Machine Learning — Singular Value Decomposition (SVD) & Principal Component Analysis (PCA)

Jonathan Hui Follow
Mar 6 • 16 min read

Photo by Sheldon Nunes

In machine learning (ML), some of the most important linear algebra concepts are the singular value decomposition (SVD) and principal component analysis (PCA). With all the raw data collected, how can we discover structures? For example, with the interest rates of the last 6 days, can we understand its composition to spot trends?

	1 Mo	2 Mo	3 Mo	6 Mo	1 Yr	2 Yr	3 Yr	5 Yr	7 Yr	10 Yr	20 Yr	30 Yr
2/1/19	2.41	2.42	2.4	2.46	2.56	2.52	2.5	2.51	2.59	2.7	2.88	3.03
2/4/19	2.41	2.41	2.42	2.49	2.57	2.53	2.52	2.53	2.62	2.73	2.92	3.06
2/5/19	2.39	2.4	2.42	2.5	2.56	2.53	2.5	2.51	2.6	2.71	2.89	3.03
2/6/19	2.4	2.41	2.42	2.5	2.56	2.52	2.5	2.5	2.59	2.7	2.88	3.03
2/7/19	2.43	2.43	2.42	2.49	2.55	2.48	2.46	2.46	2.54	2.65	2.85	3
2/8/19	2.43	2.43	2.43	2.49	2.54	2.45	2.43	2.44	2.53	2.63	2.82	2.97

This becomes even harder for high-dimensional raw data. It is like finding a needle in a haystack. SVD allows us to extract and untangle information. In this article, we will detail SVD and PCA. We assume you have basic linear algebra knowledge including rank and eigenvectors. If you experience difficulties in reading this article, I will suggest refreshing those concepts first. At the end of the article, we will answer some questions in the interest

rate example above. This article also contains optional sections. Feel free to skip it according to your interest level.

Misconceptions (optional for beginners)

I realize a few common questions that non-beginners may ask. Let me address the elephant in the room first. Is PCA dimension reduction? PCA reduces dimension but it is far more than that. I like the Wiki description (but if you don't know PCA, this is just gibberish):

> *Principal component analysis (PCA) is a statistical procedure that uses an orthogonal transformation to convert a set of observations of possibly correlated variables (entities each of which takes on various numerical values) into a set of values of linearly uncorrelated variables called principal components.*

From a simplified perspective, PCA transforms data linearly into new properties that are not correlated with each other. For ML, positioning PCA as feature extraction may allow us to explore its potential better than dimension reduction.

What is the difference between SVD and PCA? SVD gives you the whole nine-yard of diagonalizing a matrix into special matrices that are easy to manipulate and to analyze. It lay down the foundation to untangle data into independent components. PCA skips less significant components. Obviously, we can use SVD to find PCA by truncating the less important basis vectors in the original SVD matrix.

Matrix diagonalization

In the article on eigenvalue and eigenvectors, we describe a method to decompose an n × n square matrix *A* into

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

For example,

$$A = \begin{bmatrix} | & | & | \\ \mathbf{v}_1 & \mathbf{v}_2 & \mathbf{v}_3 \\ | & | & | \end{bmatrix} \begin{bmatrix} \lambda_1 & 0 & 0 \\ 0 & \lambda_2 & 0 \\ 0 & 0 & \lambda_3 \end{bmatrix} \begin{bmatrix} | & | & | \\ \mathbf{v}_1 & \mathbf{v}_2 & \mathbf{v}_3 \\ | & | & | \end{bmatrix}^{-1}$$

square matrix

the inverse exists only if
eigenvectors are linearly independent

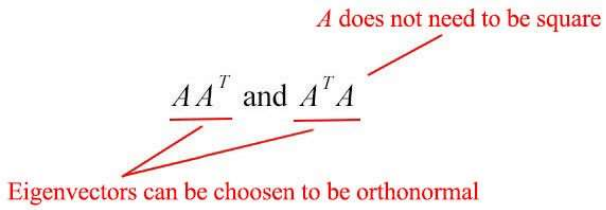
However, this is possible only if A is a square matrix and A has n linearly independent eigenvectors. Now, it is time to develop a solution for all matrices using SVD.

Singular vectors & singular values

The matrix $\boldsymbol{AA^T}$ and $\boldsymbol{A^TA}$ are very special in linear algebra. Consider any m $\times$ n matrix $\boldsymbol{A}$, we can multiply it with A^T to form $\boldsymbol{AA^T}$ and $\boldsymbol{A^TA}$ separately. These matrices are

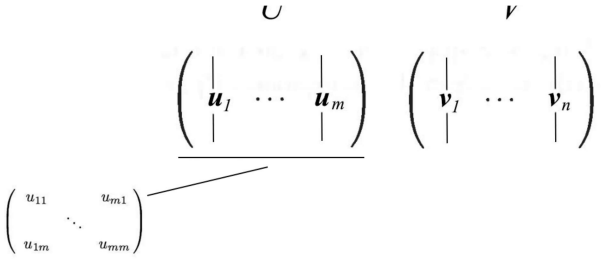
- symmetrical,
- square,
- at least positive semidefinite (eigenvalues are zero or positive),
- both matrices have the same positive eigenvalues, and
- both have the same rank r as A.

In addition, the covariance matrices that we often use in ML are in this form. Since they are symmetric, we can choose its eigenvectors to be orthonormal (perpendicular to each other with unit length) — this is a fundamental property for symmetric matrices.



Let's introduce some terms that frequently used in SVD. We name the eigenvectors for $\boldsymbol{AA^T}$ as u_i and $\boldsymbol{A^TA}$ as v_i here and call these sets of eigenvectors u and v the **singular vectors** of A. Both matrices have the same positive eigenvalues. The square roots of these eigenvalues are called **singular values**.

Not too many explanations so far but let's put everything together first and the explanations will come next. We concatenate vectors u_i into U and v_i into V to form orthogonal matrices.



Since these vectors are orthonormal, it is easy to prove that U and V obey

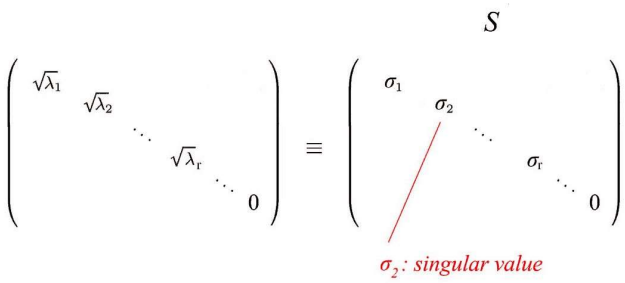
$$U^T U = I$$
$$V^T V = I$$

SVD

Let's start with the hard part first. SVD states that **any** matrix A can be factorized as:

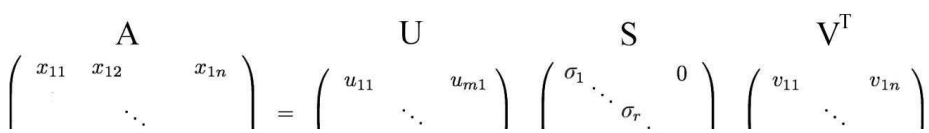
$$A = U S V^T$$

where U and V are orthogonal matrices with orthonormal eigenvectors chosen from $\boldsymbol{AA^T}$ and $\boldsymbol{A^TA}$ respectively. $\boldsymbol{S}$ is a diagonal matrix with r elements equal to the root of the positive eigenvalues of AA^T or A^TA (both matrics have the same positive eigenvalues anyway). The diagonal elements are composed of singular values.



i.e. an m× n matrix can be factorized as:

$$A_{m\times n} = U_{m\times m} S_{m\times n} V^T_{n\times n}$$



$$\underset{m \times n}{\begin{pmatrix} x_{m1} & & x_{mn} \end{pmatrix}} \quad \underset{m \times m}{\begin{pmatrix} u_{1m} & & u_{mm} \end{pmatrix}} \quad \underset{m \times n}{\begin{pmatrix} 0 & \ddots & 0 \end{pmatrix}} \quad \underset{n \times n}{\begin{pmatrix} v_{n1} & & v_{nn} \end{pmatrix}}$$

We can arrange eigenvectors in different orders to produce *U* and *V*. To standardize the solution, we order the eigenvectors such that vectors with higher eigenvalues come before those with smaller values.

$$\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_m$$
$$\begin{pmatrix} | & & | \\ u_1 & \ldots & u_m \\ | & & | \end{pmatrix}$$

Comparing to eigendecomposition, SVD works on non-square matrices. *U* and *V* are invertible for any matrix in SVD and they are orthonormal which we love it. Without proof here, we also tell you that singular values are more numerical stable than eigenvalues.

Example (Source of the example)

Before going too far, let's demonstrate it with a simple example. This will make things very easy to understand.

$$A = \begin{pmatrix} 3 & 2 & 2 \\ 2 & 3 & -2 \end{pmatrix}$$

We calculate:

$$AA^T = \begin{pmatrix} 17 & 8 \\ 8 & 17 \end{pmatrix}, \qquad A^TA = \begin{pmatrix} 13 & 12 & 2 \\ 12 & 13 & -2 \\ 2 & -2 & 8 \end{pmatrix}$$

These matrices are at least positive semidefinite (all eigenvalues are positive or zero). As shown, they share the same positive eigenvalues (25 and 9). The figure below also shows their corresponding eigenvectors.

$$AA^T = \begin{pmatrix} 17 & 8 \\ 8 & 17 \end{pmatrix} \qquad A^TA = \begin{pmatrix} 13 & 12 & 2 \\ 12 & 13 & -2 \\ 2 & -2 & 8 \end{pmatrix}$$

eigenvalues: $\lambda_1 = 25, \lambda_2 = 9$ eigenvalues: $\lambda_1 = 25, \lambda_2 = 9, \lambda_3 = 0$

eigenvectors eigenvectors

$$u_1 = \begin{pmatrix} 1/\sqrt{2} \\ \end{pmatrix} \quad u_2 = \begin{pmatrix} 1/\sqrt{2} \\ \end{pmatrix} \qquad v_1 = \begin{pmatrix} 1/\sqrt{2} \\ 1/\sqrt{2} \end{pmatrix} \quad v_2 = \begin{pmatrix} 1/\sqrt{18} \\ -1/\sqrt{18} \end{pmatrix} \quad v_3 = \begin{pmatrix} 2/3 \\ -2/3 \end{pmatrix}$$

$$\begin{pmatrix} 1/\sqrt{2} \end{pmatrix} \quad \begin{pmatrix} -1/\sqrt{2} \end{pmatrix} \quad \begin{pmatrix} 0 \end{pmatrix} \quad \begin{pmatrix} 4/\sqrt{18} \end{pmatrix} \quad \begin{pmatrix} -1/3 \end{pmatrix}$$

The singular values are the square root of positive eigenvalues, i.e. 5 and 3. Therefore, the SVD composition is

$$A = USV^T = \begin{pmatrix} 1/\sqrt{2} & 1/\sqrt{2} \\ 1/\sqrt{2} & -1/\sqrt{2} \end{pmatrix} \begin{pmatrix} 5 & 0 & 0 \\ 0 & 3 & 0 \end{pmatrix} \begin{pmatrix} 1/\sqrt{2} & 1/\sqrt{2} & 0 \\ 1/\sqrt{18} & -1/\sqrt{18} & 4/\sqrt{18} \\ 2/3 & -2/3 & -1/3 \end{pmatrix}$$

Proof (optional)

To proof SVD, we want to solve *U, S,* and *V* with:

$$A = U S V^T$$
$$U^T U = I$$
$$V^T V = I$$

We have 3 unknowns. Hopefully, we can solve them with the 3 equations above. The transpose of *A* is

$$A = U S V^T$$
$$A^T = (U S V^T)^T = V S^T U^T = V S U^T$$

Knowing

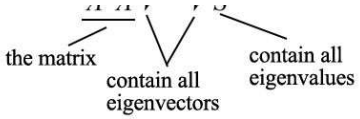
$$U^T U = I$$
$$V^T V = I$$

We compute $A^T A$,

$$A^T A = V S U^T (U S V^T) = V S^2 V^T$$
$$A^T A V = V S^2$$

The last equation is equilvant to the eigenvector definition for the matrix ($A^T A$). We just put all eigenvectors in a matrix.

$$A' v = \lambda v$$
$$A^T A v_i = \sigma_i^2 v_i$$
$$A^T A V = V S^2$$



with VS^2 equals

$$\begin{bmatrix} | & & | \\ v_1 & \cdots & v_n \\ | & & | \end{bmatrix} \begin{bmatrix} \sigma_1^2 & & 0 \\ & \ddots & \\ 0 & & \sigma_n^2 \end{bmatrix} = \begin{bmatrix} | & & | \\ \sigma_1^2 v_1 & \cdots & \sigma_n^2 v_n \\ | & & | \end{bmatrix}$$

V S^2

V hold all the eigenvectors v_i of A^TA and S hold the square roots of all eigenvalues of A^TA. We can repeat the same process for AA^T and come back with a similar equation.

$$A A^T U = U S^2$$

$$\begin{pmatrix} | & & | \\ u_1 & \cdots & u_m \\ | & & | \end{pmatrix} \qquad \begin{pmatrix} \sigma_1^2 & & 0 \\ & \ddots \; \sigma_r^2 & \\ 0 & & 0 \end{pmatrix}$$

Now, we just solve U, V and S for

$$A = U S V^T$$

and prove the theorem.

Recap

The following is a recap of SVD.

$$A = U S V^T$$

where

$$U \qquad V \qquad S$$

$$\begin{pmatrix} | & & | \\ u_1 & \cdots & u_m \\ | & & | \end{pmatrix} \begin{pmatrix} | & & | \\ v_1 & \cdots & v_n \\ | & & | \end{pmatrix} \begin{pmatrix} \sqrt{\lambda_1} & & & \\ & \sqrt{\lambda_2} & & \\ & & \ddots & \\ & & & \sqrt{\lambda_r} \\ & & & & \ddots \; 0 \end{pmatrix}$$

eigenvectors of AA^T A^TA

Reformulate SVD

Since matrix V is orthogonal, V^TV equals I. We can rewrite the SVD equation as:

$$A = U S V^T$$
$$AV = US$$

This equation establishes an important relationship between u_i and v_i.

Recall

$$AB = \left[Ab_1 \;\; Ab_2 \;\; \dots \;\; Ab_n \right]$$

($Ab_1, Ab_2, \dots, Ab_n$: columns of B)

Apply $AV = US$,

$$\underset{m \times n}{A} \quad \underset{n \times r}{V} \quad = \quad \underset{m \times r}{U} \quad \underset{r \times r}{S}$$

$$\begin{pmatrix} x_{11} & x_{12} & & x_{1n} \\ & & \ddots & \\ x_{m1} & & & x_{mn} \end{pmatrix} \begin{pmatrix} v_{11} & & v_{r1} \\ & \ddots & \\ v_{1n} & & v_{rn} \end{pmatrix} = \begin{pmatrix} u_{11} & & u_{r1} \\ & \ddots & \\ u_{1m} & & u_{rm} \end{pmatrix} \begin{pmatrix} \sigma_1 & & 0 \\ & \ddots & \\ 0 & & \sigma_r \end{pmatrix}$$

(Ab_1)

$$A \begin{pmatrix} v_{11} \\ \vdots \\ v_{1n} \end{pmatrix} = \begin{pmatrix} u_{11} & & u_{r1} \\ & \ddots & \\ u_{1m} & & u_{rm} \end{pmatrix} \begin{pmatrix} \sigma_1 \\ \vdots \\ 0 \end{pmatrix}$$

$$= \begin{pmatrix} u_{11} \\ \vdots \\ u_{1m} \end{pmatrix} \sigma_1$$

$$A v_1 = \sigma_1 u_1$$

This can be generalized as

$$\boxed{Av_i = \sigma_i u_i}$$

Recall,

$$\begin{bmatrix} | & & | \\ u_1 & \dots & u_n \\ | & & | \end{bmatrix} \begin{bmatrix} \sigma_1 & & 0 \\ & \ddots & \\ 0 & & \sigma_n \end{bmatrix} = \begin{bmatrix} | & & | \\ \sigma_1 u_1 & \dots & \sigma_n u_n \\ | & & | \end{bmatrix}$$

$$\begin{bmatrix} | & & | \\ u_1 & \cdots & u_m \\ | & & | \end{bmatrix} \begin{bmatrix} \sigma_1 & & \\ & \ddots & \\ & & \sigma_n \end{bmatrix} \begin{bmatrix} — & v_1 & — \\ & \vdots & \\ — & v_n & — \end{bmatrix}$$

and

$$\begin{pmatrix} | & & | \\ u_1 & \cdots & u_m \\ | & & | \end{pmatrix} \begin{pmatrix} — & v_1 & — \\ & \vdots & \\ — & v_n & — \end{pmatrix} = \begin{pmatrix} | \\ u_1 \\ | \end{pmatrix} (— v_1 —) + \ldots + \begin{pmatrix} | \\ u_n \\ | \end{pmatrix} (— v_n —)$$

The SVD decomposition can be recognized as a series of outer products of u_i and v_i.

$$A = \sigma_1 u_1 v_1^T + \ldots + \sigma_r u_r v_r^T$$

This formularization of SVD is the key to understand the components of A. It provides an important way to break down an m × n array of entangled data into r components. Since u_i and v_i are unit vectors, we can even ignore terms ($\sigma_i u_i v_i^T$) with very small singular value σ_i. (We will come back to this later.)

Let's first reuse the example before and show how it works.

$$A = \begin{pmatrix} 3 & 2 & 2 \\ 2 & 3 & -2 \end{pmatrix}$$

The matrix A above can be decomposed as

$$= 5 \begin{pmatrix} 1/\sqrt{2} \\ 1/\sqrt{2} \end{pmatrix} \begin{pmatrix} 1/\sqrt{2} & 1/\sqrt{2} & 0 \end{pmatrix} + 3 \begin{pmatrix} 1/\sqrt{2} \\ -1/\sqrt{2} \end{pmatrix} \begin{pmatrix} 1/\sqrt{18} & -1/\sqrt{18} & 4/\sqrt{18} \end{pmatrix}$$

$$= \begin{pmatrix} 3 & 2 & 2 \\ 2 & 3 & -2 \end{pmatrix}$$

Column space, row space, left nullspace and nullspace (Optional-for advanced users)

Next, we will take a look at what U & V composed of. Let's say A is an m × n matrix of rank r. A^TA will be an n× n symmetric matrix. All symmetric matrices can choose n orthonormal eigenvectors v_j. Because of $Av_i = \sigma_i u_i$ and v_j are orthonormal eigenvectors of A^TA, we can calculate the value of $u_i^T u_j$ as

$$u^{T}_i u_j = \left(\frac{Av_i}{\sigma_i}\right)^{T} \frac{Av_j}{\sigma_j} = \frac{v_i{}^T A^T A v_j}{\sigma_i \sigma_j} = \frac{\sigma_j^2}{\sigma_i \sigma_j} v_i{}^T v_j = 0$$

It equals zero. i.e. u_i and u_j are orthogonal with each other. As shown previously, they are also eigenvectors of AA^T.

From $Av_i = \sigma u_i$, we can recognize that u_i is a column vector of A.

$$Av_i = \sigma_i u_i \quad \text{— a column vector}$$
$$Ax = x_1 a_1 + \ldots + x_n a_n \quad \text{(in general)}$$
$$\text{column vectors}$$

Because A has a rank of r, we can choose these r u_i vectors to be orthonormal. So what are the remaining m - r orthogonal eigenvectors for AA^T? Since left nullspace of A is orthogonal to the column space, it is very natural to pick them as the remaining eigenvector. (The left nullsapce $N(A^T)$ is the space span by x in $A^Tx=0$.) A similar argument will work for the eigenvectors for A^TA. Therefore,

A : m × n matrix of rank r

eigenvectors

$$AA^T \longrightarrow \underbrace{u_1, u_2, u_3, \ldots, u_r}_{\text{span column space of } A}, \underbrace{\ldots, u_m}_{\text{span left nullspace of } A\text{: } N(A^T)}$$

$$A^TA \longrightarrow \underbrace{v_1, v_2, v_3, \ldots, v_r}_{\text{span row space of } A}, \underbrace{\ldots, v_n}_{\text{span nullspace of } A\text{: } N(A)}$$

To get back to the former SVD equation from

$$\underset{m \times n}{A} \quad \underset{n \times r}{V} \quad \underset{m \times r}{U} \quad \underset{r \times r}{S}$$
$$\begin{pmatrix} x_{11} & x_{12} & & x_{1n} \\ & & \ddots & \\ x_{m1} & & & x_{mn} \end{pmatrix} \begin{pmatrix} v_{11} & & v_{r1} \\ & \ddots & \\ v_{1n} & & v_{rn} \end{pmatrix} = \begin{pmatrix} u_{11} & & u_{r1} \\ & \ddots & \\ u_{1m} & & u_{rm} \end{pmatrix} \begin{pmatrix} \sigma_1 & & 0 \\ & \ddots & \\ 0 & & \sigma_r \end{pmatrix}$$

We simply put back the eigenvectors in the left nullspace and nullspace.

$$\underset{m \times n}{A} \quad \underset{m \times m}{U} \quad \underset{m \times n}{S} \quad \underset{n \times n}{V^T}$$
$$\begin{pmatrix} x_{11} & x_{12} & & x_{1n} \\ & & \ddots & \\ x_{m1} & & & x_{mn} \end{pmatrix} = \begin{pmatrix} u_{11} & & u_{m1} \\ & \ddots & \\ u_{1m} & & u_{mm} \end{pmatrix} \begin{pmatrix} \sigma_1 & & & 0 \\ & \ddots & & \\ & & \sigma_r & \\ 0 & & \ldots & 0 \end{pmatrix} \begin{pmatrix} v_{11} & & v_{1n} \\ & \ddots & \\ v_{n1} & & v_{nn} \end{pmatrix}$$

Moore-Penrose Pseudoinverse

For a linear equation system, we can compute the inverse of a square matrix A to solve x.

$$Ax = b$$
$$x = A^{-1} b$$

But not all matrices are invertible. Also, in ML, it will be unlikely to find an exact solution with the presence of noise in data. Our objective is to find the model that best fit the data. To find the best-fit solution, we compute a pseudoinverse

$$A^{+}$$

which minimizes the least square error below.

$$\|AA^{+} - I_n\|_2$$

And the solution for x can be estimated as,

$$Ax = b$$
$$x \approx A^{+} b$$

In a linear regression problem, $\boldsymbol{x}$ is our linear model, $\boldsymbol{A}$ contains the training data and b contains the corresponding labels. We can solve x by

$$Ax = b$$
$$(U, D, V) \leftarrow svd(A)$$
$$A^{+} = VD^{+}U^{T}$$
$$x = A^{+}b$$

where $\boldsymbol{D}^{+}$ takes the reciprocal $\frac{1}{x_i}$ of the non-zero elements of $\boldsymbol{D}$

$$\begin{bmatrix} -3 & 0 & 0 \\ 0 & 4 & 0 \\ 0 & 0 & 0 \end{bmatrix}^{+} = \begin{bmatrix} -1/3 & 0 & 0 \\ 0 & -1/4 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

Here is an example.

$$A = \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix} \text{ (singular, inverse does not exist)}$$

$$A = U\Sigma V^T = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}\begin{bmatrix} 2 & 0 \\ 0 & 0 \end{bmatrix}\frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$$

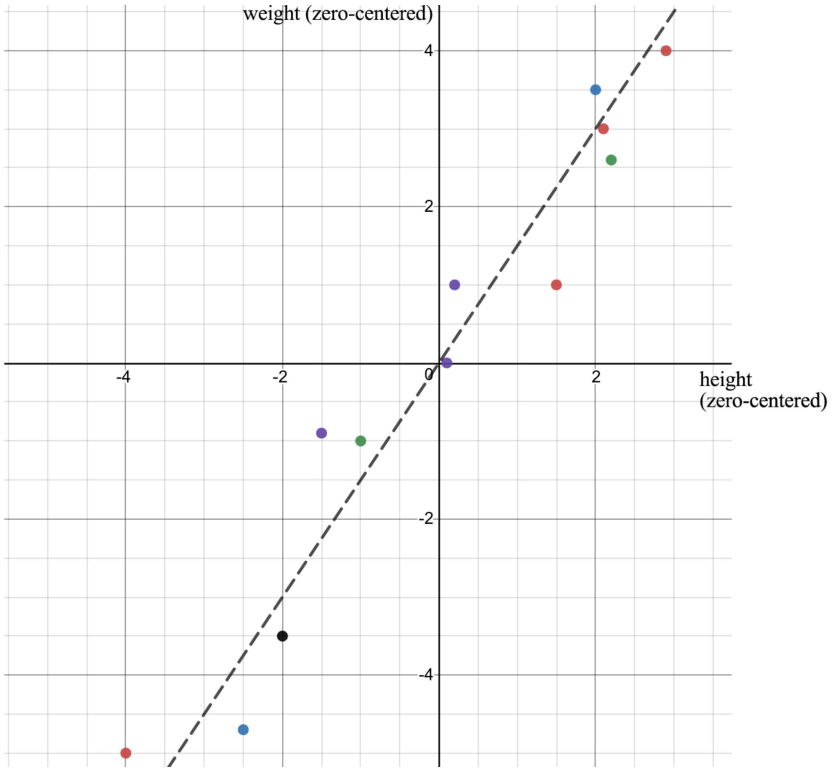
$$A^+ = V\Sigma^+U^T = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}\begin{bmatrix} 1/2 & 0 \\ 0 & 0 \end{bmatrix}\frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} = \frac{1}{4}\begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}$$

Variance & covariance

In ML, we identify patterns and relationship. How do we identify the correlation of properties in data? Let's start the discussion with an example. We sample the height and weight of 12 people and compute their means. We zero-center the original values by subtracting them with its mean. For example, Matrix *A* below holds the adjusted zero-centered height and weight.

$$A = \begin{bmatrix} 2.9 & -1.5 & 0.1 & -1.0 & 2.1 & -4.0 & -2.0 & 2.2 & 0.2 & 2.0 & 1.5 & -2.5 \\ 4.0 & -0.9 & 0.0 & -1.0 & 3.0 & -5.0 & -3.5 & 2.6 & 1.0 & 3.5 & 1.0 & -4.7 \end{bmatrix} \begin{matrix} \text{height} \\ \text{weight} \end{matrix}$$

As we plot the data points, we can recognize height and weight are positively related. But how can we quantify such a relationship?



First, how does a property vary? We probably learn the variance from high school. Let's introduce its cousin. **Sample variance** is defined as :

$$\text{Sample variance } S^2 = \frac{1}{(n-1)}\left[(x_1 - m)^2 + \ldots + (x_n - m)^2\right] = \frac{53.46}{11} \text{ (for the height)}$$

Note, it is divided by *n-1* instead of *n* in the **variance**. With a limited size of the samples, the sample mean is biased and correlated with the samples. The average square distance from this mean will be smaller than that from the general population. The sample covariance S^2, divided by *n-1,* compensates for the smaller value and can be proven to be an unbiased estimate for variance σ^2. (The proof is not very important so I will simply provide a link for the proof here.)

Covariance matrices

Variance measures how a variable varies between itself while covariance is between two variables (*a* and *b*).

$$\sigma_{ab}^2 = \text{cov}(a, b) = \text{E}[(a - \bar{a})(b - \bar{b})]$$
$$\sigma_a^2 = \text{var}(a) = \text{cov}(a, a) = \text{E}[(a - \bar{a})^2]$$

We can hold all these possible combinations of covariance in a matrix called the **covariance matrix** Σ.

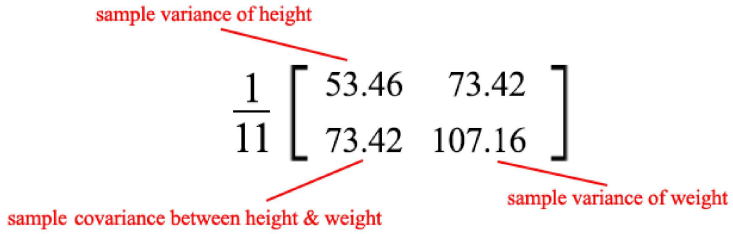
$$\Sigma = \begin{pmatrix} E[(x_1 - \mu_1)(x_1 - \mu_1)] & E[(x_1 - \mu_1)(x_2 - \mu_2)] & \cdots & E[(x_1 - \mu_1)(x_p - \mu_p)] \\ E[(x_2 - \mu_2)(x_1 - \mu_1)] & E[(x_2 - \mu_2)(x_2 - \mu_2)] & \cdots & E[(x_2 - \mu_2)(x_p - \mu_p)] \\ \vdots & \vdots & \ddots & \vdots \\ E[(x_p - \mu_p)(x_1 - \mu_1)] & E[(x_p - \mu_p)(x_2 - \mu_2)] & \cdots & E[(x_p - \mu_p)(x_p - \mu_p)] \end{pmatrix}$$

We can rewrite this in a simple matrix form.

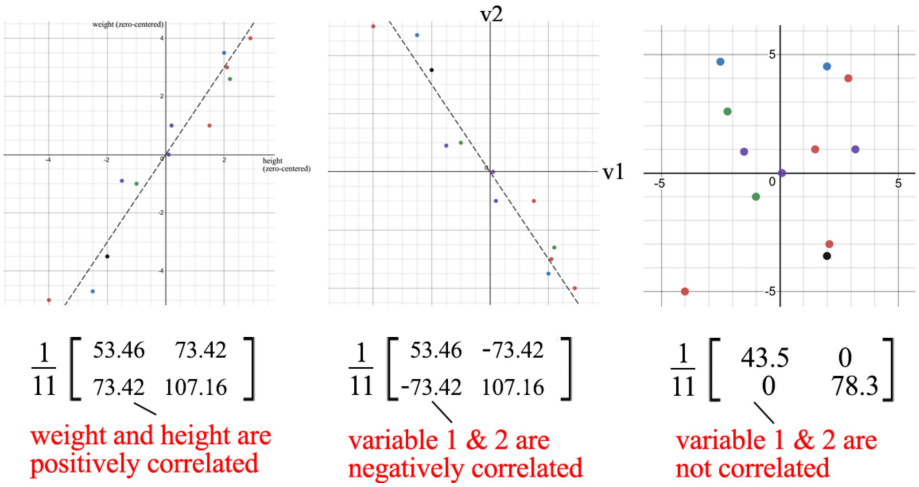
$$\Sigma = \text{E}[(X - \bar{X})(X - \bar{X})^{\text{T}}]$$
$$\Sigma = \frac{XX^{\text{T}}}{n} \quad (\text{if } X \text{ is already zero centered})$$

The diagonal elements hold the variances of individual variables (like height) and the non-diagonal elements hold the covariance between two variables. Let's compute the sample covariance now.

$$\text{Sample covariance } S^2 = \frac{AA^T}{(n-1)} = \frac{1}{11}\begin{bmatrix} 53.46 & 73.42 \\ 73.42 & 107.16 \end{bmatrix}$$



The positive sample covariance indicates weight and height are positively correlated. It will be negative if they are negatively correlated and zero if they are independent.



$$\frac{1}{11}\begin{bmatrix} 53.46 & 73.42 \\ 73.42 & 107.16 \end{bmatrix}$$

$$\frac{1}{11}\begin{bmatrix} 53.46 & -73.42 \\ -73.42 & 107.16 \end{bmatrix}$$

$$\frac{1}{11}\begin{bmatrix} 43.5 & 0 \\ 0 & 78.3 \end{bmatrix}$$

Covariance matrix & SVD

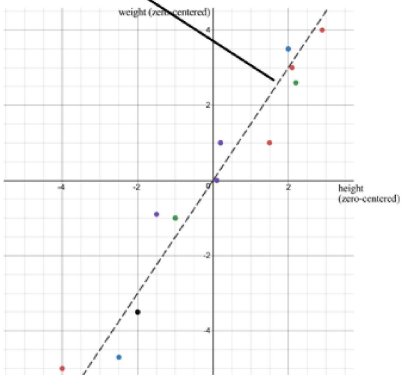
We can use SVD to decompose the sample covariance matrix. Since $\sigma 2$ is relatively small compared with $\sigma 1$, we can even ignore the $\sigma 2$ term. When we train an ML model, we can perform a linear regression on the weight and height to form a new property rather than treating them as two separated and correlated properties (where entangled data usually make model training harder).

$$\frac{1}{11}\begin{bmatrix} 53.46 & 73.42 \\ 73.42 & 107.16 \end{bmatrix}$$

$$\sigma_1 = 14.4$$
$$\sigma_2 = 0.19$$
$$S = \sigma_1 u_1 v_1^T + \sigma_2 u_2 v_2^T$$

$u1$ has one significant importance. It is the principal component of S.

$$S = \sigma_1 u_1 v_1^T + \sigma_2 u_2 v_2^T$$

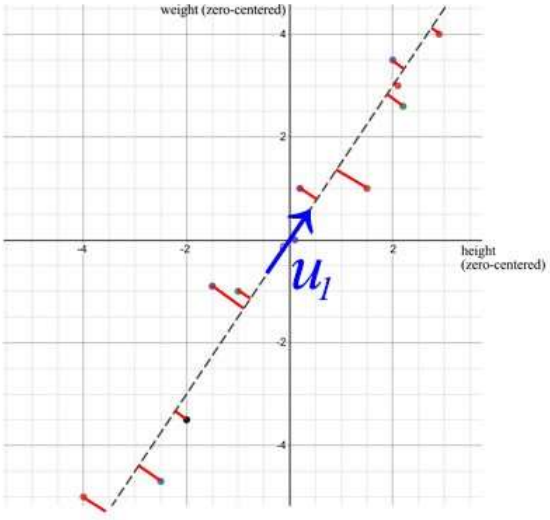


There are a few properties about a sample covariance matrix under the context of SVD:

- The total variance of the data equals the sum of squares of *S*'s singular values. Equipped with this, we can calculate the ratio of variance lost if we drop smaller σ_i terms. This reflects the amount of information lost if we eliminate them.

$$\mathrm{Tr}(S) = \sigma_1^2 + \sigma_2^2 + \ldots + \sigma_m^2$$

- The first eigenvector *u1* of *S* points to the most important direction of the data. In our example, it quantifies the typical ratio between weight and height.



Perpendicular least squares

- The error, calculated as the sum of the perpendicular squared distance from the sample points to *u1*, is the minimum when SVD is used.

Property

Covariance matrices are not only symmetric but they are also positive semidefinite. Because variance is positive or zero, u^TVu below is always greater or equal zero. By the energy test, V is positive semidefinite.

$$variance \geq 0$$

$$\begin{aligned} var(u^TX) &= \mathrm{E}[(u^TX - u^T\bar{X})\,(u^TX - u^T\bar{X})^T] \\ &= u^T\,\mathrm{E}[(X - \bar{X})\,(X - \bar{X})^T]\,u \\ &= u^TVu \end{aligned}$$

Therefore,

$$S = Q\Lambda Q^T \qquad \lambda \geq 0$$

Often, after some linear transformation A, we want to know the covariance of the transformed data. This can be calculated with the transformation matrix A and the covariance of the original data.

$$Z = AX$$
$$V_z = A\,V_x\,A^T$$

V_z: covariance of z; V_x: covariance of x

Correlation matrix

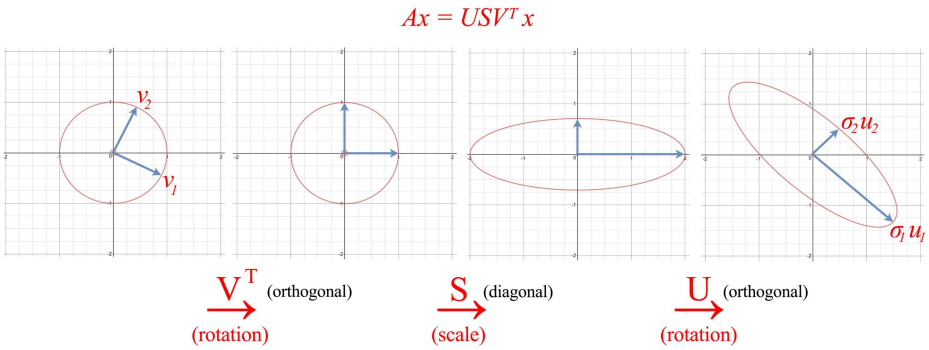
A correlation matrix is a scaled version of the covariance matrix. A correlation matrix standardizes (scale) the variables to have a standard deviation of 1.

$$R = \begin{pmatrix} 1 & \frac{E[(x_1-\mu_1)(x_2-\mu_2)]}{\sigma_1\sigma_2} & \cdots & \frac{E[(x_1-\mu_1)(x_p-\mu_p)]}{\sigma_1\sigma_p} \\ \frac{E[(x_2-\mu_2)(x_1-\mu_1)]}{\sigma_1\sigma_2} & 1 & & \\ \vdots & & \ddots & \vdots \\ \frac{E[(x_p-\mu_p)(x_1-\mu_1)]}{\sigma_1\sigma_p} & & \cdots & 1 \end{pmatrix}$$

Correlation matrix will be used if variables are in scales of very different magnitudes. Bad scaling may hurt ML algorithms like gradient descent.

Visualization

So far, we have a lot of equations. Let's visualize what SVD does and develop the insight gradually. SVD factorizes a matrix A into USV^T. Applying A to a vector x (Ax) can be visualized as performing a rotation (V^T), a scaling (S) and another rotation (U) on x.



As shown above, the eigenvector v_i of V is transformed into:

$$Av_i = \sigma_i u_i$$

Or in the full matrix form

$$\underset{m \times n}{A} = \underset{m \times m}{U} \underset{m \times n}{S} \underset{n \times n}{V^T}$$

rotation stretch rotation

demonstrate for r = m < n

Insight of SVD

As described before, the SVD can be formulated as

$$A = USV^T = \sigma_1 u_1 v_1^T + \ldots + \sigma_r u_r v_r^T$$

more significant, less significant

$m \times n$ = $m \times 1$ $1 \times n$ + ... + $m \times n$

Since u_i and v_i have unit length, the most dominant factor in determining the significance of each term is the singular value σ_i. We purposely sort σ_i in

the descending order. If the eigenvalues become too small, we can ignore the remaining terms ($+ \sigma_i u_i v_i^T + \ldots$).

$$\begin{pmatrix} 2.5 & 2.5 & 0 \\ 2.5 & 2.5 & 0 \end{pmatrix} \qquad \begin{pmatrix} 0.5 & -0.5 & 2 \\ -0.5 & 0.5 & -2 \end{pmatrix}$$

$$= 5 \begin{pmatrix} 1/\sqrt{2} \\ 1/\sqrt{2} \end{pmatrix} \begin{pmatrix} 1/\sqrt{2} & 1/\sqrt{2} & 0 \end{pmatrix} + 3 \begin{pmatrix} 1/\sqrt{2} \\ -1/\sqrt{2} \end{pmatrix} \begin{pmatrix} 1/\sqrt{18} & -1/\sqrt{18} & 4/\sqrt{18} \end{pmatrix}$$

This formularization has some interesting implications. For example, we have a matrix contains the return of stock yields traded by different investors.

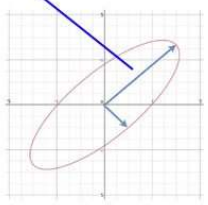
	A	B	C	D	E	F	G
1		GOOG	AMZN	FB	SNAP	...	CRM
2	John	0.5	0.2	0.1	0.04		0.3
3	Mary	0	0	0.2	0.8		0.25
4	Sam	0.5	0.67	0	0.04		0.1
5	Tomas	0.75	0.2	0.3	0.15		0.4
6	...						
7	Mark	0	0.3	0.6	0.02		0

As a fund manager, what information can we get out of it? Finding patterns and structures will be the first step. Maybe, we can identify the combination of stocks and investors that have the largest yields. SVD decompose an n × n matrix into *r* components with the singular value σ_i demonstrating its significant. Consider this as a way to extract entangled and related properties into fewer principal directions with no correlations.

$$A = \sigma_1 u_1 v_1^T + \ldots + \sigma_r u_r v_r^T$$

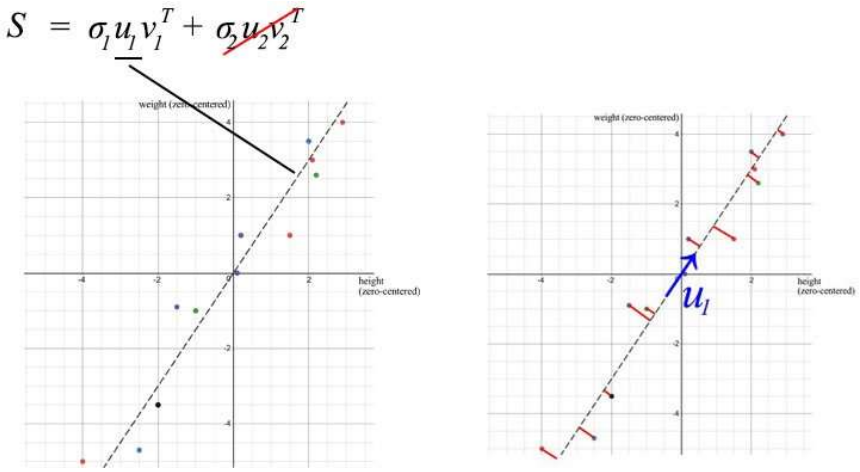
If data is highly correlated, we should expect many σ_i values to be small and can be ignored.

$$A = U S V^T = \sigma_1 u_1 v_1^T + \ldots + \sigma_n u_n v_n^T$$



In our previous example, weight and height are highly related. If we have a matrix containing the weight and height of 1000 people, the first component in the SVD decomposition will dominate. The *u1* vector indeed

demonstrates the ratio between weight and height among these 1000 people as we discussed before.



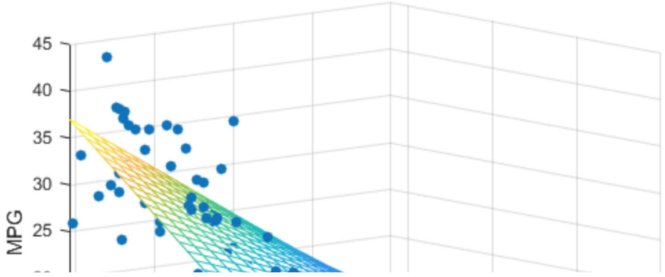
Principal Component Analysis (PCA)

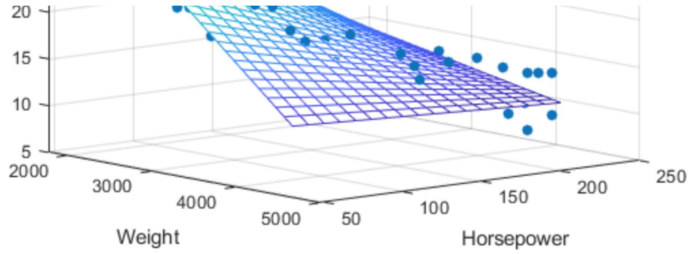
Technically, SVD extracts data in the directions with the highest variances respectively. PCA is a linear model in mapping *m*-dimensional input features to *k*-dimensional latent factors (*k* principal components). If we ignore the less significant terms, we remove the components that we care less but keep the principal directions with the highest variances (largest information).

$$W = \begin{pmatrix} w_{11} & w_{12} & w_{13} & w_{14} & w_{15} & w_{16} \\ w_{21} & w_{22} & w_{23} & w_{24} & w_{25} & w_{26} \\ w_{31} & w_{32} & w_{33} & w_{34} & w_{35} & w_{36} \end{pmatrix}, \quad x = \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \end{pmatrix}$$

$$v = Wx = \begin{pmatrix} v_1 \\ v_2 \\ v_3 \end{pmatrix}$$

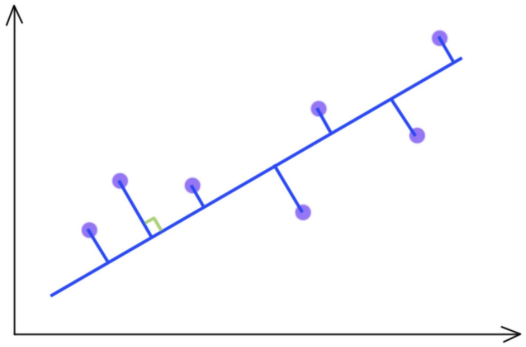
Consider the 3-dimensional data points that displayed as blue dots below. It can be approximated by a plane easily.



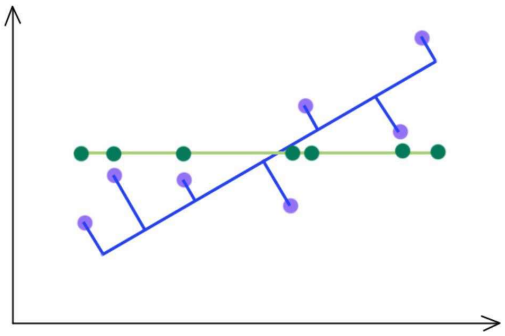


Source

You may quickly realize that we can use SVD to find the matrix *W*. Consider the data points below that lie on a 2-D space.



SVD selects a projection that maximizes the variance of their output. Hence, PCA will pick the blue line over the green line if it has a higher variance.



As indicated below, we keep the eigenvectors that have the top *k*th highest singular value.

$$A \qquad U \qquad S \qquad V^T$$

$$\begin{pmatrix} x_{11} & & x_{1n} \\ & \ddots & \\ x_{m1} & & x_{mn} \end{pmatrix}_{m \times n} \rightarrow \begin{pmatrix} u_{11} & u_{kl} & u_{r1} \\ & \ddots & \ddots \\ u_{1m} & u_{km} & u_{rm} \end{pmatrix} \begin{pmatrix} \sigma_1 & \ddots & 0 \\ & \sigma_k & \ddots \\ 0 & & \sigma_r \end{pmatrix} \begin{pmatrix} v_{11} & v_{1k} & v_{1n} \\ & \ddots & \ddots \\ v_{r1} & v_{rk} & v_{rn} \end{pmatrix}$$

$$\rightarrow \begin{pmatrix} u_{11} & u_{kl} \\ & \ddots \\ u_{1m} & u_{km} \end{pmatrix} \begin{pmatrix} \sigma_1 & 0 \\ & \ddots \\ 0 & \sigma_k \end{pmatrix} \begin{pmatrix} v_{11} & v_{1n} \\ & \ddots \\ v_{kl} & v_{kn} \end{pmatrix}$$

Interest rate

Let's illustrate the concept deeper by retracing an example here with the interest rate data originated from the US Treasurer Department. The basis point for 9 different interests rates (from 3 months, 6 months, … to 20 years) over 6 consecutive business days are stored in *A* below. ***A*** has its elements subtracted by its mean over the 6-day period already. i.e. it is zero-centered (across its row).

$$A = \begin{bmatrix} -5.4 & -19.4 & -12.4 & 19.6 & 17.6 \\ -4.6 & -14.6 & -12.6 & 14.4 & 17.4 \\ 1 & -14 & -14 & 9 & 18 \\ 9.6 & -10.4 & -16.4 & 2.6 & 14 \\ 13.4 & -10.6 & -17.6 & 1.4 & 13.4 \\ 18.6 & -11.4 & -15.4 & -0.4 & 8.6 \\ 20.8 & -11.2 & -14.2 & 0.8 & 3.8 \\ 20.8 & -12.2 & -11.2 & -0.2 & 2.8 \\ 14.6 & -7.4 & -7.4 & 0.6 & -0.4 \end{bmatrix}$$

$$A^T = \begin{bmatrix} -5.4 & -4.6 & 1 & 9.6 & 13.4 & 18.6 & 20.8 & 20.8 & 14.6 \\ -19.4 & -14.6 & -14 & -10.4 & -10.6 & -11.4 & -11.2 & -12.2 & -7.4 \\ -12.4 & -12.6 & -14 & -16.4 & -17.6 & -15.4 & -14.2 & -11.2 & -7.4 \\ 19.6 & 14.4 & 9 & 2.6 & 1.4 & -0.4 & 0.8 & -0.2 & 0.6 \\ 17.6 & 17.4 & 18 & 14 & 13.4 & 8.6 & 3.8 & 2.8 & -0.4 \end{bmatrix}$$

$$= \begin{bmatrix} 1253.2 & 1052.8 & 933 & 650.64 & 614.8 & 455.20 & 363.60 & 308.60 & 161.20 \\ 1052.8 & 903.2 & 819 & 595.36 & 568.2 & 418.80 & 324.4 & 269.40 & 135.80 \\ 933 & 819 & 798 & 660.20 & 662 & 545. & 452 & 397 & 220 \\ 650.64 & 595.36 & 660.2 & 672.04 & 718.76 & 669.04 & 604.32 & 548.92 & 334.44 \\ 614.8 & 568.2 & 662 & 718.76 & 783.2 & 755.80 & 699.4 & 642.40 & 399.80 \\ 455.2 & 418.8 & 545 & 669.04 & 755.8 & 787.2 & 765.6 & 722.6 & 466.2 \\ 363.6 & 324.4 & 452 & 604.32 & 699.4 & 765.6 & 774.8 & 738.8 & 490.6 \\ 308.6 & 269.4 & 397 & 548.92 & 642.4 & 722.6 & 738.8 & 714.8 & 475.6 \\ 161.2 & 135.8 & 220. & 334.44 & 399.8 & 466.2 & 490.6 & 475.6 & 323.20 \end{bmatrix}$$

$$AA^T$$

measures the covariance of interest rates with different maturities

The sample covariance matrix equals $S = AA^T/(5–1)$.

$$\begin{bmatrix} 313.3 & 263.2 & 233.25 & 162.66 & 153.7 & 113.8 & 90.9 & 77.15 & 40.3 \\ 263.2 & 225.8 & 204.75 & 148.84 & 142.05 & 104.7 & 81.1 & 67.35 & 33.95 \\ 233.25 & 204.75 & 199.5 & 165.05 & 165.5 & 136.25 & 113 & 99.25 & 55 \end{bmatrix}$$

162.66	148.84	165.05	168.01	179.69	167.26	151.08	137.23	83.61
153.7	142.05	165.5	179.69	195.8	188.95	174.85	160.6	99.95
113.8	104.7	136.25	167.26	188.95	196.8	191.4	180.65	116.55
90.9	81.1	113	151.08	174.85	191.4	193.7	184.7	122.65
77.15	67.35	99.25	137.23	160.6	180.65	184.7	178.7	118.9
40.3	33.95	55	83.61	99.95	116.55	122.65	118.9	80.8

Now we have the covariance matrix S that we want to factorize. The SVD decomposition is

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

Matrix U:

-0.3833	-0.5302	0.4795	-0.0682	0.1048	0.4589	0.2131	0.0536	-0.2598
-0.3366	-0.4376	0.0414	-0.2007	-0.1434	-0.7669	-0.1969	0.0627	-0.0424
-0.3584	-0.2643	-0.2330	0.5053	-0.1726	0.1715	-0.0890	-0.1437	0.6359
-0.3492	0.0333	-0.4418	-0.1571	0.8106	-0.0000	-0.0000	-0.0000	-0.0000
-0.3718	0.1302	-0.4405	-0.2266	-0.4496	0.2544	-0.1926	-0.3331	-0.4296
-0.3505	0.2925	-0.1224	0.2030	-0.1904	-0.1896	0.7274	0.3519	-0.1014
-0.3242	0.3648	0.2276	-0.4517	-0.1182	0.1784	-0.2936	0.4664	0.3954
-0.2984	0.3771	0.3541	0.5696	0.1594	-0.1212	-0.4172	-0.0258	-0.3226
-0.1848	0.2803	0.3642	-0.2322	0.0623	-0.1571	0.2855	-0.7208	0.2667

Singular values:

1320.3140	397.3139	33.3097	1.4606	0.0118	0.0000	0.0000	0.0000	0.0000

Matrix V:

-0.3833	-0.5302	0.4795	-0.0682	0.1048	0.5584	-0.0862	0.0774	-0.0348
-0.3366	-0.4376	0.0414	-0.2007	-0.1434	-0.5694	0.4515	-0.1687	0.2759
-0.3584	-0.2643	-0.2330	0.5053	-0.1726	-0.3200	-0.4631	0.1788	-0.3369
-0.3492	0.0333	-0.4418	-0.1571	0.8106	-0.0000	0.0000	-0.0000	0.0000
-0.3718	0.1302	-0.4405	-0.2266	-0.4496	0.3548	-0.1455	-0.4920	0.0901
-0.3505	0.2925	-0.1224	0.2030	-0.1904	0.1830	0.2948	0.6496	0.3959
-0.3242	0.3648	0.2276	-0.4517	-0.1182	-0.1515	0.1268	0.1889	-0.6459
-0.2984	0.3771	0.3541	0.5696	0.1594	-0.0088	0.2421	-0.4835	-0.0317
-0.1848	0.2803	0.3642	-0.2322	0.0623	-0.2814	-0.6257	0.0069	0.4755

From the SVD decomposition, we realize that we can focus on the first three principal components.

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

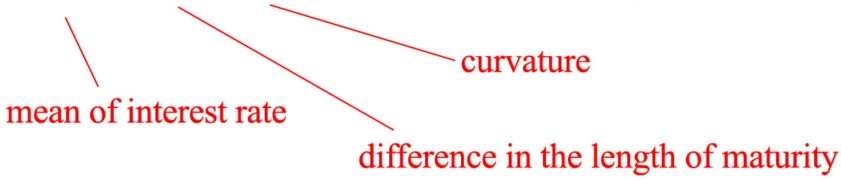
Singular values:

1320.3140	397.3139	33.3097	1.4606	0.0118	0.0000	0.0000	0.0000	0.0000

Matrix U:

-0.3833	-0.5302	0.4795	-0.0682	0.1048	0.4589	0.2131	0.0536	-0.2598
-0.3366	-0.4376	0.0414	-0.2007	-0.1434	-0.7669	-0.1969	0.0627	-0.0424
-0.3584	-0.2643	-0.2330	0.5053	-0.1726	0.1715	-0.0890	-0.1437	0.6359
-0.3492	0.0333	-0.4418	-0.1571	0.8106	-0.0000	-0.0000	-0.0000	-0.0000

-0.3718	0.1302	-0.4405	-0.2266	-0.4496	0.2544	-0.1926	-0.3331	-0.4296
-0.3505	0.2925	-0.1224	0.2030	-0.1904	-0.1896	0.7274	0.3519	-0.1014
-0.3242	0.3648	0.2276	-0.4517	-0.1182	0.1784	-0.2936	0.4664	0.3954
-0.2984	0.3771	0.3541	0.5696	0.1594	-0.1212	-0.4172	-0.0258	-0.3226
-0.1848	0.2803	0.3642	-0.2322	0.0623	-0.1571	0.2855	-0.7208	0.2667



As often, when we are shopping for a mortgage, we often ask what is the average interest rate these days and then look into the difference between the 7-year, 15-year, and 30-year mortgage. Interestingly, the US Treasure Bill yields from the data above demonstrate a structure that has a parallel similarity. The largest principal tries to average out all the interest rates with different maturities. The second principal accounts for the difference between the short and long term interest rate. (The third principal component is likely the curvature — a second-degree derivative.)

The example here is just a demonstration. I will not trade my day job with bond trading. It requires far more data analysis. We understand the relationship between the interesting rate and maturity well in our daily life. But when we are presented with unfamiliar raw data, PCA is very helpful to extract the principal components of your data. This may answer some questions on how to find a needle in a haystack.

Tips

Scale features before performing SVD.

$$x_j' = \frac{x_j - \bar{x}_j}{\sigma_j}$$

Say, we want to retain 99% variance, we can choose k such that

$$\underset{r \times r}{\begin{pmatrix} \sigma_1 & & 0 \\ & \ddots & \\ 0 & & \sigma_r \end{pmatrix}} \quad \frac{\sum_{i=1}^{k} \sigma_i}{\sum_{j=1}^{r} \sigma_j} \geq 0.99$$

Machine Learning Artificial Intelligence Mathematics