import matplotlib.pyplot as plt

from mpl\_toolkits.mplot3d import Axes3D

from sklearn import datasets

from sklearn.decomposition import PCA

# import some data to play with

iris = datasets.load\_iris()

X = iris.data[:, :2] # we only take the first two features.

y = iris.target

x\_min, x\_max = X[:, 0].min() - .5, X[:, 0].max() + .5

y\_min, y\_max = X[:, 1].min() - .5, X[:, 1].max() + .5

plt.figure(2, figsize=(8, 6))

plt.clf()

# Plot the training points

plt.scatter(X[:, 0], X[:, 1], c=y, cmap=plt.cm.Set1,

edgecolor='k')

plt.xlabel('Sepal length')

plt.ylabel('Sepal width')

plt.xlim(x\_min, x\_max)

plt.ylim(y\_min, y\_max)

plt.xticks(())

plt.yticks(())

# To getter a better understanding of interaction of the dimensions

# plot the first three PCA dimensions

fig = plt.figure(1, figsize=(8, 6))

ax = Axes3D(fig, elev=-150, azim=110)

X\_reduced = PCA(n\_components=3).fit\_transform(iris.data)

ax.scatter(X\_reduced[:, 0], X\_reduced[:, 1], X\_reduced[:, 2], c=y,

cmap=plt.cm.Set1, edgecolor='k', s=40)

ax.set\_title("First three PCA directions")

ax.set\_xlabel("1st eigenvector")

ax.w\_xaxis.set\_ticklabels([])

ax.set\_ylabel("2nd eigenvector")

ax.w\_yaxis.set\_ticklabels([])

ax.set\_zlabel("3rd eigenvector")

ax.w\_zaxis.set\_ticklabels([])

plt.show()