

# Buổi 6. Chọn lựa đặc trưng

## 1 Thông tin chung

### Mục tiêu

- Giới thiệu thao tác chọn lựa thuộc tính/đặc trưng quan trọng của dữ liệu.
- Giới thiệu thuật toán Random Forest thường được dùng để chọn các thuộc tính quan trọng của dữ liệu.

### Kết quả đạt được

Sinh viên sau khi thực hành sẽ: - Nắm vững được các bước tiền xử lý dữ liệu và cài đặt được thuật toán Random Forest. - Cài đặt được các ví dụ thực hành

**Thời gian thực hành:** 3 tiết

**Công cụ thực hành:** Google Colab, Anaconda

## 2 Nội dung lý thuyết

### 2.1 Chọn đặc trưng quan trọng

#### Một số nhóm phương pháp phổ biến

- Phương pháp Filter.
- Phương pháp Wrapper.
- Phương pháp Embedded: Random Forest
- Phương pháp Hybrid.

### 2.2 Thuật toán Random Forest

**Random Forest:** là thuật toán được xây dựng từ một tập hợp nhiều cây quyết định. Mỗi cây quyết định được tạo ngẫu nhiên bằng cách chọn lại mẫu (bootstrap, random sampling) và chỉ dùng một phần nhỏ tập các thuộc tính ngẫu nhiên (random features) từ toàn bộ các thuộc tính trong dữ liệu.

## 3 Nội dung thực hành

### 3.1 Mô tả bài toán

Tiếp tục với tập dữ liệu vụ tai nạn chìm tàu Titanic.

---

PassengerId    Mã hành khách

Survived	Sống sót
Pclass	Hạng vé
Name	Tên
Sex	Giới tính
Age	Tuổi
SibSp	Số anh em/Vợ chồng
Parch	Số cha mẹ/con
Ticket	Số vé
Fare	Tiền vé
Cabin	Phòng
Embarked	Bến tàu

Embarked: (C = Cherbourg Q = Queenstown S = Southampton )

Hãy thực hiện các thao tác tiền xử lý dữ liệu.

### 3.2 Các thao tác tiền xử lý dữ liệu

- Đọc dữ liệu từ tập train và tập test.

```
[226]: import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns

from sklearn.ensemble import RandomForestClassifier
from sklearn.model_selection import KFold
from sklearn.model_selection import cross_val_score
from sklearn.metrics import confusion_matrix
```

```
[227]: train = pd.read_csv('titanic_train.csv')
test = pd.read_csv('titanic_test.csv')
```

```
[228]: train.head()
```

```
[228]: 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

```

```
[229]: test.head()
```

```

[229]:   PassengerId  Pclass                      Name  Sex \
0          892      3                Kelly, Mr. James  male
1          893      3      Wilkes, Mrs. James (Ellen Needs)  female
2          894      2                Myles, Mr. Thomas Francis  male
3          895      3                Wirz, Mr. Albert  male
4          896      3  Hirvonen, Mrs. Alexander (Helga E Lindqvist)  female

   Age  SibSp  Parch  Ticket    Fare Cabin Embarked
0  34.5     0     0  330911    7.8292   NaN        Q
1  47.0     1     0  363272    7.0000   NaN        S
2  62.0     0     0  240276    9.6875   NaN        Q
3  27.0     0     0  315154    8.6625   NaN        S
4  22.0     1     1  3101298   12.2875   NaN        S

```

```
[230]: train.info()
```

```

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 891 entries, 0 to 890
Data columns (total 12 columns):
PassengerId    891 non-null int64
Survived       891 non-null int64
Pclass         891 non-null int64
Name           891 non-null object
Sex            891 non-null object
Age           714 non-null float64
SibSp         891 non-null int64
Parch         891 non-null int64
Ticket        891 non-null object
Fare          891 non-null float64
Cabin         204 non-null object
Embarked      889 non-null object
dtypes: float64(2), int64(5), object(5)
memory usage: 83.7+ KB

```

```
[231]: test.info()
```

```

<class 'pandas.core.frame.DataFrame'>

```

```
RangeIndex: 418 entries, 0 to 417
Data columns (total 11 columns):
PassengerId    418 non-null int64
Pclass         418 non-null int64
Name           418 non-null object
Sex            418 non-null object
Age           332 non-null float64
SibSp          418 non-null int64
Parch          418 non-null int64
Ticket         418 non-null object
Fare           417 non-null float64
Cabin          91 non-null object
Embarked       418 non-null object
dtypes: float64(2), int64(4), object(5)
memory usage: 36.0+ KB
```

- Các thống kê cơ bản của những thuộc tính dữ liệu số.

```
[232]: train.describe()
```

```
[232]:
```

	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

```
[233]: test.describe()
```

```
[233]:
```

	PassengerId	Pclass	Age	SibSp	Parch	Fare
count	418.000000	418.000000	332.000000	418.000000	418.000000	417.000000
mean	1100.500000	2.265550	30.272590	0.447368	0.392344	35.627188
std	120.810458	0.841838	14.181209	0.896760	0.981429	55.907576
min	892.000000	1.000000	0.170000	0.000000	0.000000	0.000000
25%	996.250000	1.000000	21.000000	0.000000	0.000000	7.895800

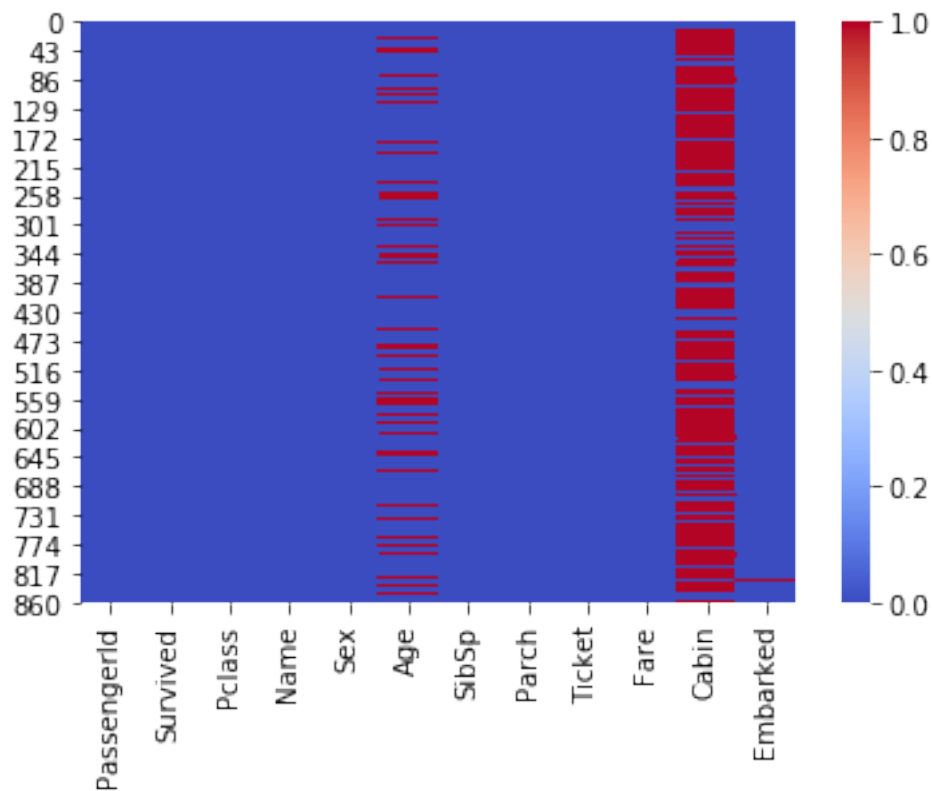
50%	1100.500000	3.000000	27.000000	0.000000	0.000000	14.454200
75%	1204.750000	3.000000	39.000000	1.000000	0.000000	31.500000
max	1309.000000	3.000000	76.000000	8.000000	9.000000	512.329200

### 3.2.1 Làm sạch dữ liệu

- Kiểm tra và xử lý các thuộc tính/đặc trưng bị lỗi, thiếu dữ liệu.

```
[234]: sns.heatmap(train.isnull(), cmap='coolwarm')
```

```
[234]: <matplotlib.axes._subplots.AxesSubplot at 0x2af2bf995c8>
```



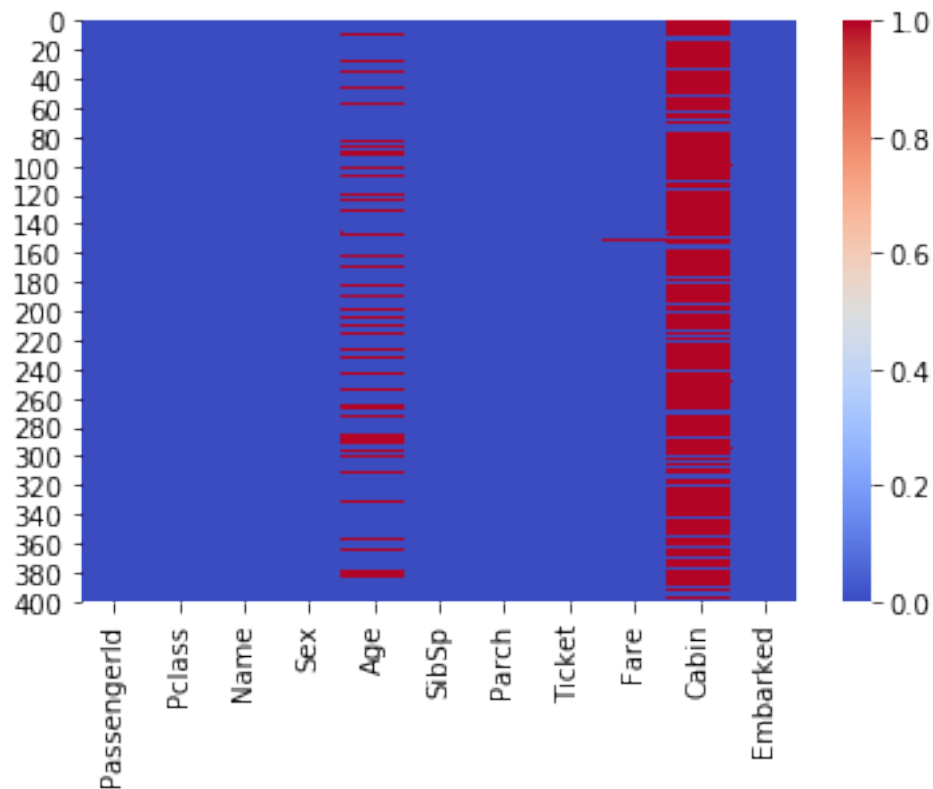
```
[235]: train.isnull().sum()
```

```
[235]: 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
```

```
[236]: sns.heatmap(test.isnull(), cmap='coolwarm')
```

```
[236]: <matplotlib.axes._subplots.AxesSubplot at 0x2af2da08e08>
```



```
[237]: test.isnull().sum()
```

```
[237]: PassengerId    0
Pclass           0
Name             0
Sex              0
Age             86
SibSp            0
Parch            0
Ticket           0
Fare             1
Cabin          327
```

```
Embarked      0
dtype: int64
```

**Nhận xét:** - Tập train có 3 đặc trưng có giá trị null là: Age, Cabin, Embarked. - Tập test có 2 đặc trưng có giá trị null là: Age, Fare, Cabin.

**Tạo một tập dữ liệu tổng hợp từ tập train và test**

```
[238]: train_test_data = [train, test]
       train_test_data
```

```
[238]: [   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],
      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]]

```

### 3.2.2 Kiểm tra tỷ lệ sống sót và gặp nạn của từng thuộc tính

- Cài đặt hàm vẽ biểu đồ cột với tham số là tên thuộc tính của dữ liệu

```
[239]: features = train.columns.to_list()
features
```

```
[239]: ['PassengerId',
'Survived',
'Pclass',
'Name',
'Sex',
'Age',
'SibSp',
'Parch',
'Ticket',
```



```
'Fare',
'Cabin',
'Embarked']
```

```
[240]: def bar_chart(features):
        survived = train[train['Survived'] == 1][features].value_counts()
        dead = train[train['Survived'] == 0][features].value_counts()
        df = pd.DataFrame([survived, dead])
        df.index = ['Survived', 'Dead']
        df.plot(kind = 'bar', stacked = True, figsize = (15, 5))
```

**Thuộc tính Title** - Thêm mới thuộc tính Title dựa vào cột Name: Mr, Mrs, Miss, ...

```
[241]: def add_title(train_test_data):
        for data in train_test_data:
            data['Title'] = data['Name'].str.extract(' ([A-Za-z]+)\.', expand =_
            →False)
```

```
[242]: add_title(train_test_data)
        train.head()
```

```
[242]: 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 Title
0      0   A/5 21171   7.2500   NaN      S    Mr
1      0   PC 17599  71.2833   C85      C   Mrs
2      0 STON/O2. 3101282   7.9250   NaN      S  Miss
3      0   113803  53.1000  C123      S   Mrs
4      0   373450   8.0500   NaN      S    Mr
```

```
[243]: print(train['Title'].count())
        print(train['Title'].value_counts())
```

```
891
Mr      517
Miss    182
```

```

Mrs      125
Master   40
Dr        7
Rev       6
Mlle     2
Major    2
Col      2
Ms       1
Sir      1
Countess 1
Lady     1
Mme      1
Capt    1
Jonkheer 1
Don      1
Name: Title, dtype: int64

```

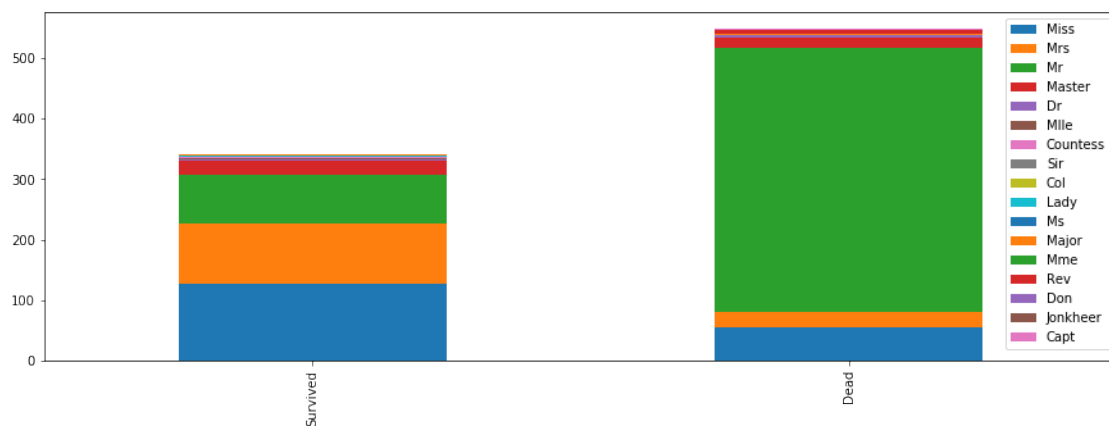
```
[244]: print(test['Title'].count())
       print(test['Title'].value_counts())
```

```

418
Mr      240
Miss    78
Mrs     72
Master  21
Col     2
Rev     2
Ms      1
Dona    1
Dr      1
Name: Title, dtype: int64

```

```
[245]: bar_chart('Title')
```



```
[246]: def transform_title(train_test_data):
        title_mapping = {"Mr": 0, "Miss": 1, "Mrs": 2,
                          "Master": 3, "Dr": 3, "Rev": 3, "Col": 3, "Major": 3,
                          "Mlle": 3, "Countess": 3,
                          "Ms": 3, "Lady": 3, "Jonkheer": 3, "Don": 3, "Dona" : 3,
                          "Mme": 3, "Capt": 3, "Sir": 3 }
        for data in train_test_data:
            data['Title'] = data['Title'].map(title_mapping)
```

```
[247]: transform_title(train_test_data)
train.head()
```

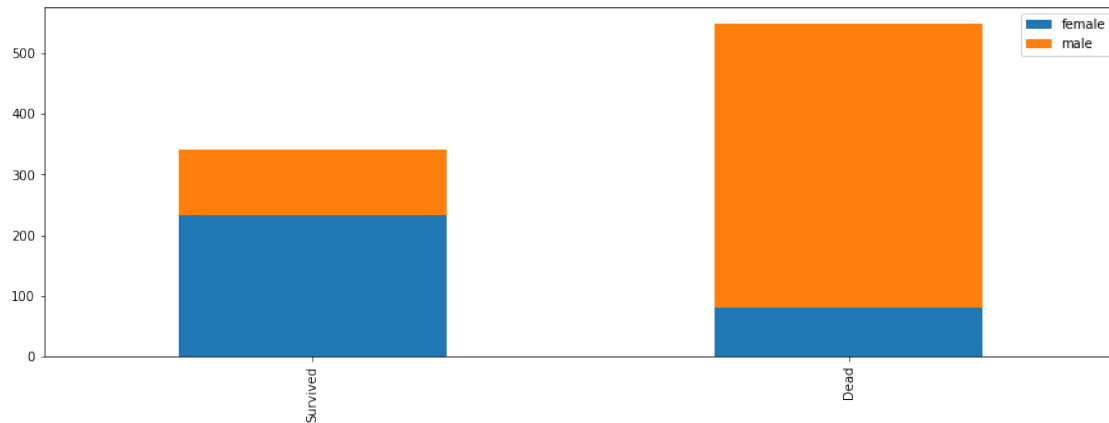
```
[247]: PassengerId  Survived  Pclass  \
0             1         0         3
1             2         1         1
2             3         1         3
3             4         1         1
4             5         0         3

                                                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  Title
0      0   A/5 21171   7.2500   NaN        S      0
1      0    PC 17599  71.2833   C85        C      2
2      0  STON/O2. 3101282   7.9250   NaN        S      1
3      0    113803  53.1000  C123        S      2
4      0    373450   8.0500   NaN        S      0
```

### Thuộc tính Sex

```
[248]: bar_chart('Sex')
```



```
[249]: def tranform_sex(train_test_data):
        sex_mapping = {'male': 0, 'female': 1}
        for data in train_test_data:
            data['Sex'] = data['Sex'].map(sex_mapping)
```

```
[250]: tranform_sex(train_test_data)
        train.head()
```

```
[250]: 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  Parch  \
0                                Braund, Mr. Owen Harris    0  22.0    1    0
1  Cumings, Mrs. John Bradley (Florence Briggs Th...    1  38.0    1    0
2                                Heikkinen, Miss. Laina    1  26.0    0    0
3  Futrelle, Mrs. Jacques Heath (Lily May Peel)    1  35.0    1    0
4                                Allen, Mr. William Henry    0  35.0    0    0
```

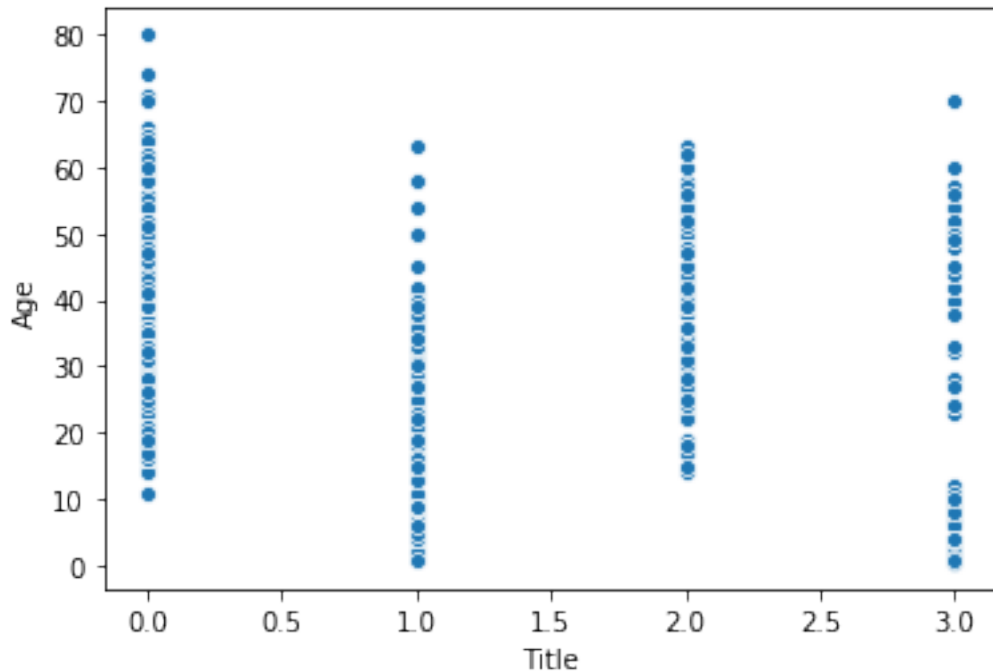
```

      Ticket     Fare Cabin Embarked  Title
0  A/5 21171   7.2500   NaN         S      0
1    PC 17599  71.2833   C85         C      2
2  STON/O2. 3101282   7.9250   NaN         S      1
3    113803  53.1000  C123         S      2
4    373450   8.0500   NaN         S      0
```

**Thuộc tính Age** - Dữ liệu thiếu có thể được điền theo từng nhóm của thuộc tính Title.

```
[251]: sns.scatterplot(x = train.Title, y = train.Age)
```

[251]: <matplotlib.axes.\_subplots.AxesSubplot at 0x2af2bd1bfc8>



```
[252]: def clean_age(train_test_data):
        for data in train_test_data:
            data['Age'].fillna(train.groupby('Title')['Age'].transform('median'),
                                inplace = True)
```

```
[253]: clean_age(train_test_data)

print(train['Age'].isnull().sum())
print(test['Age'].isnull().sum())
```

0  
0

```
[254]: def transform_age(train_test_data):
        for data in train_test_data:
            data.loc[data['Age'] <= 16, 'Age'] = 0,
            data.loc[(data['Age'] > 16) & (data['Age'] <= 26), 'Age'] = 1,
            data.loc[(data['Age'] > 26) & (data['Age'] <= 36), 'Age'] = 2,
            data.loc[(data['Age'] > 36) & (data['Age'] <= 62), 'Age'] = 3,
            data.loc[data['Age'] > 62, 'Age'] = 4
```

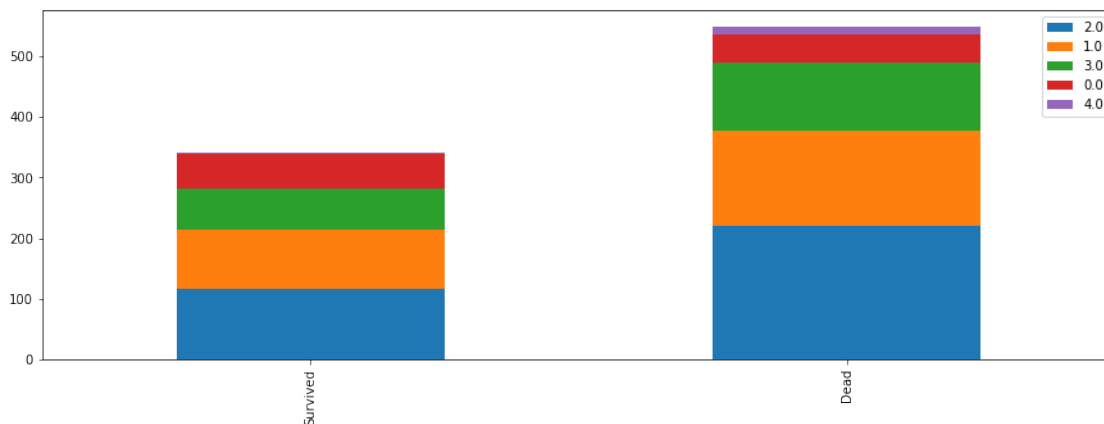
```
[255]: transform_age(train_test_data)
train.head()
```

```
[255]: 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  Parch  \
0                                Braund, Mr. Owen Harris    0  1.0    1     0
1  Cumings, Mrs. John Bradley (Florence Briggs Th...    1  3.0    1     0
2                                Heikkinen, Miss. Laina    1  1.0    0     0
3  Futrelle, Mrs. Jacques Heath (Lily May Peel)    1  2.0    1     0
4                                Allen, Mr. William Henry    0  2.0    0     0

      Ticket      Fare Cabin Embarked  Title
0  A/5 21171   7.2500   NaN         S      0
1  PC 17599  71.2833   C85         C      2
2  STON/O2. 3101282   7.9250   NaN         S      1
3    113803  53.1000  C123         S      2
4    373450   8.0500   NaN         S      0
```

```
[256]: bar_chart('Age')
```



### Thuộc tính Sibsp và Parch

- Thêm vào thuộc tính FamilySize dựa vào hai thuộc tính SibSp và Parch.

```
[257]: def add_family_size(train_test_data):
        for data in train_test_data:
            data['FamilySize'] = data['SibSp'] + data['Parch'] + 1
```

```
[258]: add_family_size(train_test_data)
        train.head()
```

```
[258]:
```

	PassengerId	Survived	Pclass	\
0	1	0	3	
1	2	1	1	
2	3	1	3	
3	4	1	1	
4	5	0	3	

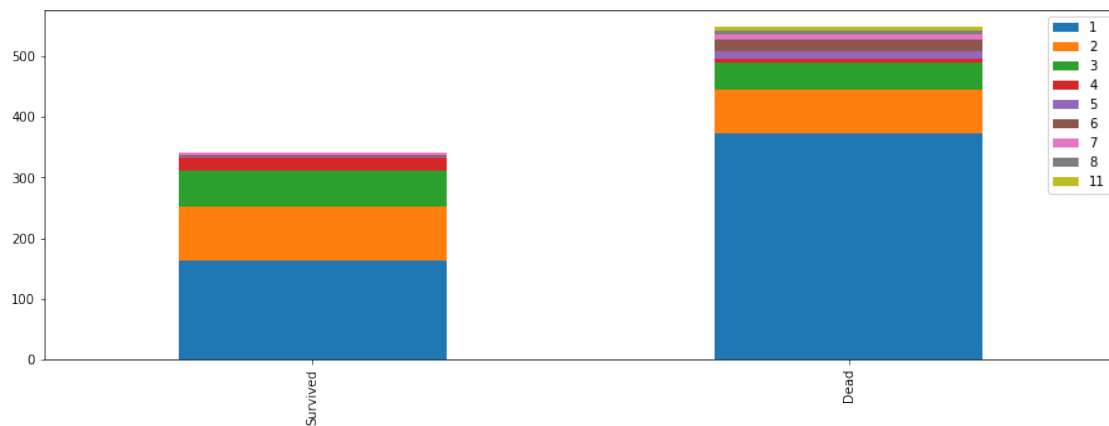
  

	Name	Sex	Age	SibSp	Parch	\
0	Braund, Mr. Owen Harris	0	1.0	1	0	
1	Cumings, Mrs. John Bradley (Florence Briggs Th...	1	3.0	1	0	
2	Heikkinen, Miss. Laina	1	1.0	0	0	
3	Futrelle, Mrs. Jacques Heath (Lily May Peel)	1	2.0	1	0	
4	Allen, Mr. William Henry	0	2.0	0	0	

	Ticket	Fare	Cabin	Embarked	Title	FamilySize
0	A/5 21171	7.2500	NaN	S	0	2
1	PC 17599	71.2833	C85	C	2	2
2	STON/O2. 3101282	7.9250	NaN	S	1	1
3	113803	53.1000	C123	S	2	2
4	373450	8.0500	NaN	S	0	1

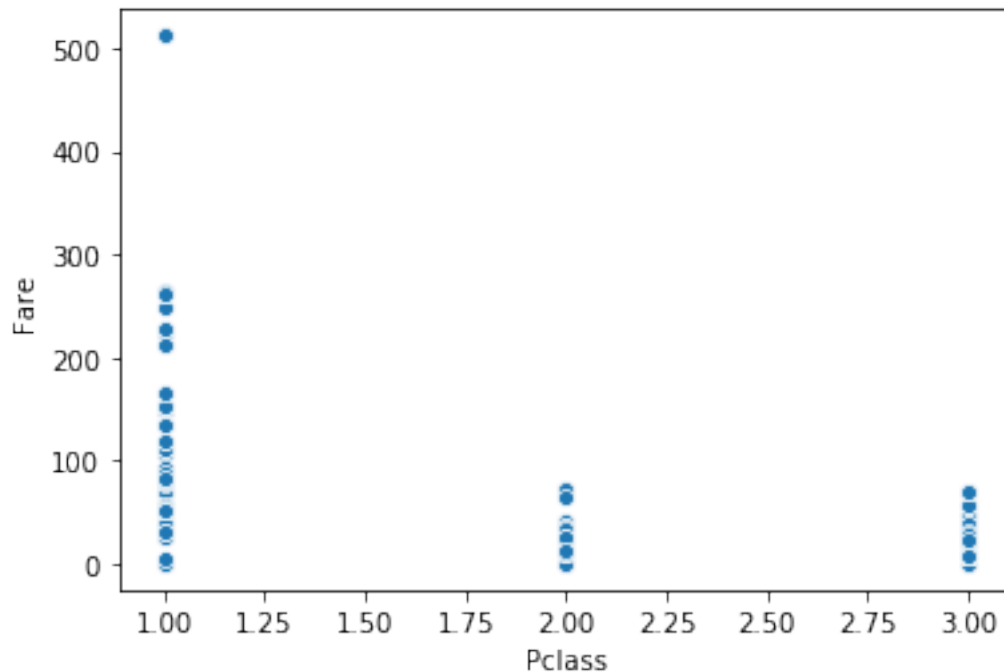
```
[259]: bar_chart('FamilySize')
```



### Thuộc tính Fare

```
[260]: sns.scatterplot(x = train.Pclass, y = train.Fare)
```

```
[260]: <matplotlib.axes._subplots.AxesSubplot at 0x2af2b3f7388>
```



```
[261]: def clean_fare(train_test_data):
        for data in train_test_data:
            data['Fare'].fillna(train.groupby('Pclass')['Fare'].
→transform('median'), inplace = True)
```

```
[262]: clean_fare(train_test_data)

print(train['Fare'].isnull().sum())
print(test['Fare'].isnull().sum())
```

0  
0

```
[263]: def tranform_fare(train_test_data):
        for data in train_test_data:
            data.loc[data['Fare'] <= 17, 'Fare'] = 0,
            data.loc[(data['Fare'] > 17) & (data['Fare'] <= 30), 'Fare'] = 1,
            data.loc[(data['Fare'] > 30) & (data['Fare'] <= 100), 'Fare'] = 2,
            data.loc[data['Fare'] > 100, 'Fare'] = 3
```

```
[264]: tranform_fare(train_test_data)
train.head()
```

```
[264]: PassengerId  Survived  Pclass  \
0               1         0       3
```

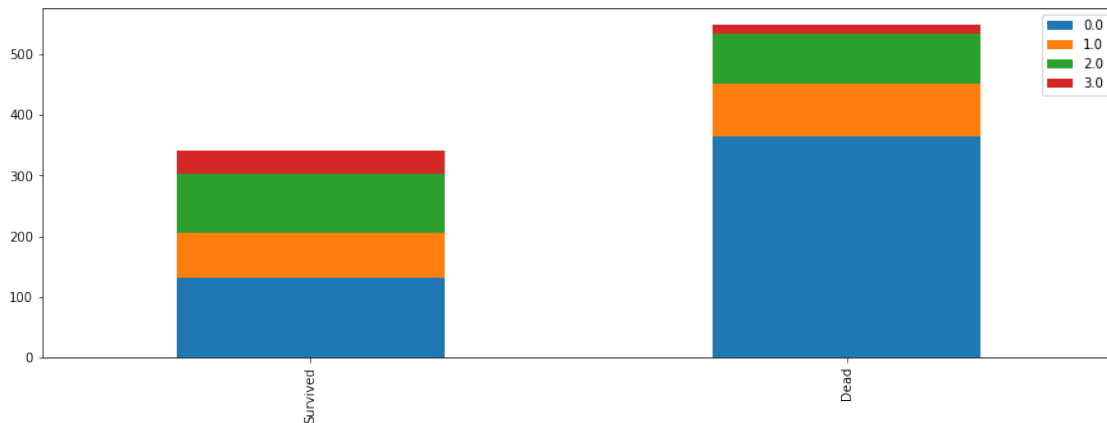


```
1      2      1      1
2      3      1      3
3      4      1      1
4      5      0      3
```

	Name	Sex	Age	SibSp	Parch	\
0	Braund, Mr. Owen Harris	0	1.0	1	0	
1	Cumings, Mrs. John Bradley (Florence Briggs Th...	1	3.0	1	0	
2	Heikkinen, Miss. Laina	1	1.0	0	0	
3	Futrelle, Mrs. Jacques Heath (Lily May Peel)	1	2.0	1	0	
4	Allen, Mr. William Henry	0	2.0	0	0	

	Ticket	Fare	Cabin	Embarked	Title	FamilySize
0	A/5 21171	0.0	NaN	S	0	2
1	PC 17599	2.0	C85	C	2	2
2	STON/O2. 3101282	0.0	NaN	S	1	1
3	113803	2.0	C123	S	2	2
4	373450	0.0	NaN	S	0	1

```
[265]: bar_chart('Fare')
```



### Thuộc tính Cabin

```
[266]: train['Cabin'].value_counts()
```

```
[266]: C23 C25 C27      4
        G6              4
        B96 B98         4
        D               3
        E101            3
        ..
        B71             1
```

```
T          1
C101       1
E77        1
C70        1
Name: Cabin, Length: 147, dtype: int64
```

```
[267]: test['Cabin'].value_counts()
```

```
[267]: B57 B59 B63 B66      3
      B45                2
      C116               2
      C55 C57            2
      F4                 2
      ..
      D                  1
      D38                1
      C86                1
      C51                1
      C97                1
Name: Cabin, Length: 76, dtype: int64
```

```
[268]: def get_class_cabin(train_test_data):
      for data in train_test_data:
          data['Cabin'] = data['Cabin'].str[:1]
```

```
[269]: get_class_cabin(train_test_data)
      train.head()
```

```
[269]: PassengerId  Survived  Pclass  \
0             1         0         3
1             2         1         1
2             3         1         3
3             4         1         1
4             5         0         3
```

```

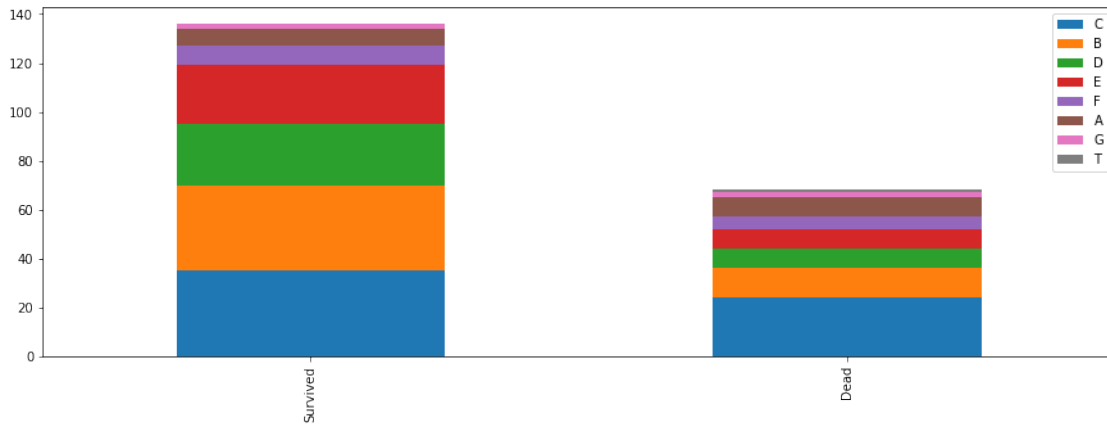
                                Name  Sex  Age  SibSp  Parch  \
0                        Braund, Mr. Owen Harris    0  1.0    1    0
1  Cumings, Mrs. John Bradley (Florence Briggs Th...    1  3.0    1    0
2                        Heikkinen, Miss. Laina    1  1.0    0    0
3  Futrelle, Mrs. Jacques Heath (Lily May Peel)    1  2.0    1    0
4                        Allen, Mr. William Henry    0  2.0    0    0
```

```

      Ticket  Fare  Cabin  Embarked  Title  FamilySize
0      A/5 21171   0.0   NaN        S      0          2
1      PC 17599   2.0    C        C      2          2
2  STON/O2. 3101282   0.0   NaN        S      1          1
3      113803   2.0    C        S      2          2
```

4                      373450      0.0      NaN              S              0                      1

[270]: `bar_chart('Cabin')`



```
[271]: def transform_cabin(train_test_data):
        cabin_mapping = {"A": 0, "B": 0.4, "C": 0.8, "D": 1.2, "E": 1.6, "F": 2.0,
        → "G": 2.4, "T": 2.8}
        for data in train_test_data:
            data['Cabin'] = data['Cabin'].map(cabin_mapping)
```

```
[272]: transform_cabin(train_test_data)
train.head()
```

```
[272]: 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  Parch  \
0                Braund, Mr. Owen Harris    0  1.0    1    0
1  Cumings, Mrs. John Bradley (Florence Briggs Th...    1  3.0    1    0
2                Heikkinen, Miss. Laina    1  1.0    0    0
3  Futrelle, Mrs. Jacques Heath (Lily May Peel)    1  2.0    1    0
4                Allen, Mr. William Henry    0  2.0    0    0
```

```

      Ticket  Fare  Cabin Embarked  Title  FamilySize
0    A/5 21171   0.0   NaN        S      0          2
1    PC 17599   2.0   0.8        C      2          2
2  STON/O2. 3101282   0.0   NaN        S      1          1
3    113803   2.0   0.8        S      2          2
```

```
4          373450    0.0    NaN    S    0    1
```

```
[273]: def clean_cabin(train_test_data):
        for data in train_test_data:
            data['Cabin'].fillna(train.groupby('Pclass')['Cabin'].
→transform('median'), inplace = True)
```

```
[274]: clean_cabin(train_test_data)

print(train['Cabin'].isnull().sum())
print(test['Cabin'].isnull().sum())
```

```
0
```

```
0
```

### Thuộc tính Embraked

```
[275]: def clean_embarked(train_test_data):
        for data in train_test_data:
            data['Embarked'] = data['Embarked'].fillna('S')
```

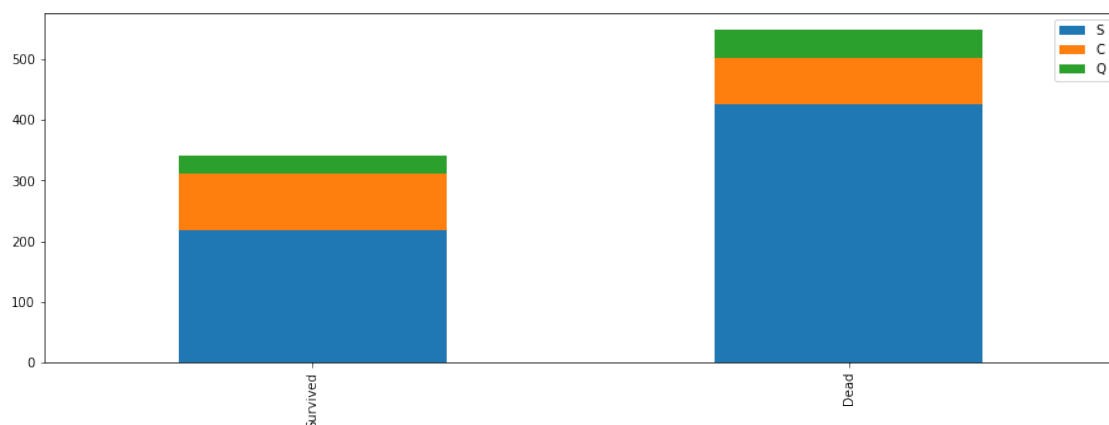
```
[276]: clean_embarked(train_test_data)

print(train['Embarked'].isnull().sum())
print(test['Embarked'].isnull().sum())
```

```
0
```

```
0
```

```
[277]: bar_chart('Embarked')
```



```
[278]: def tranform_embarked(train_test_data):
        embarked_mapping = {"S": 0, "C": 1, "Q": 2}
```

```
for data in train_test_data:
    data['Embarked'] = data['Embarked'].map(embarked_mapping)
```

```
[279]: transform_embarked(train_test_data)
train.head()
```

```
[279]:
```

	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	Parch	\
0	Braund, Mr. Owen Harris	0	1.0	1	0	
1	Cumings, Mrs. John Bradley (Florence Briggs Th...	1	3.0	1	0	
2	Heikkinen, Miss. Laina	1	1.0	0	0	
3	Futrelle, Mrs. Jacques Heath (Lily May Peel)	1	2.0	1	0	
4	Allen, Mr. William Henry	0	2.0	0	0	

	Ticket	Fare	Cabin	Embarked	Title	FamilySize
0	A/5 21171	0.0	2.0	0	0	2
1	PC 17599	2.0	0.8	1	2	2
2	STON/O2. 3101282	0.0	2.0	0	1	1
3	113803	2.0	0.8	0	2	2
4	373450	0.0	2.0	0	0	1

### 3.3 Chọn thuộc tính quan trọng

#### 3.3.1 Kiểm tra độ tương quan giữa các thuộc tính

```
[280]: train.isnull().sum()
```

```
[280]: PassengerId    0
Survived          0
Pclass            0
Name              0
Sex               0
Age               0
SibSp             0
Parch            0
Ticket            0
Fare              0
Cabin             0
Embarked          0
Title             0
FamilySize        0
```

```
dtype: int64
```

```
[281]: test.isnull().sum()
```

```
[281]: PassengerId    0
      Pclass        0
      Name          0
      Sex           0
      Age           0
      SibSp         0
      Parch         0
      Ticket        0
      Fare          0
      Cabin         0
      Embarked      0
      Title         0
      FamilySize    0
      dtype: int64
```

```
[282]: train.head()
```

```
[282]:   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  Parch  \
0   Braund, Mr. Owen Harris    0  1.0    1    0
1  Cumings, Mrs. John Bradley (Florence Briggs Th...    1  3.0    1    0
2   Heikkinen, Miss. Laina    1  1.0    0    0
3  Futrelle, Mrs. Jacques Heath (Lily May Peel)    1  2.0    1    0
4   Allen, Mr. William Henry    0  2.0    0    0

      Ticket  Fare  Cabin  Embarked  Title  FamilySize
0    A/5 21171   0.0    2.0         0     0           2
1    PC 17599   2.0    0.8         1     2           2
2  STON/O2. 3101282   0.0    2.0         0     1           1
3    113803   2.0    0.8         0     2           2
4    373450   0.0    2.0         0     0           1
```

```
[283]: test.head()
```

```
[283]:   PassengerId  Pclass  Name  Sex  \
0         892       3   Kelly, Mr. James    0
1         893       3  Wilkes, Mrs. James (Ellen Needs)    1
```

2	894	2	Myles, Mr. Thomas Francis	0
3	895	3	Wirz, Mr. Albert	0
4	896	3	Hirvonen, Mrs. Alexander (Helga E Lindqvist)	1

	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked	Title	FamilySize
0	2.0	0	0	330911	0.0	2.0	2	0	1
1	3.0	1	0	363272	0.0	0.8	0	2	2
2	3.0	0	0	240276	0.0	2.0	2	0	1
3	2.0	0	0	315154	0.0	0.8	0	0	1
4	1.0	1	1	3101298	0.0	2.0	0	2	3

```
[284]: corr = train.corr()
corr
```

```
[284]:
```

	PassengerId	Survived	Pclass	Sex	Age	SibSp	\
PassengerId	1.000000	-0.005007	-0.035144	-0.042939	0.035896	-0.057527	
Survived	-0.005007	1.000000	-0.338481	0.543351	-0.090499	-0.035322	
Pclass	-0.035144	-0.338481	1.000000	-0.131900	-0.314046	0.083081	
Sex	-0.042939	0.543351	-0.131900	1.000000	-0.130828	0.114631	
Age	0.035896	-0.090499	-0.314046	-0.130828	1.000000	-0.268832	
SibSp	-0.057527	-0.035322	0.083081	0.114631	-0.268832	1.000000	
Parch	-0.001652	0.081629	0.018443	0.245489	-0.171533	0.414838	
Fare	0.004665	0.292884	-0.693736	0.216932	0.071564	0.341187	
Cabin	-0.037406	-0.279477	0.864697	-0.093314	-0.265643	0.072361	
Embarked	-0.030467	0.106811	0.045702	0.116569	-0.036063	-0.059961	
Title	-0.002555	0.445792	-0.164205	0.557890	-0.171512	0.287060	
FamilySize	-0.040143	0.016639	0.065997	0.200988	-0.269433	0.890712	

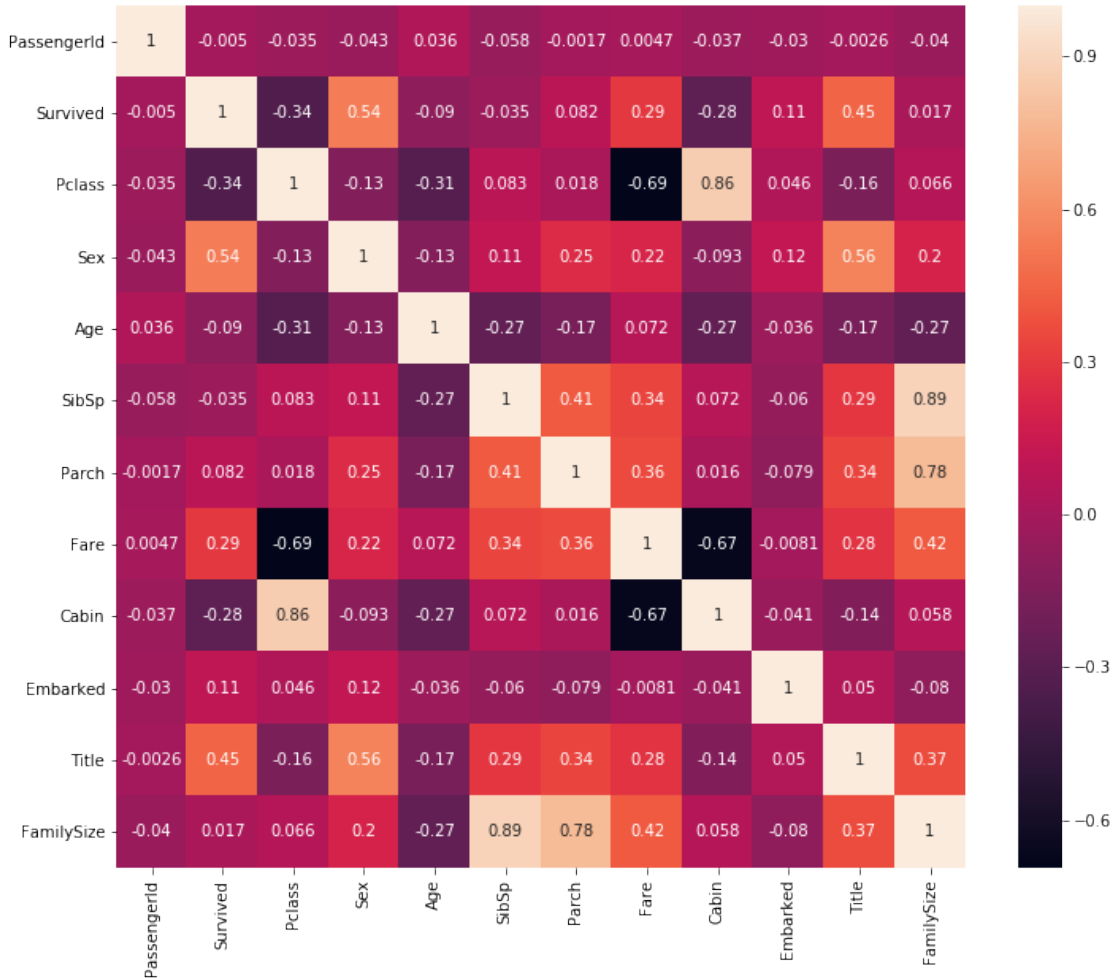
	Parch	Fare	Cabin	Embarked	Title	FamilySize
PassengerId	-0.001652	0.004665	-0.037406	-0.030467	-0.002555	-0.040143
Survived	0.081629	0.292884	-0.279477	0.106811	0.445792	0.016639
Pclass	0.018443	-0.693736	0.864697	0.045702	-0.164205	0.065997
Sex	0.245489	0.216932	-0.093314	0.116569	0.557890	0.200988
Age	-0.171533	0.071564	-0.265643	-0.036063	-0.171512	-0.269433
SibSp	0.414838	0.341187	0.072361	-0.059961	0.287060	0.890712
Parch	1.000000	0.364097	0.016359	-0.078665	0.341660	0.783111
Fare	0.364097	1.000000	-0.668376	-0.008131	0.278740	0.415087
Cabin	0.016359	-0.668376	1.000000	-0.041238	-0.141768	0.057629
Embarked	-0.078665	-0.008131	-0.041238	1.000000	0.049699	-0.080281
Title	0.341660	0.278740	-0.141768	0.049699	1.000000	0.366883
FamilySize	0.783111	0.415087	0.057629	-0.080281	0.366883	1.000000

```
[285]: plt.figure(figsize = (12, 10))
ax = sns.heatmap(corr, annot = True)

## Đoạn lệnh sửa lỗi mất dòng đầu, dòng cuối ở phiên bản matplotlib 3.1.1
bottom, top = ax.get_ylim()
```

```
ax.set_ylim(bottom + 0.5, top - 0.5)
##

plt.show()
```



- Lấy trị tuyệt đối độ tương quan của các thuộc tính.

```
[286]: corr_features = abs(corr['Survived'])
```

- Lấy các thuộc tính có độ tương quan > 0.1

```
[287]: relevant_features = corr_features[corr_features > 0.1]
relevant_features
```

```
[287]: Survived    1.000000
Pclass         0.338481
Sex            0.543351
```



```
Fare          0.292884
Cabin         0.279477
Embarked      0.106811
Title         0.445792
Name: Survived, dtype: float64
```

```
[288]: features = train.columns.tolist()
       features
```

```
[288]: ['PassengerId',
       'Survived',
       'Pclass',
       'Name',
       'Sex',
       'Age',
       'SibSp',
       'Parch',
       'Ticket',
       'Fare',
       'Cabin',
       'Embarked',
       'Title',
       'FamilySize']
```

```
[289]: features.remove('PassengerId')
       features.remove('Survived')
       features.remove('Name')
       features.remove('Ticket')
```

```
[290]: features
```

```
[290]: ['Pclass',
       'Sex',
       'Age',
       'SibSp',
       'Parch',
       'Fare',
       'Cabin',
       'Embarked',
       'Title',
       'FamilySize']
```

### 3.3.2 Thuật toán Random Forest

```
[291]: rf = RandomForestClassifier(n_estimators = 10)
```

```
[292]: target = train['Survived']
       rf.fit(train[features], target)
```

```
[292]: RandomForestClassifier(bootstrap=True, class_weight=None, criterion='gini',
                             max_depth=None, max_features='auto', max_leaf_nodes=None,
                             min_impurity_decrease=0.0, min_impurity_split=None,
                             min_samples_leaf=1, min_samples_split=2,
                             min_weight_fraction_leaf=0.0, n_estimators=10,
                             n_jobs=None, oob_score=False, random_state=None,
                             verbose=0, warm_start=False)
```

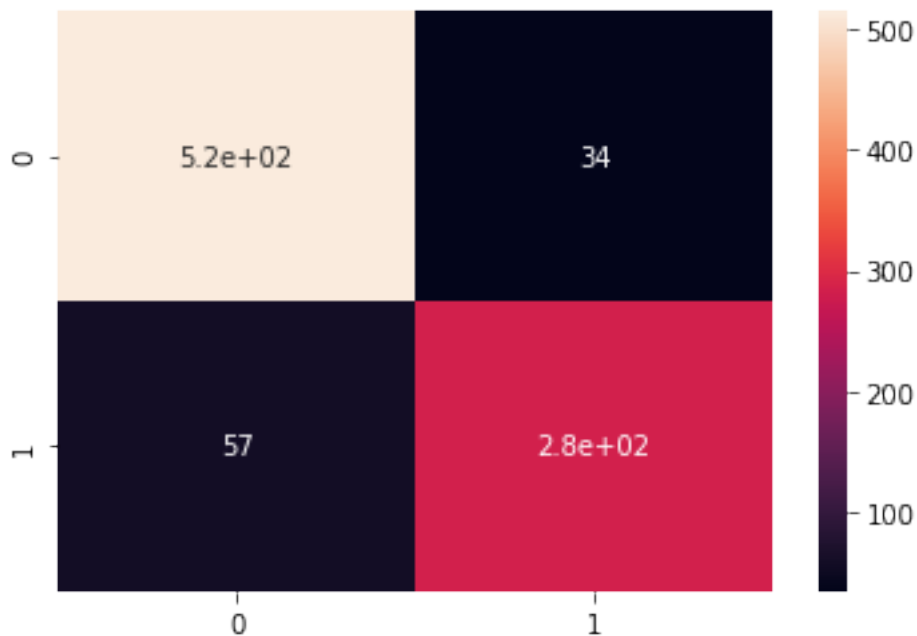
```
[293]: prediction = rf.predict(test[features])
```

```
[294]: confusion = confusion_matrix(train['Survived'], rf.predict(train[features]))

ax = sns.heatmap(confusion, annot = True)

## Đoạn lệnh sửa lỗi mất dòng đầu, dòng cuối ở phiên bản matplotlib 3.1.1
bottom, top = ax.get_ylim()
ax.set_ylim(bottom + 0.5, top - 0.5)
##
```

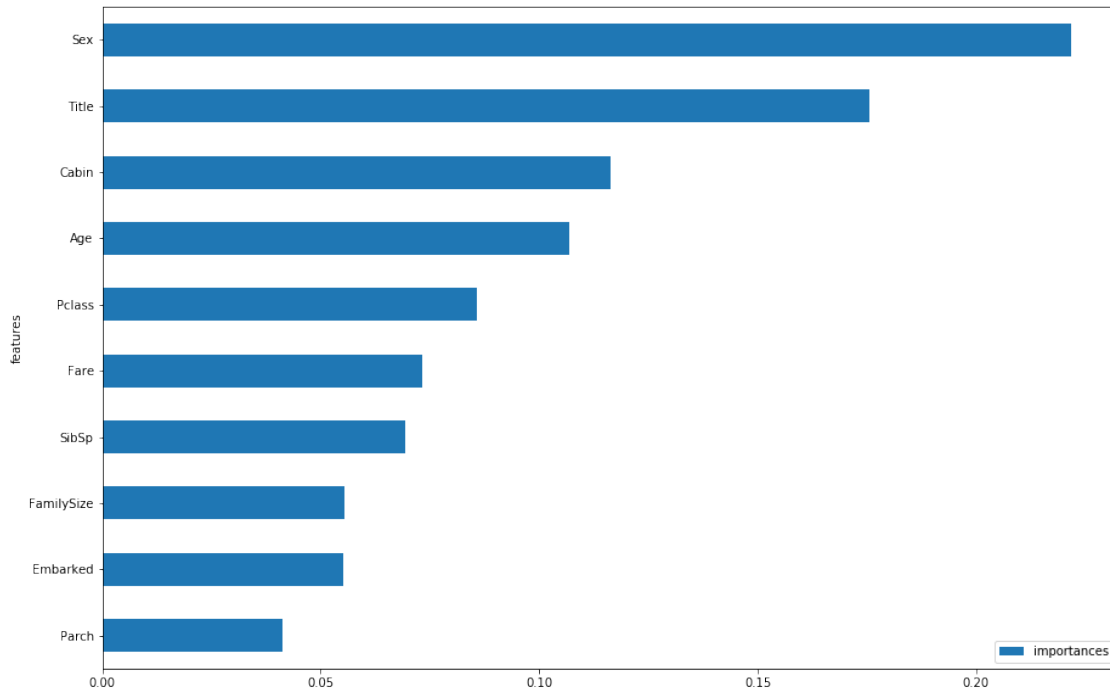
```
[294]: (2.0, 0.0)
```



- Thuật toán Random có thể được dùng để tìm các thuộc tính quan trọng trong dữ liệu.

```
[295]: features_importance = pd.DataFrame()
features_importance['features'] = features
features_importance['importances'] = rf.feature_importances_
features_importance.sort_values(by = ['importances'], ascending = True, inplace_
    => True)
features_importance.set_index('features', inplace = True)
features_importance.plot(kind = 'barh', figsize = (15, 10))
```

[295]: <matplotlib.axes.\_subplots.AxesSubplot at 0x2af2d1e4608>



- Độ chính xác của từng thuộc tính.

```
[296]: k_fold = KFold(n_splits = 10, shuffle = True, random_state = 0)
```

```
[297]: scoring = 'accuracy'
score = pd.DataFrame()
score['Feature'] = features
score['Accuracy'] = cross_val_score(rf, train[features], target, cv = k_fold,
    n_jobs = 1, scoring = scoring)
print(score)
```

	Feature	Accuracy
0	Pclass	0.800000
1	Sex	0.786517
2	Age	0.808989

```
3      SibSp  0.786517
4      Parch  0.887640
5      Fare   0.764045
6      Cabin  0.786517
7  Embarked  0.808989
8      Title  0.752809
9  FamilySize 0.786517
```

- Trung bình độ chính xác của tất cả các thuộc tính.

```
[298]: round(np.mean(score) * 100, 2)
```

```
[298]: Accuracy    79.69
dtype: float64
```

```
[299]: submission = pd.DataFrame({
        'PassengerId': test['PassengerId'],
        'Survived': prediction
    })
#submission
```

```
[300]: submission.to_csv('titanic_output.csv', index = False)
```

## 4 Bài tập

Cho bảng dữ liệu hoa Iris.csv gồm các thuộc tính sau:

- sepal\_length: chiều dài đài hoa (cm)
- sepal\_width: chiều rộng đài hoa (cm)
- petal\_length: chiều dài cánh hoa (cm)
- petal\_width: chiều rộng cánh hoa (cm)
- species: 3 loài hoa (setosa, versicolour, virginia)

Chọn lựa các thuộc tính quan trọng của tập dữ liệu hoa Iris.