Supplementary Material

1 THREE PCA METHODS IMPLEMENTED WITH NUMPY IN PYTHON

We can obtain the PCA components ($\mathbf{U}$), the eigenvectors ($\mathbf{V}$), and the eigenvalues ($\mathbf{\Lambda} = \mathbf{S}^2$) of a $m \times n$ matrix $\mathbf{X}$ via the following three methods with the NumPy linear algebra. Let consider $\mathbf{X}$ defined as:

```
import numpy as np
np.random.seed(50)
n = 3; m = 10
X = np.random.random((m , n))
```

Singular Value Decomposition (SVD). The SVD of $\mathbf{X} = \mathbf{U}\mathbf{S}\mathbf{V}^\intercal$ can be conducted as follows:

```
U, S, VT = np.linalg.svd(X, full_matrices=False)
V = VT.T
eigenVectors = V
eigenValues = S**2
```

Eigendecomposition. The eigenvalue decomposition (EVD) of the covariance matrix $\mathbf{C} = \mathbf{X}^\intercal\mathbf{X} = \mathbf{V}\mathbf{\Lambda}\mathbf{V}^\intercal$ and the least-squares solution ($\mathbf{U}$) to $\mathbf{X}\mathbf{V} = \mathbf{U}\mathbf{S}$ can be computed as follows:

```
C = np.dot(X.T, X)
eigenValues, eigenVectors = np.linalg.eigh(C)
idx_sort = eigenValues.argsort()[::-1]
eigenValues = eigenValues[idx_sort]
eigenVectors = eigenVectors[:,idx_sort]
S = eigenValues**0.5
V = eigenVectors
XV = np.dot(X,V)
UT, resid, rank, sv = np.linalg.lstsq(np.diag(S),XV.T, rcond=None)
U = UT.T
```

QR Decomposition. The QR decomposition of $\mathbf{X} = \mathbf{Q}\mathbf{R}$ and the SVD of $\mathbf{R}^\intercal = \bar{\mathbf{U}}\bar{\mathbf{S}}\bar{\mathbf{V}}^\intercal$ can also be implemented using the NumPy functions, leading to $\mathbf{S} = \bar{\mathbf{S}}$, $\mathbf{V} = \bar{\mathbf{U}}$, and $\mathbf{U} = \mathbf{Q}\bar{\mathbf{V}}$:

```
Q, R = np.linalg.qr(X)
U_bar, S_bar, VT_bar = np.linalg.svd(R.T)
V_bar = VT_bar.T
U = np.dot(Q, V_bar)
V = U_bar
S = S_bar
eigenVectors = V
eigenValues = S**2
```

These different methods can sometimes result in a sign difference in one PCA component and its eigenvector compared to each other.