411 | 1303 | 1 | Minahan, Mrs. William Edward (Lillian E Thorpe) | female | 37.0 | 1 | 0 | 19928 | 90.0000 | C78 |
| 414 | 1306 | 1 | Oliva y Ocana, Dona. Fermina | female | 39.0 | 0 | 0 | PC 17758 | 108.9000 | C105 |

87 rows × 11 columns



```
In [34]: from sklearn.model_selection import train_test_split
x_train,x_test,y_train,y_test=train_test_split(x,y,train_size=0.70)
```



```
In [35]: from sklearn.ensemble import RandomForestClassifier
rfc = RandomForestClassifier()
rfc.fit(x_train,y_train)
```

Out[35]: RandomForestClassifier()

```
In [36]: parameters = {'max_depth':[1,2,3,4,5], 'min_samples_leaf':[5,10,15,20,25],
                        'n_estimators': [10,20,30,40,50]}
}
```

```
In [37]: from sklearn.model_selection import GridSearchCV
grid_search = GridSearchCV(estimator=rfc,param_grid=parameters,cv=2,scoring="a
grid_search.fit(x_train,y_train)
```

C:\ProgramData\Anaconda3\lib\site-packages\sklearn\model_selection_split.py:
666: UserWarning: The least populated class in y has only 1 members, which is
less than n_splits=2.
warnings.warn(("The least populated class in y has only %d"

Out[37]: GridSearchCV(cv=2, estimator=RandomForestClassifier(),
param_grid={'max_depth': [1, 2, 3, 4, 5],
'min_samples_leaf': [5, 10, 15, 20, 25],
'n_estimators': [10, 20, 30, 40, 50]},
scoring='accuracy')

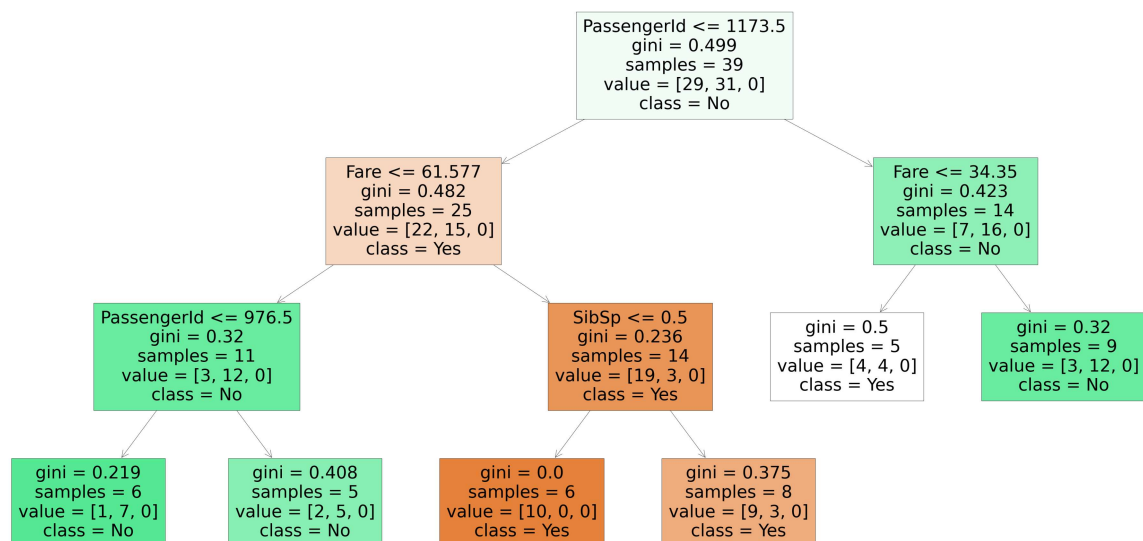
```
In [38]: grid_search.best_score_
```

Out[38]: 0.6499999999999999

```
In [39]: rfc_best = grid_search.best_estimator_
```

```
In [40]: from sklearn.tree import plot_tree
plt.figure(figsize = (80,40,))
plot_tree(rfc_best.estimators_[5],feature_names=x.columns,class_names=['Yes','No'])
```

```
Out[40]: [Text(2637.818181818182, 1902.6000000000001, 'PassengerId <= 1173.5\nngini = 0.499\nnsamples = 39\nnvalue = [29, 31, 0]\nnclass = No'),
Text(1623.2727272727273, 1359.0, 'Fare <= 61.577\nngini = 0.482\nnsamples = 25\nnvalue = [22, 15, 0]\nnclass = Yes'),
Text(811.6363636363636, 815.4000000000001, 'PassengerId <= 976.5\nngini = 0.32\nnsamples = 11\nnvalue = [3, 12, 0]\nnclass = No'),
Text(405.8181818181818, 271.79999999999995, 'gini = 0.219\nnsamples = 6\nnvalue = [1, 7, 0]\nnclass = No'),
Text(1217.4545454545455, 271.79999999999995, 'gini = 0.408\nnsamples = 5\nnvalue = [2, 5, 0]\nnclass = No'),
Text(2434.909090909091, 815.4000000000001, 'SibSp <= 0.5\nngini = 0.236\nnsamples = 14\nnvalue = [19, 3, 0]\nnclass = Yes'),
Text(2029.090909090909, 271.79999999999995, 'gini = 0.0\nnsamples = 6\nnvalue = [10, 0, 0]\nnclass = Yes'),
Text(2840.7272727272725, 271.79999999999995, 'gini = 0.375\nnsamples = 8\nnvalue = [9, 3, 0]\nnclass = Yes'),
Text(3652.3636363636365, 1359.0, 'Fare <= 34.35\nngini = 0.423\nnsamples = 14\nnvalue = [7, 16, 0]\nnclass = No'),
Text(3246.5454545454545, 815.4000000000001, 'gini = 0.5\nnsamples = 5\nnvalue = [4, 4, 0]\nnclass = Yes'),
Text(4058.181818181818, 815.4000000000001, 'gini = 0.32\nnsamples = 9\nnvalue = [3, 12, 0]\nnclass = No')]
```



In []: