

Handout 05: Singular Value Decomposition

Today we are going to work with a particular type of matrix factorization called the singular value decomposition. Start by assuming that we have a matrix A with n rows and p columns such that $n \geq p$. The (thin) singular value decomposition, or SVD, is given by the matrix product:

$$A = UDV^t \quad (5.1)$$

With the following dimensions:

$$A \in \mathbb{R}^{n \times p} \quad (5.2)$$

$$U \in \mathbb{R}^{n \times p} \quad (5.3)$$

$$D \in \mathbb{R}^{p \times p} \quad (5.4)$$

$$V \in \mathbb{R}^{p \times p} \quad (5.5)$$

Furthermore, D is a diagonal matrix with non-negative entries along the diagonal ordered from the largest to the smallest value:

$$D = \begin{bmatrix} \sigma_1 & 0 & \cdots & 0 \\ 0 & \sigma_2 & \cdots & 0 \\ 0 & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \sigma_p \end{bmatrix}, \quad \sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_p \geq 0. \quad (5.6)$$

The values σ_k are called the *singular values* of the matrix A . Also, V is an orthogonal matrix such that (we showed in Handout 03 that this corresponds to a rotation):

$$V^t V = V V^t = I_p. \quad (5.7)$$

The matrix U is not square, so it cannot be completely orthogonal, but its columns are orthogonal to one another so we have:

$$U^t U = I_p. \quad (5.8)$$

The singular value decomposition exists for any matrix, and so we can use it without any assumptions on the matrix we are working with. This has important geometric implications: **any** linear function can be written as a rotation, a fixed scaling of the components, and another rotation.

SVD and the Normal Equations

If we take the SVD of the data matrix X , we have

$$X = UDV^t. \quad (5.9)$$

Plugging this into the ordinary least squares estimator gives:

$$\beta = (X^t X)^{-1} X^t y \quad (5.10)$$

$$= (V D^t U^t U D V^t)^{-1} V D^t U^t y \quad (5.11)$$

$$= (V D (U^t U) D V^t)^{-1} V D U^t y \quad (5.12)$$

$$= (V D I_p D V^t)^{-1} V D U^t y \quad (5.13)$$

$$= (V D^2 V^t)^{-1} V D U^t y \quad (5.14)$$

By taking the fact that a diagonal matrix is its own transpose and using that $U^t U$ is equal to the identity. Note that D^2 is just a matrix with the squared singular values along the diagonal.

Now, notice that the inverse of V is V^t , and vice-versa. Further, the inverse of D^2 is equal to a diagonal matrix with the inverse of the squared singular values along the diagonal (this exists if we assume that $\sigma_1 > 0$). Therefore:

$$(V D^2 V^t)^{-1} = (V^t)^{-1} D^{-2} V^{-1} = V D^{-2} V^t \quad (5.15)$$

And we can further simplify the equation for the ordinary least squares estimator:

$$\beta = (V D^2 V^t)^{-1} V D U^t y \quad (5.16)$$

$$= V D^{-2} V^t V D U^t y \quad (5.17)$$

$$= V D^{-2} D U^t y \quad (5.18)$$

$$= V D^{-1} U^t y. \quad (5.19)$$

This gives us a compact way to write the ordinary least squares estimator. It is also far more numerically stable to use this formula to compute the estimate β from a dataset. Most importantly, it will yield a lot of intuition for what makes some estimation tasks hard and motivate how we can (partially) address the most challenging regression problems.

SVD in R

In R, you can create the singular value decomposition of a matrix using the function `svd`. To see this, let's construct some simulated data:

```
set.seed(1)
n <- 1e4; p <- 4
X <- matrix(rnorm(n*p), ncol = p)
b <- c(1,2,3,4)
epsilon <- rnorm(n)
y <- X %*% b + epsilon
```

Now, we take the singular value decomposition of the matrix. I will also explicitly extract out and save the matrices U and V as well as the singular values *sigma*:

```
svd_output <- svd(X)
U <- svd_output[["u"]]
V <- svd_output[["v"]]
sigma <- svd_output[["d"]]
```

Now, let's compute the ordinary least square matrix with this data:

```
beta <- V %*% diag(1 / sigma) %*% t(U) %*% y
beta
```

```
      [,1]
[1,] 0.9870134
[2,] 1.9876739
[3,] 3.0045489
[4,] 4.0102080
```

We can verify that this is equivalent to our old form of the estimator by:

```
solve(t(X) %*% X) %*% t(X) %*% y
```

```
      [,1]
[1,] 0.9870134
[2,] 1.9876739
[3,] 3.0045489
[4,] 4.0102080
```

Notice that both are close to the value of b in the simulation.

LAB QUESTIONS

1. I showed you how to get a nice equation for β in the ordinary least squares equation. Using the SVD of X , compute a compact formula for the values $\hat{y} = X\beta$.
2. We glossed over the case where one or more of the singular values is equal to zero. In this question I will show you why we cannot deal with this case in the construction of β . Let V_p denote the last column of V (these columns are called the *right singular vectors*). Argue that:

$$V^t V_p = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix} \quad (5.20)$$

Now, assume that $\sigma_p = 0$. Show that (Hint: expand X with the SVD):

$$X V_p = 0. \quad (5.21)$$

3. Assume that we have a potential candidate β for the regression vector. Show that the fitted values \hat{y} :

$$\hat{y} = X\beta = X(\beta + a \cdot V_p), \quad \forall a \in \mathbb{R}. \quad (5.22)$$

Explain why this implies that we cannot uniquely determine a value for β according the minimization of the loss function on the training data when $\sigma_1 = 0$.

4. Let X be a matrix with SVD equal to UDV^t and w be a p -dimensional vector with Euclidean norm equal to one:

$$\|w\|_2^2 = w^t w = \sum_k w_k^2 = 1. \quad (5.23)$$

It is generally true that we can write the vector w as a weighted sum of the columns of V :

$$w = \sum_k a_k \cdot V_k. \quad (5.24)$$

I want you to show that $\sum_k a_k^2 = 1$. This is straightforward *assuming that you approach the problem in a particular way*. Start by writing out $\|w\|_2^2$ as an inner product and expanding in the basis of V :

$$1 = \|w\|_2^2 = w^t w = \left(\sum_k a_k \cdot V_k \right)^t \left(\sum_k a_k \cdot V_k \right) \quad (5.25)$$

$$= \left(\sum_k a_k \cdot V_k^t \right) \left(\sum_k a_k \cdot V_k \right) \quad (5.26)$$

Then, take the cross terms to write this as a double sum and simplify the result.

5. Let X be a matrix with SVD equal to UDV^t and w be a p -dimensional vector. Show that:

$$||Xw||_2^2 = ||DV^t w||_2^2 \quad (5.27)$$

In other words, the matrix U does not effect the size of the product Xw .

6. Let X be a matrix with SVD equal to UDV^t and w be a p -dimensional vector that we will write as:

$$w = \sum_k a_k \cdot V_k. \quad (5.28)$$

Show that:

$$V^t w = \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_p \end{bmatrix}. \quad (5.29)$$

And therefore, we have:

$$DV^t w = \begin{bmatrix} \sigma_1 \cdot a_1 \\ \sigma_2 \cdot a_2 \\ \vdots \\ \sigma_p \cdot a_p \end{bmatrix} \quad (5.30)$$

7. Using the set-up from the previous question, show that:

$$||Xw||_2^2 = \sum_k a_k^2 \sigma_k^2. \quad (5.31)$$

8. Let X be a matrix with SVD equal to UDV^t . Consider the ℓ_2 -ball given by all vectors with a Euclidean norm of 1:

$$B_p = \{v \in \mathbb{R}^p, \quad \text{s.t. } ||v||_2 = 1\}. \quad (5.32)$$

Argue that:

$$\min_{v \in B_p} \{||Xv||_2^2\} = \sigma_p^2 \quad (5.33)$$

And

$$\max_{v \in B_p} \{||Xv||_2^2\} = \sigma_1^2. \quad (5.34)$$

9. Finally, complete the questions in the file `class05.Rmd`.