ISyE 6416: Computational Statistics
Spring 2017

Lecture 11: Principal Component Analysis (PCA)

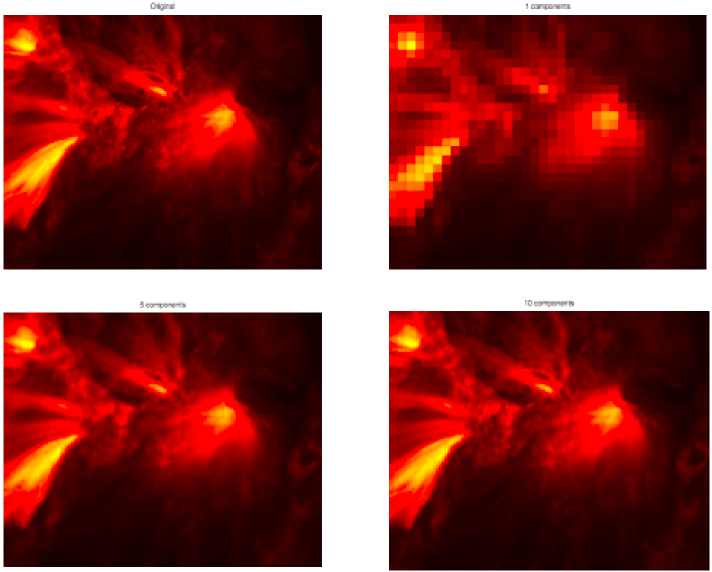
Prof. Yao Xie

H. Milton Stewart School of Industrial and Systems Engineering
Georgia Institute of Technology

Why PCA

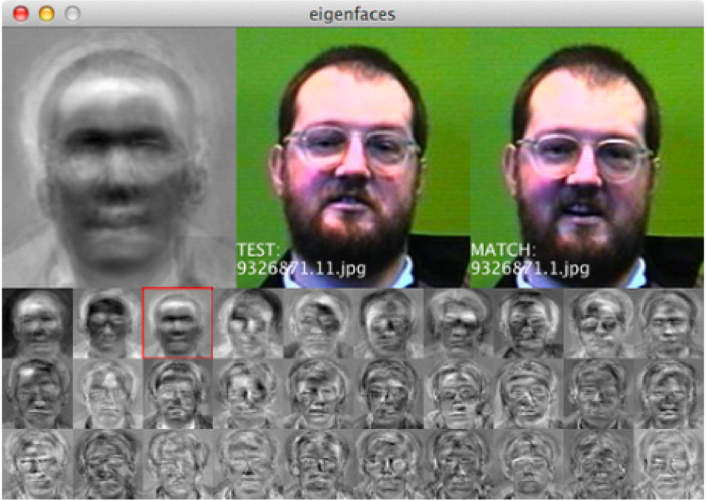
- Real-world data sets usually exhibit structures among their variables
- Principal component analysis (PCA) rotates the original data to new coordinates
  - Dimension reduction
  - Classification
  - Denoising

Data compression

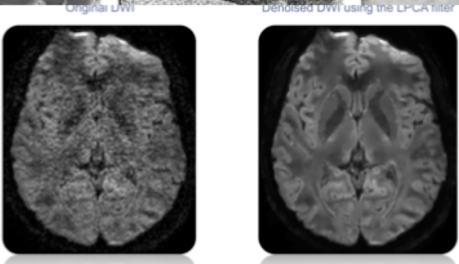


Face recognition

Eigenfaces



Denoising



Data visualization

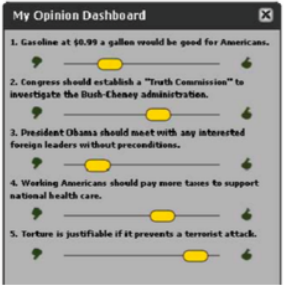


Figure 9. Region labeling of the topics of responses in CCA space.

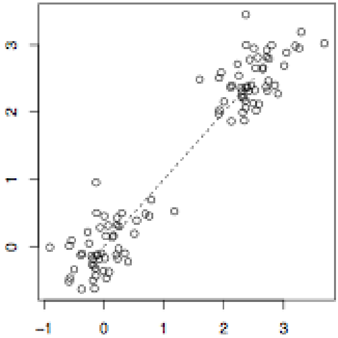
"The system collects opinions on statements as scalar values on a continuous scale and applies dimensionality reduction to project the data onto a two-dimensional plane for visualization and navigation." Using Canonical Correla.on Analysis (CCA) for Opinion Visualiza.on, Faridani et.al, UC Berkeley, 2010.

What is PCA

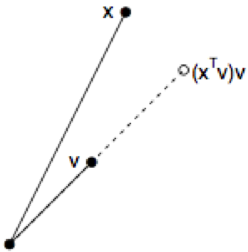
- Given a data matrix $X \in \mathbb{R}^{n \times p}$: n samples, and p variables
- Transform data set to one with a few number of *principal components*

$$v_j^T x_i, \quad j = 1, 2, \ldots, K, i = 1, 2, \ldots, n.$$

- It is a form of *linear dimension reduction*
- Specifically, "interesting directions" means "high variance"



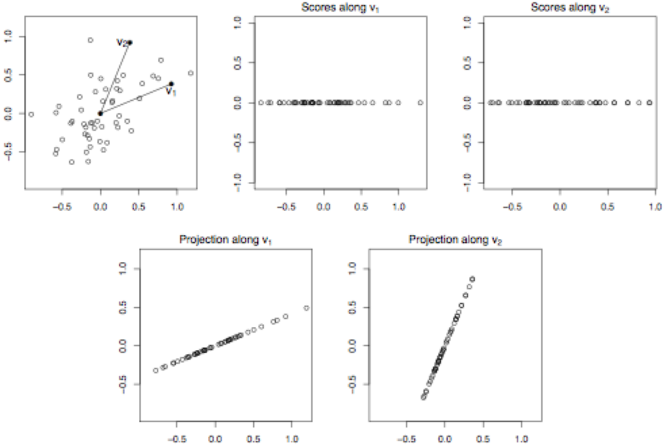
Projection onto unit vectors



$(x^T v) \in \mathbb{R}$: **score**
$(x^T v)v \in \mathbb{R}^p$: **projection**

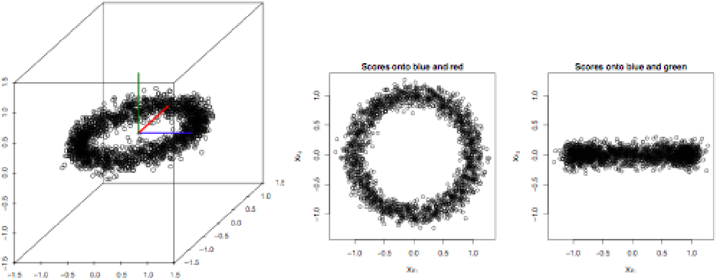
Example: projection onto unit vectors

Example: $X \in \mathbb{R}^{50\times 2}$, $v_1, v_2 \in \mathbb{R}^2$



Projections onto orthonormal vectors

Example: $X \in \mathbb{R}^{2000\times 3}$, and $v_1, v_2, v_3 \in \mathbb{R}^3$ are the unit vectors parallel to the coordinate axes



Not all linear projections are equal! What makes a good one?

Source: R. Tibshrani.

First principal component

- The first principal component direction of X is the unit vector $v_1 \in \mathbb{R}^p$ that maximizes the sample variance of $Xv_1 \in \mathbb{R}^n$ among all unit length vector

  $$v_1 = \arg\max_{\|v\|_2=1} (Xv)^T(Xv)$$

  Note that

  $$Xv = \begin{bmatrix} x_1^T v \\ \vdots \\ x_n^T v \end{bmatrix}$$

  are projection of each of the sample on a unit length vector
- Xv_1 is the *first principal component score*

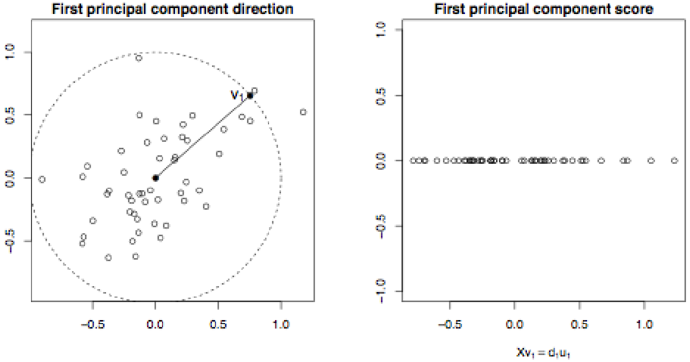
Eigenvector and eigenvalue

$$v_1 = \arg\max_{\|v\|_2=1} (Xv)^T(Xv)$$

- v_1 corresponds to the largest eigenvector of X^TX: sample covariance matrix
- $v_1^T X^T X v_1$, the largest eigenvalue of X^TX is the *variance explained by* v_1
- **Rayleigh quotient** $R(v) = \frac{v^TAv}{v^Tv}$

$$\lambda_{\min} \leq \frac{v^TAv}{v^Tv} \leq \lambda_{\max}$$

Example: $X \in \mathbb{R}^{50\times 2}$



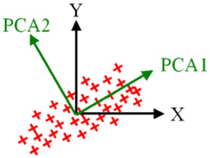
Beyond first principal component

- What's next? The idea is the find *orthogonal* directions of the remaining highest variance
- Orthogonal: since we have already explained the variance along v_1, we need new direction that has no "overlap" with v_1 to avoid redundancy.

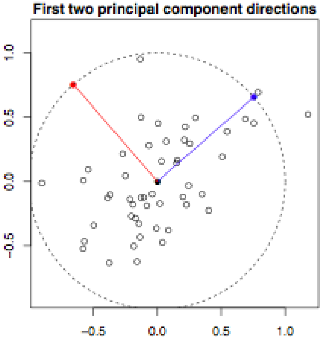
$$v_2 = \arg \max_{\|v\|_2=1, v^T v_1 = 0} (Xv)^T (Xv)$$

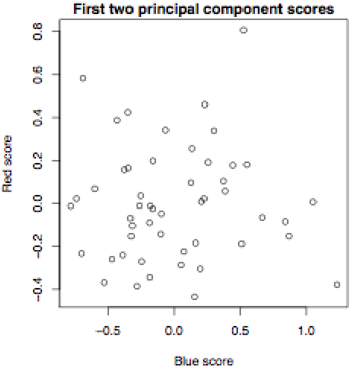
- Can repeat this process to find the kth principal component direction

$$v_k = \arg \max_{\|v\|_2=1, v^T v_j = 0, j=1,\ldots,k-1} (Xv)^T (Xv)$$



Example: $X \in \mathbb{R}^{50\times 2}$





Properties

- There are at most p principal components
- $[v_1, v_2, \ldots, v_p]$ can be found from eigendecomposition
  Let $\Sigma = X^T X$
  $$\Sigma = U \Lambda U^T$$
  where $\Lambda = \mathrm{diag}\{\lambda_1, \ldots, \lambda_p\}$, U is orthogonal matrix
- Can be computed efficiently if you only need the first principle component: power's method
  $$v^{(k)} := \Sigma v^{(k-1)} / \|\Sigma v^{(k-1)}\|$$
- In general can be computed using Jacobi's method
- Sparse Σ

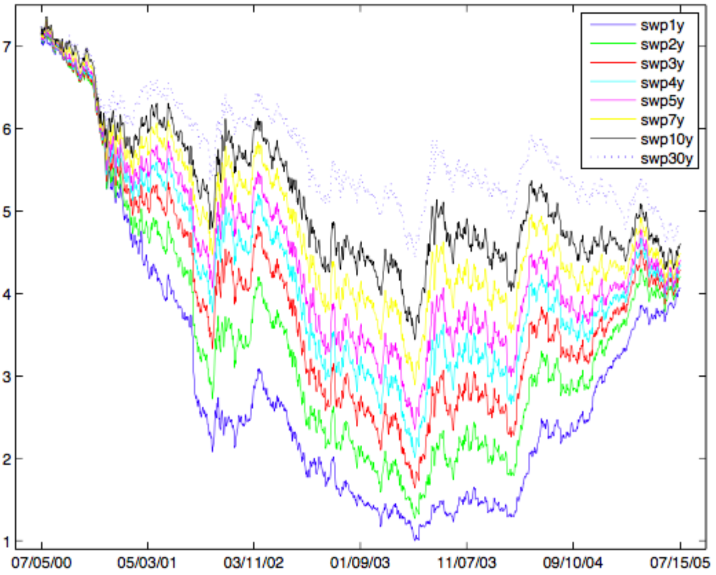
PCA example: financial data analysis

- Daily swap rates of eight maturities from 7/3/2000 to 7/15/2005

```
%%% (Data source: Economic, LLC)
%swp1y swp2y swp3y sw4y sw5y sw7y sw10y sw30y
7.1 7.16 7.17 7.17 7.17 7.2 7.24 7.24
7.03 7.06 7.07 7.07 7.08 7.11 7.14 7.16
7.07 7.13 7.14 7.15 7.16 7.19 7.21 7.21
7.01 7.04 7.06 7.06 7.07 7.1 7.14 7.14
7.04 7.09 7.11 7.13 7.14 7.17 7.2 7.19
7.04 7.1 7.11 7.13 7.14 7.18 7.22 7.2
7.06 7.12 7.14 7.15 7.17 7.2 7.23 7.19
7.04 7.09 7.1 7.12 7.13 7.16 7.19 7.13
7.08 7.14 7.16 7.17 7.19 7.21 7.23 7.17
7.12 7.21 7.23 7.25 7.28 7.31 7.35 7.28
7.12 7.21 7.23 7.25 7.28 7.31 7.35 7.29
7.13 7.22 7.25 7.27 7.29 7.32 7.36 7.3
7.07 7.14 7.16 7.18 7.21 7.24 7.29 7.23
7.03 7.09 7.11 7.12 7.14 7.18 7.21 7.16
```

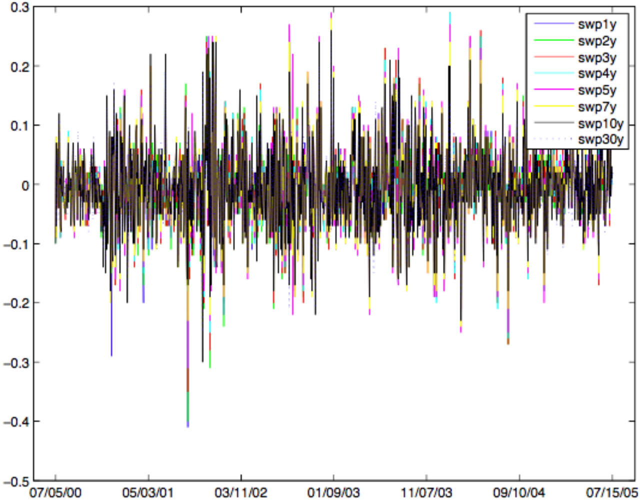
data from

`http://www.stanford.edu/~xing/statfinbook/data.html`



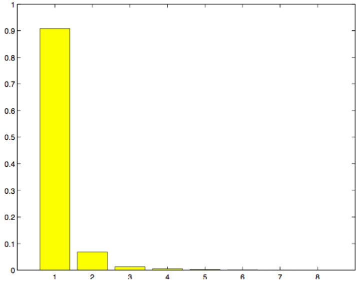
swp1y
swp2y
swp3y
swp4y
swp5y
swp7y
swp10y
swp30y
7
6
5
4
3
2
1
07/05/00
05/03/01
03/11/02
01/09/03
11/07/03
09/10/04
07/15/05

We typically take difference of these time series: $y_t = x_t - x_{t-1}$, to make it more "stationary".

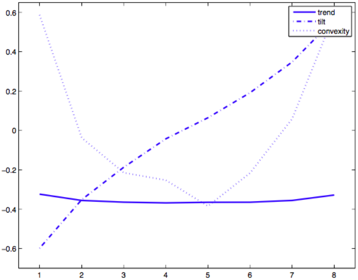


Use the MATLAB command to perform PCA

```
>> data = load('d_swap.txt');
[coeff, eigenvalue, explained]  = pcacov(corrcoef(diff(data)));
eigenvalue', explained'
ans =
    7.2649    0.5477    0.1032    0.0408    0.0221    0.0105    0.0058    0.0051
ans =
   90.8111    6.8459    1.2895    0.5099    0.2757    0.1314    0.0725    0.0640
```



eigenvalues



first 3 eigenvectors

Resulted first 3 reduced dimensions