Programming Assignment – 2 Titanic

INTRODUCTION:

The given dataset is on Titanic (A British liner that sank in the North Atlantic Ocean). A model will be built and run to predict the Survival chances of the passengers on Board and in the end the same model will be run to predict my survival chances on this Ship.

ABOUT THE DATASET: (TITANIC3.XLS)

- Survived: Survived (1) or died (0)
- Pclass: Passenger's class
- Name: Passenger's name
- Sex: Passenger's sex
- Age: Passenger's age
- SibSp: Number of siblings/spouses aboard
- Parch: Number of parents/children aboard
- Ticket: Ticket number
- Fare: Fare (in British Pound)
- Cabin: Cabin
- Embarked: Port of embarkation
- Boat: Lifeboat (if survived)
- Body: Body number (if did not survive and body was recovered)
- Home.dest: Home/Destination

EXPLORATORY DATA ANALYSIS [EDA]:

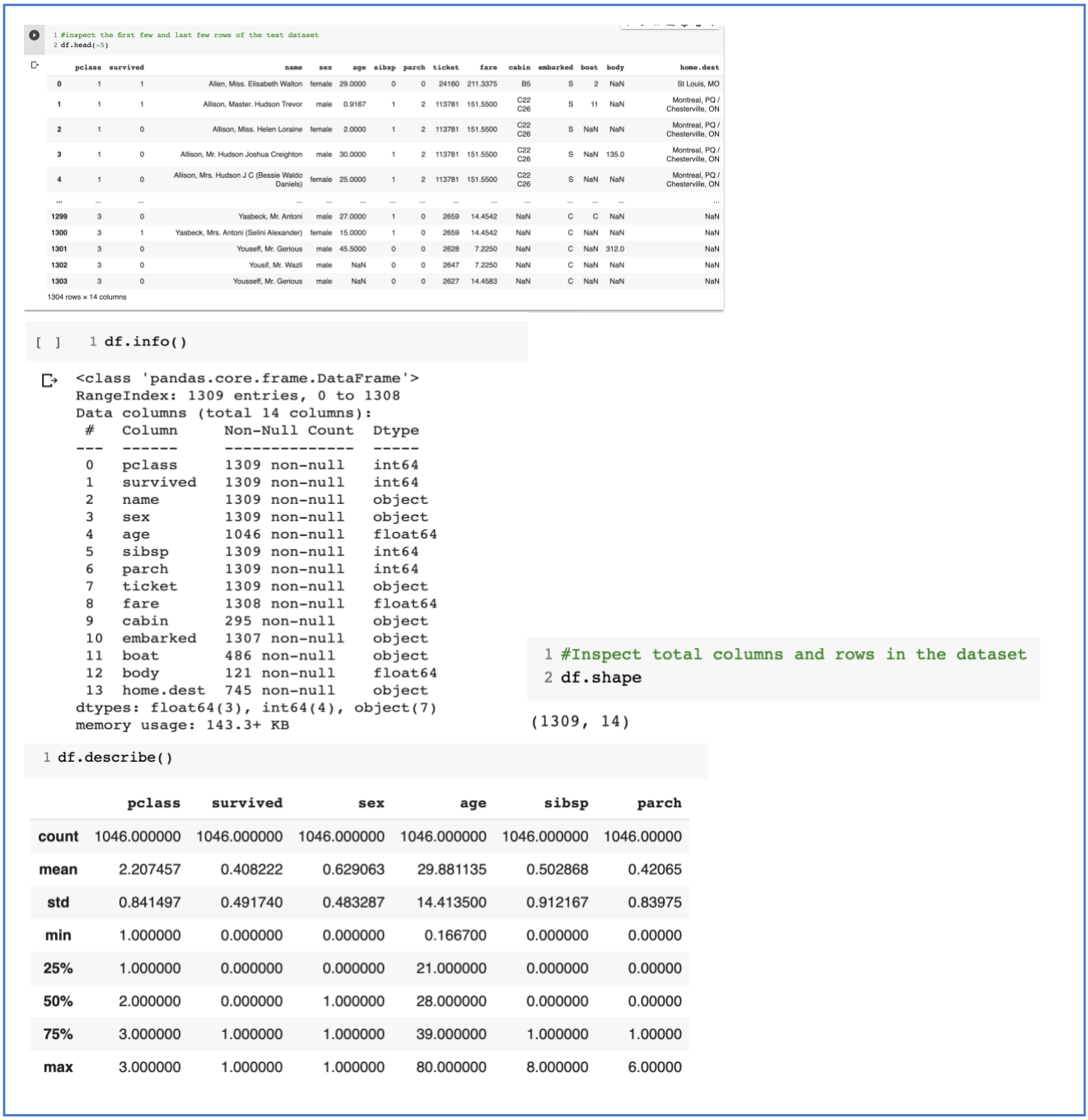
I have used the following codes to do exploratory analysis on the given dataset.

1. df.shape()
2. df.info()
3. df.describe()
4. df.head(-5)

Observation after running the above codes:

- The Titanic dataset has 1309 rows and 14 columns.

- Name, sex, ticket, cabin, embarked and boat have dtype as object and might need to be changed while creating models.
- Some statistical data about the Titanic dataset can be read with this command. For e.g.: The youngest person on-board was a 16 months old baby and the maximum age was 80 years. Average age of the Voyagers is 29.
- With this command we can inspect few top and last rows of the dataset.
- It can be noticed that few columns have null values like age, cabin, boat, body and Home.dest but we don't know yet how many rows have null values.



```
1 #inspect the first few and last few rows of the test dataset
2 df.head(-5)
```

	pclass	survived	name	sex	age	sibsp	parch	ticket	fare	cabin	embarked	boat	body	home.dest
0	1	1	Allen, Miss. Elisabeth Walton	female	29.0000	0	0	24160	211.3375	B5	S	2	NaN	St Louis, MO
1	1	1	Allison, Master. Hudson Trevor	male	0.9167	1	2	113781	151.5500	C22 C26	S	11	NaN	Montreal, PQ / Chesterville, ON
2	1	0	Allison, Miss. Helen Loraine	female	2.0000	1	2	113781	151.5500	C22 C26	S	NaN	NaN	Montreal, PQ / Chesterville, ON
3	1	0	Allison, Mr. Hudson Joshua Creighton	male	30.0000	1	2	113781	151.5500	C22 C26	S	NaN	135.0	Montreal, PQ / Chesterville, ON
4	1	0	Allison, Mrs. Hudson J C (Bessie Waldo Daniels)	female	25.0000	1	2	113781	151.5500	C22 C26	S	NaN	NaN	Montreal, PQ / Chesterville, ON
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
1299	3	0	Yasbeck, Mr. Antoni	male	27.0000	1	0	2659	14.4542	NaN	C	C	NaN	NaN
1300	3	1	Yasbeck, Mrs. Antoni (Selini Alexander)	female	15.0000	1	0	2659	14.4542	NaN	C	NaN	NaN	NaN
1301	3	0	Youseff, Mr. Gerious	male	45.5000	0	0	2628	7.2250	NaN	C	NaN	312.0	NaN
1302	3	0	Yousif, Mr. Wazli	male	NaN	0	0	2647	7.2250	NaN	C	NaN	NaN	NaN
1303	3	0	Yousseff, Mr. Gerious	male	NaN	0	0	2627	14.4583	NaN	C	NaN	NaN	NaN

1304 rows × 14 columns

```
1 df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1309 entries, 0 to 1308
Data columns (total 14 columns):
 #   Column     Non-Null Count  Dtype
---  ------     --------------  -----
 0   pclass     1309 non-null   int64
 1   survived   1309 non-null   int64
 2   name       1309 non-null   object
 3   sex        1309 non-null   object
 4   age        1046 non-null   float64
 5   sibsp      1309 non-null   int64
 6   parch      1309 non-null   int64
 7   ticket     1309 non-null   object
 8   fare       1308 non-null   float64
 9   cabin      295 non-null    object
 10  embarked   1307 non-null   object
 11  boat       486 non-null    object
 12  body       121 non-null    float64
 13  home.dest  745 non-null    object
dtypes: float64(3), int64(4), object(7)
memory usage: 143.3+ KB
```

```
1 #Inspect total columns and rows in the dataset
2 df.shape
```

```
(1309, 14)
```

```
1 df.describe()
```

	pclass	survived	sex	age	sibsp	parch
count	1046.000000	1046.000000	1046.000000	1046.000000	1046.000000	1046.00000
mean	2.207457	0.408222	0.629063	29.881135	0.502868	0.42065
std	0.841497	0.491740	0.483287	14.413500	0.912167	0.83975
min	1.000000	0.000000	0.000000	0.166700	0.000000	0.00000
25%	1.000000	0.000000	0.000000	21.000000	0.000000	0.00000
50%	2.000000	0.000000	1.000000	28.000000	0.000000	0.00000
75%	3.000000	1.000000	1.000000	39.000000	1.000000	1.00000
max	3.000000	1.000000	1.000000	80.000000	8.000000	6.00000

Screenshots of the Output upon running the above codes

5. Create datadict to identify datatype, missing values, unique values and count.

```
# identify datatypes of the 14 columns and also name that columns as DataType
datadict = pd.DataFrame(df.dtypes)
datadict
datadict = datadict.rename(columns={0:'DataType'})
datadict

# identify missing values of the 14 columns
datadict['MissingVal'] = df.isnull().sum()
datadict

# Identify number of unique values
datadict['NUnique']=df.nunique()
datadict

# Identify the count for each variable, add the stats to datadict
datadict['Count']=df.count()
datadict
```

	DataType	MissingVal	NUnique	Count
pclass	int64	0	3	1309
survived	int64	0	2	1309
name	object	0	1307	1309
sex	object	0	2	1309
age	float64	263	98	1046
sibsp	int64	0	7	1309
parch	int64	0	8	1309
ticket	object	0	939	1309
fare	float64	1	281	1308
cabin	object	1014	186	295
embarked	object	2	3	1307
boat	object	823	28	486
body	float64	1188	121	121
home.dest	object	564	369	745

Observation:

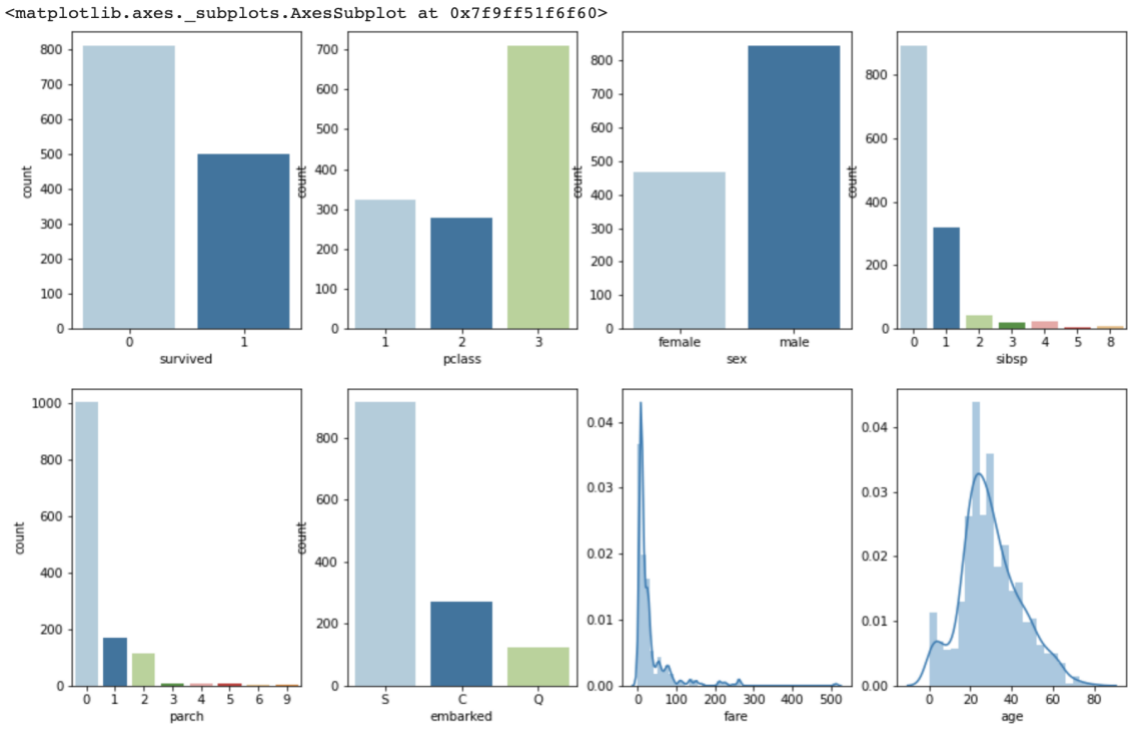
- Int64, object and float are the 3 types of datatype.
- Cabin, boat and body has a large amount of missing values ranging from 263 to 1014 hence these need to be removed from the study.

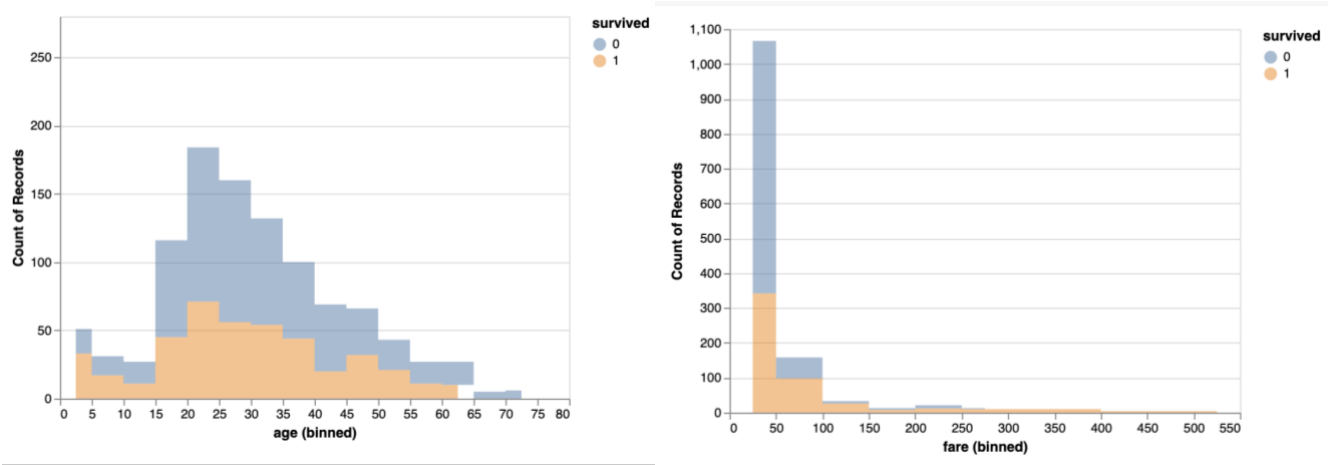
6. Graphical Data Analysis

I have used seaborn and Altair library to do the following visualization charts for Exploratory Data Analysis.

Observations based on the below visualization graphs:

- Total of 800(61%) died and 500(38%) survived the sinking Ship.
- Highest number of people on board belonged to pclass=3.
- Male to Female ratio is 800:450
- Record number of Voyagers embarked from Southampton.
- A great number of Voyagers onboard paid the fare between 0 to 100 Pounds and very few paid the highest fare of 500 pounds.
- Most of the passengers were in the age group of 20-30 and Highest number of deaths fall into the age bracket of 20-25 years.
- More number of females survived than the males, the evacuation policy of females and kids first can be the reason behind this.
- Passengers belonging to class 3 have the highest chances of not surviving the mishap as opposed to class 1 and 2.
- Higher the fare paid higher the chances of survival.





Graphical Exploratory Data Analysis

DATA PRE-PROCESSING:

Data preprocessing is an important step to prepare the data to form a model. We will do data cleaning, data transformation, and feature selection which will help to remove outliers and standardize the data so that they take a form that can be easily used to create a model.

The following pre-processing has been done before splitting the data for modeling:

- Dropped the columns name, embarked, ticket, fare, cabin, body, boat and home.dest as they are not needed for the modeling.
- Filled the missing values in Age column and dropped all the rest rows with the missing values.
- Convert the datatype of sex column from object to integer that is from ‘male’, ‘female’ to ‘0’ and ‘1’.
- Reset the index after dropping few columns.
- Lastly, split the dataset into train and test for Data Modeling.

```
[ ]  1 df=df.reset_index(drop=True)
```

```
[ ]  1 split=train_test_split(df,test_size=0.2)
```

```
[ ]  1 split
```

```
[      pclass  survived  sex      age  sibsp  parch
228         1         1    0  40.0000      0      0
700         3         1    0   0.1667      1      2
491         2         0    1  36.0000      0      0
600         3         0    1  32.0000      1      0
598         3         1    0  13.0000      0      0
...       ...       ...  ...      ...    ...    ...
490         2         1    0  33.0000      0      2
256         1         1    0  39.0000      1      1
1032        3         0    1  18.0000      1      0
733         3         0    0   9.0000      2      2
527         2         0    1  23.0000      1      0

[836 rows x 6 columns],      pclass  survived  sex    age  sibsp  parch
770         3         0    1  11.00      0      0
489         2         1    0   8.00      1      1
664         3         1    0  22.00      0      0
791         3         0    1  17.00      1      0
863         3         0    1  24.00      1      0
..        ...       ...  ...    ...    ...    ...
374         2         0    1  26.00      0      0
603         3         1    0   0.75      2      1
669         3         0    1  27.00      0      0
104         1         1    0  60.00      1      4
339         2         1    0  31.00      1      1

[210 rows x 6 columns]]
```

```
1 #new df after dropping all the unwanted rows and columns and
2 #after converting sex column into '0' and '1'.
3 df
```

	pclass	survived	sex	age	sibsp	parch
0	1	1	0	29.0000	0	0
1	1	1	1	0.9167	1	2
2	1	0	0	2.0000	1	2
3	1	0	1	30.0000	1	2
4	1	0	0	25.0000	1	2
...	...	...	...	...	...	...
1301	3	0	1	45.5000	0	0
1304	3	0	0	14.5000	1	0
1306	3	0	1	26.5000	0	0
1307	3	0	1	27.0000	0	0
1308	3	0	1	29.0000	0	0

1046 rows × 6 columns

```
1 df= df.drop(columns=['name','embarked','ticket','fare','home.dest','cabin','boat','body'])
```

```
1 df = df.fillna(value=0.0)
```

```
1 df= df[df['age']!=0]
```

```
1 from sklearn.preprocessing import LabelEncoder
2 labelencoder= LabelEncoder()
3
4 df.iloc[:,2]=labelencoder.fit_transform(df.iloc[:,2].values)
```

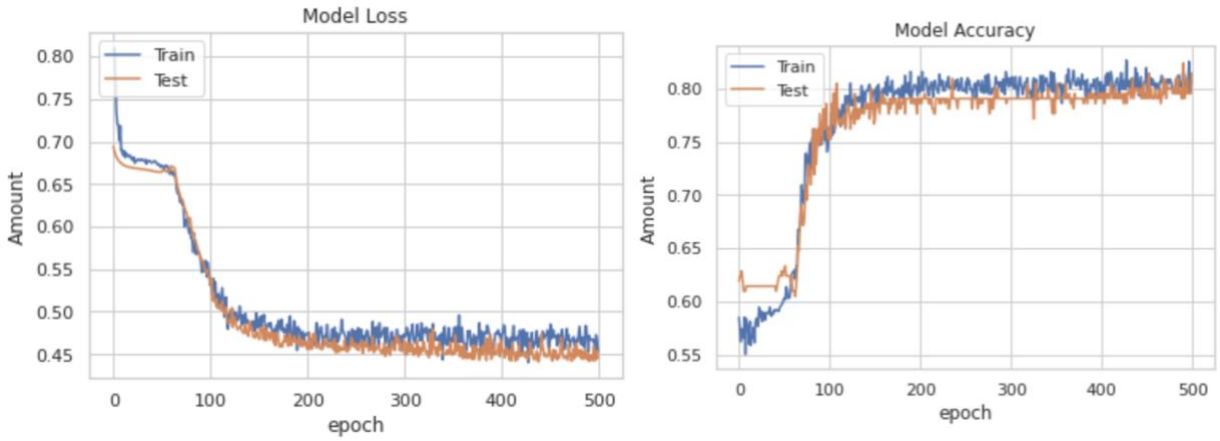
```
1 print(df['sex'].unique())
```

```
[0 1]
```

DATA MODELING:

3 Data Models have been created using TensorFlow and Python in the current assignment as mentioned below along with their accuracy score. Highlighted are the models that would have best predictions based on train and test data.

Models	Accuracy Score on Train Data	Accuracy Score on Test Data
Sequential Model (TensorFlow)	80.62%	81.90%
Decision Tree Model	97.97%	72.38%
KD Tree Model	-	60.95%



Below attached are some Screenshots of all the 3 Models:

```
def create_model( regress=True, output=1):
  x=Input(shape=(5,))
  i= x
  x= Flatten()(x)
  x= Activation("relu")(x)
  x= Dense(5)(x)
  x= Activation("relu")(x)
  x= Dropout(0.1)(x)
  x= Dense(2)(x)
  x= Activation("relu")(x)
  x= Dropout(0.1)(x)
  #x= Dense(3)(x)
  #x= Activation("relu")(x)
  #x= Dropout(0.1)(x)
  x= Dense(1,activation='sigmoid')(x)
  model=Model(i,x)
  return model
```

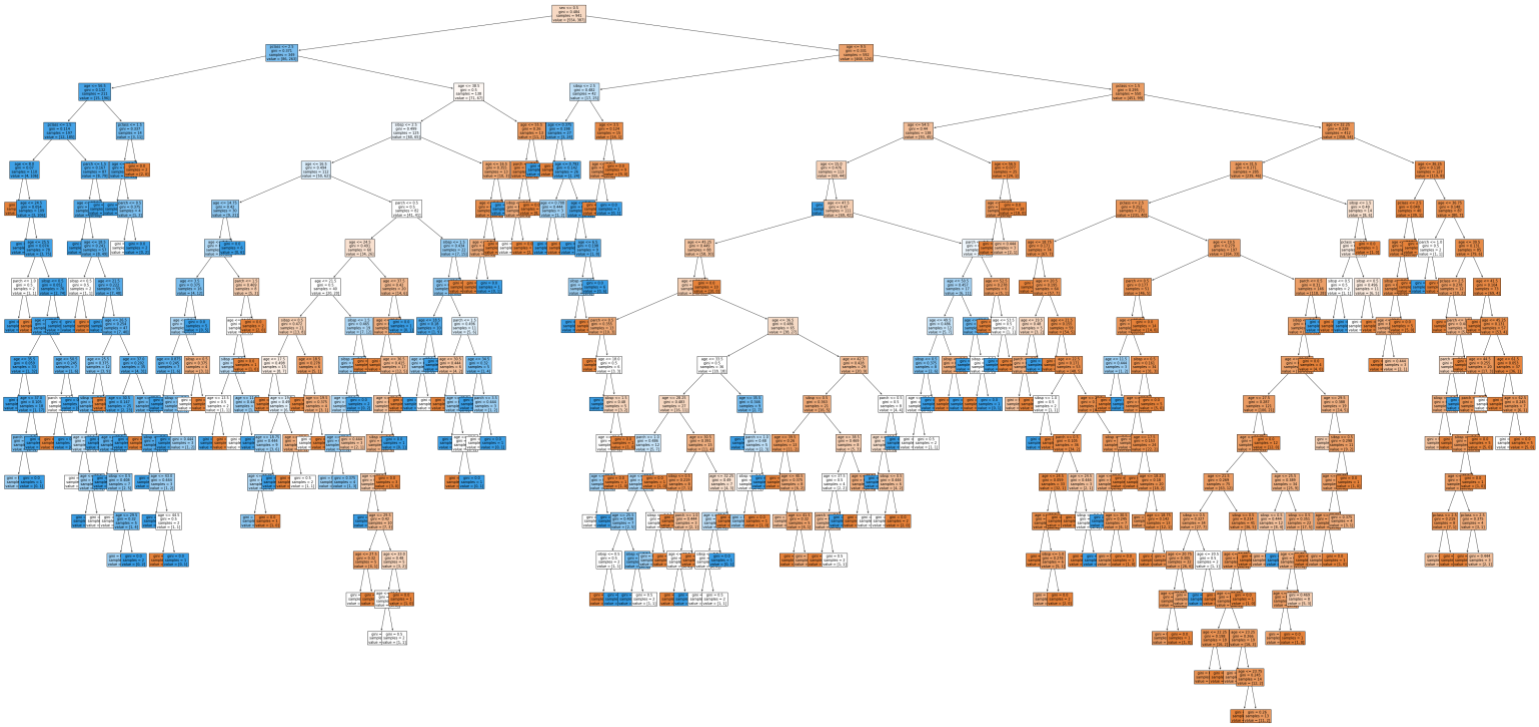
```
model = create_model()
#opt = Adam(learning_rate=0.001, epsilon = 1e-7)
model.compile(loss='binary_crossentropy', optimizer='adam', metrics=['accuracy'])
train_history = model.fit(x=trainx,
                          y=trainy,
                          epochs=500,
                          validation_data=(testx, testy), verbose=1,batch_size=20

                          )
```

Epoch 100: Accuracy:61.43%
Epoch 200: Accuracy: 76.19%
Epoch 300: Accuracy: 61.43%
Epoch 400: Accuracy: 78.10%
Epoch 500: Accuracy: 79.52%

This is how the accuracy of the model kept on changing and then getting stable with the changing range of Epochs.

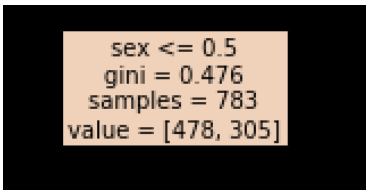
The same can be seen graphically in the above attached graphs on Model Loss and Model Accuracy.

PS: Model will give different predictions on re-running multiple times.



Interpretations from the Decision Tree Model:

- From the root node, it can be noted that gender is the most important factor in predicting the survival of passengers on Titanic.
- Passengers with less than 2.5 siblings had higher chance of surviving and so did those children who had both parents onboard.
- Children below the age of 3.5 died.




```
from scipy.spatial import KDTree
```

```
x= X_train.to_numpy()

X_tree= KDTree(x)

x1= X_test.to_numpy()
```

```
from scipy.stats import mode
kn=X_tree.query(x1,11)
Pre_label= np.empty(len(x1))
for K,N in enumerate(kn[1]):
    N_labels= y.loc[N]
    Pre_label[K]=mode(N_labels)[0][0]


```

```
y1=y_test.to_numpy()
```

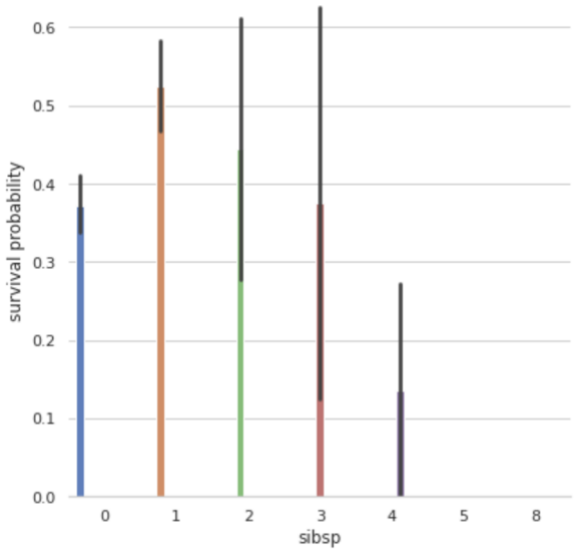
```
accuracy=np.array(y1==Pre_label).astype(int).sum()/len(y1)*100
```

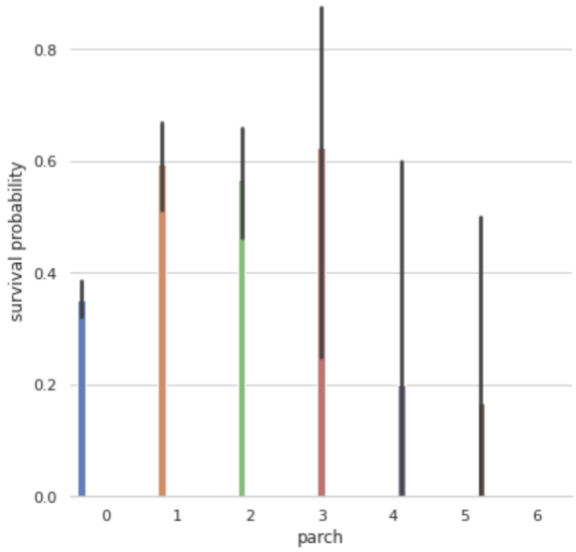
```
accuracy
```

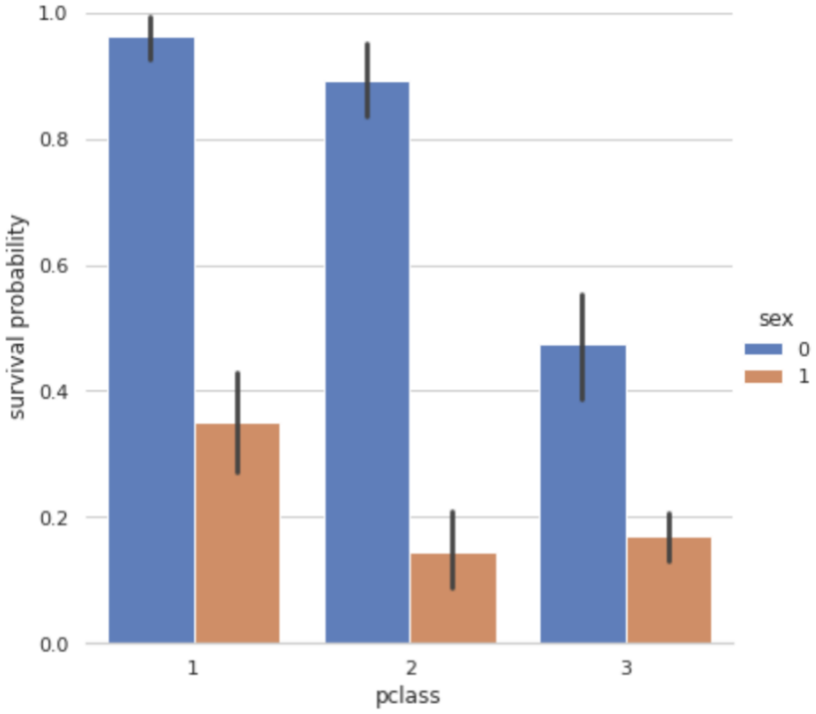
```
60.952380952380956
```

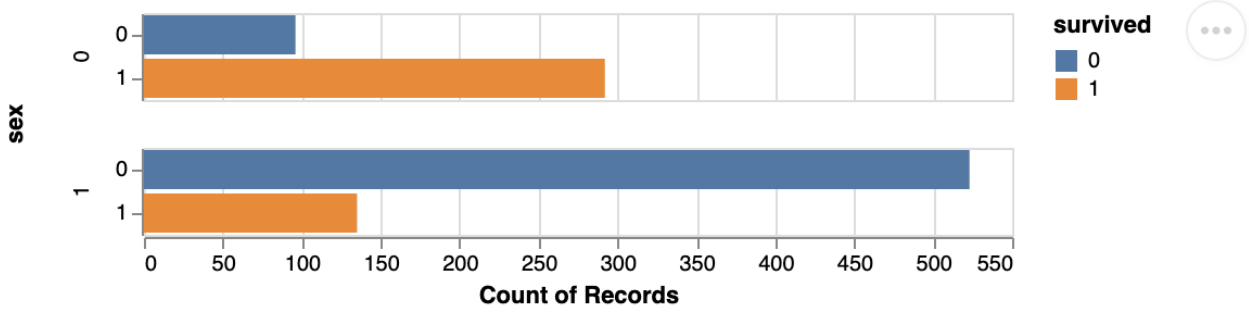
KD TREE MODEL

VISUALIZATIONS WITH THE FINALIZED DATA:









MY CHANCES OF SURVIVAL ON THE TITANIC-

I will predict my chances of Survival based on the following values in each attribute:

Name: Priya

Sex: Female '0'

Pclass: 2nd (Based on the Assumption that 2nd class would cost no more than $2500)

Age: 29

Sibsp: 1 (I would onboard it with my Husband)

Parch: 0 (I wouldn't be traveling along with any of my parent and don't have any child)

Output from the TensorFlow model:

Survival Rate: 86.98%

```
survive=model.predict(jack_rose_priya_df)
sdf = jack_rose_priya_df
sdf['survived']=survive
sdf['Name']=['Jack','Rose','Priya']
print(sdf)
```

```
   pclass  sex  age  sibsp  parch  survived   Name
0       3    1   23      0      0  0.172574   Jack
1       1    0   20      1      1  0.979631   Rose
2       2    0   29      1      0  0.869891  Priya
```

Output from the KD Tree model:

Survival Output: Survived

```
Jack = pd.Series([3,1,23,0,0])
Rose = pd.Series([1, 0, 20, 1, 1])
Priya = pd.Series( [2, 0, 29, 1, 0] )
```

```
jack_rose_priya_df = pd.DataFrame( [list(Jack),list(Rose),list(Priya)], columns=['pclass', 'sex', 'age', 'sibsp',
```

```
Name=('Jack','Rose','Priya')

result=[]
for i in model.predict(jack_rose_priya_df):
    if i:
        result.append('Survived')
    else:
        result.append('Dead')
print(Name)
print(result)
```

```
('Jack', 'Rose', 'Priya')
['Dead', 'Survived', 'Survived']
```

CONCLUSION:

Based on the observations done in Exploratory Analysis above as well as upon entering different values for each attribute and getting the resulting output by the models the below mentioned factors can be concluded to be playing a vital role in the survival chances on the Titanic:

Factors that increase the chances of survival:

- **Gender** plays an important role in survival → Sex: Female (it can be said that Women and children were more likely to survive because of the policy that favored them.)
- **Social status** is also an important distinguisher → Pclass: 1
- **Age** is another significant attribute to consider → (Young women and children had greater chances, again owing to the mentioned policy of women and children first on the life boat.)
- You have a better chance of survival if you have more family members aboard especially children → **Parch** >=1
- The vice-versa represents the combination of factors that decrease the chances of survival, like gender is male, social status is low that is pclass=3 and age is high and status is single).
- Thus, we can say that Gender, Socio- economic factor, Age and whether you are alone or with the family all play a deciding role into the survival rate on the Titanic.

Best Combination of Attributes:

- sex=female, pclass=1, age=20
- sex=female, pclass=2, age=20, sibsp=1, parch=1

(Attachments: 1 PDF file, 2-ipynb files, 1 png file)

REFERENCE:

https://www.google.co.in/search?ei=8oi5Xt3XOsGsytMPodOT2Ao&q=titanic+machine+learning&oq=titanic+machine+learning&gs_lcp=CgZwc3ktYWIQAzICCAAyAggAMgIIADICCAAyAggAMgIIADICCAAyAggAMgIIADICCAA6BAgAEEc6BwgAEIMBEEM6BAgAEEM6BQgAEIMBOgUIABCRAlC3DljSUGD0UWgAcAF4AIABoQGIAaAIkgEEMTYuMZgBAKABAaoBB2d3cy13aXo&sclient=psy-ab&ved=0ahUKEwjdmLXPqKzpAhVBlnIEHaHpBKsQ4dUDCAw&uact=5
https://towardsdatascience.com/kaggle-titanic-machine-learning-model-top-7-fa4523b7c40
https://towardsdatascience.com/predicting-the-survival-of-titanic-passengers-30870ccc7e8
https://www.kaggle.com/vinothan/titanic-model-with-90-accuracy
https://rstudio-pubs-static.s3.amazonaws.com/512981_db3552e79dc74dd0ada230125c27a4f3.html#preprocess1-link
https://medium.com/@mjamilmoughal786/exploratory-data-analysis-of-titanic-dataset-with-python-94b0c84cd108
https://towardsdatascience.com/predicting-the-survival-of-titanic-passengers-30870ccc7e8
https://www.youtube.com/watch?v=rODWw2_1mCI&t=5573s
https://www.kdnuggets.com/2016/07/titanic-machine-learning-guide-part-2.html/2
https://www.kaggle.com/lisastuart/titanic-python-decision-trees