# Titanic

December 15, 2022

```
In [114]: import pandas as pd
          import numpy as np
          from sklearn import preprocessing
          from sklearn.model_selection import train_test_split
          from sklearn.metrics import accuracy_score
          from sklearn.linear_model import LogisticRegression

          df = pd.read_csv('/public/bmort/python/titanic.csv')
          print(df.isnull().sum())
          print("")
          print("Yes there is missing values in the data frame")
          print("There are 177 missing values in the Age column")
          print("There are 687 missing values in the Cabin column")
          print("There are 2 missing values in the Embarked column")
          print("")

          sur = df['Survived']
          sum = sur.sum()
          surv = ((sum)/sur.count())*100
          print(surv)
          print("38.38% of passengers survived")
          print("")

          fare = df['Fare']
          max = fare[0]

          for i in range (0,len(fare)):
              if (fare[i] > max):
                  max = fare[i]

          print(max)
          print("The maximum fare that was paid to purchase a ticket by a passenger was 512.329
          print("")

          emb = df['Embarked']
          emblist = emb.tolist()
          embSet = set(emblist)
          print(embSet)
```

```
        print("There are 3 unique places the passengers embarked from")
        print(df)
```

```
PassengerId      0
Survived         0
Pclass           0
Name             0
Sex              0
Age            177
SibSp            0
Parch            0
Ticket           0
Fare             0
Cabin          687
Embarked         2
dtype: int64

Yes there is missing values in the data frame
There are 177 missing values in the Age column
There are 687 missing values in the Cabin column
There are 2 missing values in the Embarked column

38.38383838383838
38.38% of passengers survived

512.3292
The maximum fare that was paid to purchase a ticket by a passenger was 512.3292

{nan, 'S', 'Q', 'C'}
There are 3 unique places the passengers embarked from
     PassengerId  Survived  Pclass  \
0              1         0       3
1              2         1       1
2              3         1       3
3              4         1       1
4              5         0       3
..           ...       ...     ...
886          887         0       2
887          888         1       1
888          889         0       3
889          890         1       1
890          891         0       3

                                                  Name     Sex   Age  SibSp  \
0                              Braund, Mr. Owen Harris    male  22.0      1
1    Cumings, Mrs. John Bradley (Florence Briggs Th...  female  38.0      1
2                               Heikkinen, Miss. Laina  female  26.0      0
3         Futrelle, Mrs. Jacques Heath (Lily May Peel)  female  35.0      1
```

```
4                             Allen, Mr. William Henry    male  35.0      0
..                                                 ...     ...   ...    ...
886                              Montvila, Rev. Juozas    male  27.0      0
887                       Graham, Miss. Margaret Edith  female  19.0      0
888           Johnston, Miss. Catherine Helen "Carrie"  female   NaN      1
889                              Behr, Mr. Karl Howell    male  26.0      0
890                                Dooley, Mr. Patrick    male  32.0      0

     Parch            Ticket     Fare Cabin Embarked
0        0         A/5 21171   7.2500   NaN        S
1        0          PC 17599  71.2833   C85        C
2        0  STON/O2. 3101282   7.9250   NaN        S
3        0            113803  53.1000  C123        S
4        0            373450   8.0500   NaN        S
..     ...               ...      ...   ...      ...
886      0            211536  13.0000   NaN        S
887      0            112053  30.0000   B42        S
888      2        W./C. 6607  23.4500   NaN        S
889      0            111369  30.0000  C148        C
890      0            370376   7.7500   NaN        Q

[891 rows x 12 columns]
```

```
In [115]: print("")
          imputed_value = df['Age'].median()
          df['Age'].fillna(imputed_value)
          df['Age'] = df['Age'].fillna(imputed_value)
          age = np.array(df['Age'])
          SibSp = np.array(df['SibSp'])
          Parch = np.array(df['Parch'])
          Fare = np.array(df['Fare'])

          norm_age = preprocessing.normalize([age])
          norm_SibSp = preprocessing.normalize([SibSp])
          norm_parch = preprocessing.normalize([Parch])
          norm_fare = preprocessing.normalize([Fare])



          df['Age'] = norm_age.T
          df['SibSp'] = norm_SibSp.T
          df['Parch'] = norm_parch.T
          df['Fare'] = norm_fare.T


          df.head()
```

```
Out[115]:    PassengerId  Survived  Pclass  \
          0            1         0       3
          1            2         1       1
          2            3         1       3
          3            4         1       1
          4            5         0       3

                                                          Name     Sex       Age  \
          0                            Braund, Mr. Owen Harris    male  0.022949
          1  Cumings, Mrs. John Bradley (Florence Briggs Th...  female  0.039639
          2                             Heikkinen, Miss. Laina  female  0.027122
          3       Futrelle, Mrs. Jacques Heath (Lily May Peel)  female  0.036510
          4                           Allen, Mr. William Henry    male  0.036510

                SibSp  Parch            Ticket      Fare Cabin Embarked
          0  0.027462    0.0         A/5 21171  0.004103   NaN        S
          1  0.027462    0.0          PC 17599  0.040344   C85        C
          2  0.000000    0.0  STON/O2. 3101282  0.004485   NaN        S
          3  0.027462    0.0            113803  0.030053  C123        S
          4  0.000000    0.0            373450  0.004556   NaN        S
```

```
In [116]: print("")
          le = preprocessing.LabelEncoder()
          le.fit(df['Pclass'])
          le.transform(df['Pclass'])
          df['le_Pclass'] = le.transform(df['Pclass'])

          le.fit(df['Sex'])
          le.transform(df['Sex'])
          df['le_Sex'] = le.transform(df['Sex'])

          le.fit(df['Embarked'])
          le.transform(df['Embarked'])
          df['le_Embarked'] = le.transform(df['Embarked'])
          df.head()
```

```
Out[116]:    PassengerId  Survived  Pclass  \
          0            1         0       3
          1            2         1       1
          2            3         1       3
          3            4         1       1
          4            5         0       3

                                                          Name     Sex       Age  \
          0                            Braund, Mr. Owen Harris    male  0.022949
          1  Cumings, Mrs. John Bradley (Florence Briggs Th...  female  0.039639
          2                             Heikkinen, Miss. Laina  female  0.027122
          3       Futrelle, Mrs. Jacques Heath (Lily May Peel)  female  0.036510
          4                           Allen, Mr. William Henry    male  0.036510

                SibSp  Parch            Ticket      Fare Cabin Embarked  le_Pclass  \
          0  0.027462    0.0         A/5 21171  0.004103   NaN        S          2
          1  0.027462    0.0          PC 17599  0.040344   C85        C          0
          2  0.000000    0.0  STON/O2. 3101282  0.004485   NaN        S          2
          3  0.027462    0.0            113803  0.030053  C123        S          0
          4  0.000000    0.0            373450  0.004556   NaN        S          2

             le_Sex  le_Embarked
          0       1            2
          1       0            0
          2       0            2
          3       0            2
          4       1            2
```

```
In [117]: from sklearn import svm
          test = pd.read_csv('/public/bmort/python/test.csv')


          train_x = df[['le_Pclass', 'le_Sex', 'Age', 'SibSp','Parch', 'Fare', 'le_Embarked']]
          train_y = df['Survived'].values

          svm = svm.SVC(kernel='linear')


          # split the data into training and test sets
          X_train, X_test, y_train, y_test = train_test_split(train_x, train_y, test_size=0.2)


          svm.fit(X_train,y_train)

Out[117]: SVC(kernel='linear')
```

```
In [118]: from sklearn.model_selection import KFold, cross_val_score
          kfold = KFold(n_splits = 5, shuffle = True)
          scores = cross_val_score(model, X_train, y_train, cv=kfold)
          scores

          print("Accuracy: %0.2f +/- %0.2f" % (scores.mean(), scores.std()))

Accuracy: 0.78 +/- 0.02
```

```
In [119]: test['Age'].isna().sum()
          test['Age'] = test['Age'].fillna(imputed_value)
```

```
y_pred = clf.predict(X_test)
print("0 = not survived, 1 = survived")
print(y_pred)
```

```
0 = not survived, 1 = survived
[0 1 1 0 0 1 1 0 1 0 0 0 0 1 0 1 0 1 0 0 0 0 1 1 0 0 1 0 0 1 0 0 0 0 0 1 1
 1 0 0 0 0 0 0 0 1 0 1 0 0 0 0 0 1 0 0 1 1 1 0 1 0 0 1 0 0 0 0 1 0 0 0 1 1
 0 1 0 0 0 0 0 0 0 0 1 0 1 0 1 1 1 0 1 0 0 0 1 0 1 0 1 0 0 0 0 0 0 0 0 0 0
 1 0 0 0 0 1 0 1 0 0 0 1 0 1 0 1 0 1 1 1 1 0 1 1 0 0 0 1 0 0 1 0 0 0 0 0 0
 1 1 0 1 0 0 0 1 0 0 0 1 0 0 0 1 1 0 1 1 1 0 1 0 0 0 1 1 0 1 1]
```