McMaster University

Integrated Biomedical Engineering & Health Sciences Program

IBEHS - 4QZ3

Modelling of Biological Systems

Lecture 6

TAYLOR DEVET MASC.

PHD. CANDIDATE BIOLOGICAL AND BIOMEDICAL ENGINEERING
MCGILL UNIVERSITY

SHRINERS HOSPITAL FOR CHILDREN

Todays Aims…

PCA

ICA

Too much data, no idea where to start?

```
>> load cities
```

9 different indicators of the quality of life in 329 U.S. cities.

For each category, a higher rating is better.

represent a multivariate data table as smaller set of variables (summary indices)

	Climate	Housing	Health	Crime	Transportation	Education	Arts	Recreation	Economics
Abilene, TX	521	6200	237	923	4031	2757	996	1405	7633
Akron, OH	575	8138	1656	886	4883	2438	5564	2632	4350
Albany, GA	468	7339	618	970	2531	2560	237	859	5250
Albany-Troy, NY	476	7908	1431	610	6883	3399	4655	1617	5864
Albuquerque, NM	659	8393	1853	1483	6558	3026	4496	2612	5727

. . .

Principal Component Analysis

PCA is a linear transformation used to transform one set of variables into another set of variables.

PCA is used to provide information on the true dimensionality of a data set

PCA tells you if the data set can be transformed into a fewer number of variables that still contain most of the essential information.

PCA also implements that transformation.

PCA

Used in complex systems such as neuroscience, imaging photometry, meteorology, oceanography

- the number of variables to measure can occasionally be huge, and knowledge of which variables to measure is not apparent

- at times underlying relationships can often be quite simple but are obscured.

PCA

Model free multivariate statistical approach

PCA rotates (transforms) a data set until the all the variables are uncorrelated.

- Uncorrelated variables provide no information on one another.

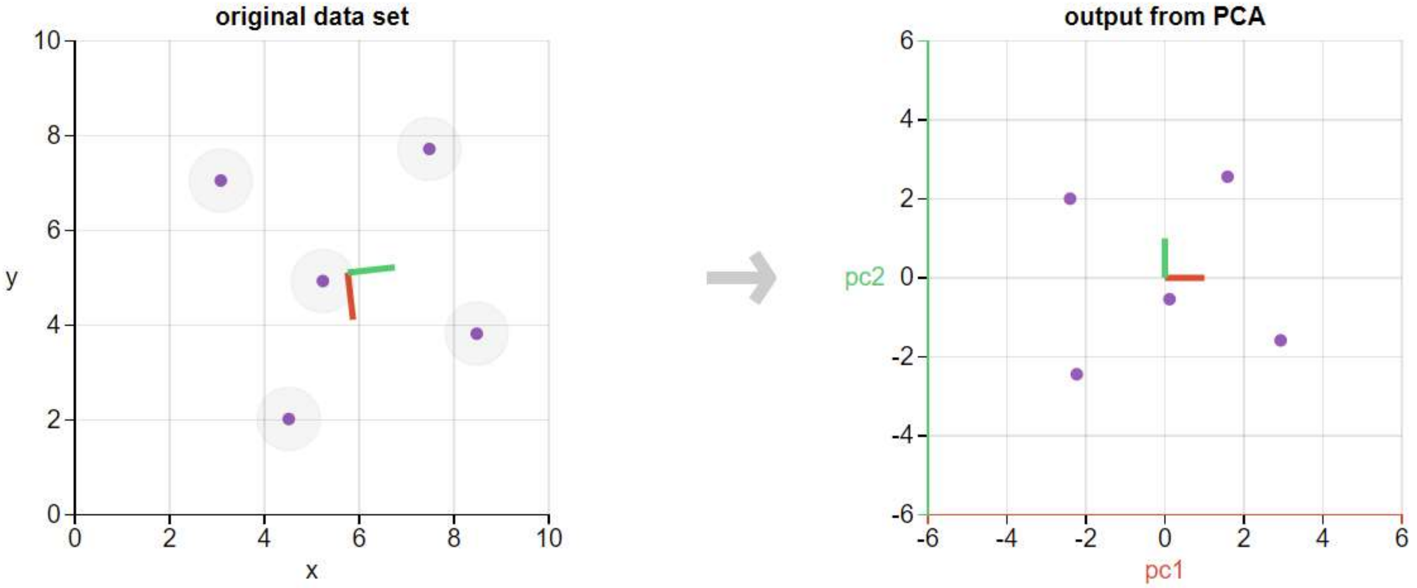
- summarize data variability into as few spatial components (“eigenvalues or eigenimages”) as possible.

-1st eigenvalue represents the largest source of variance, the second the largest source of residual variance orthogonal to first, etc.

- groups of multivariate data often subtly reflect the same driving process/behavior of the system.

- simplify the problem by replacing a group of variables with a single new variable.

Interactive Example



https://setosa.io/ev/principal-component-analysis/

Principal Component Analysis (PCA)

Linear decomposition of very correlated data to a new coordinate system along maximal variance

$$X = TP^T$$

- Each principal component (p_i) is vector in direction of largest variance in data and orthogonal to all other components
- Projection of each variable onto principal component produces loading vector (t_i)

Usually not all components needed due to correlation

- Allows to create statistical model ($\hat{X}$) of data with minimal assumptions

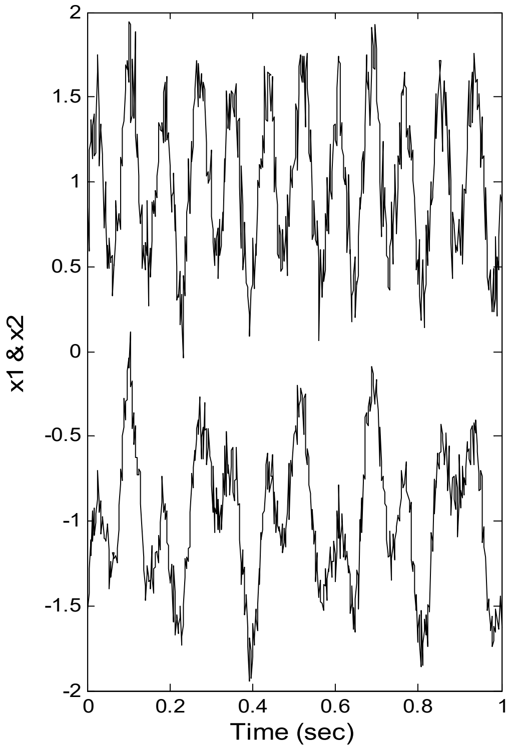
$$\begin{aligned} X &= t_1 p_1^T + t_2 p_2^T + \cdots + t_K p_K^T \\ &= t_1 p_1^T + t_2 p_2^T + \cdots + t_A p_A^T + E \qquad A < K \\ &= \hat{X} + E \end{aligned}$$

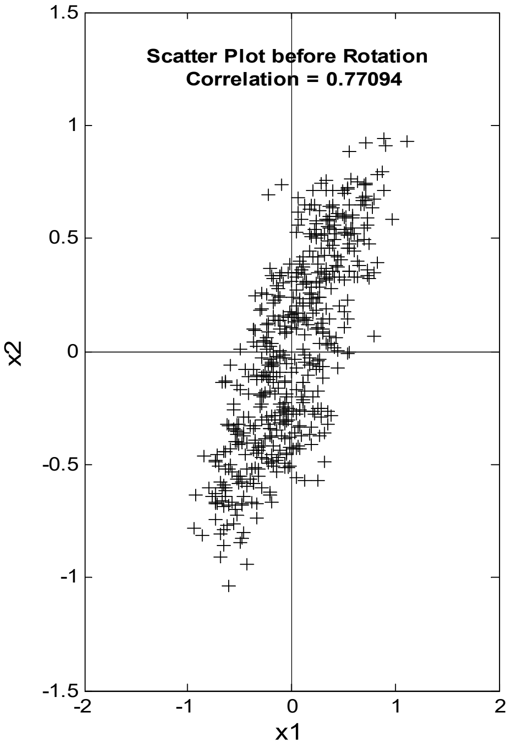
K is number of data sources

PCA Example

E.g. 2-variable data set made from the mixtures of two sine waves. Each mixture contains different amplitudes of the two sinusoids and noise.

- There is a high degree of correlation between x_1 and x_2.
- Knowing the value of x_1 gives you a range of possible x_2 values.

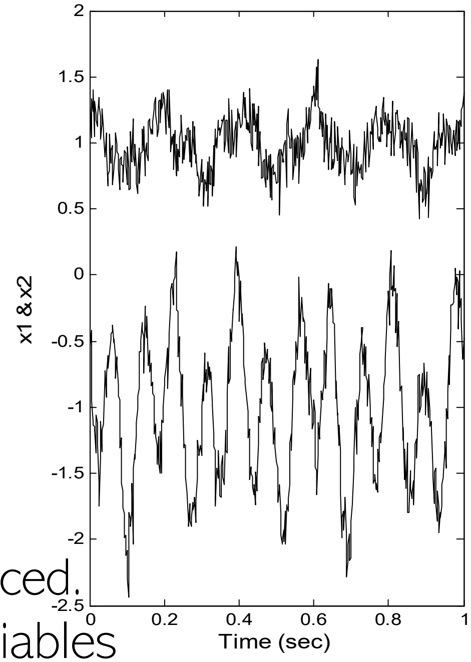


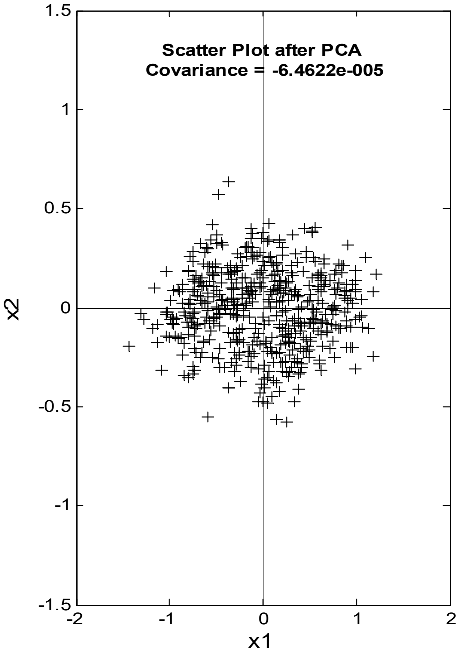

Scatter Plot before Rotation
Correlation = 0.77094

PCA Example

After rotation the two variables are no longer correlated.

- However the new variables are still mixtures of the two sources (just a different mixing).
- The new variables are not more meaningful than the original variables.





- Moreover this data set can not be reduced. In this case, you really do need two variables to represent the two sinusoids.

Uncorrelated Does Not Mean Independent

If two (or more) variables are statistically independent

- they will also be uncorrelated

If two (or more) variables are uncorrelated

- they may not be statistically independent

This is suggested by the plots for two-variable data set plotted as time and scatter plots in the next slide.

The two signals are uncorrelated, but they are highly related and not independent.

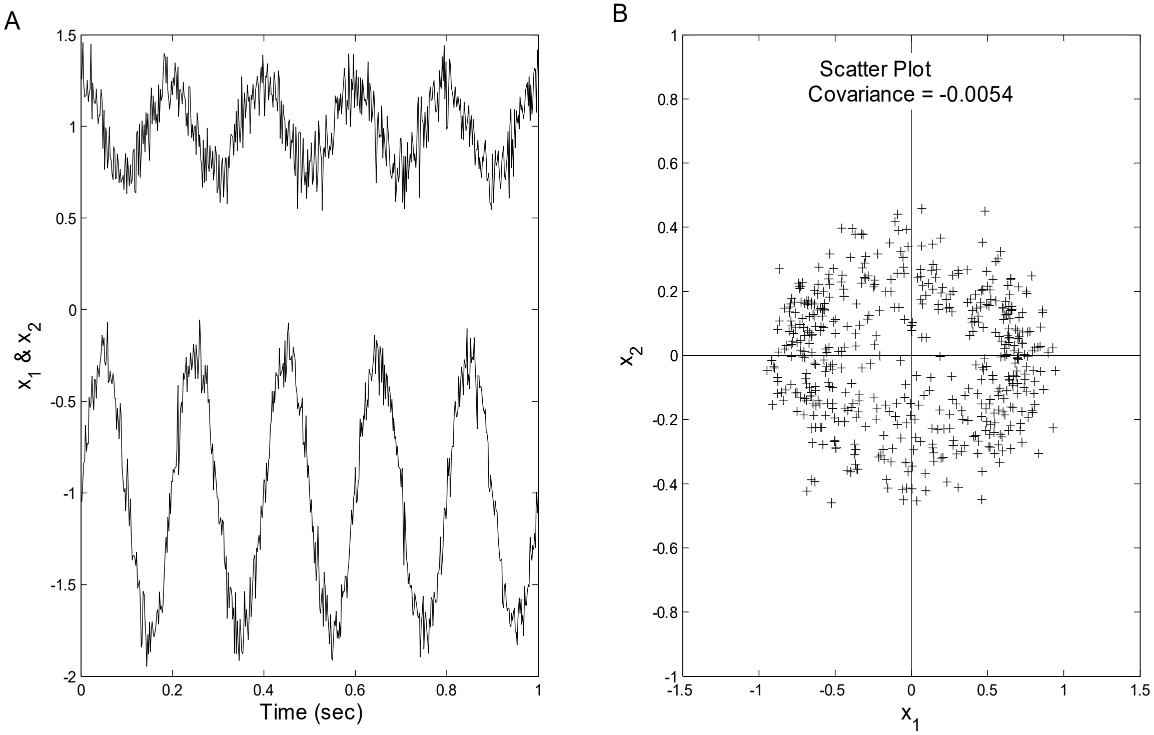
Example

Time (A) and scatter plot (B) of two variables that are uncorrelated, but not independent.

In fact, the two variables are highly dependent.

There is clearly a relationship (nonlinear) between them.

In fact, they were generated by a single equation, that of a circle, with noise added.



Spring Example

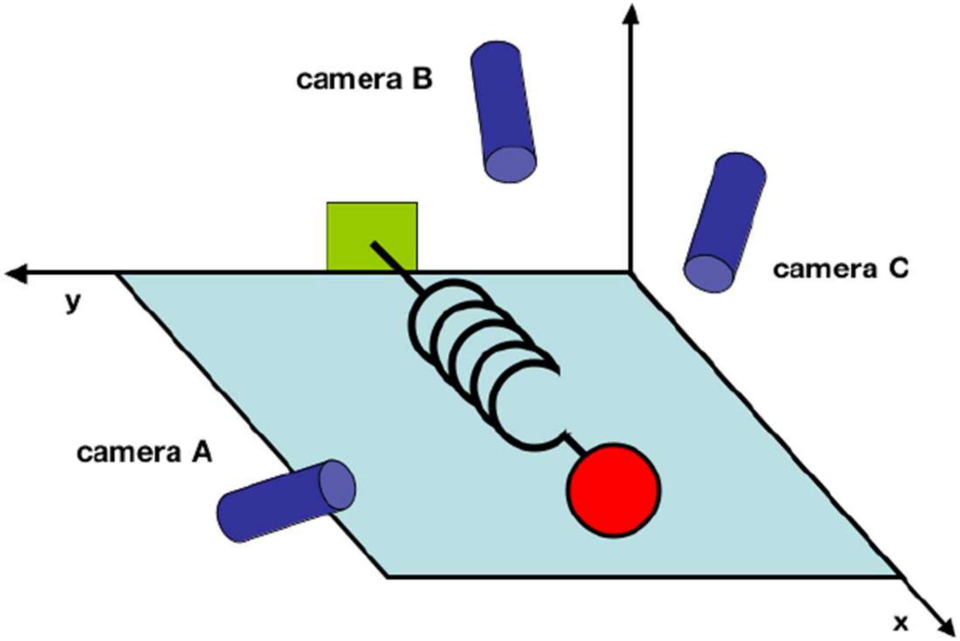
e.g. simple toy problem from physics: motion of an ideal spring.

- system consists of a ball of mass m attached to a massless, friction-less spring.

The ball is released a small distance away from equilibrium (i.e. the spring is stretched).

Because the spring is "ideal," it oscillates indefinitely along the x-axis about its equilibrium at a set frequency.

Spring Example



Each camera is at some arbitrary angle sampling images at 120 Hz (i.e. camera records an image indicating the two dimensional position of the ball, a projection)

Spring Example

- record for several minutes. The big question remains: how do we get from this data set to a simple equation of x?

- also what about noise?

- The goal of PCA is to compute the most meaningful basis to re-express a noisy data set.

- hopefully this new basis will filter out the noise and reveal hidden structure.

Spring Example

- For each time point a camera records ball position

- at each time point we have 6 variables measured

- so with 120Hz sampling we have 72000 values in 10 minutes

$$\vec{X} = \begin{bmatrix} x_A \\ y_A \\ x_B \\ y_B \\ x_C \\ y_C \end{bmatrix}$$

- 3 cameras
- 2D info (x and y)

- Each sample *X* is an m-dimensional vector

- *m* is the number of measurement types.

Spring Example

From linear algebra we know that all measurement vectors form a linear combination of this set of unit length basis vectors. What is this orthonormal basis?

Pretend we gathered our toy example data above, but only looked at camera A. What is an orthonormal basis for (xA, yA)?

In the two dimensional case, {(1, 0), (0, 1)} can be recast as individual row vectors to make a 2x2 identity matrix, I

$$\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} = I$$

Spring Example

generalize this to the m-dimensional case by constructing an m × m identity matrixx

each row is an orthornormal basis vector bi with m components

All of the camera data has been recorded in this basis and thus it can be trivially expressed as a linear combination of {bi}.

$$\mathbf{B} = \begin{bmatrix} \mathbf{b_1} \\ \mathbf{b_2} \\ \vdots \\ \mathbf{b_m} \end{bmatrix} = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \end{bmatrix} = \mathbf{I}$$

PCA: Linearity

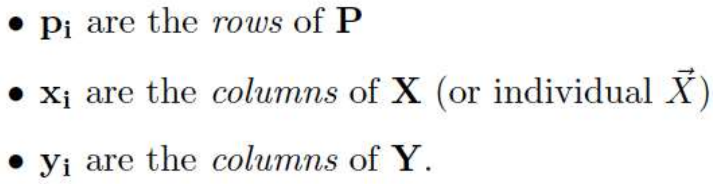
PCA: Is there another basis, which is a linear combination of the original basis, that best re-expresses the data?

One strict assumption: linearity. This vastly simplifies the problem by

(1) restricting the set of potential bases

(2) formalizing the implicit assumption of continuity in a data set

$$PX = Y$$



- $\mathbf{p_i}$ are the *rows* of $\mathbf{P}$
- $\mathbf{x_i}$ are the *columns* of $\mathbf{X}$ (or individual $\vec{X}$)
- $\mathbf{y_i}$ are the *columns* of $\mathbf{Y}$.

In the toy example X is an m × n matrix where m = 6 and n = 72000.

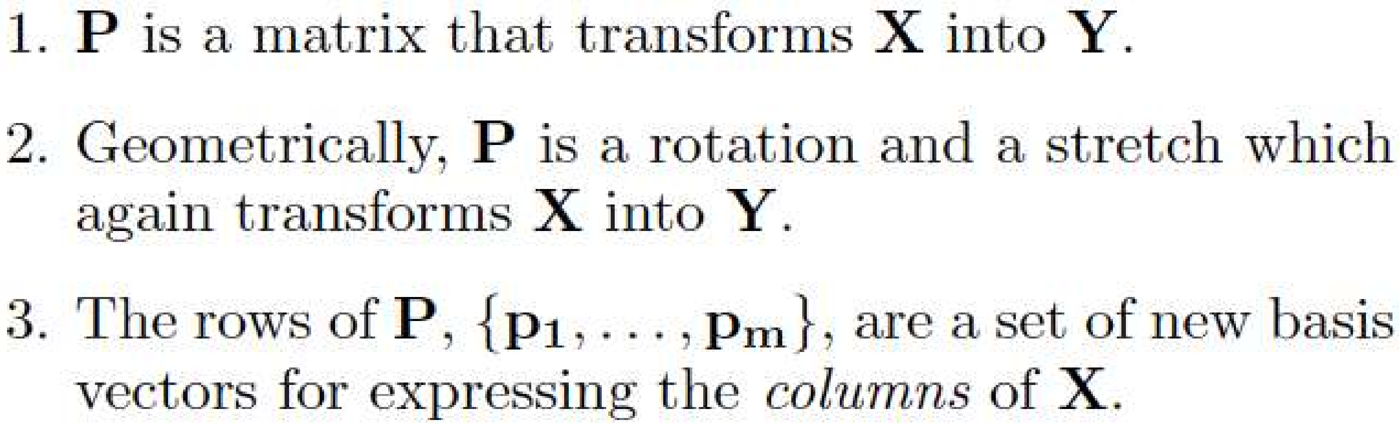
Let Y be another m × n matrix related by a linear transformation P.

X is the original recorded data set and Y is a re-representation of that data set.

PCA: Linearity

This equation represents a change of basis and can be interpreted a number of ways:

$$PX = Y$$



1. $\mathbf{P}$ is a matrix that transforms $\mathbf{X}$ into $\mathbf{Y}$.
2. Geometrically, $\mathbf{P}$ is a rotation and a stretch which again transforms $\mathbf{X}$ into $\mathbf{Y}$.
3. The rows of $\mathbf{P}$, $\{\mathbf{p_1}, \ldots, \mathbf{p_m}\}$, are a set of new basis vectors for expressing the *columns* of $\mathbf{X}$.

PCA: Linearity

By assuming linearity the problem reduces to finding the appropriate change of basis

But, need to deal with:

1) noise

2) rotation

3) redundancy

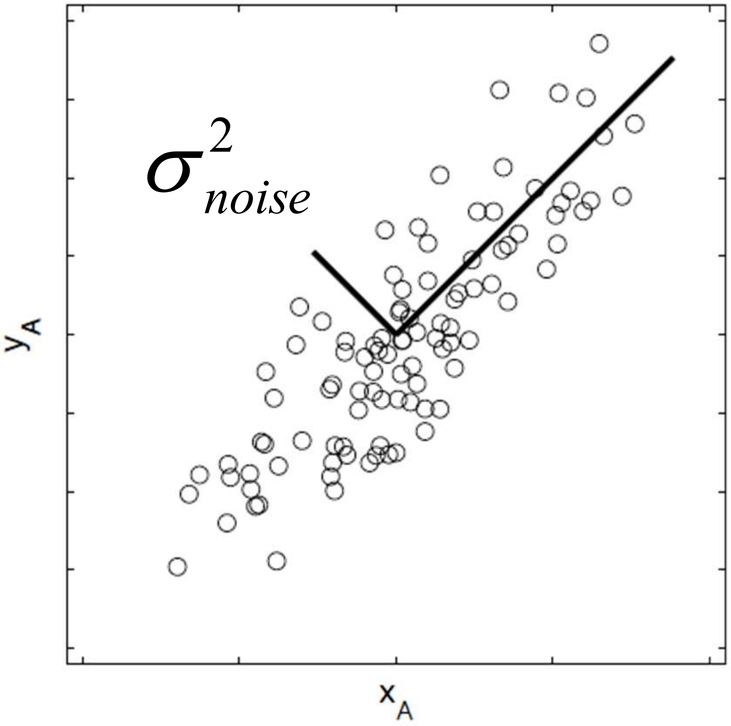
Noise and Rotation

Measurement noise must be low or else no information about the system can be extracted! This is irrespective of the analysis technique

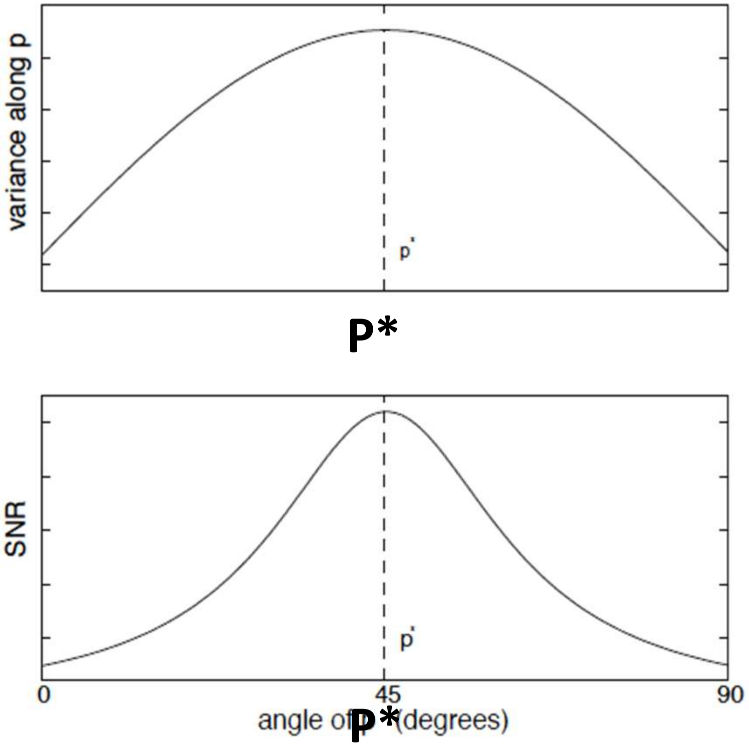
$$SNR = \frac{\sigma^2_{signal}}{\sigma^2_{noise}}$$

- high SNR ($\gg$ 1) indicates high precision data, while low SNR indicates noise contaminated data

Revisiting the Spring



σ^2_{signal}



P*

P*

2D PCA

- Quantitatively assume that directions with largest variances in our measurement vector space contain the dynamics of interest (i.e. direction along the long axis of the cloud).

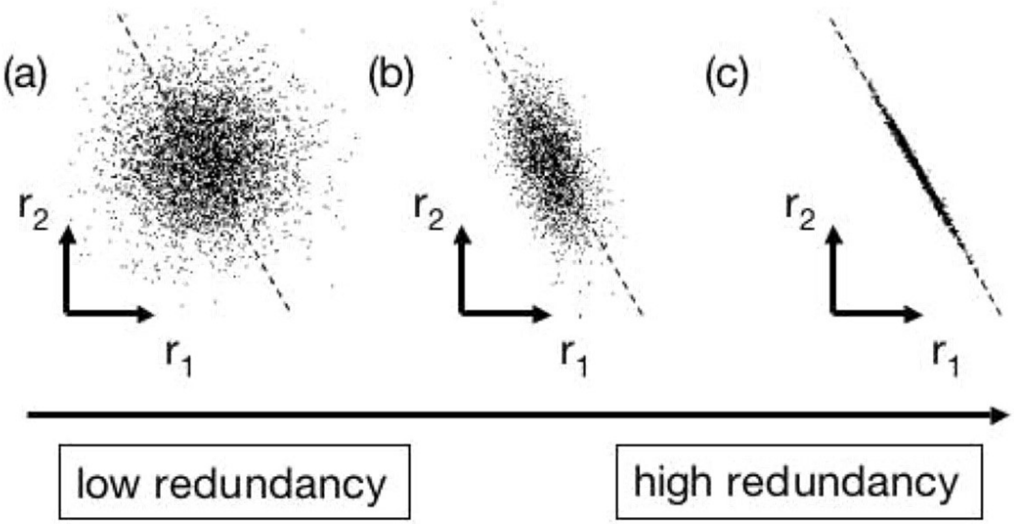
- Maximizing the variance (and by assumption the SNR) corresponds to finding the appropriate rotation of the naive basis (i.e. direction of P*)

- In this simple 2-dimensional case P* falls along the direction of the best-fit line for the data cloud. Thus, rotating the naive basis to lie parallel to P* would reveal the direction of motion of the spring.

What about multiple dimensions??

Redundancy

- an additional potential confounding factor. i.e. multiple sensors recording the same information

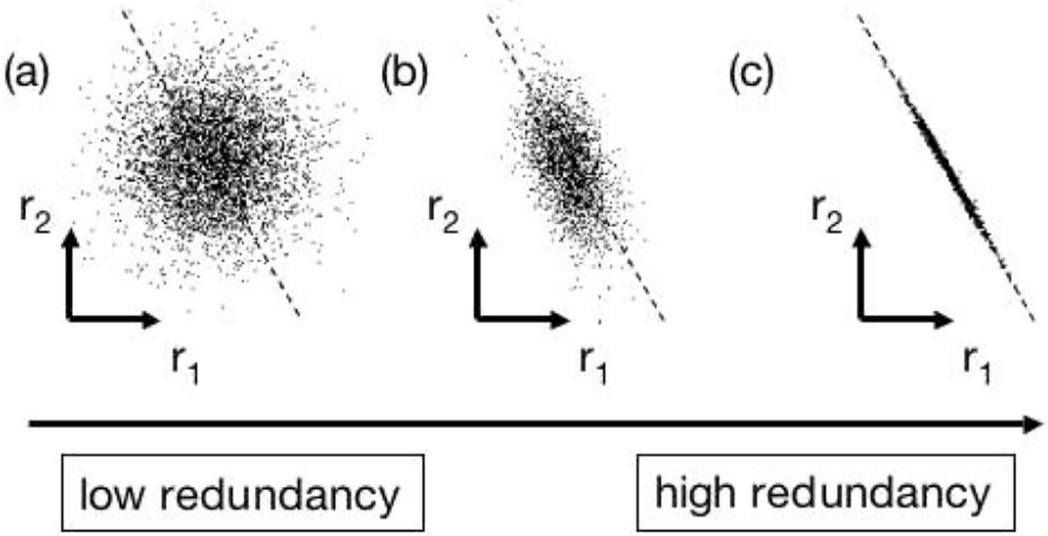


In (c) it would have been more meaningful to just have recorded a single variable, not both. Because one can calculate r_1 from r_2 (or vice versa) using the best fit line.
Recording only 1 expresses data more concisely and reduces number of sensor recordings (dimensional reduction).

Redundancy

simple to identify redundancy in 2 variable cases: find slope of the best-fit line and judge the quality of the fit.

BUT: How can this be generalized to higher dimensions?



Measuring Redundancy

Consider two sets of measurements with zero means, where in both cases the subscript denotes the sample number:

Variances:

$$A = \{a_1, a_2, \ldots, a_n\} \qquad \sigma_A^2 = \langle a_i a_i \rangle_i$$

$$B = \{b_1, b_2, \ldots, b_n\} \qquad \sigma_B^2 = \langle b_i b_i \rangle_i$$

$$covariance\ of\ A\ and\ B \equiv \sigma_{AB}^2 = \langle a_i b_i \rangle_i$$

- covAB measures the degree of linearity between 2 variables.

- If covAB is large there is high redundancy.

Measuring Redundancy

$$\sigma^2_{AB} \geq 0$$

$\sigma^2{}_{AB} = 0$ iff A and B are totally uncorrelated

$\sigma^2{}_{AB} = \sigma^2{}_{A} = \sigma^2{}_{B}$ *(if $A = B$)*

Variance

$$\sigma^2_{AB} \geq 0$$

$\boldsymbol{\sigma}^2_{AB}$= 0 iff A and B are totally uncorrelated

$\boldsymbol{\sigma}^2_{AB} = \boldsymbol{\sigma}^2_{A} = \boldsymbol{\sigma}^2_{B}$ *(if A = B)*

- can equivalently convert A and B into corresponding row vectors:

$$\mathbf{a} = [a_1 \; a_2 \; \ldots \; a_n]$$

$$\mathbf{b} = [b_1 \; b_2 \; \ldots \; b_n]$$

now express the covariance as a dot product:

$$\sigma^2_{\mathbf{ab}} \equiv \frac{1}{n-1}\mathbf{ab}^T$$

(note: $(n-1)^{-1}$ is used for normalization)

Combining vectors

Now, we are not stuck with only 2 vectors in real life. So we can arbitrarily call them:

$$\mathbf{x_1} \equiv \mathbf{a}$$
$$\mathbf{x_2} \equiv \mathbf{b}$$

Where additional indexed row vectors:

$$\mathbf{x_3}, \ldots, \mathbf{x_m}$$

Define new m x n matrix, X:

- rows of X correspond to all measurements of a particular type (xi).

Each column of X corresponds to a set of measurements from one particular trial

$$\mathbf{X} = \begin{bmatrix} \mathbf{x_1} \\ \vdots \\ \mathbf{x_m} \end{bmatrix}$$

Covariance Matrix

- matrix XX^T computes ij^{th} element of C_X

- ij^{th} element of CX is dot product between the vector of the ith measurement type with the vector of the jth measurement type

properties of CX:

1) CX is a square symmetric m × m matrix

2) Diagonal terms of CX are the variance of particular

measurement types (large=interesting; small=noise)

3) Off-diagonal terms of CX are the covariance between measurement types (large=high redundancy)

$$\mathbf{C_X} \equiv \frac{1}{n-1}\mathbf{X}\mathbf{X}^T$$

Goals of PCA

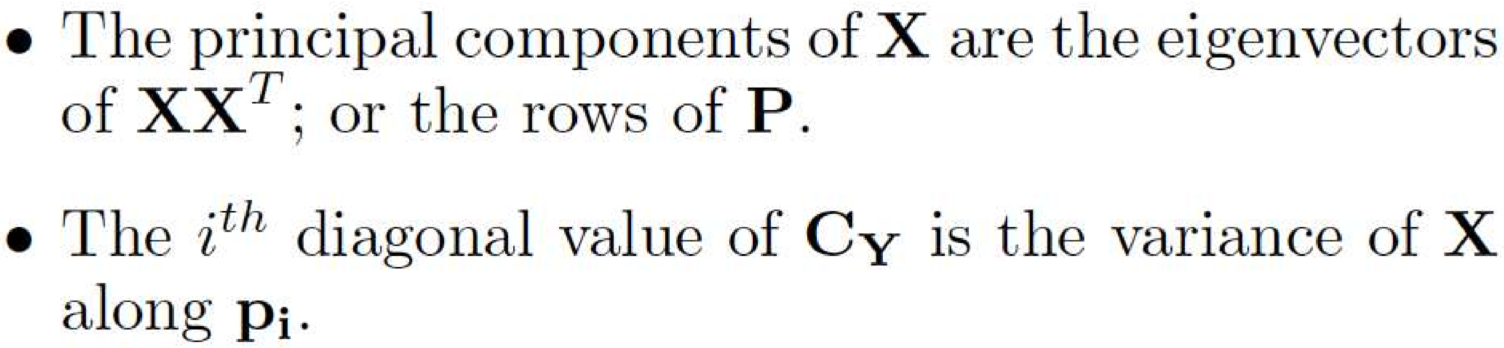
$$PX = Y$$

(1) minimize redundancy, measured by covariance

(2) maximize signal, measured by variance.

By definition covariances must be non-negative, thus the minimal covariance is zero.

Therefore, the "optimized" covariance matrix of CY would have all off-diagonal terms equal to zero (i.e. CY must be diagonal). So, need to diagonalize CY.



- The principal components of $\mathbf{X}$ are the eigenvectors of $\mathbf{XX}^T$; or the rows of $\mathbf{P}$.
- The i^{th} diagonal value of $\mathbf{C_Y}$ is the variance of $\mathbf{X}$ along $\mathbf{p_i}$.

Performing PCA

1. Organize a data set as an m x n matrix, where m is the number of measurement types and n is the number of trials

2. Subtract the mean for each measurement type or row xi

3. Calculate the eigenvectors of the covariance matrix.

A Multivariate Example

In multivariate analysis, multiple variables are analyzed together and often represented as a single vector variable that includes the different variables :

$$x = \left[x_m(1), x_m(2), \ldots x_m(N)\right]^T \quad \text{for } 1 \le m \le M$$

The 'T' stands for transposed.

The variable X is composed of M variables (rows) each containing N observations (columns). In signal analysis, the observations are time samples, (t = 1,…N)

$$X = \begin{bmatrix} x_1[1] & x_1[2] & x_1[3] & \ldots & x_1[N] \\ x_2[1] & x_2[2] & x_2[3] & & x_2[N] \\ x_3[1] & x_3[2] & \ddots & & x_3[N] \\ \vdots & \vdots & & \ddots & \vdots \\ x_M[1] & x_M[2] & x_M[3] & \cdots & x_M[N] \end{bmatrix}$$

Multivariate Analysis (cont)

In signal processing the observations are time samples while in image processing they are pixels.

Multivariate data, as represented by X above, can also be considered to reside in M-dimensional space, where each spatial dimension contains one set of observations.

Multivariate analysis takes into account relationships between and within the multiple variables.

(For example, the covariance matrix includes information about the relationship between variables as well as about the variables themselves.)

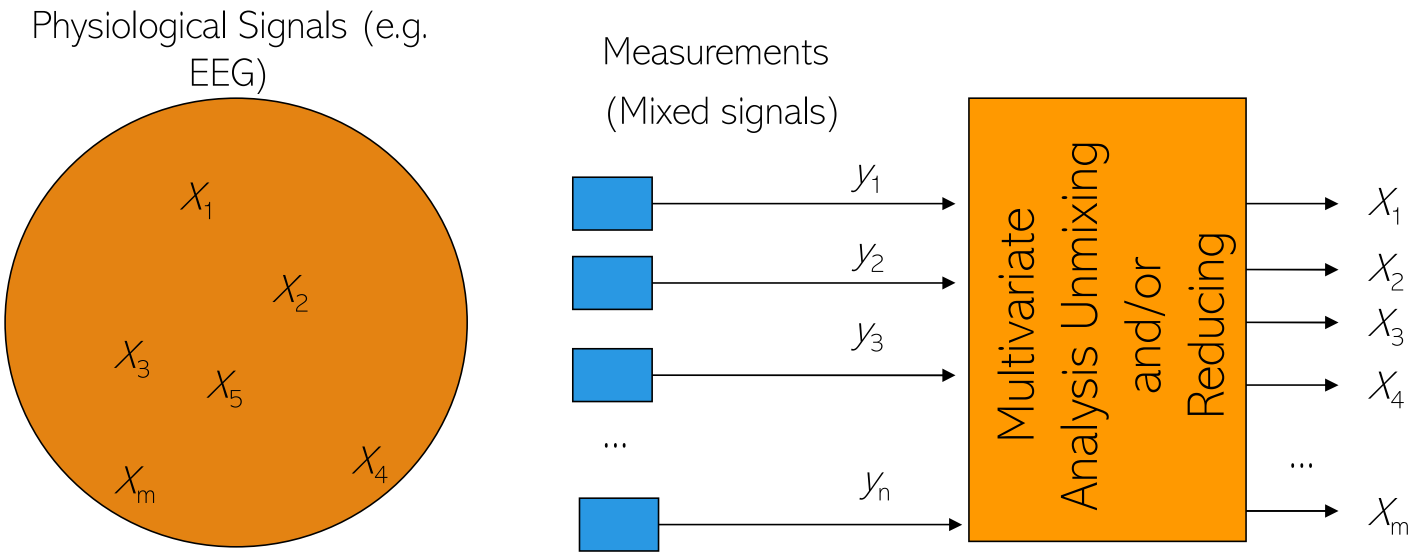
Multivariate Analysis (continued)

A major concern of multivariate analysis is to find transformations of the data that make the data set smaller or easier to understand.

1) Can the relevant information contained in a multi-dimensional variable be expressed using fewer variables? Are variables redundant?

2) Can the data be transformed to be more meaningful?
If so, the more meaningful variables are described as hidden, or latent, in the original data (they are the latent variables).

Multivariate Applications in Behavioral Physiology



Physiological Signals (e.g. EEG)

Measurements

(Mixed signals)

Multivariate Analysis Unmixing and/or Reducing

The *y*' s (measured) are mixtures of the *x*' s.

You have the y' s, but you want the *x*' s.

Usually $n > m$.

Multivariate Data Transformations

In transformations that reduce the dimensionality of a multi-variable data set, the idea is to transform one set of variables into a new set where some of the new variables have values that are much smaller that the rest.

The data transformation used to produce the new set of variables is often a linear function.

A linear transformation can be represent mathematically as:

$$\begin{bmatrix} y_1(t) \\ y_2(t) \\ \vdots \\ y_M(t) \end{bmatrix} = W \begin{bmatrix} x_1(t) \\ x_2(t) \\ \vdots \\ x_M(t) \end{bmatrix}$$

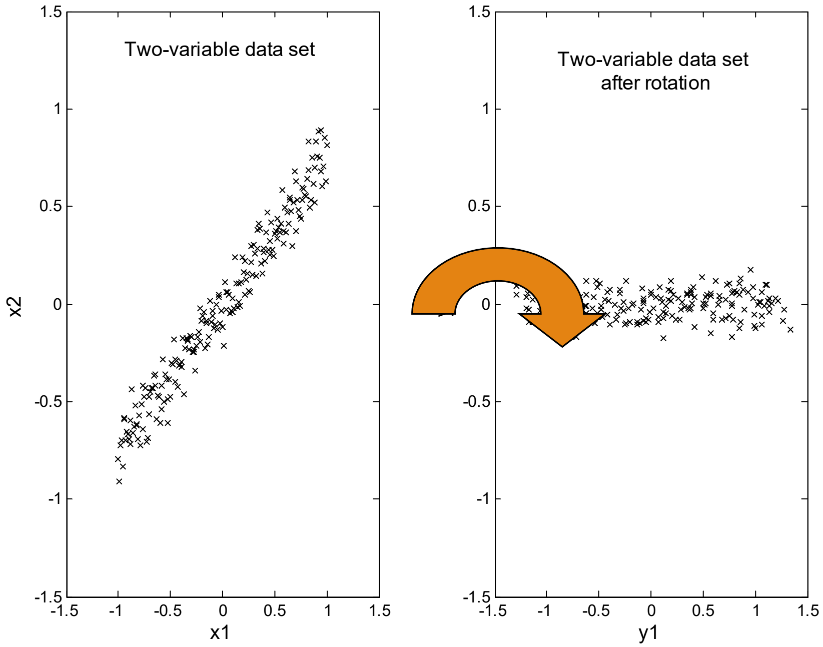
or: $y_i(t) = \sum_{j=1}^{M} w_{ij} x_j(t) \qquad i = 1, \ldots N$

where W (= w_{ij}) is a matrix that defines the transformation.

Linear Transformations

A linear transformation can be interpreted as a rotation of the data set.

- The rotated data set still contains two variables, but the variance of one of the variables is quite small compared to the other.
- Perhaps this new variable (y_2) just represents noise and could be eliminated



Determining PCA using Singular Value Decomposition

The easiest way to implement PCA is using singular value decomposition (SVD).

Singular value decomposition factors a data matrix, X, into the product of three matrices:

1) A diagonal matrix, S, containing the square root of the eigenvalues
2) A principal components matrix, U,
3) and its orthonormal version, V

$$X = USV^T$$

The symbols and the details of SVD vary depending on the source; here the definitions are based on MATLAB

SVD (continued)

$$\mathbf{X} = \mathbf{USV}^{\mathbf{T}}$$

where X is the m × n data matrix. This matrix is decomposed into:

U = m × m orthonormal matrix;

S = m × n diagonal matrix;

V = n × n matrix containing the principle components

**these are based on MATLABS SVD routines, can be different depending on the source

SVD (continued)

U provides a transformation matrix that will produce a rotated data set which has zero covariance.

S is the covariance matrix of the new (rotated) data set.
It is a diagonal matrix where the diagonals are the variances of the new data set

Squaring the diagonals produces the eigenvalues denoted: λ1, λ2, ...λn which are ordered by magnitude: λ1 > λ2 > ... λn .

V has columns that are the characteristic vectors, or eigenvectors u1, u2,..., un.

When scaled by their respective variances (the square root of the eigenvalues) they become the principle components, the new data set.

Data Reduction: The Scree Plot

The eigenvalues are related to the variances of the principle components.

They are a measure of the associated importance of each principal component

The eigenvalues are in order of magnitude:

1) The first principal component accounts for the maximum variance possible in a single component.

2) The second component accounts for the maximum of the remaining variance for a single component .....

:

N) The last principal component accounts for the smallest amount of variance.

Data Rotation

Many multivariate operations involve data rotation. From basic trigonometry, it is easy to show that, in two dimensions, rotation of a data point (x1, x2) can be achieved by multiplying the data points by the sines and cosines of the rotation angle:

$$y_1 = x_1 \cos(\theta) + x_2 \sin(\theta)$$
$$y_2 = -x_1 \sin(\theta) + x_2 \cos(\theta)$$

where θ is the angle through which the data set is rotated in radians. Using matrix notation, this operation can be done by multiplying the data matrix by a 'rotation' matrix:

Principle Component Scaling

It is common to normalize the principal components by the variances so that different components can be compared.

A number of other normalizing schemes exist.

Here we multiply the eigenvector by the square root of its eigenvalue (i.e., the variance)

This gives rise to principal components that have the same value as a manually rotated data array.

Data Reduction using PCA

The eigenvalues describe how much of the variance is accounted for by the associated principal component and if a component is really necessary.

Eigenvalues are ordered by size; that is:

$$\lambda_1 > \lambda_2 > \lambda_3 \dots > \lambda_M.$$

If an eigenvalue is zero or 'close to' zero, then its associated principal component contributes little to the data and can be eliminated. This component accounts for only a small amount of the variance in the data.

This tells us the effective dimension of the data set.

How do you decide if an eigenvalue is small enough so that its associated component can be removed from the data set?

Data Reduction (cont)

There are two popular methods for determining eigenvalue “thresholds”.

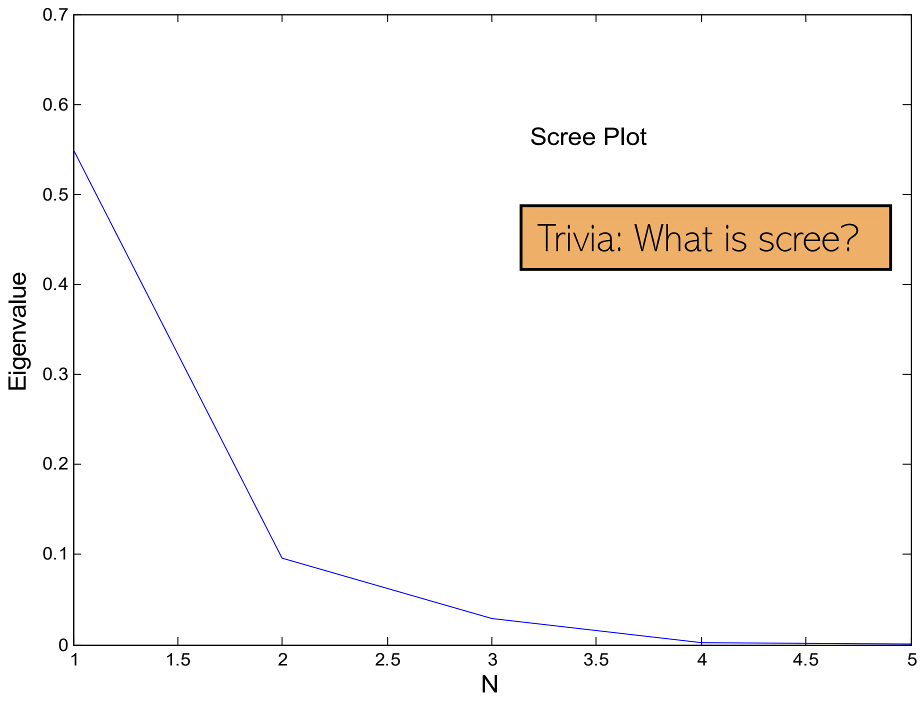
1) Take the sum of all eigenvectors (which must account for all the variance), then delete those eigenvalues that fall below some percentage of that sum (e.g. 95% or whatever you want).

2) Scree Plot: plot of eigenvalues in order and look for breakpoints in the slope of this curve. Eigenvalues representing noise should not change much in value and will plot as a flatter slope when plotted sequentially.

The Scree Plot

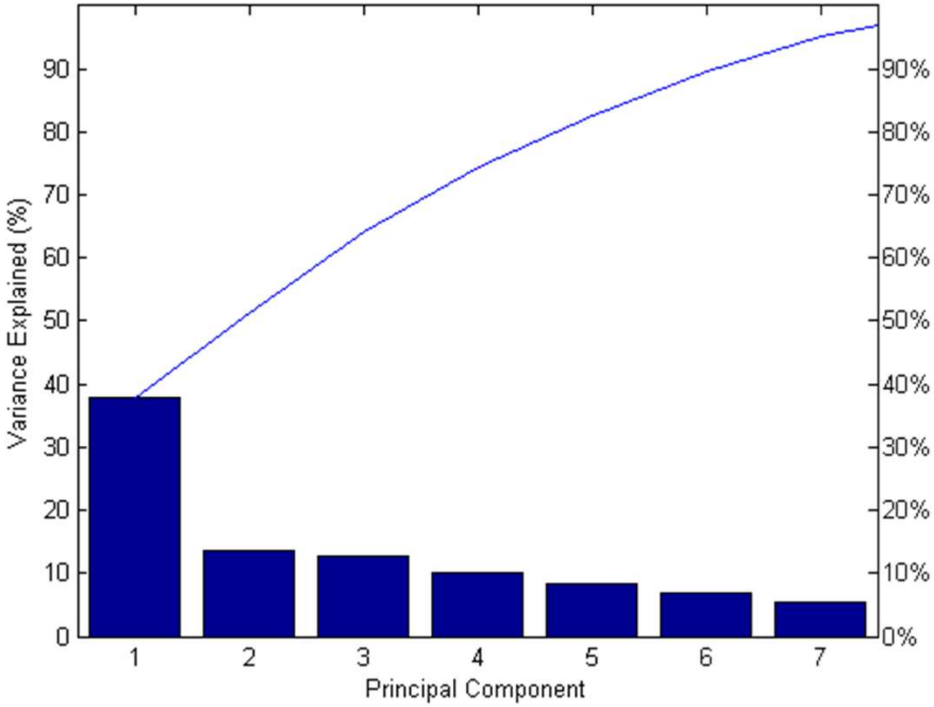
Eigenvalues are in order of large to small.

The actual dimension of the data set is taken where the Scree plot becomes more-or-less flat.



Trivia: What is scree?

Scre Plot pt 2

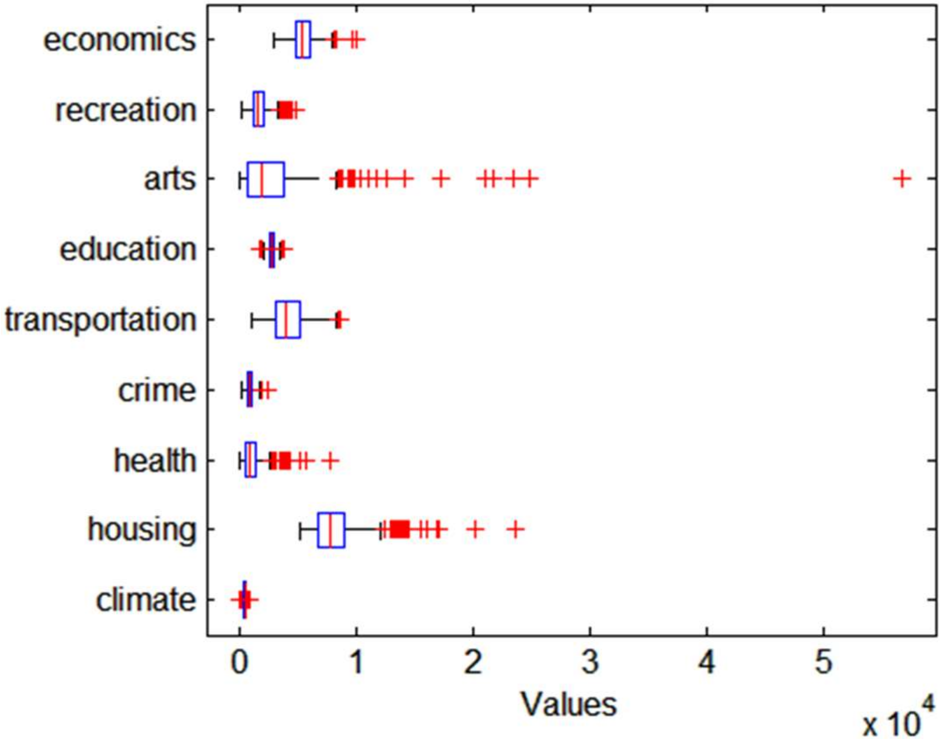


Example: Quality of Life in U.S. Cities

Matlab Sample Data: Quality of Life in U.S. Cities

```
Load cities
```

example shows how to perform a weighted principal components analysis and interpret the results. Here there are 9 components



U.S. Cities Example

- Checking pairwise correlation between the variables shows relatively strong correlation among some variables (up to 0.85).

- PCA constructs independent new variables which are linear combinations of the original variables.

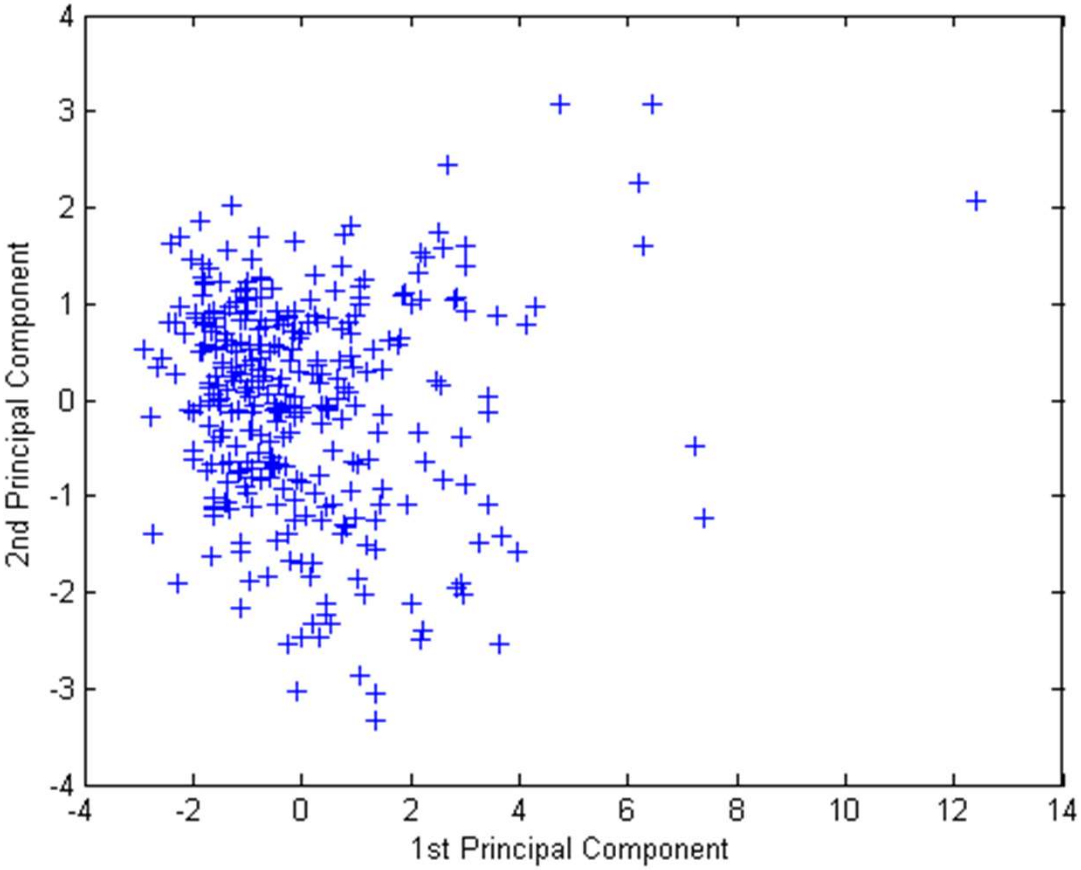
NOTE:

- when all variables have the same units, it is appropriate to compute principal components from ‘raw data’.

- When the variables are in different units or the difference in the variance of different columns is substantial (like this example), scaling of the data or use of weights is often preferable.

This plot shows the centered and scaled ratings data projected onto the first two principal components. PCA computes the scores to have mean zero.

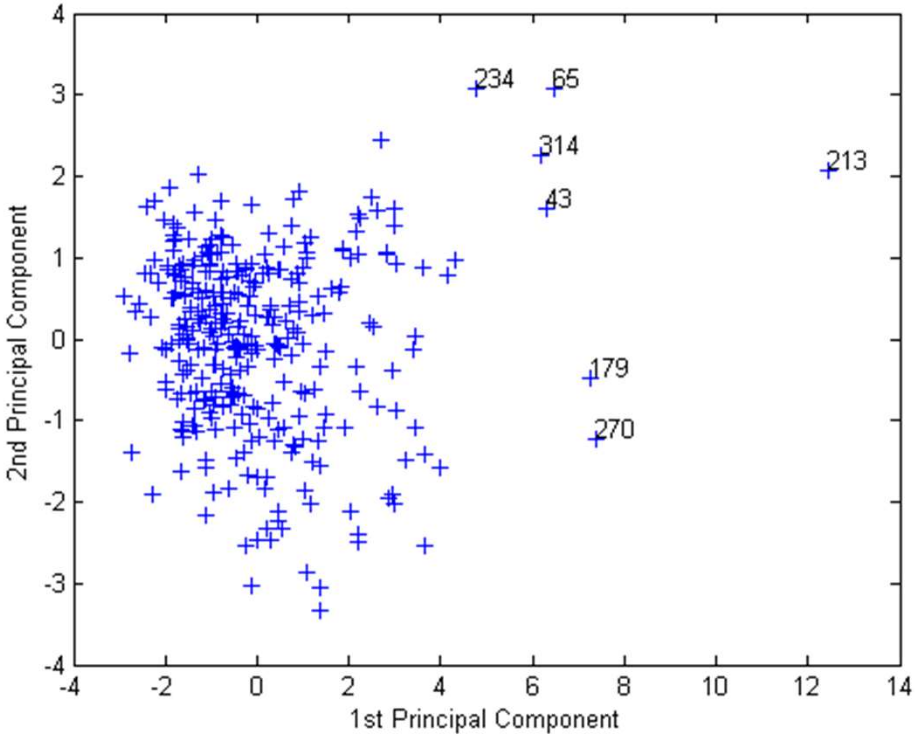
→ each + is a particular city



Matlab Multivariate Statistics Help

Note outliers. Graphically identify using **`gname`**

Boston, Chicago, Los Angeles, Long Beach, New York, Philadelphia, San Francisco, Washington DC. These labeled cities appear more extreme than the remainder of the data



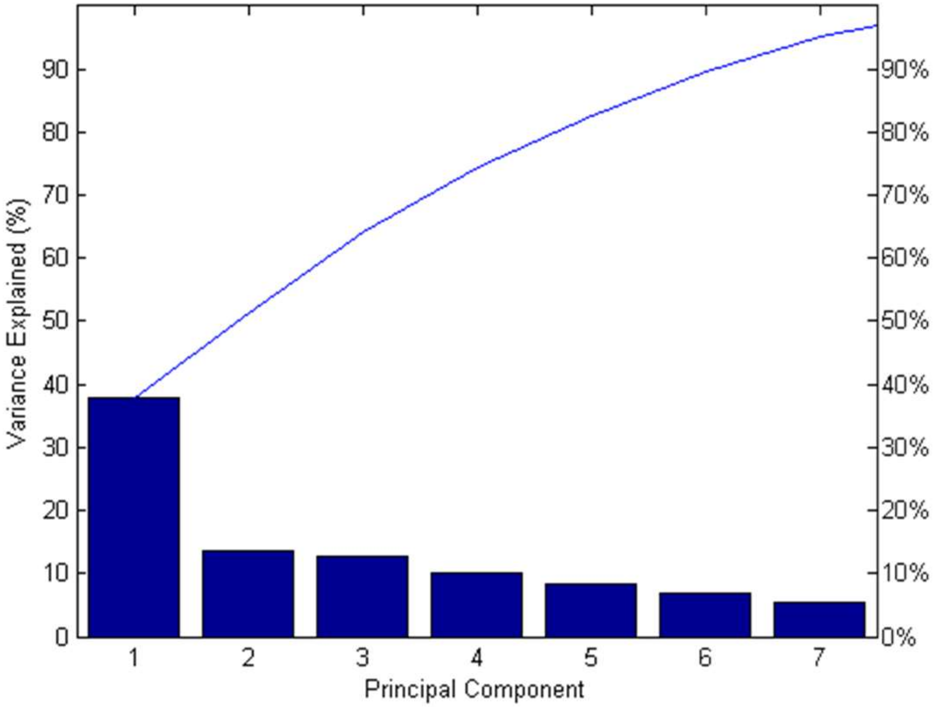
Matlab Multivariate Statistics Help

Scree Plot

What are the sources of variance (in order of importance)?

- 95% of the variance is explained by the first 7 components

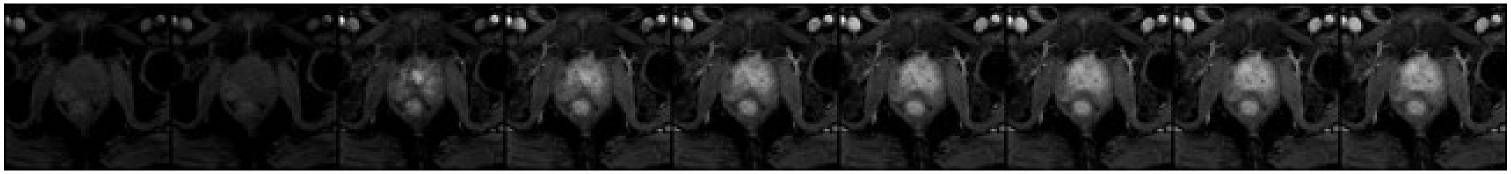
Can have 9 principle components total (i.e. # of measured 9 classes)



PCA of 4D Image Data

By “folding” spatial dimensions into matrix, X, it is possible to analyze “temporal signatures” of 4D MRI data

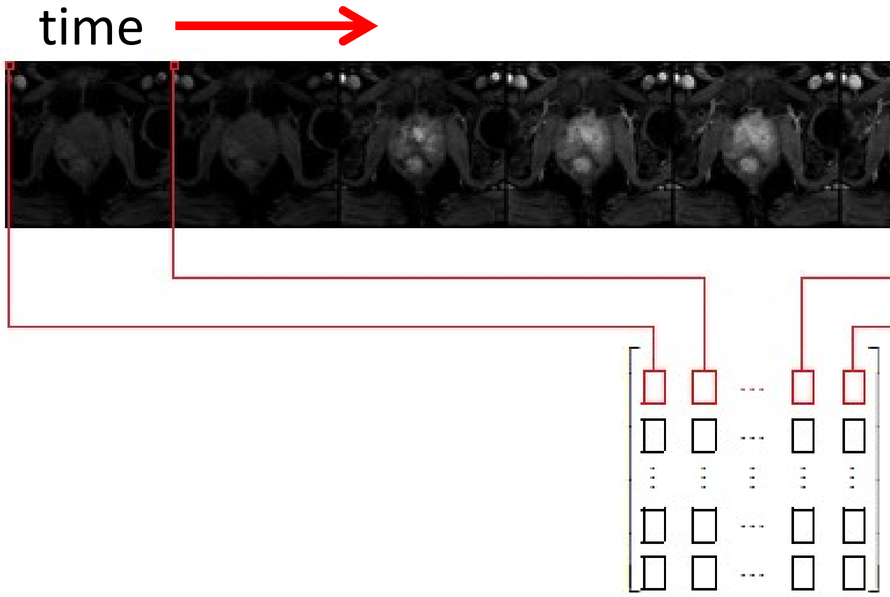
time →



$$\begin{bmatrix} \square & \square & \cdots & \square & \square \\ \square & \square & \cdots & \square & \square \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \square & \square & \cdots & \square & \square \\ \square & \square & \cdots & \square & \square \end{bmatrix}$$

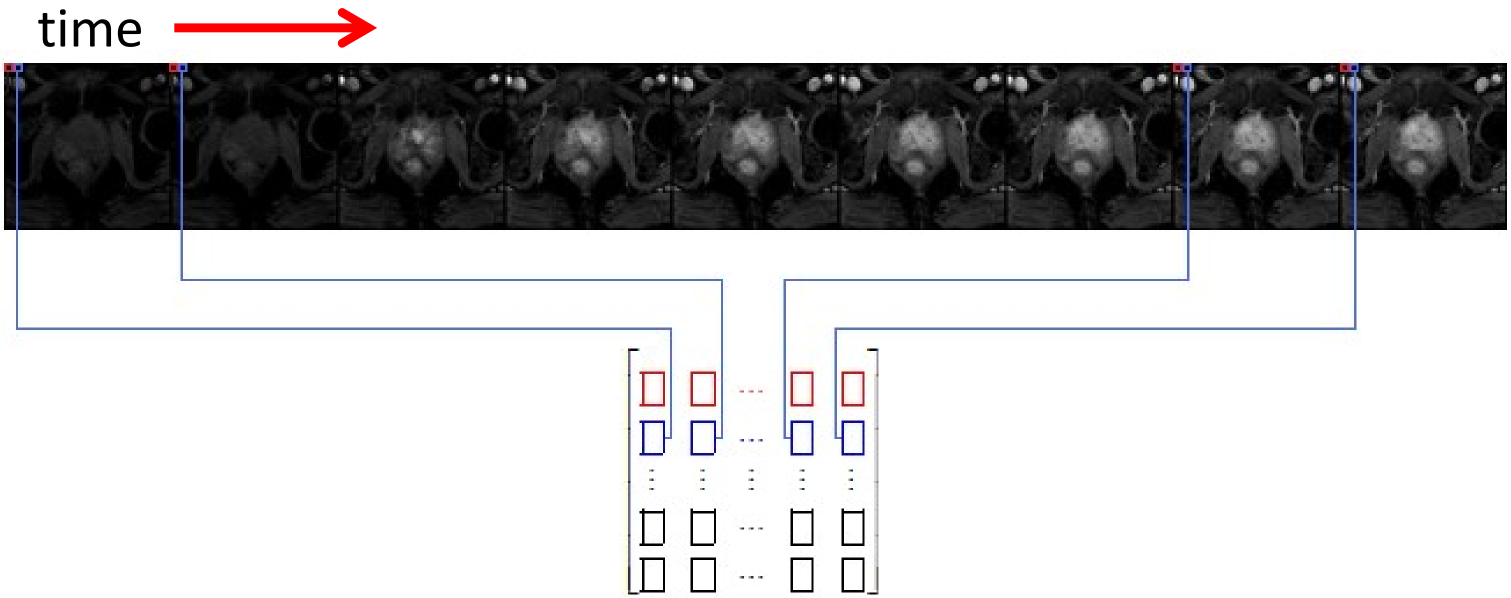
PCA of 4D Image Data

By “folding” spatial dimensions into matrix, X, it is possible to analyze “temporal signatures” of 4D MRI data



PCA of 4D Image Data

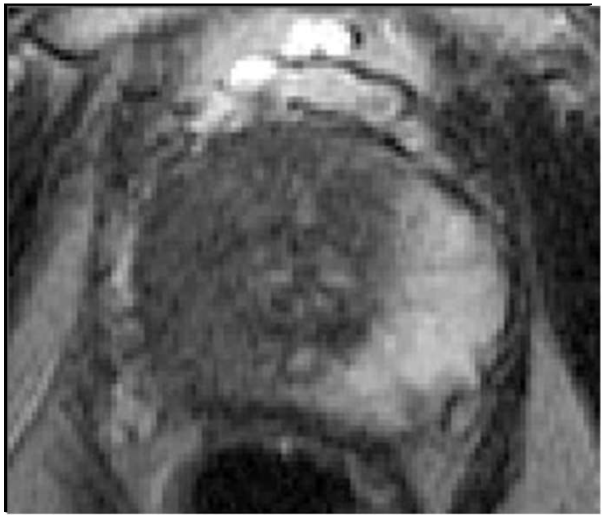
By “folding” spatial dimensions into matrix, X, it is possible to analyze “temporal signatures” of 4D MRI data

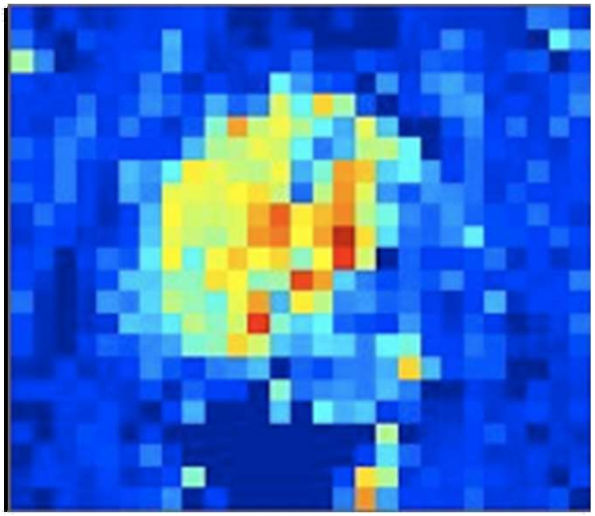


PCA of 4D Image Data

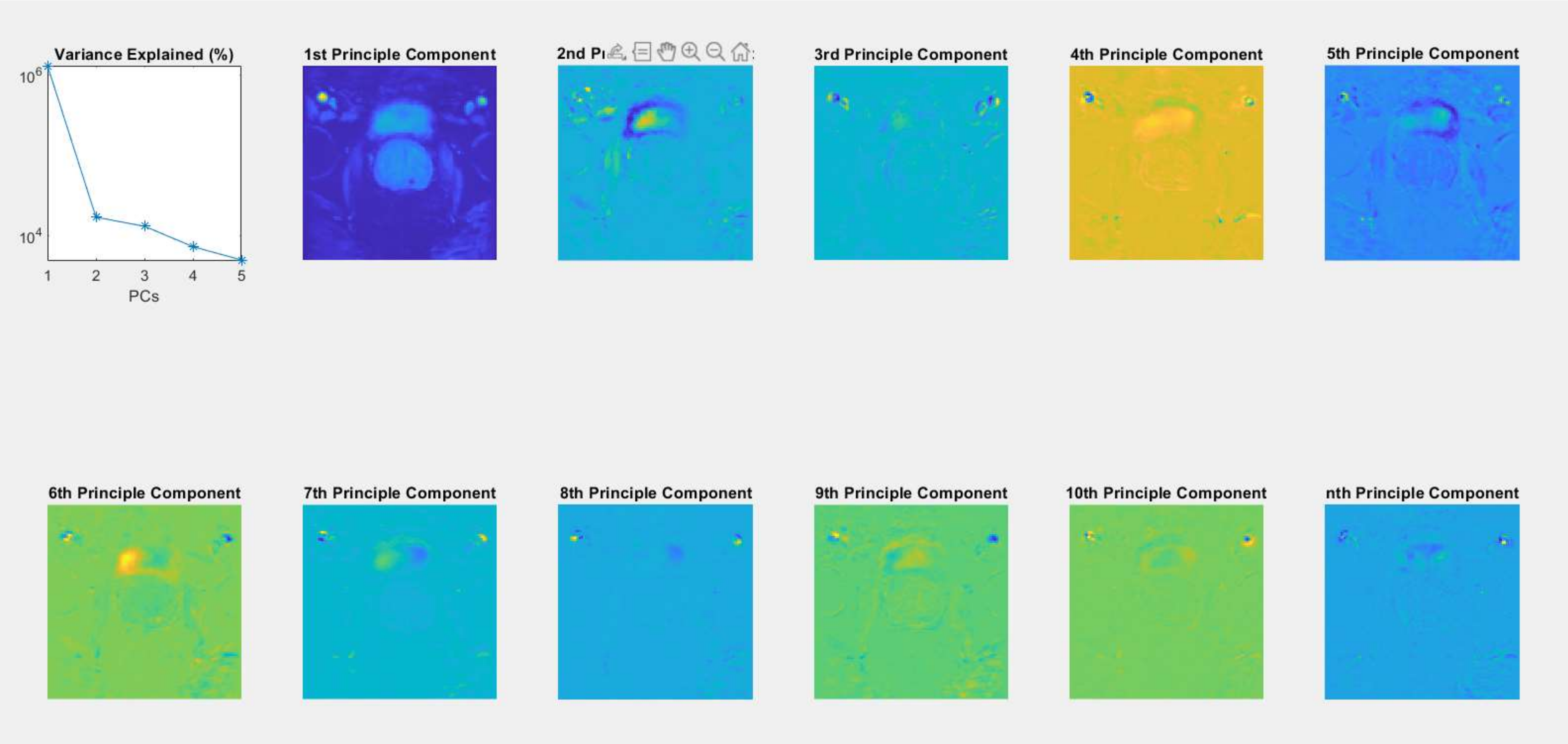
PCA generates a new set of variables: principal components.

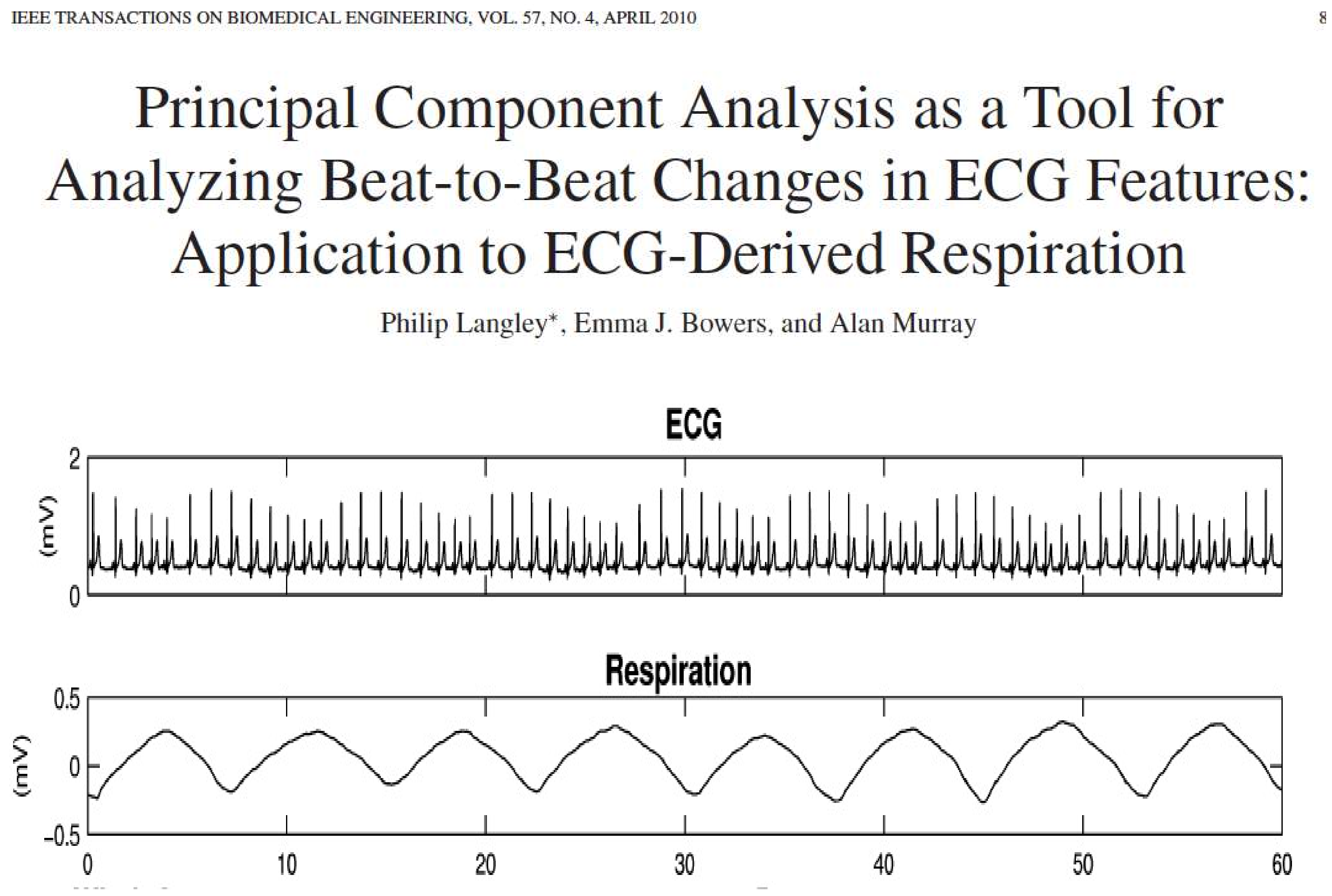
All the principal components are orthogonal to each other so there is no redundant information.





Static T2-weighted prostate image (left) and corresponding overall degree of enhancement coded green; rate of wash-in / wash-out coded red (right). *Bruwer, MacGregor, Noseworthy (2008) J. Chemometrics 22:708-716.*


Variance Explained (%)
10^6
10^4
1
2
3
4
5
PCs
1st Principle Component
2nd Pr
3rd Principle Component
4th Principle Component
5th Principle Component
6th Principle Component
7th Principle Component
8th Principle Component
9th Principle Component
10th Principle Component
nth Principle Component



IEEE TRANSACTIONS ON BIOMEDICAL ENGINEERING, VOL. 57, NO. 4, APRIL 2010 8

Principal Component Analysis as a Tool for Analyzing Beat-to-Beat Changes in ECG Features: Application to ECG-Derived Respiration

Philip Langley*, Emma J. Bowers, and Alan Murray

Independent Component Analysis

Star Trek (i.e. the Original Series). Season 1, Episode #20, “Court Martial” Aired: Feb. 2nd, 1967

Independent Component Analysis

Independent Component Analysis seeks to transform the original data set into a number of independent variables. The motivation is primarily to uncover more meaningful variables, not to reduce the dimensions of the data set.

When data set reduction is also desired it is usually accomplished by preprocessing the data set using PCA.

One of the more dramatic applications of Independent Component Analysis (ICA) is found in the ‘cocktail party problem.’ In this situation, multiple people are speaking simultaneously in the same room.

Independent Component Analysis (ICA)

- a method for extracting useful information from data.

- reveals the driving forces which underlie a set of observed phenomena. e.g. firing of a set of neurons, mobile phone signals, brain images (fMRI), stock prices, voices, etc.

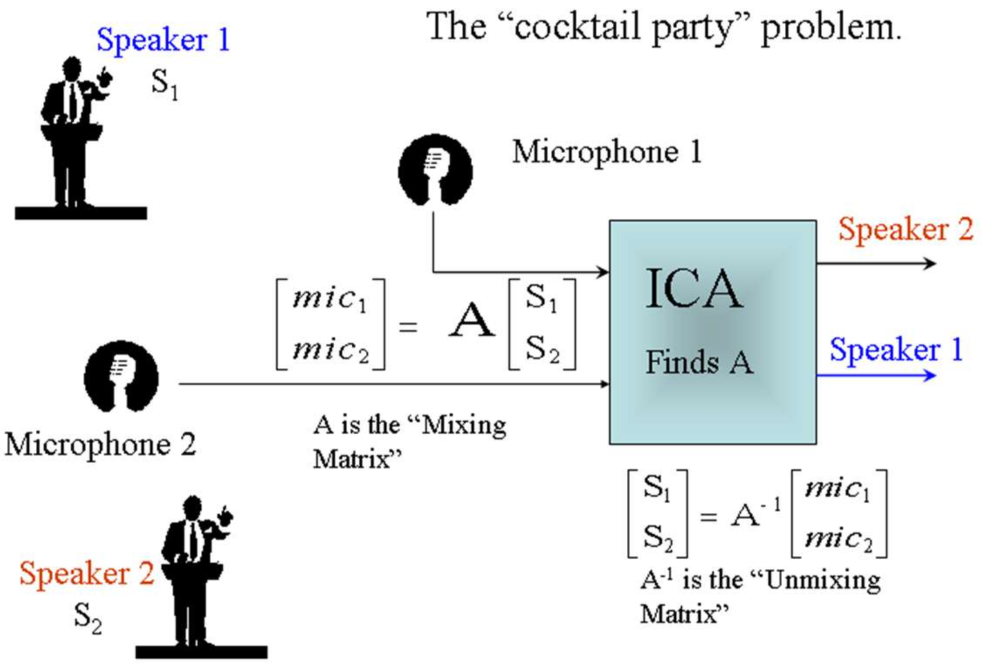
- a large set of signals are measured, and it is known that each measured signal depends on several distinct underlying factors

i.e. each measured signal is essentially a mixture of underlying factors.

The Cocktail Party Problem

ICA “unmixes” the signals in each microphone to recover the speech of each speaker.

No information is needed about either the placement of speakers or microphones nor the content of the speeches.



The “cocktail party” problem.

$$\begin{bmatrix} mic_1 \\ mic_2 \end{bmatrix} = \mathrm{A} \begin{bmatrix} S_1 \\ S_2 \end{bmatrix}$$

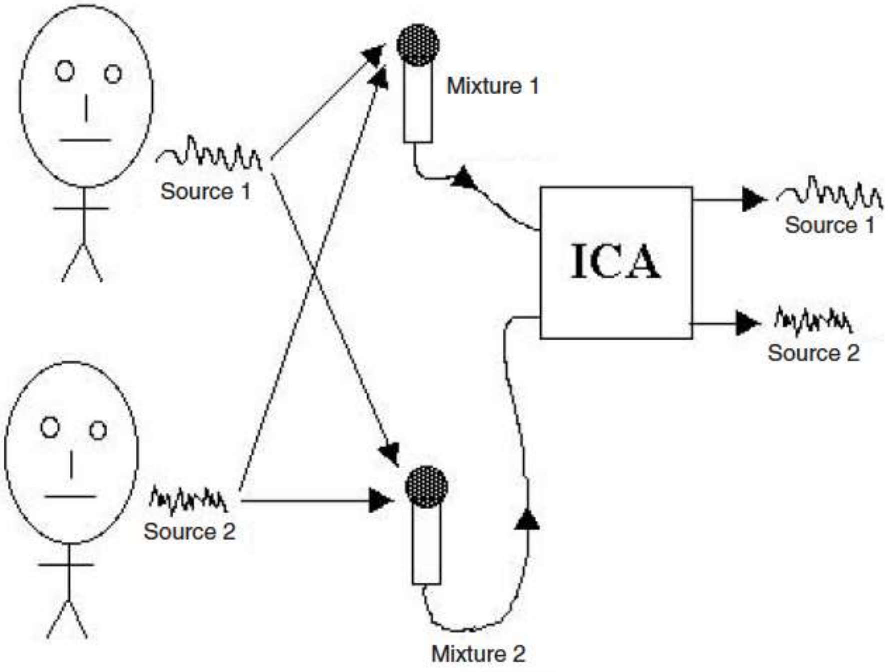
A is the “Mixing Matrix”

$$\begin{bmatrix} S_1 \\ S_2 \end{bmatrix} = \mathrm{A}^{-1} \begin{bmatrix} mic_1 \\ mic_2 \end{bmatrix}$$

A^{-1} is the “Unmixing Matrix”

(Note: Number of microphones >= number of speakers)

ICA



ICA

If each voice signal is examined at a fine time scale then it becomes apparent that the amplitude of one voice at any given point in time is unrelated to the amplitude of the other voice at that time.

The reason that the amplitudes of the two voices are unrelated is that they are generated by two unrelated physical processes (i.e., by two different people).

If we know that the voices are unrelated then one key strategy for separating voice mixtures into their constituent voice components is to look for unrelated time-varying signals within these mixtures

ICA

ICA is based on a generative model: how the measured signals are produced.

The model assumes that the measured signals are the product of instantaneous linear combinations of the independent sources. Such a model can be stated mathematically as:

$$x_i(t) = a_{i1}\, s_1(t) + a_{i2}\, s_2(t) + \ldots + a_{iN}\, s_N(t) \qquad \text{for } i = 1, \ldots, N$$

where x(t) are the signals obtained from the sources, s(t).

You have x(t), but you want s(t).

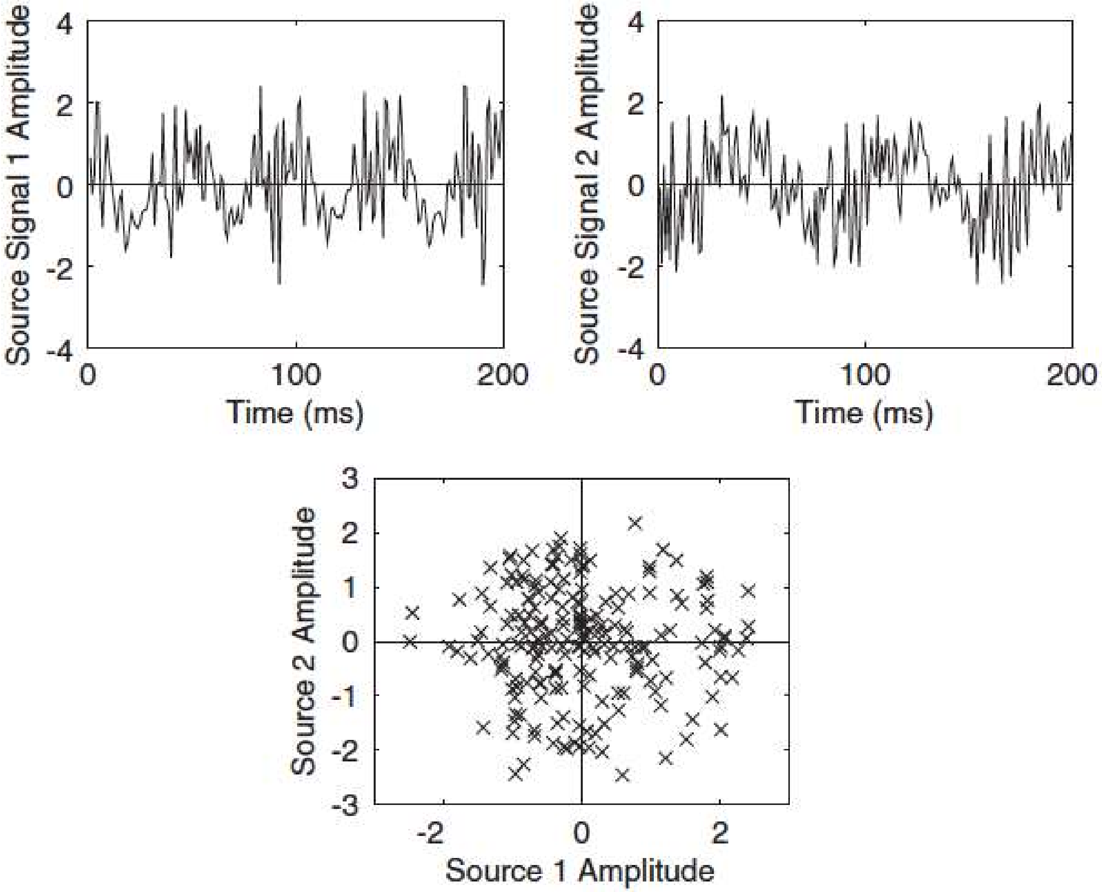
Note that this is a series of equations for the N different signal variables, $x_i(t)$, $i = 1, 2, \ldots N$

Independent Component Analysis Equations

In matrix form, this equation becomes:

$$\underset{\text{Measured}}{\begin{bmatrix} x_1(t) \\ x_2(t) \\ \vdots \\ x_N(t) \end{bmatrix}} = A \underset{\text{Hidden}}{\begin{bmatrix} s_1(t) \\ s_2(t) \\ \vdots \\ s_N(t) \end{bmatrix}} \quad \text{or} \quad \boldsymbol{x} = \boldsymbol{A}\boldsymbol{s}$$

where *s* is a vector composed of all the source signals, *A* is the mixing matrix composed of the constant elements *ai,j*, and *x* is a vector of the measured signals.


Source Signal 1 Amplitude
4
2
0
-2
-4
0
100
200
Time (ms)
Source Signal 2 Amplitude
4
2
0
-2
-4
0
100
200
Time (ms)
Source 2 Amplitude
3
2
1
0
-1
-2
-3
-2
0
2
Source 1 Amplitude

ICA

ICA separates a set of signal mixtures into a corresponding set of statistically independent component signals or source signals .

These mixtures can be sounds, electrical signals, e.g., electroencephalographic (EEG) signals, or images (e.g., faces, fMRI data).

The defining feature of the extracted signals is that each extracted signal is statistically independent of all the other extracted signals

How Independent Component Analysis Works

A form of Blind Source Separation

- ICA is based on the simple assumption that if different signals are from different physical processes (e.g., different people speaking) then those signals are statistically independent.

- ICA takes advantage of the fact that this assumption can be reversed, leading to a new assumption:

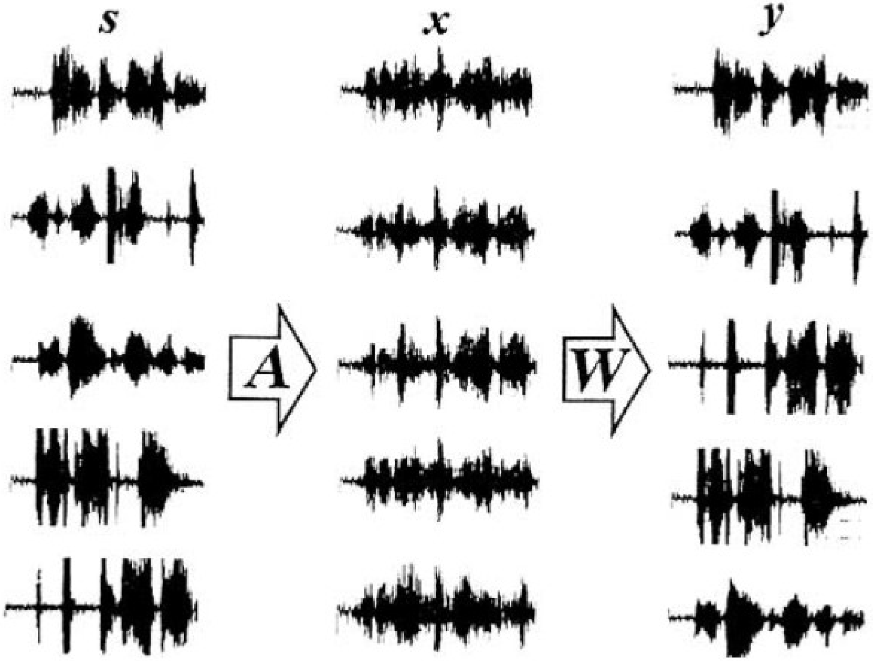
  - i.e. if statistically independent signals can be extracted from signal mixtures then these extracted signals must be from different physical processes.

The result is separation of signal mixtures into statistically independent signals.

Set of source signals (s):
- 5 people speaking

Set of Signal Mixtures (x)

Estimated extracted independent components, each an estimate of 1 of the original (y)



Mixing Process (A)
- speaker-microphone distances

Un-mixing Process (W)

Mixing Signals

When two speech source signals are mixed to make two signal mixtures 3 effects follow:

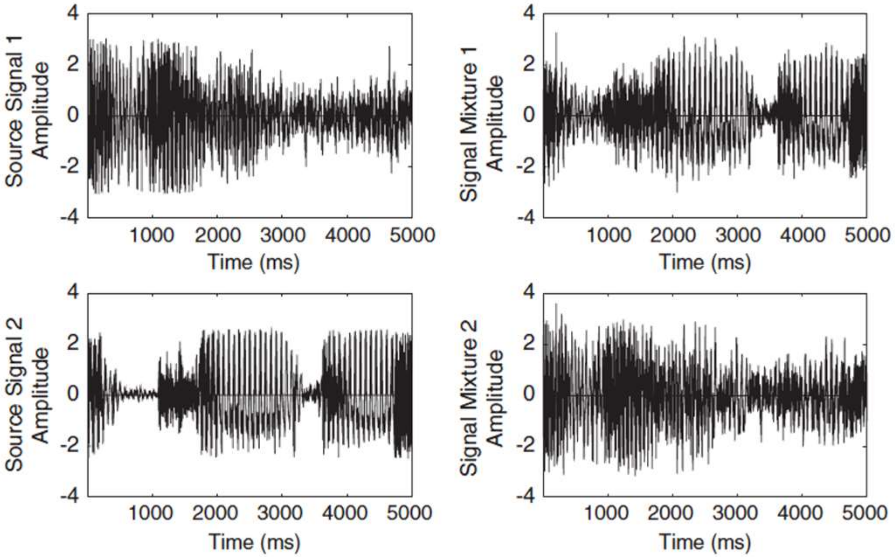
1) Independence

2) Normality

3) Complexity

Each of these effects can be used as a basis for unmixing

Voice Signals (1 and 2)

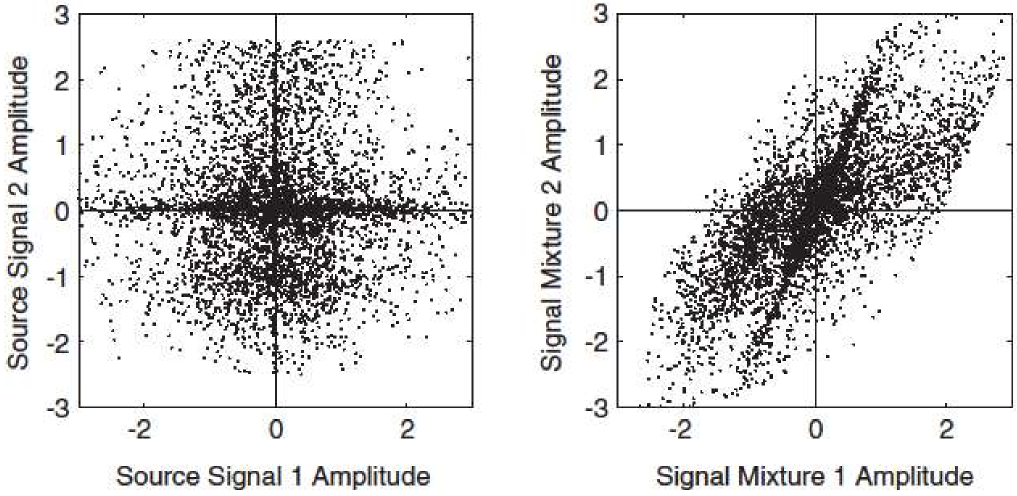


Mixtures (e.g. different microphone placements)

Independence:

Whereas speech source signals are statistically independent, their signal mixtures are not.

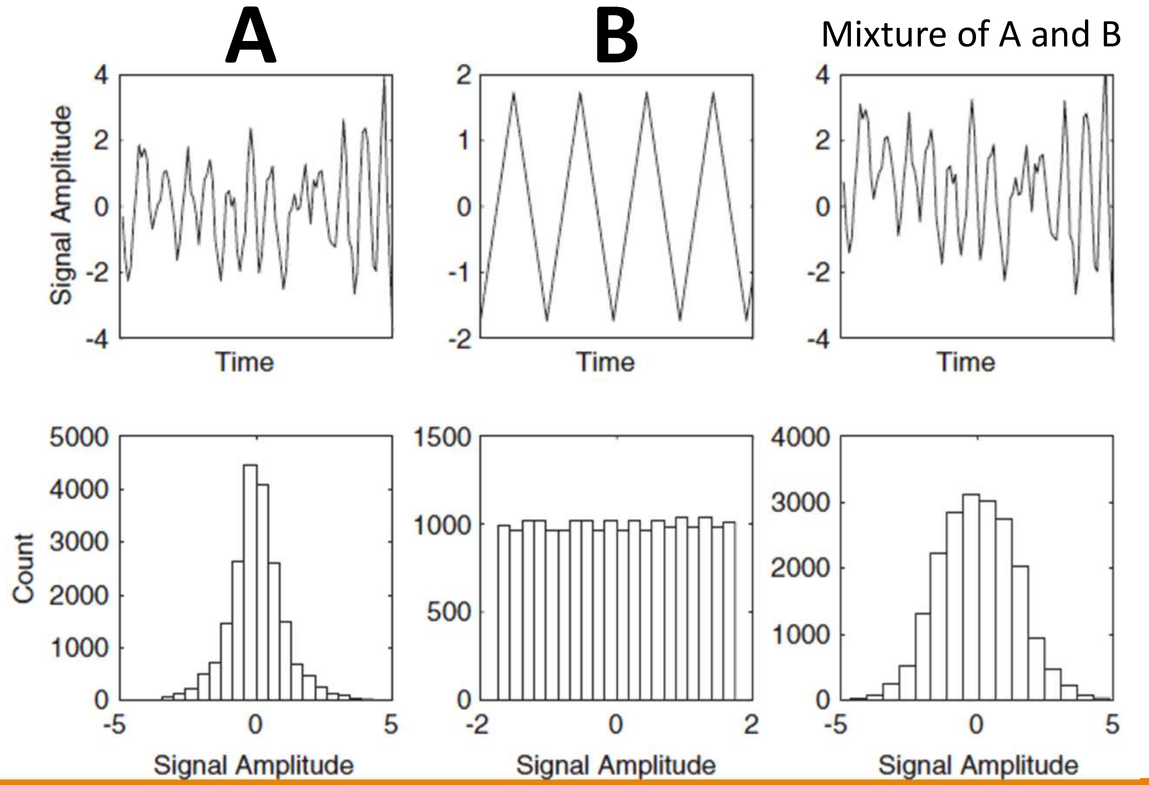
- because each source signal is shared between both mixtures such that the resultant commonality between signal mixtures ensures that they cannot be independent.



Normality:

- If the values in a speech source signal are plotted as a histogram then a Gaussian shape emerges.

- Compare to non-speech histogram of a sawtooth signal yields flat histogram.

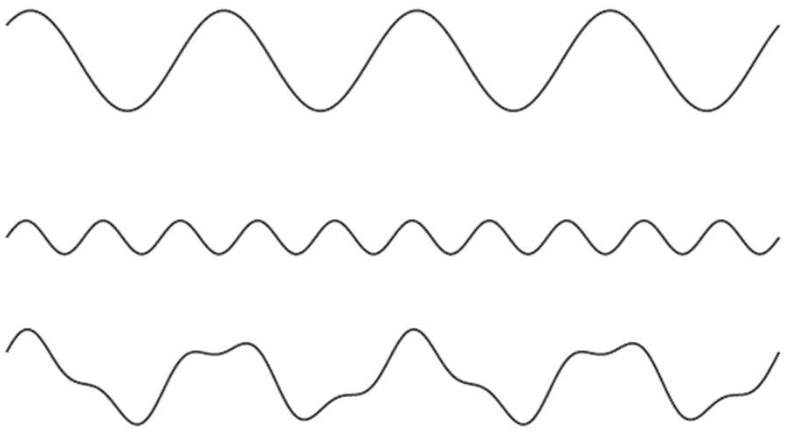


Complexity

The temporal complexity of any mixture is greater than (or equal to) that of its simplest (i.e., least complex) constituent source signal.

- This ensures that extracting the least complex signal from a set of signal mixtures yields a source signal.

- This complexity conjecture can be used as a basis for blind source separation.



Requirements for ICA

If source signals have some property X, and signal mixtures do not, then given a set of signal mixtures we should attempt to extract signals with "as much X as possible".

Thus, replacing X with independence, normality and complexity yields principles of unmixing:

1) Independence
   - If source signals are independent and signal mixtures are not then extracting independent signals from a set of signal mixtures should recover the required source signals

2) Normality
   - If source signals are non-Gaussian and signal mixtures are not then extracting signals with non-Gaussian behaviour from a set of signal mixtures should recover the required signals.

3) Complexity
   - If source signals have low complexity structure and signal mixtures do not then extracting signals with low complexity from a set of signal mixtures should recover the required signals.

For ICA:

- mixing process is specified in terms of a set of constants: mixing coefficients

- if these are known then they can be used to derive a set of unmixing coefficients, which can be used to extract source signals from signal mixtures.

$$\begin{aligned} s_1 &= (s_1^1, s_1^2, \ldots, s_1^N) \\ s_2 &= (s_2^1, s_2^2, \ldots, s_2^N) \end{aligned}$$

Two time varying signals, where s is a time-varying signal which takes amplitudes s1, s2,… sN

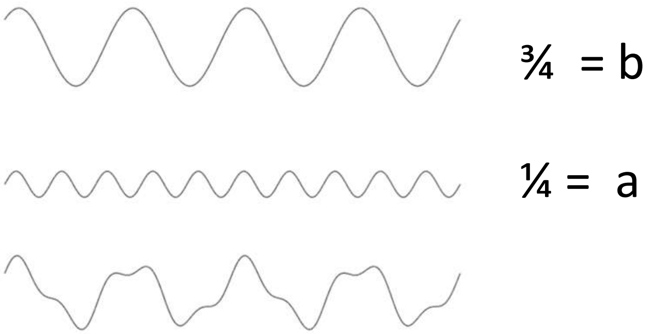
Mixing Signals

The different distance of each source from a microphone ensures that each source contributes a different amount to the mic's output x1.

As an example let's say the microphone-source distances are such that 1/4 of the mixture is from source s1 and 3/4 are from source s2.

the mixture x1 can be specified as a weighted sum of the 2 source signals.

So the mixture amplitude x1t at a given time t is the weighted sum of the source signals s1t and s2t at that time:



Mixing Signals

$$(x_1^1, x_1^2, \ldots, x_1^N) = a \times (s_1^1, s_1^2, \ldots, s_1^N) + b \times (s_2^1, s_2^2, \ldots, s_2^N)$$

Or in more compact form:

$$x_1 = as_1 + bs_2$$

What if another microphone is added?

$$x_2 = cs_1 + ds_2$$

the pair of signal mixtures *(x_1, x_2)* is analogous to the pair of source signals

Therefore *(x_1, x_2)* can be represented as a vector variable

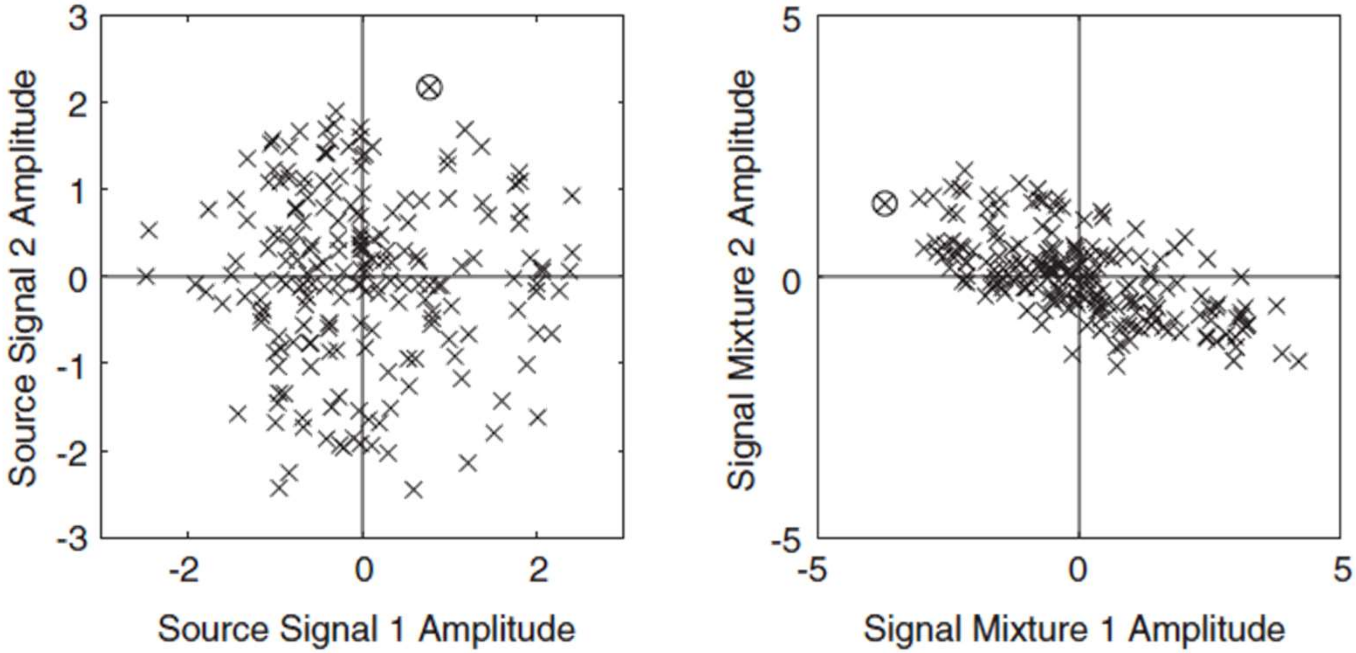
Mixing Signals

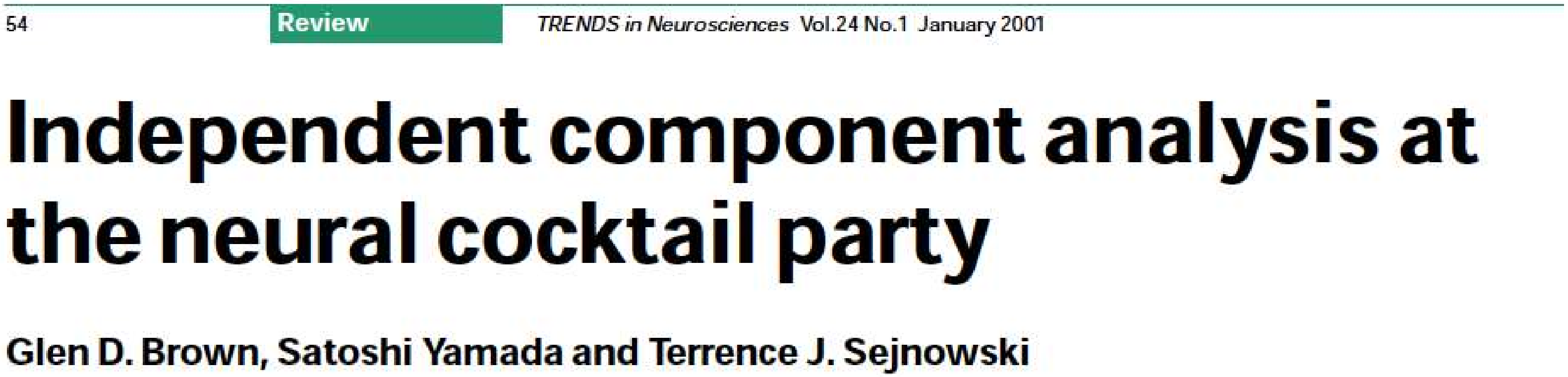
The mixing process, represented by the four mixing coefficients (a, b, c, d), transforms one vector variable s to another vector variable x

each source signal data point:

at a given time, t is transformed to a corresponding signal mixture data point:

denoted as


3
2
1
0
-1
-2
-3
-2
0
2
Source Signal 1 Amplitude
Source Signal 2 Amplitude
5
0
-5
-5
0
5
Signal Mixture 1 Amplitude
Signal Mixture 2 Amplitude

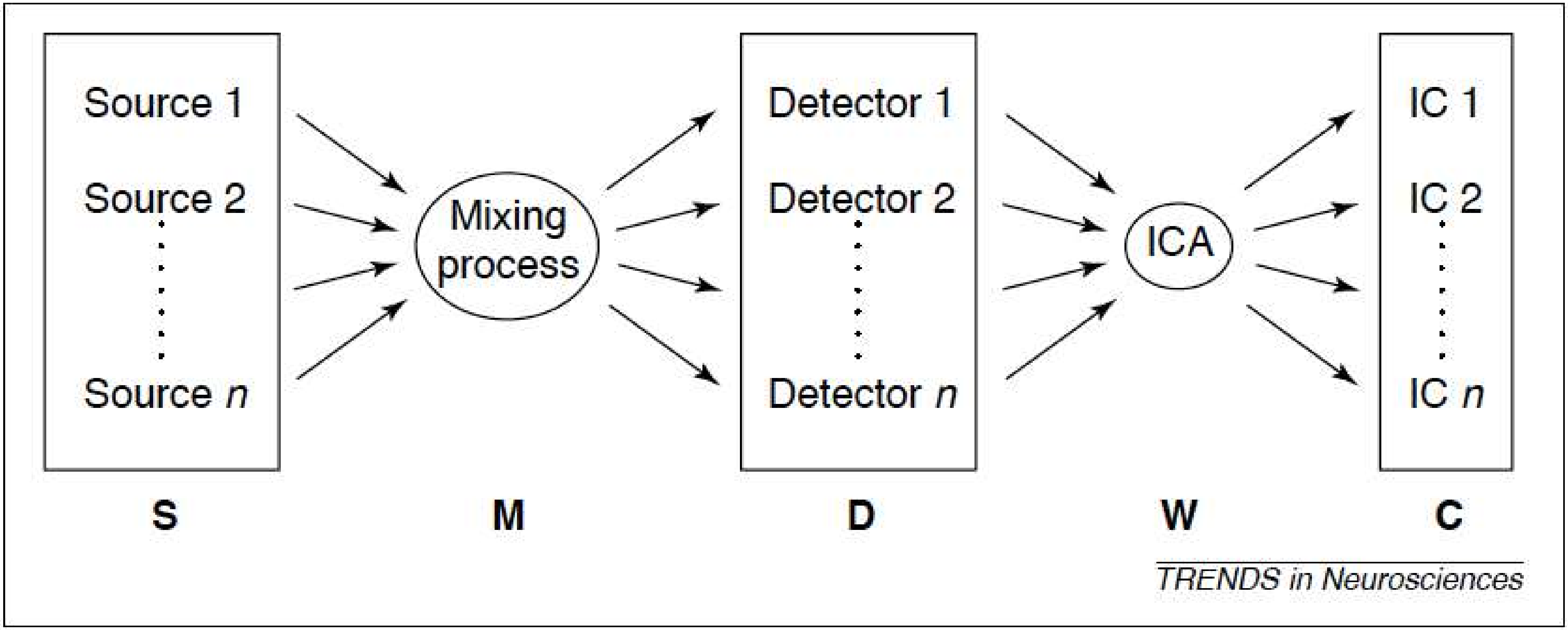


54 Review *TRENDS in Neurosciences* Vol.24 No.1 January 2001

Independent component analysis at the neural cocktail party

Glen D. Brown, Satoshi Yamada and Terrence J. Sejnowski

(Added paper to website for download)


Source 1
Source 2
Source n
Mixing process
Detector 1
Detector 2
Detector n
ICA
IC 1
IC 2
IC n
S
M
D
W
C
TRENDS in Neurosciences

Example from the Article

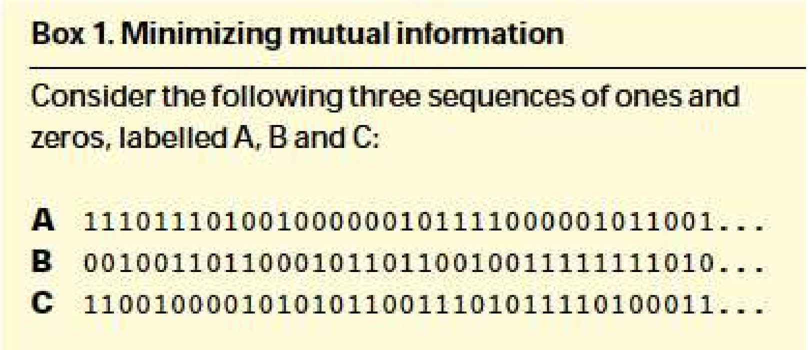
Each sequence is a measured variable

And each column is a different time point

C in this case is the XOR of A and B

0 Covariance…

PCA?



Box 1. Minimizing mutual information

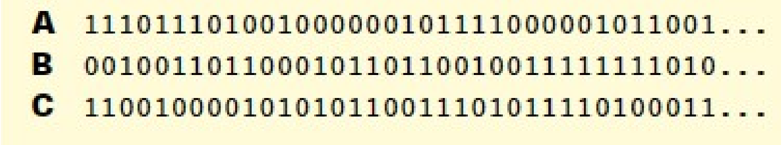
Consider the following three sequences of ones and zeros, labelled A, B and C:

```
A  111011101001000000101111000001011001...
B  001001101100010110110010011111111010...
C  110010000101010110011101011110100011...
```

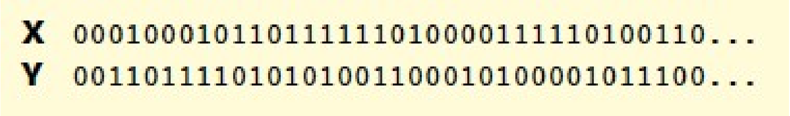
Example from the Article

ICA?

There are 8 possible combinations of 3 binary digits (2^3)



```
A  111011101001000000101111000001011001...
B  001001101100010110110010011111111010...
C  110010000101010110011101011110100011...
```



```
X  000100010110111111010000111110100110...
Y  001101111010101001100010100001011100...
```

This is called a 3rd order redundancy relationship, non linear

ICA can minimizes redundancy in data, no matter the order

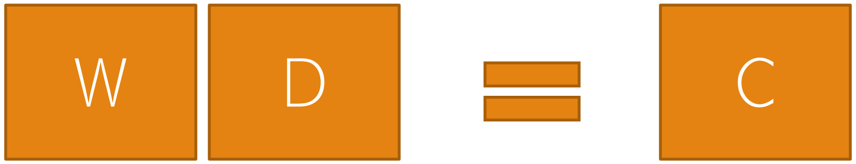
Normally data with mutual information has covariance AND higher order dependencies

Example from the Article

Signals from each detector make up D

Each column of D is t

ICA will create a square matrix W(m=n=# of detectors) so that



W D = C

C are the independent components and they are forced to be as independent as possible

W is the unmixing matrix, each row is an unmixing function

Mixing Matrix

ICA techniques are used to solve for the unmixing matrix, A^{-1}, from which the independent components, s, can be obtained:

$$s = A^{-1}x$$

where A^{-1} is the unmixing matrix.

If you know the mixing matrix, A finding the unmixing matrix is easy. But usually you do not know A.

If the measured signals, x, are related to the underlying source signals, s, by a linear transformation (i.e., a rotation and scaling operation), then some inverse transformation (rotation/scaling) should recover the original signals.

Finding the Unmixing Matrix: A^{-1}

To estimate the mixing matrix, ICA needs to make basic two assumptions:

1. The source variables, s, are truly independent.
2. They are non-Gaussian.

Both conditions are usually met when the sources are real signals.

A third restriction is that the mixing matrix must be square: the number of sources should equal the number of measured signals.

ICA Limitations

ICA can only be applied to non-Gaussian signals because it relies on higher-order statistics to separate the variables.

- Higher-order statistics (i.e. moments and related measures) of Gaussian signals are zero.

Since ICA has available only the measured variables there are some limits to what ICA can do:

- ICA cannot determine the variances, hence the energies or amplitudes, of the actual sources.
- Unlike PCA, the order of the components cannot be established. (Logical if amplitudes cannot be determined.)

ICA Algorithms

The unmixing matrix is determined by optimizing some “objective function” related to independence of data.

There a large number of ICA algorithms that have been developed. Many are available as downloadable MATLAB files.

All approaches use optimization. They differ in:

1. The specific objective function that is optimized .
2. The optimization method that is used.

Optimization Criteria

One of the most intuitive approaches uses an objective function that is related to the non-Gaussianity of the data set.

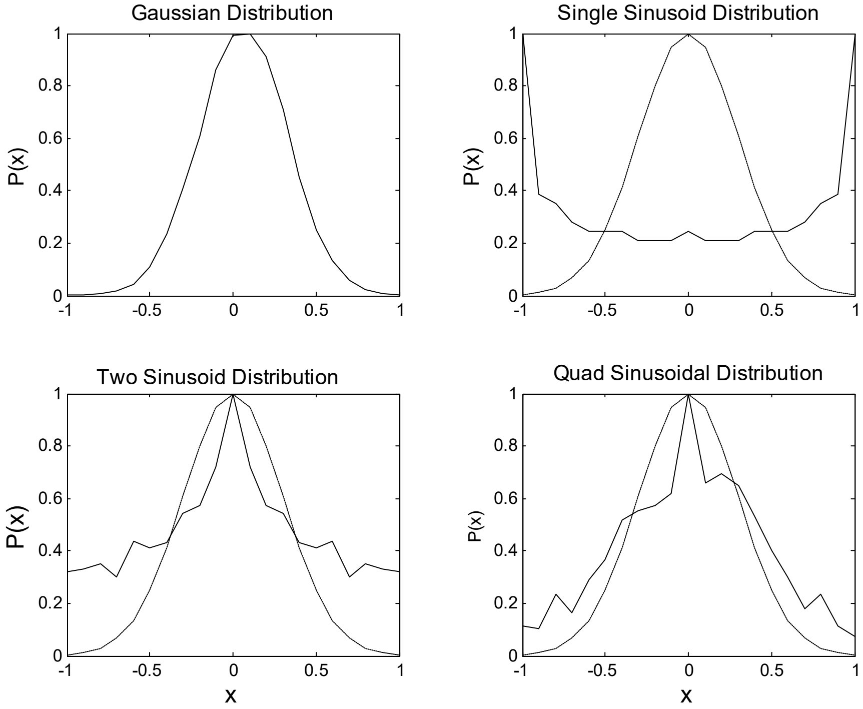
This approach takes advantage of the fact that mixtures tend to be more Gaussian than the distribution of independent sources.

Mixtures of non-Gaussian sources will be more Gaussian than the unmixed sources

Central Limit Theorem

the sum of k independent, identically distributed random variables converges to a Gaussian distribution as k becomes large

regardless of the distribution of the individual variables.



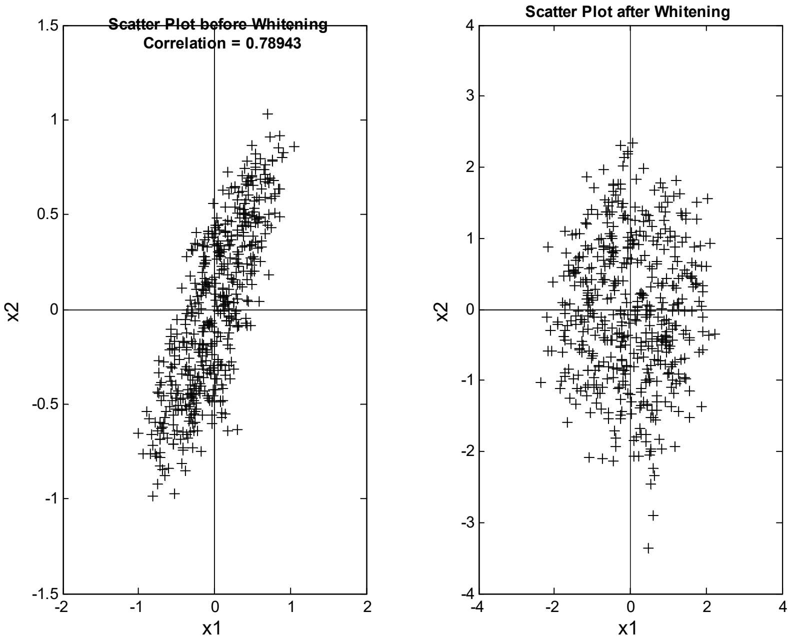
The distribution of a single sinusoid is markedly non-Gaussian.

Mixtures of even two sinusoids have distributions that look Gaussian.

Data Whitening

Most ICA algorithms begin with “data whitening:”

- Centering the data (zero means).
- Decorrelating the data (PCA)
- Scaling the data so the variances equal to 1.0



ICA Algorithm (continued)

One approach to quantifying non-Gaussianity is to use kurtosis:

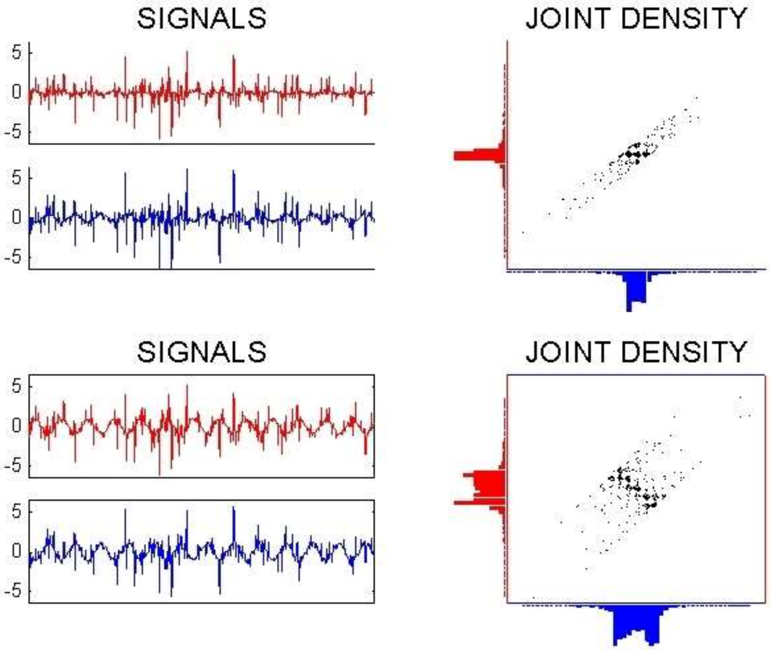
For data with zero mean: $E\{x^2\}= \boldsymbol{\sigma}^2$ and if the data are whitened, $\boldsymbol{\sigma}^2 = 1$, so the kurtosis equation becomes

Using a metric that involves the 4th power greatly enhances the influence of outliers so a nonlinear function is often used to compress the data before taking the 4th power

Example Using NonGaussianity

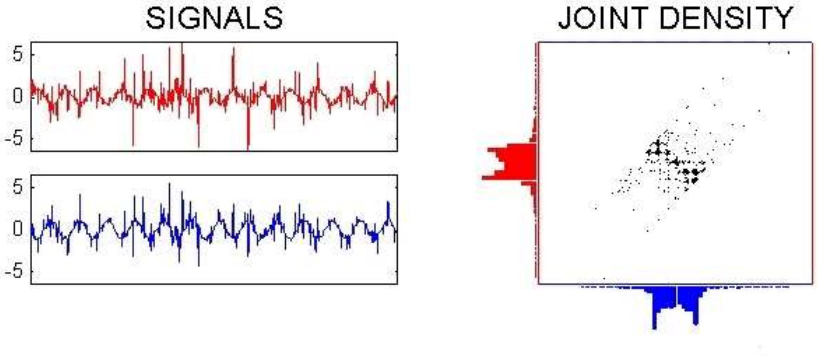
Original

Whitened

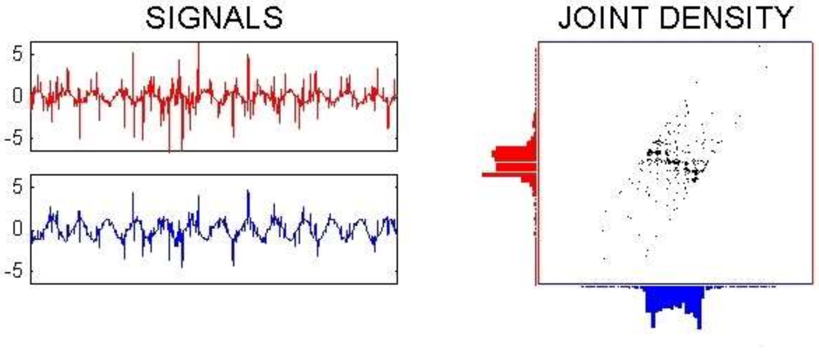


Whitened signals and density

ICA Process

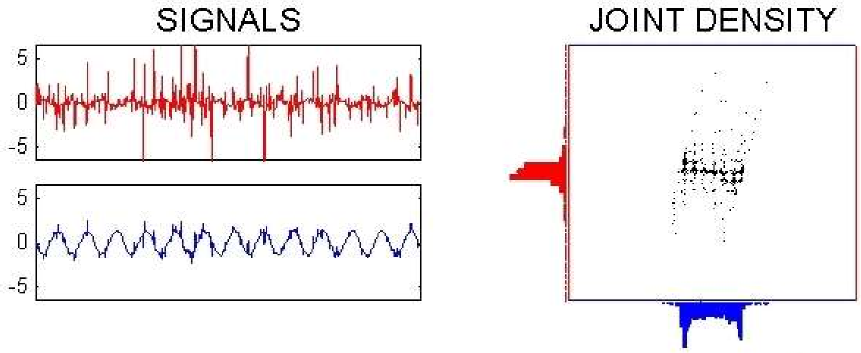


Separated signals after 1 step of FastICA

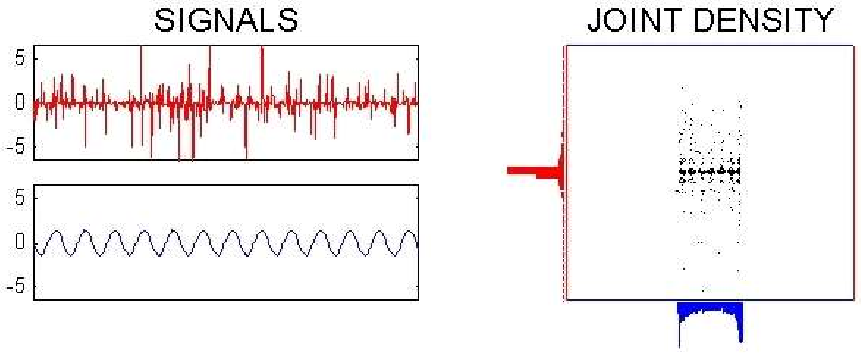


Separated signals after 2 steps of FastICA

ICA Process

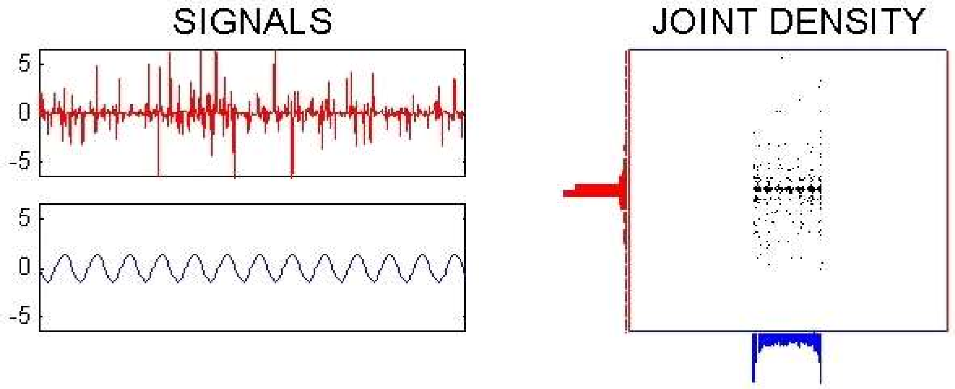


Separated signals after 3 steps of FastICA



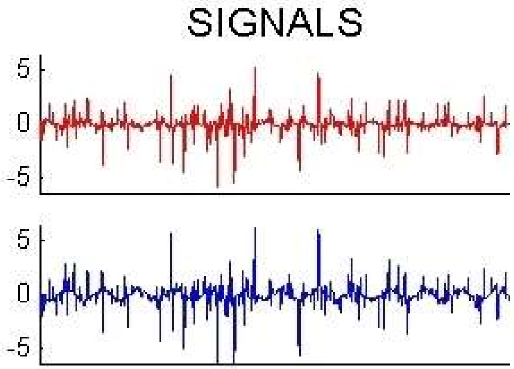
Separated signals after 4 steps of FastICA

Final Rotation

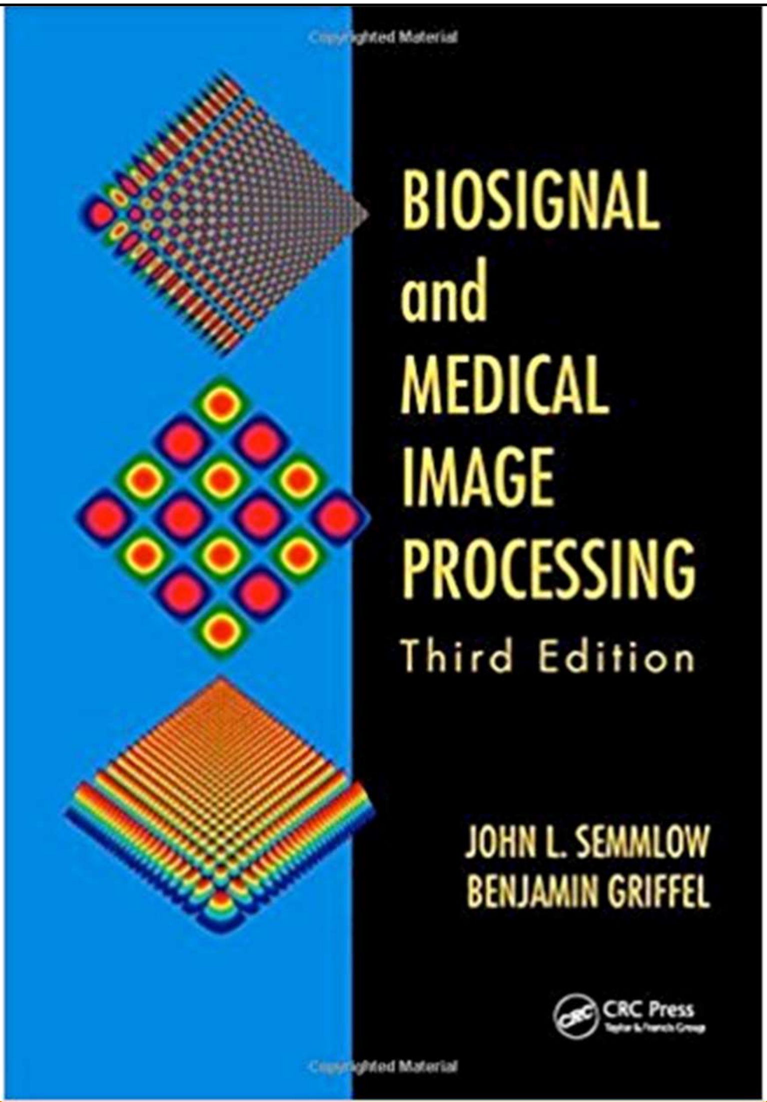


Separated signals after 5 steps of FastICA

Original Signals



PCA/ICA Reference material in this lecture:



BIOSIGNAL and MEDICAL IMAGE PROCESSING

Third Edition

JOHN L. SEMMLOW
BENJAMIN GRIFFEL

CRC Press
Taylor & Francis Group