Multi-dimensional Scaling

Debajyoti Mondal

University of Saskatchewan

Given Data → Find Plot (Classic – Multi-Dimensional-Scaling)

Algorithm

- Set up the matrix of squared distances $\mathbf{D}^2$
- Apply the double centering: $B = -\frac{1}{2} J\, \mathbf{D}^2 J$

$B = X X^T$

- Compute m eigenvalues and m eigenvectors
- Multiply eigenvectors and sqrt of eigenvalues

Classic Multi-Dimensional-Scaling (MDS)

- Started from the matrix of pairwise Euclidean distances D^2
- Computed pseudo scalar product matrix B from D^2
- Found out interesting vectors that gave us the coordinates.

n^2

$O(n^2 n)$

- What would you do if the distances are not Euclidean?

- PCA (fast)
- $\boxed{\text{Non-metric}}$ MDS (iterative, slow, better result than PCA)

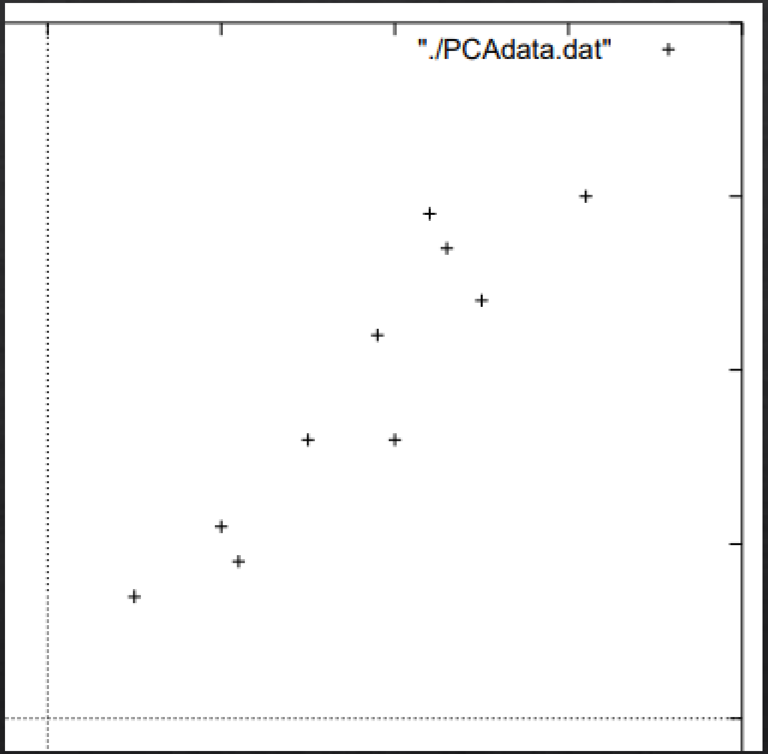
Principal Component Analysis (PCA)

- Get some data
- Subtract the mean from each dimension
- Calculate covariance matrix
- Find interesting vectors that will represent the principal components (distinguishing feature)

Data =

x	y
2.5	2.4
0.5	0.7
2.2	2.9
1.9	2.2
3.1	3.0
2.3	2.7
2	1.6
1	1.1
1.5	1.6
1.1	0.9

$\bar{X}$ $\bar{Y}$



Principal Component Analysis (PCA)

- Get some data
- Subtract the mean from each dimension
- Calculate covariance matrix
- Find interesting vectors that will represent the principal components (distinguishing feature)

Data =

x	y
2.5	2.4
0.5	0.7
2.2	2.9
1.9	2.2
3.1	3.0
2.3	2.7
2	1.6
1	1.1
1.5	1.6
1.1	0.9

$x - \bar{x}$

DataAdjust =

x	y
.69	.49
-1.31	-1.21
.39	.99
.09	.29
1.29	1.09
.49	.79
.19	-.31
-.81	-.81
-.31	-.31
-.71	-1.01

Principal Component Analysis (PCA)

- Get some data
- Subtract the mean from each dimension
- Calculate covariance matrix
- Find interesting vectors that will represent the principal components (distinguishing feature)

$d_1 \dots d_n$

$$\begin{array}{c} d_1 \\ \vdots \\ d_n \end{array}\left[\; R^2 \;\right]$$

dimension

$$s^2 = \frac{\sum_{i=1}^{n}(X_i - \bar{X})^2}{(n-1)}$$

$$var(X) = \frac{\sum_{i=1}^{n}(X_i - \bar{X})(X_i - \bar{X})}{(n-1)}$$

$$cov(X,Y) = \frac{\sum_{i=1}^{n}(X_i - \bar{X})(Y_i - \bar{Y})}{(n-1)}$$

Covariance is a measure of the joint variability of two random variables.

Principal Component Analysis (PCA)

- Get some data
- Subtract the mean from each dimension
- Calculate covariance matrix → XX^T ?
- Find interesting vectors that will represent the principal components (distinguishing feature)

$$s^2 = \frac{\sum_{i=1}^{n}(X_i - \bar{X})^2}{(n-1)}$$

$$var(X) = \frac{\sum_{i=1}^{n}(X_i - \overset{0}{\cancel{\bar{X}}})(X_i - \overset{0}{\cancel{\bar{X}}})}{(n-1)}$$

XX^T

$$cov(X,Y) = \frac{\sum_{i=1}^{n}(X_i - \overset{0}{\cancel{\bar{X}}})(Y_i - \overset{0}{\cancel{\bar{Y}}})}{(n-1)}$$

Covariance is a measure of the joint variability of two random variables.

Principal Component Analysis (PCA)

- Get some data
- Subtract the mean from each dimension
- Calculate covariance matrix
- Find interesting vectors that will represent the principal components (distinguishing feature)

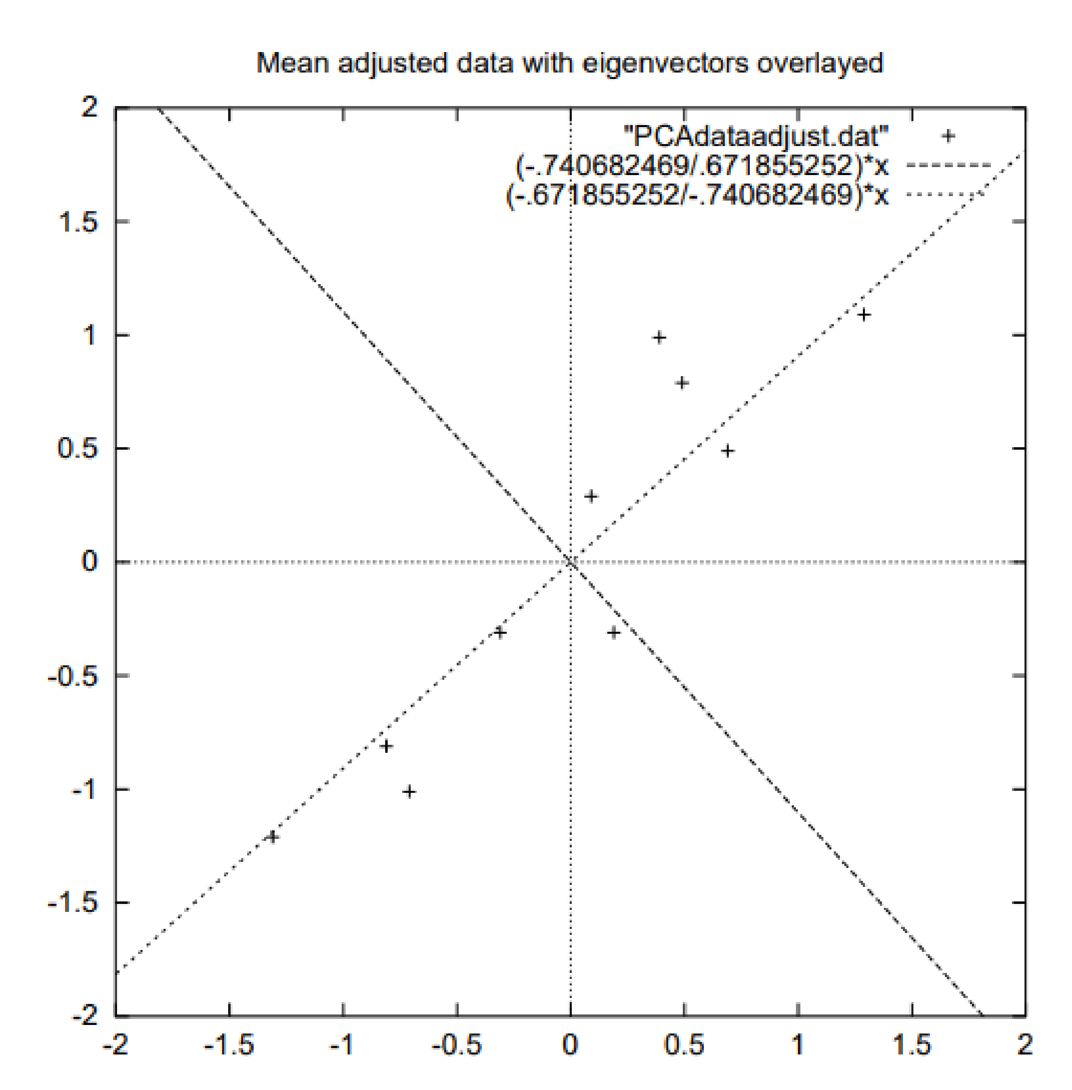
$$cov = \begin{array}{ccc} & d_1 & d_2 \\ d_1 & \multicolumn{2}{c}{\multirow{2}{*}{$\begin{pmatrix} .616555556 & .615444444 \\ .615444444 & .716555556 \end{pmatrix}$}} \\ d_2 & & \end{array}$$

$$eigenvalues = \begin{pmatrix} .0490833989 \\ 1.28402771 \end{pmatrix}$$

$$eigenvectors = \begin{pmatrix} -.735178656 & -.677873399 \\ .677873399 & -.735178656 \end{pmatrix}$$

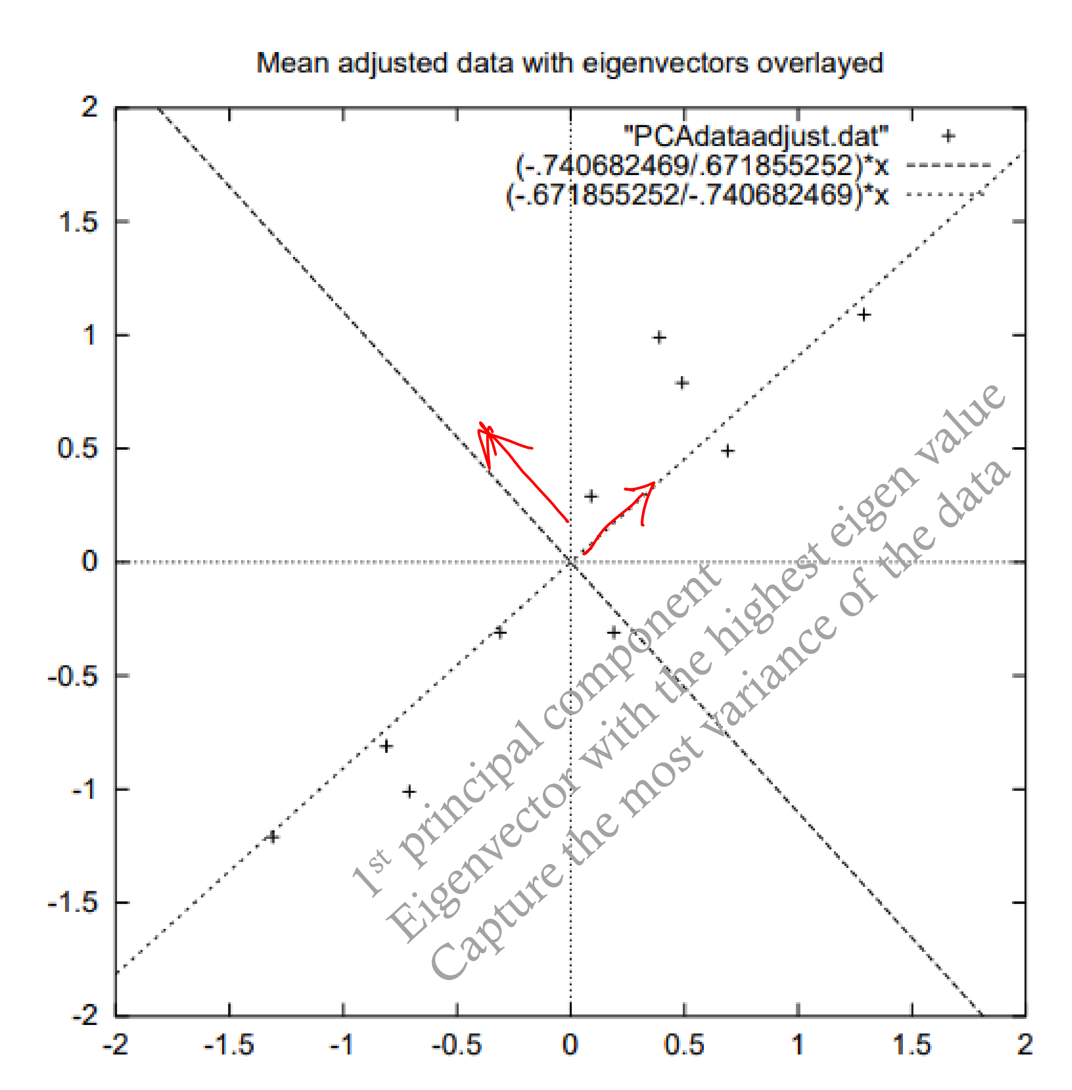
Principal Component Analysis (PCA)

- Get some data
- Subtract the mean from each dimension
- Calculate covariance matrix
- Find interesting vectors that will represent the principal components (distinguishing feature)



Principal Component Analysis (PCA)

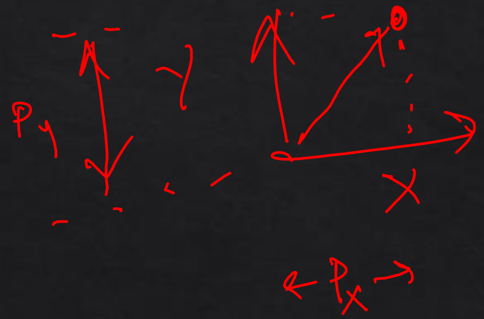
- Get some data
- Subtract the mean from each dimension
- Calculate covariance matrix
- Find interesting vectors that will represent the principal components (distinguishing feature)

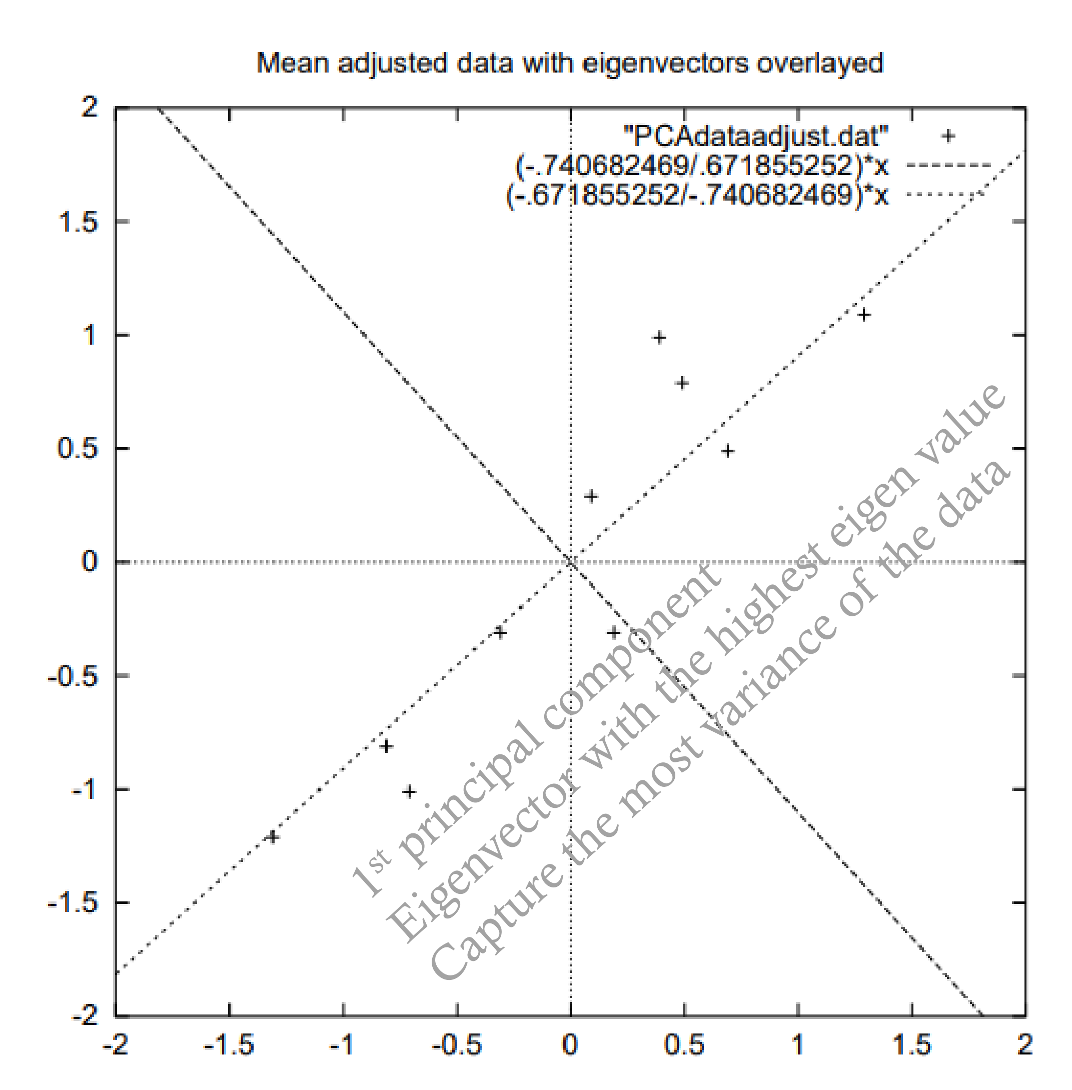


Principal Component Analysis (PCA)

- We now have a ordered list of eigenvectors – in this case only 2 vectors – shown in dashed lines

- We now want to construct feature of the datapoints

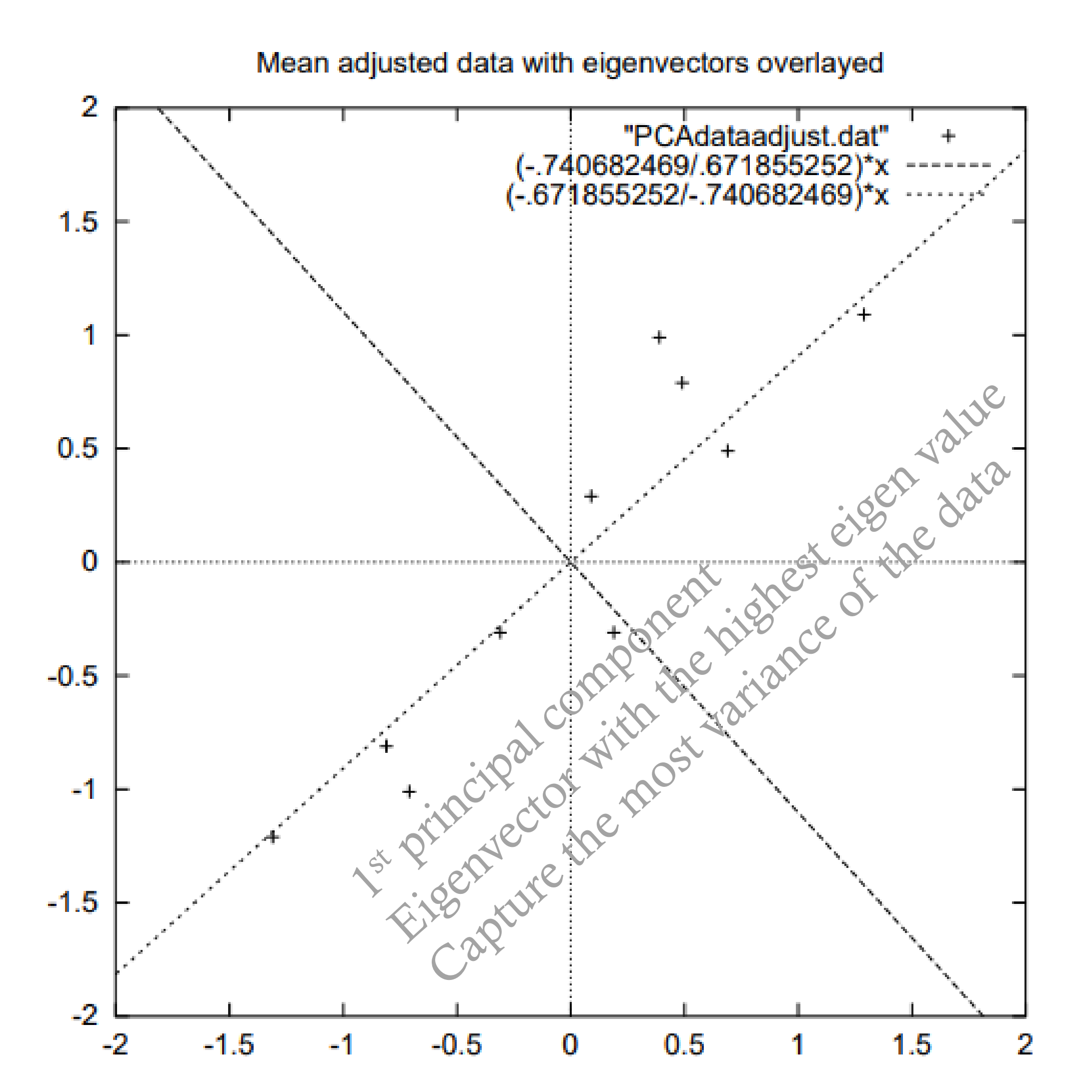




Principal Component Analysis (PCA)

- We now have a ordered list of eigenvectors – in this case only 2 vectors – shown in dashed lines

- We now want to construct a feature of the datapoints by choosing the most important *k* eigenvectors



Principal Component Analysis (PCA)

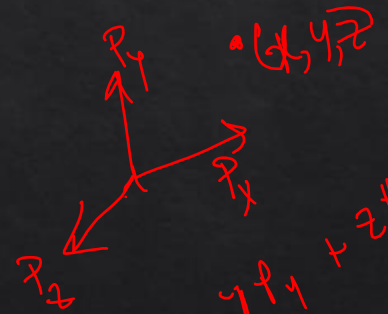
- We now have a ordered list of eigenvectors – in this case only 2 vectors – shown in dashed lines

$$eigenvalues = \begin{pmatrix} .0490833989 \\ \boxed{1.28402771} \end{pmatrix}$$

$$eigenvectors = \begin{pmatrix} -.735178656 & \boxed{\begin{matrix} -.677873399 \\ -.735178656 \end{matrix}} \\ .677873399 & \end{pmatrix}$$

- We now want to construct a feature of the datapoints by choosing the most important *k* eigenvectors

$$\begin{pmatrix} -.677873399 \\ -.735178656 \end{pmatrix}$$



$xP_x + yP_y + zP_z$

Principal Component Analysis (PCA)

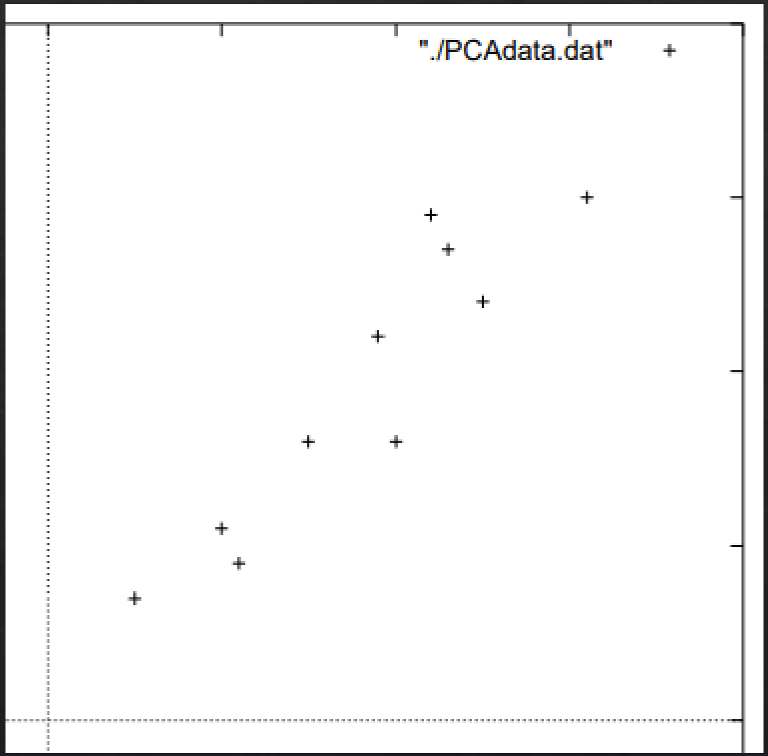
- Multiply the 'mean-subtracted data' with the selected vector to get the selected feature for each data point
- Since we have only one eigenvector, we will get one feature per datapoint

DataAdjust =

x	y
.69	.49
-1.31	-1.21
.39	.99
.09	.29
1.29	1.09
.49	.79
.19	-.31
-.81	-.81
-.31	-.31
-.71	-1.01

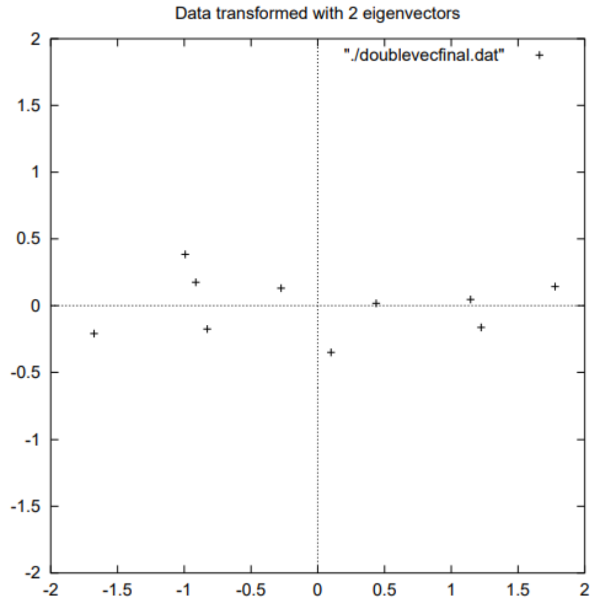
$$\begin{pmatrix} -.677873399 \\ -.735178656 \end{pmatrix}$$

Final Result



Transformed Data=

x	y
-.827970186	-.175115307
1.77758033	.142857227
-.992197494	.384374989
-.274210416	.130417207
-1.67580142	-.209498461
-.912949103	.175282444
.0991094375	-.349824698
1.14457216	.0464172582
.438046137	.0177646297
1.22382056	-.162675287



Classic Multi-Dimensional-Scaling (MDS) vs Principal Component Analysis (PCA)

- PCA minimizes dimensions, preserving covariance of data
- MDS minimizes dimensions, preserving distance between data points
- They are same, if covariance in data = Euclidean distance between data points in high dimension
- They are different, if distance measure is different.

Classic Multi-Dimensional-Scaling (MDS) vs Principal Component Analysis (PCA)

- Started from the matrix of pairwise Euclidean distances D^2
- Computed pseudo scalar product matrix B from D^2
- Found out interesting vectors that gave us the coordinates.

- Get some data
- Subtract the mean from each dimension
- Calculate covariance matrix
- Find interesting vectors that will represent the principal components (distinguishing feature)

n → data points

D → dimensions

	PCA	MDS
	Covariance matrix (D x D)	Gram matrix (n x n)
Computation	$O((n+d)D^2)$	$O((D+d)n^2)$

faster

$n >>> D$

Learning Goal

- Understand dataset types and structural properties of data
- Learn the double centering operation
- Able to relate distance square matrix to scalar product matrix
- Describe the basic idea behind multi-dimensional scaling
- Get familiar with the classic multi-dimensional scaling algorithm