```
from sklearn.datasets import load_iris
from sklearn.tree import DecisionTreeClassifier, plot_tree
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score
import matplotlib.pyplot as plt
import pandas as pd

df = pd.read_csv("/kaggle/input/spaceship-titanic/train.csv")
df.head()
```

```
  PassengerId HomePlanet CryoSleep  Cabin  Destination   Age    VIP  \
0     0001_01     Europa     False  B/0/P  TRAPPIST-1e  39.0  False
1     0002_01      Earth     False  F/0/S  TRAPPIST-1e  24.0  False
2     0003_01     Europa     False  A/0/S  TRAPPIST-1e  58.0   True
3     0003_02     Europa     False  A/0/S  TRAPPIST-1e  33.0  False
4     0004_01      Earth     False  F/1/S  TRAPPIST-1e  16.0  False

   RoomService  FoodCourt  ShoppingMall     Spa  VRDeck
Name  \
0          0.0        0.0           0.0     0.0     0.0    Maham
Ofracculy
1        109.0        9.0          25.0   549.0    44.0       Juanna
Vines
2         43.0     3576.0           0.0  6715.0    49.0      Altark
Susent
3          0.0     1283.0         371.0  3329.0   193.0       Solam
Susent
4        303.0       70.0         151.0   565.0     2.0  Willy
Santantines

   Transported
0        False
1         True
2        False
3        False
4         True
```

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 8693 entries, 0 to 8692
Data columns (total 14 columns):
 #   Column        Non-Null Count  Dtype
---  ------        --------------  -----
 0   PassengerId   8693 non-null   object
 1   HomePlanet    8492 non-null   object
 2   CryoSleep     8476 non-null   object
 3   Cabin         8494 non-null   object
 4   Destination   8511 non-null   object
 5   Age           8514 non-null   float64
 6   VIP           8490 non-null   object
 7   RoomService   8512 non-null   float64
 8   FoodCourt     8510 non-null   float64
 9   ShoppingMall  8485 non-null   float64
 10  Spa           8510 non-null   float64
 11  VRDeck        8505 non-null   float64
 12  Name          8493 non-null   object
 13  Transported   8693 non-null   bool
dtypes: bool(1), float64(6), object(7)
memory usage: 891.5+ KB
```

```
df.head()
```

```
  HomePlanet CryoSleep  Cabin  Destination   Age    VIP
RoomService  \
0     Europa     False  B/0/P  TRAPPIST-1e  39.0  False          0.0

1      Earth     False  F/0/S  TRAPPIST-1e  24.0  False        109.0

2     Europa     False  A/0/S  TRAPPIST-1e  58.0   True         43.0

3     Europa     False  A/0/S  TRAPPIST-1e  33.0  False          0.0

4      Earth     False  F/1/S  TRAPPIST-1e  16.0  False        303.0


   FoodCourt  ShoppingMall      Spa  VRDeck  Transported
0        0.0           0.0      0.0     0.0        False
1        9.0          25.0    549.0    44.0         True
2     3576.0           0.0   6715.0    49.0        False
3     1283.0         371.0   3329.0   193.0        False
4       70.0         151.0    565.0     2.0         True
```

```
X = df.drop(columns=['Transported'])
y = df['Transported']
y = y.to_numpy()
```

```
X.head()
```

```
  HomePlanet CryoSleep  Cabin   Age    VIP  RoomService  FoodCourt  \
0     Europa     False  B/0/P  39.0  False          0.0        0.0
1      Earth     False  F/0/S  24.0  False        109.0        9.0
2     Europa     False  A/0/S  58.0   True         43.0     3576.0
3     Europa     False  A/0/S  33.0  False          0.0     1283.0
4      Earth     False  F/1/S  16.0  False        303.0       70.0

   ShoppingMall      Spa  VRDeck
0           0.0      0.0     0.0
1          25.0    549.0    44.0
2           0.0   6715.0    49.0
```

```
3         371.0  3329.0   193.0
4         151.0   565.0     2.0
```

```
from sklearn.preprocessing import StandardScaler, LabelEncoder
from sklearn.preprocessing import OneHotEncoder
from sklearn.compose import ColumnTransformer

X['HomePlanet'] = LabelEncoder().fit_transform(X['HomePlanet'])
X['CryoSleep'] = LabelEncoder().fit_transform(X['CryoSleep'])
X['FoodCourt'] =
StandardScaler().fit_transform(X['FoodCourt'].to_numpy().reshape(-
1,1))
X['ShoppingMall'] =
StandardScaler().fit_transform(X['ShoppingMall'].to_numpy().reshape(-
1,1))
X['Spa'] =
StandardScaler().fit_transform(X['Spa'].to_numpy().reshape(-1,1))
X['VRDeck'] =
StandardScaler().fit_transform(X['VRDeck'].to_numpy().reshape(-1,1))
X['VIP'] = LabelEncoder().fit_transform(X['VIP'])
X['Cabin'] = LabelEncoder().fit_transform(X['Cabin'])
X['Age'] =
StandardScaler().fit_transform(X['Age'].to_numpy().reshape(-1,1))
X['RoomService'] =
StandardScaler().fit_transform(X['RoomService'].to_numpy().reshape(-
1,1))
X.tail()
```

```
      HomePlanet  CryoSleep  Cabin       Age  VIP  RoomService
FoodCourt  \
8688           1          0    146  0.848924    1    -0.482039
2.989544
8689           0          1   5280 -0.755179    0    -0.482039  -
0.490206
8690           0          0   5285 -0.197230    0    -0.482039  -
0.490206
8691           1          0   2131  0.221232    0    -0.482039
1.361151
8692           1          0   2131  1.058155    0    -0.107410
2.796642

      ShoppingMall       Spa    VRDeck
8688     -0.463857  2.249560 -0.254117
8689     -0.463857 -0.480047 -0.473852
8690      2.924236 -0.477118 -0.473852
8691     -0.463857  0.460086  2.831082
8692     -0.463857 -0.480047 -0.438219
```

```
X['HomePlanet'] = X['HomePlanet'].fillna(X['HomePlanet'].mean())
X['Cabin'] = X['Cabin'].fillna(X['Cabin'].mean())
```

```
X['VIP'] = X['VIP'].fillna(X['VIP'].mean())
X['RoomService'] = X['RoomService'].fillna(X['RoomService'].mean())
X['Age'] = X['Age'].fillna(X['Age'].mean())
X['CryoSleep'] = X['CryoSleep'].fillna(X['CryoSleep'].mean())
X['RoomService'] = X['RoomService'].fillna(X['RoomService'].mean())
X['FoodCourt'] = X['FoodCourt'].fillna(X['FoodCourt'].mean())
X['Spa'] = X['Spa'].fillna(X['Spa'].mean())
X['ShoppingMall'] = X['ShoppingMall'].fillna(X['ShoppingMall'].mean())
X['VRDeck'] = X['VRDeck'].fillna(X['VRDeck'].mean())

X_train, X_test, y_train, y_test = train_test_split(X, y,
test_size=0.33, random_state=46)
model = LogisticRegression()
model.fit(X_train, y_train)
y_preds = model.predict(X_test)
y_preds = model.predict(X_test)
from sklearn.metrics import accuracy_score
accuracy_score(y_test, y_preds)
```

```
/usr/local/lib/python3.10/dist-packages/sklearn/linear_model/
_logistic.py:458: ConvergenceWarning: lbfgs failed to converge
(status=1):
STOP: TOTAL NO. of ITERATIONS REACHED LIMIT.

Increase the number of iterations (max_iter) or scale the data as
shown in:
    https://scikit-learn.org/stable/modules/preprocessing.html
Please also refer to the documentation for alternative solver options:

https://scikit-learn.org/stable/modules/linear_model.html#logistic-
regression
  n_iter_i = _check_optimize_result(
```

```
0.7647263855001742
```

```
df2 = pd.read_csv("/kaggle/input/spaceship-titanic/test.csv")

pid = df2['PassengerId']
df2 = df2.drop(columns=["Destination", "Name", "PassengerId"])

df2['HomePlanet'] = LabelEncoder().fit_transform(df2['HomePlanet'])
df2['CryoSleep'] = LabelEncoder().fit_transform(df2['CryoSleep'])
df2['FoodCourt'] =
StandardScaler().fit_transform(df2['FoodCourt'].to_numpy().reshape(-
1,1))
df2['ShoppingMall'] =
StandardScaler().fit_transform(df2['ShoppingMall'].to_numpy().reshape(
-1,1))
df2['Spa'] =
StandardScaler().fit_transform(df2['Spa'].to_numpy().reshape(-1,1))
```

```
df2['VRDeck'] = 
StandardScaler().fit_transform(df2['VRDeck'].to_numpy().reshape(-1,1))
df2['VIP'] = LabelEncoder().fit_transform(df2['VIP'])
df2['Cabin'] = LabelEncoder().fit_transform(df2['Cabin'])
df2['Age'] = 
StandardScaler().fit_transform(df2['Age'].to_numpy().reshape(-1,1))
df2['RoomService'] = 
StandardScaler().fit_transform(df2['RoomService'].to_numpy().reshape(-
1,1))

df2.tail()
```

```
/usr/local/lib/python3.10/dist-packages/pandas/io/formats/
format.py:1458: RuntimeWarning: invalid value encountered in greater
  has_large_values = (abs_vals > 1e6).any()
/usr/local/lib/python3.10/dist-packages/pandas/io/formats/format.py:14
59: RuntimeWarning: invalid value encountered in less
  has_small_values = ((abs_vals < 10 ** (-self.digits)) & (abs_vals > 
0)).any()
/usr/local/lib/python3.10/dist-packages/pandas/io/formats/format.py:14
59: RuntimeWarning: invalid value encountered in greater
  has_small_values = ((abs_vals < 10 ** (-self.digits)) & (abs_vals > 
0)).any()

      HomePlanet  CryoSleep  Cabin       Age  VIP  RoomService  
FoodCourt  \
4272           0          1   2679  0.376787    0    -0.361266  -
0.287719   
4273           0          0   3265  0.941066    0    -0.361266   
0.266790   
4274           2          1    602       NaN    0    -0.361266  -
0.287719   
4275           1          0    603       NaN    0    -0.361266   
1.466805   
4276           0          1   2680  1.011601    0    -0.361266  -
0.287719   

      ShoppingMall       Spa    VRDeck  
4272     -0.316173 -0.271297 -0.249197  
4273     -0.285857 -0.262344 -0.133705  
4274     -0.316173 -0.271297 -0.249197  
4275     -0.316173 -0.271297  0.170262  
4276     -0.316173 -0.271297 -0.249197  
```

```
df2['HomePlanet'] = df2['HomePlanet'].fillna(df2['HomePlanet'].mean())
df2['Cabin'] = df2['Cabin'].fillna(df2['Cabin'].mean())
df2['VIP'] = df2['VIP'].fillna(df2['VIP'].mean())
df2['RoomService'] = 
df2['RoomService'].fillna(df2['RoomService'].mean())
df2['Age'] = df2['Age'].fillna(df2['Age'].mean())
```

```
df2['CryoSleep'] = df2['CryoSleep'].fillna(df2['CryoSleep'].mean())
df2['RoomService'] =
df2['RoomService'].fillna(df2['RoomService'].mean())
df2['FoodCourt'] = df2['FoodCourt'].fillna(df2['FoodCourt'].mean())
df2['Spa'] = df2['Spa'].fillna(df2['Spa'].mean())
df2['ShoppingMall'] =
df2['ShoppingMall'].fillna(df2['ShoppingMall'].mean())
df2['VRDeck'] = df2['VRDeck'].fillna(df2['VRDeck'].mean())

preds = model.predict(df2)
output = pd.DataFrame({'Passengerid':pid.to_numpy(),
'Transported':preds})

output.to_csv("/kaggle/working/submission.csv", index=False)
```