

## pca

April 7, 2025

```
[1]: import numpy as np

X = np.array([
    [3, 22000, 1],
    [6, 75000, 0],
    [2, 14000, 1],
    [8, 85000, 0],
    [4, 50000, 1]
], dtype=float)

# Step 1: Standardize
mean = np.mean(X, axis=0)
std = np.std(X, axis=0)
X_std = (X - mean) / std

# Step 2: Covariance Matrix
cov_matrix = np.cov(X_std.T)

# Step 3: Eigen decomposition
eigen_values, eigen_vectors = np.linalg.eig(cov_matrix)

# Step 4: Sort eigenvalues and eigenvectors
idx = np.argsort(eigen_values)[::-1]
eigen_values = eigen_values[idx]
eigen_vectors = eigen_vectors[:, idx]

# Step 5: Project onto top 2 components
X_pca = np.dot(X_std, eigen_vectors[:, :2])
print("Reduced X (PCA):\n", X_pca)
```

Reduced X (PCA):

```
[[ 1.46060919 -0.02977285]
 [-1.6093965  -0.3597175 ]
 [ 1.89708745 -0.31961626]
 [-2.35803385  0.12179788]
 [ 0.60973371  0.58730873]]
```