Singular Value Decomposition (SVD) Principal Component Analysis (PCA) and Linear Discriminant Analysis (LDA) Machine Learning for Finance (FIN 570)

Instructor: Jaehyuk Choi

Peking University HSBC Business School, Shenzhen, China

2023-24 Module 3 (Spring 2024)

Eigen(spectral) decomposition

For a matrix $\boldsymbol{A}$, eigenvalue λ_k and eigenvector $\boldsymbol{v}_k$ satisfy

$$\boldsymbol{A}\boldsymbol{v}_k = \lambda_k \boldsymbol{v}_k.$$

The matrix $\boldsymbol{A}$ can be decomposed into

$$\boldsymbol{A} = \boldsymbol{Q}\boldsymbol{\Lambda}\boldsymbol{Q}^{-1},$$

where $\boldsymbol{\Lambda}$ is a diagonal matrix with values λ_k and $\boldsymbol{Q} = (\boldsymbol{v}_1 \cdots \boldsymbol{v}_n)$, i.e., $\boldsymbol{Q}_{*j} = \boldsymbol{v}_j$.
When $\boldsymbol{A}$ is real and symmetric, $\boldsymbol{Q}$ is an orthonormal matrix, $\boldsymbol{Q}\boldsymbol{Q}^T = \boldsymbol{I}$,

$$\boldsymbol{A} = \boldsymbol{Q}\boldsymbol{\Lambda}\boldsymbol{Q}^T,$$

Singular Value Decomposition (SVD)

The single most useful practical concept in linear algebra:

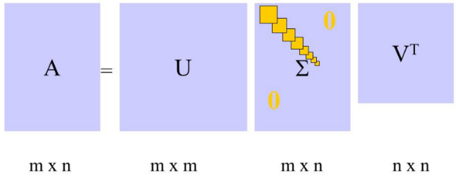
- Any matrix (even rectangular) has a SVD.
- SVD tells everything on a matrix.

For any $m \times n$ matrix A, there is a unique decomposition:

$$A = USV^T,$$

where

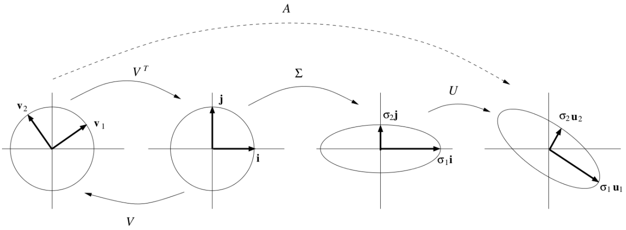
- U $(m \times m)$: orthonormal $(UU^T = U^TU = I)$
- S $(m \times n)$: diagonal. The singular values, s_k for $1 \le k \le m \wedge n$, are positive and in a decreasing order.
- V $(n \times n)$: orthonormal $(VV^T = V^TV = I)$

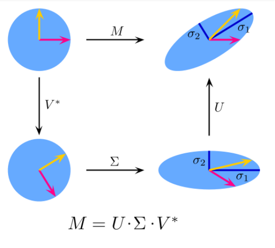


SVD: Intuition

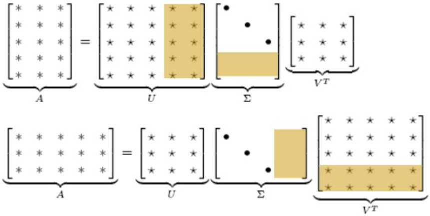
Linear transformation A is decomposed into

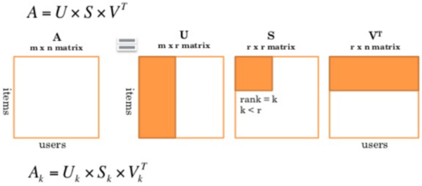
- a rotation by V^T
- a scaling by S
- a rotation by U





SVD: Compact Form, Low Rank Approximation





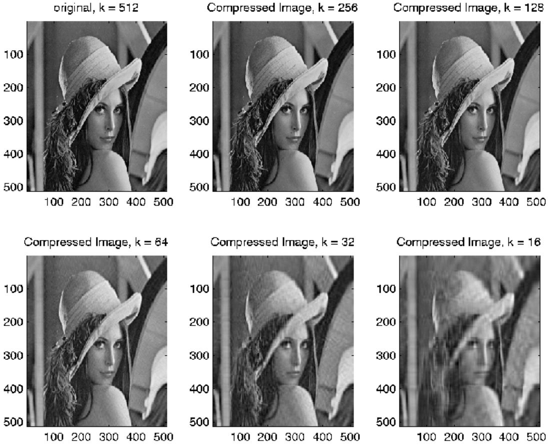
- For a non-square matrix, a compact form is enough: $U\ (m \times r)$, $S\ (r \times r)$, $V\ (n \times r)$ where $r = \min(m, n)$.
- If the rank is k ($\leq r$), $s_{j>k} = 0$: $U\ (m \times k)$, $S\ (k \times k)$, $V\ (n \times k)$
- Using the first j ($\leq k$) biggest singular values,

$$A_j = U_j S_j V_j^T = \sum_{i=1}^{j} \boldsymbol{u}_i s_i \boldsymbol{v}_i^T, \quad U_j\,(m \times j),\ S_j\,(j \times j),\ V_j\,(n \times j)$$

is the best approximation with rank j minimizing the norm $\|A - A_j\|_F$

SVD: Image Compression

An image file is nothing but a matrix, so the low-rank approximation of SVD works as an image compression method. The storage is reduced from mn to $(m + n + 1)k$.



Principal Component Analysis (PCA)

If $\boldsymbol{X}$ is a matrix of n samples of p features $(n \times p)$, the covariance matrix is

$$\boldsymbol{\Sigma} = \frac{1}{n}\boldsymbol{X}^T\boldsymbol{X} : (p \times p) \text{ symmetric matrix}$$

The covariance matrix of the transformed space $\boldsymbol{Z} = \boldsymbol{X}\boldsymbol{W}$ is

$$\text{Cov}(\boldsymbol{Z}) = \frac{1}{n}(\boldsymbol{X}\boldsymbol{W})^T(\boldsymbol{X}\boldsymbol{W}) = \frac{1}{n}\boldsymbol{W}^T(\boldsymbol{X}^T\boldsymbol{X})\boldsymbol{W} = \boldsymbol{W}^T\Sigma\boldsymbol{W}$$

If we pick $\boldsymbol{W}$ to be the orthogonal transformation of SVD, i.e., $\boldsymbol{\Sigma} = \boldsymbol{W}\boldsymbol{S}\boldsymbol{W}^T$,

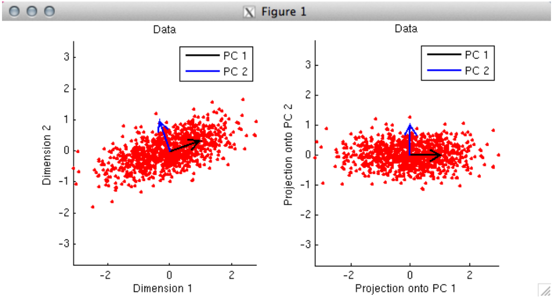
$$\text{Cov}(\boldsymbol{Z}) = \boldsymbol{S} = \text{diag}(S_{11}, \cdots, S_{pp}).$$

Notice that $\text{Cov}(Z_i, Z_j) = \boldsymbol{W}_{*i}^T\boldsymbol{\Sigma}\boldsymbol{W}_{*j} = S_{ij}$ is zero if $i \neq j$, so the extracted features are orthogonal.

Process of finding $\boldsymbol{W}$

Let $\boldsymbol{W} = (\boldsymbol{W}_{*1}\ \boldsymbol{W}_{*2}\ \cdots \boldsymbol{W}_{*p})$.

- Find $\boldsymbol{W}_{*1}$ such that $|\boldsymbol{W}_{*1}| = 1$ and $|\boldsymbol{W}_{*1}^T \boldsymbol{\Sigma} \boldsymbol{W}_{*1}|$ is maximized.
- Find $\boldsymbol{W}_{*2}$ such that $|\boldsymbol{W}_{*2}| = 1$, $|\boldsymbol{W}_{*2}^T \boldsymbol{\Sigma} \boldsymbol{W}_{*2}|$ is maximized and $\boldsymbol{W}_{*1}^T \boldsymbol{W}_{*2} = 0$.
- $\cdots$
- Find $\boldsymbol{W}_{*k}$ such that $|\boldsymbol{W}_{*k}| = 1$, $|\boldsymbol{W}_{*k}^T \boldsymbol{\Sigma} \boldsymbol{W}_{*k}|$ is maximized and $\boldsymbol{W}_{*k}$ is orthogonal to $\{\boldsymbol{W}_{*j}\}$ for $j < k$.



Total and Explained Variance

The total variance is the variance of all original features. Under PCA,

$$\sum_{k=1}^{p} \mathrm{Var}(X_k) = \sum_{k=1}^{p} S_{kk}.$$

Therefore the ratio

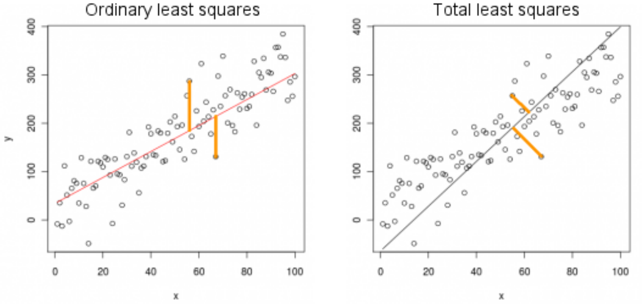
$$\frac{\sum_{j=1}^{k} S_{jj}}{\sum_{j=1}^{p} S_{jj}}$$

indicates how much of the total variance is *explained* by the first k PCA factors. Extracting features from PCA is an unsupervised learning, NOT supervised learning, because the response variable is not associated.

PCA vs Simple Linear Regression for (x, y)

PCA is not same as Simple Linear regression (OLS)!

- **Linear Regression** minimize the the (squared) distance in y-axis.
- **PCA** (1st component) minimize the (squared) shortest distance.



Linear Discriminant Analysis (LDA) as a classifier

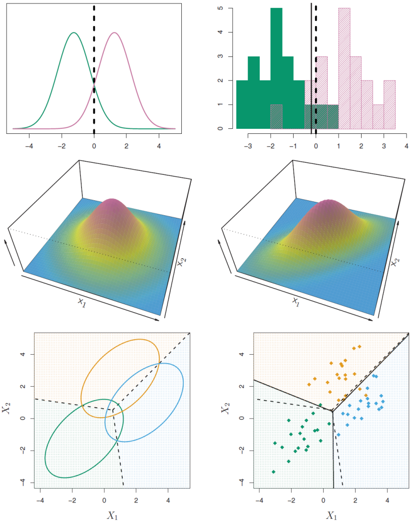
- Assume the samples in each class is distributed by a multivariate normal distribution:

$$f_k(\boldsymbol{x}) = n(\boldsymbol{x}|\hat{\boldsymbol{m}}_k, \hat{\boldsymbol{\Sigma}}_k)$$

- Estimate mean $\hat{\boldsymbol{m}}_k$ and variance $\hat{\boldsymbol{\Sigma}}_k$ of from the samples in the class k.
- Assume that the covariance $\boldsymbol{\Sigma}_k$ is assume to be the average of $\boldsymbol{\Sigma}_k$ (within covariance):

$$\boldsymbol{\Sigma}_W = \frac{1}{N_{\text{total}}} \sum_{k=1}^{K} N_k \boldsymbol{\Sigma}_k$$

- A test sample $\boldsymbol{x}$ is classified to the class k for which $f_k(\boldsymbol{x})$ is largest.
- In quadratic discriminant analysis (QDA), different $\boldsymbol{\Sigma}_k$ are assumed for each k, but the estimation is more complicated.



LDA as a dimensionality reduction tool

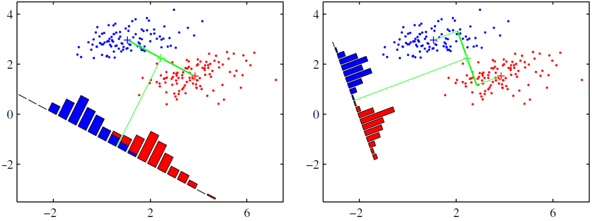
- In the LDA assumption (shared within covariance), which direction $\boldsymbol{w}$ best separates the classes?
- $\boldsymbol{w} \propto (\boldsymbol{m}_2 - \boldsymbol{m}_1)^T$? Probably not the best.
- If $(m_{1,2},\ \sigma^2_{1,2})$ is the mean and variance pair of the samples projected on the $\boldsymbol{w}$ direction ($y_i = \boldsymbol{x}_i \boldsymbol{w}$ with $|\boldsymbol{w}| = 1$), we want to maximize the Fisher criterion:

$$J(\boldsymbol{w}) = \frac{N_{\text{total}}(m_2 - m_1)^2}{N_1\sigma_1^2 + N_2\sigma_2^2} = \frac{\boldsymbol{w}^T(\boldsymbol{m}_2 - \boldsymbol{m}_1)^T(\boldsymbol{m}_2 - \boldsymbol{m}_1)\boldsymbol{w}}{\boldsymbol{w}^T\left(N_1\boldsymbol{\Sigma}_1 + N_2\boldsymbol{\Sigma}_2\right)\boldsymbol{w}} = \frac{\boldsymbol{w}^T\boldsymbol{S}_B\boldsymbol{w}}{\boldsymbol{w}^T\boldsymbol{S}_W\boldsymbol{w}},$$

where $\boldsymbol{S}_W$ and $\boldsymbol{S}_B$ are *within*- and *between*-class variance matrices

$$\boldsymbol{S}_W = \sum_{k=1,2} N_k\,\boldsymbol{\Sigma}_k, \quad \boldsymbol{S}_B = N_{\text{total}}(\boldsymbol{m}_2 - \boldsymbol{m}_1)^T(\boldsymbol{m}_2 - \boldsymbol{m}_1)$$

LDA as a dimensionality reduction



- The direction maximizing is $J(\boldsymbol{w})$ is given by

$$\boldsymbol{w} \propto \boldsymbol{S}_W^{-1}(\boldsymbol{m}_2 - \boldsymbol{m}_1).$$

- In general, the PCA components of $\boldsymbol{S}_W^{-1}\boldsymbol{S}_B$, $\boldsymbol{W}$, are the best directions to separate the classes.
- Similar in PCA, the first few components of $\boldsymbol{W}$ are chosen to form $\hat{\boldsymbol{W}}$.
- The transformation $\boldsymbol{z} = \boldsymbol{x}\hat{\boldsymbol{W}}$ is the extracted factors with reduced dimension can be used as inputs to other ML methods.

PCA versus LDA

- LDA is a supervised method (using y) whereas PCA is an unsupervised method (not using y).
- LDA is not necessarily better than PCA. The performance depends on the classification problem.
- Both are the dimensionality reduction tool. The transformed data is used as inputs to other ML method.
- Both are based on PCA: the whole covariance matrix (PCA) versus $\boldsymbol{S}_W^{-1}\boldsymbol{S}_B$ (LDA).
- They may not work well on nonlinear data.