# The Landscape of Dimension Reduction Methods

Mark Reimers, PhD
Neuroscience Program,
Michigan State University

# Goals of Dimension Reduction for Neural Data

- As we record more neurons, pixels, or voxels, it gets harder to interpret or recognize patterns
- Our visual system is best at apprehending subtle unexpected patterns
  - But… we can display all these measures!
- Thinking of measures as dimensions, we want to reduce dimensions to visualize

- For some purposes (e.g. electrical recordings, 2P images) we want to identify true sources

# Dimension Reduction Approaches

Unbiased (sometimes called ‘blind’) approaches

- Linear: PCA/SVD; FA; ICA
- Nonlinear: Isomap; diffusion maps

Constrained approaches

- NMF; sparse PCA

Biased (outcome informed) approaches

- Systematic approaches:
  - Directed PCA; Sufficient Dimension Reduction
- Ad-hoc approaches
  - Mante, Machens, Churchland, Pillow, etc.

Dynamic time series methods

- Dynamic mode decomposition; JADE

# Unbiased Dimension Reduction Algorithms

***Linear approaches***

- Principal Component Analysis (PCA)/Singular Value Decomposition (SVD)
- Factor Analysis (FA)
- Independent Component Analysis (ICA)
- Non-negative matrix factorization (NMF)

***Nonlinear approaches***

- Isomap
- Diffusion maps
- T-SNE
- … many more

# When to Use Different Linear Methods

If you care about representing most variation AND measures follow roughly bell-shaped distribution

Use PCA

If you are trying to extract underlying sources from a mixture

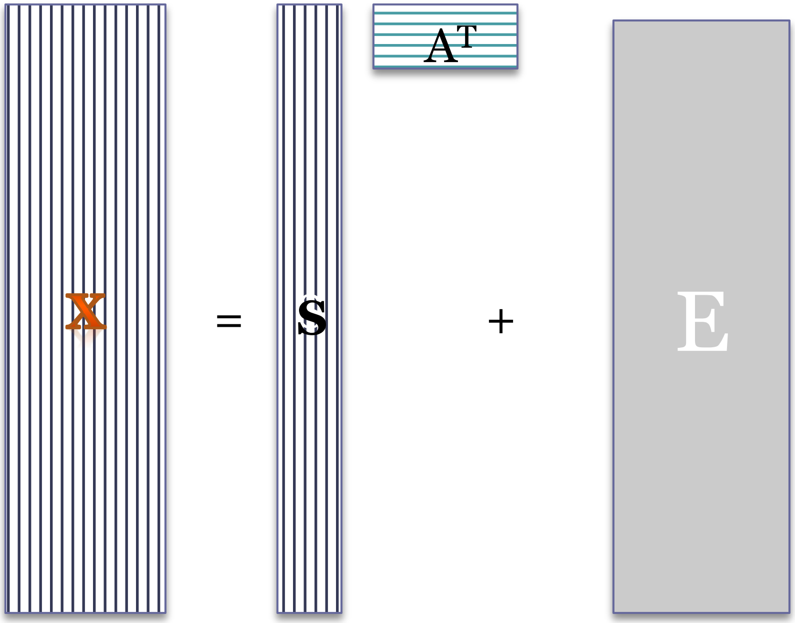
If all components are positive, use NMF

If mixed sign, try ICA but cross-validate!

- Works well for eye-blink artifacts in EEG

# Linear Dimension Reduction Expressed in Terms of Data Matrix

- $X = SA^T + E$, where X represents the data matrix with measures (electrodes, pixels, …) in columns
  - S represents the values (scores) of factors (left singular vectors)
  - A represents the inferred linear combinations (loadings) of factors in terms of variables
  - E are (small) residual errors
- Sometimes seen as



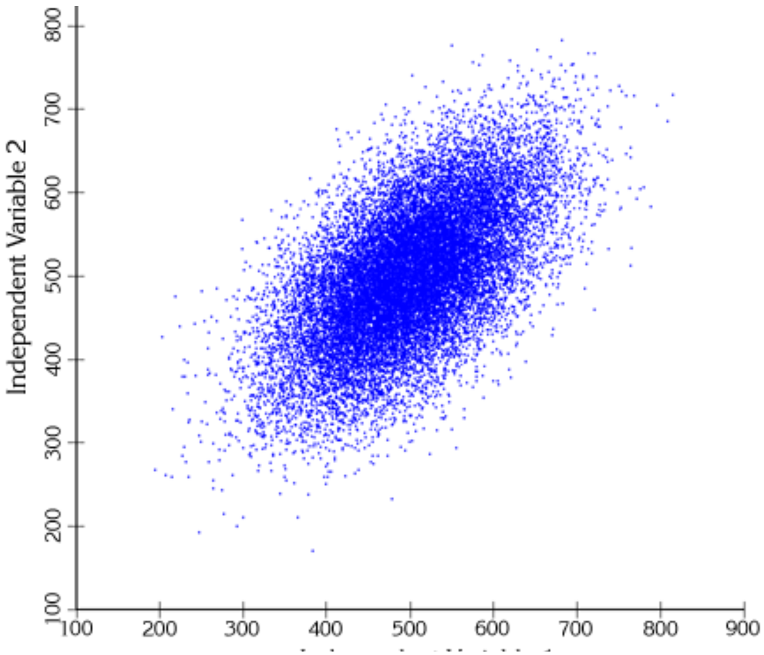
$X = AS^T+E$, where the rows of X represent the measures and the columns are samples or frames (engineering convention)

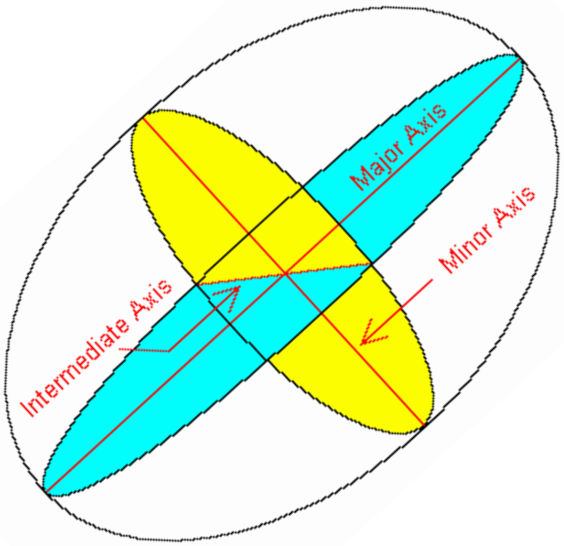
# Principal Components Analysis (PCA)

- The 'workhorse' method for visualizing neural data
- Summarizes variation most efficiently of all methods
- Three main issues
    - Works for measures with roughly Gaussian distribution
        - Less well for skewed data
        - Doesn't work at all for data with outliers
    - components are not usually meaningful
    - doesn't fully represent really high-dimensional neural data

# Principal Components

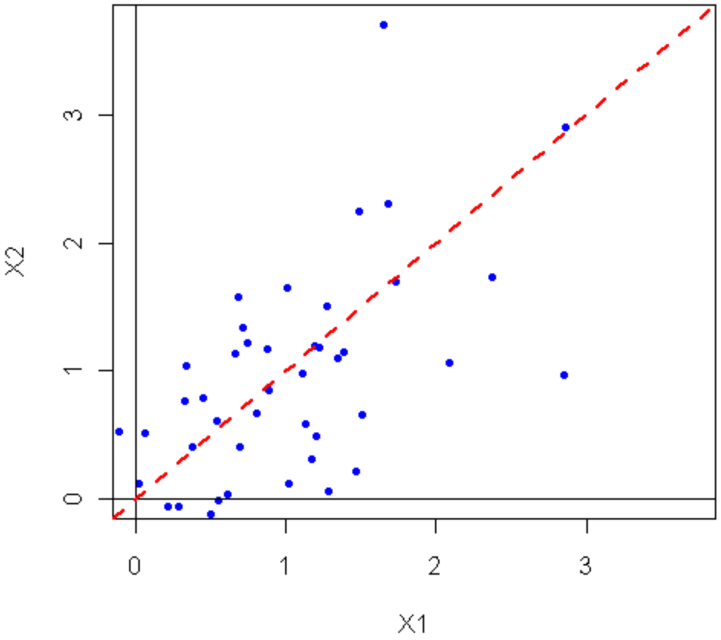
PCA is rooted in the classical theory of the multivariate Gaussian (Normal) distribution. The contours of constant density form ellipsoids; the major axes of the ellipsoid around the data cloud, are defined by the eigenvectors of the covariance matrix. If the data are Normal, the covariance matrix is all we need to know. If the data are very different from Normal, then the theory won’t work.

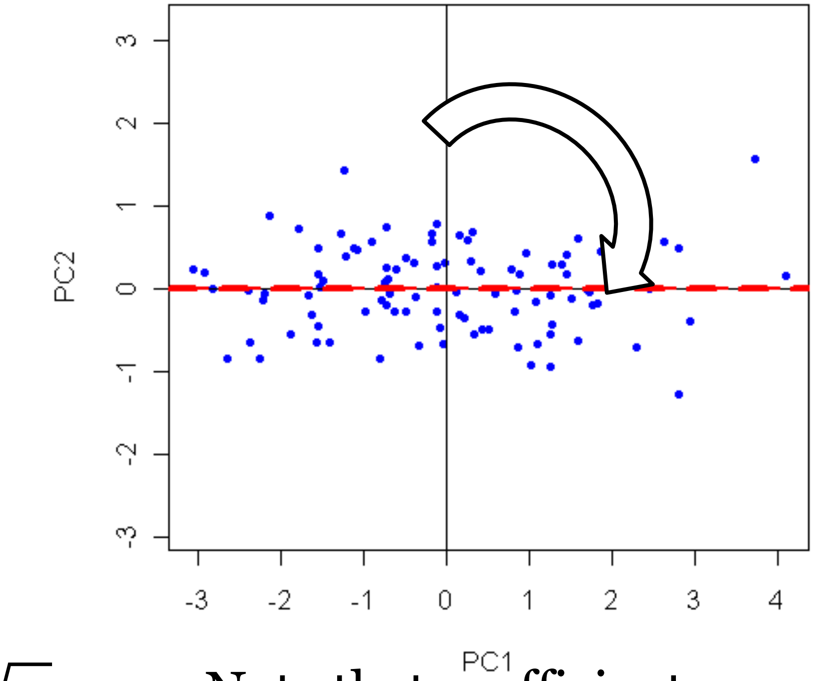




# What PCA Does

PCA rotates the co-ordinate axes so that the new coordinate axes line up with the long axes of the cloud of data points





$$\hat{s}_1(t) = (x_1(t) - x_2(t)) / \sqrt{2}$$

$$\hat{s}_2(t) = (x_1(t) + x_2(t)) / \sqrt{2}$$

Note that coefficients are ‘normalized’ so that the sum of squares of each combination is 1

# How Accurate is PCA? - Variances

- Mathematical theory tells us that the PCA estimates of variances (eigenvalues) are accurate to within about

$$\sqrt{\frac{2}{N}}\,\lambda$$

… provided that the distributions are fairly Normal

- Example: if we have an estimated variance of 10, using a sample of N = 300 then the uncertainty (SD) of that estimate is

$$\sqrt{\frac{2}{N}}\,\lambda = \sqrt{2/300}\;10 = 0.816$$

# How Accurate is PCA? - Loadings

- The accuracy of loadings (coefficients) depends crucially on how distinctive the corresponding variance (eigenvalue) is
- For the largest eigenvalues the variance of the estimates are like

$$\varepsilon_k \approx \lambda_k \sum_{j \neq k} \frac{\lambda_j}{\lambda_k - \lambda_j}$$

- Loadings for small eigenvalues can never be estimated well

# Strengths and Weaknesses of PCA

**Strengths**

- Clear mathematical theory with error bounds on estimates of eigenvalues and on loadings
- The procedure always gives the same result on any data set, unlike most other methods

**Weaknesses**

- Cannot get good estimates of loadings or of time courses for smaller components
- Extreme points (outliers) influence components very strongly, and may even define their own component
- Loadings are often hard to interpret
  - Rarely 'sparse'

# Factor Analysis

# What Factor Analysis Does

- Factor analysis tries to find factors that capture the correlations better than PCA but may not explain as much of the individual variation of each measure.
    - Better if most inputs are unobserved
- Factor analysis is a statistical model, with an error estimate, rather than a geometric technique like PCA
- Factors are ‘rotated’ to be sparse (and often positive)
    - Factors not uniquely defined
    - Scores are less accurately known

# How Many Factors?

- Factors are not well-defined if KP > P(P-1)/2
- A more sophisticated theory based on likelihood tells us how likely it is that we could observe a the covariance matrix that we see with a given factor structure

  … provided the data are Gaussian
- If we aren't likely to observe it then probably there are more factors

# The Factor Model and Rotation

- The factor model S = LL’ + Ψ, where L is a p x k matrix of loadings
- We could take any orthogonal combinations (rotations) of these factor loadings and get the same covariance matrix, because LOO’L = LL’
- We typically rotate the factors to obtain desirable properties – e.g. sparse loadings, hence easier interpretation

# Terminology and Conventions

- Terminology
  - ▫ Uniquenesses
    - How much variance is left over for each variable that is not explained by common factors
- Conventions:

Factor scores are scaled to have variance 1; therefore loadings are scaled to represent contribution to variance. Unlike PCA, factor loadings do not have sum of squares 1.

# PCA and Factor Analysis

- PCA & FA assume that all measures and factors have a Normal distribution
  - They work best when this is roughly true
- In both PCA and FA the factors are uncorrelated (independent if they have Normal distribution)
- In PCA and FA the linear coefficients defining different factors are orthogonal vectors
- Both PCA and FA are ambiguous about the what the factors mean (or if they are real)
  - Drivers, sources, or summaries?

# Factor Loadings Differ from PCA

- Suppose four measures $x_1$, $x_2$, $x_3$, $x_4$.
  - ▫ $r_{1,2} = .8$; $r_{3,4} = .8$; $r_{1,3} = r_{2,4} = .4$
- Then PCA will find two strong PCs
  - ▫ (½, ½, ½, ½); (½, ½, -½, -½)
- Factor analysis will find two factors
  - ▫ (.7, .7, 0, 0); (0, 0, .7, .7)

# What Do Factors/Components Mean?

- In most cases the loadings identify coalitions of neurons or regions, which are active together more often than if they were independent
- These are not causal structures but may point you to causal structures

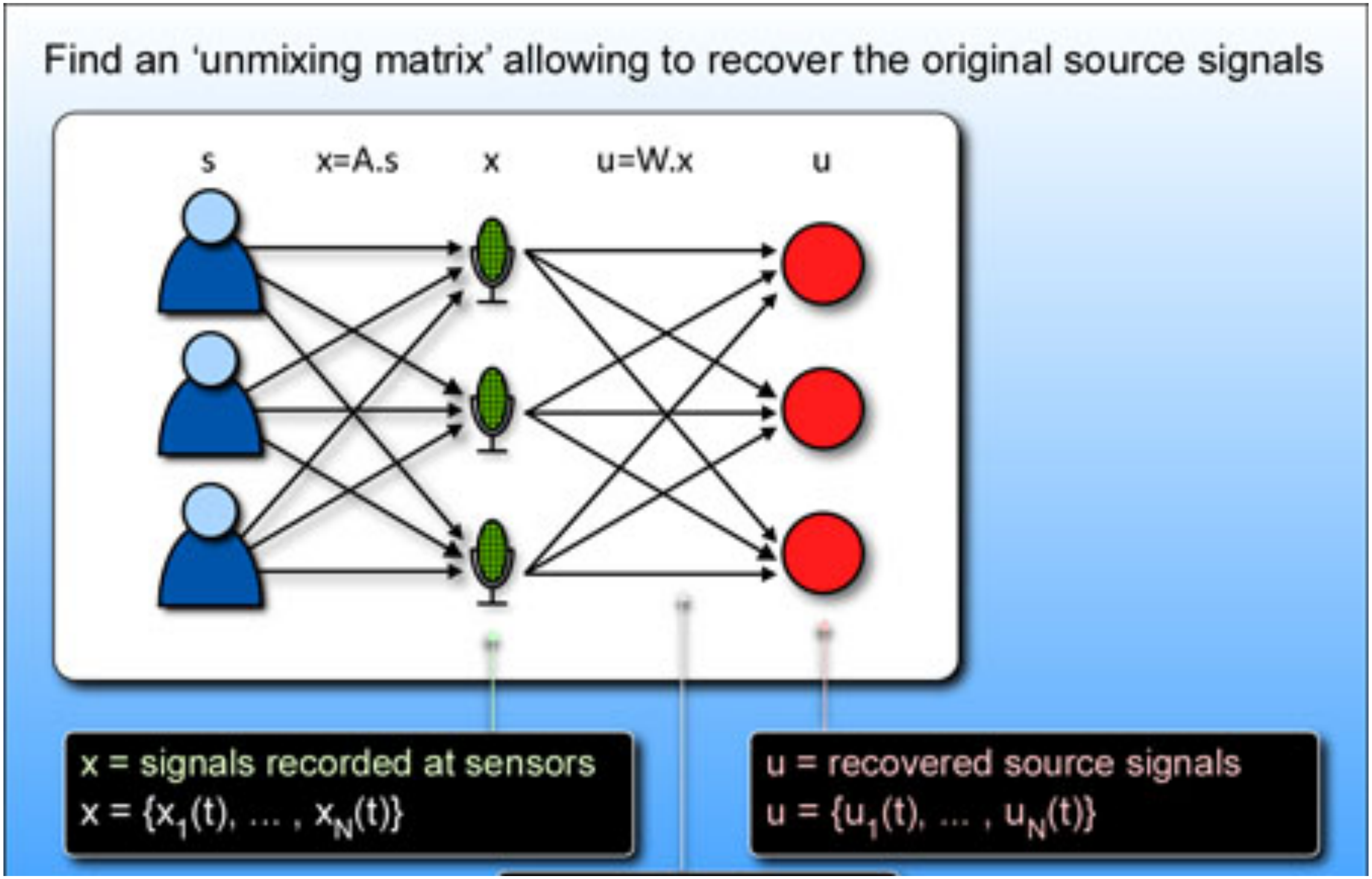
- Sometimes (spike sorting, 2P image analysis) we seek underlying sources

# Independent Components Analysis (ICA)

# The Ideas of ICA

- What if the components are not Gaussian?
- What if the loadings are not orthogonal?
- This is often the case
- Theory based on Gaussian variables may be misleading
- A method that enforces orthogonal axes cannot estimate non-orthogonal loadings well

# The ‘Cocktail Party Problem’



Find an ‘unmixing matrix’ allowing to recover the original source signals

x = signals recorded at sensors

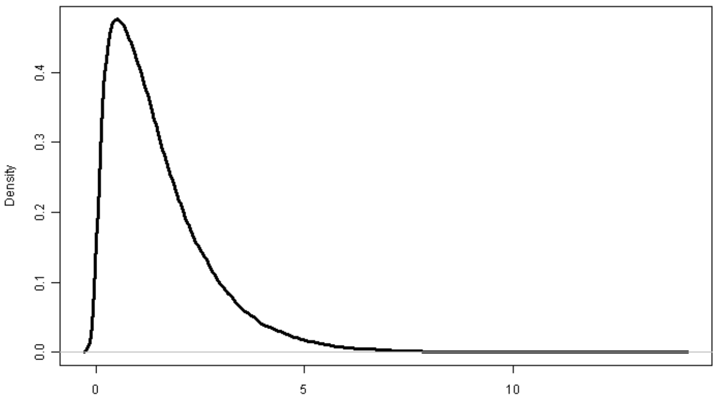
$x = \{x_1(t), \ldots, x_N(t)\}$

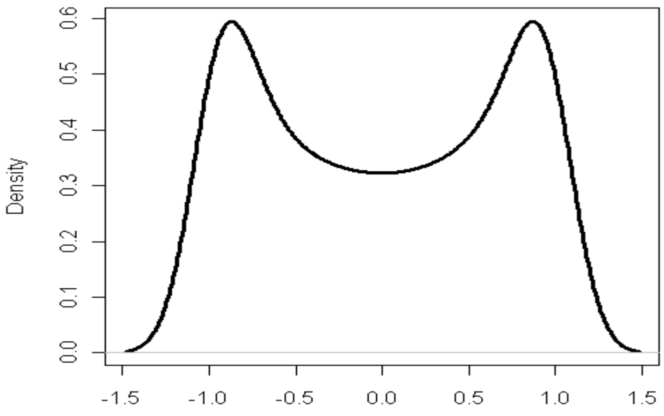
u = recovered source signals

$u = \{u_1(t), \ldots, u_N(t)\}$

# The ICA Model

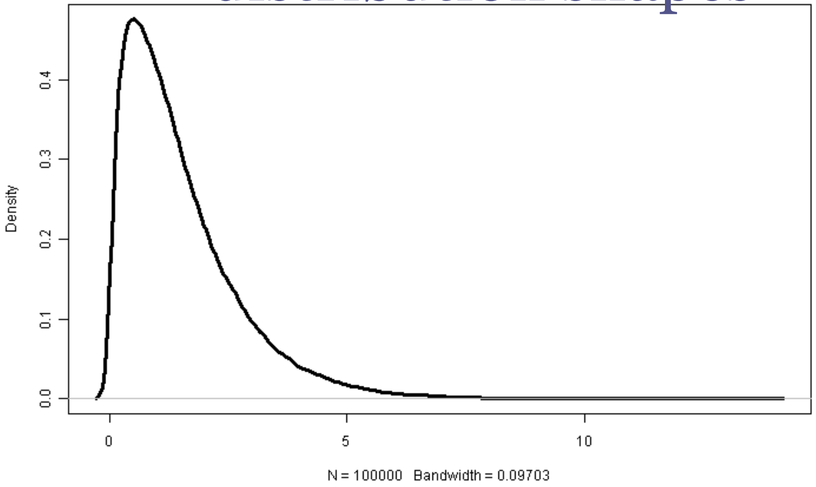
- ICA was designed to distinguish sources that may not be orthogonal
- The theory is that ICA looks for maximally statistically independent combinations - factors that give no information about each other (statistically independent)
- This is equivalent to a search for factors with the most non-Normal distributions possible

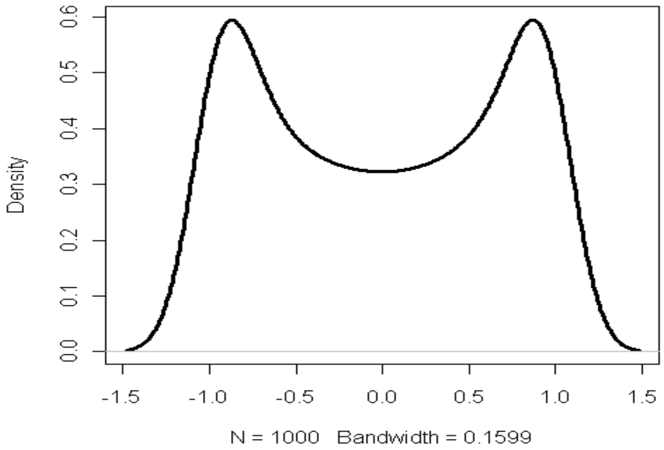




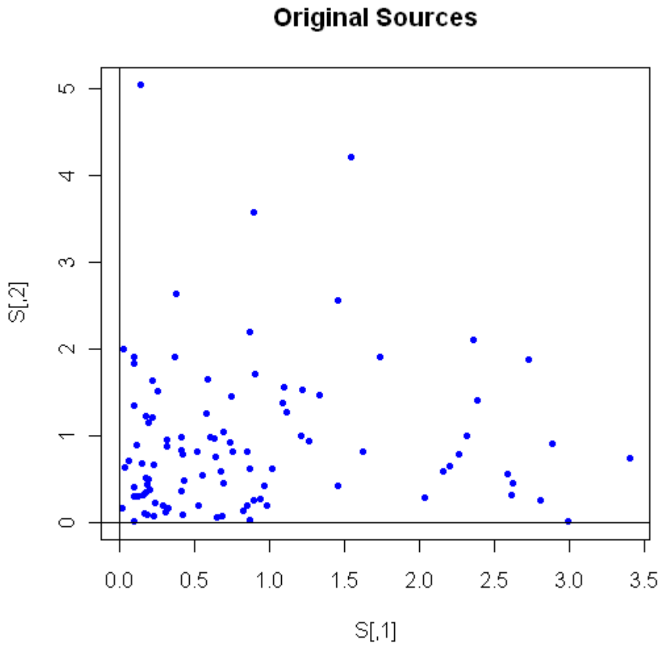
# ICA vs. PCA

- Therefore PCA and ICA attempt to find linear combinations of variables such that:
  - PCA maximizes the ‘size’ of the factors
    - This also ensures that factors are uncorrelated
  - ICA tries to find factors that give no information about each other (statistically independent)
    - Using information theory, this is equivalent to finding factors with the most distinctive non-Normal distribution shapes

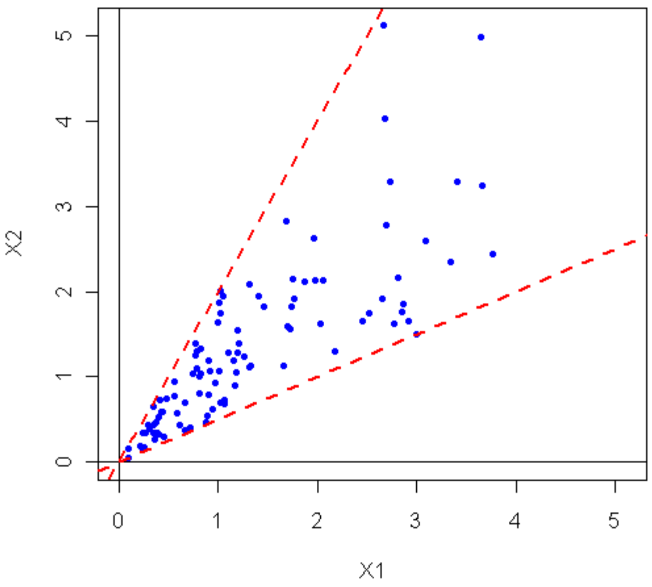




# ICA Example I – Two Sources



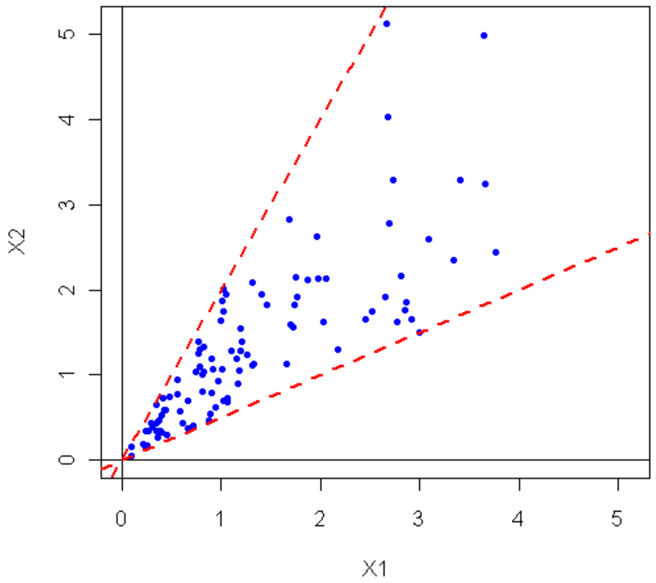
Observed Mixture

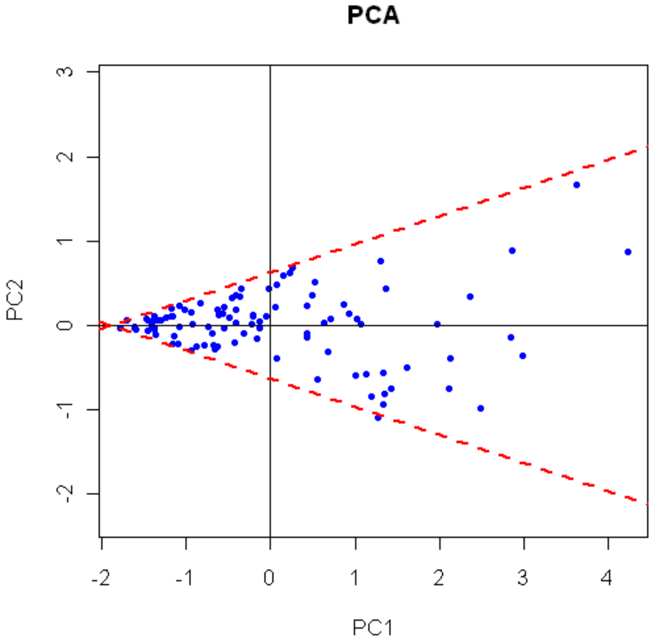


- Underlying sources at top left are mixed to give observed measures at upper right
  - $x_1 = 2/3s_1 + 1/3s_2$
  - $x_2 = 1/3s_1 + 2/3s_2$
- Original coordinate axes are now dashed red

# ICA Example I – Two Sources

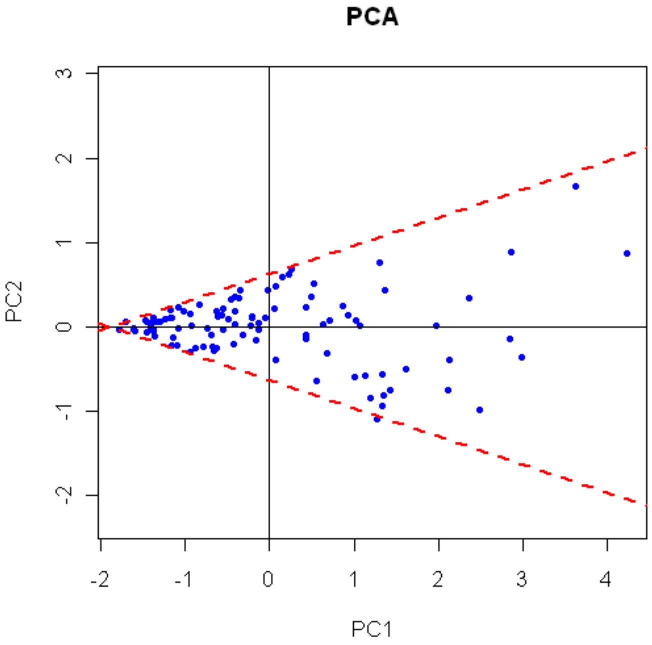
Observed Mixture



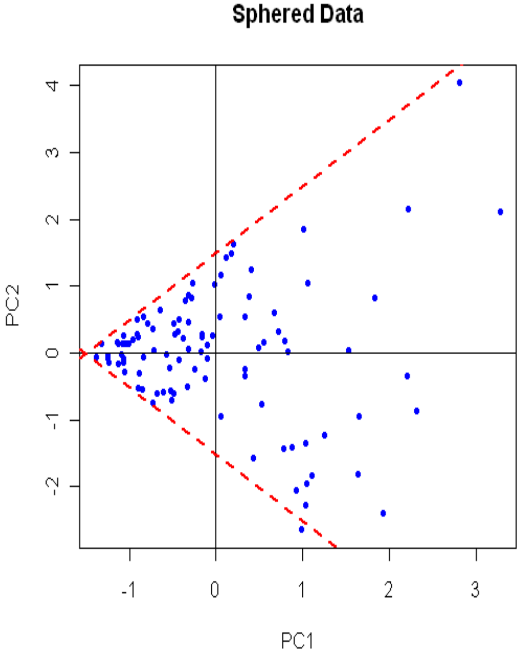


- PCA rotates axes to make first coordinate represent distance along long axis of observations cloud

# ICA Example I – Two Sources



Observed Mixture



Re-scale the axes so that all variances = 1

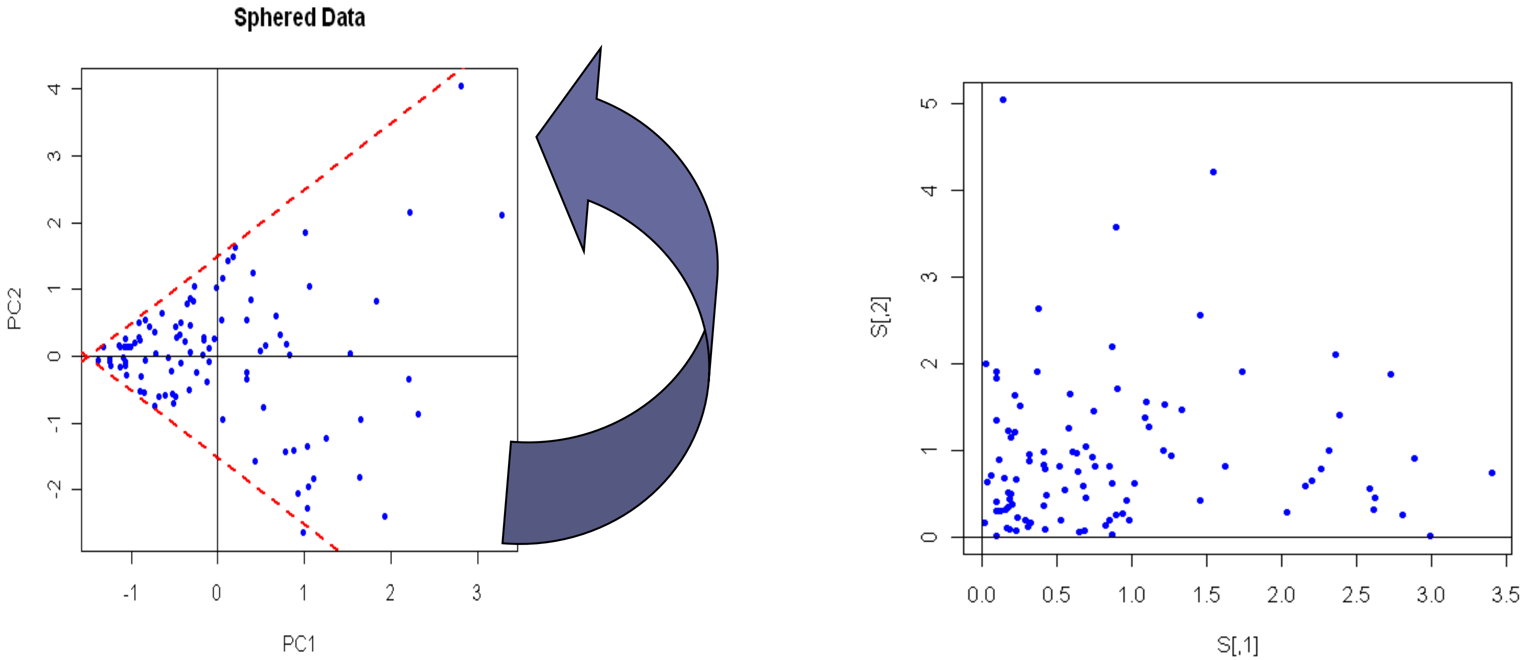
Now correlations are 0

(and would be for all rotations of these axes)

BUT…

- Variables are mutually informative

# ICA Example I – Two Sources



- Rotate the rescaled axes so that variables are the least mutually informative
- Equivalent to rotating so reconstructed coordinate variables are maximally non-Normal

# ICA Needs Non-Normal Sources

- If you try this with Normal data, all the information is in the covariance matrix, and you can’t improve independence
- If you have a mixture of some Normal sources and some highly skewed, the ICA procedure can’t separate the Gaussian sources from each other by variance (as PCA does)

# How to Order IC’s

ICA algorithms do not order the components because there is no natural order; the variance contributed by each component may be partially offset by others.

However a rough ordering may be obtained by summing the squares or absolute values of the columns

# ICA Compared in Practice to PCA

- PCA is well-defined and unique but factors may not be meaningful because real source combinations are not orthogonal
- ICA factors are often more meaningful but are usually not unique and sometimes results from different methods or even from the same method run several times, are not even close

# Validating ICA in Practice

To assess confidence in the imputed sources one typically runs the ICA process many times from various random starting points and looks for factors that crop up, with nearly identical scores, in all runs of the algorithm

Issue: What is a reasonable threshold for identity? R > .95?

# Advantages of ICA

- ICA can resolve many more components with roughly comparable contributions to variance than can PCA
- ICA can resolve components with non-orthogonal loadings, which is tricky (above two or three) for factor analysis
- ICA can find non-Gaussian systematic components with comparable variance to (Gaussian) noise components

# Disadvantages of ICA

- There is no statistical theory: i.e. we don’t know how accurate the estimates are
- The process of fitting is a search procedure from a random initial guess
  - Running ICA twice on the same data may give different decompositions
- There are several different algorithms, which often give very different results on same data

# Uses of ICA in Neuroscience

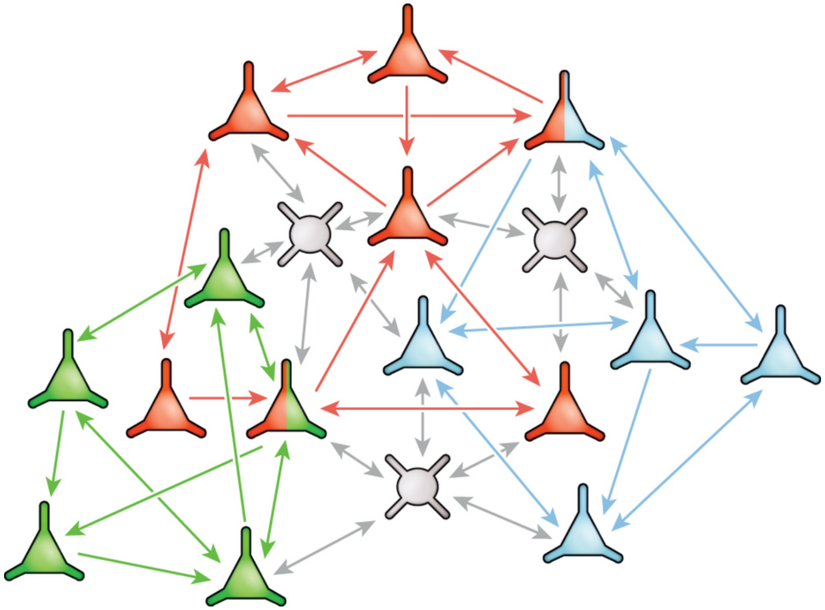
- Commonly used to identify artifacts in EEG/ MEG and fMRI data analysis
- Often used to identify specific networks of correlated regions in fMRI data analysis
  - This use is somewhat controversial
  - Often the distribution of IC scores nearly Gaussian
    - A sign that ICA can’t be working

# Non-negative Matrix Factorization

# Non-negative Matrix Factorization

- NNMF tries to find a decomposition of a positive matrix (e.g. spike counts) into a mixing matrix and a matrix of non-negative time series
- A variety of iterative approaches, like ICA
- No canonical algorithm like PCA

- Appropriate for model of intermingled clusters of E-cells (e.g. Harris & Mrsic-Flogel, Nature 2013)



# Basic Ideas of NMF

- If values are positive and underlying factors can only be positive or 0, then
- PCA will always, and FA will usually, give components with many negative loadings
- Can we identify a distinctive decomposition A=WH with non-negative W & H?
- No unique solution
- Two iterative approaches:
- Multiply and divide (slow)
- Alternating least squares (fast, but often fails)