

---

Singular value decomposition (SVD) generalizes matrix diagonalization to non-square matrices. It is powerful because it applies to **any**  $m \times n$  matrix.

First we give the recipe:

$$A = U\Sigma V^T. \tag{SVD}$$

Compute  $AA^T$ . This product is guaranteed to have  $m$  linearly independent eigenvectors by the **spectral theorem**. Collate those eigenvectors into the columns of a matrix. That matrix is  $U$ .

Then, compute  $A^T A$ , which is guaranteed to have  $n$  linearly independent eigenvectors that you can use to create  $V$ .

Next compute the non-zero eigenvalues  $\lambda_i$  of  $A^T A$ , take their square roots  $\sigma_i = \sqrt{\lambda_i}$ , and stick them into the diagonal of  $\Sigma$ .

Now,  $\Sigma$  will consist of  $\text{rank}(A)$   $\sigma_i$ 's on the diagonal and zeroes elsewhere. Pad more zeroes on the diagonal to form a  $\min(m, n) \times \min(m, n)$  square matrix. Then add the sufficient amount of rows/columns to form a  $m \times n$  matrix—obviously the rows/columns consist entirely of zeroes.

We are almost done. Conventionally, the singular values in  $\Sigma$  are ordered along the diagonal from greatest to least. This ordering of singular values should correspond to the ordering of eigenvectors in  $U$  and  $V$  (i.e.  $Av_i = \sigma_i u_i$ ).

Next we look at the intuition. The first perspective we'll look at is SVD as a way of encoding linear transformations. For example, imagine you had any square defined by the two vectors at its sides—say  $\vec{v}_1$  and  $\vec{v}_2$ . Left-multiplying these vectors by some  $A$  would then apply some linear transformation to the unit square. Here's the crucial claim of SVD: **any such linear transformation is equivalent to a rotation and a scaling, provided you are able to rotate the vector first**. More specifically the claim is

$$\begin{aligned} MR_1 &= R_2 \Sigma \Rightarrow \\ M &= R_2 \Sigma R_1^T \end{aligned}$$

which comes from right multiplication by  $R_1^T$  (which is orthogonal). This seems obvious for many transformations, such as rotations, scalings, and translations, but surprisingly it is true for shearing as well. Even more surprisingly, this holds true for transformations that change the dimension of the input vectors.

I like the other perspective as well. The SVD of  $A$  represents the action of  $A$  in the basis consisting of the eigenvectors of  $AA^T$ . It's probably useful to take a step back and look at the analog for matrix diagonalization.

---

### Matrix Diagonalization Analog

Consider the standard basis and the basis defined by the eigenvectors of a diagonalizable  $A$ , which we'll call  $B_1$  and  $B_2$  respectively. In  $B_2$ , the matrix  $\Lambda$  transforms the vector coordinates of the point  $P$  to  $P'$ . When  $P$  is expressed using the standard basis vectors in  $B_1$ , what matrix  $A$  transforms  $P$  to  $P'$ ?

To answer this question first take the vector coordinates representing  $P$  in the standard basis and plop them onto  $B_2$  to get some (probably totally) different point  $Q$ . It turns out that  $S^{-1}$ , where the columns of  $S$  consist of the eigenvectors of  $A$ , transforms  $Q$  to  $P$ . And as stated before  $\Lambda$  transforms  $P$  to  $P'$ .

We're not done though. We know how to take a set of coordinates representing  $P$  in  $B_1$  and turn them into a set of coordinates representing  $P'$  in  $B_2$ . We want to represent  $P'$  in  $B_1$ . For this reason we left-multiply by  $S$ —this results in  $P'$  going to some other different point  $W$  in  $B_2$ , but it turns out that if you take the vectors representing this  $W$  in  $B_2$  and plop them onto  $B_1$  you get  $P'$ .

All of this is to say that  $A = S\Lambda S^{-1}$  is the matrix that transforms  $P$  to  $P'$  in  $B_1$ , given that  $\Lambda$  transforms  $P$  to  $P'$  in  $B_2$ .

Now that we understand the matrix diagonalization analog, we can view SVD as a transformation in a new basis of vectors.