lp3-ml-lab-exp-no-6

October 5, 2024

[1]:

```
import pandas as pd
import numpy as np
from matplotlib import pyplot as plt
import seaborn as sns
from sklearn.metrics import accuracy_score
from sklearn.preprocessing import LabelEncoder, OneHotEncoder
from sklearn.impute import KNNImputer
```

[2]:

```
train_data=pd.read_csv("C:/Users/Atharva/Downloads/train.csv")
train_data.head()
```

[2]:

```
   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  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

   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
```

[3]:

```
train_data.shape
```

[3]:

```
(891, 12)
```

[4]:

```
train_data.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
```

[5]:

```
train_data['Survived'].value_counts()
```

[5]:

```
Survived
0    549
1    342
Name: count, dtype: int64
```

[6]:

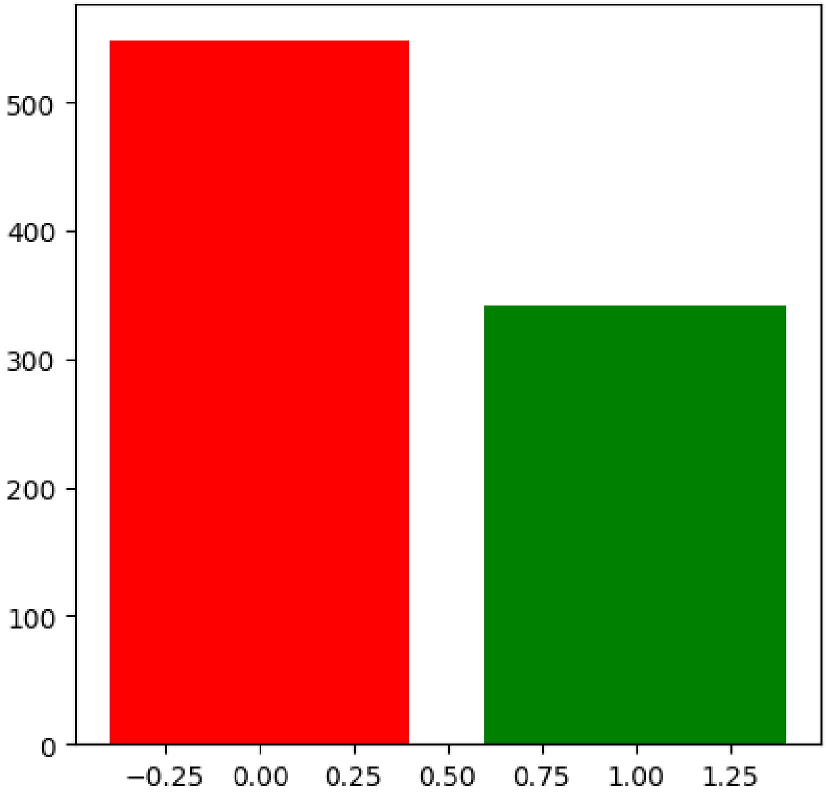
```
train_data['Survived'].value_counts().keys()
```

[6]:

```
Index([0, 1], dtype='int64', name='Survived')
```

[7]:

```
plt.figure(figsize=(5,5))
plt.bar(list(train_data["Survived"].value_counts().
  ↪keys()),list(train_data["Survived"].value_counts()),color=["r","g"])
plt.show()
```



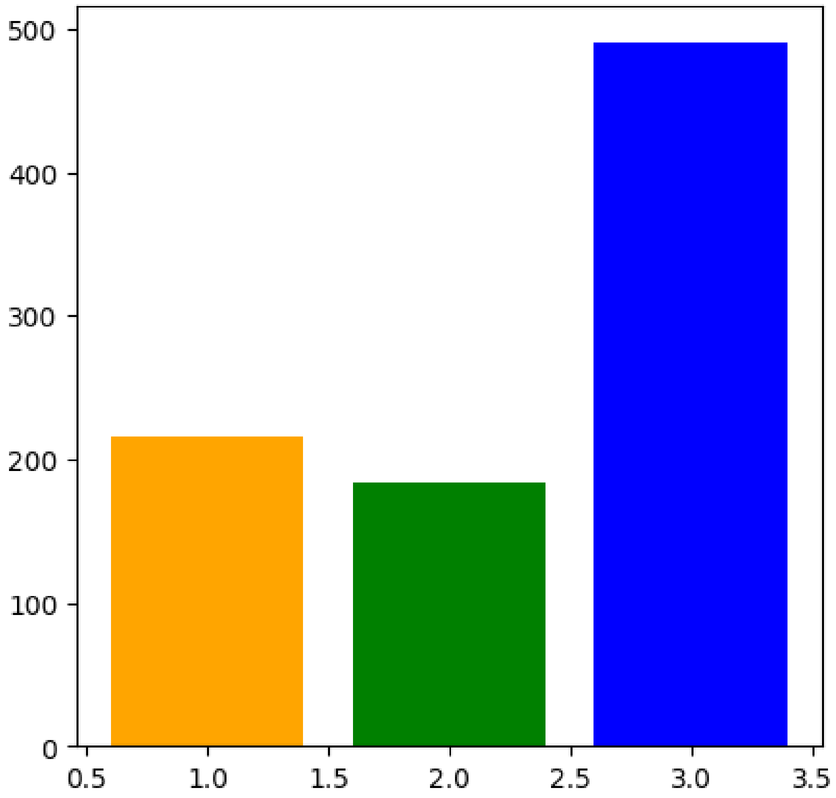
[8]:
```
train_data['Pclass'].value_counts()
```

[8]:
```
Pclass
3    491
1    216
2    184
Name: count, dtype: int64
```

[9]:
```
train_data['Pclass'].value_counts().keys()
```

[9]:
```
Index([3, 1, 2], dtype='int64', name='Pclass')
```

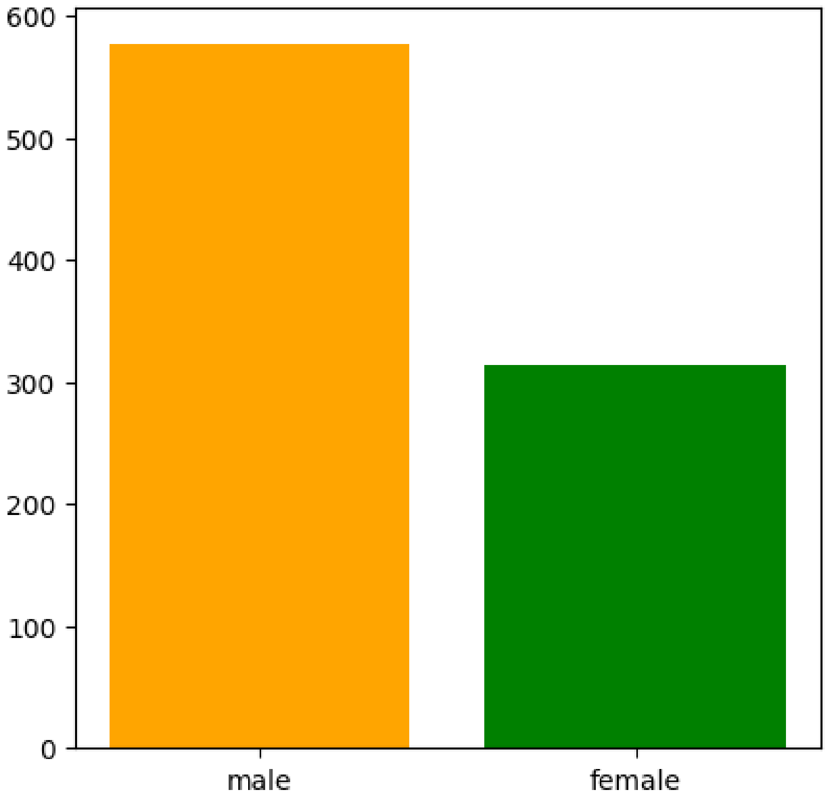
[10]:
```
plt.figure(figsize=(5,5))
plt.bar(list(train_data["Pclass"].value_counts().
  ↪keys()),list(train_data["Pclass"].
  ↪value_counts()),color=["blue","orange","green"])
plt.show()
```



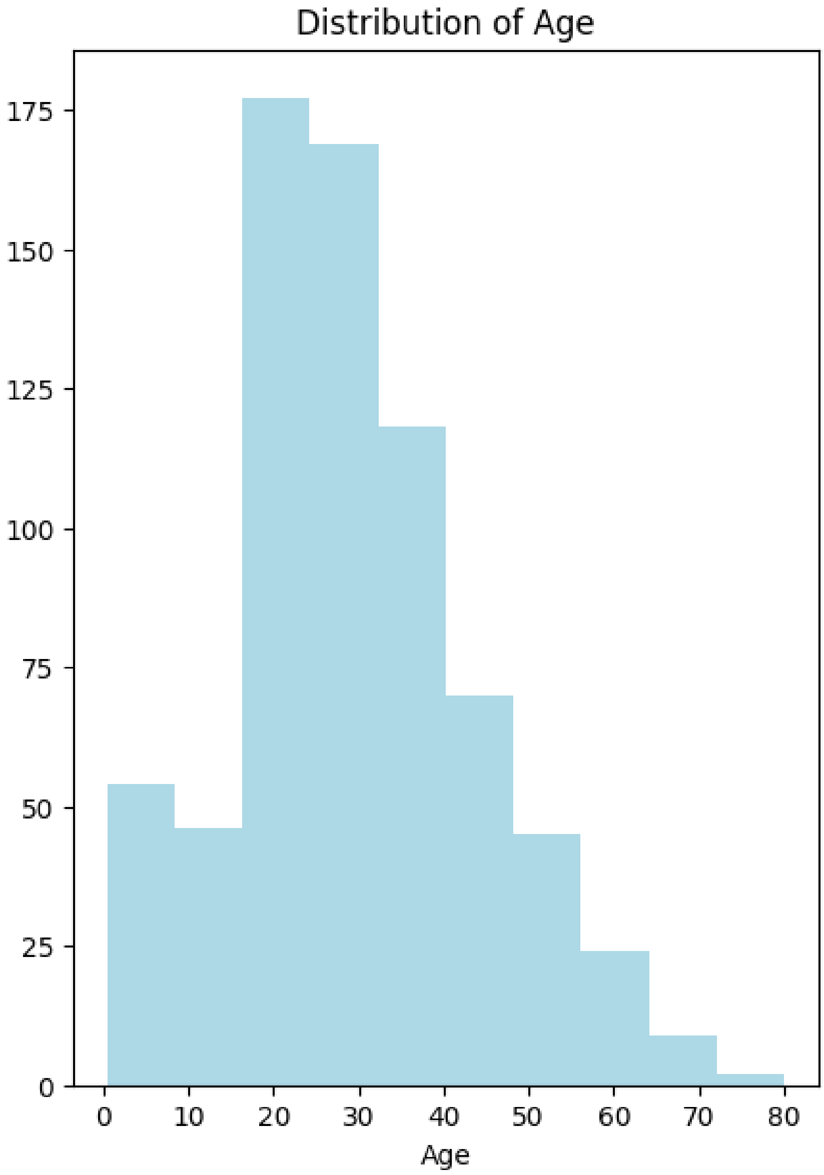
```
[11]: train_data['Sex'].value_counts()
```

```
[11]: Sex
      male      577
      female    314
      Name: count, dtype: int64
```

```
[12]: plt.figure(figsize=(5,5))
      plt.bar(list(train_data["Sex"].value_counts().keys()),list(train_data["Sex"].
       ↪value_counts()),color=["orange","green"])
      plt.show()
```

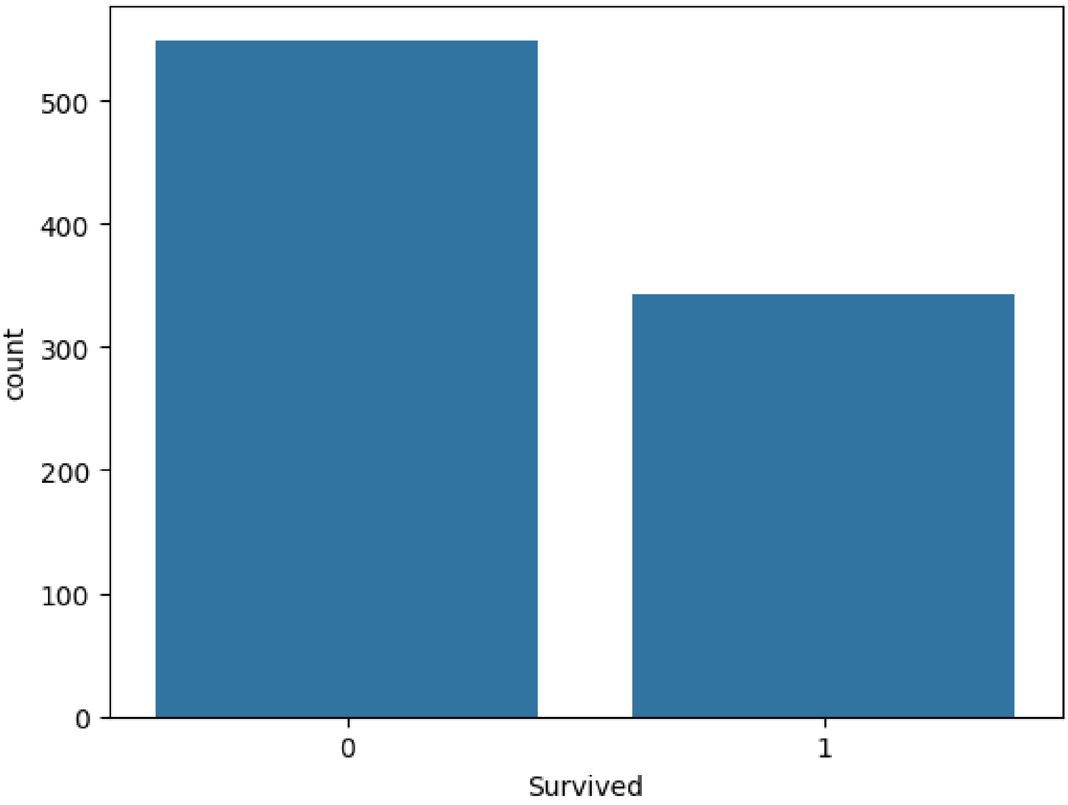


[13]:
```
plt.figure(figsize=(5,7))
plt.hist(train_data["Age"],color="lightblue")
plt.title("Distribution of Age")
plt.xlabel("Age")
plt.show()
```



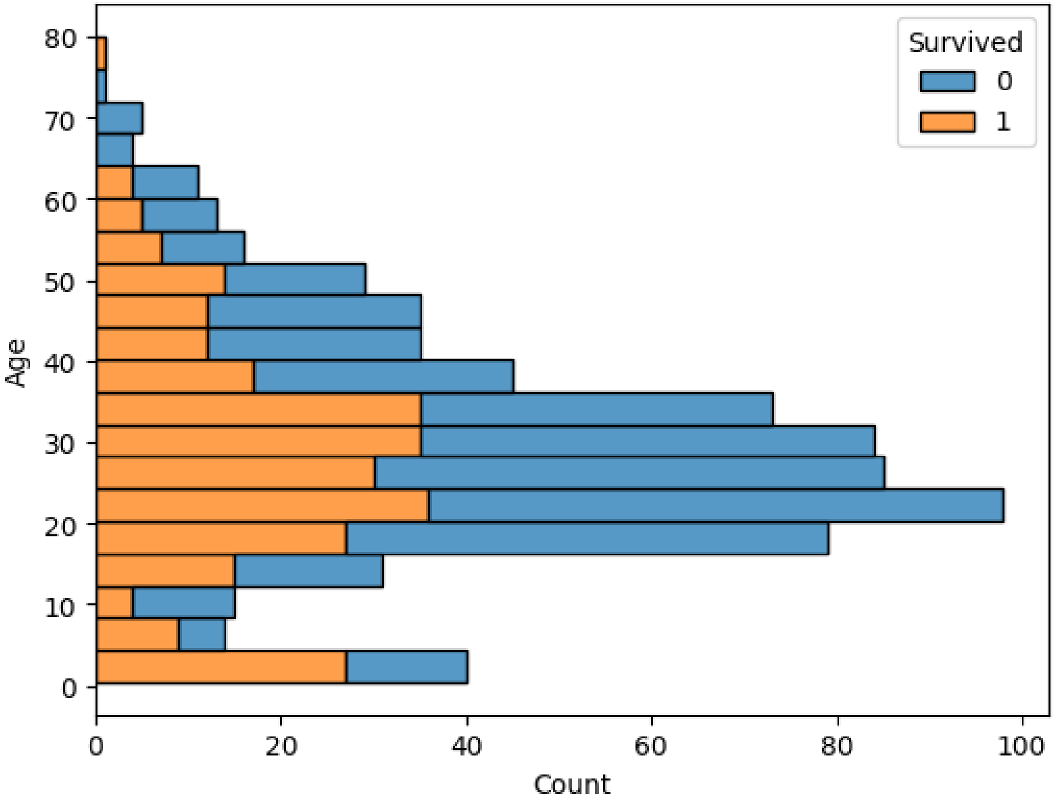
```
[14]: sns.countplot(x="Survived",data=train_data)
```

```
[14]: <Axes: xlabel='Survived', ylabel='count'>
```



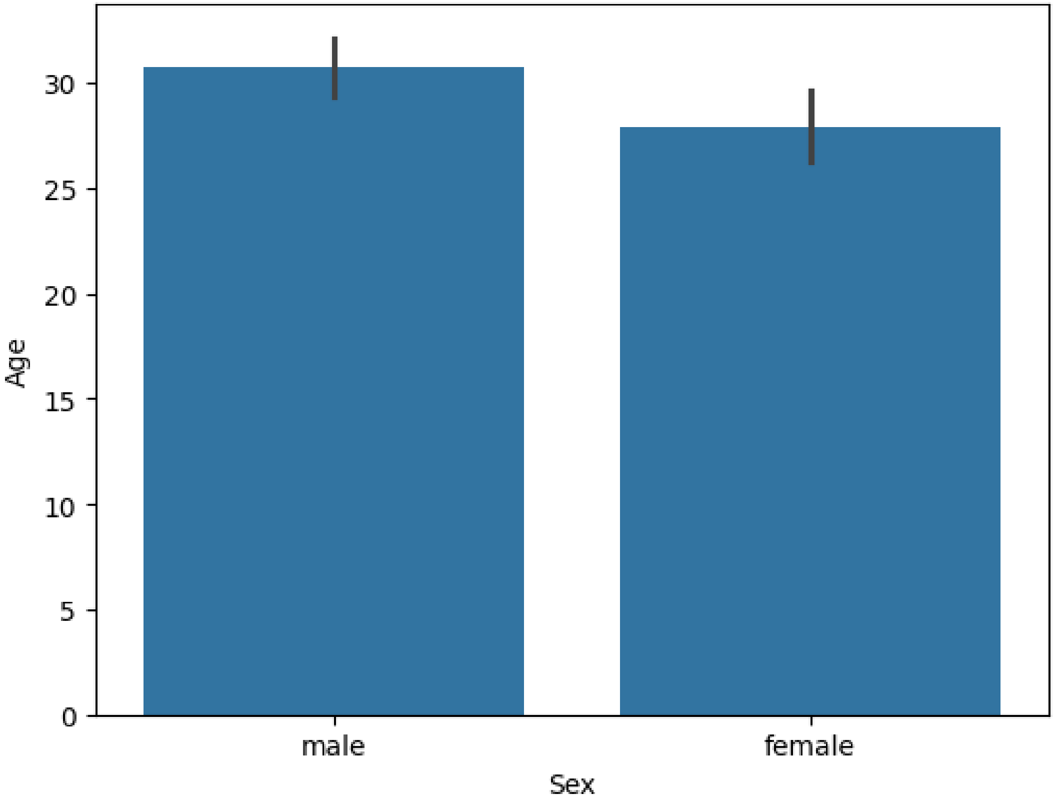
[15]:
```
sns.histplot(y=train_data["Age"],hue=train_data["Survived"],multiple="stack")
```

[15]:
```
<Axes: xlabel='Count', ylabel='Age'>
```



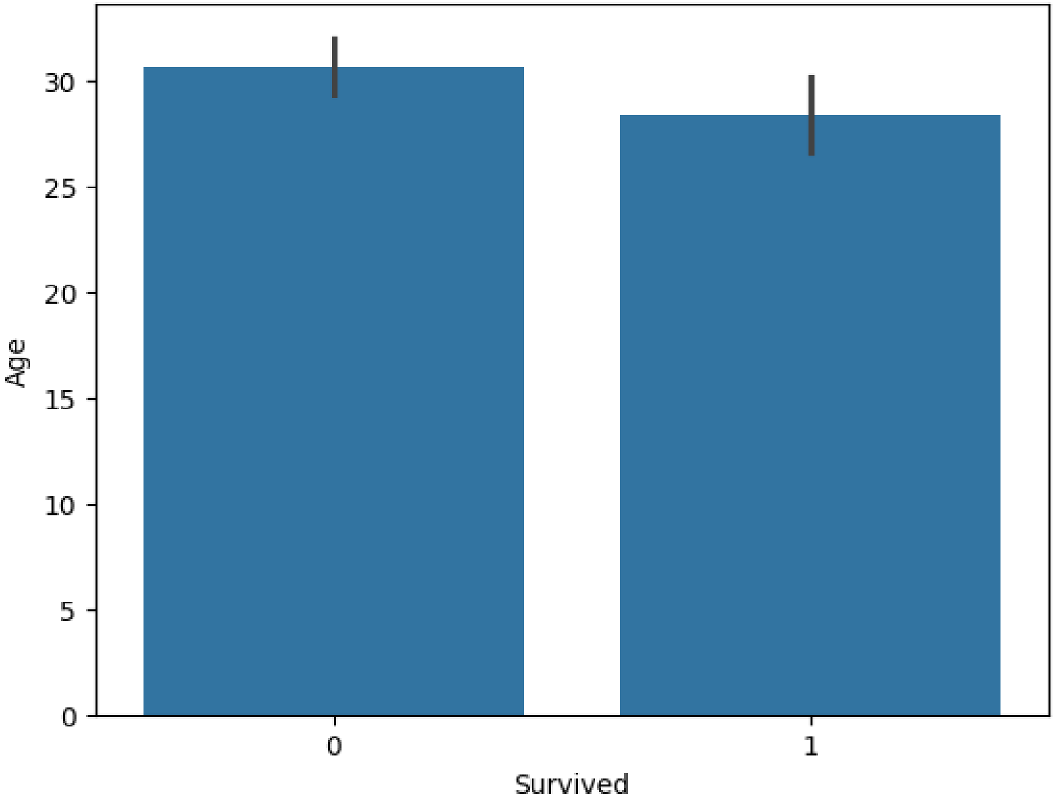
```
[16]: sns.barplot(x='Sex',y='Age',data=train_data)
```

```
[16]: <Axes: xlabel='Sex', ylabel='Age'>
```



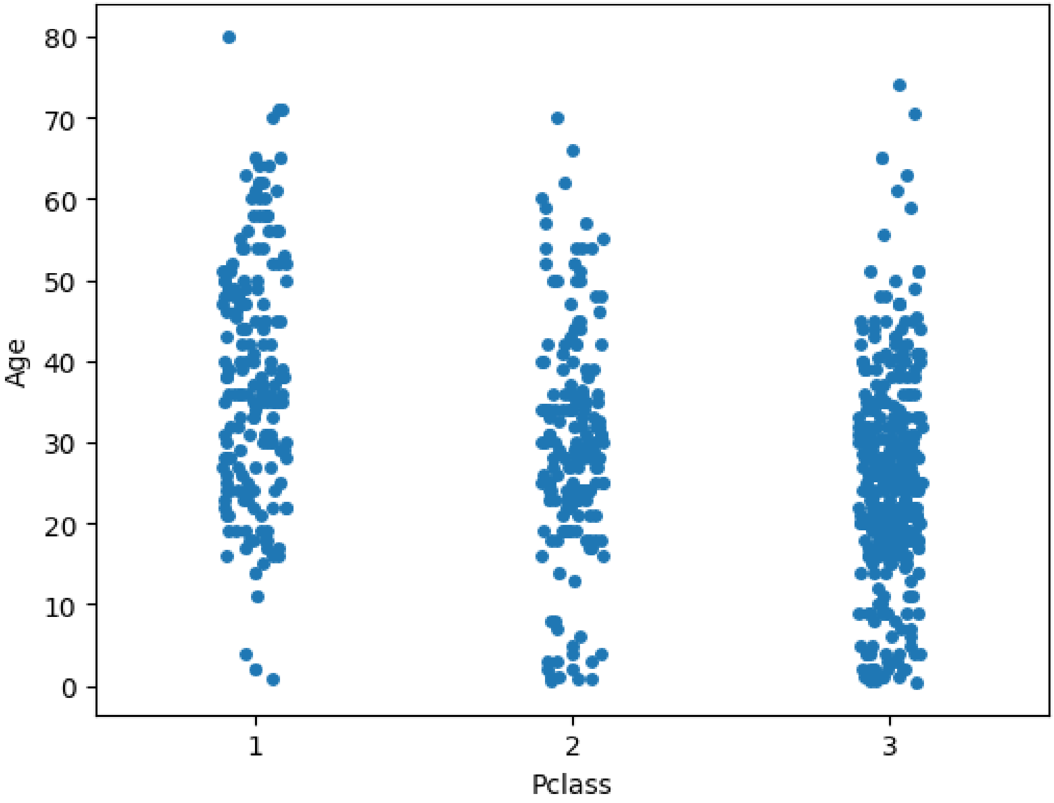
```
[17]: sns.barplot(x='Survived',y='Age',data=train_data)
```

```
[17]: <Axes: xlabel='Survived', ylabel='Age'>
```



```
[18]: sns.stripplot(x='Pclass', y='Age', data=train_data)
```

```
[18]: <Axes: xlabel='Pclass', ylabel='Age'>
```



```
[19]: train_data.describe()
```

```
[19]:        PassengerId    Survived      Pclass         Age       SibSp  \
      count   891.000000  891.000000  891.000000  714.000000  891.000000
      mean    446.000000    0.383838    2.308642   29.699118    0.523008
      std     257.353842    0.486592    0.836071   14.526497    1.102743
      min       1.000000    0.000000    1.000000    0.420000    0.000000
      25%     223.500000    0.000000    2.000000   20.125000    0.000000
      50%     446.000000    0.000000    3.000000   28.000000    0.000000
      75%     668.500000    1.000000    3.000000   38.000000    1.000000
      max     891.000000    1.000000    3.000000   80.000000    8.000000

                  Parch        Fare
      count  891.000000  891.000000
      mean     0.381594   32.204208
      std      0.806057   49.693429
      min      0.000000    0.000000
      25%      0.000000    7.910400
      50%      0.000000   14.454200
      75%      0.000000   31.000000
      max      6.000000  512.329200
```

[20]:

```
train_data.drop(['PassengerId', 'Name', 'Ticket', 'Embarked'], axis=1,␣
 ↪inplace=True)
train_data['Sex'] = LabelEncoder().fit_transform(train_data['Sex'])
```

[21]:

```
train_data['Cabin'] = train_data['Cabin'].str[0]
train_data = pd.get_dummies(train_data, columns=['Cabin'])
```

[22]:

```
train_data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 891 entries, 0 to 890
Data columns (total 15 columns):
 #   Column    Non-Null Count  Dtype
---  ------    --------------  -----
 0   Survived  891 non-null    int64
 1   Pclass    891 non-null    int64
 2   Sex       891 non-null    int32
 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_A   891 non-null    bool
 8   Cabin_B   891 non-null    bool
 9   Cabin_C   891 non-null    bool
 10  Cabin_D   891 non-null    bool
 11  Cabin_E   891 non-null    bool
 12  Cabin_F   891 non-null    bool
 13  Cabin_G   891 non-null    bool
 14  Cabin_T   891 non-null    bool
dtypes: bool(8), float64(2), int32(1), int64(4)
memory usage: 52.3 KB
```

[23]:

```
imputer = KNNImputer(n_neighbors=5)
train_data[['Age', 'Fare']] = imputer.fit_transform(train_data[['Age', 'Fare']])
```

[24]:

```
train_data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 891 entries, 0 to 890
Data columns (total 15 columns):
 #   Column    Non-Null Count  Dtype
---  ------    --------------  -----
 0   Survived  891 non-null    int64
 1   Pclass    891 non-null    int64
 2   Sex       891 non-null    int32
 3   Age       891 non-null    float64
 4   SibSp     891 non-null    int64
 5   Parch     891 non-null    int64
 6   Fare      891 non-null    float64
 7   Cabin_A   891 non-null    bool
 8   Cabin_B   891 non-null    bool
 9   Cabin_C   891 non-null    bool
 10  Cabin_D   891 non-null    bool
 11  Cabin_E   891 non-null    bool
 12  Cabin_F   891 non-null    bool
 13  Cabin_G   891 non-null    bool
 14  Cabin_T   891 non-null    bool
dtypes: bool(8), float64(2), int32(1), int64(4)
memory usage: 52.3 KB
```

[25]:

```
X_train = train_data.drop('Survived', axis=1)
y_train = train_data['Survived']
```

[26]:

```
from sklearn.tree import DecisionTreeClassifier
from sklearn.preprocessing import LabelEncoder, OneHotEncoder, StandardScaler
```

[27]:

```
model = DecisionTreeClassifier(random_state=42)
```

[28]:

```
model.fit(X_train, y_train)
```

[28]:

```
DecisionTreeClassifier(random_state=42)
```

[29]:

```
y_train_pred = model.predict(X_train)
```

[30]:

```
train_accuracy = accuracy_score(y_train, y_train_pred)
print("Accuracy:", train_accuracy)
```

```
Accuracy: 0.9865319865319865
```