GENERAL ASSEMBLY

SINGULAR VALUE DECOMPOSITION

Brian Chung

LAST TIME:

- PRINCIPAL COMPONENT ANALYSIS
- COVARIANCE MATRICES
- EIGENVALUE DECOMPOSITION

TYPES OF ML SOLUTIONS

	Continuous	Categorical
Supervised	Regression	Classification
Unsupervised	Dimension Reduction	Clustering

I. EIGENVALUE DECOMPOSITION EXAMPLE
II. SINGULAR VALUE DECOMPOSITION
III. OTHER METHODS

EXERCISE:
IV. SVD IN SCIKIT-LEARN

I. EIGENVALUE DECOMPOSITION

Principal component analysis is a dimension reduction technique that can be used on a matrix of any dimensions.

This procedure produces a new basis, each of whose components retain as much variance from the original data as possible.

The PCA of a matrix A boils down to the **eigenvalue decomposition** ***of the*** **covariance matrix** ***of A.***

Q: We get how to build the covariance matrix. How do you perform an eigenvalue decomposition?

Q: We get how to build the covariance matrix. How do you perform an eigenvalue decomposition?

A: The eigenvalue decomposition of a square matrix A is given by:

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

The columns of Q are the **eigenvectors** *of A, and the values in Λ are the associated* **eigenvalues** *of A.*

For an eigenvector v of A and its eigenvalue λ, we have the important relation:

$$Av = \lambda v$$

$$A = \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix}$$

$$Av = \lambda v$$

$$Av = \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \lambda \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

$$2x_1 + 1x_2 = \lambda x_1$$
$$1x_1 + 2x_2 = \lambda x_2$$

$$2x_1 + 1x_2 = \lambda x_1$$
$$1x_1 + 2x_2 = \lambda x_2$$

$$(2-\lambda)x_1 + x_2 = 0$$
$$x_1 + (2-\lambda)x_2 = 0$$

$$\begin{vmatrix} 2-\lambda & 1 \\ 1 & 2-\lambda \end{vmatrix} = 0$$

$$\begin{vmatrix} 2-\lambda & 1 \\ 1 & 2-\lambda \end{vmatrix} = 0$$

$$(2-\lambda)(2-\lambda)-1=0$$

$$4-4\lambda+\lambda^2-1=0$$

$$\lambda^2-4\lambda+3=0$$

$$(\lambda-3)(\lambda-1)=0$$

$$\lambda=1,3$$

$$\begin{vmatrix} 2-\lambda & 1 \\ 1 & 2-\lambda \end{vmatrix} = 0$$

$$(2-\lambda)(2-\lambda)-1=0$$

$$4-4\lambda+\lambda^2-1=0$$

$$\lambda^2-4\lambda+3=0$$

$$(\lambda-3)(\lambda-1)=0$$

$$\lambda=1,3$$

These are the eigenvalues

$$(2-\lambda)x_1 + x_2 = 0$$
$$x_1 + (2-\lambda)x_2 = 0$$

$$\lambda = 3$$
$$-x_1 + x_2 = 0$$
$$x_1 = x_2$$
$$v_1 = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

$$\lambda = 1$$
$$x_1 + x_2 = 0$$
$$x_1 = -x_2$$
$$v_1 = \begin{bmatrix} 1 \\ -1 \end{bmatrix}$$

$$(2-\lambda)x_1 + x_2 = 0$$
$$x_1 + (2-\lambda)x_2 = 0$$

$$\lambda = 3$$
$$-x_1 + x_2 = 0$$
$$x_1 = x_2$$
$$v_1 = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

These are the eigenvectors

$$\lambda = 1$$
$$x_1 + x_2 = 0$$
$$x_1 = -x_2$$
$$v_1 = \begin{bmatrix} 1 \\ -1 \end{bmatrix}$$

$$A = \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix}$$

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

$$Q = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$$

$$Q^{-1} = \begin{bmatrix} 0.5 & 0.5 \\ 0.5 & -0.5 \end{bmatrix}$$

$$\Lambda = \begin{bmatrix} 3 & 0 \\ 0 & 1 \end{bmatrix}$$

$$A = \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix}$$

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

$$A = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}\begin{bmatrix} 3 & 0 \\ 0 & 1 \end{bmatrix}\begin{bmatrix} .5 & .5 \\ .5 & -.5 \end{bmatrix}$$

$$= \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix}$$

II. SINGULAR VALUE DECOMPOSITION

Lots of math ahead. If you are bewildered, don't worry.

Take your time to read through things later, and take it slow :)

Consider a matrix A with n rows and d features.

Consider a matrix A with n rows and d features.

The **singular value decomposition** *of A is given by:*

$$A = U \Sigma V^T$$

Consider a matrix A with n rows and d features.

The **singular value decomposition** *of A is given by:*

$$\underset{(n \times d)}{A} = \underset{(n \times n)}{U} \ \underset{(n \times d)}{\Sigma} \ \underset{(d \times d)}{V^T}$$

Consider a matrix A with n rows and d features.

The **singular value decomposition** *of A is given by:*

$$\underset{(n \times d)}{A} = \underset{(n \times n)}{U} \; \underset{(n \times d)}{\Sigma} \; \underset{(d \times d)}{V^T}$$

st. U, V are **orthonormal** *matrices and Σ is a* **diagonal** *matrix.*

Consider a matrix A with n rows and d features.

The **singular value decomposition** *of A is given by:*

$$\underset{(n \times d)}{A} = \underset{(n \times n)}{U} \; \underset{(n \times d)}{\Sigma} \; \underset{(d \times d)}{V^T}$$

st. U, V are **orthonormal** *matrices and Σ is a* **diagonal** *matrix.*

→ $UU^T = I_n, \; VV^T = I_d$ → $\Sigma_{ij} = 0 \;\; (i \neq j)$

The **singular value decomposition** *of A is given by:*

$$\underset{(n \times d)}{A} = \underset{(n \times n)}{U} \; \underset{(n \times d)}{\Sigma} \; \underset{(d \times d)}{V^T}$$

The columns of U & V are the (left- and right-) **singular vectors** *of A.*

The **singular value decomposition** *of A is given by:*

$$\underset{(n \times d)}{A} = \underset{(n \times n)}{U}\ \underset{(n \times d)}{\Sigma}\ \underset{(d \times d)}{V^T}$$

The columns of U & V are the (left- and right-) **singular vectors** *of A.*

These singular vectors provide **orthonormal bases** *for the spaces K_n & K_d (columns of U & V, respectively).*

The **singular value decomposition** *of A is given by:*

$$\underset{(n \times d)}{A} = \underset{(n \times n)}{U} \; \underset{(n \times d)}{\Sigma} \; \underset{(d \times d)}{V^T}$$

The nonzero entries of Σ are the **singular values** *of A. These are real, nonnegative, and rank-ordered (decreasing from left to right).*

The **singular value decomposition** *of A is given by:*

$$\underset{(n \times d)}{A} = \underset{(n \times n)}{U}\ \underset{(n \times d)}{\Sigma}\ \underset{(d \times d)}{V^T}$$

NOTE

The number of singular values is equal to the *rank* of A.

The rank of a matrix measures its *non-degeneracy.*

The nonzero entries of Σ are the **singular values** *of A. These are real, nonnegative, and rank-ordered (decreasing from left to right).*

For a general SVD, the columns of U are the eigenvectors of AA^T, and the columns of V are the eigenvectors of A^TA.

Also, the singular values of A are the square roots of the eigenvalues of AA^T and A^TA.

For a general SVD, the columns of U are the eigenvectors of AA^T, and the columns of V are the eigenvectors of A^TA.

Also, the singular values of A are the square roots of the eigenvalues of AA^T and A^TA.

NOTE

If the data is centered, these are proportional to covariance matrices

Q: How do you interpret the SVD?

Q: How do you interpret the SVD?

A: Recall that given a set of n points in d-dimensional space (eg, a matrix A), we want to find the best $k < d$ dimensional subspace to represent the data.

Q: How do you interpret the SVD?

A: Recall that given a set of n points in d-dimensional sp… matrix A), we want to find the best $k < d$ dimensional … represent the data.

NOTE

Here "best" refers to the representation that minimizes the squared *orthogonal* distances from the points to the subspace.

Q: How do you interpret the SVD?

A: Recall that given a set of n points in d-dimensional space (eg, a matrix A), we want to find the best $k < d$ dimensional subspace to represent the data.

For $k = 1$, this subspace is a line passing through the origin.

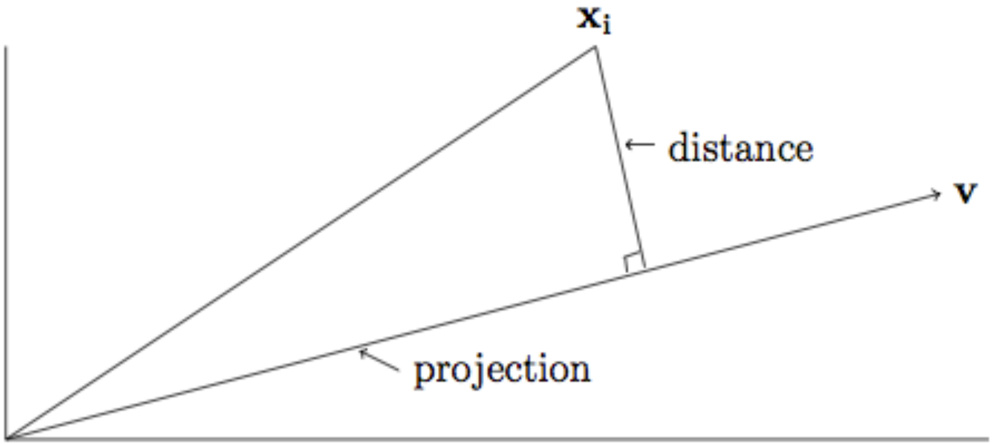


Figure 4.1: The projection of the point $\mathbf{x_i}$ onto the line through the origin in the direction of $\mathbf{v}$

source: http://www.cs.princeton.edu/courses/archive/spring12/cos598C/svdchapter.pdf

For a geometric interpretation of the singular values, consider a unit sphere in R_n and a linear map T (eg, a rotation and a stretch) that sends this sphere to an ellipsoid in R_d.

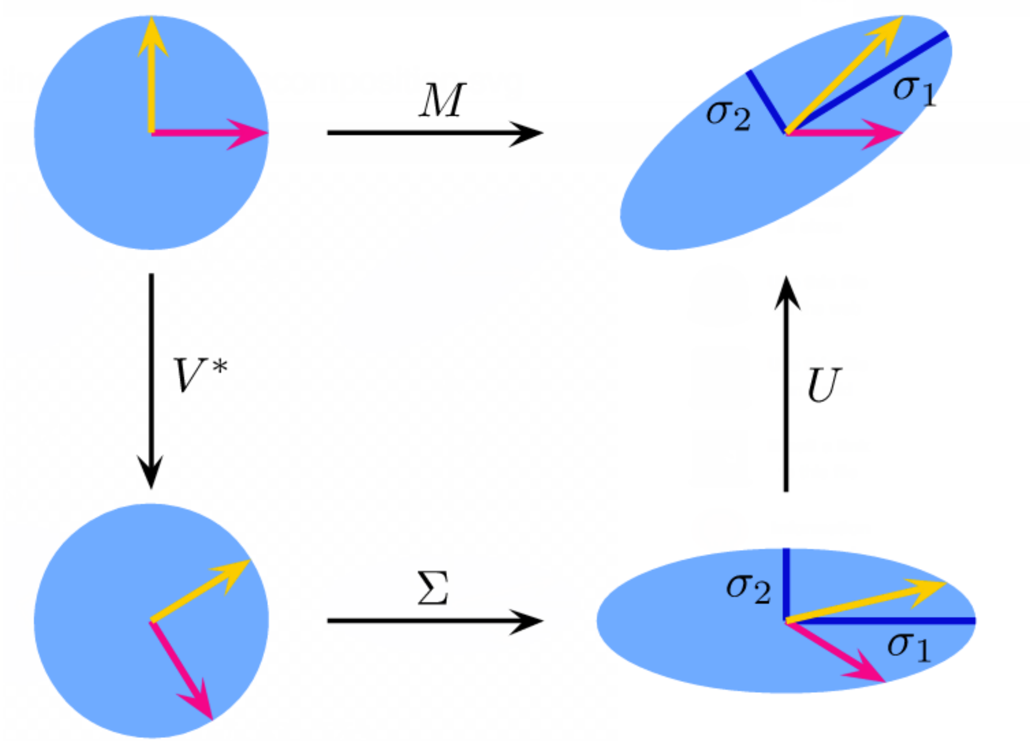
For a geometric interpretation of the singular values, consider a unit sphere in R_n and a linear map T (eg, a rotation and a stretch) that sends this sphere to an ellipsoid in R_d.

The singular vectors of T correspond to the lengths of the axes of the d-dimensional ellipsoid.

For a geometric interpretation of the singular values, consider a unit sphere in R_n and a linear map T (eg, a rotation and a stretch) that sends this sphere to an ellipsoid in R_d.

The singular vectors of T correspond to the lengths of the axes of the d-dimensional ellipsoid.

The singular values give the magnitudes of the projection of each column of the original dataset on the elements of the new basis.



$$M = U \cdot \Sigma \cdot V^*$$

source: http://en.wikipedia.org/wiki/Singular_value_decomposition

- *More numerically stable and efficient to calculate than PCA*
- *Can be used in Latent semantic analysis and recommendation systems*

III. OTHER METHODS

Whereas PCA and SVD create new coordinates by transform the old coordinates, **factor analysis** *requires new coordinates to be specified externally.*

Whereas PCA and SVD create new coordinates by transform the old coordinates, **factor analysis** *requires new coordinates to be specified externally.*

These new coordinates are associated with hidden or latent features that we think our data depends on.

Whereas PCA and SVD create new coordinates by transform the old coordinates, **factor analysis** *requires new coordinates to be specified externally.*

These new coordinates are associated with hidden or latent features that we think our data depends on.

The old coordinates are then modeled as linear combinations of the latent features.

For example, consider a dataset that represents the results of a decathalon (rows = participants, columns = events, entries = times).

For example, consider a dataset that represents the results of a decathalon (rows = participants, columns = events, entries = times).

Though this dataset contains 10 features X_i, we may be interested in modeling these features as functions of latent variables such as the speed and strength of the participants:

$$X_i = \lambda_1 f_1 + \lambda_2 f_2 + \varepsilon$$

For example, consider a dataset that represents the results of a decathalon (rows = participants, columns = events, entries = times).

Though this dataset contains 10 features X_i, we may be interested in modeling these features as functions of latent variables such as the speed and strength of the participants:

$$X_i = \lambda_1 f_1 + \lambda_2 f_2 + \varepsilon$$

This would allow us to analyze the data in a more fundamental way.

SVD, PCA, and factor analysis are all linear techniques (eg, we use a linear transformation to embed the data in a lower-dimensional space).

But as we saw with SVM's, sometimes linear techniques are not sufficient.

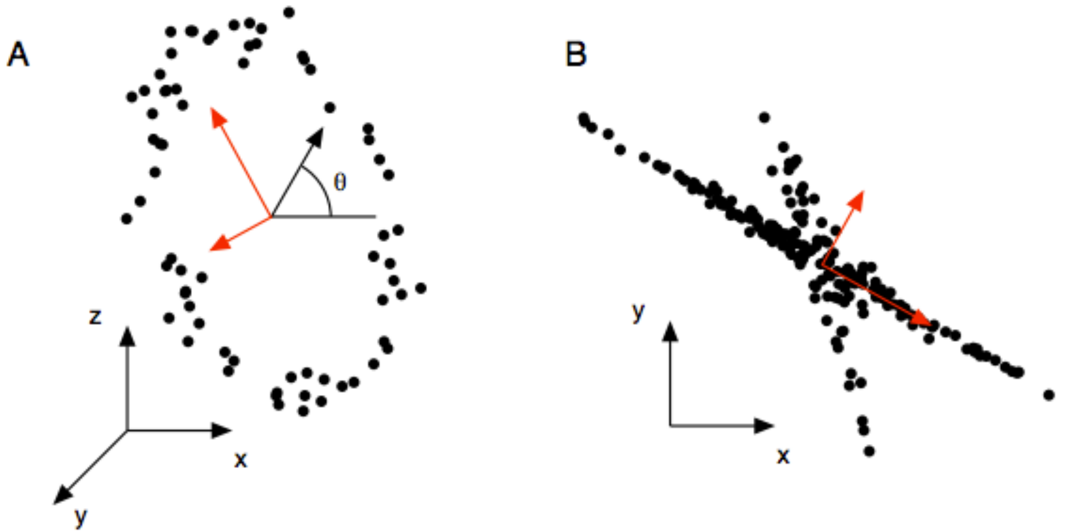
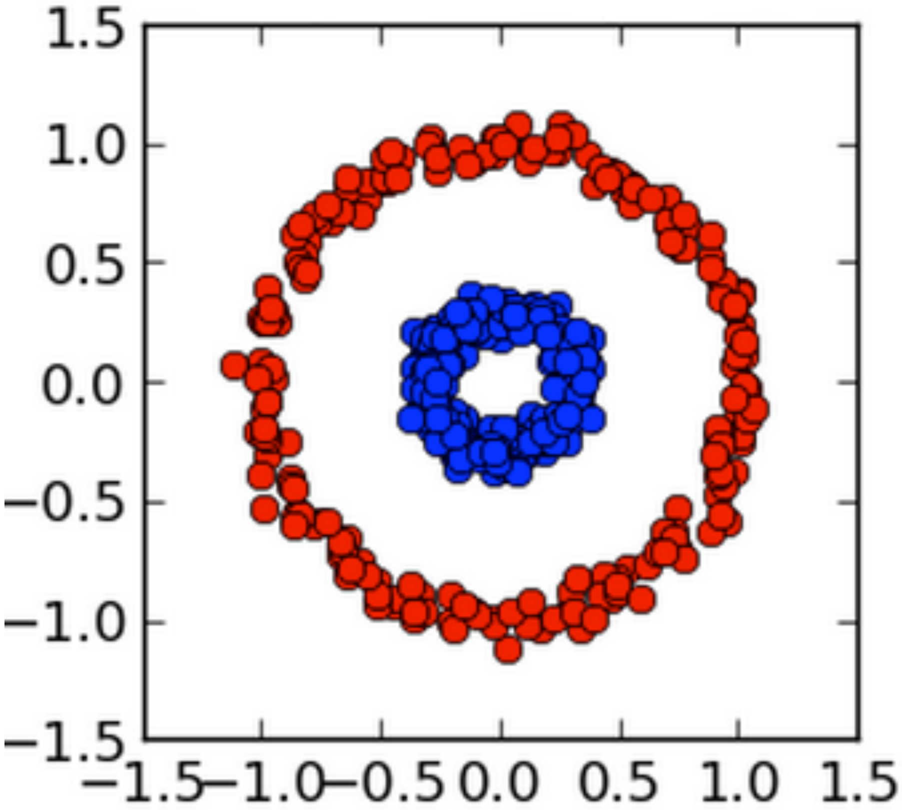


FIG. 6 Example of when PCA fails (red lines). (a) Tracking a person on a ferris wheel (black dots). All dynamics can be described by the phase of the wheel θ, a non-linear combination of the naive basis. (b) In this example data set, non-Gaussian distributed data and non-orthogonal axes causes PCA to fail. The axes with the largest variance do not correspond to the appropriate answer.


1.5
1.0
0.5
0.0
−0.5
−1.0
−1.5
−1.5
−1.0
−0.5
0.0
0.5
1.0
1.5

Some methods for nonlinear dimensional reduction (or manifold learning) include:

multidimensional scaling: *low-dim embedding that preserves pairwise distances*

locally linear embedding: *approximates local structure of data*

Some methods for nonlinear dimensional reduction (or manifold learning) include:

NOTE
See sklearn.manifold

multidimensional scaling: *low-dim embedding that pres*

pairwise distances

locally linear embedding: *approximates local structure of data*

Some methods for nonlinear dimensional reduction (or manifold learning) include:

kernel PCA: *exploits PCA dependence on inner product (same logic as SVM)*

isomap: *nonlinear dim reduction via MDS using geodesic (surface-bound) distances*

In any case, the key difficulties with dimensionality reduction are time/space complexity, randomness (eg different results for different runs), and selecting the number of dimensions in the lower-dim subspace.

In any case, the key difficulties with dimensionality reduction are time/space complexity, randomness (eg different results for different runs), and selecting the number of dimensions in the lower-dim subspace.

Furthermore, there's an obvious (bias/variance) tradeoff between the number of subspace dimensions and the size of approximation error.

THAT'S IT!

- Exit Tickets: DAT1 - Lesson 15 - Guest
- More DR resources:
- PCA https://www.cs.princeton.edu/picasso/mats/PCA-Tutorial-Intuition_jp.pdf
- PCA http://ufldl.stanford.edu/wiki/index.php/PCA
- SVD vs PCA http://math.stackexchange.com/questions/3869/what-is-the-intuitive-relationship-between-svd-and-pca
- How are projects coming?