15 Singular Value Decomposition

For any high-dimensional data analysis, one's first thought should often be: *can I use an SVD?* The singular value decomposition is an invaluable analysis tool for dealing with large high-dimensional data. In many cases, data in high dimensions, most of the dimensions do not contribute to the structure of the data. But filtering these takes some care since it may not be clear which ones are important, if the importance may come from a combination of dimensions. The singular value decomposition can be viewed as a way of finding these important dimensions, and thus the key relationships in the data.

On the other hand, the SVD is often viewed as a numerical linear algebra operation that is done on a matrix. It decomposes a matrix down into three component matrices. These matrices have structure, being orthogonal or diagonal.

The goal of this note is to bridge these views and in particular to provide geometric intuition for the SVD.

Sometimes this geometric interpretation of the SVD is known as PCA (Principal Component Analysis).

Data. We will focus on a dataset $P \subset \mathbb{R}^d$ where P is a set of n "points." At the same time, we will think of P as a $n \times d$ matrix. Each row corresponds to a point, and each column corresponds to a dimension.

Then the goal will be to find a projection $\mu : \mathbb{R}^d \to \mathbb{R}^k$, where $k \ll d$ and in particular $\mathbb{R}^k \subset \mathbb{R}^d$, so that we minimize

$$\sum_{p \in P} (p - \mu(p))^2.$$

The SVD will precisely provide us this answer!

15.1 The SVD Operator

Here we document what the following operation in matlab does:

$$[U, S, V] = \texttt{svd}(P)$$

The backend of this (in almost any language) goes back to some very carefully optimized Fortran code as part of the LAPACK library.

First of all, no information is lost since we can simply recover the original data: $P = USV^T$.

But there is more structure lurking in these matrices. $S = \mathsf{diag}(\sigma_1, \sigma_2, \ldots, \sigma_r)$ where $r \leq d$ where r is the *rank* of P. That is, S is a *diagonal matrix* with only entries on the diagonal. These values are (generally) output in non-increasing order so $\sigma_1 \geq \sigma_2 \geq \ldots \geq s_r \geq 0$. They are known as the *singular values* of P. S has size $(n \times d)$.

Both U (size $n \times n$) and V (size $d \times d$) are *orthogonal matrices*. An orthogonal matrix is also a *rotation* matrix (more on this later), that can also allow mirror flips. They have the following properties:

- each column u_i has $\|u_i\| = 1$
- each pair of columns u_i, u_j have $\langle u_i, u_j \rangle = 0$
- Its transpose is its inverse $U^T = U^{-1}$, so $U^T U = I$.

Moreover the columns (and rows) of U form a d-dimensional orthogonal basis (usually not the original basis). That is, for any $p \in \mathbb{R}^d$ we can write

$$p = \sum_{i=1}^{t} a_i u_i$$

for some scalar $a_i = \langle p, u_i \rangle$. This is the ith coordinate in the new basis.

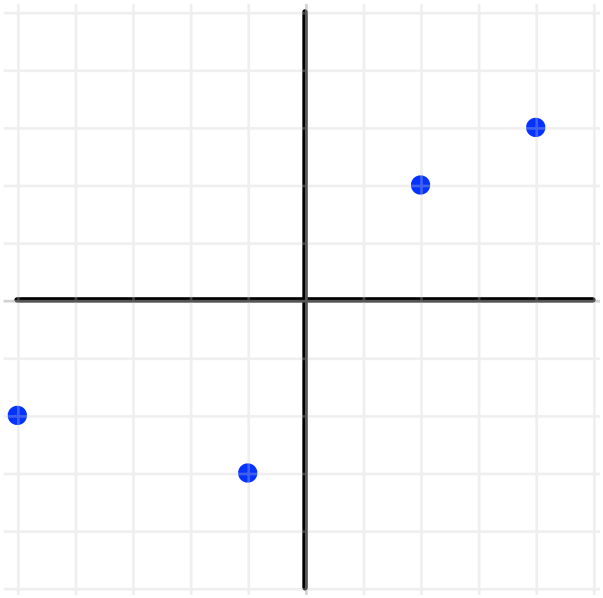
This implies that for any $x \in \mathbb{R}^d$ that $\|Ux\| = \|x\|$ (it can only rotate or flip).

Moreover the columns of $V = [v_1\ v_2\ \ldots\ v_d]$ are known as the *right singular vectors* and the columns of $U = [u_1\ u_2\ \ldots\ u_n]$ are known as the *left singular vectors*.

15.1.1 Example

Consider the set of $n = 4$ points in $\mathbb{R}^2$ $\{p_1 = (4,3), p_2 = (1,2), p_3 = (-1,-3), p_4 = (-4,2)\}$. Note, these are chosen so that average x- and average y-coordinate is 0; it is *centered*. We can equivalently write this as a the matrix

$$P = \begin{pmatrix} 4 & 3 \\ 2 & 2 \\ -1 & -3 \\ -5 & -2 \end{pmatrix}.$$



Then $[U, S, V] = \texttt{svd}(P)$ where

$$U = \begin{pmatrix} -0.6122 & 0.0523 & 0.0642 & 0.7864 \\ -0.3415 & 0.2026 & 0.8489 & -0.3487 \\ 0.3130 & -0.8070 & 0.4264 & 0.2625 \\ 0.6408 & 0.5522 & 0.3057 & 0.4371 \end{pmatrix},$$

$$S = \begin{pmatrix} 8.1655 & 0 \\ 0 & 2.3074 \\ 0 & 0 \\ 0 & 0 \end{pmatrix},$$

$$V = \begin{pmatrix} -0.8142 & -.5805 \\ -0.5805 & 0.8142 \end{pmatrix}.$$

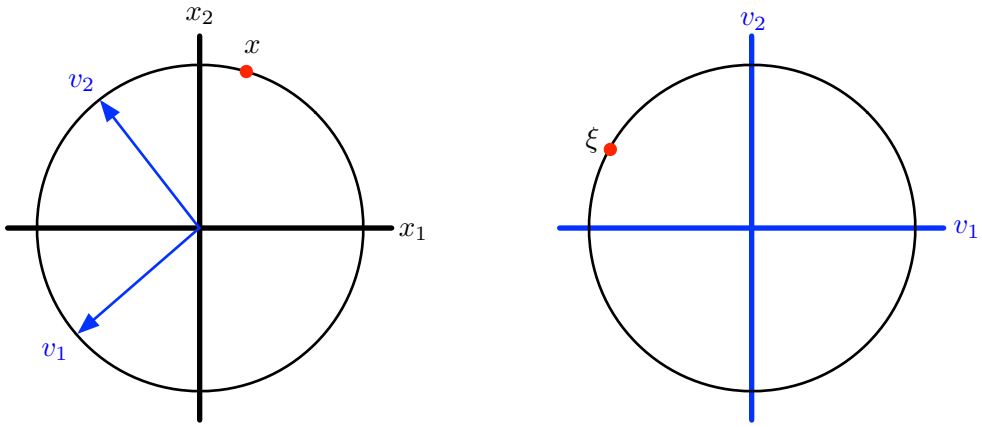
We will continue to use this example as we explain the geometry.

15.1.2 Geometry of the SVD

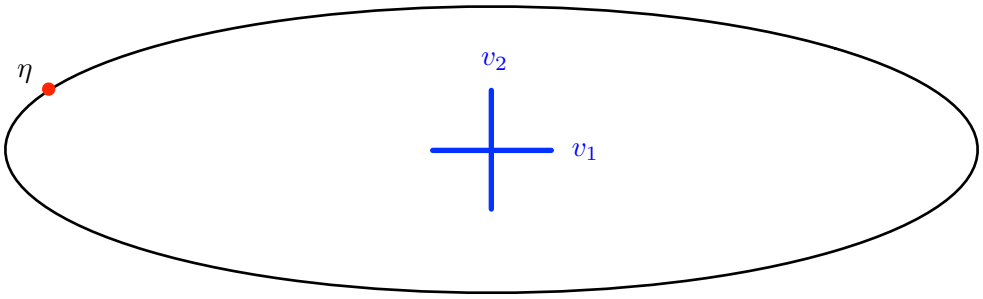
We will see how the matrix P transforms a circle in $\mathbb{R}^2$ to a two-dimensional ellipse the lives in $\mathbb{R}^4$. This ellipse will represent the size and magnitude the principal components.

We will start with an arbitrary point x such that $\|x\| = 1$ (so it is on the unit circle), and see what happens in $b = Px$. Specifically we will break down the process $b = USV^T x$ by examining $\xi = V^T x$, then $\eta = S\xi = SV^t x$ and then $b = U\eta = USV^T x$.

Step 1 ($\xi = V^T x$): Since V^T is orthogonal (and $x \in \mathbb{R}^2$) then it acts as a rotation. It puts x in the basis of V^T as a point ξ. Note that the orthogonal vectors v_1 and v_2 (of $V = [v_1, v_2]$) become the axis to describe ξ.



Step 2: ($\eta = S\xi$): Note that $\xi = (\xi_1, \xi_2)$ is still on a circle since it still has $\|\xi\| = 1$. The S matrix is a diagonal matrix, so it just scales the axis. Each ith axis is scaled according to σ_i. Specifically, $\eta_1 = \sigma_1 \xi_1$ and $\eta_2 = \sigma_2 \xi_2$, where ξ_1 and ξ_2 are coordinates in the basis along v_1 and v_2, respectively.



Step 3: ($b = U\eta$): We now apply another rotation. Notice that S had two rows that were all 0. This effectively scales η along two axes it did not know it had. But it sets these values to 0. So $\eta = (\eta_1, \eta_2, 0, 0) \in \mathbb{R}^4$.

Now we again use that U is a rotation (with possible mirror flips), but for points in $\mathbb{R}^4$. Each axis now represents the component along the direction of a point. Note that now $\|\eta\| = \|b\| = \|S\| = \|P\|$.

Unfortunately, this is harder to draw, but it looks like step 1, but in higher dimensions.

15.1.3 Principal Component Analysis

So how do we get this subspace that we project onto?

The vectors $V = [v_1, v_2, \ldots, v_d]$ are such that they describe the most *important* axis of the data in the following sense. The first right singular vector v_1 describes which direction has the most variance. The

variance is precisely described by σ_1. Then since $v_2, \ldots, v_d$ are each orthogonal to v_1, this implies that v_2 is the direction (after v_1 has been factored out) that has the most variance.

So the k-dimensional subspace of $\mathbb{R}^d$ is defined by basis $[v_1, v_2, \ldots, v_k]$, the first k-right singular vectors.

So how large should k be? The amount of squared "mass" captured by v_k is σ_k. So if σ_{k+1} is small, we do not need to keep v_{k+1}. It means there is little variation along that direction, and each other directions not yet captured. Or use "elbow" technique where the difference between $\sigma_k - \sigma_{k+1}$ is large.

- In many statistical and numerical datasets, often σ_k decay quickly. Usually for k not too large s_{k+1} is very close to 0.
- In many internet scale datasets (think Facebook graph), then typically σ_k decay slowly (they have a "heavy tail"). Often $\sum_{j=k+1}^{\infty} \geq 10\%$, even at the appropriate cut-off k. (This may (or may not) indicate that PCA is the wrong approach to finding core structure.)

So what do we know:

- V does "bookkeeping" of moving original basis to new one,
- S stretches it accordingly (along new basis), and
- U describes results with respect to actual data.

So to get the projection of the points P in the new subspace as $\mathbb{R}^k$ we create $P_k \in \mathbb{R}^k$ as

$$P_k = U_k S_k V_k^T$$

where $U_k = [u_1, u_2, \ldots, u_k]$, $S = \mathsf{diag}(\sigma_1, \sigma_2, \ldots, \sigma_k)$ and $V = [v_1, v_2, \ldots, v_k]$.

So the subspace is defined using just V_k. Then S_k describes the importance along each direction, and U_k relates it to the actual points.

Relationship to eigen-decomposition. We can write

$$P^T P V = (VSU^T)(USV^T)V = VS^2$$

so v_i are the eigenvectors of $P^T P$. Similarly

$$PP^T U = (USV^T)(VSU^T)U = US^2$$

so u_i are the eigenvectors of PP^T. Thus also the squared singular values σ_i^2 are eigenvalues of $P^T P$ and of PP^T.