In [1]:
```
import numpy as np
import matplotlib.pyplot as plt
from mpl_toolkits.mplot3d import Axes3D
from sklearn import decomposition
from sklearn import datasets
```

In [2]:
```
np.random.seed(5)

centers = [[1, 1], [-1, -1], [1, -1]]
iris = datasets.load_iris()
```

In [3]:
```
X = iris.data
y = iris.target
```

In [4]:
```
fig = plt.figure(1, figsize=(6, 4))
plt.clf()
ax = Axes3D(fig, rect=[0, 0, .95, 1], elev=48, azim=134)
```

In [5]:
```
plt.cla()
pca = decomposition.PCA(n_components=3)
pca.fit(X)
X = pca.transform(X)
```

In [6]:
```
X.shape
```

Out[6]:
```
(150, 3)
```

In [7]:
```
for name, label in [('Setosa', 0), ('Versicolour', 1), ('Virginica', 2)
]:
    ax.text3D(X[y == label, 0].mean(),
              X[y == label, 1].mean() + 1.5,
              X[y == label, 2].mean(), name,
              horizontalalignment='center',
              bbox=dict(alpha=.5, edgecolor='w', facecolor='w'))
y = np.choose(y, [1, 2, 0]).astype(np.float)
```

In [ ]:
```
ax.scatter(X[:, 0], X[:, 1], X[:, 2], c=y,cmap=plt.cm.spectral,
           edgecolor='k')
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()
```