

```
In [9]: import pandas as pd

In [13]: dataset = pd.read_csv("Iris.csv")

In [14]: dataset.head()

Out[14]:
   Id  SepalLengthCm  SepalWidthCm  PetalLengthCm  PetalWidthCm  Species
0    1             5.1           3.5           1.4           0.2  Iris-setosa
1    2             4.9           3.0           1.4           0.2  Iris-setosa
2    3             4.7           3.2           1.3           0.2  Iris-setosa
3    4             4.6           3.1           1.5           0.2  Iris-setosa
4    5             5.0           3.6           1.4           0.2  Iris-setosa

In [15]: dataset.shape

Out[15]:
(150, 6)

In [17]: from sklearn.tree import DecisionTreeClassifier
from sklearn.tree import export_graphviz
from sklearn.model_selection import train_test_split

In [20]: featurenames = ['SepalLengthCm', 'SepalWidthCm', 'PetalLengthCm', 'PetalWidthCm']
#separating features
features = dataset.loc[:, featurenames]
#separating target
labels = dataset.loc[:, 'Species']

In [22]: from sklearn.preprocessing import StandardScaler
from sklearn.decomposition import PCA

In [23]: features = StandardScaler().fit_transform(features)
print(features)

[[-9.00681170e-01  1.03205722e+00 -1.34127240e+00 -1.31297673e+00]
 [-1.14301691e+00 -1.24957601e-01 -1.34127240e+00 -1.31297673e+00]
 [-1.38535265e+00  3.37848329e-01 -1.39813811e+00 -1.31297673e+00]
 [-1.50652052e+00  1.06445364e-01 -1.28440670e+00 -1.31297673e+00]
 [-1.02184904e+00  1.26346019e+00 -1.34127240e+00 -1.31297673e+00]
 [-5.37177559e-01  1.95766909e+00 -1.17067529e+00 -1.05080379e+00]
 [-1.50652052e+00  8.00654259e-01 -1.34127240e+00 -1.18150376e+00]
 [-1.02184904e+00  8.00654259e-01 -1.28440670e+00 -1.31297673e+00]
 [-1.74885626e+00 -3.5630566e-01 -1.34127240e+00 -1.31297673e+00]
 [-1.14301691e+00  1.06445364e-01 -1.28440670e+00 -1.44444970e+00]
 [-5.37177559e-01  1.49486315e+00 -1.28440670e+00 -1.31297673e+00]
 [-1.26418478e+00  8.00654259e-01 -1.22754100e+00 -1.31297673e+00]
 [-1.26418478e+00 -1.24957601e-01 -1.34127240e+00 -1.44444970e+00]
 [-1.87002413e+00 -1.24957601e-01 -1.51186952e+00 -1.44444970e+00]
 [-5.25060772e-02  2.18907295e+00 -1.45500381e+00 -1.31297673e+00]
 [-1.73673948e-01  3.11468331e+00 -1.28440670e+00 -1.05080379e+00]
 [-5.37177559e-01  1.95766909e+00 -1.39813811e+00 -1.05080379e+00]
 [-9.00681170e-01  1.03205722e+00 -1.34127240e+00 -1.18150376e+00]
 [-1.73673948e-01  1.72626612e+00 -1.17067529e+00 -1.18150376e+00]
 [-9.00681170e-01  1.72626612e+00 -1.28440670e+00 -1.18150376e+00]
 [-5.37177559e-01  8.00654259e-01 -1.17067529e+00 -1.31297673e+00]
 [-9.00681170e-01  1.49486315e+00 -1.28440670e+00 -1.05080379e+00]
 [-1.50652052e+00  1.26346019e+00 -1.50873522e+00 -1.31297673e+00]
 [-9.00681170e-01  5.69251294e-01 -1.17067529e+00 -9.18557817e-01]
 [-1.26418478e+00  8.00654259e-01 -1.05694380e+00 -1.31297673e+00]
 [-1.02184904e+00 -1.24957601e-01 -1.22754100e+00 -1.31297673e+00]
 [-1.02184904e+00  8.00654259e-01 -1.22754100e+00 -1.05080379e+00]
 [-7.79513300e-01  1.03205722e+00 -1.28440670e+00 -1.31297673e+00]
 [-7.79513300e-01  8.00654259e-01 -1.34127240e+00 -1.31297673e+00]
 [-1.38535265e+00  3.37848329e-01 -1.22754100e+00 -1.31297673e+00]
 [-1.26418478e+00  1.06445364e-01 -1.22754100e+00 -1.31297673e+00]
 [-5.37177559e-01  8.00654259e-01 -1.28440670e+00 -1.05080379e+00]
 [-1.73673948e-01  2.42047502e+00 -1.28440670e+00 -1.44444970e+00]
 [-4.16090689e-01  2.65187798e+00 -1.34127240e+00 -1.31297673e+00]
 [-1.14301691e+00  1.06445364e-01 -1.28440670e+00 -1.44444970e+00]
 [-1.02184904e+00  3.37848329e-01 -1.45500381e+00 -1.31297673e+00]
 [-4.16090689e-01  1.03205722e+00 -1.39813811e+00 -1.31297673e+00]
 [-1.14301691e+00  1.06445364e-01 -1.28440670e+00 -1.44444970e+00]
 [-1.74885626e+00 -1.24957601e-01 -1.39813811e+00 -1.31297673e+00]
 [-9.00681170e-01  8.00654259e-01 -1.28440670e+00 -1.31297673e+00]
 [-1.02184904e+00  1.03205722e+00 -1.39813811e+00 -1.18150376e+00]
 [-1.02708839e+00 -1.74477836e+00 -1.39813811e+00 -1.18150376e+00]
 [-1.74885626e+00  3.37848329e-01 -1.39813811e+00 -1.31297673e+00]
 [-1.02184904e+00  1.03205722e+00 -1.22754100e+00 -7.87084847e-01]
 [-9.00681170e-01  1.72626612e+00 -1.05694380e+00 -1.05080379e+00]
 [-1.26418478e+00 -1.24957601e-01 -1.34127240e+00 -1.18150376e+00]
 [-9.00681170e-01  1.72626612e+00 -1.22754100e+00 -1.31297673e+00]
 [-1.50652052e+00  3.37848329e-01 -1.34127240e+00 -1.31297673e+00]
 [-6.58345239e-01  1.49486315e+00 -1.28440670e+00 -1.31297673e+00]
 [-1.14301691e+00  5.69251294e-01 -1.34127240e+00 -1.31297673e+00]
 [-1.40150837e+00  3.37848329e-01  5.35295827e-01  2.64698913e-01]
 [ 6.74501145e-01  3.37848329e-01  4.21564419e-01  3.96171883e-01]
 [ 1.28034050e+00  1.06445364e-01  6.49027235e-01  3.96171883e-01]
 [-4.16090689e-01 -1.74477836e+00  1.37235899e-01  1.33225943e-01]
 [ 7.9569016e-01 -5.87763531e-01  4.7502123e-01  3.96171883e-01]
 [-1.73673948e-01 -5.87763531e-01  4.21564419e-01  1.33225943e-01]
 [ 5.5333275e-01  5.69251294e-01  5.35295827e-01  5.27644853e-01]
 [-1.14301691e+00 -1.5337539e+00 -2.60824029e-01 -2.61192967e-01]
 [ 8.06617933e-02 -3.37848329e-01  5.92161531e-01  7.90590793e-01]
 [ 3.10997534e-01 -5.87763531e-01  1.37235899e-01  1.33225943e-01]
 [-1.73673948e-01 -5.87763531e-01  4.7502123e-01  3.96171883e-01]
 [ 5.5333275e-01 -1.28197243e+00  6.49027235e-01  3.96171883e-01]
 [ 3.10997534e-01 -5.87763531e-01  5.35295827e-01  1.75297293e-03]
 [ 6.74501145e-01 -3.5630566e-01  3.07833011e-01  1.33225943e-01]
 [ 1.05080379e-01 -1.24957601e-01  3.64698715e-01  2.64698913e-01]
 [ 1.15917263e+00 -5.87763531e-01  5.92161531e-01  2.64698913e-01]
 [ 1.03800476e+00 -1.24957601e-01  7.05892939e-01  6.59117823e-01]
 [-1.73673948e-01 -3.5630566e-01  4.21564419e-01  3.96171883e-01]
 [-1.73673948e-01 -1.05056946e+00 -1.47092621e-01 -2.61192967e-01]
 [-4.16090689e-01 -1.5137539e+00 -2.35044910e-02 -1.29719997e-01]
 [-4.16090689e-01 -1.5137539e+00 -3.33612130e-02 -2.61192967e-01]
 [-5.25060772e-02 -8.19166497e-01  8.03701950e-02  1.75297293e-03]
 [ 7.9569016e-01 -8.19166497e-01  7.62758643e-01  5.27644853e-01]
 [-5.37177559e-01 -1.24957601e-01  4.21564419e-01  3.96171883e-01]
 [ 1.89829664e-01  8.00654259e-01  4.21564419e-01  5.27644853e-01]
 [ 5.5333275e-01 -1.74477836e+00  3.64698715e-01  1.33225943e-01]
 [-2.94841818e-01 -1.24957601e-01  1.94011603e-01  1.33225943e-01]
 [-4.16090689e-01 -1.28197243e+00  1.37235899e-01  1.33225943e-01]
 [-4.16090689e-01 -1.05056946e+00  3.64698715e-01  1.75297293e-03]
 [ 3.10997534e-01 -1.24957601e-01  1.60801057e+00  2.64698913e-01]
 [-5.25060772e-02 -1.05056946e+00  1.37235899e-01  1.75297293e-03]
 [-1.02184904e+00 -1.74477836e+00 -2.60824029e-01 -2.61192967e-01]
 [-2.94841818e-01 -8.19166497e-01  2.50907307e-01  1.33225943e-01]
 [-1.73673948e-01 -1.24957601e-01  2.50907307e-01  1.75297293e-03]
 [-1.73673948e-01 -3.5630566e-01  2.50907307e-01  1.33225943e-01]
 [ 4.32165405e-01 -5.87763531e-01  3.07833011e-01  1.33225943e-01]
 [-9.00681170e-01 -1.28197243e+00 -4.31421141e-01 -1.29719997e-01]
 [ 1.73673948e-01 -5.87763531e-01  1.94011603e-01  1.33225943e-01]
 [ 5.5333275e-01 -5.69251294e-01  1.94011603e-01  1.71090158e+00]
 [-5.25060772e-02 -8.19166497e-01  7.62758643e-01  9.22063763e-01]
 [ 1.52267624e+00 -1.24957601e-01  1.21768427e+00  1.18500970e+00]
 [ 5.5333275e-01 -3.5630566e-01  1.04708716e+00  7.90590793e-01]
 [ 7.9569016e-01 -1.24957601e-01  1.10801057e+00  1.31648267e+00]
 [ 2.12851559e+00 -1.24957601e-01  1.61574420e+00  1.18500970e+00]
 [-1.14301691e+00 -1.28197243e+00  4.21564419e-01  6.59117823e-01]
 [ 1.76501198e+00 -3.5630566e-01  1.44514709e+00  7.90590793e-01]
 [ 1.03800476e+00 -1.28197243e+00  1.16080105e+00  7.90590793e-01]
 [ 1.04384411e+00 -1.26346019e+00  1.33141568e+00  1.71090158e+00]
 [ 7.9569016e-01  3.37848329e-01  7.62758643e-01  1.05353673e+00]
 [ 6.74501145e-01 -8.19166497e-01  8.764900951e-01  9.22063763e-01]
 [ 1.15917263e+00 -1.24957601e-01  9.90221459e-01  1.18500970e+00]
 [ 1.03800476e+00 -1.24957601e-01  1.06801057e+00  1.05353673e+00]
 [-5.25060772e-02 -5.87763531e-01  7.62758643e-01  1.57942861e+00]
 [ 6.74501145e-01  3.37848329e-01  8.764900951e-01  1.44795564e+00]
 [ 7.9569016e-01 -1.24957601e-01  9.90221459e-01  7.90590793e-01]
 [ 2.24968346e+00 -1.72626612e+00  1.67260991e+00  1.31648267e+00]
 [ 2.24968346e+00 -1.05056946e+00  1.78634131e+00  1.44795564e+00]
 [ 1.89829664e-01 -1.97618132e+00  7.05892939e-01  3.96171883e-01]
 [ 1.28034050e+00  3.37848329e-01  1.10395287e+00  1.44795564e+00]
 [ 2.94841818e-01 -5.87763531e-01  1.10395287e+00  1.05353673e+00]
 [ 2.24968346e+00 -5.87763531e-01  1.67260991e+00  1.05353673e+00]
 [ 5.5333275e-01 -8.19166497e-01  6.49027235e-01  7.90590793e-01]
 [ 1.03800476e+00 -5.69251294e-01  1.10395287e+00  1.18500970e+00]
 [ 1.04384411e+00 -1.24957601e-01  1.27540908e+00  7.90590793e-01]
 [ 4.32165405e-01 -5.87763531e-01  5.92161531e-01  7.90590793e-01]
 [ 3.10997534e-01 -1.24957601e-01  6.49027235e-01  7.90590793e-01]
 [ 6.74501145e-01 -5.87763531e-01  1.04708716e+00  1.18500970e+00]
 [ 1.04384411e+00 -1.24957601e-01  1.16080105e+00  5.27644853e-01]
 [ 1.05080379e+00 -5.87763531e-01  1.33141568e+00  9.22063763e-01]
 [ 2.49201920e+00  1.72626612e+00  1.50201270e+00  1.05353673e+00]
 [ 6.74501145e-01 -5.87763531e-01  1.04708716e+00  1.31648267e+00]
 [ 5.5333275e-01 -5.87763531e-01  7.62758643e-01  3.96171883e-01]
 [ 3.10997534e-01 -1.05056946e+00  1.04708716e+00  2.64698913e-01]
 [ 2.24968346e+00 -1.24957601e-01  1.33141568e+00  1.44795564e+00]
 [ 5.5333275e-01  8.00654259e-01  1.04708716e+00  1.57942861e+00]
 [ 6.74501145e-01  1.06445364e-01  9.90221459e-01  7.90590793e-01]
 [ 1.89829664e-01 -1.24957601e-01  5.92161531e-01  7.90590793e-01]
 [ 1.28034050e+00  1.06445364e-01  9.3335755e-01  1.18500970e+00]
 [ 1.03800476e+00  1.06445364e-01  1.04708716e+00  1.57942861e+00]
 [ 1.28034050e+00  1.06445364e-01  7.62758643e-01  1.44795564e+00]
 [-5.25060772e-02 -8.19166497e-01  7.62758643e-01  9.22063763e-01]
 [ 1.15917263e+00  3.37848329e-01  1.21768427e+00  1.44795564e+00]
 [ 1.03800476e+00  5.69251294e-01  1.10395287e+00  1.71090158e+00]
 [ 1.03800476e+00 -1.24957601e-01  8.19024347e-01  1.44795564e+00]
 [ 5.5333275e-01 -1.28197243e+00  7.05892939e-01  9.22063763e-01]
 [ 7.9569016e-01 -1.24957601e-01  8.19024347e-01  1.05353673e+00]
 [ 4.32165405e-01  8.00654259e-01  9.3335755e-01  1.44795564e+00]
 [ 6.86617933e-02 -1.24957601e-01  7.62758643e-01  7.90590793e-01]]

In [25]: import numpy as np
pca = PCA(n_components=2)
pct = pca.fit_transform(features)
print(np.cumsun(pca.explained_variance_ratio_))
principal_df = pd.DataFrame(pct, columns=['pc1', 'pc2'])
display(principal_df)
finaldf = pd.concat([principal_df, labels], axis=1)

[0.72770452 0.95809975]

   pc1    pc2
0 -2.264542  0.505704
1 -2.086426 -0.655405
2 -2.367950 -0.318477
3 -2.304197 -0.575368
4 -2.388777  0.674767
...
145 1.870522  0.382822
146 1.558492 -0.905314
147 1.520845  0.266795
148 1.376391  1.016362
149 0.959299 -0.022284

150 rows x 2 columns

In [26]: import matplotlib.pyplot as plt

In [27]: print(principal_df)

   pc1    pc2
0 -2.264542  0.505704
1 -2.086426 -0.655405
2 -2.367950 -0.318477
3 -2.304197 -0.575368
4 -2.388777  0.674767
...
145 1.870522  0.382822
146 1.558492 -0.905314
147 1.520845  0.266795
148 1.376391  1.016362
149 0.959299 -0.022284

[150 rows x 2 columns]

In [34]: display(finaldf)

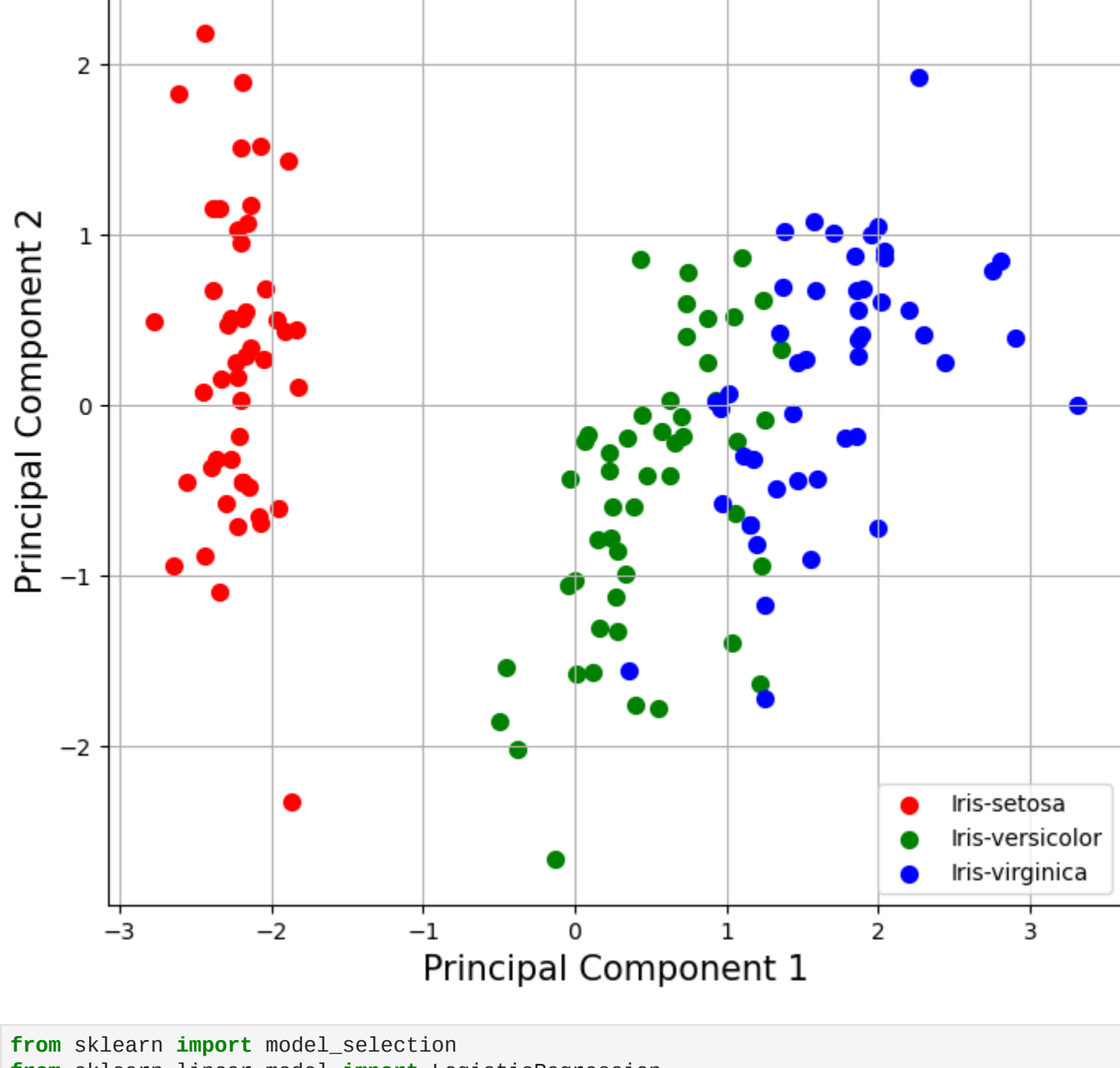
   pc1    pc2  Species
0 -2.264542  0.505704  Iris-setosa
1 -2.086426 -0.655405  Iris-setosa
2 -2.367950 -0.318477  Iris-setosa
3 -2.304197 -0.575368  Iris-setosa
4 -2.388777  0.674767  Iris-setosa
...
145 1.870522  0.382822  Iris-virginica
146 1.558492 -0.905314  Iris-virginica
147 1.520845  0.266795  Iris-virginica
148 1.376391  1.016362  Iris-virginica
149 0.959299 -0.022284  Iris-virginica

150 rows x 3 columns

In [31]: import seaborn as sns

In [37]: fig = plt.figure(figsize = (8,8))
ax = fig.add_subplot(1,1,1)
ax.set_xlabel('Principal Component 1', fontsize = 15)
ax.set_ylabel('Principal Component 2', fontsize = 15)
ax.set_title('2 component PCA', fontsize = 20)
targets = ['Iris-setosa', 'Iris-versicolor', 'Iris-virginica']
colors = ['r', 'g', 'b']
for target, color in zip(targets, colors):
    indicesTokeep = labels == target
    ax.scatter(finaldf.loc[indicesTokeep, 'pc1'], finaldf.loc[indicesTokeep, 'pc2'], c = color, s = 50)
    ax.legend(targets)
    ax.grid()

2 component PCA



In [38]: from sklearn import model_selection
from sklearn.linear_model import LogisticRegression
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score

features = finaldf[['pc1', 'pc2']]
labels = finaldf[['Species']]

x_train, x_test, y_train, y_test = train_test_split(features, label, test_size = 0.35, random_state=42)
model = LogisticRegression()
model.fit(x_train, y_train)

train_pred = model.predict(x_train)
test_pred = model.predict(x_test)

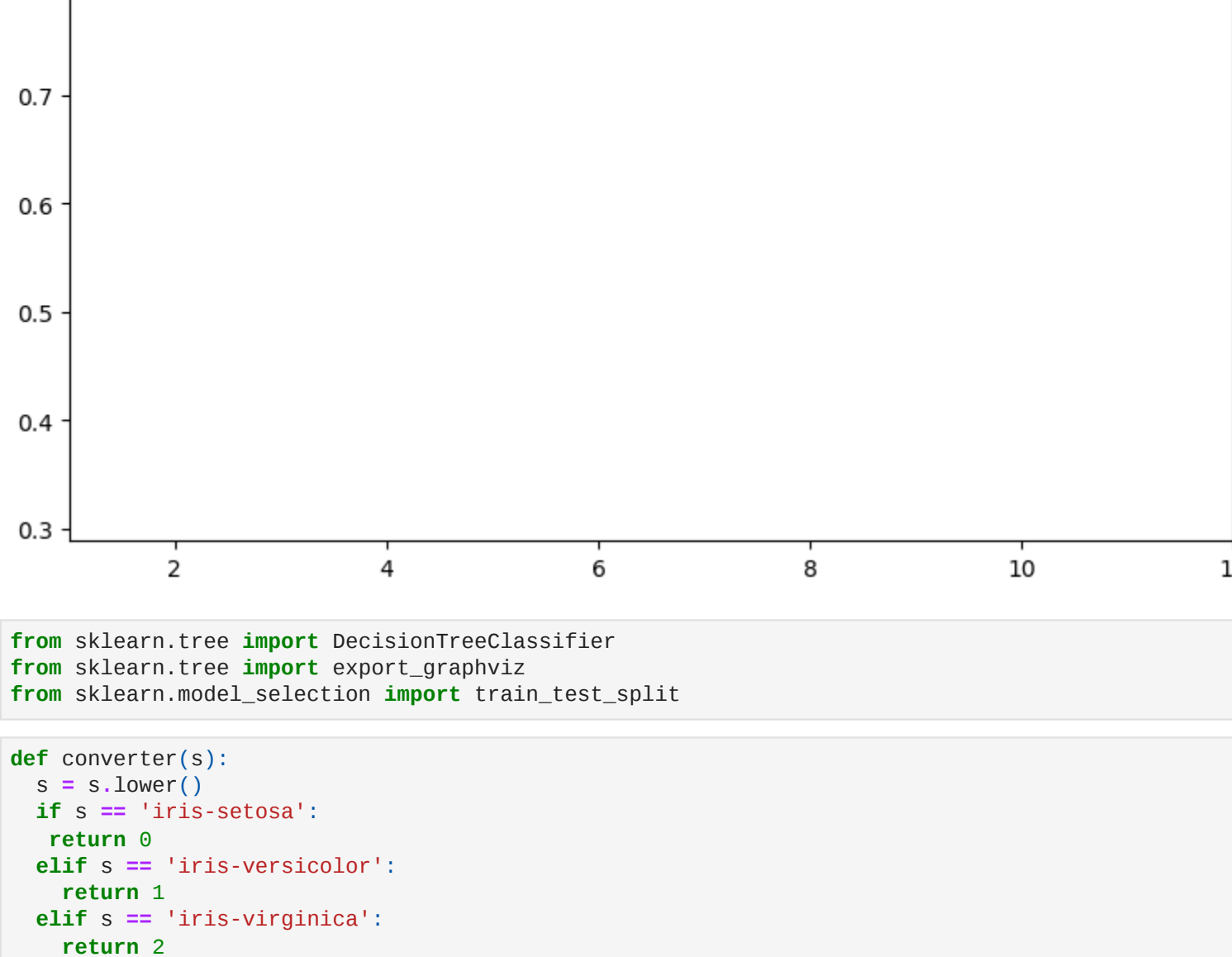
print(f"Train Accuracy: {accuracy_score(y_train, train_pred):.80:.2f}")
print(f"Test Accuracy: {accuracy_score(y_test, test_pred):.80:.2f}")

Train Accuracy: 89.69
Test Accuracy: 92.45

In [39]: from pandas.core.internals.construction import range_to_ndarray
from sklearn.neighbors import KNeighborsClassifier

train_acc = []
test_acc = []
k_range = list(range(1, 90, 2))
for k in k_range:
    model = KNeighborsClassifier(k)
    model.fit(x_train, y_train)
    y_pred_test = model.predict(x_test)
    y_pred_train = model.predict(x_train)
    train_acc.append(accuracy_score(y_train, y_pred_train))
    test_acc.append(accuracy_score(y_test, y_pred_test))

plt.figure(figsize=(20, 6))
plt.subplot(122)
plt.xlim(1,12)
plt.subplot(122)
plt.plot(k_range, test_acc)
plt.xlim(1,12)
plt.show()



In [40]: from sklearn.tree import DecisionTreeClassifier
from sklearn.tree import export_graphviz
from sklearn.model_selection import train_test_split

In [41]: def convert(s):
s = s.lower()
if s == 'Iris-setosa':
    return 0
elif s == 'Iris-versicolor':
    return 1
elif s == 'Iris-virginica':
    return 2
else:
    return -1

In [43]: features = dataset[['SepalLengthCm', 'SepalWidthCm', 'PetalLengthCm', 'PetalWidthCm']]
labels = dataset['Species']

In [44]: from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(features, labels, test_size = 0.2, random_state=42)

In [45]: model = DecisionTreeClassifier(criterion='entropy', max_depth=3, random_state=32)
model.fit(X_train, y_train)

Out[45]:
DecisionTreeClassifier

DecisionTreeClassifier(criterion='entropy', max_depth=3, random_state=32)

In [46]: model.predict(X_test)

Out[46]:
array(['Iris-versicolor', 'Iris-setosa', 'Iris-virginica',
       'Iris-versicolor', 'Iris-versicolor', 'Iris-setosa',
       'Iris-versicolor', 'Iris-virginica', 'Iris-versicolor',
       'Iris-setosa', 'Iris-virginica', 'Iris-setosa', 'Iris-setosa',
       'Iris-versicolor', 'Iris-versicolor', 'Iris-virginica',
       'Iris-setosa', 'Iris-virginica', 'Iris-setosa', 'Iris-virginica',
       'Iris-virginica', 'Iris-virginica', 'Iris-virginica',
       'Iris-virginica', 'Iris-setosa', 'Iris-setosa'], dtype=object)
```