

```
In [48]: #data preprocessing
#import the libraries
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
```

```
In [49]: #import dataset
df = pd.read_excel('titanic.xls')
```

```
In [50]: #first 5 rows in the dataframe
df.head()
```

Out[50]:

	pclass	survived	name	sex	age	sibsp	parch	ticket	fare	cabin	embarke
0	1	1	Allen, Miss. Elisabeth Walton	female	29.0000	0	0	24160	211.3375	B5	S
1	1	1	Allison, Master. Hudson Trevor	male	0.9167	1	2	113781	151.5500	C22 C26	S
2	1	0	Allison, Miss. Helen Loraine	female	2.0000	1	2	113781	151.5500	C22 C26	S
3	1	0	Allison, Mr. Hudson Joshua Creighton	male	30.0000	1	2	113781	151.5500	C22 C26	S

pclass	survived	name	sex	age	sibsp	parch	ticket	fare	cabin	embarked	
4	1	0	Allison, Mrs. Hudson J C (Bessie Waldo Daniels)	female	25.0000	1	2	113781	151.5500	C22 C26	S

In [51]: `df.describe()`

Out[51]:

	pclass	survived	age	sibsp	parch	fare	body
count	1309.000000	1309.000000	1046.000000	1309.000000	1309.000000	1308.000000	121.000000
mean	2.294882	0.381971	29.881135	0.498854	0.385027	33.295479	160.809917
std	0.837836	0.486055	14.413500	1.041658	0.865560	51.758668	97.696922
min	1.000000	0.000000	0.166700	0.000000	0.000000	0.000000	1.000000
25%	2.000000	0.000000	21.000000	0.000000	0.000000	7.895800	72.000000
50%	3.000000	0.000000	28.000000	0.000000	0.000000	14.454200	155.000000
75%	3.000000	1.000000	39.000000	1.000000	0.000000	31.275000	256.000000
max	3.000000	1.000000	80.000000	8.000000	9.000000	512.329200	328.000000

In [52]: `df.info()`

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1309 entries, 0 to 1308
Data columns (total 14 columns):
pclass      1309 non-null int64
survived    1309 non-null int64
name        1309 non-null object
sex         1309 non-null object
age         1046 non-null float64
sibsp       1309 non-null int64
parch       1309 non-null int64
```

```
ticket      1309 non-null object
fare        1308 non-null float64
cabin       295 non-null object
embarked    1307 non-null object
boat        486 non-null object
body        121 non-null float64
home.dest   745 non-null object
dtypes: float64(3), int64(4), object(7)
memory usage: 143.2+ KB
```

```
In [53]: df.count()
```

```
Out[53]: pclass      1309
survived    1309
name        1309
sex         1309
age         1046
sibsp       1309
parch       1309
ticket      1309
fare        1308
cabin       295
embarked    1307
boat        486
body        121
home.dest   745
dtype: int64
```

```
In [54]: #drop the columns with alot of null values
df.drop('cabin', axis = 1 , inplace= True)
df.drop('body', axis = 1 , inplace= True)
df.drop('boat', axis = 1 , inplace= True)
df.drop('home.dest', axis = 1 , inplace= True)
```

```
In [55]: df.head()
```

```
Out[55]:
```

pclass	survived	name	sex	age	sibsp	parch	ticket	fare	embarked
--------	----------	------	-----	-----	-------	-------	--------	------	----------

	pclass	survived	name	sex	age	sibsp	parch	ticket	fare	embarked
0	1	1	Allen, Miss. Elisabeth Walton	female	29.0000	0	0	24160	211.3375	S
1	1	1	Allison, Master. Hudson Trevor	male	0.9167	1	2	113781	151.5500	S
2	1	0	Allison, Miss. Helen Loraine	female	2.0000	1	2	113781	151.5500	S
3	1	0	Allison, Mr. Hudson Joshua Creighton	male	30.0000	1	2	113781	151.5500	S
4	1	0	Allison, Mrs. Hudson J C (Bessie Waldo Daniels)	female	25.0000	1	2	113781	151.5500	S

```
In [56]: #check the null values
df.isnull().sum()
```

```
Out[56]: pclass      0
survived    0
name        0
sex         0
age        263
sibsp       0
parch       0
ticket      0
fare        1
embarked    2
dtype: int64
```

```
In [57]: #fill null values
mean_age = df['age'].mean()
int_age = int(mean_age)
```

```
In [58]: df['age'].fillna(int_age , inplace = True)
```

```
In [59]: df['embarked'].fillna(df['embarked'].mode()[0] , inplace = True)
```

```
In [60]: df['fare'].fillna(df['fare'].median() , inplace = True)
```

```
In [61]: df.isnull().sum()
```

```
Out[61]: pclass      0
survived    0
name        0
sex         0
age         0
sibsp       0
parch       0
ticket      0
fare        0
embarked    0
dtype: int64
```

```
In [72]: #convert the categorical data to numerical data
from sklearn.preprocessing import LabelEncoder
```

```
In [73]: df.head()
```

```
Out[73]:
```

	pclass	survived	name	sex	age	sibsp	parch	ticket	fare	embarked
0	1	1	Allen, Miss. Elisabeth Walton	0	29.0000	0	0	24160	211.3375	2
1	1	1	Allison, Master. Hudson Trevor	1	0.9167	1	2	113781	151.5500	2
2	1	0	Allison, Miss. Helen Loraine	0	2.0000	1	2	113781	151.5500	2
3	1	0	Allison, Mr. Hudson Joshua Creighton	1	30.0000	1	2	113781	151.5500	2

pclass	survived	name	sex	age	sibsp	parch	ticket	fare	embarked	
4	1	0	Allison, Mrs. Hudson J C (Bessie Waldo Daniels)	0	25.0000	1	2	113781	151.5500	2

```
In [74]: sex_encode = LabelEncoder()
df['sex'] = sex_encode.fit_transform(df['sex'])
```

```
In [75]: embarked_encode = LabelEncoder()
df['embarked'] = embarked_encode.fit_transform(df['embarked'])
```

```
In [76]: df.head()
```

Out[76]:

	pclass	survived	name	sex	age	sibsp	parch	ticket	fare	embarked
0	1	1	Allen, Miss. Elisabeth Walton	0	29.0000	0	0	24160	211.3375	2
1	1	1	Allison, Master. Hudson Trevor	1	0.9167	1	2	113781	151.5500	2
2	1	0	Allison, Miss. Helen Loraine	0	2.0000	1	2	113781	151.5500	2
3	1	0	Allison, Mr. Hudson Joshua Creighton	1	30.0000	1	2	113781	151.5500	2
4	1	0	Allison, Mrs. Hudson J C (Bessie Waldo Daniels)	0	25.0000	1	2	113781	151.5500	2

```
In [77]: #splittting data
from sklearn.model_selection import train_test_split
```

```
In [78]: X_columns = ['pclass', 'sex', 'age', 'sibsp', 'parch', 'embarked']
X = df[X_columns].values
```

```
y = df['survived'].values
```

```
In [79]: X_train, X_test, y_train, y_test = train_test_split(X, y, test_size =  
0.35, random_state = 0)
```

```
In [90]: #import the machine model  
from sklearn.linear_model import LogisticRegression
```

```
In [91]: # create and configure the model  
classifier = LogisticRegression(solver='lbfgs')
```

```
In [92]: #fit the model  
classifier.fit(X_train, y_train)
```

```
Out[92]: LogisticRegression(C=1.0, class_weight=None, dual=False, fit_intercept=  
True,  
                           intercept_scaling=1, l1_ratio=None, max_iter=100,  
                           multi_class='warn', n_jobs=None, penalty='l2',  
                           random_state=None, solver='lbfgs', tol=0.0001, verbo  
se=0,  
                           warm_start=False)
```

```
In [93]: #predict regressor(test)  
y_pred = classifier.predict(X_test)
```

```
In [94]: y_pred
```

```
Out[94]: array([0, 1, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 1, 0, 1, 0, 0,  
0,  
           0, 1, 0, 1, 0, 0, 0, 1, 1, 1, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0,  
0,  
           1, 0, 0, 1, 0, 0, 1, 0, 0, 1, 0, 0, 1, 0, 1, 1, 0, 0, 0, 0, 1,  
1,  
           0, 0, 0, 1, 1, 0, 0, 0, 1, 1, 0, 1, 1, 0, 0, 1, 0, 0, 0, 1, 0,  
0,  
           0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 1, 0, 0,  
0,
```

```

0, 1, 0, 0, 1, 0, 0, 0, 0, 1, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0,
1,
0, 1, 0, 0, 0, 1, 0, 1, 0, 1, 0, 0, 0, 0, 1, 0, 1, 0, 1, 0,
1,
0, 0, 1, 1, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 1, 0, 1,
0,
0, 1, 1, 1, 1, 0, 1, 0, 1, 1, 1, 0, 0, 0, 0, 0, 1, 0, 0, 1, 0,
1,
0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0, 0,
0,
1, 1, 1, 1, 1, 0, 1, 1, 1, 1, 1, 0, 0, 1, 0, 0, 1, 0, 0, 0, 0,
0,
0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0,
1,
0, 1, 0, 0, 0, 1, 0, 1, 0, 0, 0, 1, 1, 1, 0, 1, 0, 1, 0, 0, 1,
0,
0, 0, 1, 0, 1, 0, 1, 1, 1, 0, 0, 0, 0, 0, 1, 0, 0, 1, 0, 1,
0,
1, 0, 1, 0, 0, 0, 0, 1, 0, 1, 0, 1, 0, 0, 1, 0, 1, 0, 0, 0, 0,
0,
1, 0, 0, 1, 0, 1, 0, 0, 1, 0, 1, 0, 1, 1, 1, 1, 0, 0, 0, 0, 1,
1,
0, 0, 1, 0, 0, 0, 1, 0, 0, 0, 0, 0, 1, 1, 1, 0, 0, 0, 1, 0, 1,
0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 0, 0, 1, 1, 1,
1,
0, 1, 0, 1, 1, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 1, 0, 1, 1, 0, 0,
0,
0, 0, 1, 0, 1, 0, 0, 0, 1, 0, 0, 0, 0, 0, 1, 0, 0, 0, 1, 0, 0,
0,
0, 0, 0, 1, 1, 0, 0, 1, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0],
dtype=int64)

```

```

In [95]: #evaluation
from sklearn.metrics import accuracy_score

```

```

In [96]: round(accuracy_score(y_pred,y_test)*100,1)

```

```

Out[96]: 80.2

```



In [ ]:

In [ ]:

In [ ]: