

```
In [1]: import pandas as pd
```

```
In [2]: df = pd.read_csv('data/train.csv')
df.head(5)
```

Out[2]:

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin
0	1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.2500	NaN
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	female	38.0	1	0	PC 17599	71.2833	C85
2	3	1	3	Heikkinen, Miss. Laina	female	26.0	0	0	STON/O2. 3101282	7.9250	NaN
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	0	113803	53.1000	C123
4	5	0	3	Allen, Mr. William Henry	male	35.0	0	0	373450	8.0500	NaN



```
In [3]: df_test = pd.read_csv('data/test.csv')
df_test.head(5)
```

Out[3]:

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

```
In [4]: print("train data:", df.shape)
```

```
print("test data:", df_test.shape)
```

train data: (891, 12)
test data: (418, 11)

```
In [5]: 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 [6]: df.head(10)
```

Out[6]:	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin
0	1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.2500	NaN
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	female	38.0	1	0	PC 17599	71.2833	C85
2	3	1	3	Heikkinen, Miss. Laina	female	26.0	0	0	STON/O2. 3101282	7.9250	NaN
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	0	113803	53.1000	C123
4	5	0	3	Allen, Mr. William Henry	male	35.0	0	0	373450	8.0500	NaN
5	6	0	3	Moran, Mr. James	male	NaN	0	0	330877	8.4583	NaN
6	7	0	1	McCarthy, Mr. Timothy J	male	54.0	0	0	17463	51.8625	E46
7	8	0	3	Palsson, Master. Gosta Leonard	male	2.0	3	1	349909	21.0750	NaN

PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin
8	9	1	3 Johnson, Mrs. Oscar W (Elisabeth Vilhelmina Berg)	female	27.0	0	2	347742	11.1333	NaN
9	10	1	2 Nasser, Mrs. Nicholas (Adele Achem)	female	14.0	1	0	237736	30.0708	NaN



```
In [7]: df['Age'] = df['Age'].fillna(df['Age'].mean())
```

```
In [8]: avg = df['Age'].mean()
df['Age'] = df['Age'].fillna(avg)
```

```
In [9]: 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             891 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]: df_test['Age'] = df_test['Age'].fillna(df_test['Age'].mean())
```

```
In [11]: df.loc[df['Age'] < 10, 'Age'] = 0
df.loc[(df['Age'] >= 10)&(df['Age'] < 20), 'Age'] = 1
df.loc[(df['Age'] >= 20)&(df['Age'] < 30), 'Age'] = 2
df.loc[(df['Age'] >= 30)&(df['Age'] < 40), 'Age'] = 3
df.loc[(df['Age'] >= 40)&(df['Age'] < 50), 'Age'] = 4
df.loc[df['Age'] >= 50, 'Age'] = 5
```

```
In [12]: df_test.loc[df_test['Age'] < 10, 'Age'] = 0
df_test.loc[(df_test['Age'] >= 10)&(df_test['Age'] < 20), 'Age'] = 1
df_test.loc[(df_test['Age'] >= 20)&(df_test['Age'] < 30), 'Age'] = 2
df_test.loc[(df_test['Age'] >= 30)&(df_test['Age'] < 40), 'Age'] = 3
df_test.loc[(df_test['Age'] >= 40)&(df_test['Age'] < 50), 'Age'] = 4
df_test.loc[df_test['Age'] >= 50, 'Age'] = 5
```

```
In [13]: df.head()
```

```
Out [13]:
```

PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin
-------------	----------	--------	------	-----	-----	-------	-------	--------	------	-------

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin
0	1	0	3	Braund, Mr. Owen Harris	male	2.0	1	0	A/5 21171	7.2500	NaN
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...)	female	3.0	1	0	PC 17599	71.2833	C85
2	3	1	3	Heikkinen, Miss. Laina	female	2.0	0	0	STON/O2. 3101282	7.9250	NaN
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	3.0	1	0	113803	53.1000	C123
4	5	0	3	Allen, Mr. William Henry	male	3.0	0	0	373450	8.0500	NaN



In [14]:

```
df['FamilySize'] = df['SibSp'] + df['Parch']
df_test['FamilySize'] = df_test['SibSp'] + df_test['Parch']
```

In [15]:

```
df.head()
```

Out [15]:

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin
0	1	0	3	Braund, Mr. Owen Harris	male	2.0	1	0	A/5 21171	7.2500	NaN
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...)	female	3.0	1	0	PC 17599	71.2833	C85
2	3	1	3	Heikkinen, Miss. Laina	female	2.0	0	0	STON/O2. 3101282	7.9250	NaN
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	3.0	1	0	113803	53.1000	C123
4	5	0	3	Allen, Mr. William Henry	male	3.0	0	0	373450	8.0500	NaN



```
In [16]: df.isnull().sum()
```

```
Out[16]: PassengerId    0
Survived              0
Pclass               0
Name                 0
Sex                  0
Age                  0
SibSp                0
Parch                0
Ticket              0
Fare                 0
Cabin                687
Embarked             2
FamilySize           0
dtype: int64
```

```
In [17]: df_test['Fare'] = df_test['Fare'].fillna(df_test['Fare'].mean())
```

```
In [18]: df['Embarked'].value_counts()
```

```
Out[18]: S    644
C    168
Q     77
Name: Embarked, dtype: int64
```

```
In [19]: df['Embarked'] = df['Embarked'].fillna('S')
```

```
In [20]: train = df[['Survived', 'Sex', 'Age', 'FamilySize', 'Fare', 'Embarked']]
```

```
In [21]: train.head()
```

```
Out[21]:
```

	Survived	Sex	Age	FamilySize	Fare	Embarked
0	0	male	2.0	1	7.2500	S
1	1	female	3.0	1	71.2833	C
2	1	female	2.0	0	7.9250	S
3	1	female	3.0	1	53.1000	S
4	0	male	3.0	0	8.0500	S

```
In [22]: test = df_test[['Sex', 'Age', 'FamilySize', 'Fare', 'Embarked']]
```

```
In [23]: test.head()
```

```
Out[23]:
```

	Sex	Age	FamilySize	Fare	Embarked
0	male	3.0	0	7.8292	Q
1	female	4.0	1	7.0000	S
2	male	5.0	0	9.6875	Q
3	male	2.0	0	8.6625	S
4	female	2.0	2	12.2875	S

```
In [24]: train.loc[train['Sex'] == 'male', 'Sex'] = 0
train.loc[train['Sex'] == 'female', 'Sex'] = 1

train.loc[train['Embarked'] == 'S', 'Embarked'] = 0
```

```
train.loc[train['Embarked'] == 'C', 'Embarked'] = 1
train.loc[train['Embarked'] == 'Q', 'Embarked'] = 2
```

C:\Users\wsamsung\Wanaconda3\lib\site-packages\pandas\core\indexing.py:1765: SettingWithCopyWarning:

A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
isetter(loc, value)

```
In [25]: test.loc[test['Sex'] == 'male', 'Sex'] = 0
test.loc[test['Sex'] == 'female', 'Sex'] = 1

test.loc[test['Embarked'] == 'S', 'Embarked'] = 0
test.loc[test['Embarked'] == 'C', 'Embarked'] = 1
test.loc[test['Embarked'] == 'Q', 'Embarked'] = 2
```

```
In [26]: train.head()
```

```
Out[26]:
```

	Survived	Sex	Age	FamilySize	Fare	Embarked
0	0	0	2.0	1	7.2500	0
1	1	1	3.0	1	71.2833	1
2	1	1	2.0	0	7.9250	0
3	1	1	3.0	1	53.1000	0
4	0	0	3.0	0	8.0500	0

```
In [27]: test.head()
```

```
Out[27]:
```

	Sex	Age	FamilySize	Fare	Embarked
0	0	3.0	0	7.8292	2
1	1	4.0	1	7.0000	0
2	0	5.0	0	9.6875	2
3	0	2.0	0	8.6625	0
4	1	2.0	2	12.2875	0

```
In [28]: x_train = train[['Sex', 'Age', 'FamilySize', 'Fare', 'Embarked']]
y_train = train['Survived']
```

```
In [29]: y_train.head()
```

```
Out[29]:
```

0	0
1	1
2	1
3	1
4	0

Name: Survived, dtype: int64

```
In [30]: from sklearn.tree import DecisionTreeClassifier
```

```
In [31]: tree = DecisionTreeClassifier()
tree.fit(x_train, y_train)
```

Out[31]: DecisionTreeClassifier()

```
In [32]: print('training set accuracy:', tree.score(x_train, y_train))
```

training set accuracy: 0.9450056116722784

```
In [33]: result = tree.predict(test)
result
```

```
Out[33]: array([0, 0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0, 1, 0, 1, 1, 0, 0, 0, 0, 0, 1,
 1, 1, 1, 0, 1, 0, 1, 0, 0, 0, 1, 0, 1, 1, 0, 0, 0, 1, 0, 1, 0, 1,
 1, 0, 1, 0, 1, 1, 0, 0, 1, 1, 0, 0, 1, 0, 0, 0, 0, 0, 1, 0, 1,
 1, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 1, 0, 0, 1, 0, 1, 1, 0, 0, 0, 0, 1, 1,
 0, 1, 0, 1, 0, 1, 0, 1, 0, 1, 1, 1, 1, 0, 1, 0, 0, 1, 0, 1, 0, 0, 0, 0,
 0, 0, 1, 1, 1, 1, 0, 1, 1, 0, 0, 1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 1,
 1, 0, 1, 1, 0, 0, 1, 0, 1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0,
 0, 0, 1, 1, 0, 1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 1, 0, 0, 0, 1, 0,
 1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 1, 0, 0, 0, 1, 1, 1, 1,
 1, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 1, 1,
 0, 0, 0, 0, 0, 0, 1, 0, 1, 1, 0, 1, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0,
 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1,
 0, 1, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 1, 0, 0, 0,
 1, 1, 0, 1, 0, 0, 0, 1, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 0, 1, 1, 0,
 0, 0, 1, 0, 1, 0, 0, 1, 0, 1, 1, 0, 1, 0, 0, 1, 1, 1, 0, 1, 0, 0,
 1, 1, 0, 0, 1, 0, 1, 0, 0, 0, 1, 0, 1, 0, 0, 0, 0, 1, 1, 0, 0, 0, 1,
 0, 1, 0, 0, 1, 0, 1, 0, 0, 0, 1, 0, 0, 0, 1, 0, 1, 0, 0, 1, 0, 0, 0],
dtype=int64)
```

```
In [34]: # x_valid=x_train[0:100]
# y_valid=y_train[0:100]
```

```
In [35]: # x_train = x_train[100:]
# y_train = y_train[100:]
```

```
In [36]: from sklearn.ensemble import RandomForestClassifier

forest = RandomForestClassifier(n_estimators=100, max_depth=13)
forest.fit(x_train, y_train)

print('training set accuracy:', forest.score(x_train, y_train))

prediction=tree.predict(test)
prediction
```

training set accuracy: 0.9450056116722784

```
Out[36]: array([0, 0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0, 1, 0, 1, 1, 0, 0, 0, 0, 0, 1,
 1, 1, 1, 0, 1, 0, 1, 0, 0, 0, 1, 0, 1, 1, 0, 0, 0, 1, 0, 1, 0, 1,
 1, 0, 1, 0, 1, 1, 0, 0, 1, 1, 0, 0, 0, 0, 0, 1, 0, 0, 0, 1, 0, 1,
 1, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0,
 0, 1, 1, 0, 1, 0, 1, 0, 1, 0, 1, 1, 1, 1, 0, 1, 0, 0, 1, 0, 1, 0, 0, 0, 0,
 0, 0, 1, 1, 1, 1, 0, 1, 1, 0, 0, 1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 1,
 1, 0, 1, 1, 0, 0, 1, 0, 1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0,
 0, 0, 1, 1, 0, 1, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 1, 0, 0, 0, 1, 0,
 1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 1, 0, 0, 0, 1, 1, 0, 0, 1, 0,
 1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
 1, 0, 1, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 1, 0, 0, 0,
 1, 1, 0, 1, 0, 0, 0, 1, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 0, 1, 1, 0,
 0, 0, 1, 0, 1, 0, 0, 1, 0, 1, 1, 0, 1, 0, 0, 1, 1, 1, 0, 1, 0, 0,
 1, 1, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 0, 0, 1, 1, 0, 0, 0, 1,
```

```
0, 1, 0, 0, 1, 0, 1, 0, 0, 1, 0, 0, 0, 1, 0, 1, 0, 0, 1, 0, 0, 0],  
dtype=int64)
```

```
In [37]: submit = pd.DataFrame({  
        'Passenger Id': df_test['Passenger Id'],  
        'Survived': prediction  
    })  
  
submit.to_csv('submit10.csv', index=False)  
  
# 0.74880 at n_estimators=1000  
# 0.76555 at n_estimators=500  
# 0.76555 at n_estimators=100, max_depth=10  
# 0.76555 at n_estimator=100, max_depth=12  
# 0.76315 at n_estimator=100, max_depth=13  
# 0.76315 at n_estimators=100, max_depth=5  
# 0.75119 at n_estimators=100, max_depth=15  
# 0.74880 at undefined  
# 0.75358 at n_estimators=100, max_depth=7
```

```
In [ ]:
```