DecisionTree

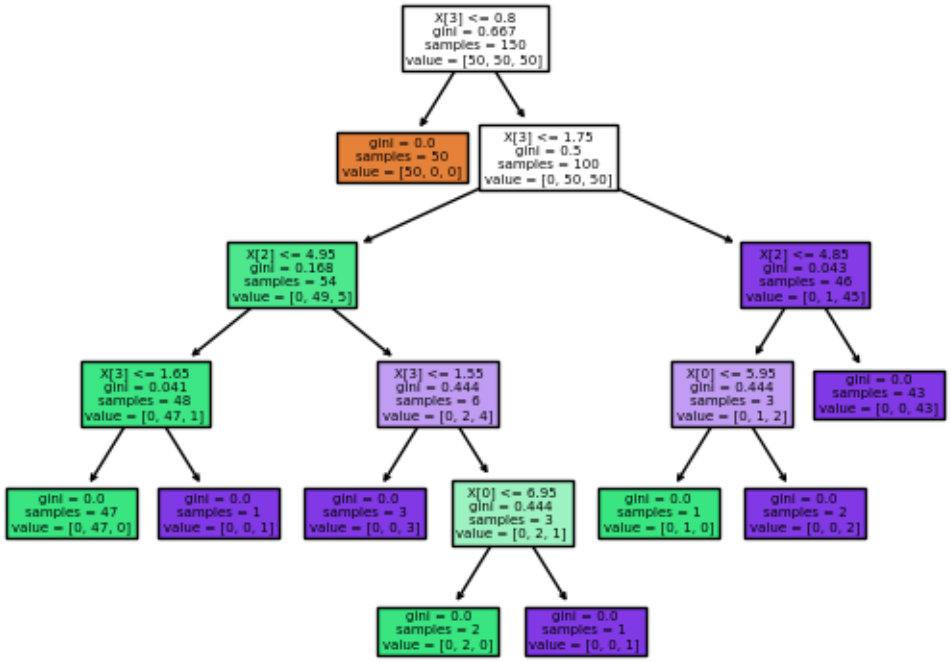
April 16, 2023

[2]:
```
from sklearn import tree
from sklearn.datasets import load_iris
from sklearn import tree
```

[4]:
```
clf = tree.DecisionTreeClassifier()
X, y = load_iris(return_X_y=True)
clf = clf.fit(X, y)
tree.plot_tree(clf,filled=True)
```

[4]:
```
[Text(0.5, 0.9166666666666666, 'X[3] <= 0.8\ngini = 0.667\nsamples = 150\nvalue
= [50, 50, 50]'),
 Text(0.4230769230769231, 0.75, 'gini = 0.0\nsamples = 50\nvalue = [50, 0, 0]'),
 Text(0.5769230769230769, 0.75, 'X[3] <= 1.75\ngini = 0.5\nsamples = 100\nvalue
= [0, 50, 50]'),
 Text(0.3076923076923077, 0.5833333333333334, 'X[2] <= 4.95\ngini =
0.168\nsamples = 54\nvalue = [0, 49, 5]'),
 Text(0.15384615384615385, 0.4166666666666667, 'X[3] <= 1.65\ngini =
0.041\nsamples = 48\nvalue = [0, 47, 1]'),
 Text(0.07692307692307693, 0.25, 'gini = 0.0\nsamples = 47\nvalue = [0, 47,
0]'),
 Text(0.23076923076923078, 0.25, 'gini = 0.0\nsamples = 1\nvalue = [0, 0, 1]'),
 Text(0.46153846153846156, 0.4166666666666667, 'X[3] <= 1.55\ngini =
0.444\nsamples = 6\nvalue = [0, 2, 4]'),
 Text(0.38461538461538464, 0.25, 'gini = 0.0\nsamples = 3\nvalue = [0, 0, 3]'),
 Text(0.5384615384615384, 0.25, 'X[0] <= 6.95\ngini = 0.444\nsamples = 3\nvalue
= [0, 2, 1]'),
 Text(0.46153846153846156, 0.08333333333333333, 'gini = 0.0\nsamples = 2\nvalue
= [0, 2, 0]'),
 Text(0.6153846153846154, 0.08333333333333333, 'gini = 0.0\nsamples = 1\nvalue =
[0, 0, 1]'),
 Text(0.8461538461538461, 0.5833333333333334, 'X[2] <= 4.85\ngini =
0.043\nsamples = 46\nvalue = [0, 1, 45]'),
 Text(0.7692307692307693, 0.4166666666666667, 'X[0] <= 5.95\ngini =
0.444\nsamples = 3\nvalue = [0, 1, 2]'),
 Text(0.6923076923076923, 0.25, 'gini = 0.0\nsamples = 1\nvalue = [0, 1, 0]'),
 Text(0.8461538461538461, 0.25, 'gini = 0.0\nsamples = 2\nvalue = [0, 0, 2]'),
 Text(0.9230769230769231, 0.4166666666666667, 'gini = 0.0\nsamples = 43\nvalue =
```

```
[0, 0, 43]')]
```



[5]:
```
import pandas as pd
import numpy as np
```

[9]:
```
df = pd.read_csv("titanic_train.csv")
y = df['survived']
```

[10]:
```
df.head()
```

[10]:

	passenger_id	pclass	name	sex	age	sibsp	parch	ticket	fare	cabin	embarked	boat	body
0	1216	3	Smyth, Miss. Julia	female	NaN	0	0	335432	7.7333	NaN	Q	13	NaN
1	699	3	Cacic, Mr. Luka	male	38.0	0	0	315089	8.6625	NaN	S	NaN	NaN
2	1267	3	Van Impe, Mrs. Jean Baptiste (Rosalie Paula Go…	female	30.0	1	1	345773	24.1500	NaN	S	NaN	NaN
3	449	2	Hocking, Mrs. Elizabeth (Eliza Needs)	female	54.0	1	3	29105	23.0000	NaN	S	4	NaN
4	576	2	Veal, Mr. James	male	40.0	0	0	28221	13.0000	NaN	S	NaN	NaN

```
                  home.dest  survived
0                       NaN         1
1                   Croatia         0
2                       NaN         0
3       Cornwall / Akron, OH         1
4  Barre, Co Washington, VT          0
```

[11]:
```
df.drop('home.dest',axis=1,inplace=True)
df.drop('pclass',axis=1,inplace=True)
df.drop('passenger_id',axis=1,inplace=True)
df.drop('name',axis=1,inplace=True)
df.head()
```

[11]:
```
      sex   age  sibsp  parch  ticket     fare cabin embarked boat  body  \
0  female   NaN      0      0  335432   7.7333   NaN        Q   13   NaN
1    male  38.0      0      0  315089   8.6625   NaN        S  NaN   NaN
2  female  30.0      1      1  345773  24.1500   NaN        S  NaN   NaN
3  female  54.0      1      3   29105  23.0000   NaN        S    4   NaN
4    male  40.0      0      0   28221  13.0000   NaN        S  NaN   NaN

   survived
0         1
1         0
2         0
3         1
4         0
```

[12]:
```
df.drop('survived',axis=1,inplace=True)
```

[13]:
```
X = df
```

[14]:
```
clf = clf.fit(X, y)
tree.plot_tree(clf,filled=True)
```

```
---------------------------------------------------------------------------
ValueError                                Traceback (most recent call last)
/tmp/ipykernel_27816/1732488384.py in <module>
----> 1 clf = clf.fit(X, y)
      2 tree.plot_tree(clf,filled=True)

~/anaconda3/lib/python3.9/site-packages/sklearn/tree/_classes.py in fit(self, X
 ↪y, sample_weight, check_input, X_idx_sorted)
    935         """
    936
--> 937         super().fit(
    938             X,
```

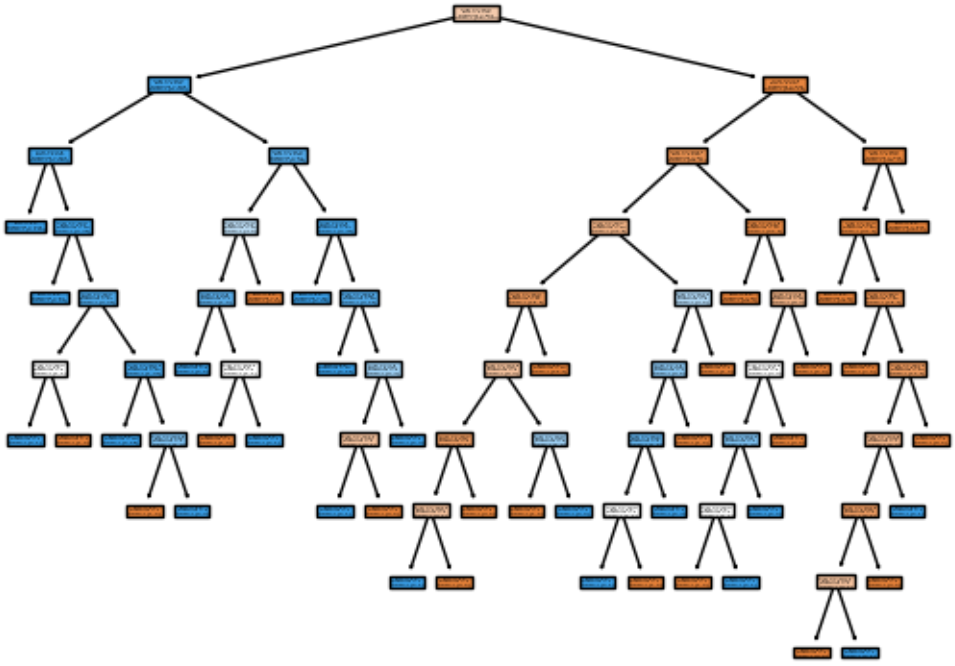
```
    939             y,

~/anaconda3/lib/python3.9/site-packages/sklearn/tree/_classes.py in fit(self, X␣
↪y, sample_weight, check_input, X_idx_sorted)
    163             check_X_params = dict(dtype=DTYPE, accept_sparse="csc")
    164             check_y_params = dict(ensure_2d=False, dtype=None)
--> 165             X, y = self._validate_data(
    166                 X, y, validate_separately=(check_X_params,␣
↪check_y_params)
    167             )

~/anaconda3/lib/python3.9/site-packages/sklearn/base.py in _validate_data(self,␣
↪X, y, reset, validate_separately, **check_params)
    576                 # :(
    577                 check_X_params, check_y_params = validate_separately
--> 578                 X = check_array(X, **check_X_params)
    579                 y = check_array(y, **check_y_params)
    580             else:

~/anaconda3/lib/python3.9/site-packages/sklearn/utils/validation.py in␣
↪check_array(array, accept_sparse, accept_large_sparse, dtype, order, copy,␣
↪force_all_finite, ensure_2d, allow_nd, ensure_min_samples,␣
↪ensure_min_features, estimator)
    744                     array = array.astype(dtype, casting="unsafe",␣
↪copy=False)
    745                 else:
--> 746                     array = np.asarray(array, order=order, dtype=dtype)
    747             except ComplexWarning as complex_warning:
    748                 raise ValueError(

~/anaconda3/lib/python3.9/site-packages/pandas/core/generic.py in␣
↪__array__(self, dtype)
   2062
   2063     def __array__(self, dtype: npt.DTypeLike | None = None) -> np.
↪ndarray:
-> 2064         return np.asarray(self._values, dtype=dtype)
   2065
   2066     def __array_wrap__(

ValueError: could not convert string to float: 'female'
```

[15]:
```
from sklearn import preprocessing
label_encoder = preprocessing.LabelEncoder()
for column in X:
    X[column] = label_encoder.fit_transform(X[column])
```

[16]:
```
clf = clf.fit(X, y)
tree.plot_tree(clf,filled=True)
```

[16]:
```
[Text(0.4984375, 0.95, 'X[8] <= 25.5\ngini = 0.465\nsamples = 850\nvalue = [537,
313]'),
 Text(0.175, 0.85, 'X[8] <= 20.5\ngini = 0.051\nsamples = 308\nvalue = [8,
300]'),
 Text(0.05, 0.75, 'X[0] <= 0.5\ngini = 0.016\nsamples = 254\nvalue = [2, 252]'),
 Text(0.025, 0.65, 'gini = 0.0\nsamples = 176\nvalue = [0, 176]'),
 Text(0.075, 0.65, 'X[1] <= 43.5\ngini = 0.05\nsamples = 78\nvalue = [2, 76]'),
 Text(0.05, 0.55, 'gini = 0.0\nsamples = 54\nvalue = [0, 54]'),
 Text(0.1, 0.55, 'X[1] <= 46.0\ngini = 0.153\nsamples = 24\nvalue = [2, 22]'),
 Text(0.05, 0.45, 'X[5] <= 95.5\ngini = 0.5\nsamples = 2\nvalue = [1, 1]'),
 Text(0.025, 0.35, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
 Text(0.075, 0.35, 'gini = 0.0\nsamples = 1\nvalue = [1, 0]'),
 Text(0.15, 0.45, 'X[4] <= 590.0\ngini = 0.087\nsamples = 22\nvalue = [1, 21]'),
 Text(0.125, 0.35, 'gini = 0.0\nsamples = 18\nvalue = [0, 18]'),
 Text(0.175, 0.35, 'X[4] <= 595.5\ngini = 0.375\nsamples = 4\nvalue = [1, 3]'),
 Text(0.15, 0.25, 'gini = 0.0\nsamples = 1\nvalue = [1, 0]'),
 Text(0.2, 0.25, 'gini = 0.0\nsamples = 3\nvalue = [0, 3]'),
 Text(0.3, 0.75, 'X[8] <= 21.5\ngini = 0.198\nsamples = 54\nvalue = [6, 48]'),
 Text(0.25, 0.65, 'X[1] <= 47.5\ngini = 0.48\nsamples = 10\nvalue = [4, 6]'),
 Text(0.225, 0.55, 'X[2] <= 0.5\ngini = 0.245\nsamples = 7\nvalue = [1, 6]'),
 Text(0.2, 0.45, 'gini = 0.0\nsamples = 5\nvalue = [0, 5]'),
 Text(0.25, 0.45, 'X[5] <= 103.0\ngini = 0.5\nsamples = 2\nvalue = [1, 1]'),
 Text(0.225, 0.35, 'gini = 0.0\nsamples = 1\nvalue = [1, 0]'),
 Text(0.275, 0.35, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
 Text(0.275, 0.55, 'gini = 0.0\nsamples = 3\nvalue = [3, 0]'),
 Text(0.35, 0.65, 'X[2] <= 0.5\ngini = 0.087\nsamples = 44\nvalue = [2, 42]'),
 Text(0.325, 0.55, 'gini = 0.0\nsamples = 26\nvalue = [0, 26]'),
 Text(0.375, 0.55, 'X[4] <= 184.5\ngini = 0.198\nsamples = 18\nvalue = [2,
16]'),
 Text(0.35, 0.45, 'gini = 0.0\nsamples = 12\nvalue = [0, 12]'),
 Text(0.4, 0.45, 'X[3] <= 0.5\ngini = 0.444\nsamples = 6\nvalue = [2, 4]'),
 Text(0.375, 0.35, 'X[6] <= 106.0\ngini = 0.444\nsamples = 3\nvalue = [2, 1]'),
 Text(0.35, 0.25, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
 Text(0.4, 0.25, 'gini = 0.0\nsamples = 2\nvalue = [2, 0]'),
 Text(0.425, 0.35, 'gini = 0.0\nsamples = 3\nvalue = [0, 3]'),
 Text(0.821875, 0.85, 'X[0] <= 0.5\ngini = 0.047\nsamples = 542\nvalue = [529,
13]'),
 Text(0.71875, 0.75, 'X[4] <= 278.0\ngini = 0.198\nsamples = 99\nvalue = [88,
11]'),
 Text(0.6375, 0.65, 'X[1] <= 42.0\ngini = 0.4\nsamples = 29\nvalue = [21, 8]'),
 Text(0.55, 0.55, 'X[1] <= 25.5\ngini = 0.245\nsamples = 21\nvalue = [18, 3]'),
 Text(0.525, 0.45, 'X[5] <= 106.5\ngini = 0.444\nsamples = 9\nvalue = [6, 3]'),
 Text(0.475, 0.35, 'X[1] <= 22.0\ngini = 0.278\nsamples = 6\nvalue = [5, 1]'),
 Text(0.45, 0.25, 'X[4] <= 189.0\ngini = 0.444\nsamples = 3\nvalue = [2, 1]'),
```

```
 Text(0.425, 0.15, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
 Text(0.475, 0.15, 'gini = 0.0\nsamples = 2\nvalue = [2, 0]'),
 Text(0.5, 0.25, 'gini = 0.0\nsamples = 3\nvalue = [3, 0]'),
 Text(0.575, 0.35, 'X[4] <= 72.5\ngini = 0.444\nsamples = 3\nvalue = [1, 2]'),
 Text(0.55, 0.25, 'gini = 0.0\nsamples = 1\nvalue = [1, 0]'),
 Text(0.6, 0.25, 'gini = 0.0\nsamples = 2\nvalue = [0, 2]'),
 Text(0.575, 0.45, 'gini = 0.0\nsamples = 12\nvalue = [12, 0]'),
 Text(0.725, 0.55, 'X[5] <= 125.0\ngini = 0.469\nsamples = 8\nvalue = [3, 5]'),
 Text(0.7, 0.45, 'X[3] <= 1.5\ngini = 0.408\nsamples = 7\nvalue = [2, 5]'),
 Text(0.675, 0.35, 'X[5] <= 26.0\ngini = 0.278\nsamples = 6\nvalue = [1, 5]'),
 Text(0.65, 0.25, 'X[5] <= 19.5\ngini = 0.5\nsamples = 2\nvalue = [1, 1]'),
 Text(0.625, 0.15, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
 Text(0.675, 0.15, 'gini = 0.0\nsamples = 1\nvalue = [1, 0]'),
 Text(0.7, 0.25, 'gini = 0.0\nsamples = 4\nvalue = [0, 4]'),
 Text(0.725, 0.35, 'gini = 0.0\nsamples = 1\nvalue = [1, 0]'),
 Text(0.75, 0.45, 'gini = 0.0\nsamples = 1\nvalue = [1, 0]'),
 Text(0.8, 0.65, 'X[4] <= 577.5\ngini = 0.082\nsamples = 70\nvalue = [67, 3]'),
 Text(0.775, 0.55, 'gini = 0.0\nsamples = 59\nvalue = [59, 0]'),
 Text(0.825, 0.55, 'X[2] <= 0.5\ngini = 0.397\nsamples = 11\nvalue = [8, 3]'),
 Text(0.8, 0.45, 'X[4] <= 648.5\ngini = 0.5\nsamples = 6\nvalue = [3, 3]'),
 Text(0.775, 0.35, 'X[4] <= 605.0\ngini = 0.375\nsamples = 4\nvalue = [1, 3]'),
 Text(0.75, 0.25, 'X[1] <= 69.5\ngini = 0.5\nsamples = 2\nvalue = [1, 1]'),
 Text(0.725, 0.15, 'gini = 0.0\nsamples = 1\nvalue = [1, 0]'),
 Text(0.775, 0.15, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
 Text(0.8, 0.25, 'gini = 0.0\nsamples = 2\nvalue = [0, 2]'),
 Text(0.825, 0.35, 'gini = 0.0\nsamples = 2\nvalue = [2, 0]'),
 Text(0.85, 0.45, 'gini = 0.0\nsamples = 5\nvalue = [5, 0]'),
 Text(0.925, 0.75, 'X[5] <= 28.5\ngini = 0.009\nsamples = 443\nvalue = [441,
2]'),
 Text(0.9, 0.65, 'X[5] <= 27.5\ngini = 0.04\nsamples = 99\nvalue = [97, 2]'),
 Text(0.875, 0.55, 'gini = 0.0\nsamples = 79\nvalue = [79, 0]'),
 Text(0.925, 0.55, 'X[1] <= 85.0\ngini = 0.18\nsamples = 20\nvalue = [18, 2]'),
 Text(0.9, 0.45, 'gini = 0.0\nsamples = 7\nvalue = [7, 0]'),
 Text(0.95, 0.45, 'X[4] <= 450.0\ngini = 0.26\nsamples = 13\nvalue = [11, 2]'),
 Text(0.925, 0.35, 'X[4] <= 445.5\ngini = 0.408\nsamples = 7\nvalue = [5, 2]'),
 Text(0.9, 0.25, 'X[4] <= 432.5\ngini = 0.278\nsamples = 6\nvalue = [5, 1]'),
 Text(0.875, 0.15, 'X[4] <= 430.0\ngini = 0.444\nsamples = 3\nvalue = [2, 1]'),
 Text(0.85, 0.05, 'gini = 0.0\nsamples = 2\nvalue = [2, 0]'),
 Text(0.9, 0.05, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
 Text(0.925, 0.15, 'gini = 0.0\nsamples = 3\nvalue = [3, 0]'),
 Text(0.95, 0.25, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
 Text(0.975, 0.35, 'gini = 0.0\nsamples = 6\nvalue = [6, 0]'),
 Text(0.95, 0.65, 'gini = 0.0\nsamples = 344\nvalue = [344, 0]')]
```



[ ]: