Week 8

Cogs 109: Data Analysis and Modeling

Fall 2017
Prof. Eran Mukamel

Recap of nonlinear regression methods

- Polynomial regression
- Basis functions
- Regression splines
  - Use piecewise polynomials (linear, cubic, …)
  - Knots: constrain the function to be continuous
- Smoothing splines
  - Regularization enforces smoothness
  - Knots are located at the data points

Recap of the course so far

- Modeling concepts: Prediction/Inference, Bias/Variance, Complexity/Interpretability
- Linear regression
- Classification: Logistic regression, Linear discriminant analysis, K-Nearest neighbors
- Resampling methods: Cross-validation, Bootstrap
- Model selection: Stepwise feature selection; Ridge regression, LASSO regression, Principal Component regression
- Non-linear regression: Polynomial, Step function, Spline regressions; Smoothing splines

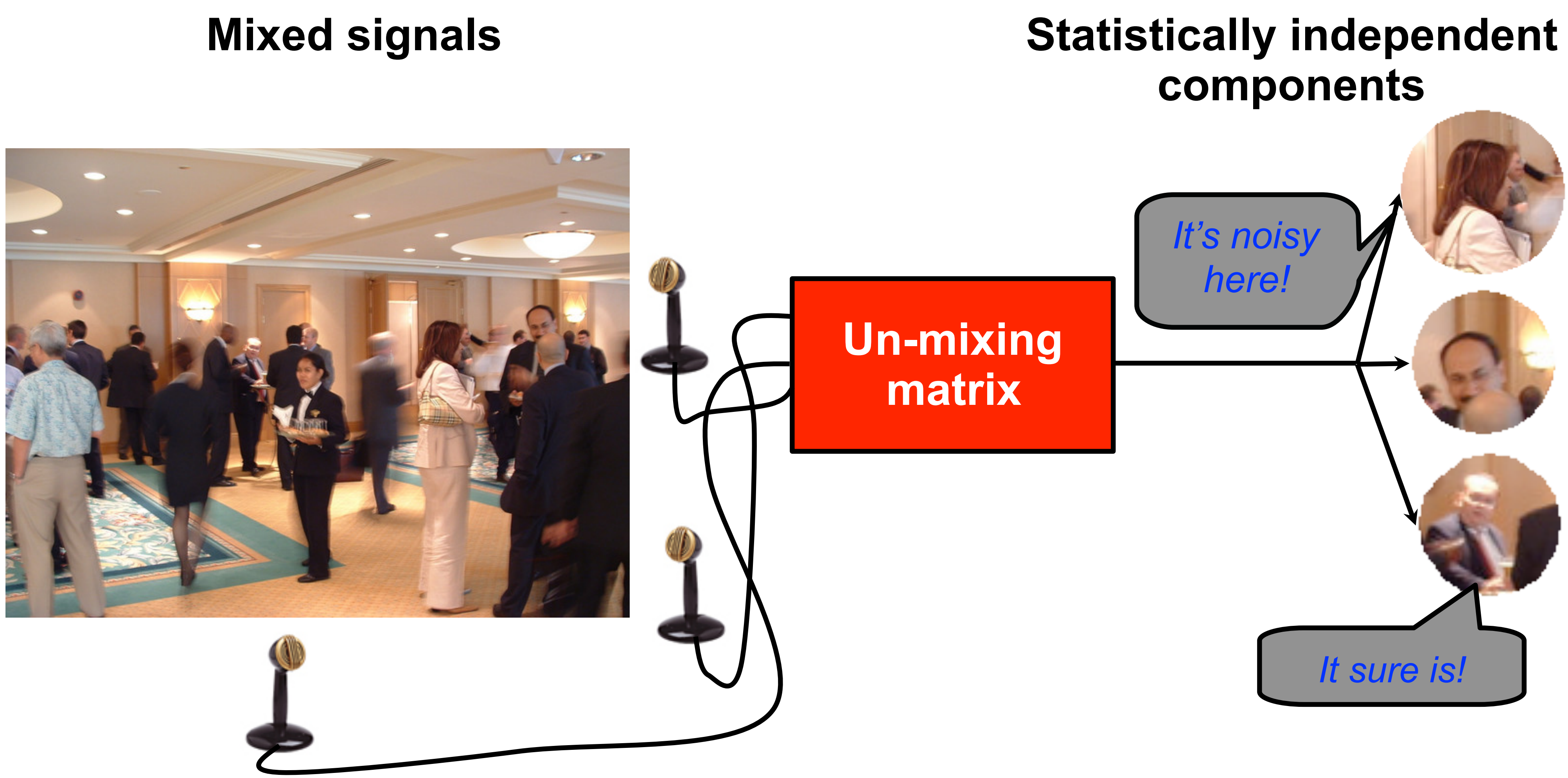
— — — — — —

- Unsupervised learning: Principal Component Analysis (PCA), Clustering
- Support Vector Machines

Unsupervised learning

- Trying to find structure without training labels
- There is often no “right” answer
- Unlike in supervised learning, there may be no objective and unique test to decide between competing models (like the test-set MSE)
- Unsupervised learning can be useful for *exploration* and *discovery*
- Unsupervised learning is like many cognitive problems faced by both machines and biological organisms: Finding useful structure in a complex world

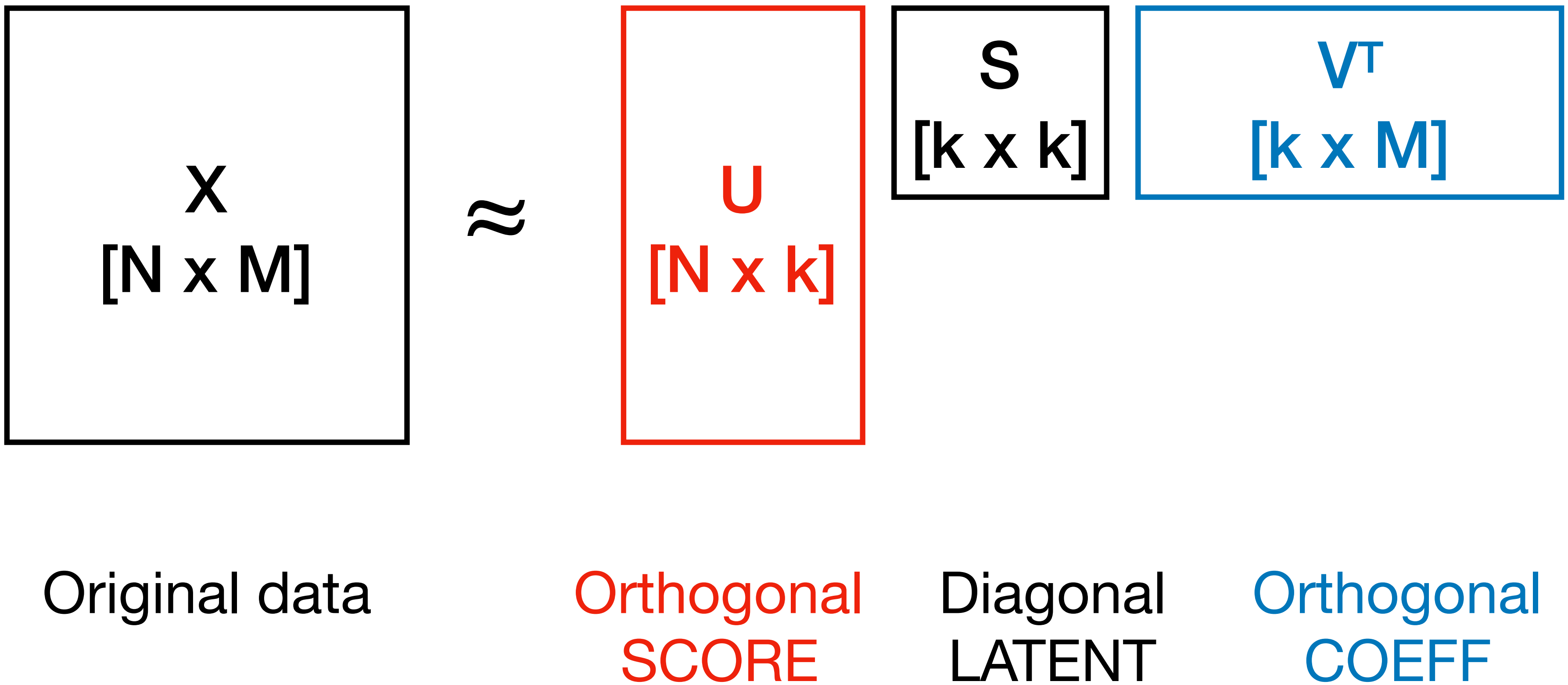
Example of unsupervised learning: The Cocktail Party Problem (a.k.a. Blind Source Separation)



Comon, P. (1994); A.J. Bell and T.J. Sejnowski (1995) *Neural Comput.*

Principal components analysis (PCA)

- We used PC regression to reduce data dimensionality for a regression (supervised learning)
- PCA is also useful in the unsupervised setting
- The PCs form a <u>compressed</u> representation of the data



- The PCA decomposition seeks the best approximation of the original data using a compressed set of PCs

Mean squared error for PCA

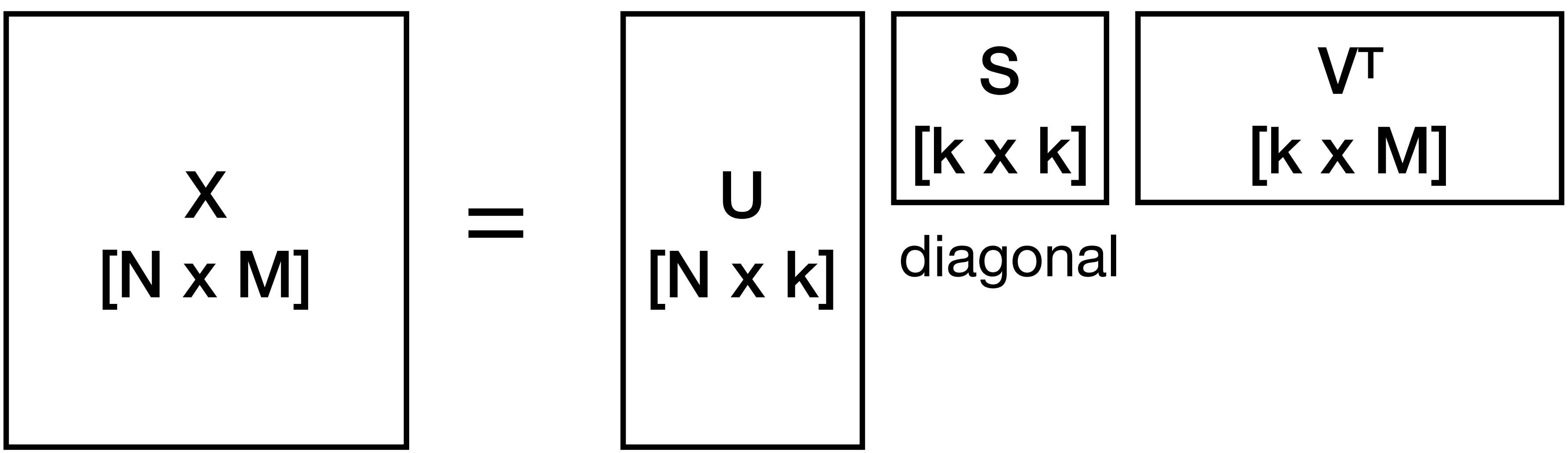
- Original data $X = USV^T$
- Approximate data after compression $\tilde{X} = U^{(k)}S^{(k)}[V^{(k)}]^T$
- Mean squared error $MSE = \sum(X - \tilde{X})^2$

Principal components analysis (PCA) as data set compression

- Original data matrix: X has (NM) data points

- Compressed: Using the top k PCs we have:
  $(Nk)+k+(Mk) = (N+M+1)k$

- Example: N=1000, M = 10,000, k = 10:
  $NM = (10^3 \times 10^4) = 10^7$
  $(N+M+1)k = (11{,}001)*10 = 110{,}010 \approx 10^5$



Principal components analysis (PCA)

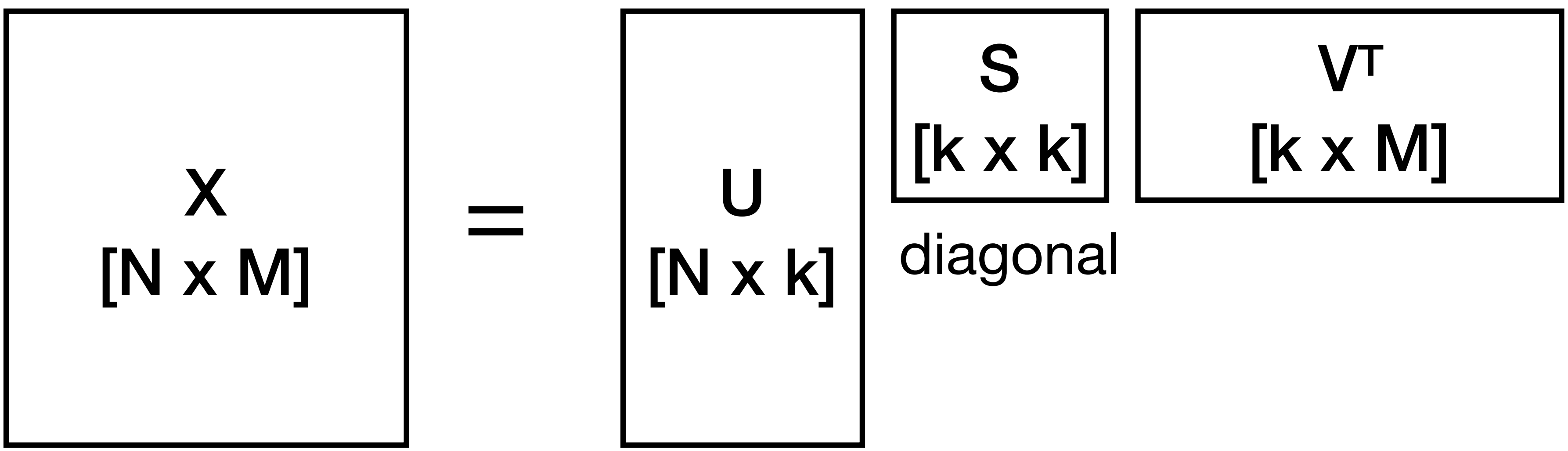
Data: X is an N x D matrix, where each column is one data dimension and each row is one observation.

X_{ij} Value of the data for observation i and feature j

Step 0: Remove the mean for each observation, $Y_{ij} = X_{ij} - \frac{1}{N}\sum_j X_{ij}$.

Step 1: Covariance matrix $C_{ik} = \sum_j Y_{ij} Y_{kj} = (YY^T)_{ij}$

Step 2: Eigenvalues, eigenvectors: $C = USU^T$, where U is an orthogonal matrix and S is diagonal



Example: Anesthesia data

X
[N=926 time points x M=103 frequencies]

≈

U
[926 x 20]

S
[20 x 20]

V^T
[20 x 103]

Original data

Orthogonal
SCORE

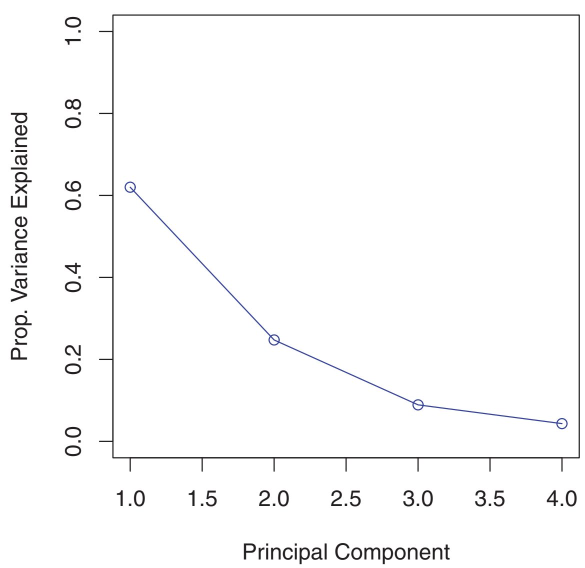
Diagonal
LATENT

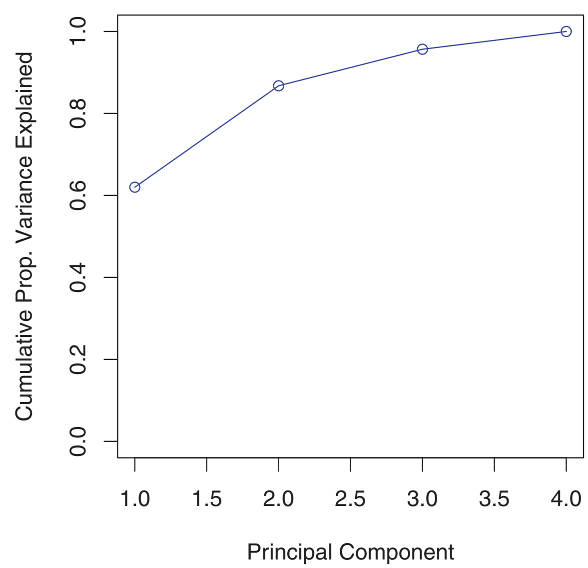
Orthogonal
COEFF

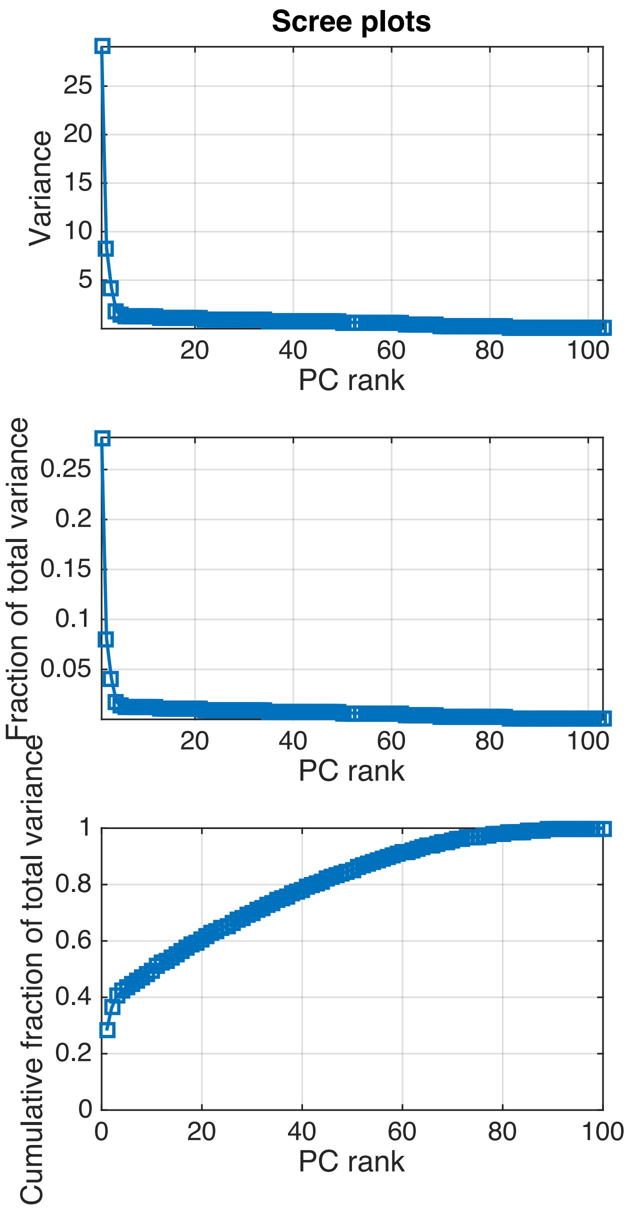
PCA plots

- We can visualize the PCA results in 3 ways:
    - Scree plot showing the variance for all PCs (S)
    - Scatter plot of the coefficients, e.g. PC1 vs. PC2
    - Scatter plot of the scores, e.g. PC1 vs. PC2

PCA plots 1: Scree plot







PCA plots 2

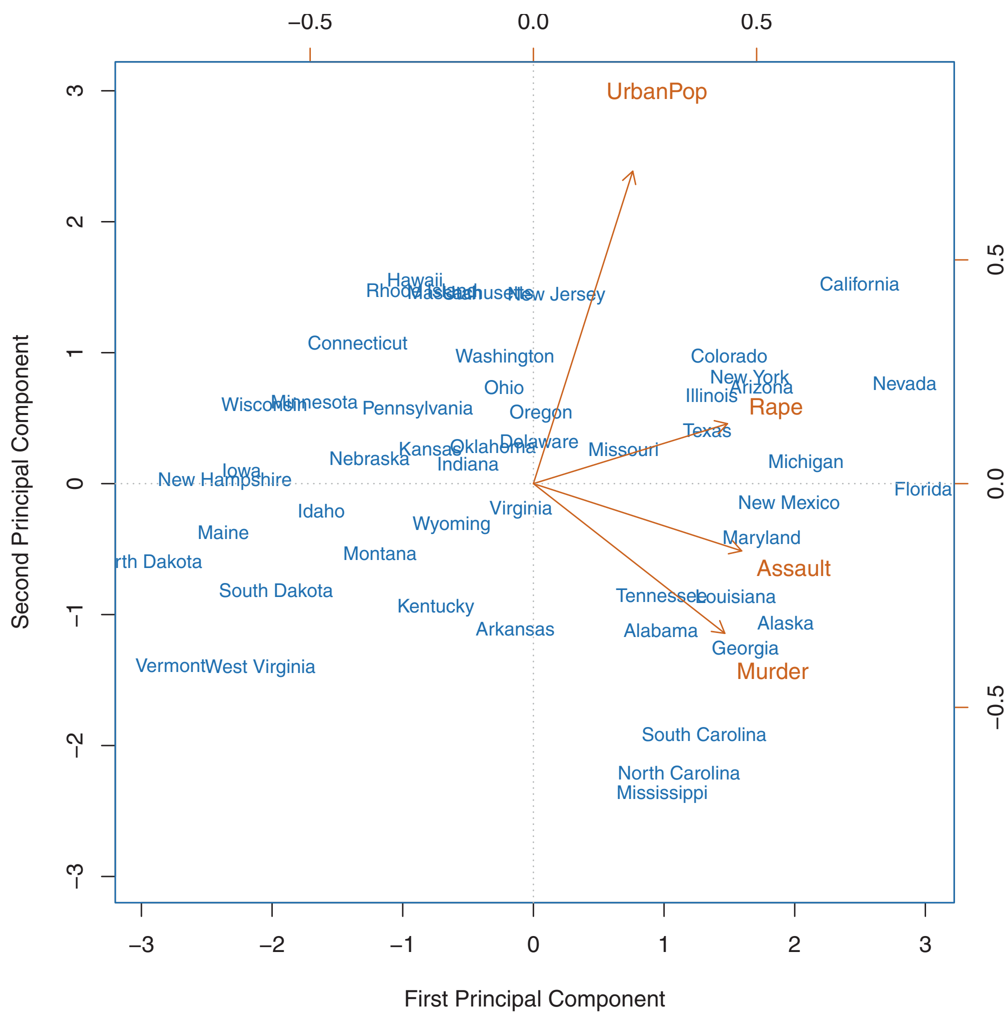
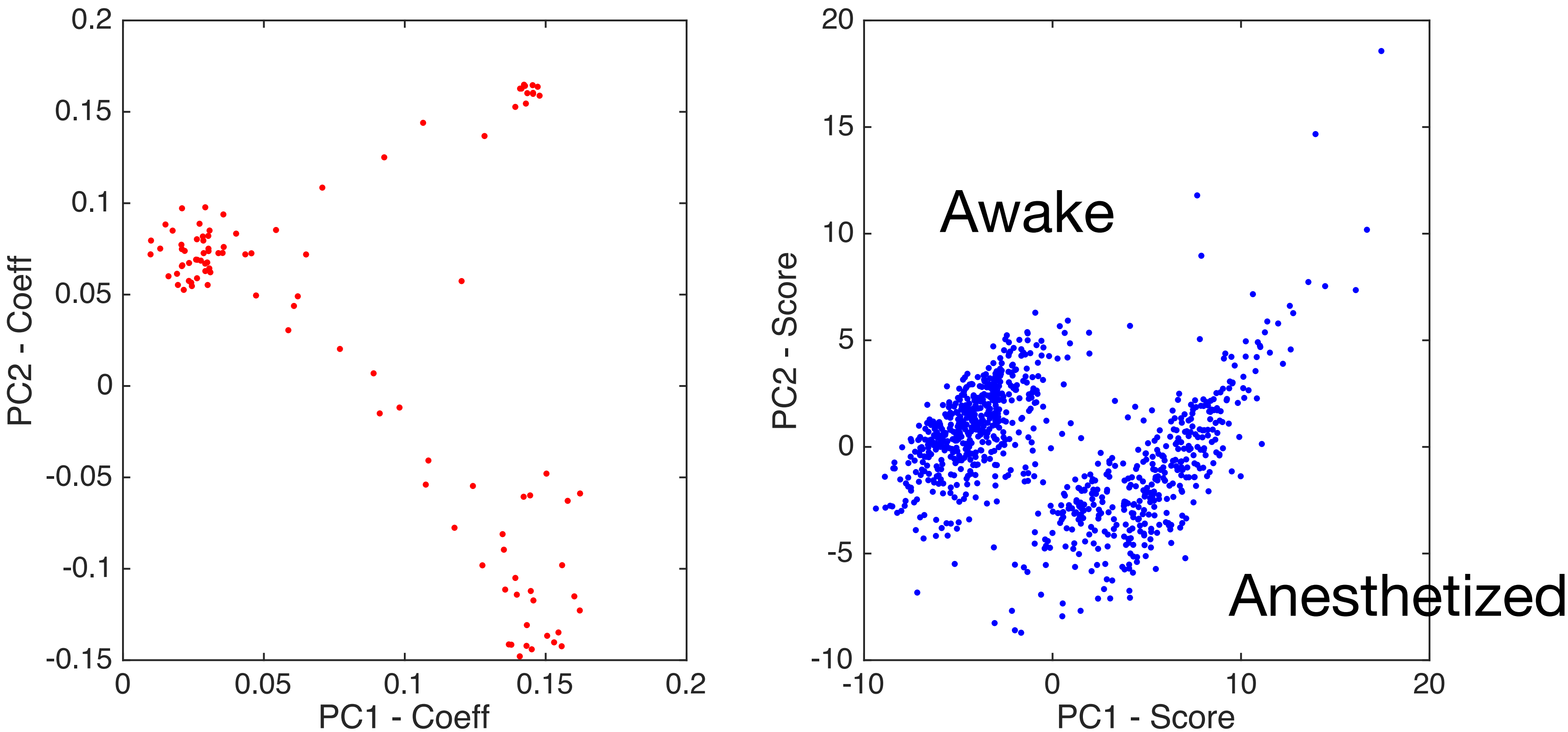
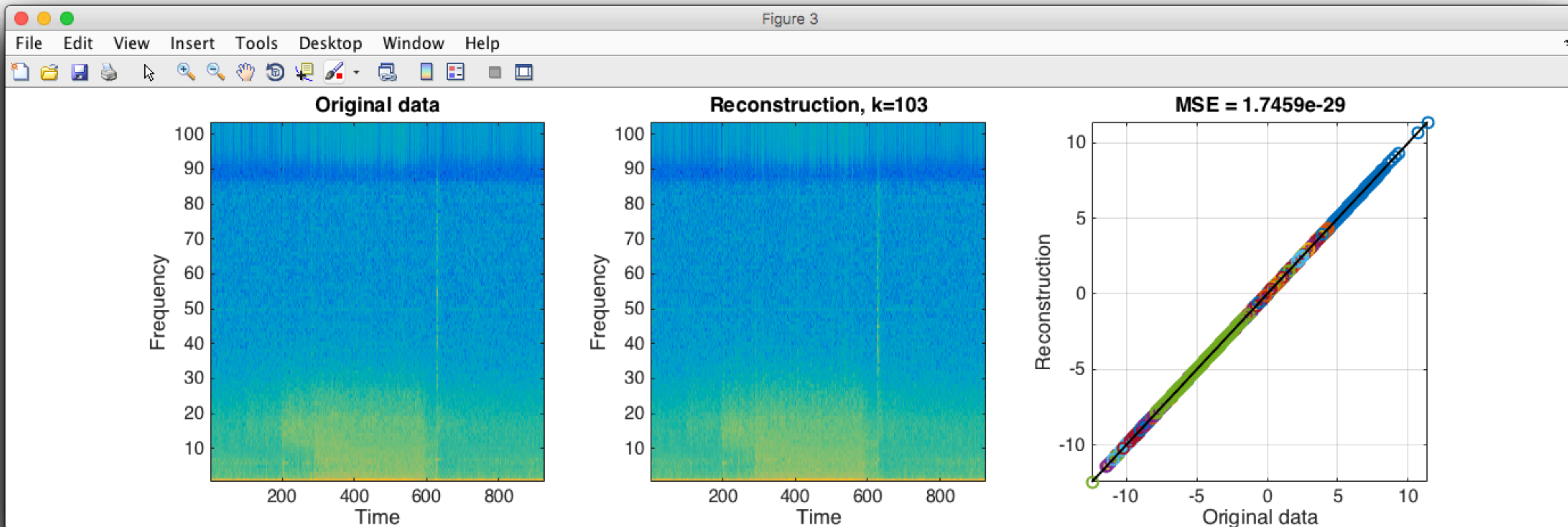


FIGURE 10.1. *The first two principal components for the* `USArrests` *data. The blue state names represent the scores for the first two principal components. The orange arrows indicate the first two principal component loading vectors (with axes on the top and right). For example, the loading for* `Rape` *on the first component is* 0.54*, and its loading on the second principal component* 0.17 *(the word* `Rape` *is centered at the point* $(0.54, 0.17)$*). This figure is known as a biplot, because it displays both the principal component scores and the principal component loadings.*

PCA plots 2


0.2
0.15
0.1
0.05
0
-0.05
-0.1
-0.15
0
0.05
0.1
0.15
0.2
PC2 - Coeff
PC1 - Coeff
20
15
10
5
0
-5
-10
-10
0
10
20
PC2 - Score
PC1 - Score
Awake
Anesthetized

Reconstructing the data from the PCs

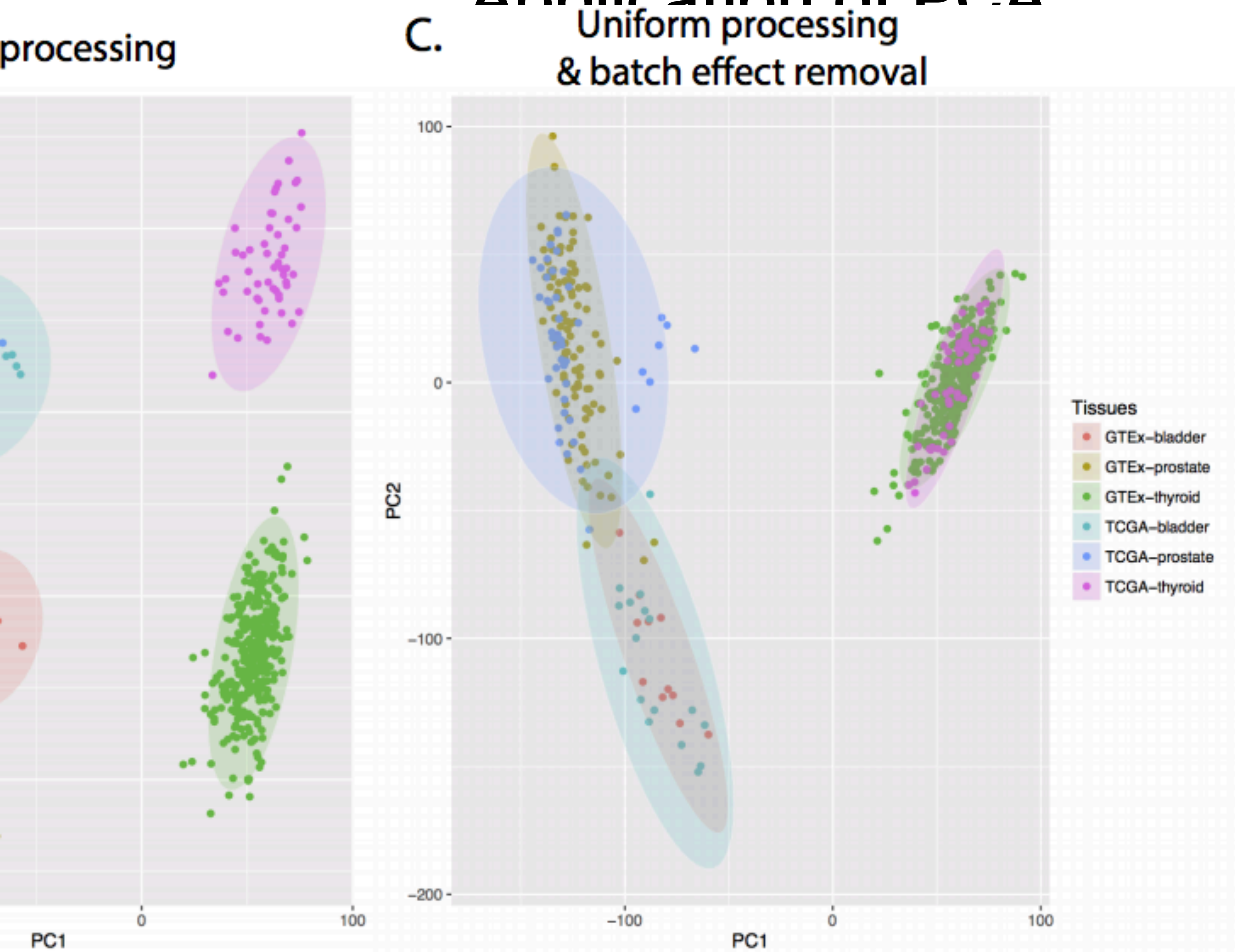


Application of PCA:

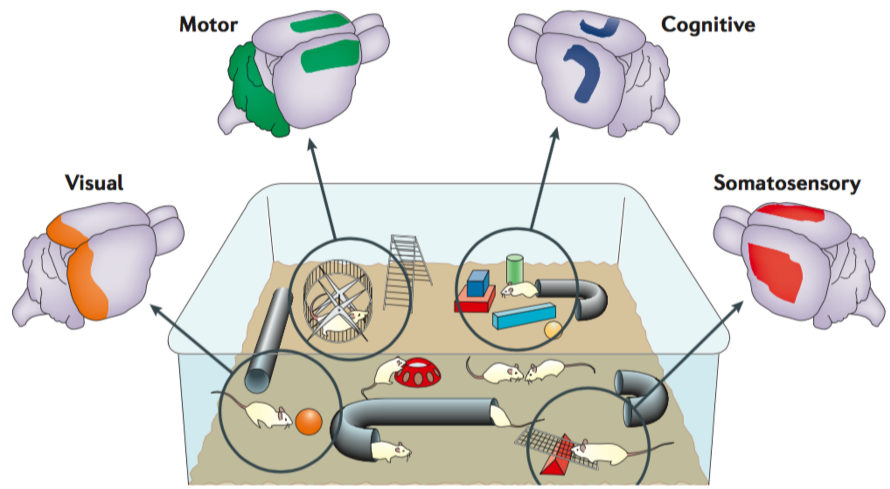
data

s in

ferences

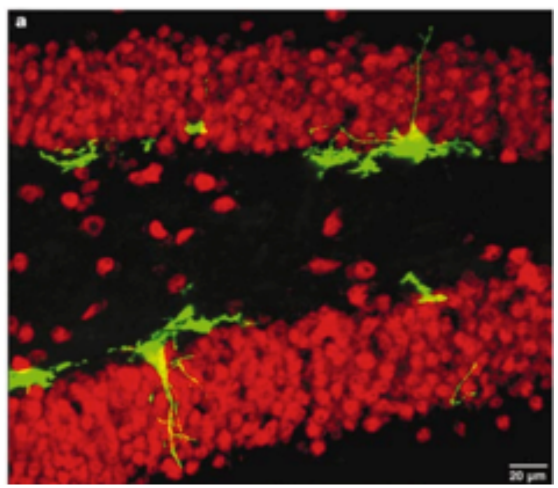


Environmental enrichment promotes adult neurogenesis, learning and memory



(Nithianantharajah 2006)

Adult newborn neurons in dentate gyrus (DG) of the hippocampus

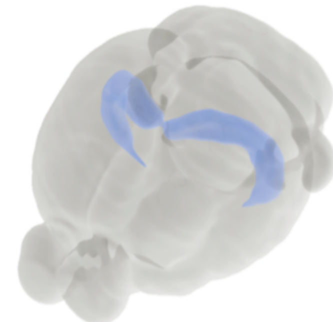


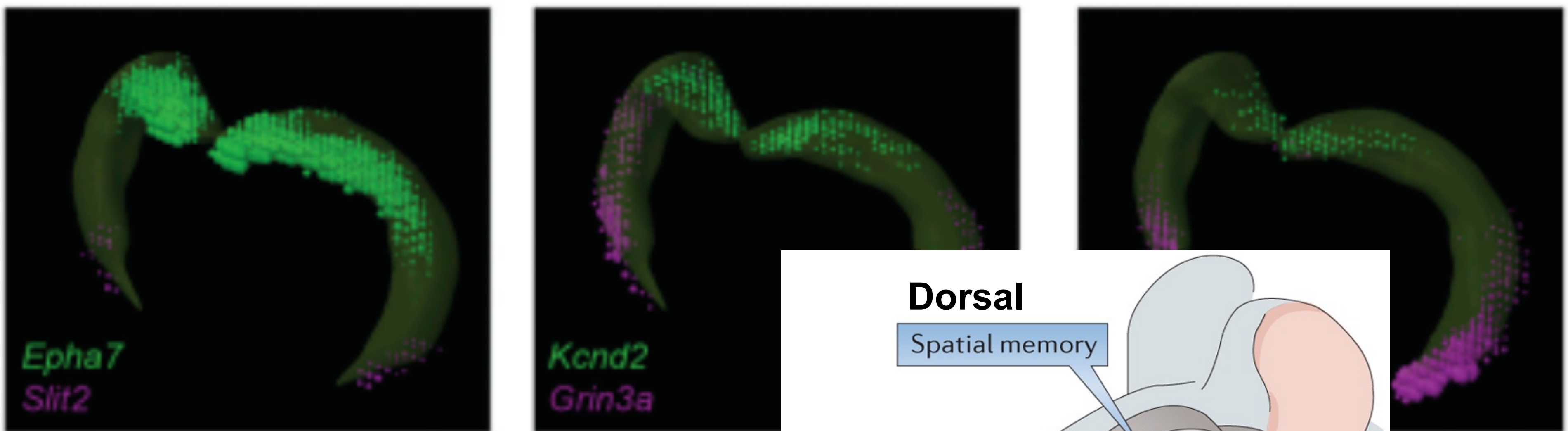
(Eriksson et al., *Nat. Med.*1998)

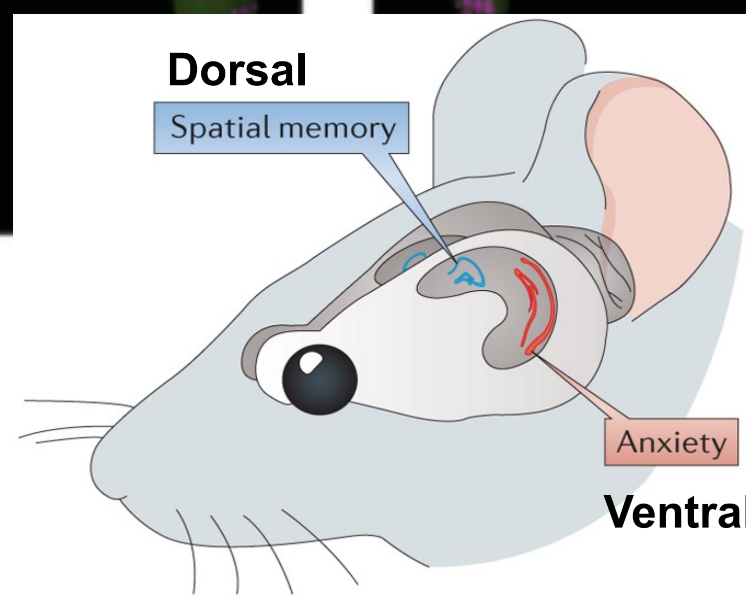
Chris Keown (UCSD)

with: Tie Yuan Zhang, Michael Meaney (McGill)

Dorsal – Ventral gradients in “uniform” hippocampal neuron populations

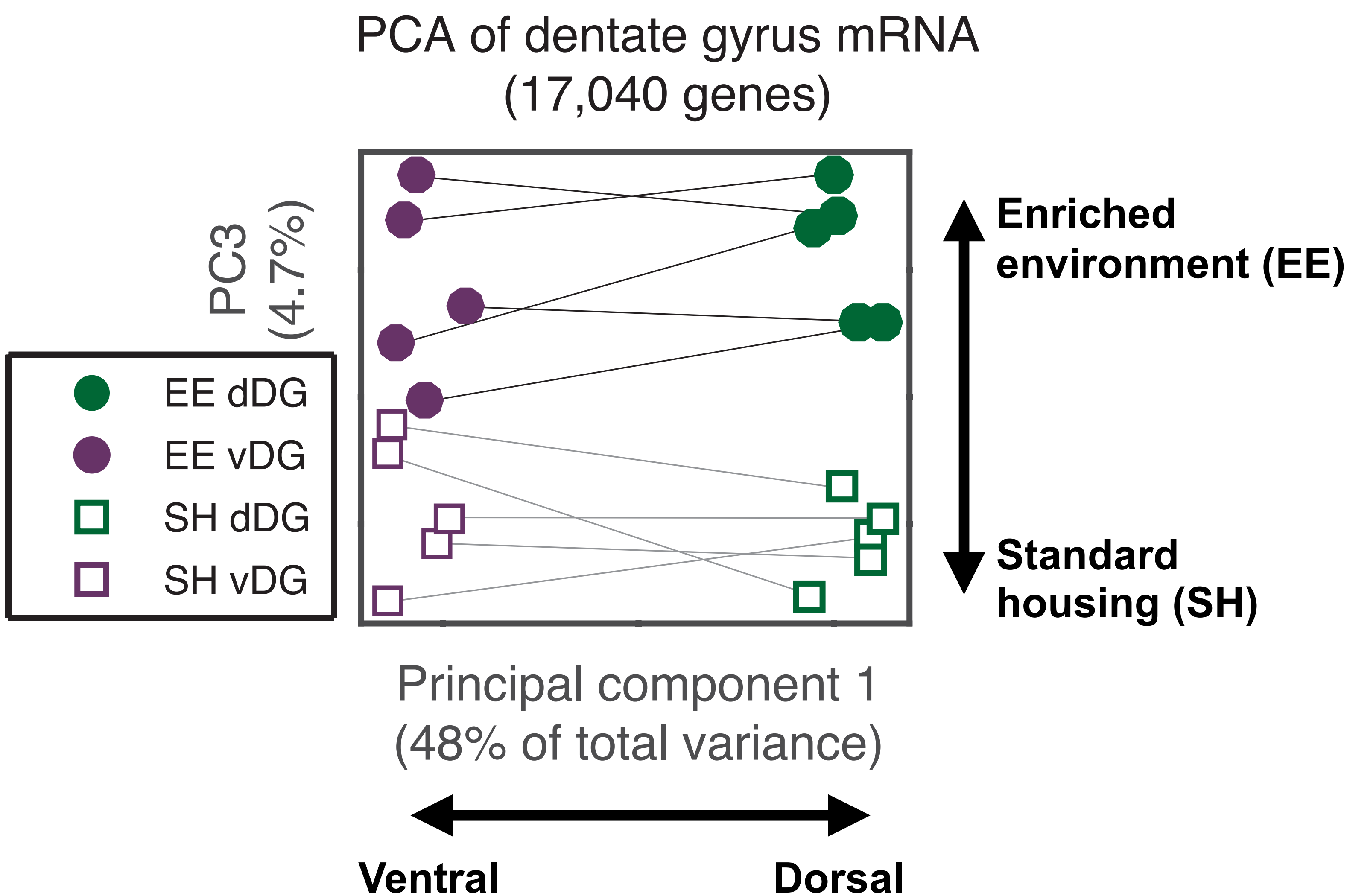






Cembrowski *et al.*, *eLife* and *Neuron* 2016

PCA of RNA-Sequencing data shows effects of brain region (dorsal-ventral) and of environment



Example of PCA: Eigenfaces

Original data: 16 face images

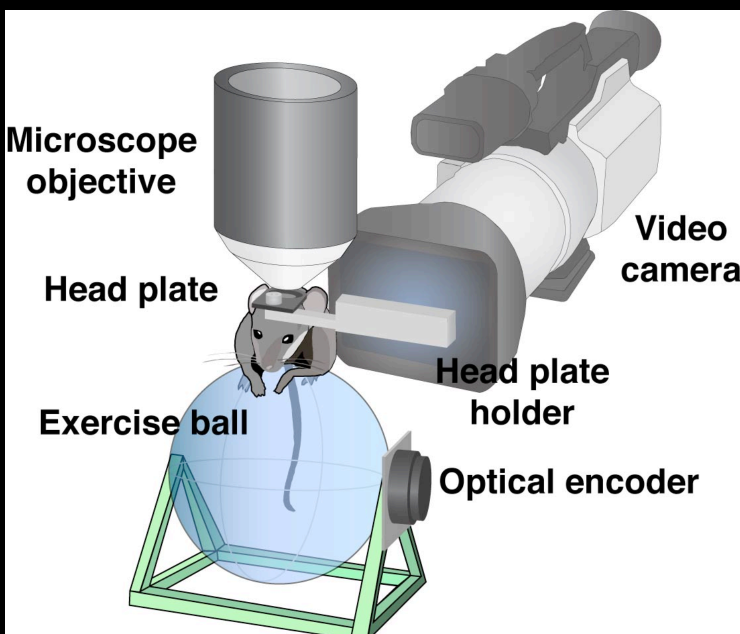
- Mean

- Eigenfaces are PC scores

- http://laid.delanover.com/explanation-face-recognition-using-eigenfaces/

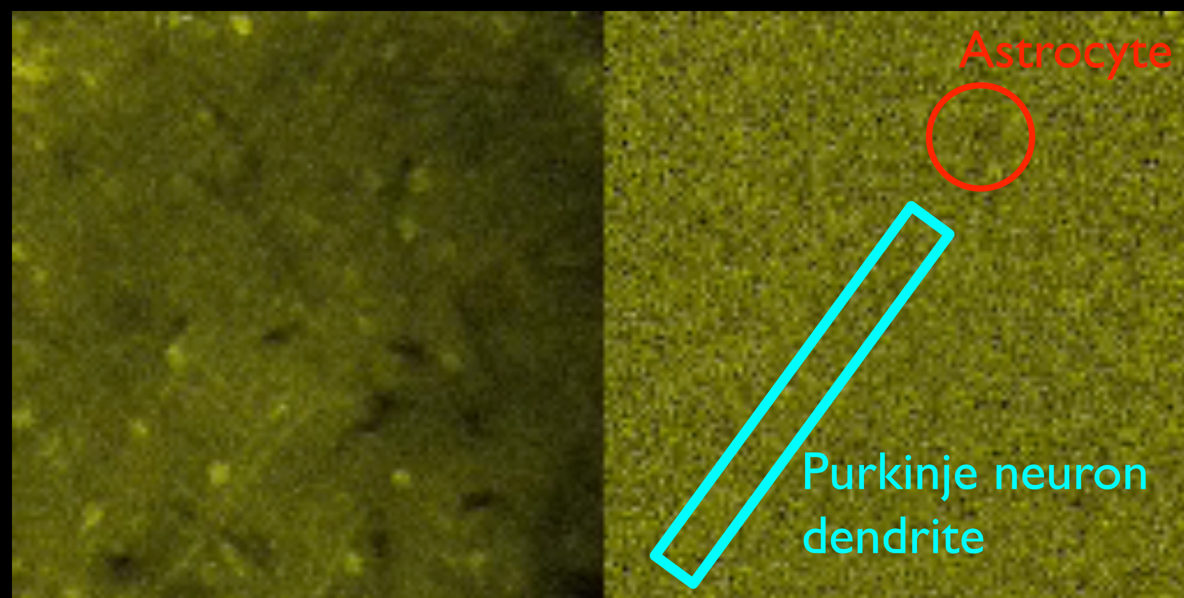
Watching (many) neurons in action:
Optical imaging in awake mice during locomotion

Head-fixed imaging in behaving mouse



Raw fluorescence

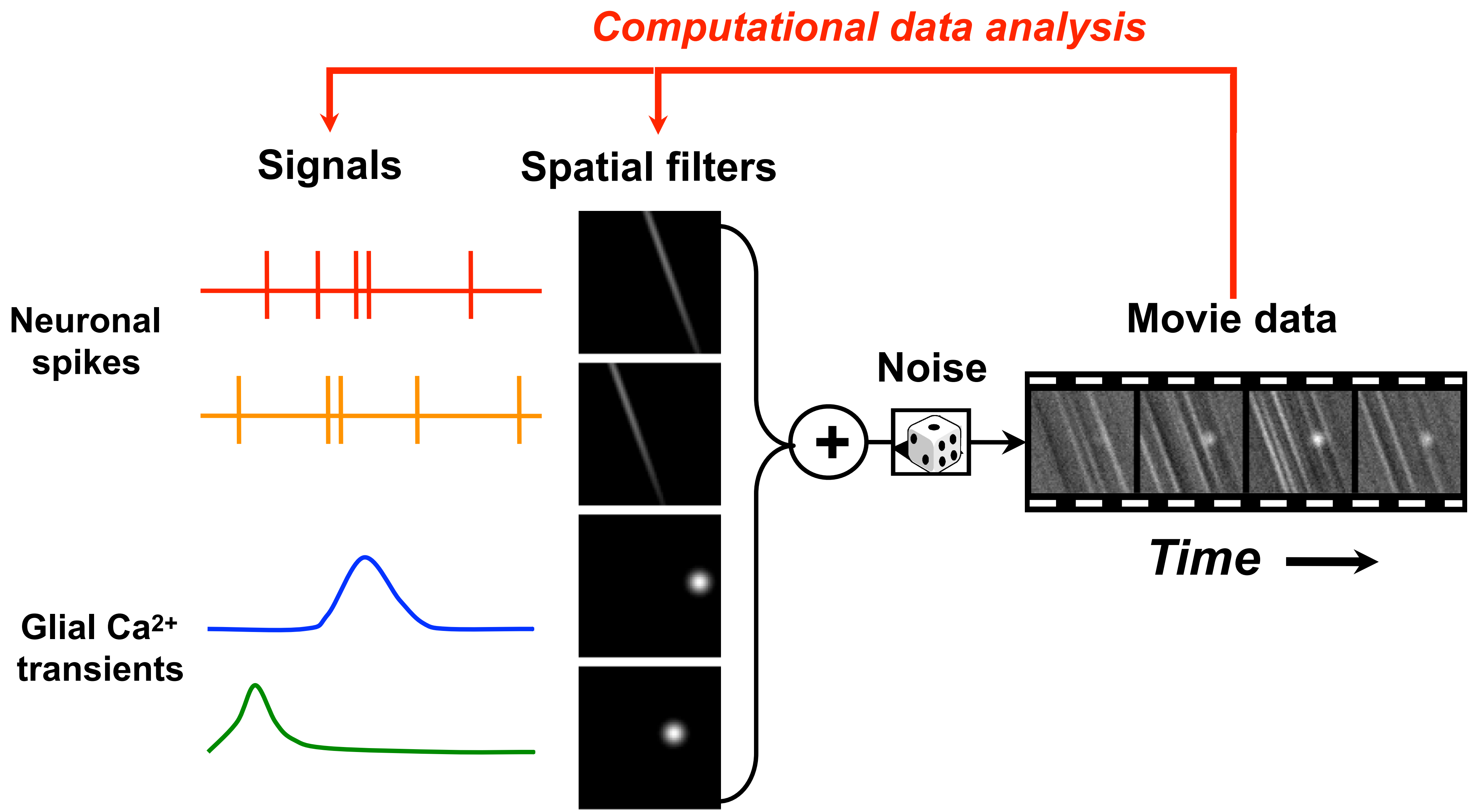
Normalized calcium signals ($\Delta F/F$)



50 μm

Mukamel, Nimmerjahn and Schnitzer, *Neuron* (2009)

A statistical "mixed-sources" model of optical neuroimaging data



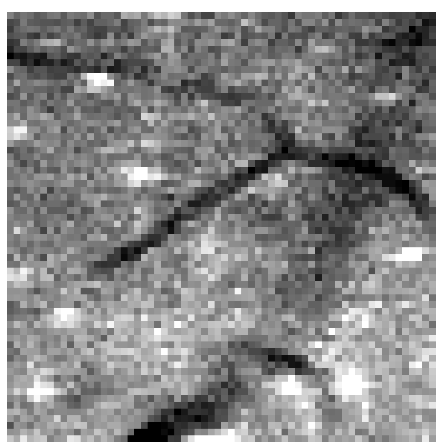
Using PCA to remove noise from a movie data set

```
>> whos movie

  Name         Size                    Bytes   Class      Attributes

 movie       64x64x200              6553600   double
```

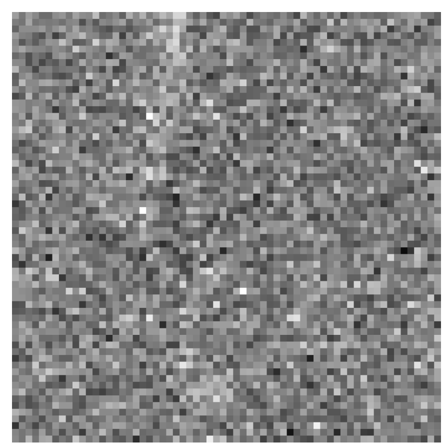
- Use "reshape" to convert the data from a 3D array (X x Y x T) to a 2D array (XY x T)

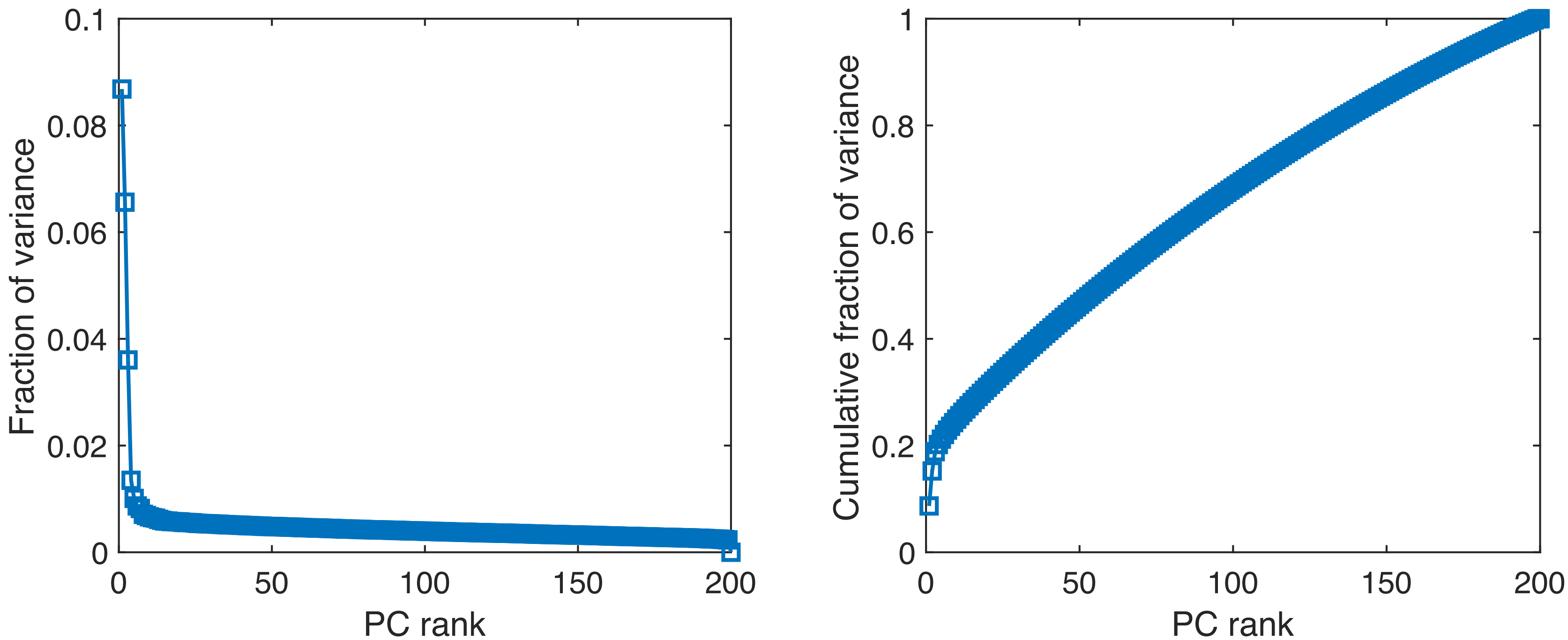
```
moviedata = reshape(movie_centered,4096,200);
[coeff,score,latent,~,~,mu] = pca(moviedata,'numcomponents',50);
```

Original movie



Mean subtracted, t = 7

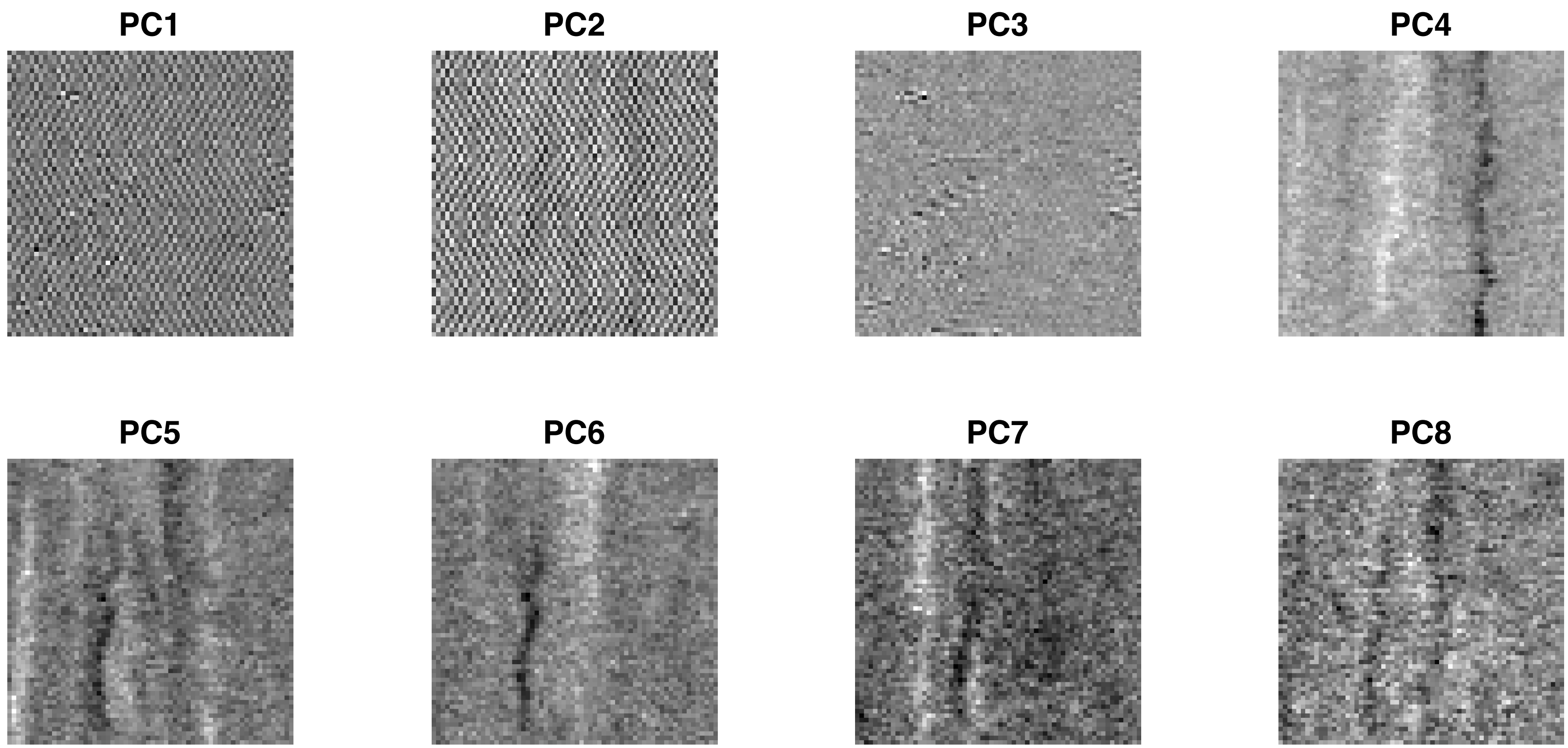




0.1
0.08
0.06
0.04
0.02
0
Fraction of variance
0
50
100
150
200
PC rank
1
0.8
0.6
0.4
0.2
0
Cumulative fraction of variance
0
50
100
150
200
PC rank

- Plot the PC scores — the value of each component at each pixel

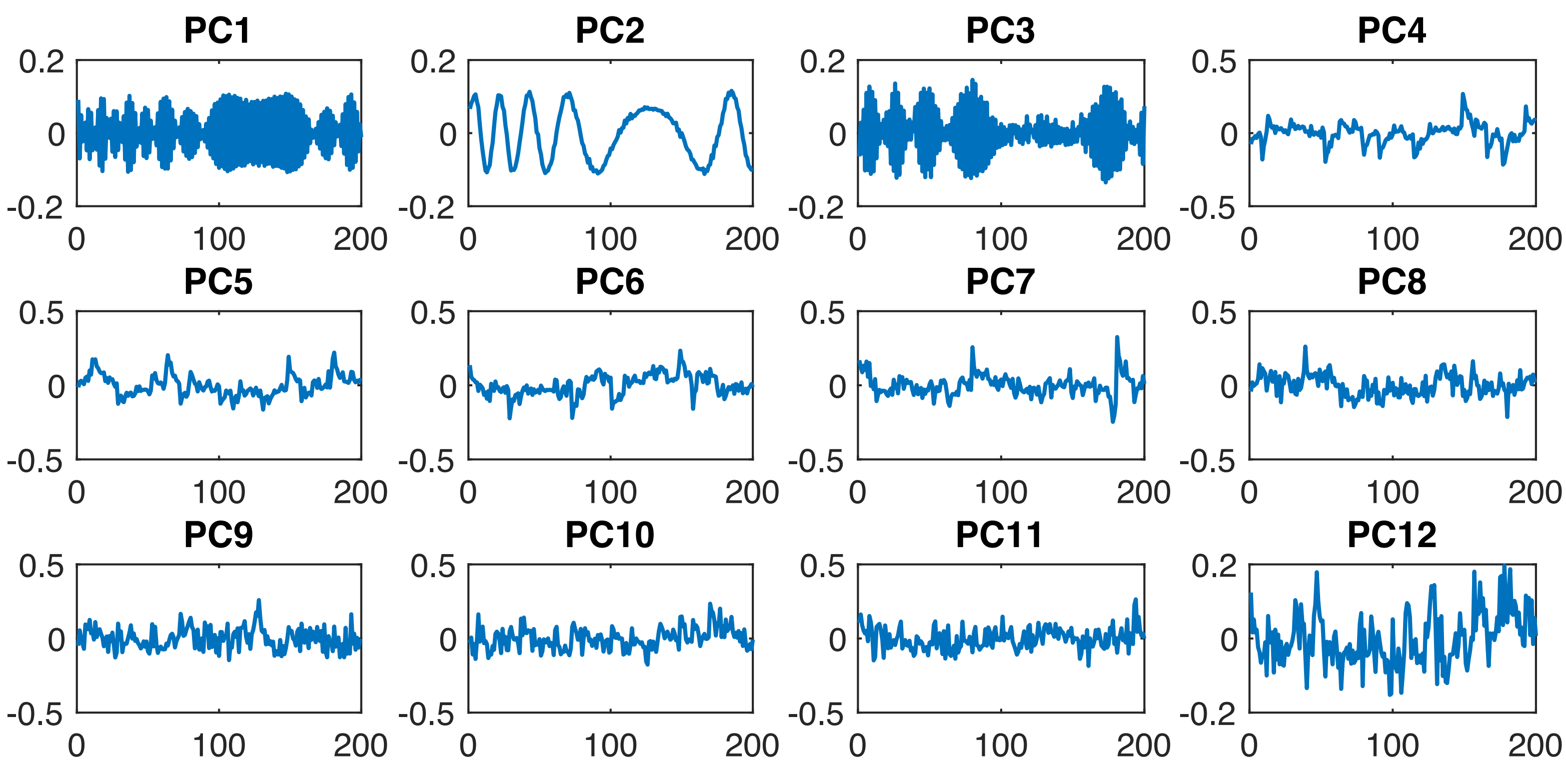
```
%% PC scores

figure(3);
clf
colormap(gray)
for k=1:12
    subplot(3,4,k)
    imagesc(reshape(score(:,k),64,64))
    axis equal off
    title(['PC',num2str(k)])
end
```



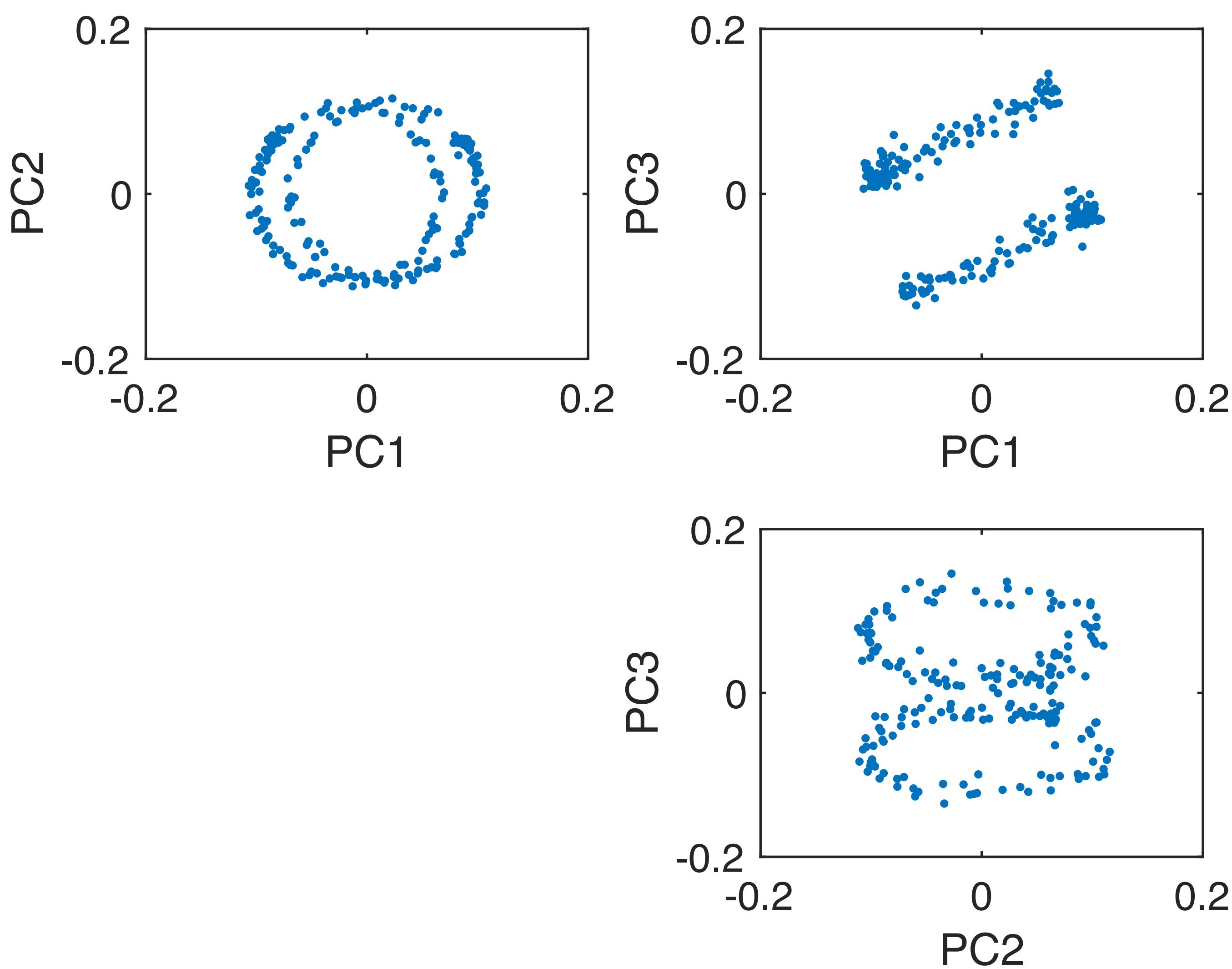
- Plot the PC coefficients — the value of each component at each time point

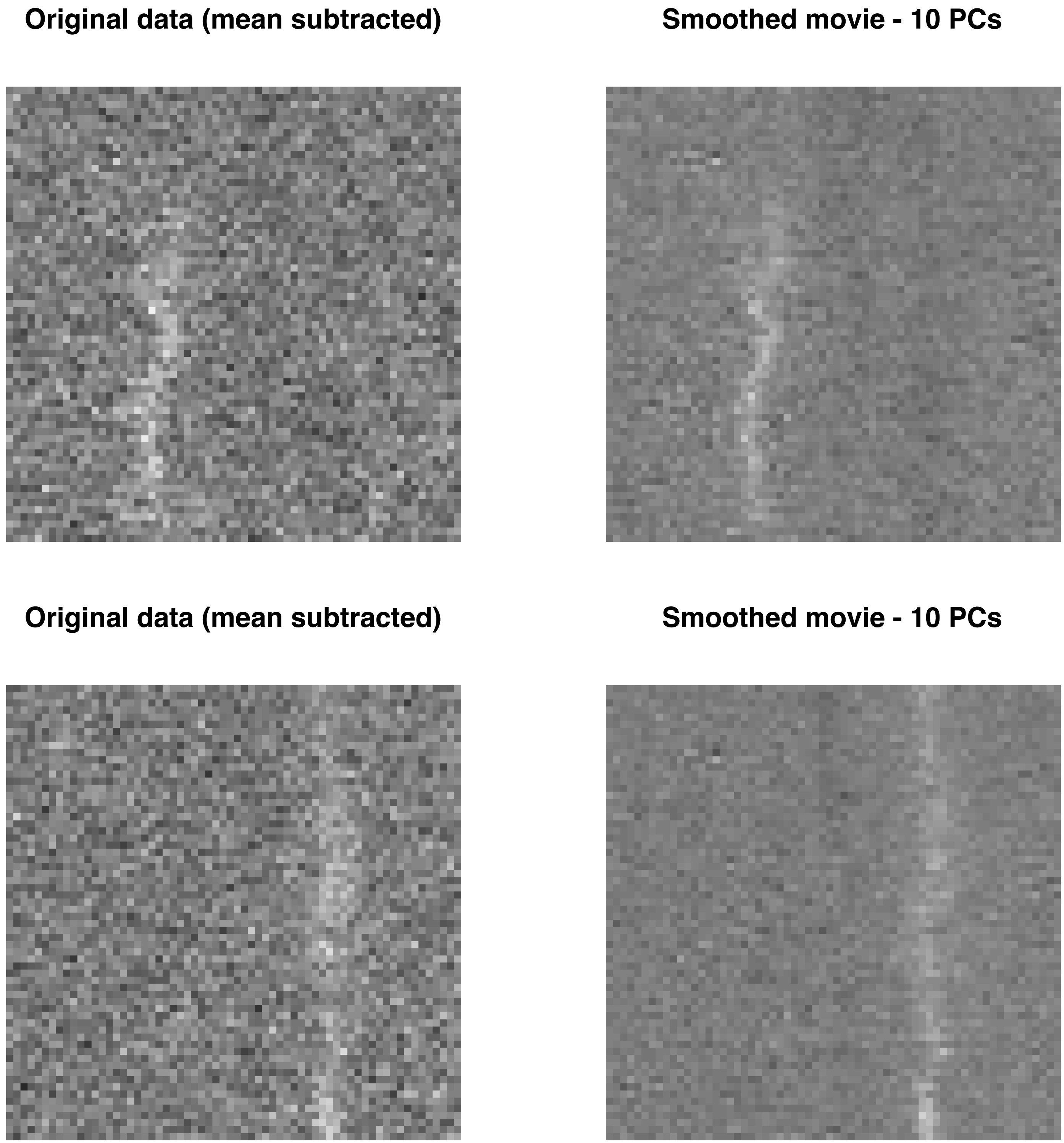
```
%% PC coefficients

figure(4); clf
for k=1:20
    subplot(5,4,k)
    plot(coeff(:,k))
    title(['PC',num2str(k)])
end
```

- PC1 and PC2 (and even PC3) appear to contain artifacts



- PC1 and PC2 (and even PC3) appear to contain artifacts




Original data (mean subtracted)
Smoothed movie - 10 PCs
Original data (mean subtracted)
Smoothed movie - 10 PCs