classifying-irises

February 13, 2025

[1]:
```
from sklearn.datasets import load_iris
```

[2]:
```
X, y = load_iris(as_frame=True, return_X_y=True)
```

Scale the features as we did with the hierarchical clustering demo

[3]:
```
from sklearn.preprocessing import StandardScaler
```

[4]:
```
scaler = StandardScaler()
X_scaled = scaler.fit_transform(X)
X_scaled
```

[4]:
```
array([[-9.00681170e-01,  1.01900435e+00, -1.34022653e+00,
        -1.31544430e+00],
       [-1.14301691e+00, -1.31979479e-01, -1.34022653e+00,
        -1.31544430e+00],
       [-1.38535265e+00,  3.28414053e-01, -1.39706395e+00,
        -1.31544430e+00],
       [-1.50652052e+00,  9.82172869e-02, -1.28338910e+00,
        -1.31544430e+00],
       [-1.02184904e+00,  1.24920112e+00, -1.34022653e+00,
        -1.31544430e+00],
       [-5.37177559e-01,  1.93979142e+00, -1.16971425e+00,
        -1.05217993e+00],
       [-1.50652052e+00,  7.88807586e-01, -1.34022653e+00,
        -1.18381211e+00],
       [-1.02184904e+00,  7.88807586e-01, -1.28338910e+00,
        -1.31544430e+00],
       [-1.74885626e+00, -3.62176246e-01, -1.34022653e+00,
        -1.31544430e+00],
       [-1.14301691e+00,  9.82172869e-02, -1.28338910e+00,
        -1.44707648e+00],
       [-5.37177559e-01,  1.47939788e+00, -1.28338910e+00,
        -1.31544430e+00],
       [-1.26418478e+00,  7.88807586e-01, -1.22655167e+00,
        -1.31544430e+00],
       [-1.26418478e+00, -1.31979479e-01, -1.34022653e+00,
        -1.44707648e+00],
```

```
[-1.87002413e+00, -1.31979479e-01, -1.51073881e+00,
 -1.44707648e+00],
[-5.25060772e-02,  2.16998818e+00, -1.45390138e+00,
 -1.31544430e+00],
[-1.73673948e-01,  3.09077525e+00, -1.28338910e+00,
 -1.05217993e+00],
[-5.37177559e-01,  1.93979142e+00, -1.39706395e+00,
 -1.05217993e+00],
[-9.00681170e-01,  1.01900435e+00, -1.34022653e+00,
 -1.18381211e+00],
[-1.73673948e-01,  1.70959465e+00, -1.16971425e+00,
 -1.18381211e+00],
[-9.00681170e-01,  1.70959465e+00, -1.28338910e+00,
 -1.18381211e+00],
[-5.37177559e-01,  7.88807586e-01, -1.16971425e+00,
 -1.31544430e+00],
[-9.00681170e-01,  1.47939788e+00, -1.28338910e+00,
 -1.05217993e+00],
[-1.50652052e+00,  1.24920112e+00, -1.56757623e+00,
 -1.31544430e+00],
[-9.00681170e-01,  5.58610819e-01, -1.16971425e+00,
 -9.20547742e-01],
[-1.26418478e+00,  7.88807586e-01, -1.05603939e+00,
 -1.31544430e+00],
[-1.02184904e+00, -1.31979479e-01, -1.22655167e+00,
 -1.31544430e+00],
[-1.02184904e+00,  7.88807586e-01, -1.22655167e+00,
 -1.05217993e+00],
[-7.79513300e-01,  1.01900435e+00, -1.28338910e+00,
 -1.31544430e+00],
[-7.79513300e-01,  7.88807586e-01, -1.34022653e+00,
 -1.31544430e+00],
[-1.38535265e+00,  3.28414053e-01, -1.22655167e+00,
 -1.31544430e+00],
[-1.26418478e+00,  9.82172869e-02, -1.22655167e+00,
 -1.31544430e+00],
[-5.37177559e-01,  7.88807586e-01, -1.28338910e+00,
 -1.05217993e+00],
[-7.79513300e-01,  2.40018495e+00, -1.28338910e+00,
 -1.44707648e+00],
[-4.16009689e-01,  2.63038172e+00, -1.34022653e+00,
 -1.31544430e+00],
[-1.14301691e+00,  9.82172869e-02, -1.28338910e+00,
 -1.31544430e+00],
[-1.02184904e+00,  3.28414053e-01, -1.45390138e+00,
 -1.31544430e+00],
[-4.16009689e-01,  1.01900435e+00, -1.39706395e+00,
```

```
 -1.31544430e+00],
[-1.14301691e+00,  1.24920112e+00, -1.34022653e+00,
 -1.44707648e+00],
[-1.74885626e+00, -1.31979479e-01, -1.39706395e+00,
 -1.31544430e+00],
[-9.00681170e-01,  7.88807586e-01, -1.28338910e+00,
 -1.31544430e+00],
[-1.02184904e+00,  1.01900435e+00, -1.39706395e+00,
 -1.18381211e+00],
[-1.62768839e+00, -1.74335684e+00, -1.39706395e+00,
 -1.18381211e+00],
[-1.74885626e+00,  3.28414053e-01, -1.39706395e+00,
 -1.31544430e+00],
[-1.02184904e+00,  1.01900435e+00, -1.22655167e+00,
 -7.88915558e-01],
[-9.00681170e-01,  1.70959465e+00, -1.05603939e+00,
 -1.05217993e+00],
[-1.26418478e+00, -1.31979479e-01, -1.34022653e+00,
 -1.18381211e+00],
[-9.00681170e-01,  1.70959465e+00, -1.22655167e+00,
 -1.31544430e+00],
[-1.50652052e+00,  3.28414053e-01, -1.34022653e+00,
 -1.31544430e+00],
[-6.58345429e-01,  1.47939788e+00, -1.28338910e+00,
 -1.31544430e+00],
[-1.02184904e+00,  5.58610819e-01, -1.34022653e+00,
 -1.31544430e+00],
[ 1.40150837e+00,  3.28414053e-01,  5.35408562e-01,
  2.64141916e-01],
[ 6.74501145e-01,  3.28414053e-01,  4.21733708e-01,
  3.95774101e-01],
[ 1.28034050e+00,  9.82172869e-02,  6.49083415e-01,
  3.95774101e-01],
[-4.16009689e-01, -1.74335684e+00,  1.37546573e-01,
  1.32509732e-01],
[ 7.95669016e-01, -5.92373012e-01,  4.78571135e-01,
  3.95774101e-01],
[-1.73673948e-01, -5.92373012e-01,  4.21733708e-01,
  1.32509732e-01],
[ 5.53333275e-01,  5.58610819e-01,  5.35408562e-01,
  5.27406285e-01],
[-1.14301691e+00, -1.51316008e+00, -2.60315415e-01,
 -2.62386821e-01],
[ 9.16836886e-01, -3.62176246e-01,  4.78571135e-01,
  1.32509732e-01],
[-7.79513300e-01, -8.22569778e-01,  8.07091462e-02,
  2.64141916e-01],
```

```
[-1.02184904e+00, -2.43394714e+00, -1.46640561e-01,
 -2.62386821e-01],
[ 6.86617933e-02, -1.31979479e-01,  2.51221427e-01,
  3.95774101e-01],
[ 1.89829664e-01, -1.97355361e+00,  1.37546573e-01,
 -2.62386821e-01],
[ 3.10997534e-01, -3.62176246e-01,  5.35408562e-01,
  2.64141916e-01],
[-2.94841818e-01, -3.62176246e-01, -8.98031345e-02,
  1.32509732e-01],
[ 1.03800476e+00,  9.82172869e-02,  3.64896281e-01,
  2.64141916e-01],
[-2.94841818e-01, -1.31979479e-01,  4.21733708e-01,
  3.95774101e-01],
[-5.25060772e-02, -8.22569778e-01,  1.94384000e-01,
 -2.62386821e-01],
[ 4.32165405e-01, -1.97355361e+00,  4.21733708e-01,
  3.95774101e-01],
[-2.94841818e-01, -1.28296331e+00,  8.07091462e-02,
 -1.30754636e-01],
[ 6.86617933e-02,  3.28414053e-01,  5.92245988e-01,
  7.90670654e-01],
[ 3.10997534e-01, -5.92373012e-01,  1.37546573e-01,
  1.32509732e-01],
[ 5.53333275e-01, -1.28296331e+00,  6.49083415e-01,
  3.95774101e-01],
[ 3.10997534e-01, -5.92373012e-01,  5.35408562e-01,
  8.77547895e-04],
[ 6.74501145e-01, -3.62176246e-01,  3.08058854e-01,
  1.32509732e-01],
[ 9.16836886e-01, -1.31979479e-01,  3.64896281e-01,
  2.64141916e-01],
[ 1.15917263e+00, -5.92373012e-01,  5.92245988e-01,
  2.64141916e-01],
[ 1.03800476e+00, -1.31979479e-01,  7.05920842e-01,
  6.59038469e-01],
[ 1.89829664e-01, -3.62176246e-01,  4.21733708e-01,
  3.95774101e-01],
[-1.73673948e-01, -1.05276654e+00, -1.46640561e-01,
 -2.62386821e-01],
[-4.16009689e-01, -1.51316008e+00,  2.38717193e-02,
 -1.30754636e-01],
[-4.16009689e-01, -1.51316008e+00, -3.29657076e-02,
 -2.62386821e-01],
[-5.25060772e-02, -8.22569778e-01,  8.07091462e-02,
  8.77547895e-04],
[ 1.89829664e-01, -8.22569778e-01,  7.62758269e-01,
```

```
  5.27406285e-01],
[-5.37177559e-01, -1.31979479e-01,  4.21733708e-01,
  3.95774101e-01],
[ 1.89829664e-01,  7.88807586e-01,  4.21733708e-01,
  5.27406285e-01],
[ 1.03800476e+00,  9.82172869e-02,  5.35408562e-01,
  3.95774101e-01],
[ 5.53333275e-01, -1.74335684e+00,  3.64896281e-01,
  1.32509732e-01],
[-2.94841818e-01, -1.31979479e-01,  1.94384000e-01,
  1.32509732e-01],
[-4.16009689e-01, -1.28296331e+00,  1.37546573e-01,
  1.32509732e-01],
[-4.16009689e-01, -1.05276654e+00,  3.64896281e-01,
  8.77547895e-04],
[ 3.10997534e-01, -1.31979479e-01,  4.78571135e-01,
  2.64141916e-01],
[-5.25060772e-02, -1.05276654e+00,  1.37546573e-01,
  8.77547895e-04],
[-1.02184904e+00, -1.74335684e+00, -2.60315415e-01,
 -2.62386821e-01],
[-2.94841818e-01, -8.22569778e-01,  2.51221427e-01,
  1.32509732e-01],
[-1.73673948e-01, -1.31979479e-01,  2.51221427e-01,
  8.77547895e-04],
[-1.73673948e-01, -3.62176246e-01,  2.51221427e-01,
  1.32509732e-01],
[ 4.32165405e-01, -3.62176246e-01,  3.08058854e-01,
  1.32509732e-01],
[-9.00681170e-01, -1.28296331e+00, -4.30827696e-01,
 -1.30754636e-01],
[-1.73673948e-01, -5.92373012e-01,  1.94384000e-01,
  1.32509732e-01],
[ 5.53333275e-01,  5.58610819e-01,  1.27429511e+00,
  1.71209594e+00],
[-5.25060772e-02, -8.22569778e-01,  7.62758269e-01,
  9.22302838e-01],
[ 1.52267624e+00, -1.31979479e-01,  1.21745768e+00,
  1.18556721e+00],
[ 5.53333275e-01, -3.62176246e-01,  1.04694540e+00,
  7.90670654e-01],
[ 7.95669016e-01, -1.31979479e-01,  1.16062026e+00,
  1.31719939e+00],
[ 2.12851559e+00, -1.31979479e-01,  1.61531967e+00,
  1.18556721e+00],
[-1.14301691e+00, -1.28296331e+00,  4.21733708e-01,
  6.59038469e-01],
```

```
[ 1.76501198e+00, -3.62176246e-01,  1.44480739e+00,
  7.90670654e-01],
[ 1.03800476e+00, -1.28296331e+00,  1.16062026e+00,
  7.90670654e-01],
[ 1.64384411e+00,  1.24920112e+00,  1.33113254e+00,
  1.71209594e+00],
[ 7.95669016e-01,  3.28414053e-01,  7.62758269e-01,
  1.05393502e+00],
[ 6.74501145e-01, -8.22569778e-01,  8.76433123e-01,
  9.22302838e-01],
[ 1.15917263e+00, -1.31979479e-01,  9.90107977e-01,
  1.18556721e+00],
[-1.73673948e-01, -1.28296331e+00,  7.05920842e-01,
  1.05393502e+00],
[-5.25060772e-02, -5.92373012e-01,  7.62758269e-01,
  1.58046376e+00],
[ 6.74501145e-01,  3.28414053e-01,  8.76433123e-01,
  1.44883158e+00],
[ 7.95669016e-01, -1.31979479e-01,  9.90107977e-01,
  7.90670654e-01],
[ 2.24968346e+00,  1.70959465e+00,  1.67215710e+00,
  1.31719939e+00],
[ 2.24968346e+00, -1.05276654e+00,  1.78583195e+00,
  1.44883158e+00],
[ 1.89829664e-01, -1.97355361e+00,  7.05920842e-01,
  3.95774101e-01],
[ 1.28034050e+00,  3.28414053e-01,  1.10378283e+00,
  1.44883158e+00],
[-2.94841818e-01, -5.92373012e-01,  6.49083415e-01,
  1.05393502e+00],
[ 2.24968346e+00, -5.92373012e-01,  1.67215710e+00,
  1.05393502e+00],
[ 5.53333275e-01, -8.22569778e-01,  6.49083415e-01,
  7.90670654e-01],
[ 1.03800476e+00,  5.58610819e-01,  1.10378283e+00,
  1.18556721e+00],
[ 1.64384411e+00,  3.28414053e-01,  1.27429511e+00,
  7.90670654e-01],
[ 4.32165405e-01, -5.92373012e-01,  5.92245988e-01,
  7.90670654e-01],
[ 3.10997534e-01, -1.31979479e-01,  6.49083415e-01,
  7.90670654e-01],
[ 6.74501145e-01, -5.92373012e-01,  1.04694540e+00,
  1.18556721e+00],
[ 1.64384411e+00, -1.31979479e-01,  1.16062026e+00,
  5.27406285e-01],
[ 1.88617985e+00, -5.92373012e-01,  1.33113254e+00,
```

```
  9.22302838e-01],
[ 2.49201920e+00,  1.70959465e+00,  1.50164482e+00,
  1.05393502e+00],
[ 6.74501145e-01, -5.92373012e-01,  1.04694540e+00,
  1.31719939e+00],
[ 5.53333275e-01, -5.92373012e-01,  7.62758269e-01,
  3.95774101e-01],
[ 3.10997534e-01, -1.05276654e+00,  1.04694540e+00,
  2.64141916e-01],
[ 2.24968346e+00, -1.31979479e-01,  1.33113254e+00,
  1.44883158e+00],
[ 5.53333275e-01,  7.88807586e-01,  1.04694540e+00,
  1.58046376e+00],
[ 6.74501145e-01,  9.82172869e-02,  9.90107977e-01,
  7.90670654e-01],
[ 1.89829664e-01, -1.31979479e-01,  5.92245988e-01,
  7.90670654e-01],
[ 1.28034050e+00,  9.82172869e-02,  9.33270550e-01,
  1.18556721e+00],
[ 1.03800476e+00,  9.82172869e-02,  1.04694540e+00,
  1.58046376e+00],
[ 1.28034050e+00,  9.82172869e-02,  7.62758269e-01,
  1.44883158e+00],
[-5.25060772e-02, -8.22569778e-01,  7.62758269e-01,
  9.22302838e-01],
[ 1.15917263e+00,  3.28414053e-01,  1.21745768e+00,
  1.44883158e+00],
[ 1.03800476e+00,  5.58610819e-01,  1.10378283e+00,
  1.71209594e+00],
[ 1.03800476e+00, -1.31979479e-01,  8.19595696e-01,
  1.44883158e+00],
[ 5.53333275e-01, -1.28296331e+00,  7.05920842e-01,
  9.22302838e-01],
[ 7.95669016e-01, -1.31979479e-01,  8.19595696e-01,
  1.05393502e+00],
[ 4.32165405e-01,  7.88807586e-01,  9.33270550e-01,
  1.44883158e+00],
[ 6.86617933e-02, -1.31979479e-01,  7.62758269e-01,
  7.90670654e-01]])
```

Build our decision tree model for the purposes of extracting the feature importances which we can use to update the feature weighting in the hierarchical clustering demo

[5]:
```
from sklearn.model_selection import train_test_split
```

[6]:
```
X_train, X_test, y_train, y_test = train_test_split(
    X_scaled,
    y,
```

```
    test_size=0.3,
    random_state=42)
```

[7]:
```
from sklearn.tree import DecisionTreeClassifier
from sklearn.ensemble import RandomForestClassifier
```

Tree-based classifiers like DecisionTreeClassifier and RandomForestClassifier give us weights indicating how important each feature is using feature_importances_

[8]:
```
#clf = DecisionTreeClassifier()
clf = RandomForestClassifier()

clf = clf.fit(X_train, y_train)
```

[9]:
```
y_pred = clf.predict(X_test)
y_pred
```

[9]:
```
array([1, 0, 2, 1, 1, 0, 1, 2, 1, 1, 2, 0, 0, 0, 0, 1, 2, 1, 1, 2, 0, 2,
       0, 2, 2, 2, 2, 2, 0, 0, 0, 0, 1, 0, 0, 2, 1, 0, 0, 0, 2, 1, 1, 0,
       0])
```

[10]:
```
from sklearn.metrics import accuracy_score
```

Test that our model is actually working

[11]:
```
accuracy_score(y_test, y_pred)
```

[11]:
```
1.0
```

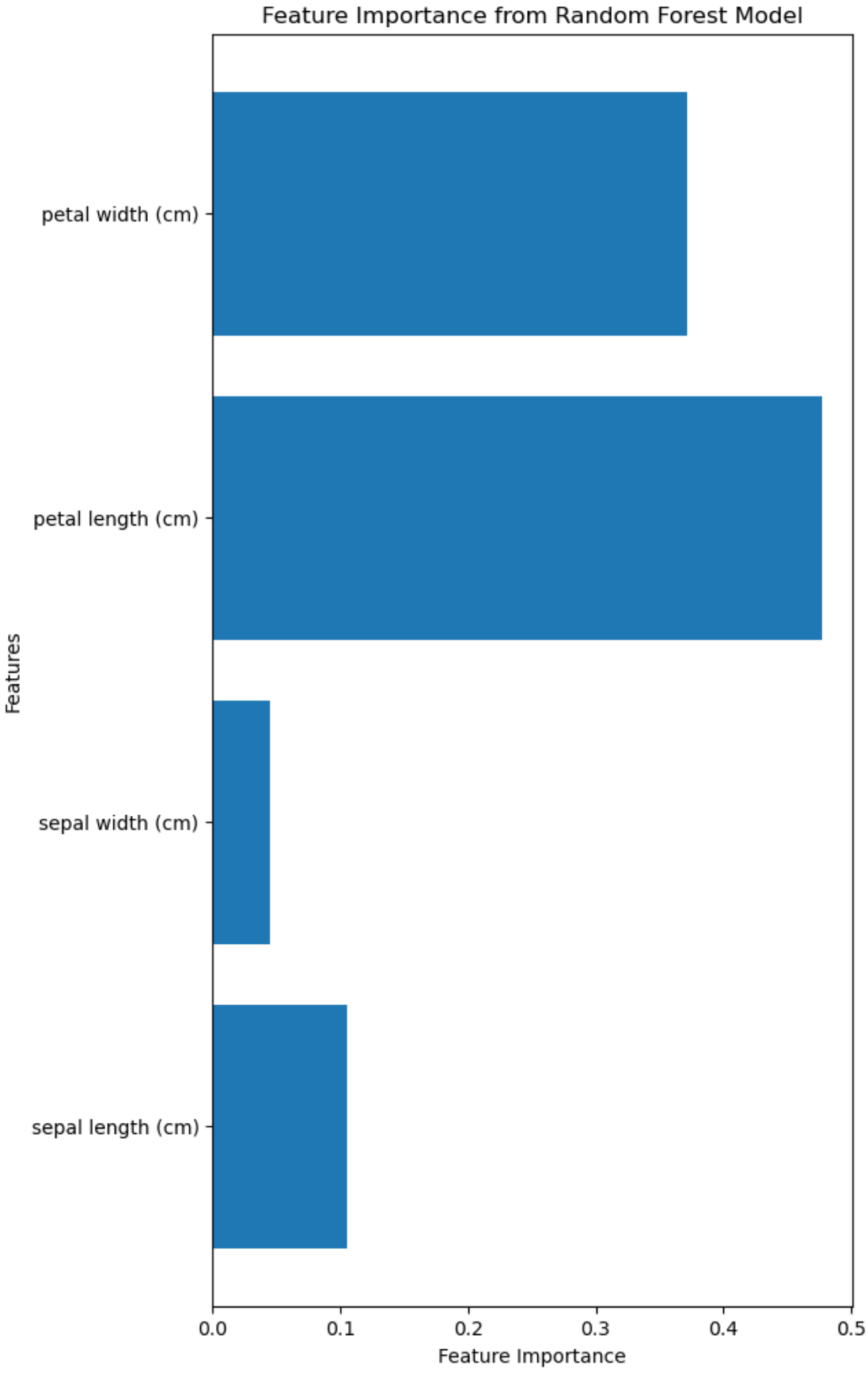
Get our feature importances and graph them for comparison

[12]:
```
clf.feature_importances_
```

[12]:
```
array([0.10587484, 0.04496103, 0.47740956, 0.37175456])
```

[13]:
```
import matplotlib.pyplot as plt
```

[14]:
```
plt.figure(figsize=(6, 12))
plt.barh(X.columns, clf.feature_importances_)
plt.xlabel('Feature Importance')
plt.ylabel('Features')
plt.title('Feature Importance from Random Forest Model')
plt.show()
```


Feature Importance from Random Forest Model
petal width (cm)
petal length (cm)
sepal width (cm)
sepal length (cm)
Features
0.0
0.1
0.2
0.3
0.4
0.5
Feature Importance

We can scale these so that the maximum value is 1, making it easier to compare their relative importance

[15]:
```
clf.feature_importances_ / max(clf.feature_importances_)
```

[15]:
```
array([0.22176942, 0.09417707, 1.        , 0.77869107])
```