```
# This Python 3 environment comes with many helpful
analytics libraries installed
# It is defined by the kaggle/python Docker image:
https://github.com/kaggle/docker-python
# For example, here's several helpful packages to load

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

# Input data files are available in the read-only
"../input/" directory
# For example, running this (by clicking run or
pressing Shift+Enter) will list all files under the
input directory

import os
for dirname, _, filenames in os.walk('/kaggle/input'):
    for filename in filenames:
        print(os.path.join(dirname, filename))

# You can write up to 20GB to the current directory
(/kaggle/working/) that gets preserved as output when
you create a version using "Save & Run All"
# You can also write temporary files to /kaggle/temp/,
but they won't be saved outside of the current session
```

```
/kaggle/input/spaceship-titanic/sample_submission.csv
/kaggle/input/spaceship-titanic/train.csv
/kaggle/input/spaceship-titanic/test.csv
```

```
from sklearn import preprocessing, svm
from sklearn.impute import SimpleImputer
from sklearn.preprocessing import LabelEncoder,
OneHotEncoder, FunctionTransformer, StandardScaler
```

读取数据集

```
spaceship = pd.read_csv('/kaggle/input/spaceship-
titanic/train.csv')
spaceship_test = pd.read_csv('/kaggle/input/spaceship-
titanic/test.csv')
space_one = spaceship_test.copy()
```

数据预处理

```
# 分割passengerId和Cabins
def SplitGroupCabin(df):
    df['passengerGroup'] =
df.PassengerId.str.split('_').str[0]
    df['cabinDeck'] = df.Cabin.str.split('/').str[0]
    df['cabinNum'] = df.Cabin.str.split('/').str[1]
    df['cabinSide'] = df.Cabin.str.split('/').str[2]
    df.drop(columns=['PassengerId', 'Name', 'Cabin'],
axis=1, inplace=True)
    return df
```

```
spaceship = SplitGroupCabin(spaceship)
# Now, let's make another function to convert features
to proper datatypes
cat_col = []
num_col = []

def convert_types(df):
    global cat_col, num_col
    cat_col = ['HomePlanet', 'CryoSleep',
'Destination', 'VIP', 'cabinDeck', 'cabinSide']
    num_col = ['Age', 'passengerGroup', 'cabinNum',
'RoomService', 'FoodCourt', 'ShoppingMall', 'Spa',
'VRDeck']
    for i in cat_col:
        df[i] = df[i].astype('category')
        print(i, df[i].dtype)
    for i in num_col:
        df[i] = df[i].astype('float')
        print(i, df[i].dtype)
    return df

spaceship = convert_types(spaceship)
# impute_cols1 = ['HomePlanet', 'CryoSleep',
'Destination', 'VIP', 'cabinDeck', 'cabinSide']
impute_cols2 = ['RoomService', 'FoodCourt',
'ShoppingMall', 'Spa', 'VRDeck']
impute_cols3 = ['Age', 'cabinNum']
# imputer1 = SimpleImputer(missing_values=np.nan,
strategy='most_frequent').fit(spaceship[impute_cols1])
imputer2 = SimpleImputer(missing_values=np.nan,
fill_value=0).fit(spaceship[impute_cols2])
imputer3 = SimpleImputer(missing_values=np.nan,
strategy='mean').fit(spaceship[impute_cols3])
```

```
# spaceship[impute_cols1] =
imputer1.transform(spaceship[impute_cols1])
spaceship[impute_cols2] =
imputer2.transform(spaceship[impute_cols2])
spaceship[impute_cols3] =
imputer3.transform(spaceship[impute_cols3])

# create function for calculating sum of all services
def CalculateServices(df):
    service_col = ['RoomService', 'FoodCourt',
'ShoppingMall', 'Spa', 'VRDeck']
    df['Service_val'] = 0
    for col in service_col:
        df['Service_val'] += df[col]
    df['Service_count'] = (df[service_col] >
0).T.sum().T
    return df

spaceship = CalculateServices(spaceship)
spaceship.head(5)
spaceshipX = spaceship.drop(columns=['Transported'])
spaceshipY = spaceship['Transported']


col_scale = ['Age', 'passengerGroup', 'cabinNum',
'RoomService', 'FoodCourt', 'ShoppingMall', 'Spa',
'VRDeck',
             'Service_val']
scaler1 = StandardScaler().fit(spaceshipX[col_scale])
spaceshipX[col_scale] =
scaler1.transform(spaceshipX[col_scale])
encoder = {}
```

```
col_encode = ['CryoSleep', 'VIP', 'cabinSide',
'Destination', 'HomePlanet', 'cabinDeck']
for col in col_encode:
    encoder[col] = LabelEncoder().fit(spaceshipX[col])

def encode_columns(df):
    for col in col_encode:
        df[col] = encoder[col].transform(df[col])
    return df

spaceshipX = encode_columns(spaceshipX)
```

结果

```
HomePlanet category
CryoSleep category
Destination category
VIP category
cabinDeck category
cabinSide category
Age float64
passengerGroup float64
cabinNum float64
RoomService float64
FoodCourt float64
ShoppingMall float64
Spa float64
VRDeck float64
```

支持向量机

```
def SVM():
    clf = svm.SVC(C=2.0, cache_size=200,
class_weight=None, coef0=0.0,
                  decision_function_shape='ovo',
degree=10, gamma='auto', kernel='rbf',
                  max_iter=10000, probability=True,
random_state=None, shrinking=True,
                  tol=0.01, verbose=True)
    clf.fit(spaceshipX, spaceshipY)
    # 给spaceship_test添加一个列
    spaceship_test['Transported'] = True
    spaceship_testX, spaceship_testY =
DataHandle(spaceship_test)
    print(spaceship_testX)
    # 预测
    y_pred = clf.predict(spaceship_testX)
    # 保存结果
    result = pd.DataFrame({'PassengerId':
space_one['PassengerId'], 'Transported': y_pred})
    result.to_csv('/kaggle/working/result_svm.csv',
index=False)
```

```
SVM()
```

预测结果

```
[LibSVM]..
Warning: using -h 0 may be faster
*..*
optimization finished, #iter = 4575
obj = -5848.850781, rho = 0.533366
nSV = 3380, nBSV = 2926
```

```
Total nSV = 3380
..
Warning: using -h 0 may be faster
*..*
optimization finished, #iter = 4421
obj = -5798.159319, rho = 0.489823
nSV = 3339, nBSV = 2884
Total nSV = 3339
...
Warning: using -h 0 may be faster
*.*
optimization finished, #iter = 4233
obj = -5849.513025, rho = 0.451940
nSV = 3354, nBSV = 2933
Total nSV = 3354
...
Warning: using -h 0 may be faster
*.*
optimization finished, #iter = 4588
obj = -5844.310467, rho = 0.466535
nSV = 3368, nBSV = 2920
Total nSV = 3368
...*.*
optimization finished, #iter = 4237
obj = -5793.434772, rho = 0.468223
nSV = 3329, nBSV = 2902
Total nSV = 3329
...
Warning: using -h 0 may be faster
*..*
optimization finished, #iter = 5449
obj = -7267.701547, rho = -0.497170
nSV = 4161, nBSV = 3635
```

```
Total nSV = 4161
HomePlanet category
CryoSleep category
Destination category
VIP category
cabinDeck category
cabinSide category
Age float64
passengerGroup float64
cabinNum float64
RoomService float64
FoodCourt float64
ShoppingMall float64
Spa float64
VRDeck float64
      HomePlanet  CryoSleep  Destination           Age
VIP  RoomService  \
0               0          1            2 -1.182216e-01
   0    -0.364780
1               0          0            2 -6.886014e-01
   0    -0.364780
2               1          1            0  1.669682e-01
   0    -0.364780
3               1          0            2  6.660505e-01
   0    -0.364780
4               0          0            2 -6.173039e-01
   0    -0.348143
...           ...        ...          ...           ...
...           ...
4272            0          1            2  3.808606e-01
   0    -0.364780
4273            0          0            2  9.512404e-01
   0    -0.364780
```

```
4274           2          1            0 -2.532995e-16
   0    -0.364780
4275           1          0            3 -2.532995e-16
   0    -0.364780
4276           0          1            1  1.022538e+00
   0    -0.364780

      FoodCourt  ShoppingMall       Spa    VRDeck
passengerGroup  cabinDeck  \
0     -0.291352     -0.319859 -0.274558 -0.251561
  -1.703425          6
1     -0.285385     -0.319859  2.283008 -0.251561
  -1.701584          5
2     -0.291352     -0.319859 -0.274558 -0.251561
  -1.701215          2
3      4.118523     -0.319859 -0.110576  0.222074
  -1.700479          2
4     -0.291352      0.825745 -0.274558 -0.251561
  -1.699743          5
...         ...           ...       ...       ...
        ...        ...
4272  -0.291352     -0.319859 -0.274558 -0.251561
   1.703574          6
4273   0.270158     -0.289189 -0.265498 -0.134974
   1.704679          8
4274  -0.291352     -0.319859 -0.274558 -0.251561
   1.705415          3
4275   1.485327     -0.319859 -0.274558  0.171877
   1.706152          3
4276  -0.291352     -0.319859 -0.274558 -0.251561
   1.707625          6
```

```
      cabinNum  cabinSide  Service_val  Service_count
0    -1.193234          1    -0.515058              0
1    -1.191269          1     0.491036              2
2    -1.199130          1    -0.515058              0
3    -1.197165          1     2.120256              3
4    -1.189304          1    -0.285916              2
...        ...        ...          ...            ...
4272  1.740825          1    -0.515058              0
4273  0.000000          2    -0.153404              4
4274 -0.617427          0    -0.515058              0
4275 -0.615462          0     0.622838              2
4276  1.744755          1    -0.515058              0

[4277 rows x 16 columns]
```

决策树

```
def Dt():
    from sklearn.tree import DecisionTreeClassifier
```

```
    clf = DecisionTreeClassifier(criterion='gini',
splitter='best', max_depth=None, min_samples_split=2,
                                 min_samples_leaf=1,
min_weight_fraction_leaf=0.0, max_features=None,
random_state=None,
                                 max_leaf_nodes=None,
min_impurity_decrease=0.0,
                                 class_weight=None)
    clf.fit(spaceshipX, spaceshipY)
    # 给spaceship_test添加一个列
    spaceship_test['Transported'] = True
    spaceship_testX, spaceship_testY =
DataHandle(spaceship_test)
    print(spaceship_testX)
    # 预测
    y_pred = clf.predict(spaceship_testX)
    # 保存结果
    result = pd.DataFrame({'PassengerId':
space_one['PassengerId'], 'Transported': y_pred})
    result.to_csv('/kaggle/working/result_dt.csv',
index=False)
```

```
Dt()
```

预测结果

```
HomePlanet category
CryoSleep category
Destination category
VIP category
cabinDeck category
cabinSide category
```

```
Age float64
passengerGroup float64
cabinNum float64
RoomService float64
FoodCourt float64
ShoppingMall float64
Spa float64
VRDeck float64
      HomePlanet  CryoSleep  Destination           Age
VIP  RoomService  \
0              0          1            2 -1.182216e-01
   0    -0.364780
1              0          0            2 -6.886014e-01
   0    -0.364780
2              1          1            0  1.669682e-01
   0    -0.364780
3              1          0            2  6.660505e-01
   0    -0.364780
4              0          0            2 -6.173039e-01
   0    -0.348143
...          ...        ...          ...           ...
...          ...
4272           0          1            2  3.808606e-01
   0    -0.364780
4273           0          0            2  9.512404e-01
   0    -0.364780
4274           2          1            0 -2.532995e-16
   0    -0.364780
4275           1          0            3 -2.532995e-16
   0    -0.364780
4276           0          1            1  1.022538e+00
   0    -0.364780
```

```
      FoodCourt  ShoppingMall       Spa    VRDeck
passengerGroup  cabinDeck  \
0     -0.291352     -0.319859 -0.274558 -0.251561
  -1.703425          6
1     -0.285385     -0.319859  2.283008 -0.251561
  -1.701584          5
2     -0.291352     -0.319859 -0.274558 -0.251561
  -1.701215          2
3      4.118523     -0.319859 -0.110576  0.222074
  -1.700479          2
4     -0.291352      0.825745 -0.274558 -0.251561
  -1.699743          5
...         ...           ...       ...       ...
        ...        ...
4272  -0.291352     -0.319859 -0.274558 -0.251561
   1.703574          6
4273   0.270158     -0.289189 -0.265498 -0.134974
   1.704679          8
4274  -0.291352     -0.319859 -0.274558 -0.251561
   1.705415          3
4275   1.485327     -0.319859 -0.274558  0.171877
   1.706152          3
4276  -0.291352     -0.319859 -0.274558 -0.251561
   1.707625          6

      cabinNum  cabinSide  Service_val  Service_count

0    -1.193234          1    -0.515058              0

1    -1.191269          1     0.491036              2

2    -1.199130          1    -0.515058              0
```

```
3    -1.197165          1     2.120256               3
4    -1.189304          1    -0.285916               2
...        ...        ...          ...             ...
4272  1.740825          1    -0.515058               0
4273  0.000000          2    -0.153404               4
4274 -0.617427          0    -0.515058               0
4275 -0.615462          0     0.622838               2
4276  1.744755          1    -0.515058               0

[4277 rows x 16 columns]
```

集成学习

```
def El():
    # 实现Adaboost算法
    from sklearn.ensemble import AdaBoostClassifier
    clf = AdaBoostClassifier(n_estimators=100,
learning_rate=1.0, algorithm='SAMME.R',
random_state=None)
    clf.fit(spaceshipX, spaceshipY)
    # 给spaceship_test添加一个列
    spaceship_test['Transported'] = True
```

```
    spaceship_testX, spaceship_testY =
DataHandle(spaceship_test)
    print(spaceship_testX)
    # 预测
    y_pred = clf.predict(spaceship_testX)
    # 保存结果
    result = pd.DataFrame({'PassengerId':
space_one['PassengerId'], 'Transported': y_pred})
    result.to_csv('/kaggle/working/result_el.csv',
index=False)
```

```
El()
```

预测结果

```
HomePlanet category
CryoSleep category
Destination category
VIP category
cabinDeck category
cabinSide category
Age float64
passengerGroup float64
cabinNum float64
RoomService float64
FoodCourt float64
ShoppingMall float64
Spa float64
VRDeck float64
      HomePlanet  CryoSleep  Destination           Age
VIP  RoomService  \
```

```
0              0         1            2 -1.182216e-01
   0    -0.364780
1              0         0            2 -6.886014e-01
   0    -0.364780
2              1         1            0  1.669682e-01
   0    -0.364780
3              1         0            2  6.660505e-01
   0    -0.364780
4              0         0            2 -6.173039e-01
   0    -0.348143
...          ...       ...          ...           ...
...          ...
4272           0         1            2  3.808606e-01
   0    -0.364780
4273           0         0            2  9.512404e-01
   0    -0.364780
4274           2         1            0 -2.532995e-16
   0    -0.364780
4275           1         0            3 -2.532995e-16
   0    -0.364780
4276           0         1            1  1.022538e+00
   0    -0.364780

      FoodCourt  ShoppingMall       Spa    VRDeck
passengerGroup  cabinDeck  \
0     -0.291352     -0.319859 -0.274558 -0.251561
  -1.703425          6
1     -0.285385     -0.319859  2.283008 -0.251561
  -1.701584          5
2     -0.291352     -0.319859 -0.274558 -0.251561
  -1.701215          2
3      4.118523     -0.319859 -0.110576  0.222074
  -1.700479          2
```

```
4     -0.291352      0.825745 -0.274558 -0.251561
  -1.699743          5
...         ...           ...       ...       ...
        ...        ...
4272  -0.291352     -0.319859 -0.274558 -0.251561
   1.703574          6
4273   0.270158     -0.289189 -0.265498 -0.134974
   1.704679          8
4274  -0.291352     -0.319859 -0.274558 -0.251561
   1.705415          3
4275   1.485327     -0.319859 -0.274558  0.171877
   1.706152          3
4276  -0.291352     -0.319859 -0.274558 -0.251561
   1.707625          6

      cabinNum  cabinSide  Service_val  Service_count

0    -1.193234          1    -0.515058              0

1    -1.191269          1     0.491036              2

2    -1.199130          1    -0.515058              0

3    -1.197165          1     2.120256              3

4    -1.189304          1    -0.285916              2

...        ...        ...          ...            ...

4272  1.740825          1    -0.515058              0

4273  0.000000          2    -0.153404              4
```

```
4274 -0.617427          0    -0.515058                0

4275 -0.615462          0     0.622838                2

4276  1.744755          1    -0.515058                0


[4277 rows x 16 columns]
```