Principal Component Analysis in a Linear Algebraic View

by Anna Orosz
under the mentorship of Jakob Hansen
Directed Reading Program at the University of Pennsylvania

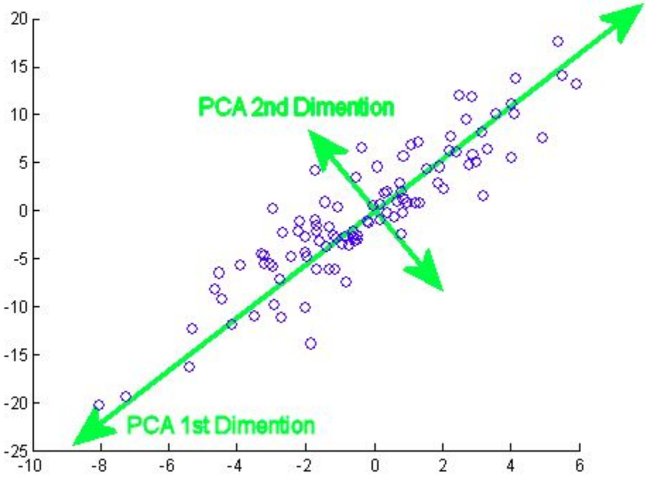
April 30th, 2020

Principal Component Analysis *as a Transformation*

- invented in 1901 by Karl Pearson
- rotation of data from one coordinate system to another
- Goal:
  dimension reduction of multidimensional datasets

Fitting the *Best Ellipsoid* on the data

- multidimensional data:
  - rows: sample values
  - columns: measured variables
- fitting a p-dimensional ellipsoid to the data
- each axis of the ellipsoid represents a principal component
- the small axes represent small variances



Computing PCA
through the EVD of the covariance matrix

1. calculate data covariance matrix of the original data
2. perform eigenvalue decomposition (EVD) on the covariance matrix

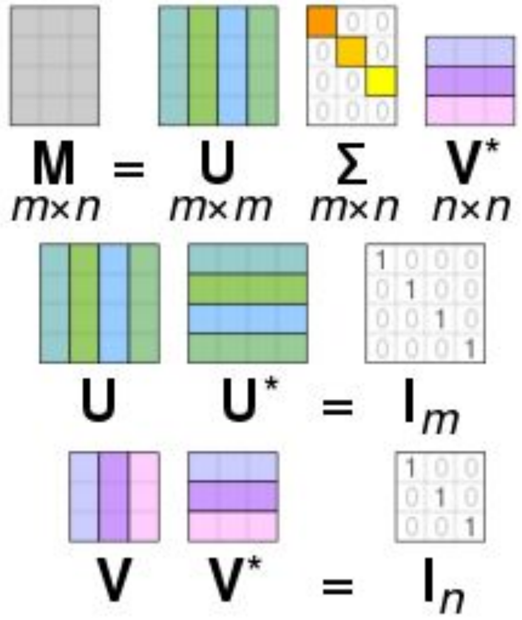
- original data matrix is Y
  - subtract data means from each point
  - X is the shifted version of Y with column-wise 0 empirical mean
- covariance matrix is $X^T * X$
- first component's direction computed by maximizing the variance:
  - other components will be computed by iterating this
  - and with the help of Gram-orthogonalization

Result of computing PCA using EVD

- this way we obtain a W matrix
  - this is *orthonormal*
- result is **T = X*W**
  - W is a p-by-p matrix of weights
  - columns: eigenvectors of $X^T * X$
- last few columns of T can be omitted, in case the majority of the variance can be explained using the first few columns
  - dimension reduction

Another Computational Method: *Singular Value Decomposition*

- factorization of a real or complex matrix
- m*n M matrix is given → SVD gives:
  M= U Σ V^T
  - **U** is m*m unitary matrix (rotation or reflection)
  - **Σ** is an m*n rectangular diagonal matrix
  - **V^T** is an n*n unitary matrix
- diagonal entries $\sigma_i = \Sigma_{ii}$ of Σ are non-negative numbers
  - known as the ***singular values of M***



Computing Principal Component Analysis *using Singular Value Decomposition*

- SVD of the data matrix X: **$X = U\Sigma W^T$**
- we get $T = U\Sigma$ form (polar decomposition of T)
  →*NO* need to determine the covariance matrix
- more numerically stable than using EVD on covariance matrix
- *Primary* method to compute PCA
  - (unless only a handful of components are required)

Why/why not use Principal Component Analysis?

Pros

- reflects our intuitions about the data
- allows estimating probabilities in high-dimensional data
- monumental reduction in size of data
    - faster processing
    - smaller storage

Cons

- cubic time of computing
    - expensive for huge datasets
- only for continuous variables
- assumes linearity of the data
- catastrophic for fine-grained tasks
    - outliers, interesting special cases

Applications of Principal Component Analysis

- quantitative finance
  - risk management of interest rate derivative portfolios
- eigen-faces
  - facial recognition →
- image compression
- countless other applications
  - for example in neuroscience, medical data correlation etc.