titanic-first

April 27, 2022

1 1 Titanic

1.1

Titanic Kaggle * PassengerId: ID * Survived: 1 0 * Pclass * Name: * Sex * Age * SibSp * Parch * Ticket * Fare * Cabin * Embarked

[ ]:
```
import numpy as np # linear algebra
import pandas as pd # data processing, CSV file I/O (e.g. pd.read_csv)
```

1.2

descibe() isnull()

[ ]:
```
test_data = pd.read_csv("kaggle/input/titanic/test.csv")
train_data = pd.read_csv("kaggle/input/titanic/train.csv")
```

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

[ ]:
```
   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
```

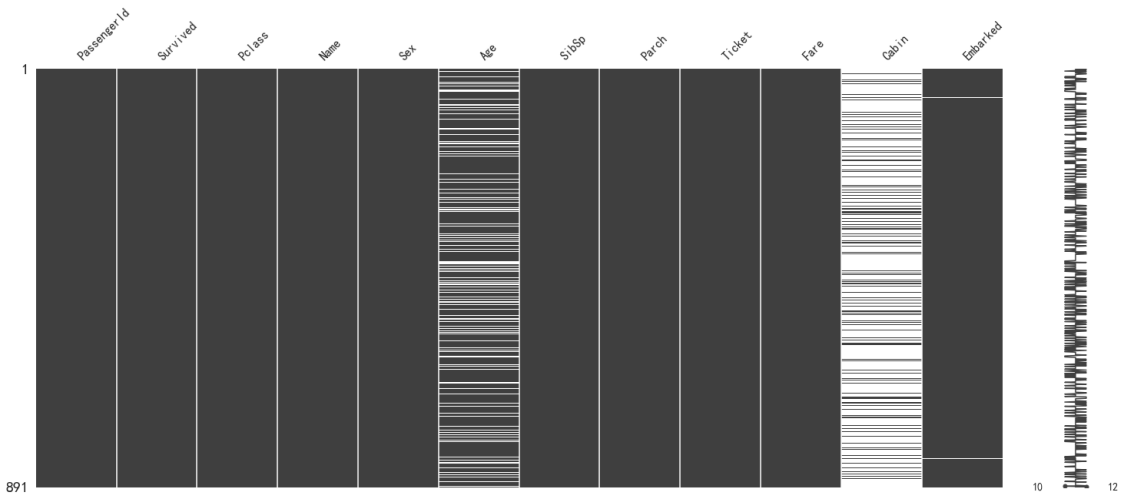
```
[ ]: train_data.isnull().sum()
```

```
[ ]: 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
```

missingno Cabin

```
[ ]: import missingno as msno
     msno.matrix(train_data)
```

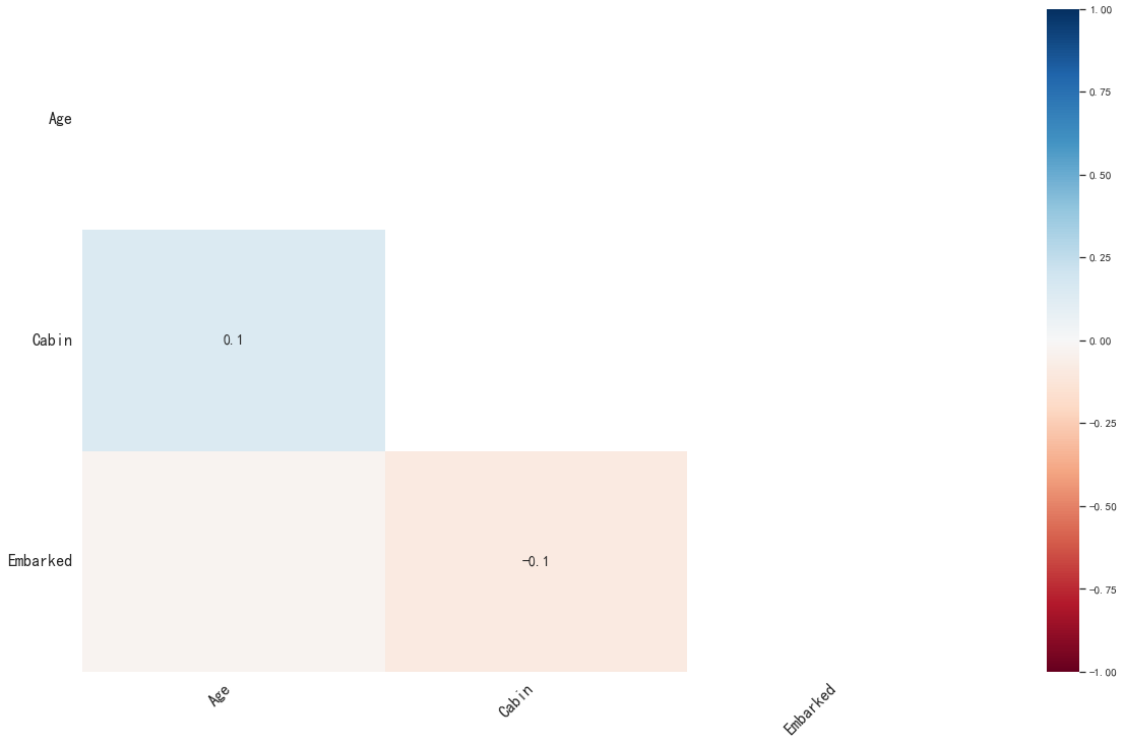
```
[ ]: <AxesSubplot:>
```



missingno heatmap

```
[ ]: msno.heatmap(train_data)
```

```
[ ]: <AxesSubplot:>
```

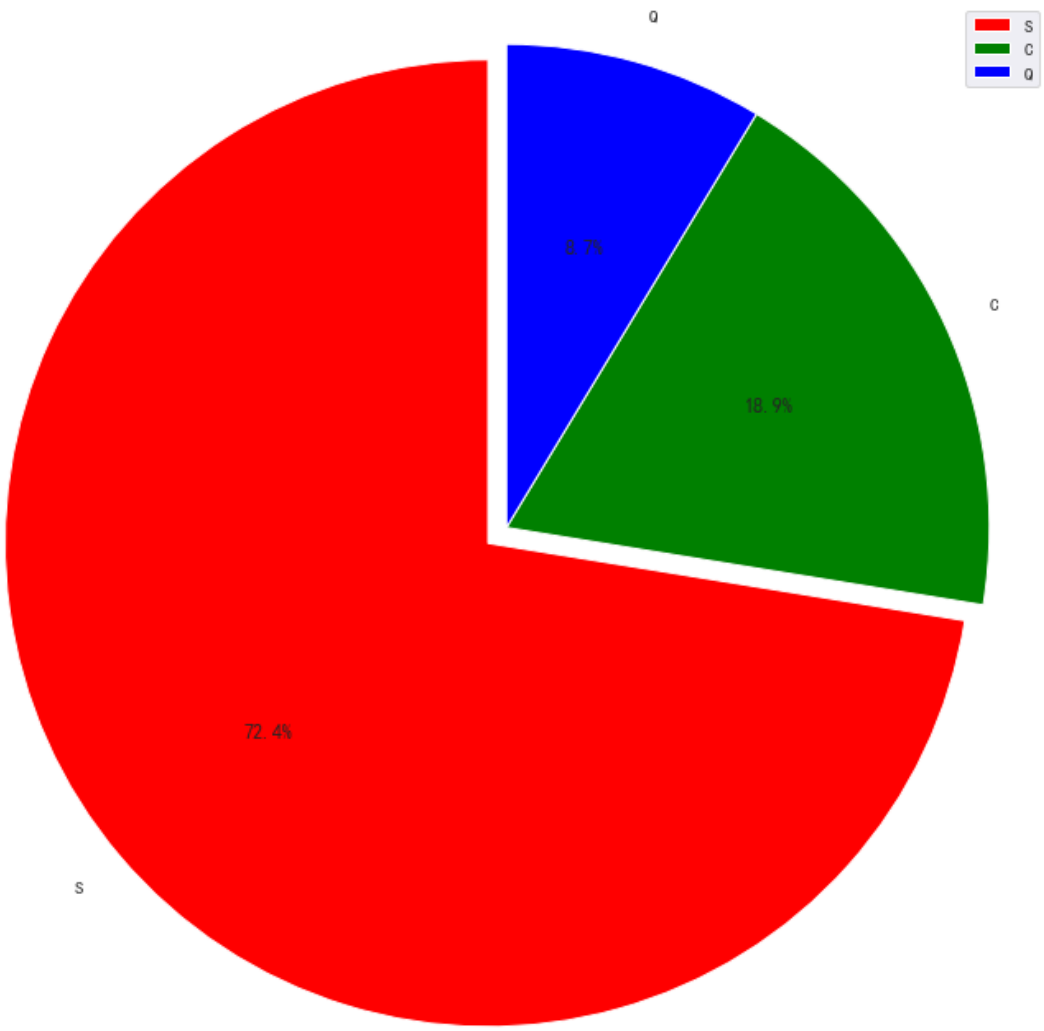


1.3

Age

Embarked( ) Embarked

[ ]:
```
import matplotlib.pyplot as plt
import matplotlib

matplotlib.rcParams['font.sans-serif'] = ['SimHei']
matplotlib.rcParams['axes.unicode_minus'] = False

label_list = ["S", "C", "Q"]
size = [train_data['Embarked'].value_counts()[0], train_data['Embarked'].
 ↪value_counts()[1], train_data['Embarked'].value_counts()[2]]
color = ["red", "green", "blue"]
explode = [0.05, 0, 0]

patches, l_text, p_text = plt.pie(size, explode=explode, colors=color,␣
 ↪labels=label_list, labeldistance=1.1, autopct="%1.1f%%", shadow=False,␣
 ↪startangle=90, pctdistance=0.6)
plt.axis("equal")
plt.legend()
plt.show()
```



Embarked S 'S'

```
[ ]: train_data['Embarked'].fillna('S', inplace = True)
```

Cabin

```
[ ]: train_data['Cabin'].fillna('U', inplace = True)
```

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

```
[ ]: PassengerId      0
     Survived         0
     Pclass           0
     Name             0
     Sex              0
```

```
Age            177
SibSp            0
Parch            0
Ticket           0
Fare             0
Cabin            0
Embarked         0
dtype: int64
```

1.4

scatter matrix( ) heatmap

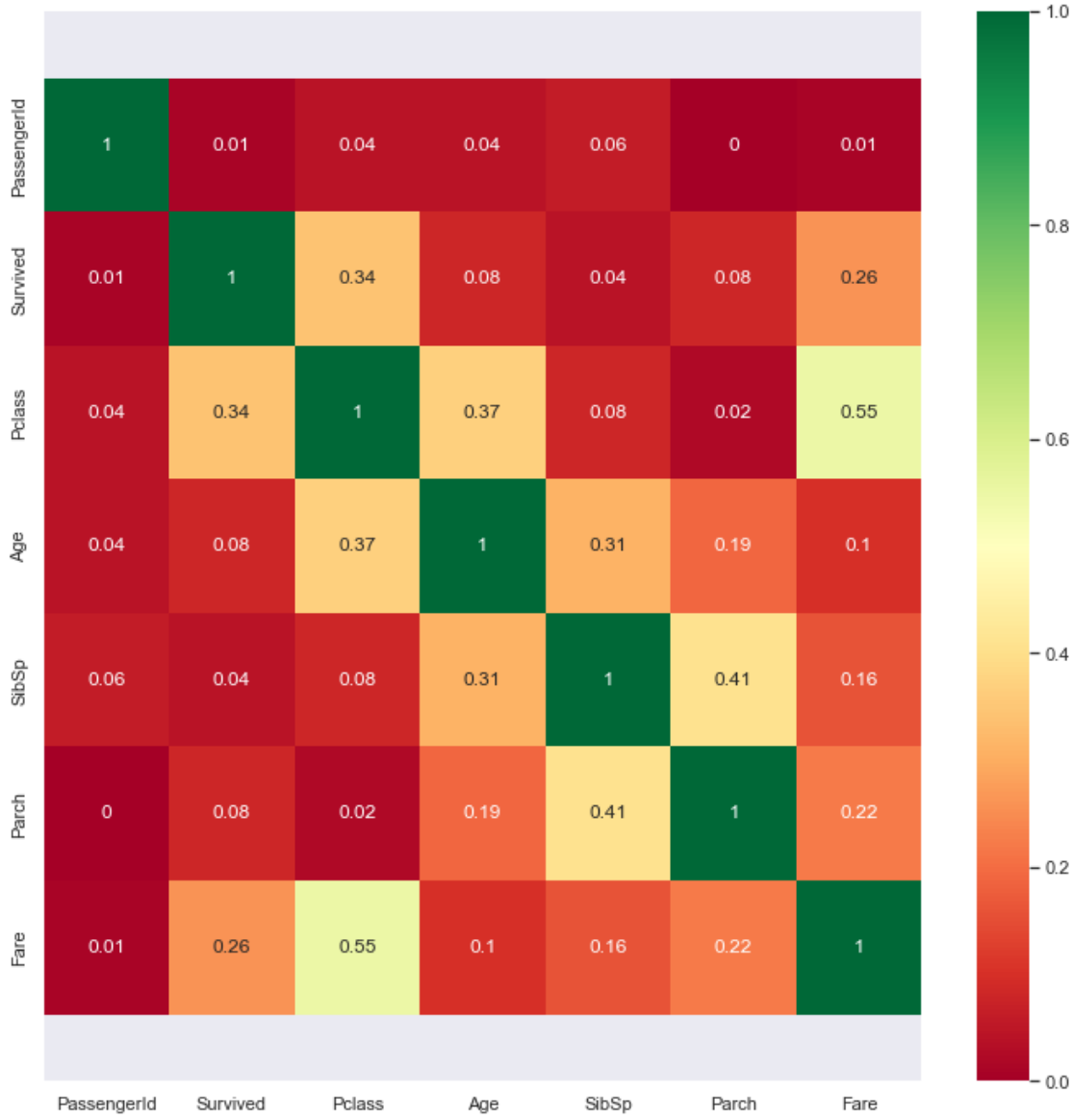
[ ]:
```
from pandas.plotting import scatter_matrix
scatter_matrix(train_data, figsize = (25, 25))
```

[ ]:
```
array([[<AxesSubplot:xlabel='PassengerId', ylabel='PassengerId'>,
        <AxesSubplot:xlabel='Survived', ylabel='PassengerId'>,
        <AxesSubplot:xlabel='Pclass', ylabel='PassengerId'>,
        <AxesSubplot:xlabel='Age', ylabel='PassengerId'>,
        <AxesSubplot:xlabel='SibSp', ylabel='PassengerId'>,
        <AxesSubplot:xlabel='Parch', ylabel='PassengerId'>,
        <AxesSubplot:xlabel='Fare', ylabel='PassengerId'>],
       [<AxesSubplot:xlabel='PassengerId', ylabel='Survived'>,
        <AxesSubplot:xlabel='Survived', ylabel='Survived'>,
        <AxesSubplot:xlabel='Pclass', ylabel='Survived'>,
        <AxesSubplot:xlabel='Age', ylabel='Survived'>,
        <AxesSubplot:xlabel='SibSp', ylabel='Survived'>,
        <AxesSubplot:xlabel='Parch', ylabel='Survived'>,
        <AxesSubplot:xlabel='Fare', ylabel='Survived'>],
       [<AxesSubplot:xlabel='PassengerId', ylabel='Pclass'>,
        <AxesSubplot:xlabel='Survived', ylabel='Pclass'>,
        <AxesSubplot:xlabel='Pclass', ylabel='Pclass'>,
        <AxesSubplot:xlabel='Age', ylabel='Pclass'>,
        <AxesSubplot:xlabel='SibSp', ylabel='Pclass'>,
        <AxesSubplot:xlabel='Parch', ylabel='Pclass'>,
        <AxesSubplot:xlabel='Fare', ylabel='Pclass'>],
       [<AxesSubplot:xlabel='PassengerId', ylabel='Age'>,
        <AxesSubplot:xlabel='Survived', ylabel='Age'>,
        <AxesSubplot:xlabel='Pclass', ylabel='Age'>,
        <AxesSubplot:xlabel='Age', ylabel='Age'>,
        <AxesSubplot:xlabel='SibSp', ylabel='Age'>,
        <AxesSubplot:xlabel='Parch', ylabel='Age'>,
        <AxesSubplot:xlabel='Fare', ylabel='Age'>],
       [<AxesSubplot:xlabel='PassengerId', ylabel='SibSp'>,
        <AxesSubplot:xlabel='Survived', ylabel='SibSp'>,
        <AxesSubplot:xlabel='Pclass', ylabel='SibSp'>,
```

```
 <AxesSubplot:xlabel='Age', ylabel='SibSp'>,
 <AxesSubplot:xlabel='SibSp', ylabel='SibSp'>,
 <AxesSubplot:xlabel='Parch', ylabel='SibSp'>,
 <AxesSubplot:xlabel='Fare', ylabel='SibSp'>],
[<AxesSubplot:xlabel='PassengerId', ylabel='Parch'>,
 <AxesSubplot:xlabel='Survived', ylabel='Parch'>,
 <AxesSubplot:xlabel='Pclass', ylabel='Parch'>,
 <AxesSubplot:xlabel='Age', ylabel='Parch'>,
 <AxesSubplot:xlabel='SibSp', ylabel='Parch'>,
 <AxesSubplot:xlabel='Parch', ylabel='Parch'>,
 <AxesSubplot:xlabel='Fare', ylabel='Parch'>],
[<AxesSubplot:xlabel='PassengerId', ylabel='Fare'>,
 <AxesSubplot:xlabel='Survived', ylabel='Fare'>,
 <AxesSubplot:xlabel='Pclass', ylabel='Fare'>,
 <AxesSubplot:xlabel='Age', ylabel='Fare'>,
 <AxesSubplot:xlabel='SibSp', ylabel='Fare'>,
 <AxesSubplot:xlabel='Parch', ylabel='Fare'>,
 <AxesSubplot:xlabel='Fare', ylabel='Fare'>]], dtype=object)
```

[ ]:
```
import seaborn as sns
correlation_matrix = np.absolute(train_data.corr().round(2))
sns.set(rc={'figure.figsize':(12, 12)})
ax = sns.heatmap(correlation_matrix, annot=True, cmap='RdYlGn')
bottom, top = ax.get_ylim()
ax.set_ylim(bottom + 0.5, top - 0.5)
```

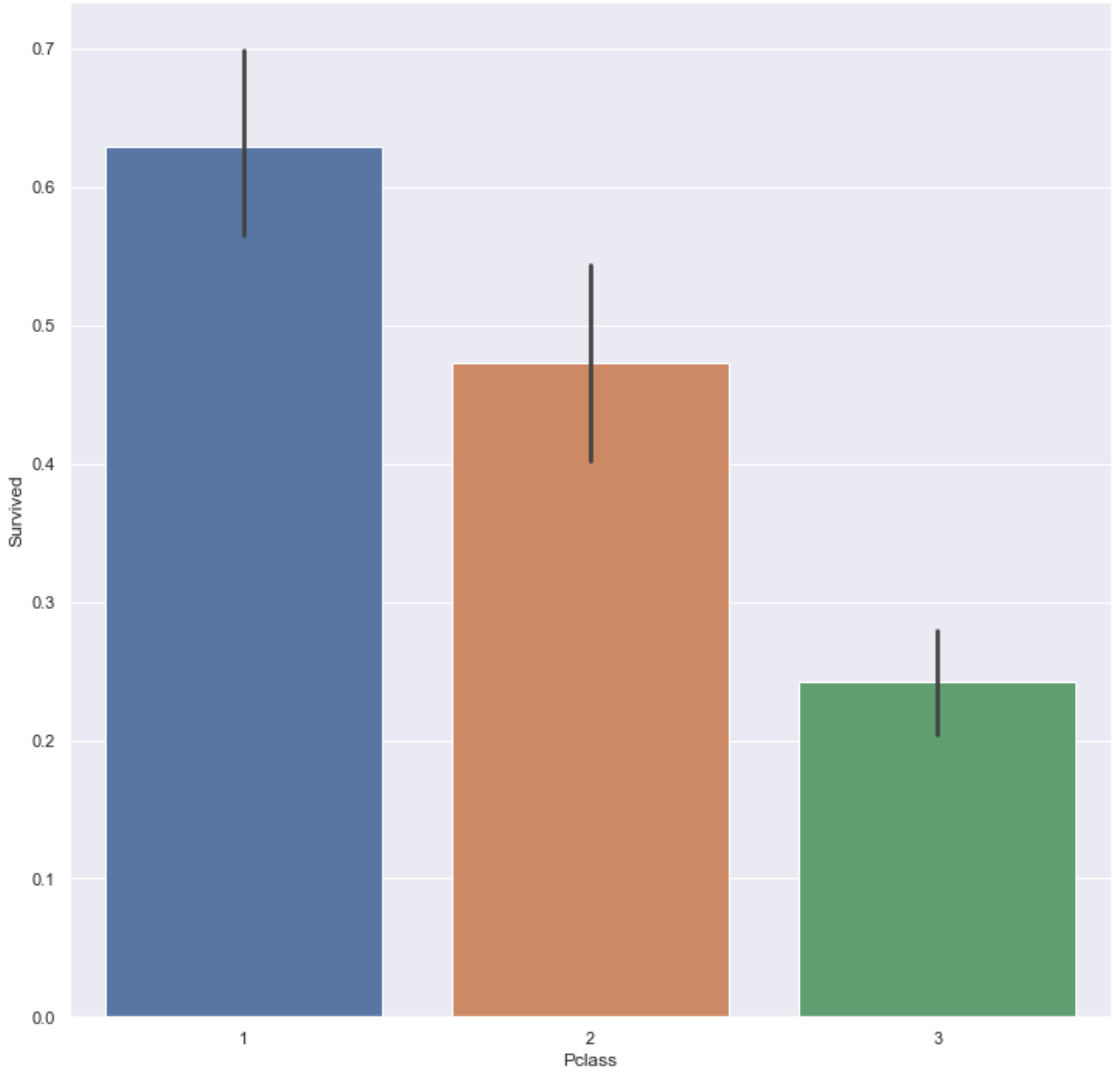
[ ]: (7.5, -0.5)



PClass( ) Fare( ) heatmap

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

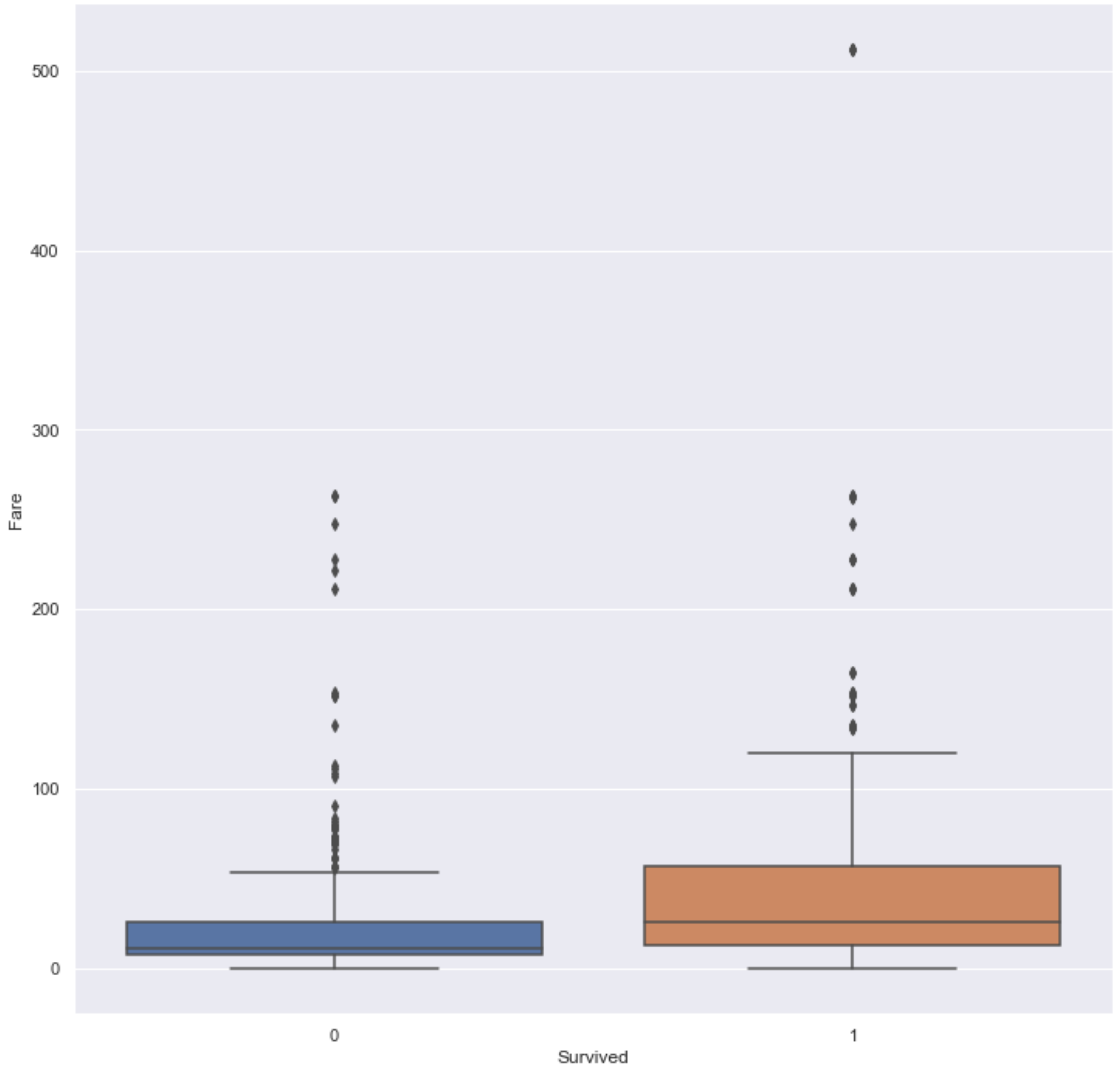
[ ]:
```
<AxesSubplot:xlabel='Pclass', ylabel='Survived'>
```



1 62%

[ ]:
```
sns.boxplot(x='Survived',y='Fare',data=train_data)
```

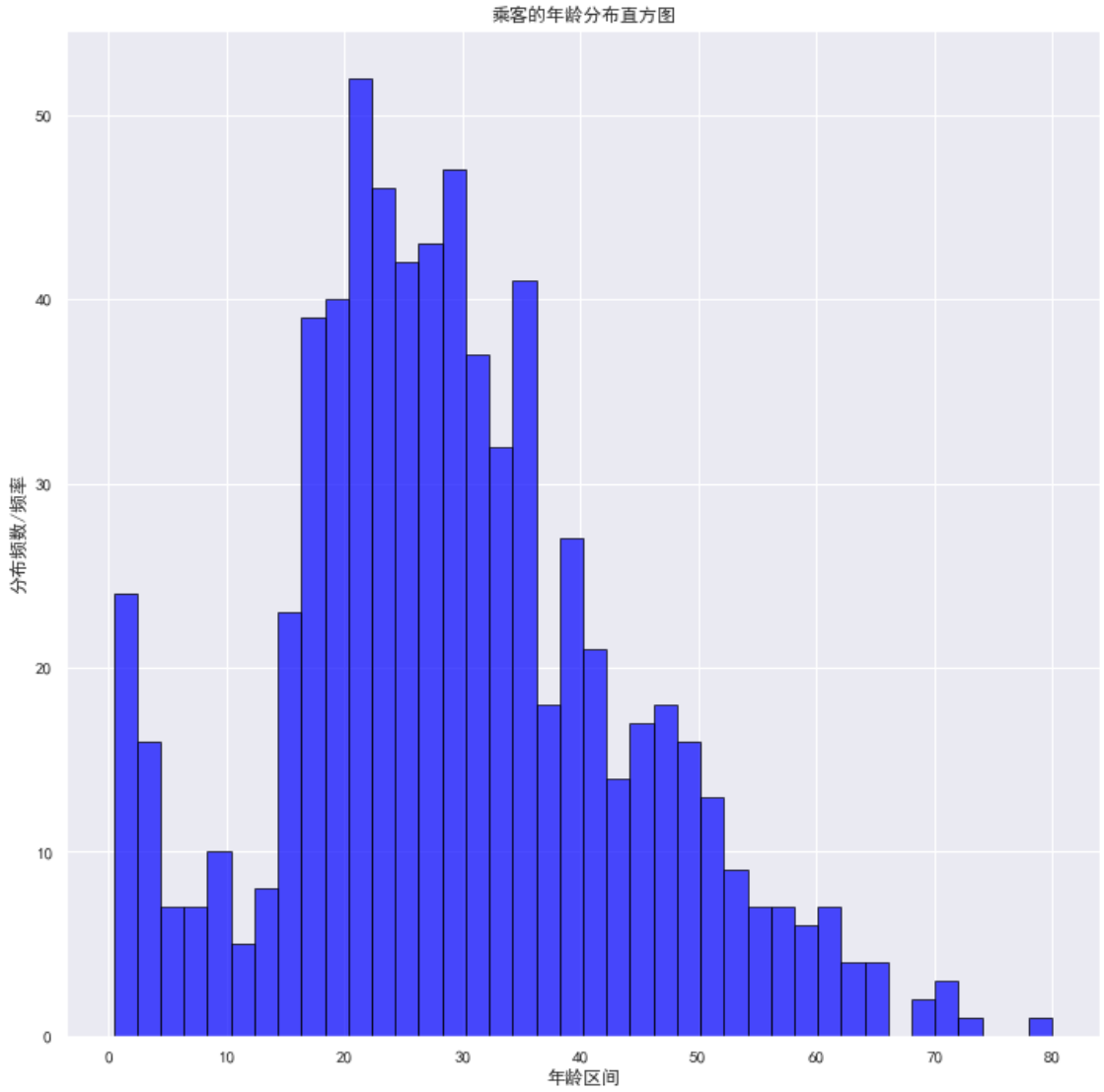
[ ]:
```
<AxesSubplot:xlabel='Survived', ylabel='Fare'>
```



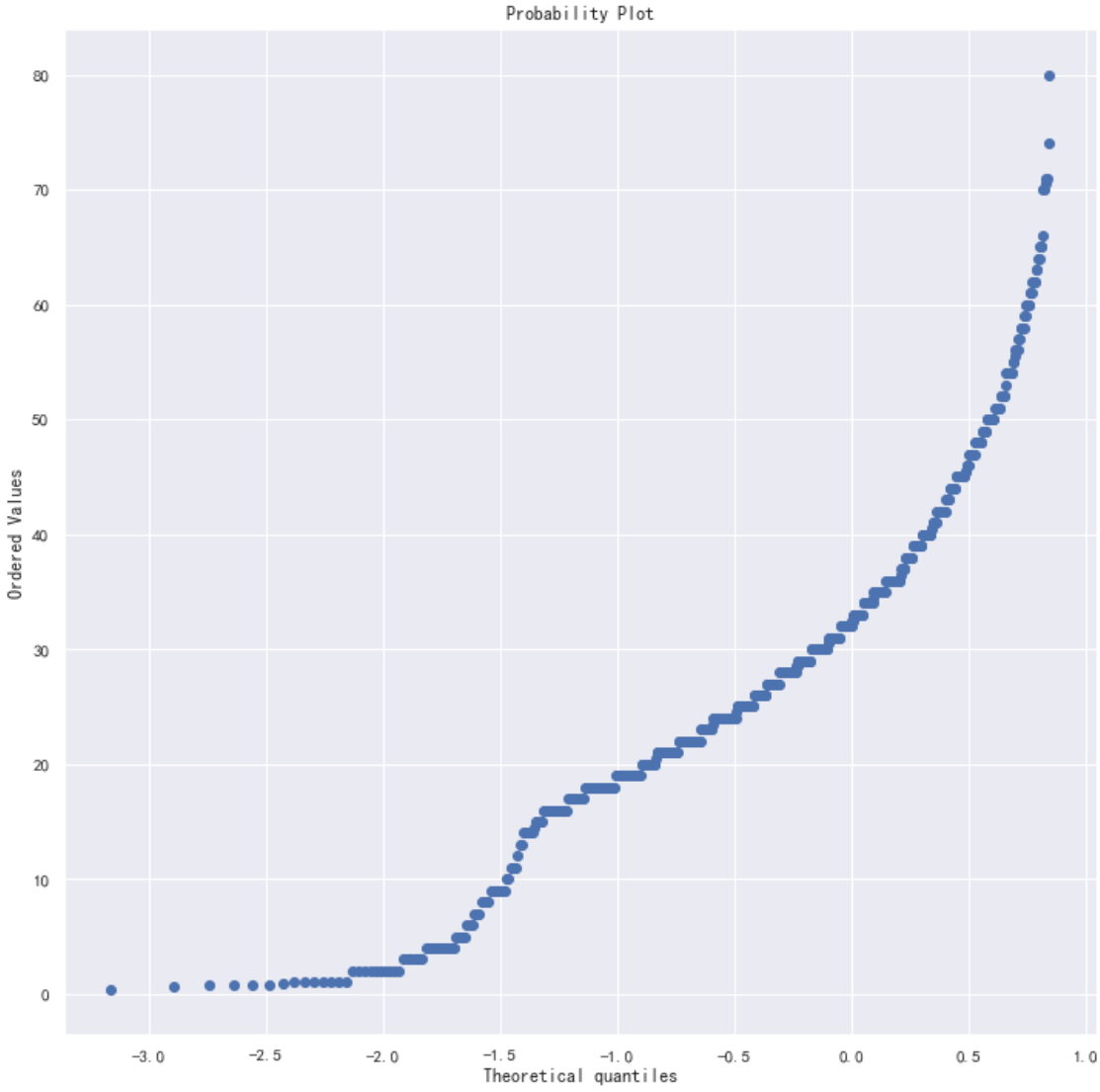
20-40 Q-Q

[ ]:
```
import matplotlib.pyplot as plt
import numpy as np
import matplotlib

matplotlib.rcParams['font.sans-serif'] = ['SimHei']
matplotlib.rcParams['axes.unicode_minus'] = False

plt.hist(train_data['Age'], bins = 40, facecolor = "blue", edgecolor = "black",␣
↪alpha = 0.7)
plt.xlabel("  ")
plt.ylabel("  / ")
```

```
plt.title("      ")
plt.show()
```



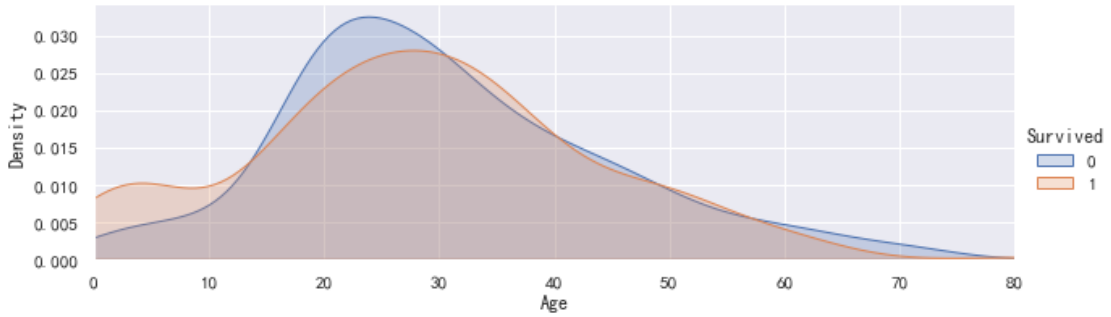
[ ]:
```
# Age's quantile-quantile plot
import scipy.stats as stats
stats.probplot(train_data['Age'], dist="norm", plot=plt)
plt.show()
```



FacetGrid 0-10

[ ]:
```
ageFacet = sns.FacetGrid(train_data,hue='Survived',aspect=3)
ageFacet.map(sns.kdeplot,'Age',shade=True)
ageFacet.set(xlim=(0,train_data['Age'].max()))
ageFacet.add_legend()
```

[ ]:
```
<seaborn.axisgrid.FacetGrid at 0x194529a5d00>
```



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

[ ]:
```
<AxesSubplot:xlabel='Sex', ylabel='Survived'>
```