

```
In [21]: print('The features Matrix Has %d Rows and %d Columns(s)'%(features_matrix.shape[0], features_matrix.shape[1]))
print('The Target Matrix has %d Rows and %d Columns(s)'%(np.array(target_vector).shape[0], np.array(target_vector).shape[1]))
```

The features Matrix Has 418 Rows and 2 Columns(s)
The Target Matrix has 418 Rows and 1 Columns(s)

```
In [23]: features_matrix_standardized=StandardScaler().fit_transform(features_matrix)
```

```
In [29]: algorithm=LogisticRegression(penalty='l2', dual=False, tol=1e-4, C=1.0, fit_intercept=True)
```

```
In [30]: Logistic_Regression_Model=algorithm.fit(features_matrix_standardized, target_vector)
```

```
In [35]: observation=[[0.4,1]]
```

```
In [36]: predictions = Logistic_Regression_Model.predict(observation)
print('The Model predicted The observation To Belong To class %s'%(predictions[0]))
```

The Model predicted The observation To Belong To class [1]

```
In [37]: print('The Algorithm Was Trained To predict one of the two classes:%s'%(algorithm.classes_[0]))
```

The Algorithm Was Trained To predict one of the two classes:[0 1]

```
In [45]: print("""The Model says The probability of the observation We passed Belonging To class %s Is %s""")
print()
print("""The Model says The probability of the observation We passed Belonging To class %s Is %s""")
```

The Model says The probability of the observation We passed Belonging To class [0] Is 0.0549611529834666

The Model says The probability of the observation We passed Belonging To class [1] Is 0.0549611529834666

Data Analysis

To Predict and Analyse which gender has a High change of survival at the time of disaster.

Import datasets,python packages and libraries

```
In [1]: import numpy as np
import pandas as pd
from sklearn import preprocessing
import matplotlib.pyplot as plt
# plt.rc("font",size=14)
import seaborn as sns
sns.set(style="white") # white background style for seaborn plots.
sns.set(style="whitegrid",color_codes=True)
import warnings
warnings.simplefilter(action='ignore')
```

```
In [2]: train_df=pd.read_csv(r"C:\Users\91756\Documents\python\train.gender_submission
train_df
```

Out[2]:

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare
0	1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.2500
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	female	38.0	1	0	PC 17599	71.2833
2	3	1	3	Heikkinen, Miss. Laina	female	26.0	0	0	STON/O2. 3101282	7.9250
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	0	113803	53.1000
4	5	0	3	Allen, Mr. William Henry	male	35.0	0	0	373450	8.0500
...
886	887	0	2	Montvila, Rev. Juozas	male	27.0	0	0	211536	13.0000
887	888	1	1	Graham, Miss. Margaret Edith	female	19.0	0	0	112053	30.0000
888	889	0	3	Johnston, Miss. Catherine Helen "Carrie"	female	NaN	1	2	W./C. 6607	23.4500
889	890	1	1	Behr, Mr. Karl Howell	male	26.0	0	0	111369	30.0000
890	891	0	3	Dooley, Mr. Patrick	male	32.0	0	0	370376	7.7500

891 rows × 12 columns



```
In [3]: test_df=pd.read_csv(r"C:\Users\91756\Documents\python\test.gender_submission.csv")
test_df
```

Out[3]:

	PassengerId	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin
0	892	3	Kelly, Mr. James	male	34.5	0	0	330911	7.8292	NaN
1	893	3	Wilkes, Mrs. James (Ellen Needs)	female	47.0	1	0	363272	7.0000	NaN
2	894	2	Myles, Mr. Thomas Francis	male	62.0	0	0	240276	9.6875	NaN
3	895	3	Wirz, Mr. Albert	male	27.0	0	0	315154	8.6625	NaN
4	896	3	Hirvonen, Mrs. Alexander (Helga E Lindqvist)	female	22.0	1	1	3101298	12.2875	NaN
...
413	1305	3	Spector, Mr. Woolf	male	NaN	0	0	A.5. 3236	8.0500	NaN
414	1306	1	Oliva y Ocana, Dona. Fermina	female	39.0	0	0	PC 17758	108.9000	C105
415	1307	3	Saether, Mr. Simon Sivertsen	male	38.5	0	0	SOTON/O.Q. 3101262	7.2500	NaN
416	1308	3	Ware, Mr. Frederick	male	NaN	0	0	359309	8.0500	NaN
417	1309	3	Peter, Master. Michael J	male	NaN	1	1	2668	22.3583	NaN

418 rows × 11 columns

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

Out[4]:

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	C.
0	1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.2500	
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	female	38.0	1	0	PC 17599	71.2833	
2	3	1	3	Heikkinen, Miss. Laina	female	26.0	0	0	STON/O2. 3101282	7.9250	
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	0	113803	53.1000	C
4	5	0	3	Allen, Mr. William Henry	male	35.0	0	0	373450	8.0500	



In [5]: `train_df.shape`

Out[5]: (891, 12)

```
In [6]: test_df.head()
```

```
Out[6]:
```

	PassengerId	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embar
0	892	3	Kelly, Mr. James	male	34.5	0	0	330911	7.8292	NaN	
1	893	3	Wilkes, Mrs. James (Ellen Needs)	female	47.0	1	0	363272	7.0000	NaN	
2	894	2	Myles, Mr. Thomas Francis	male	62.0	0	0	240276	9.6875	NaN	
3	895	3	Wirz, Mr. Albert	male	27.0	0	0	315154	8.6625	NaN	
4	896	3	Hirvonen, Mrs. Alexander (Helga E Lindqvist)	female	22.0	1	1	3101298	12.2875	NaN	



```
In [7]: test_df.shape
```

```
Out[7]: (418, 11)
```

```
In [8]: train_df.describe
```

```
Out[8]: <bound method NDFrame.describe of      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 [9]: `train_df.info()`

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 891 entries, 0 to 890
Data columns (total 12 columns):
 #   Column          Non-Null Count  Dtype
---  -
 0   PassengerId     891 non-null   int64
 1   Survived        891 non-null   int64
 2   Pclass         891 non-null   int64
 3   Name            891 non-null   object
 4   Sex             891 non-null   object
 5   Age            714 non-null   float64
 6   SibSp          891 non-null   int64
 7   Parch          891 non-null   int64
 8   Ticket         891 non-null   object
 9   Fare           891 non-null   float64
10   Cabin          204 non-null   object
11   Embarked       889 non-null   object
dtypes: float64(2), int64(5), object(5)
memory usage: 83.7+ KB
```

In [10]: `test_df.describe`

```
Out[10]: <bound method NDFrame.describe of      PassengerId  Pclass
Name
0           892      3                Kelly, Mr. James \
1           893      3          Wilkes, Mrs. James (Ellen Needs)
2           894      2          Myles, Mr. Thomas Francis
3           895      3                Wirz, Mr. Albert
4           896      3  Hirvonen, Mrs. Alexander (Helga E Lindqvist)
..          ...      ...
413          1305      3                Spector, Mr. Woolf
414          1306      1          Oliva y Ocana, Dona. Fermina
415          1307      3          Saether, Mr. Simon Sivertsen
416          1308      3          Ware, Mr. Frederick
417          1309      3      Peter, Master. Michael J

      Sex  Age  SibSp  Parch      Ticket     Fare Cabin Embarked
0    male  34.5     0     0      330911     7.8292   NaN        Q
1  female  47.0     1     0      363272     7.0000   NaN        S
2    male  62.0     0     0      240276     9.6875   NaN        Q
3    male  27.0     0     0      315154     8.6625   NaN        S
4  female  22.0     1     1      3101298    12.2875   NaN        S
..     ...   ...     ...     ...      ...      ...   ...      ...
413  male   NaN     0     0      A.5. 3236     8.0500   NaN        S
414  female  39.0     0     0      PC 17758    108.9000  C105        C
415  male   38.5     0     0  SOTON/O.Q. 3101262     7.2500   NaN        S
416  male   NaN     0     0      359309     8.0500   NaN        S
417  male   NaN     1     1        2668     22.3583   NaN        C

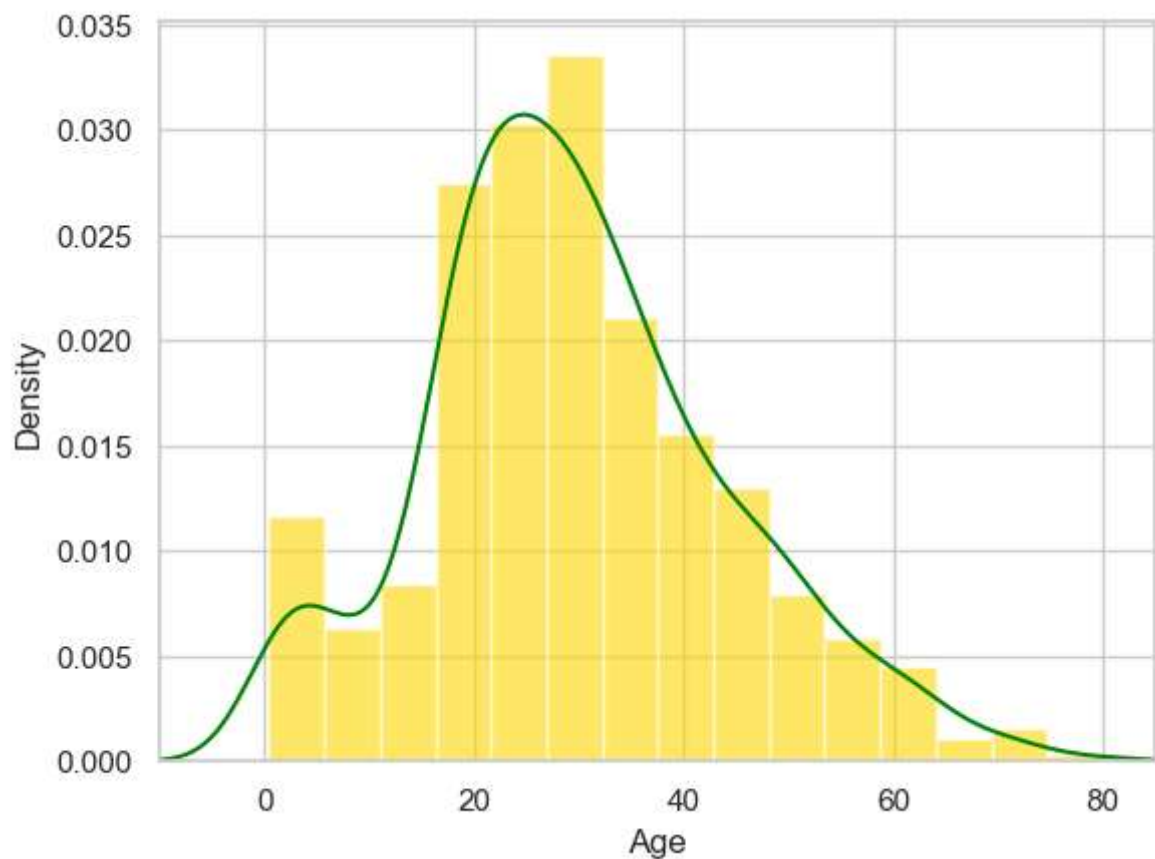
[418 rows x 11 columns]>
```



```
In [11]: test_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>  
RangeIndex: 418 entries, 0 to 417  
Data columns (total 11 columns):  
#   Column      Non-Null Count  Dtype    
---  -  
0   PassengerId  418 non-null    int64    
1   Pclass       418 non-null    int64    
2   Name         418 non-null    object    
3   Sex          418 non-null    object    
4   Age          332 non-null    float64   
5   SibSp        418 non-null    int64    
6   Parch        418 non-null    int64    
7   Ticket       418 non-null    object    
8   Fare         417 non-null    float64   
9   Cabin        91 non-null     object    
10  Embarked     418 non-null    object    
dtypes: float64(2), int64(4), object(5)  
memory usage: 36.0+ KB
```

```
In [14]: ax=train_df["Age"].hist(bins=15,density=True,stacked=True,color='gold',alpha=0.5)  
train_df["Age"].plot(kind='density',color='green')  
ax.set(xlabel='Age')  
plt.xlim(-10,85)  
plt.show()
```



```
In [16]: print(train_df["Age"].mean(skipna=True))  
print(train_df["Age"].median(skipna=True))
```

```
29.69911764705882  
28.0
```

```
In [17]: print((train_df['Cabin'].isnull().sum()/train_df.shape[0])*100)
```

```
77.10437710437711
```

```
In [18]: print((train_df['Embarked'].isnull().sum()/train_df.shape[0])*100)
```

```
0.22446689113355783
```

```
In [19]: print('Boarded passengers grouped by port of embarkation(c=Cherbourg,Q=Queenst  
print(train_df['Embarked'].value_counts())  
sns.countplot(x='Embarked',data=train_df,palette='Set2')  
plt.show()
```

Boarded passengers grouped by port of embarkation(c=Cherbourg,Q=Queenstown,s=Southampton):

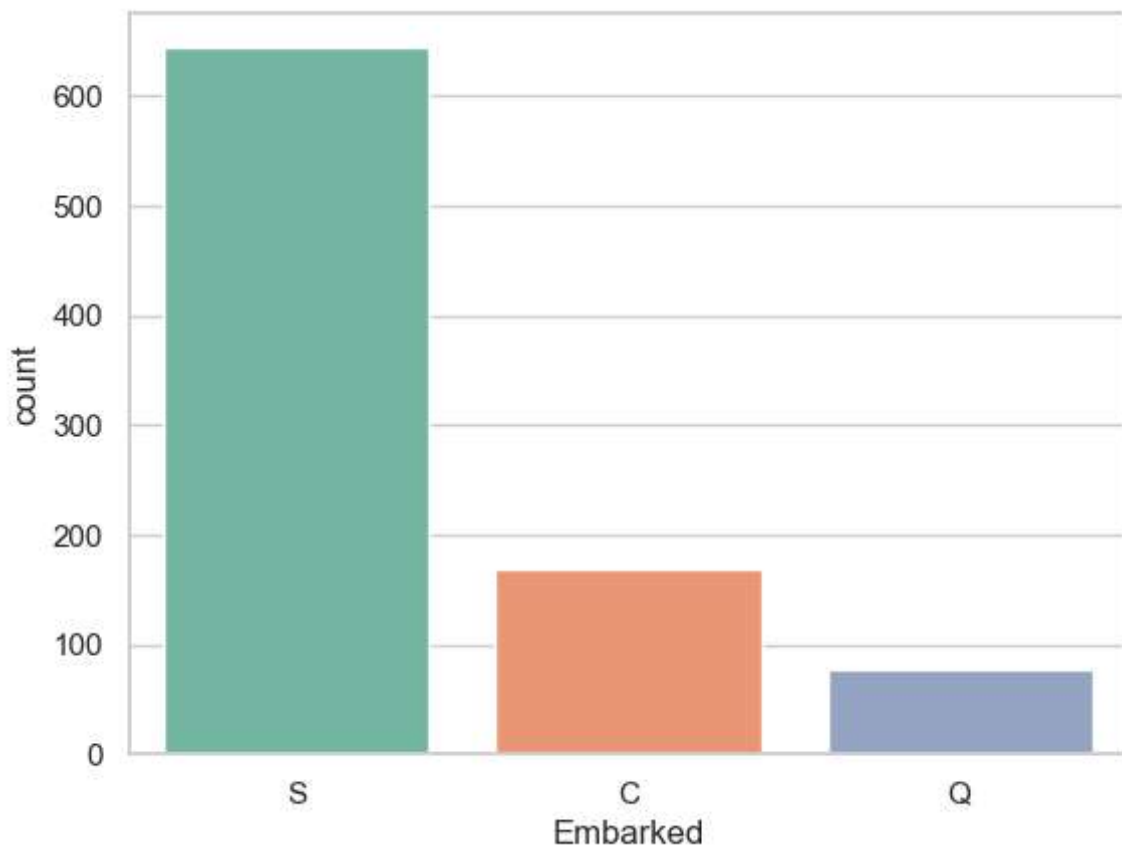
Embarked

S 644

C 168

Q 77

Name: count, dtype: int64



```
In [20]: print(train_df['Embarked'].value_counts().idxmax())
```

S

```
In [21]: train_data = train_df.copy()
```

```
In [22]: train_data['Age'].fillna(train_df["Age"].median(skipna=True),inplace=True)
train_data["Embarked"].fillna(train_df['Embarked'].value_counts().idxmax(),inplace=True)
train_data.drop('Cabin',axis=1,inplace=True)
```

```
In [23]: train_data.isnull().sum
```

```
Out[23]: <bound method NDFrame._add_numeric_operations.<locals>.sum of Passenger
Id Survived Pclass Name Sex Age SibSp Parch Ticket
0 False False False False False False False False False False
1 False False False False False False False False False False
2 False False False False False False False False False False
3 False False False False False False False False False False
4 False False False False False False False False False False
.. ... ..
886 False False False False False False False False False False
887 False False False False False False False False False False
888 False False False False False False False False False False
889 False False False False False False False False False False
890 False False False False False False False False False False

Fare Embarked
0 False False
1 False False
2 False False
3 False False
4 False False
.. ...
886 False False
887 False False
888 False False
889 False False
890 False False

[891 rows x 11 columns]>
```

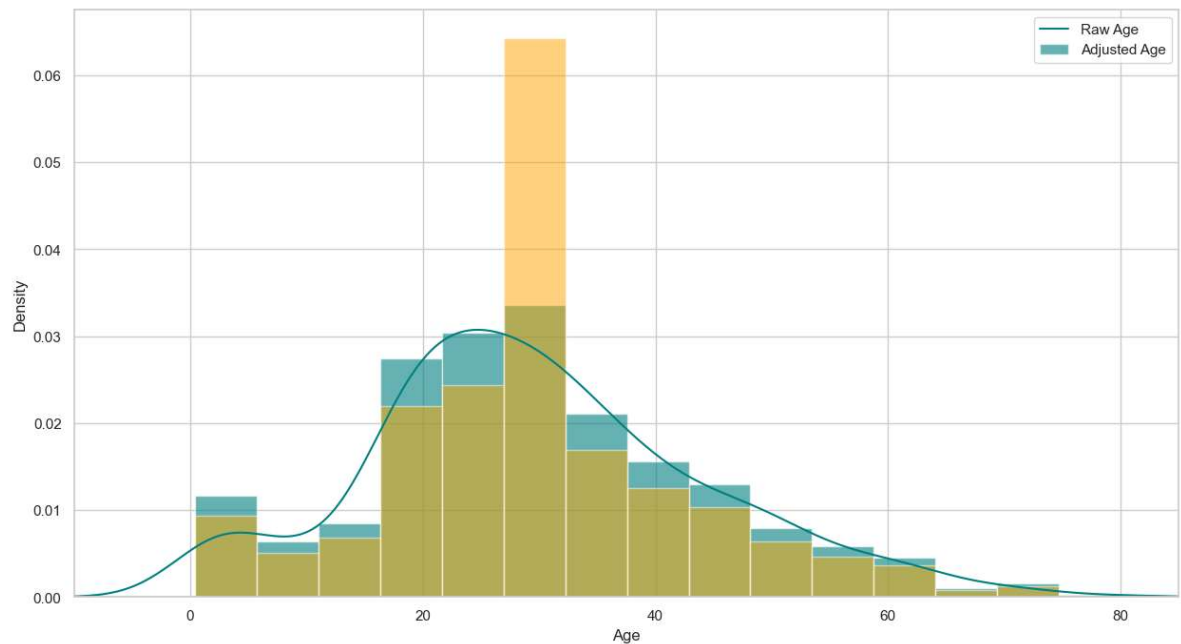
```
In [24]: train_data.head()
```

Out[24]:

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Embarked
0	1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.2500	S
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...)	female	38.0	1	0	PC 17599	71.2833	C
2	3	1	3	Heikkinen, Miss. Laina	female	26.0	0	0	STON/O2. 3101282	7.9250	S
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	0	113803	53.1000	C
4	5	0	3	Allen, Mr. William Henry	male	35.0	0	0	373450	8.0500	S

```
In [27]: plt.figure(figsize= (15,8))
ax = train_df["Age"].hist(bins=15,density=True,stacked=True,color='teal',alpha=0.5)
train_df["Age"].plot(kind='density',color='teal')
ax=train_data["Age"].hist(bins=15,density=True,stacked=True,color='orange',alpha=0.5)
ax.legend(['Raw Age', 'Adjusted Age'])
ax.set(xlabel='Age')
plt.xlim(-10,85)
```

Out[27]: (-10.0, 85.0)



```
In [33]: train_data['TravelAlone']=np.where((train_data["SibSp"]+train_data["Parch"])>0,1,0)
train_data.drop('SibSp',axis=1,inplace=True)
train_data.drop('Parch',axis=1,inplace=True)
```

```
In [38]: training=pd.get_dummies(train_data,columns=["Pclass","Embarked","Sex"])
training.drop('Sex_female',axis=1,inplace=True)
training.drop('PassengerId',axis=1,inplace=True)
training.drop('Name',axis=1,inplace=True)
training.drop('Ticket',axis=1,inplace=True)
final_train=training
final_train.head()
```

Out[38]:

	Survived	Age	Fare	TravelAlone	Pclass_1	Pclass_2	Pclass_3	Embarked_C	Embarked_S
0	0	22.0	7.2500	0	False	False	True	False	False
1	1	38.0	71.2833	0	True	False	False	True	False
2	1	26.0	7.9250	1	False	False	True	False	False
3	1	35.0	53.1000	0	True	False	False	False	False
4	0	35.0	8.0500	1	False	False	True	False	False

```
In [39]: test_df.isnull().sum()
```

```
Out[39]: PassengerId      0
Pclass      0
Name        0
Sex         0
Age        86
SibSp       0
Parch       0
Ticket      0
Fare        1
Cabin     327
Embarked    0
dtype: int64
```

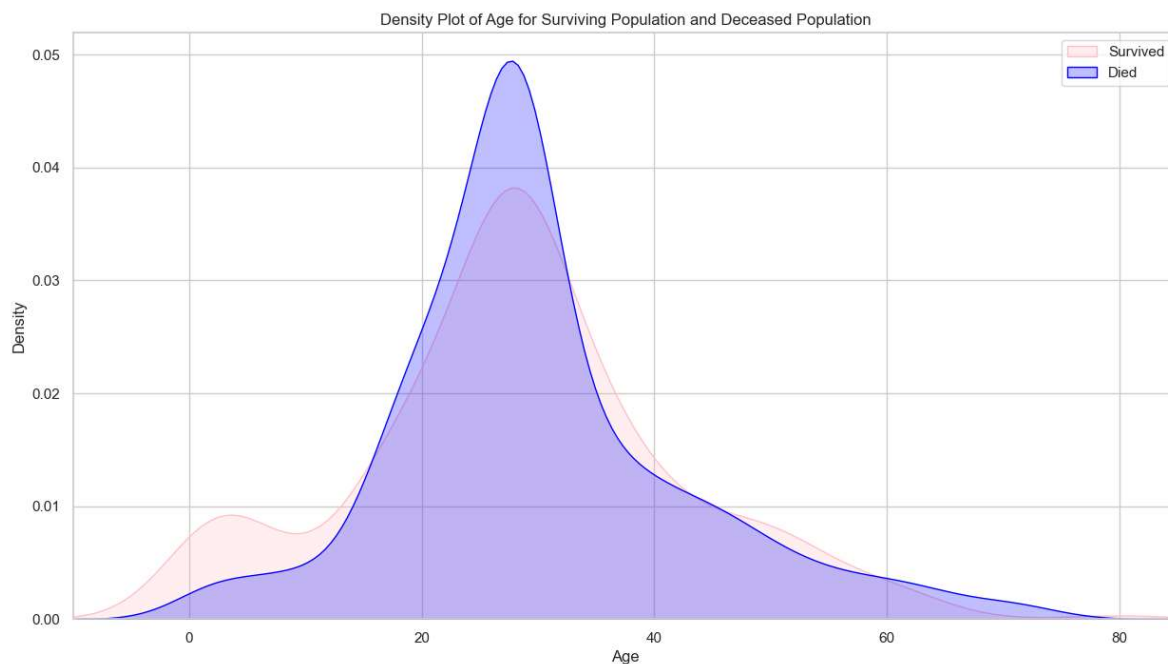
```
In [40]: test_data = test_df.copy()
test_data["Age"].fillna(train_df["Age"].median(skipna=True), inplace=True)
test_data["Fare"].fillna(train_df["Fare"].median(skipna=True), inplace=True)
test_data.drop('Cabin', axis=1, inplace=True)
test_data['TravelAlone'] = np.where((test_data["SibSp"] + test_data["Parch"]) > 0, 0, 1)
test_data.drop('SibSp', axis=1, inplace=True)
test_data.drop('Parch', axis=1, inplace=True)
testing = pd.get_dummies(test_data, columns=["Pclass", "Embarked", "Sex"])
testing.drop('Sex_female', axis=1, inplace=True)
testing.drop('PassengerId', axis=1, inplace=True)
testing.drop('Name', axis=1, inplace=True)
testing.drop('Ticket', axis=1, inplace=True)
final_test = testing
final_test.head()
```

```
Out[40]:
```

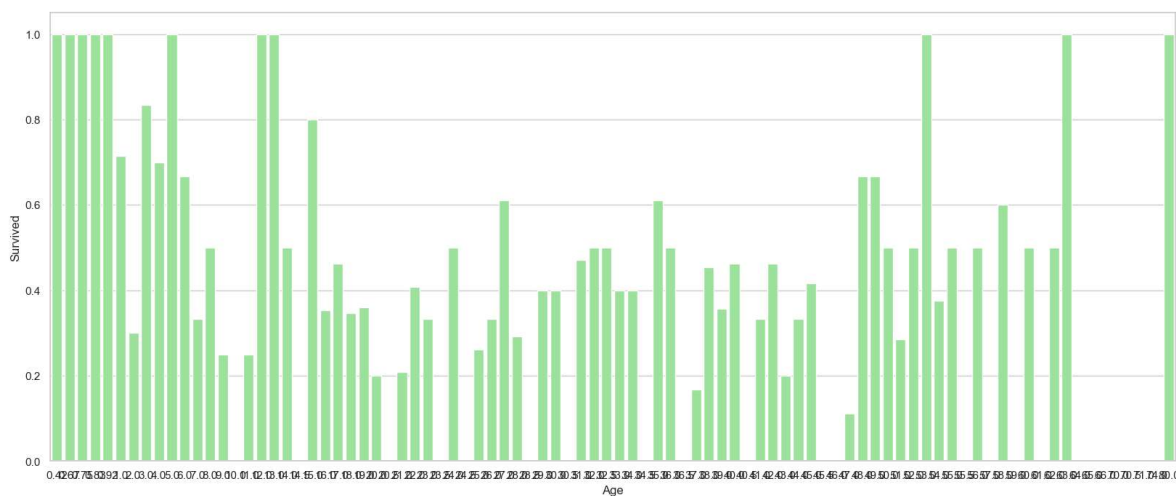
	Age	Fare	TravelAlone	Pclass_1	Pclass_2	Pclass_3	Embarked_C	Embarked_Q	Embarked_S
0	34.5	7.8292	1	False	False	True	False	True	
1	47.0	7.0000	0	False	False	True	False	False	
2	62.0	9.6875	1	False	True	False	False	True	
3	27.0	8.6625	1	False	False	True	False	False	
4	22.0	12.2875	0	False	False	True	False	False	

EXPLORATORY DATA ANALYSIS

```
In [58]: plt.figure(figsize=(15,8))
ax=sns.kdeplot(final_train["Age"][final_train.Survived ==1],color="pink",shade=True)
sns.kdeplot(final_train["Age"][final_train.Survived == 0], color="blue", shade=True)
plt.legend(['Survived', 'Died'])
plt.title('Density Plot of Age for Surviving Population and Deceased Population')
ax.set(xlabel='Age')
plt.xlim(-10,85)
plt.show()
```



```
In [53]: plt.figure(figsize=(20,8))
avg_survival_byage=final_train[["Age", "Survived"]].groupby(['Age'], as_index=False).mean()
g=sns.barplot(x='Age', y='Survived', data=avg_survival_byage, color="LightGreen")
plt.show()
```



```
In [48]: final_train['IsMinor']=np.where(final_train['Age']<=16, 1, 0)
print(final_train['IsMinor'])
```

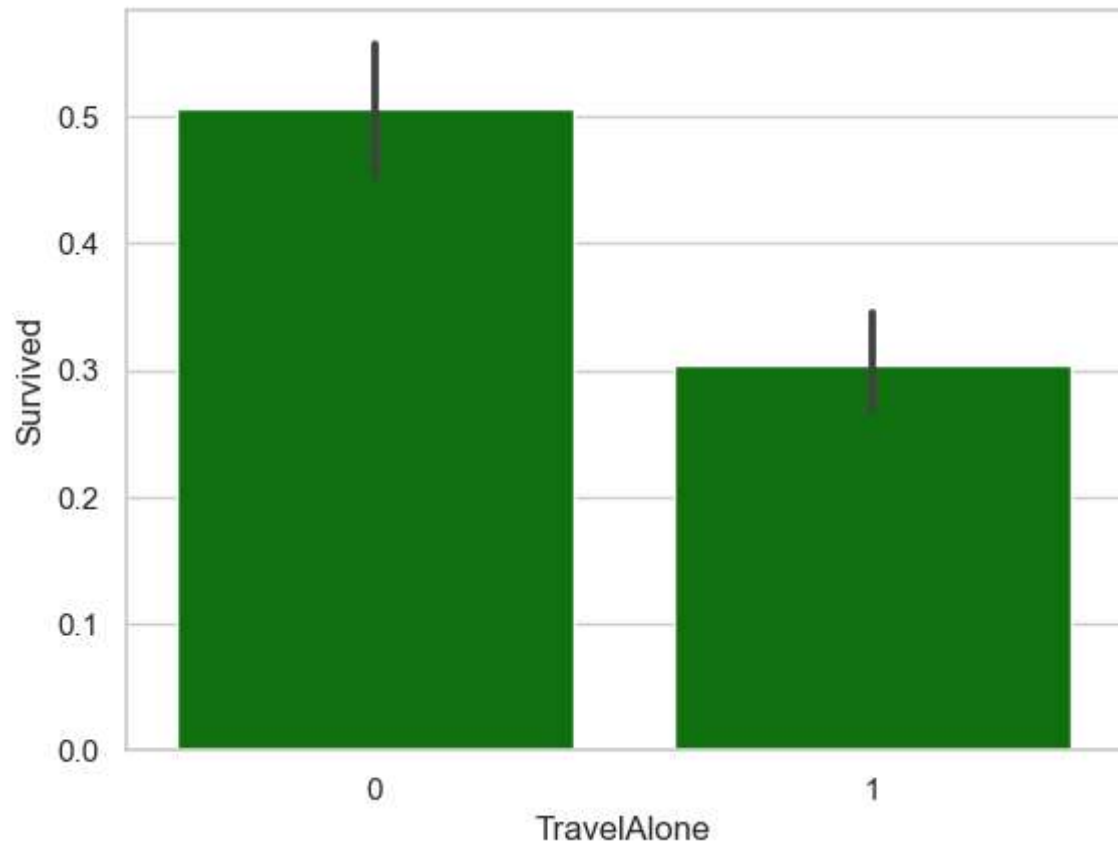
```
0      0
1      0
2      0
3      0
4      0
..
886    0
887    0
888    0
889    0
890    0
Name: IsMinor, Length: 891, dtype: int32
```

```
In [49]: final_test['IsMinor']=np.where(final_test['Age']<=16, 1, 0)
print(final_test['IsMinor'])
```

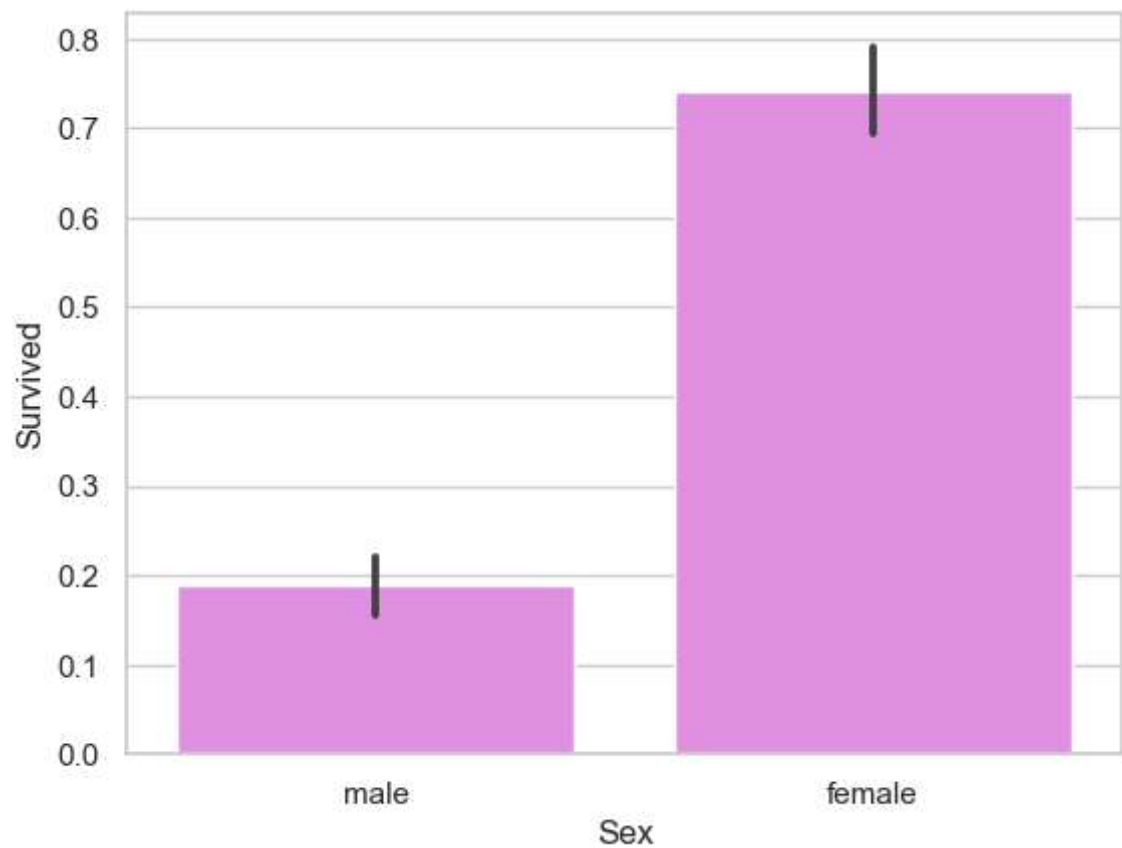
```
0      0
1      0
2      0
3      0
4      0
..
413    0
414    0
415    0
416    0
417    0
Name: IsMinor, Length: 418, dtype: int32
```



```
In [63]: sns.barplot(x='TravelAlone', y='Survived', data=final_train, color="Green")  
plt.show()
```



```
In [59]: import seaborn as sns
import matplotlib.pyplot as plt
sns.barplot(x='Sex', y='Survived', data=train_df, color='violet')
plt.show()
```



In []: