

In [5]: `%matplotlib notebook`

## PCA Example

In [6]:

```
import numpy as np
import matplotlib.pyplot as plt

from sklearn.decomposition import PCA
from sklearn import datasets

np.random.seed(0)
```

## Generate/load data for regression

### 1. Load datasets available in sklearn

In [7]:

```
from sklearn import datasets
X, y = datasets.load_iris(return_X_y=True)
```

In [9]:

```
fig = plt.figure()
ax = fig.add_subplot(111, projection="3d", elev=48, azimuth=134)
ax.set_position([0, 0, 0.95, 1])

plt.cla()
pca = PCA(n_components=2)
pca.fit(X)
X_pca = pca.transform(X)

for name, label in [("Setosa", 0), ("Versicolour", 1), ("Virginica", 2)]:
    ax.text3D(
        X[y == label, 0].mean(),
        X[y == label, 1].mean(),
        X[y == label, 2].mean(), name,
        # horizontalalignment="center",
        bbox=dict(alpha=0.5, edgecolor="w", facecolor="w"),
    )

# Plot data with colors matching the cluster predictions
ax.scatter(X[:, 0], X[:, 1], X[:, 2], c=np.choose(y, [1, 2, 0]).astype(float),
           cmap=plt.cm.jet, edgecolor="k")

ax.w_xaxis.set_ticklabels([])
ax.w_yaxis.set_ticklabels([])
ax.w_zaxis.set_ticklabels([])

plt.show()
```

