



## Read CSV File

```
df=pd.read_csv('/content/train.csv')
df.head()
```

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	
0	1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th	female	38.0	1	0	PC 17599	71.

## Remove/Drop Useless Data

```
df.drop(['PassengerId', 'Name', 'Ticket'], axis=1, inplace=True)
df.head()
```

	Survived	Pclass	Sex	Age	SibSp	Parch	Fare	Cabin	Embarked
0	0	3	male	22.0	1	0	7.2500	NaN	S
1	1	1	female	38.0	1	0	71.2833	C85	C
2	1	3	female	26.0	0	0	7.9250	NaN	S
3	1	1	female	35.0	1	0	53.1000	C123	S
4	0	3	male	35.0	0	0	8.0500	NaN	S

```
#data set Shape
df.shape
```

```
(891, 9)
```

```
#data set columns
df.columns
```

```
Index(['Survived', 'Pclass', 'Sex', 'Age', 'SibSp', 'Parch', 'Fare', 'Cabin',  
      'Embarked'],  
      dtype='object')
```

```
df.ndim
```

```
2
```

```
df.tail()
```

	Survived	Pclass	Sex	Age	SibSp	Parch	Fare	Cabin	Embarked
886	0	2	male	27.0	0	0	13.00	NaN	S
887	1	1	female	19.0	0	0	30.00	B42	S
888	0	3	female	NaN	1	2	23.45	NaN	S
889	1	1	male	26.0	0	0	30.00	C148	C
890	0	3	male	32.0	0	0	7.75	NaN	C

```
df.dtypes
```

```
Survived    int64
Pclass      int64
Sex         object
Age         float64
SibSp       int64
Parch       int64
Fare        float64
Cabin       object
Embarked    object
dtype: object
```

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 891 entries, 0 to 890
Data columns (total 9 columns):
#   Column      Non-Null Count  Dtype
---  -
0   Survived    891 non-null    int64
1   Pclass      891 non-null    int64
2   Sex         891 non-null    object
3   Age         714 non-null    float64
4   SibSp       891 non-null    int64
5   Parch       891 non-null    int64
6   Fare        891 non-null    float64
7   Cabin       204 non-null    object
8   Embarked    889 non-null    object
dtypes: float64(2), int64(4), object(3)
memory usage: 62.8+ KB
```

## Missing Value

```
import missingno as msno
```

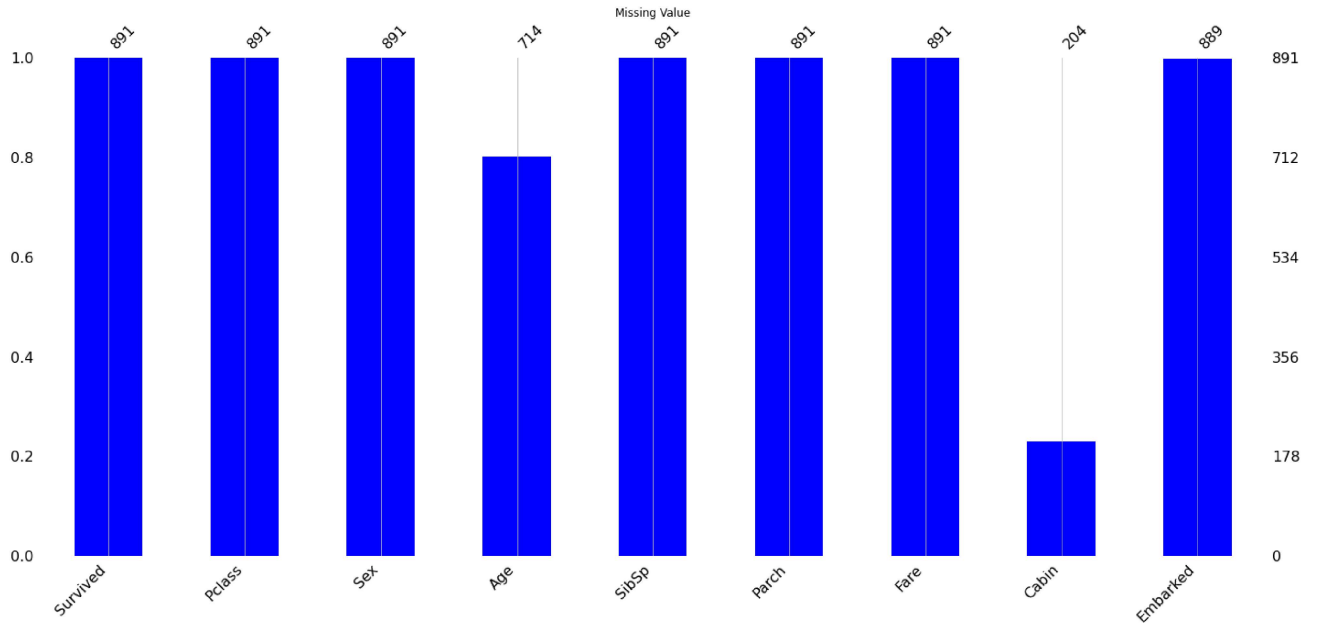
```
df.isnull().sum()
```

```
Survived    0
Pclass      0
Sex         0
Age         177
SibSp       0
Parch       0
Fare        0
```

```
Cabin      687
Embarked    2
dtype: int64
```

```
msno.bar(df, color='b')
```

```
plt.title('Missing Value')
plt.grid()
plt.show()
```



```
pip install pinguin
```

```
Requirement already satisfied: pinguin in /usr/local/lib/python3.7/dist-packages (0.0.0)
Requirement already satisfied: pandas>=0.24 in /usr/local/lib/python3.7/dist-packages (fr
Requirement already satisfied: matplotlib>=3.0.2 in /usr/local/lib/python3.7/dist-packa
Requirement already satisfied: tabulate in /usr/local/lib/python3.7/dist-packages (fr
Requirement already satisfied: numpy>=1.15 in /usr/local/lib/python3.7/dist-packages
Requirement already satisfied: seaborn>=0.9.0 in /usr/local/lib/python3.7/dist-packag
Requirement already satisfied: pandas-flavor>=0.1.2 in /usr/local/lib/python3.7/dist-
Requirement already satisfied: outdated in /usr/local/lib/python3.7/dist-packages (fr
Requirement already satisfied: scipy>=1.3 in /usr/local/lib/python3.7/dist-packages (
```

Requirement already satisfied: scikit-learn in /usr/local/lib/python3.7/dist-packages  
 Requirement already satisfied: statsmodels>=0.10.0 in /usr/local/lib/python3.7/dist-packages  
 Requirement already satisfied: python-dateutil>=2.7.3 in /usr/local/lib/python3.7/dist-packages  
 Requirement already satisfied: pytz>=2017.2 in /usr/local/lib/python3.7/dist-packages  
 Requirement already satisfied: pyparsing!=2.0.4,!=2.1.2,!=2.1.6,>=2.0.1 in /usr/local/lib/python3.7/dist-packages  
 Requirement already satisfied: kiwisolver>=1.0.1 in /usr/local/lib/python3.7/dist-packages  
 Requirement already satisfied: cyclor>=0.10 in /usr/local/lib/python3.7/dist-packages  
 Requirement already satisfied: xarray in /usr/local/lib/python3.7/dist-packages (from pandas)  
 Requirement already satisfied: requests in /usr/local/lib/python3.7/dist-packages (from pandas)  
 Requirement already satisfied: littleutils in /usr/local/lib/python3.7/dist-packages  
 Requirement already satisfied: joblib>=0.11 in /usr/local/lib/python3.7/dist-packages  
 Requirement already satisfied: patsy>=0.4.0 in /usr/local/lib/python3.7/dist-packages  
 Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.7/dist-packages (from pandas)  
 Requirement already satisfied: setuptools>=40.4 in /usr/local/lib/python3.7/dist-packages  
 Requirement already satisfied: urllib3!=1.25.0,!=1.25.1,<1.26,>=1.21.1 in /usr/local/lib/python3.7/dist-packages  
 Requirement already satisfied: chardet<4,>=3.0.2 in /usr/local/lib/python3.7/dist-packages  
 Requirement already satisfied: certifi>=2017.4.17 in /usr/local/lib/python3.7/dist-packages  
 Requirement already satisfied: idna<3,>=2.5 in /usr/local/lib/python3.7/dist-packages



```
df['Age'].fillna(df.Age.median(),inplace=True)
df.Age.isna().any()
```

False

```
np.random.seed(40)
df['Cabin'].fillna(np.random.choice(df.Cabin.unique()),inplace=True)
df.Cabin.isna().any()
```

False

```
df.drop(columns='Cabin',inplace=True)
```

```
df.dropna(inplace=True)
```

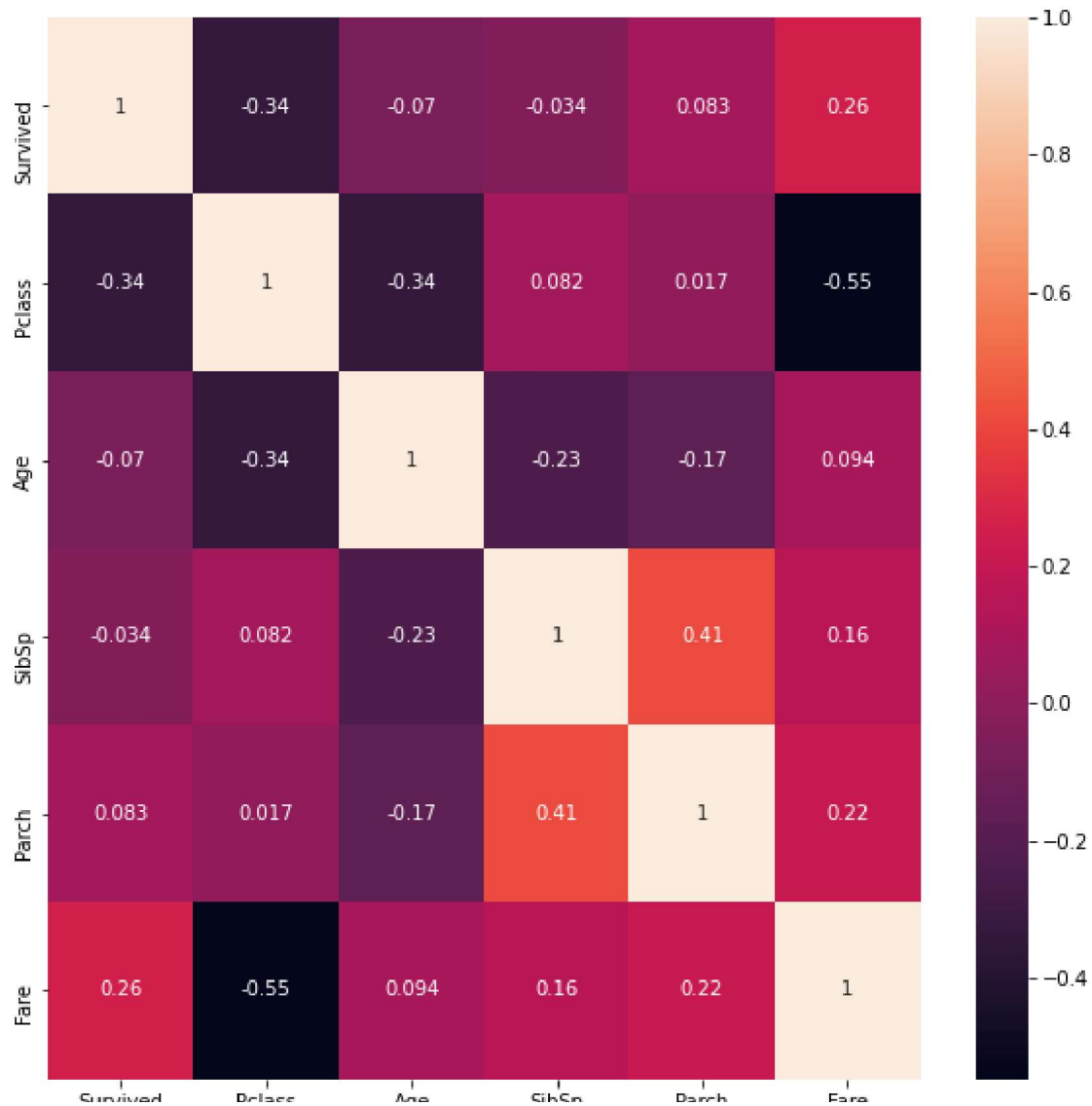
## Recheck If missing Value

```
df.isna().sum()
```

```
Survived    0
Pclass      0
Sex         0
Age         0
SibSp       0
Parch       0
Fare        0
Embarked    0
dtype: int64
```

```
plt.figure(figsize=(10,10))
sns.heatmap(data=df.corr(),annot=True,yticklabels=True,linecolor='blue',cbar=True)
```

<matplotlib.axes.\_subplots.AxesSubplot at 0x7f23a4e85cd0>

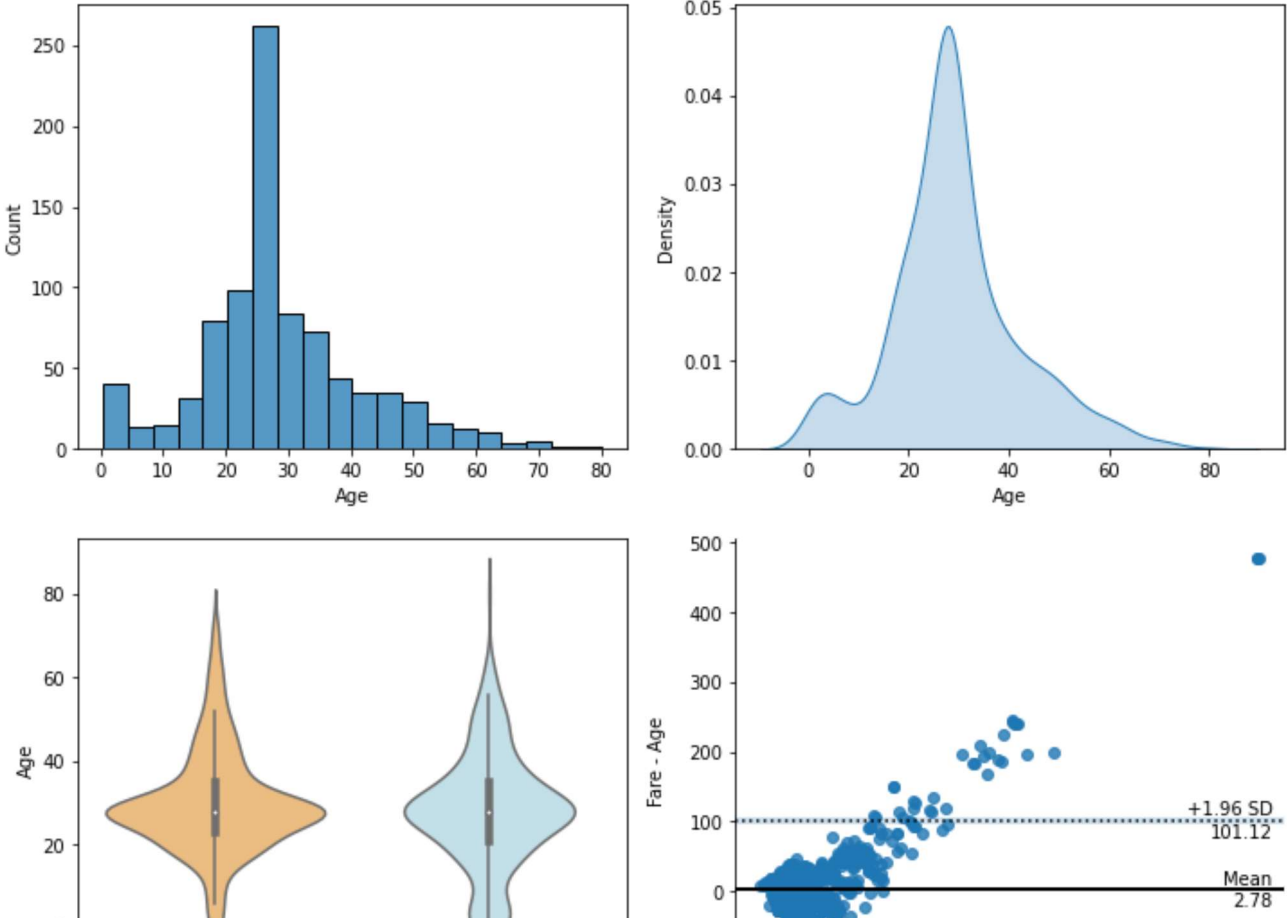


```
from pingouin import plot_blandaltman
```

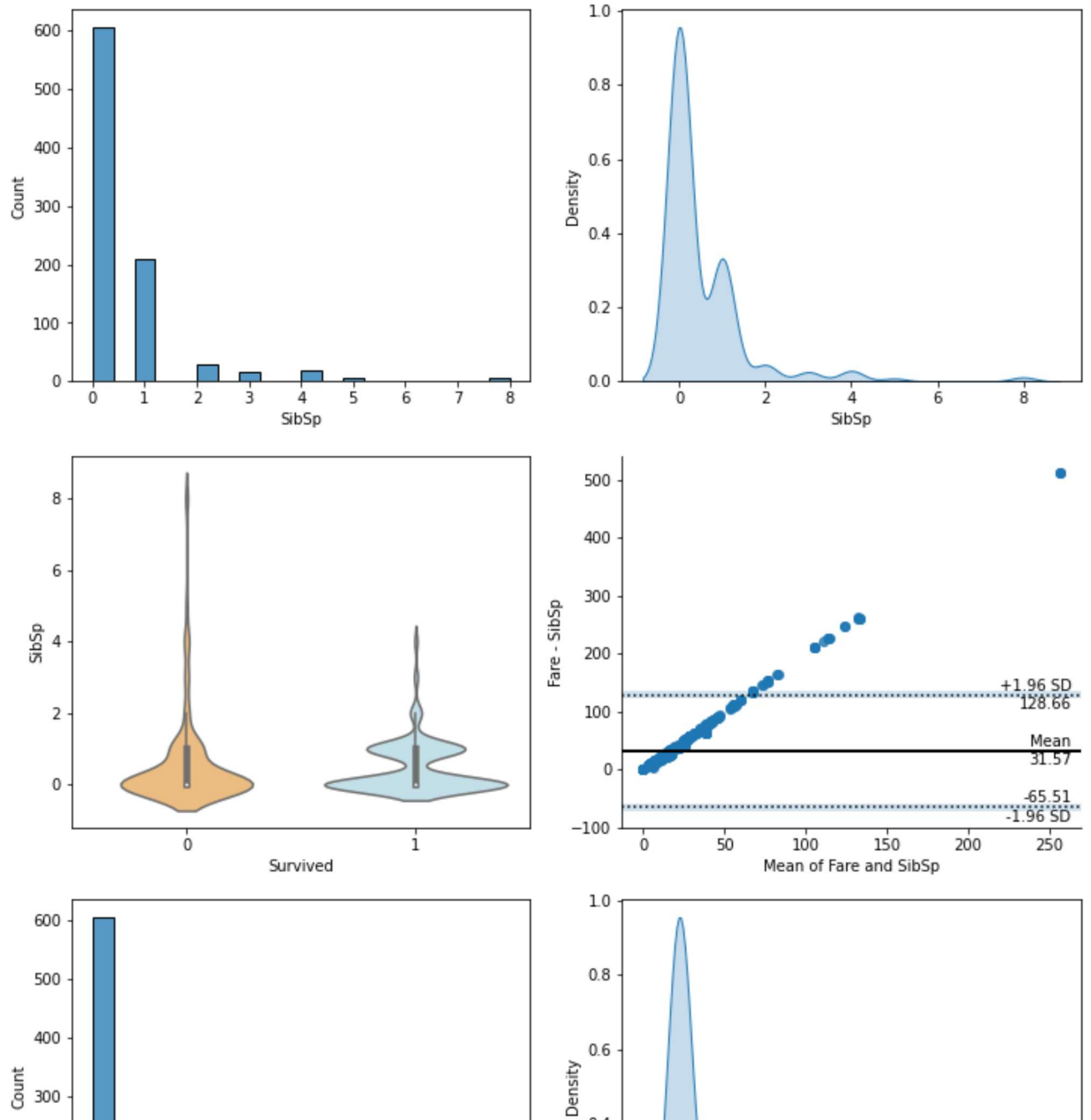
```
def plot_numerical(var):
    fig, axes = plt.subplots(2, 2, figsize=(12, 10))
    sns.histplot(x=df[var], ax=axes[0][0], bins=20)
    sns.kdeplot(x=df[var], ax=axes[0][1], fill=True)
    sns.violinplot(x=df.Survived, y=df[var], ax=axes[1][0], palette='RdYlBu')
    plot_blandaltman(x=df.Fare, y=df[var], ax=axes[1][1])
    plt.show()
```

```
numerical_columns=['Age', 'SibSp', 'Parch', 'Fare']
```

```
plot_numerical(numerical_columns[0])
```

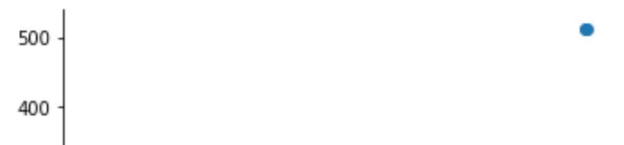
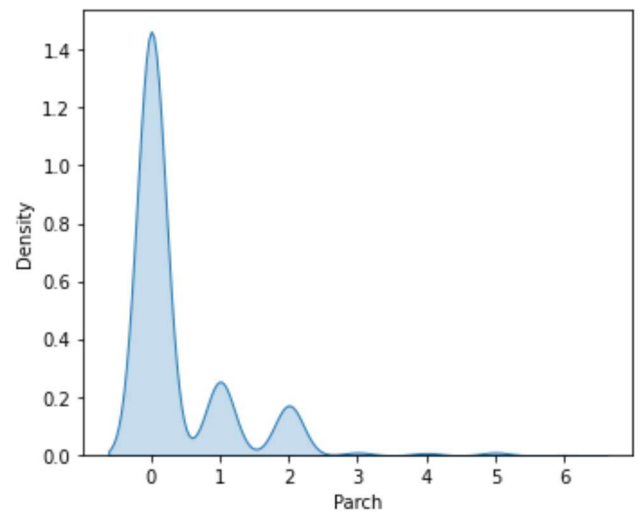
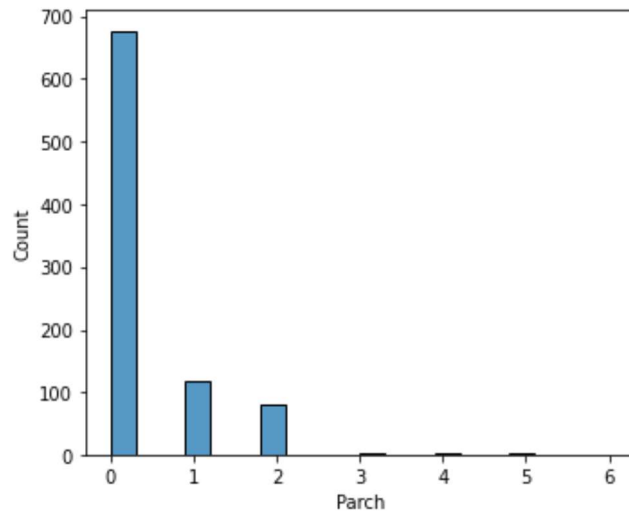
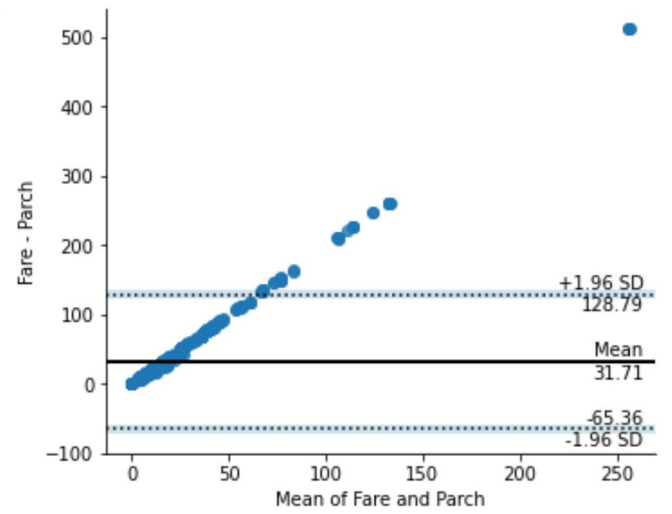
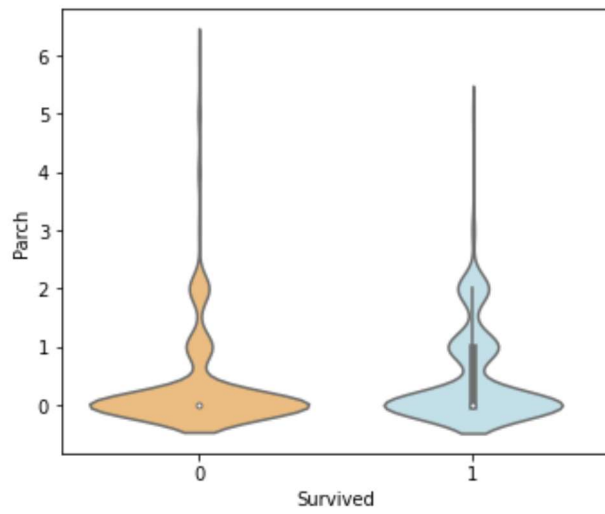
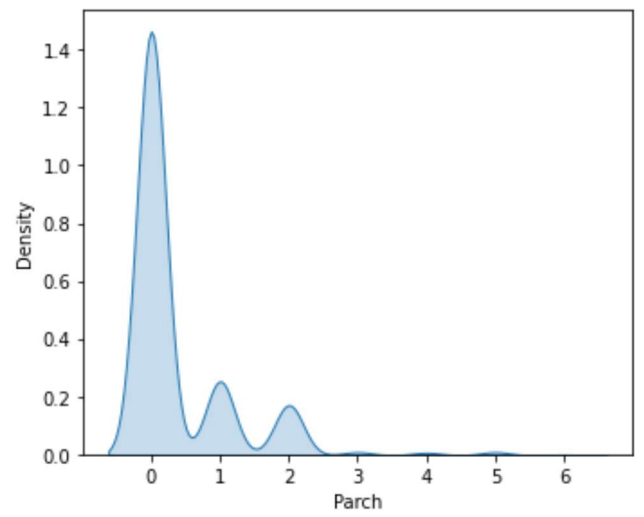
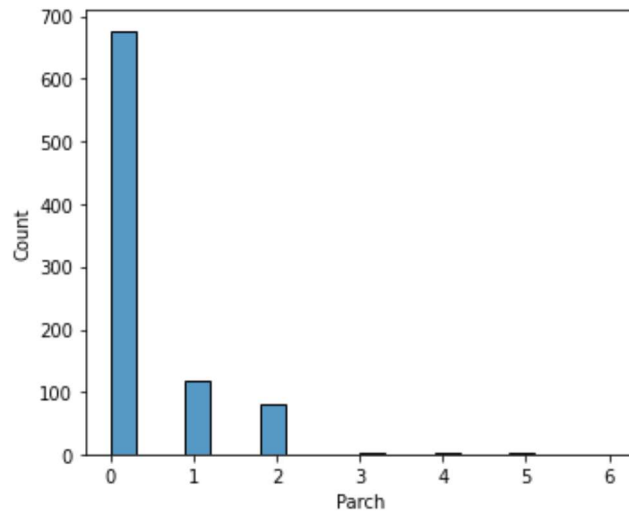


```
plot_numerical(numerical_columns[1])
```



```
plot_numerical(numerical_columns[2])
```



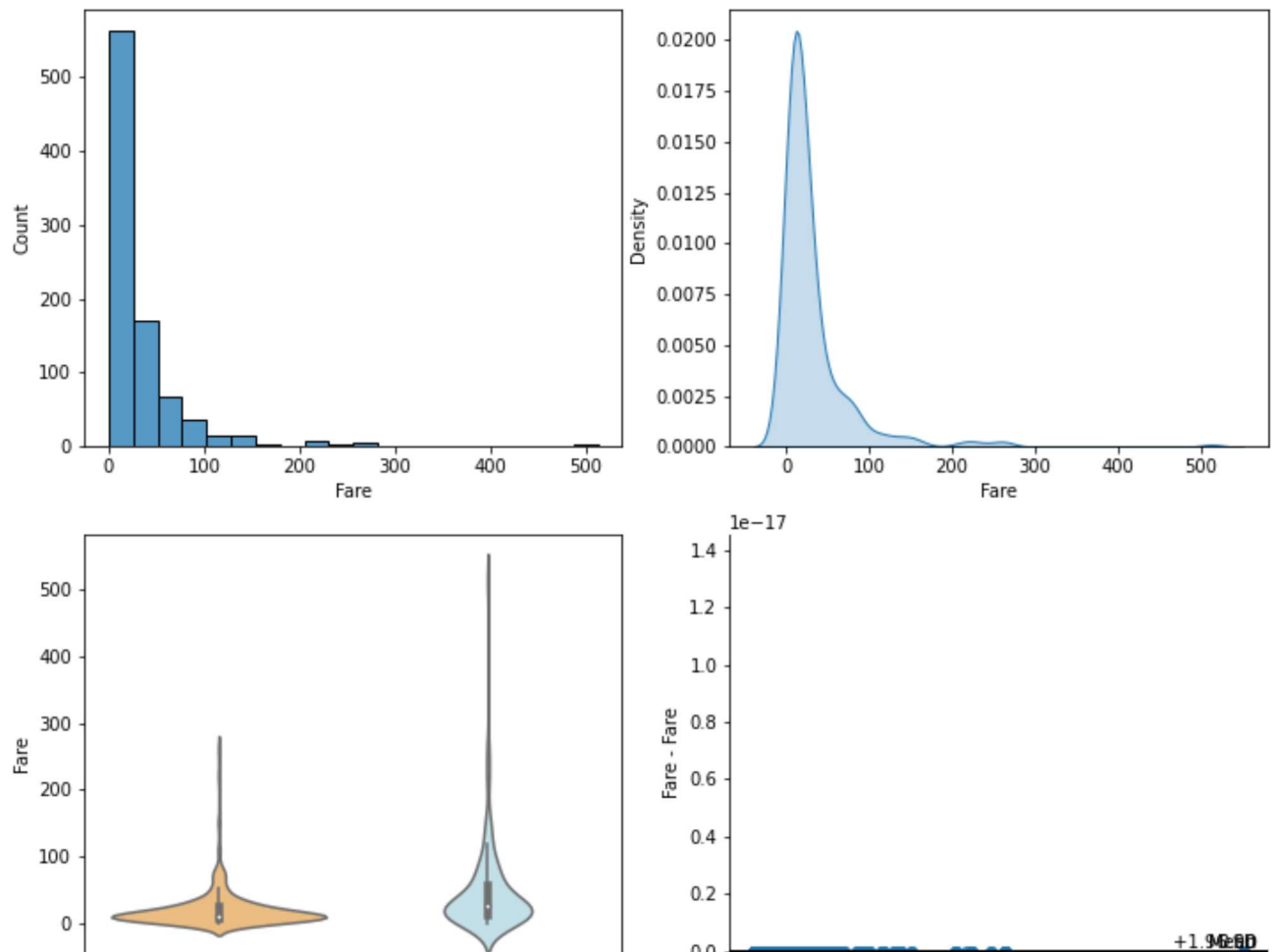


```
plot_numerical(numerical_columns[3])
```

```

/usr/local/lib/python3.7/dist-packages/scipy/stats/_distn_infrastructure.py:2003: RuntimeWarning: lower_bound = _a * scale + loc
/usr/local/lib/python3.7/dist-packages/scipy/stats/_distn_infrastructure.py:2004: RuntimeWarning: upper_bound = _b * scale + loc

```

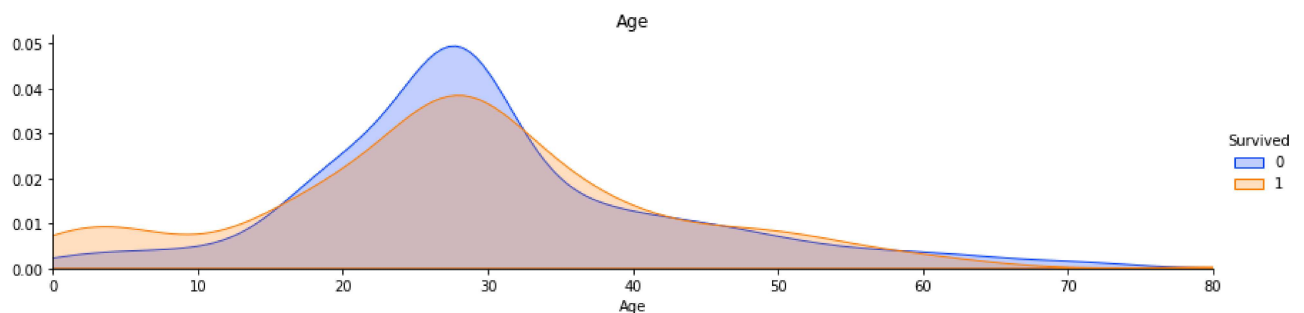


```

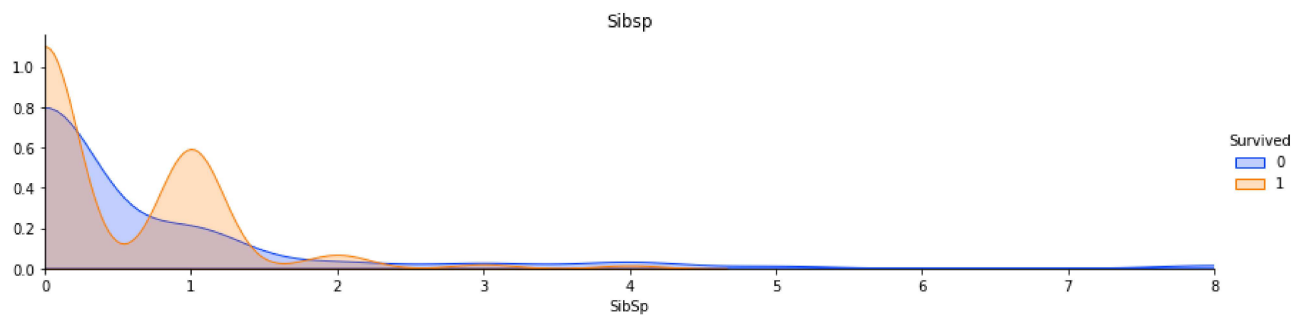
def plot_kde(var):
    facet=sns.FacetGrid(df,hue='Survived',aspect=4,palette='bright')
    facet.map(sns.kdeplot,var,shade=True)
    facet.set(xlim=(0,df[var].max()))
    facet.add_legend()
    plt.title(var.title())
    plt.show()

```

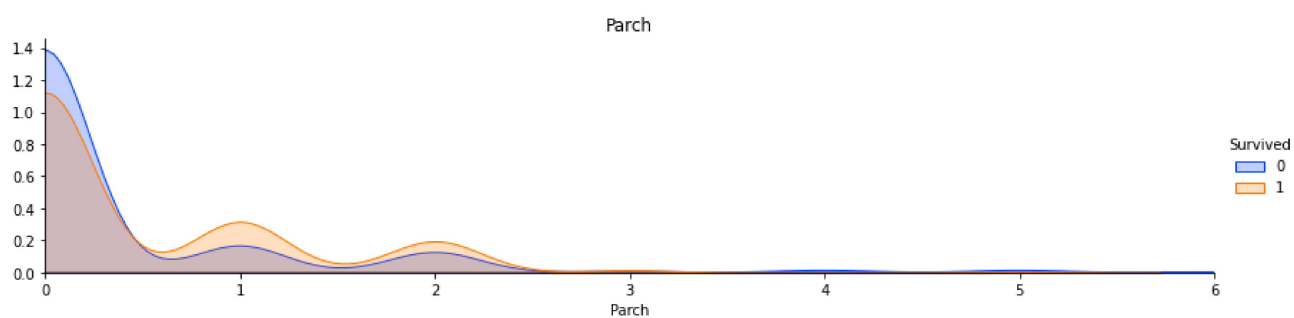
```
plot_kde(numerical_columns[0])
```



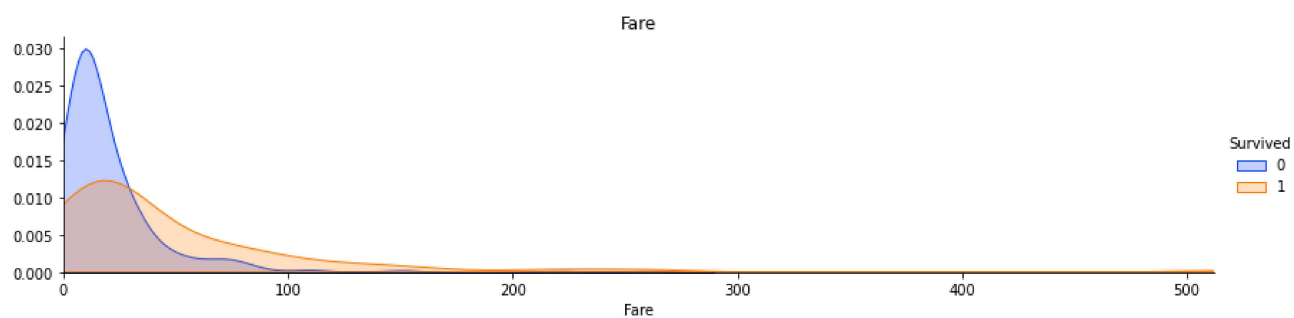
```
plot_kde(numerical_columns[1])
```



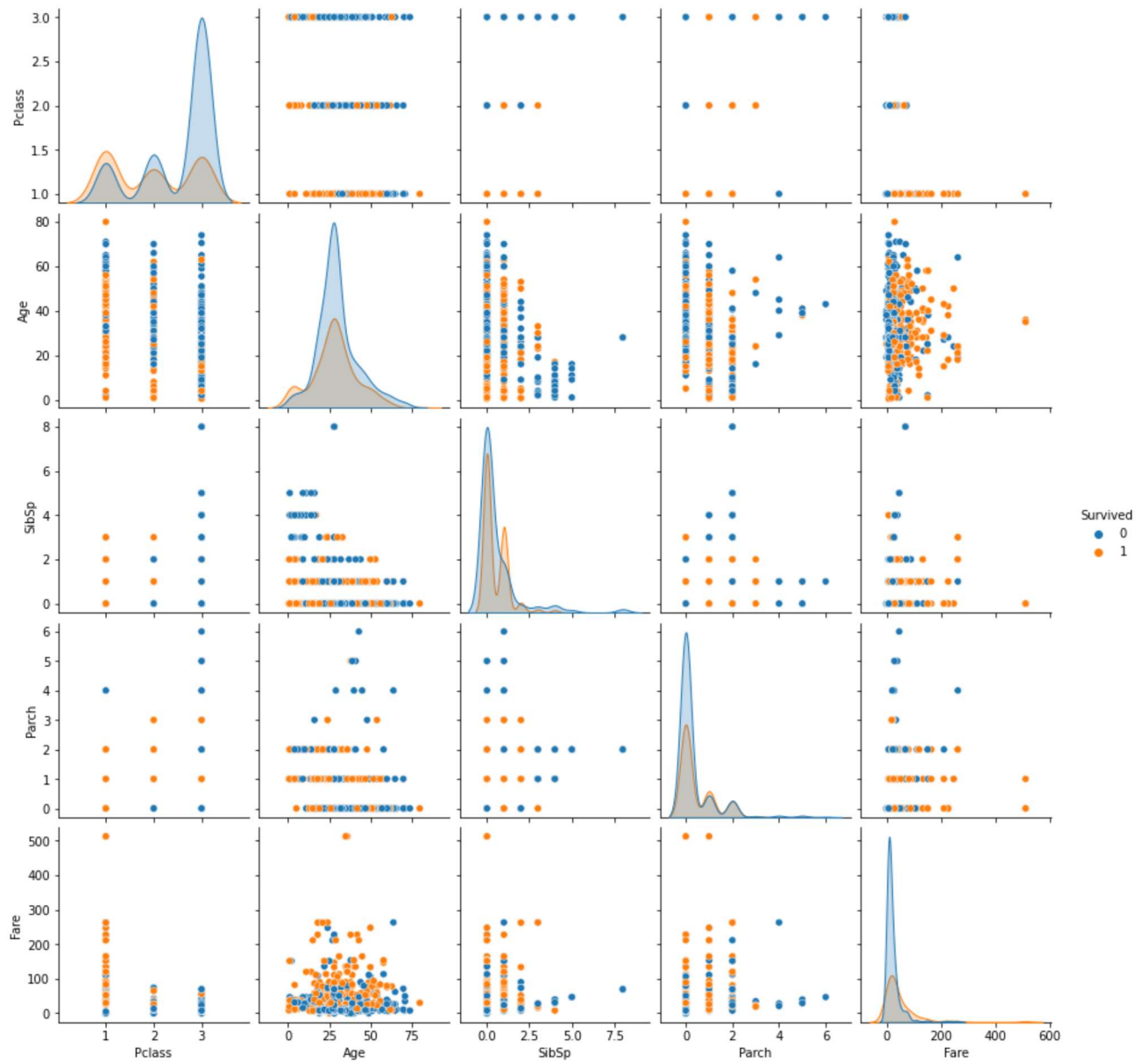
```
plot_kde(numerical_columns[2])
```



```
plot_kde(numerical_columns[3])
```



```
sns.pairplot(df,hue='Survived')  
plt.show()
```



## Data Preprocessing

```
pip install yellowbrick
```

```
Requirement already satisfied: yellowbrick in /usr/local/lib/python3.7/dist-packages
Requirement already satisfied: scikit-learn>=0.20 in /usr/local/lib/python3.7/dist-packages
Requirement already satisfied: matplotlib!=3.0.0,>=1.5.1 in /usr/local/lib/python3.7/dist-packages
Requirement already satisfied: numpy>=1.13.0 in /usr/local/lib/python3.7/dist-packages
Requirement already satisfied: scipy>=1.0.0 in /usr/local/lib/python3.7/dist-packages
Requirement already satisfied: cycler>=0.10.0 in /usr/local/lib/python3.7/dist-packages
Requirement already satisfied: joblib>=0.11 in /usr/local/lib/python3.7/dist-packages
Requirement already satisfied: pyparsing!=2.0.4,!2.1.2,!2.1.6,>=2.0.1 in /usr/local/lib/python3.7/dist-packages
Requirement already satisfied: kiwisolver>=1.0.1 in /usr/local/lib/python3.7/dist-packages
```

Requirement already satisfied: python-dateutil>=2.1 in /usr/local/lib/python3.7/dist-  
 Requirement already satisfied: six in /usr/local/lib/python3.7/dist-packages (from cy

## Data Splitting

```
x=df.drop(['Survived','Pclass','Embarked','Sex'],axis=1)
```

```
y=df['Survived']
```

```
x[:3]
```

	Age	SibSp	Parch	Fare
0	22.0	1	0	7.2500
1	38.0	1	0	71.2833
2	26.0	0	0	7.9250

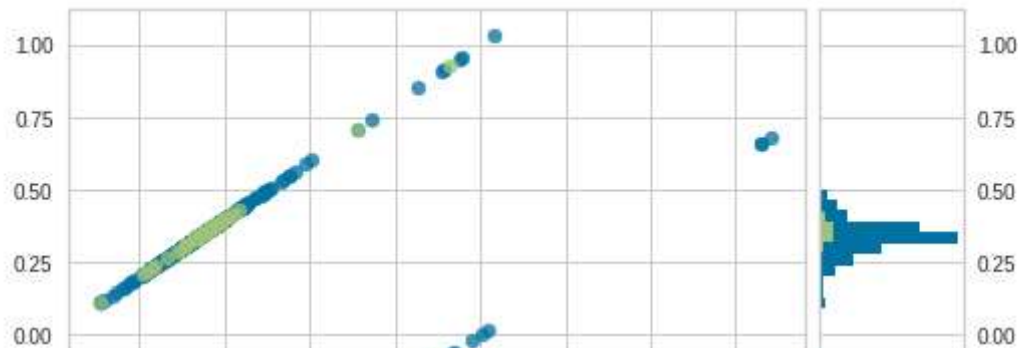
```
y[:5]
```

```
0    0
1    1
2    1
3    1
4    0
Name: Survived, dtype: int64
```

```
import yellowbrick
from yellowbrick.regressor import ResidualsPlot
from sklearn.linear_model import LinearRegression
from sklearn.model_selection import train_test_split
```

```
# Create training and test sets
X_train, X_test, y_train, y_test = train_test_split(
    x, y, test_size=0.1
)
```

```
visualizer = ResidualsPlot(LinearRegression())
visualizer.fit(X_train, y_train)
visualizer.score(X_test, y_test)
plt.show()
```



X\_train.shape

(800, 4)

y\_train.shape

(800,)

## ▼ Gaussian Navie Bayes Classifier

```
from sklearn.metrics import confusion_matrix, classification_report, accuracy_score
from sklearn.metrics import recall_score, precision_score, f1_score
```

```
from sklearn.naive_bayes import GaussianNB
clf = GaussianNB()
clf.fit(X_train, y_train)
GaussianNB()
```

```
GaussianNB(priors=None, var_smoothing=1e-09)
```

```
y_pred=clf.predict(X_test)
print("Accuracy Score:", accuracy_score(y_test, y_pred))
```

```
Accuracy Score: 0.6966292134831461
```

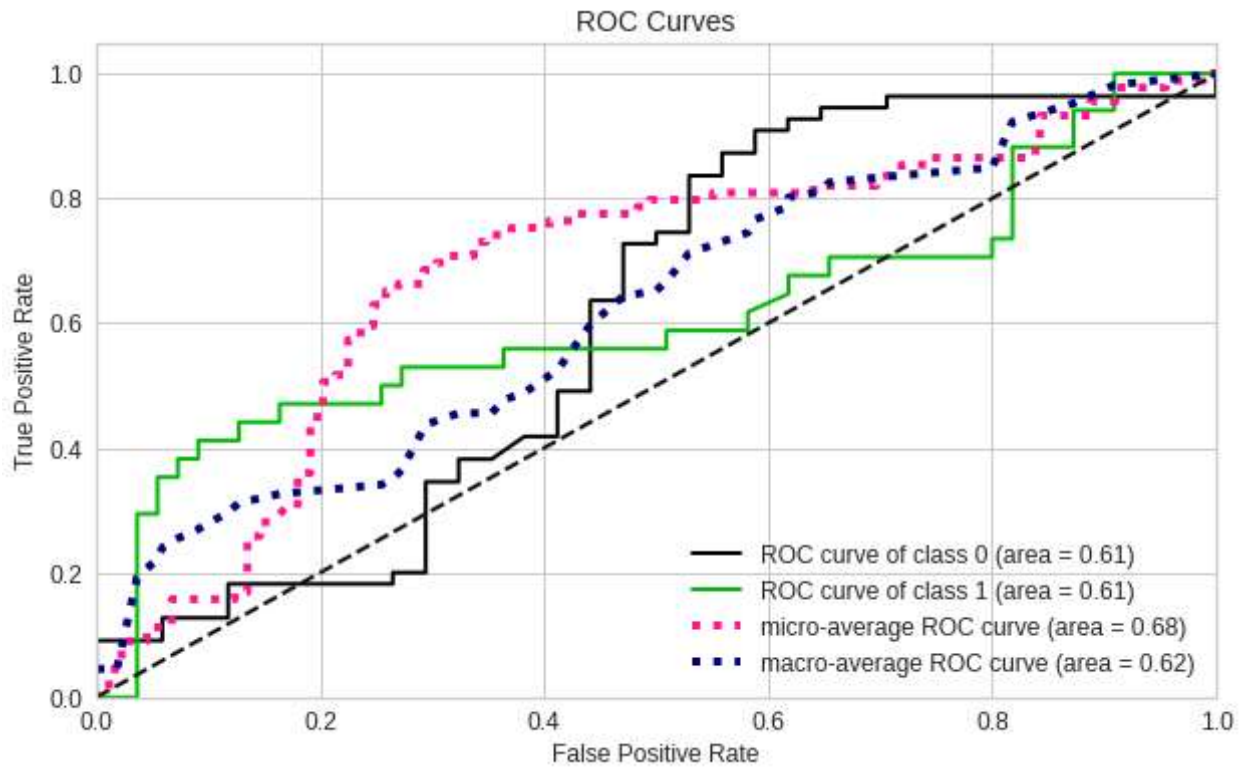
```
y_probab=clf.predict_proba(X_test)
```

## ▼ ROC Curve

```
print("Recall Score:", recall_score(y_test, y_pred))
```

```
Recall Score: 0.2647058823529412
```

```
skplt.metrics.plot_roc(y_test, y_probab, figsize=(10, 6), title_fontsize=14, text_fontsize=12)
plt.show()
```

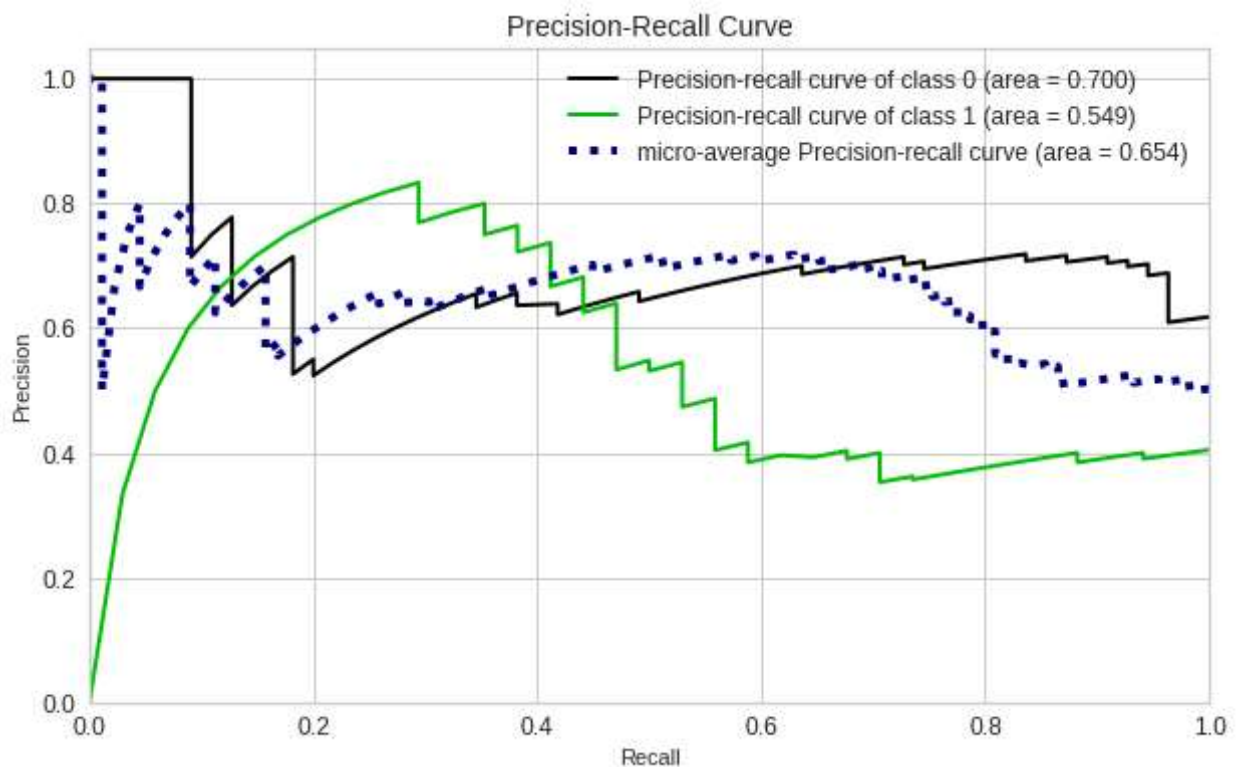


## ▼ Precision Score

```
print("Precision Score:", precision_score(y_test, y_pred))
```

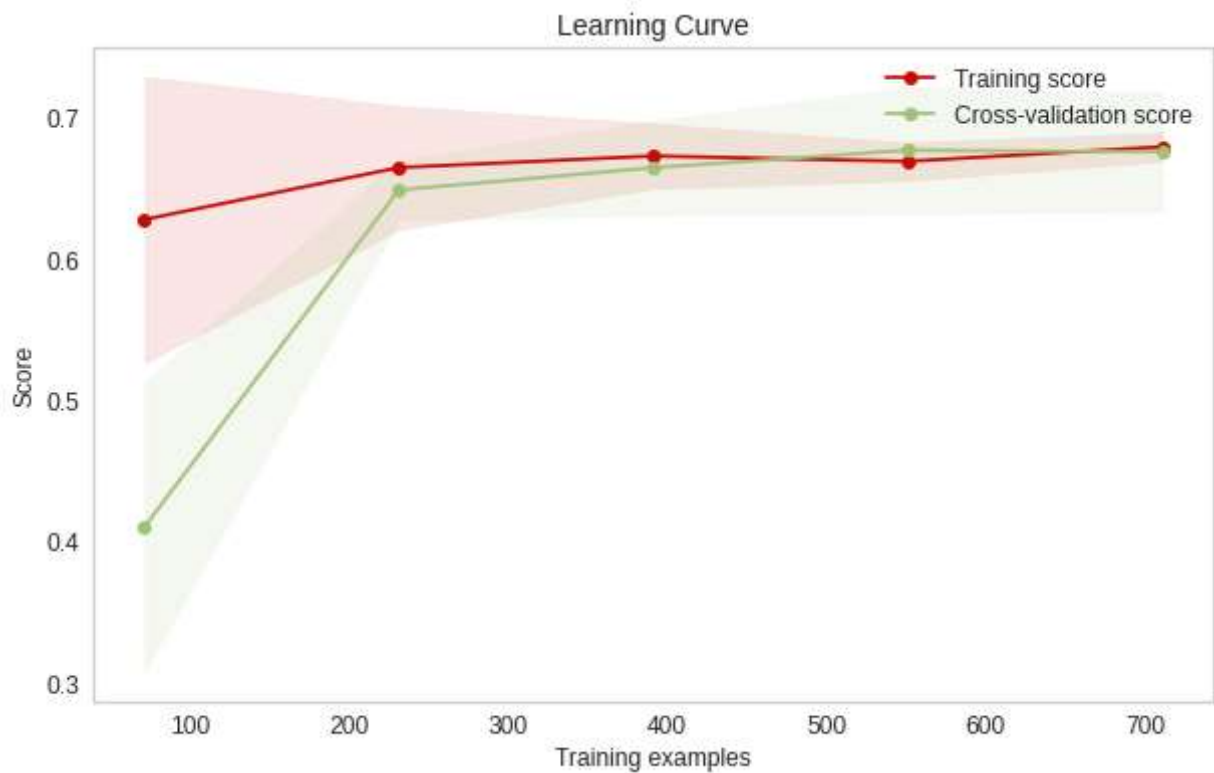
Precision Score: 0.8181818181818182

```
skplt.metrics.plot_precision_recall(y_test, y_prob, figsize=(10,6), title_fontsize=14, text_
plt.show())
```



## ▼ Learning Curve

```
skplt.estimators.plot_learning_curve(clf,x,y,figsize=(10,6),title_fontsize=14,text_fontsize=14)
plt.show()
```



## ▼ Lift Curve

```
skplt.metrics.plot_lift_curve(y_test,y_probas,figsize=(10,6),title_fontsize=14,text_fontsize=14)
plt.show()
```

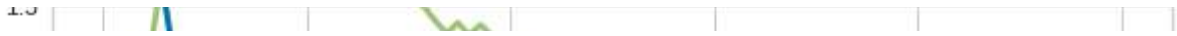


Lift Curve



```
print("F1 Score:",f1_score(y_test,y_pred))
```

F1 Score: 0.39999999999999997



## ▼ Confusion Matrix



```
confusion_matrix(y_test,y_pred)
```

```
array([[53,  2],  
       [25,  9]])
```



```
skplt.metrics.plot_confusion_matrix(y_test,y_pred,figsize=(10,6),title_fontsize=14,text_fo  
plt.show())
```

