Machine Learning

Dimensionality Reduction with PCA

Dr. Sherif Saad

Learning Objectives

Understand how unsupervised learning could be used for feature extraction and feature reduction.

Understand how we can apply the principle component analysis as a feature extraction technique.

Outlines

Dimensionality Reduction

Principal Component Analysis

How PCA Works?

Why PCA Works?

Dimensionality Reduction

Is the second most important application of unsupervised learning is dimensionality reduction.

Most of the time we are working with data of high dimensionality where each observation or sample comes with hundreds of features.

A large number of features usually will face challenges such as limited storage space and computational performance of the machine learning algorithm.

Unsupervised dimensionality reduction techniques are commonly applied as data preprocessing step to eliminate noise and compress the feature space while retaining most of the relevant information

Dimensionality Reduction

Unsupervised dimensionality reduction is a feature extraction technique to reduce the number of features in the dataset.

In feature selection techniques, we maintain the original feature set and only select the strongest features.

In feature extraction transform or project the data onto a new feature space.

dimensionality reduction or feature extraction is an approach to data compression with the goal of maintaining most of the relevant information.

Transforming Data

Transformer Algorithm: is any algorithm you apply to your dataset that changes either the feature count or feature values but does not alter the number of observations.

There are two general uses of transformer:

- Data cleaning where you mung your data as a pre-processing step before applying a ML model to it.
- Dimensionality reduction, where the number of features in your dataset is intelligently reduced to a subset of the original.

Principal Component Analysis

Is an unsupervised linear transformation technique that helps us to identify patterns in data based on the correlation between features.

PCA aims to find the directions of maximum variance in high-dimensional data and projects it onto a new subspace with fewer dimensions than the original one.

In this context PCA is a data transformation algorithm that transform data from one feature space to a new one.

Principal Component Analysis

It attempts to convert your possibly correlated features into a set of linearly uncorrelated ones.

PCA calculates those *best* view angle (e.g. Utility Pole front view and sky view)
It models a linear subspace of your data by capturing its greatest variability.

It investigate the **covariance** structure directly using matrix calculations and eigenvectors to compute the *best* unique features that describe your samples.

The PCA assume there is a linear relation between the features in the feature space.

Principal Component Analysis

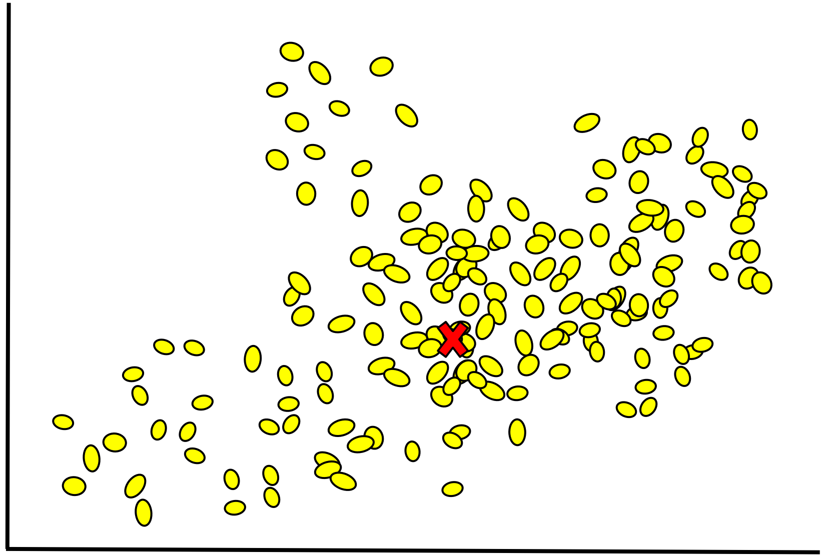
How it works?

First Find the center of your data, based on its numeric features. Next, it would search for the direction that has the most variance or widest spread of values. That direction is the principal component vector. The PCA algorithm added to a list of principal components.

By searching for more directions of maximal variance that are orthogonal to all previously computed vectors, a more principal component can then be added to the list. This set of vectors forms a new feature space that you can represent your samples with.

PCA : How it Works

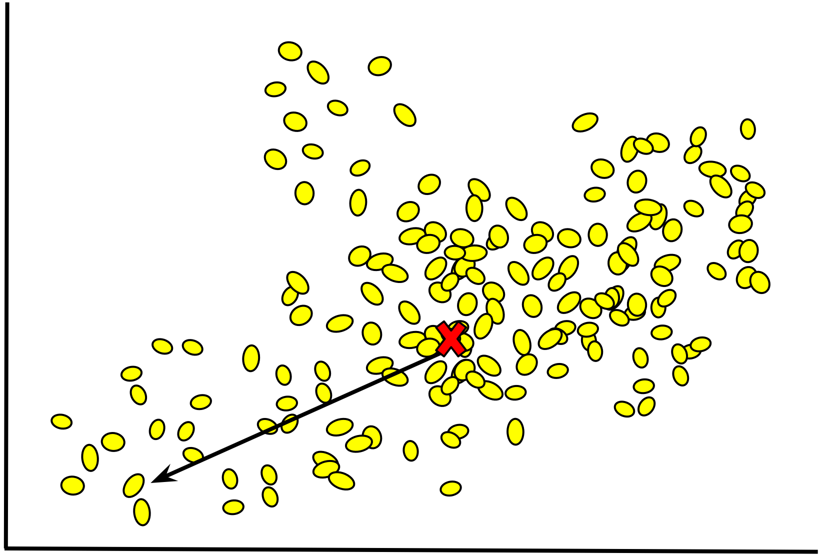
Step 1: Find the center of your data by calculating the mean.



PCA : How it Works

Step 1: Find the center of your data by calculating the mean.

Step 2: Find the direction of maximum variance, or the direction that has the most range of data. This is the principal component vector.

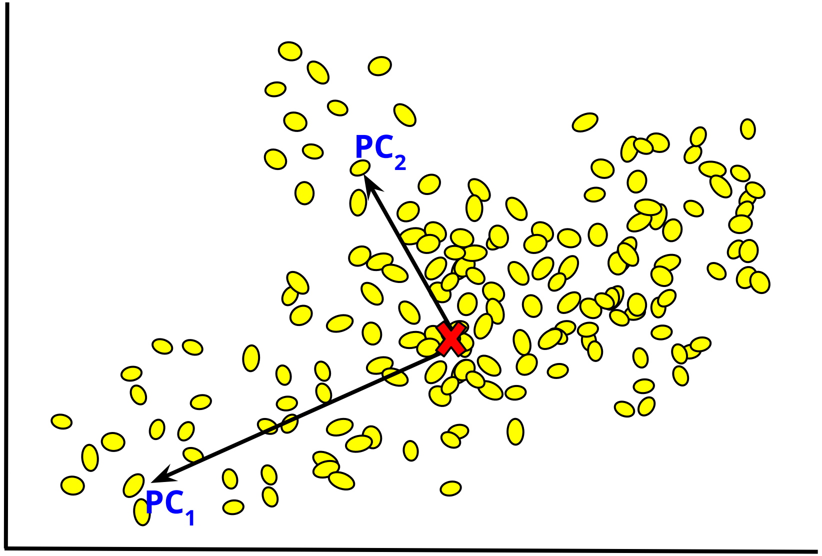


PCA : How it Works

Step 1: Find the center of your data by calculating the mean.

Step 2: Find the direction of maximum variance, or the direction that has the most range of data. This is the principal component vector.

Step 3: it continues looking for additional directions of maximum variance. And it does this with one condition, which is that any new directions that it finds has to be orthogonal to every other directions that it's previously found.



PCA : How it Works

We construct a $d \times k$ -dimensional transformation matrix W that allows us to map a sample vector x onto a new k -dimensional feature subspace that has fewer dimensions than the original d -dimensional feature space.

The first principal component will have the largest possible variance, and all consequent principal components will have the largest possible variance given that they are uncorrelated (orthogonal) to the other principal components.

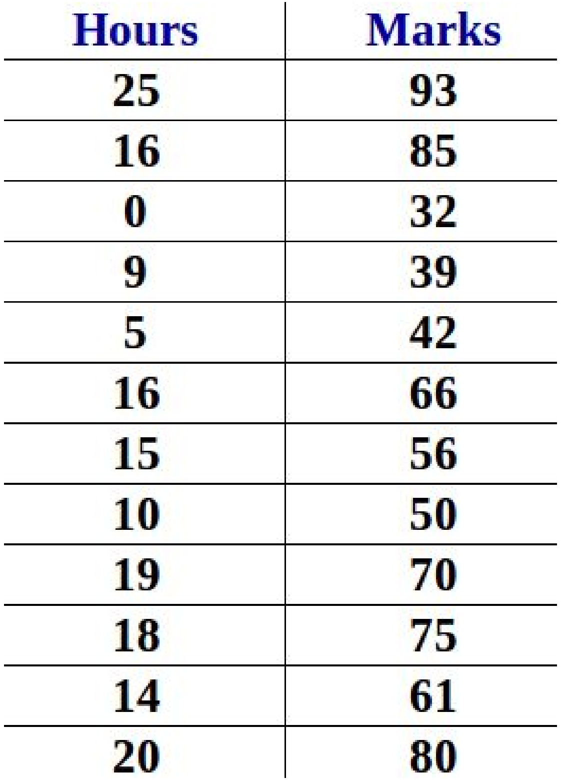
The PCA directions are highly sensitive to data scaling, and we need to standardize the features prior to PCA if the features were measured on different scales and we want to assign equal importance to all features

PCA Algorithm

1. Standardize the d -dimensional dataset.
2. Construct the covariance matrix.
3. Decompose the covariance matrix into its eigenvectors and eigenvalues.
4. Select k eigenvectors that correspond to the k largest eigenvalues, where k is the dimensionality of the new feature subspace (k ≤ d).
5. Construct a projection matrix W from the "top" k eigenvectors.
6. Transform the d -dimensional input dataset X using the projection matrix W to obtain the new k -dimensional feature subspace.

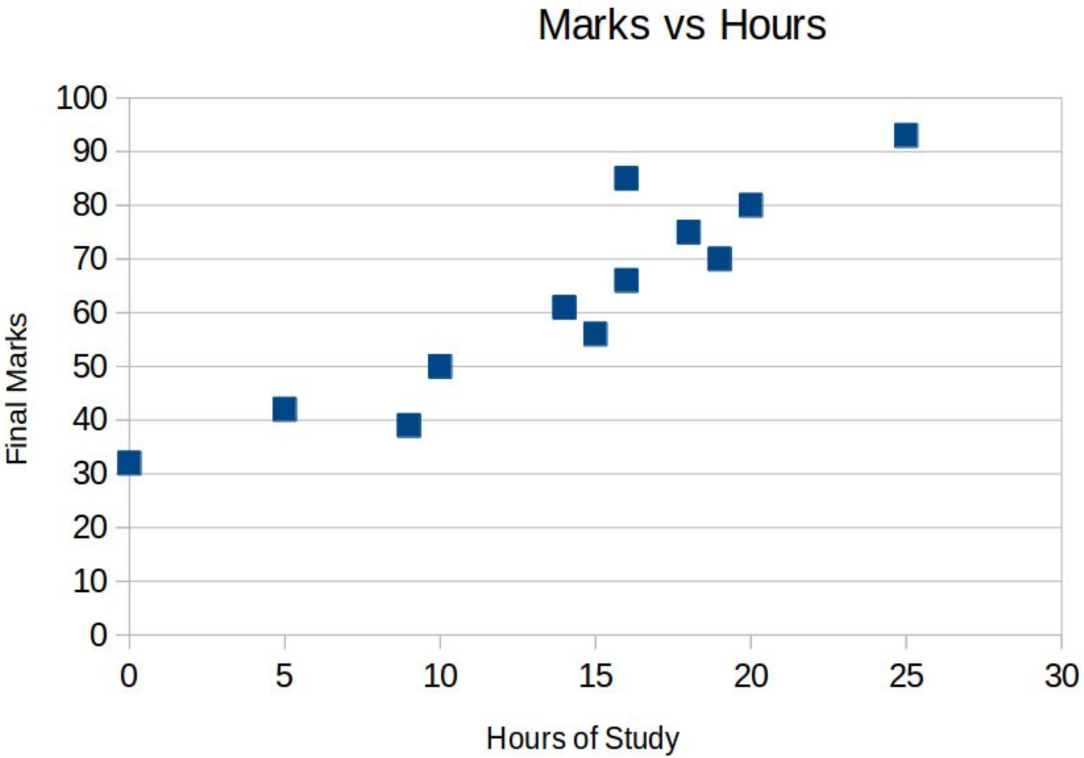
Why the PCA works?

Let us assume we have the number of hours the students in CENG420 spent to prepare for the final quiz and the total marks in the final quiz. What is the relation between these two variables?



Hours	Marks
25	93
16	85
0	32
9	39
5	42
16	66
15	56
10	50
19	70
18	75
14	61
20	80

Why the PCA works?



Why the PCA works?

The mean is given by

$$\bar{X} = \frac{\sum_{i=1}^{n} X_i}{n}$$

The variance is given by

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

Why the PCA works?

We usually use the covariance to see if there is any relationship between the variables (features). Does the number of hours has any effect on their marks in the quiz.

The covariance find out how much the features vary from the mean with respect to each other.

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

Why the PCA works?

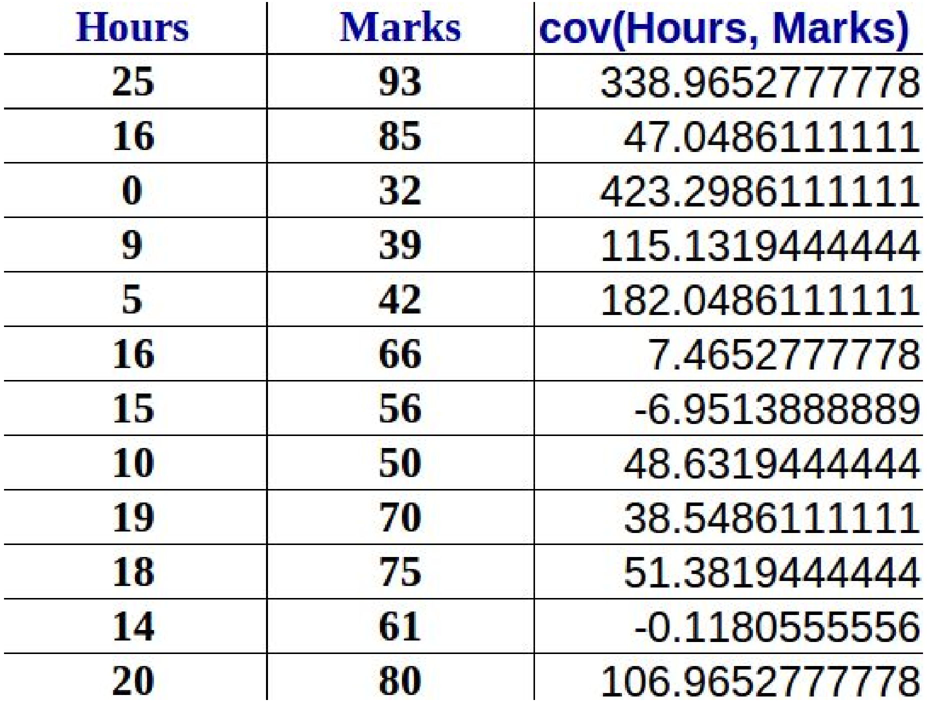
By calculating the covariance between two features we can learn about the relation between them.

If the value of the covariance is positive cov(X,Y) > 0, then that indicates that both features increase together.

If the value of the covariance is negative cov(X,Y) < 0, then hat indicates that when one feature increase the other feature decrease.

If the value of the covariance equal zero, then the two feature are independent of each other.

Why the PCA works?



Hours	Marks	cov(Hours, Marks)
25	93	338.9652777778
16	85	47.0486111111
0	32	423.2986111111
9	39	115.1319444444
5	42	182.0486111111
16	66	7.4652777778
15	56	-6.9513888889
10	50	48.6319444444
19	70	38.5486111111
18	75	51.3819444444
14	61	-0.1180555556
20	80	106.9652777778

Cov (hours, marks) = 122.84)

Why the PCA works?

What if we have more than 2 features in our dataset?

Well if we have more than 2 features such x, y, and z then we have more than one covariance. We have cov(x, y), cov (x, z), and cov(y, z).

In this case we can represent all the possible covariance values between all the different features in a matrix. This matrix is the covariance matrix.

$$cov(x,y,z) = \begin{pmatrix} cov(x,x) & cov(x,y) & cov(x,z) \\ cov(y,x) & cov(y,y) & cov(y,z) \\ cov(z,x) & cov(z,y) & cov(z,z) \end{pmatrix}$$

Why the PCA works?

Eigenvectors: a vector that when operated on by a given operator gives a scalar multiple of itself.

$$\begin{bmatrix} 2 & 3 \\ 2 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 \\ 2 \end{bmatrix} = \begin{bmatrix} 8 \\ 4 \end{bmatrix}$$

$$\begin{bmatrix} 2 & 3 \\ 2 & 1 \end{bmatrix} \cdot \begin{bmatrix} 9 \\ 6 \end{bmatrix} = \begin{bmatrix} 36 \\ 24 \end{bmatrix}$$

Why the PCA works?

Eigenvectors: a vector that when operated on by a given operator gives a scalar multiple of itself.

$$\begin{bmatrix} 2 & 3 \\ 2 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 \\ 2 \end{bmatrix} = \begin{bmatrix} 8 \\ 4 \end{bmatrix}$$

In the first case the resulting vector is not an integer multiple of the original vector.

$$\begin{bmatrix} 2 & 3 \\ 2 & 1 \end{bmatrix} \cdot \begin{bmatrix} 9 \\ 6 \end{bmatrix} = \begin{bmatrix} 36 \\ 24 \end{bmatrix}$$

In the second case the resulting vector is 4 times the original vector

Why the PCA works?

The square matrix is a transformation matrix. If you multiply this matrix on the left of a vector, the answer is another vector that is transformed from it's original position.

$$\begin{bmatrix} 2 & 3 \\ 2 & 1 \end{bmatrix} \cdot \begin{bmatrix} 9 \\ 6 \end{bmatrix} = \begin{bmatrix} 36 \\ 24 \end{bmatrix}$$

Eigenvectors can only be found for square matrices.

Note: not every square matrix has eigenvectors. All the eigenvectors of a matrix are orthogonals

Why the PCA works?

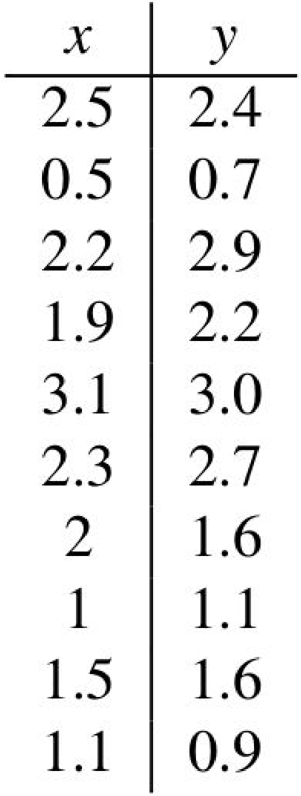
The eigenvalue is a scalar associated with a given linear transformation of a vector space.

$$\begin{bmatrix} 2 & 3 \\ 2 & 1 \end{bmatrix} \cdot \begin{bmatrix} 9 \\ 6 \end{bmatrix} = \begin{bmatrix} 36 \\ 24 \end{bmatrix}$$

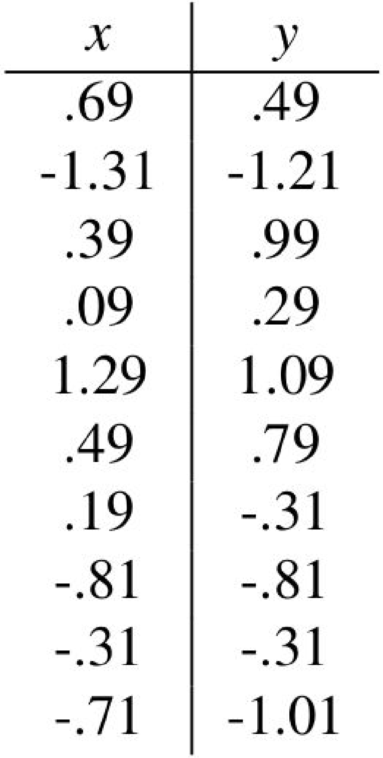
Each eigenvector has an eigenvalue associated with it. For example the eigenvalue of the vector [9, 6] is 4.

PCA How it works (Again)?

Give a data set of two feature space x and y. To apply PCA we first scale the feature values of each feature by subtracting the mean.



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



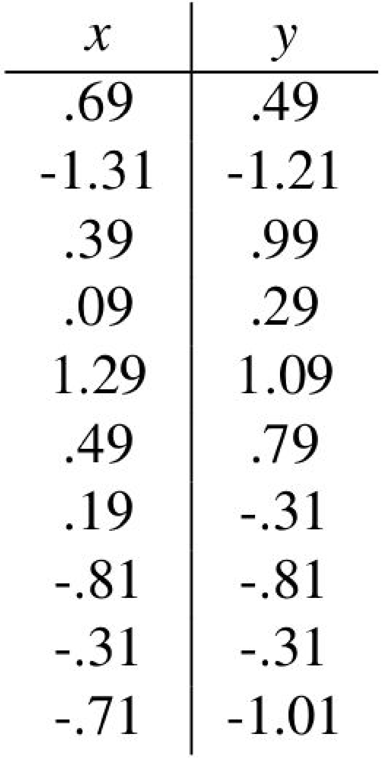
x	y
.69	.49
-1.31	-1.21
.39	.99
.09	.29
1.29	1.09
.49	.79
.19	-.31
-.81	-.81
-.31	-.31
-.71	-1.01

PCA How it works (Again)?

Then, we calculate the covariance matrix of the scaled dataset.

$$cov = \begin{pmatrix} .616555556 & .615444444 \\ .615444444 & .716555556 \end{pmatrix}$$

The non-diagonal elements in the covariance matrix are positive. What does that mean? What is the relation between x and y?



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

PCA How it works (Again)?

The next step we calculate the eigenvectors and the eigenvalues of the covariance matrix

$$cov = \begin{pmatrix} .616555556 & .615444444 \\ .615444444 & .716555556 \end{pmatrix}$$

This will gave us the following eigenvectors and eigenvalues

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

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

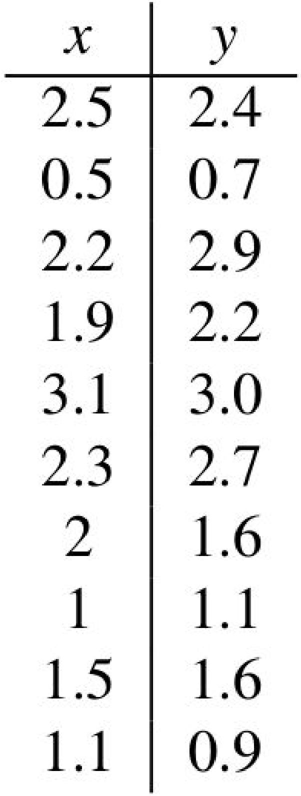
PCA How it works (Again)?

Now, the eigenvector with the highest eigenvalue is the principle component of the data set.

After calculating the eigenvectors from the covariance matrix we order them by eigenvalue in descending order (from highest to lowest).

We can decide to ignore the components of lesser significance (lowest eigenvalue) based on some threshold.

PCA How it works (Again)?

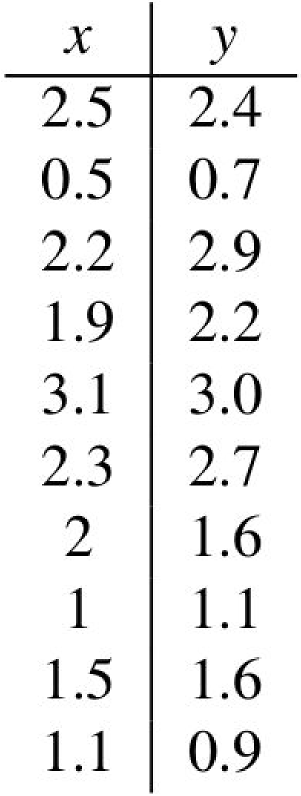


x	y
2.5	2.4
0.5	0.7
2.2	2.9
1.9	2.2
3.1	3.0
2.3	2.7
2	1.6
1	1.1
1.5	1.6
1.1	0.9

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

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

PCA How it works (Again)?



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

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

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

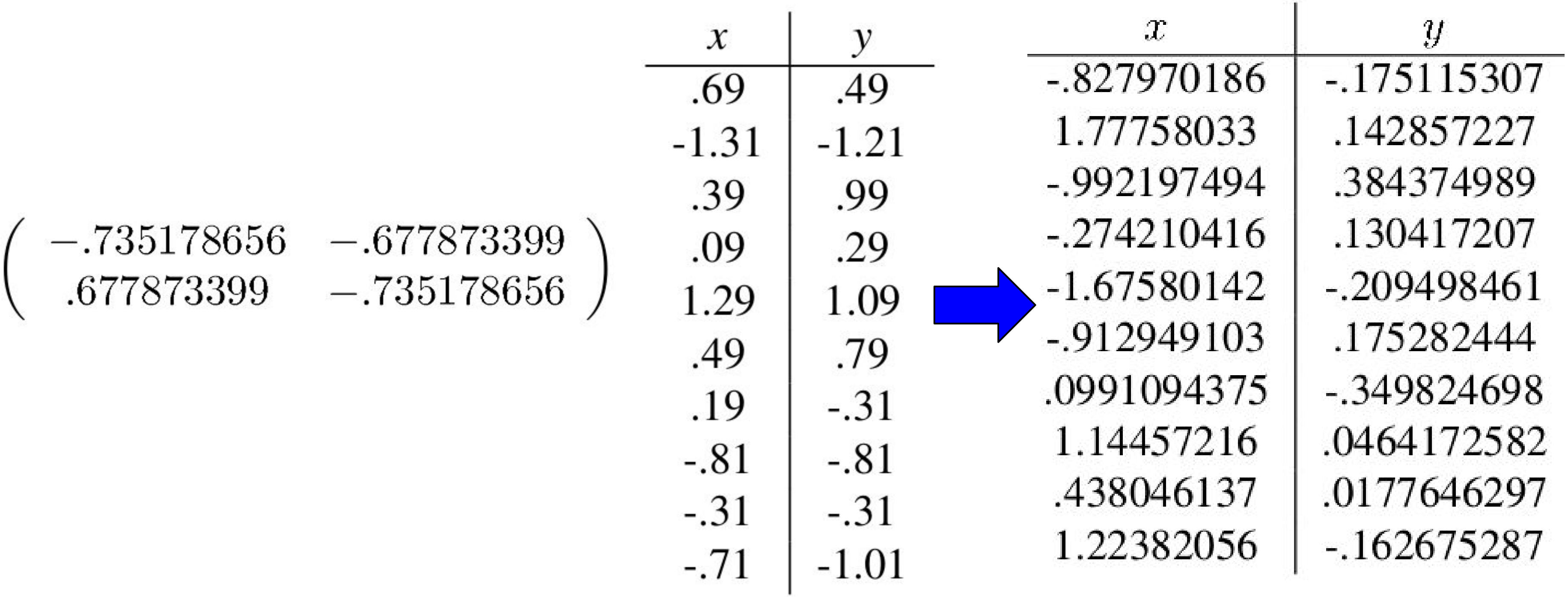
PCA How it works (Again)?

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

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

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

PCA How it works (Again)?



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

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

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

PCA How it works (Again)?

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

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

→

x
-.827970186
1.77758033
-.992197494
-.274210416
-1.67580142
-.912949103
.0991094375
1.14457216
.438046137
1.22382056

Questions