Discussion Board

For many SVD it is the most important matrix decomposition in linear algebra, because it reveals the detailed inner information about the matrix. You have learned that for a symmetric matrix the singular values and the eigenvalues are the same. How about the singular vectors and the eigenvectors?

Try to use a random A^T * A matrix in python to answer this question.

What is a symmetric matrix?

A symmetric matrix is one that is equal to its transpose $A = A^T$

Symmetric matrices have some very nice properties:

1. All their eigenvalues will be real numbers (not complex numbers)
2. Their eigenvectors are always orthogonal or can be made orthogonal. Orthogonal eigenvectors of a matrix are eigenvectors that happen to be orthogonal to each other (in other words, the eigenvectors are at a 90 degrees angle to each other) $Av = \lambda v$
3. They are always diagonalizable . Q is the matrix of orthogonal vectors. ^ is the diagonal matrix of eigenvalues. $A = Q \wedge Q^T$

What are singular values? How are they related with eigenvalues?

Singular values are the diagonal enteries of the matrix $\sum$ in the SVD formula

Singular values are the square root of the eigenvalues in matrix $A^T A$ or AA^T

Remember, eigenvalues tell us how a matrix can transform (stretches or shrinks) its own eigenvectors without changing the direction of the eigenvector. Eigenvalues can exist only with square matrices.

On the other hand, singular values tell you how much the matrix stretches space along orthogonal directions, even if the matrix does not rotate or distort vectors. Unlike eigenvalues, singular values always exist.

Purpose of a symmetric matrix and singular values

Symmetric matrices and singular values are used in singular value decompositions (SVD)

For any matrix A, the SVD is: $A = U\Sigma V^T$

U = left singular vectors (from AA^T . This matrix gives eigenvectors which form U)
$\sum$ = diagonal matrix of singular values
V = right singular vectors (from $A^T A$. This matrix gives eigenvectors which form V)

Another question, how is the MP pseudoinverse matrix computed by using SVD? What were the key points in the algorithm?.

Question 1

Find the Singular Value Decomposition for the matrix A

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

Recall Singular Value Decomposition formula $A = UDV^T$

Which one of the three matrix products represents the SVD of the matrix A?

✗ $\begin{bmatrix} -\frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \\ -\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{bmatrix} \begin{bmatrix} \sqrt{2} & 0 \\ 0 & \sqrt{2} \end{bmatrix} \begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix}$

○ $\begin{bmatrix} \frac{1}{\sqrt{2}} & 0 \\ \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{bmatrix} \begin{bmatrix} \sqrt{2} & 0 \\ 0 & \sqrt{2} \end{bmatrix} \begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix}$

○ $\begin{bmatrix} \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & \sqrt{2} \end{bmatrix} \begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix}$

Singular value decomposition (SVD) is a matrix factorization technique used in linear algebra, data science, and machine learning. It allows you to break any m x n matrix A (square or not) into three special matrices. SVD tells you how any matrix acts by breaking it into 3 transformations.

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

A The original matrix size mxn

U mxm orthogonal matrix (columns are called **left** singular vectors). Rotates the output space (image of A).

$\sum$ a mxn diagonal matrix. Entries are singular values which are always non-negative and sorted in decending order. Scaling the matrix along the axis. Tells us how much the matrix stretches or compresses.

V^T is the transpose of an nxn orthogonal matrix. Columns of V are **right** singular vectors. Rotates the matrix space.

Step 1: Compute $A^T A$

Note: To make a transpose of a matrix the rows become columns and columns become rows

$$\begin{bmatrix} 1 & 1 \\ -1 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & -1 \\ 1 & 1 \end{bmatrix} = \begin{bmatrix} 1\cdot1+1\cdot1 & 1\cdot-1+1\cdot1 \\ -1\cdot1+1\cdot1 & -1\cdot-1+1\cdot1 \end{bmatrix} = \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix}$$

Step 2: Find the eigenvalues and eigenvectors of $A^T A$

Step 2.1: Solving for eigenvalues

$$A - \lambda I = \begin{bmatrix} 2-\lambda & 0 \\ 0 & 2-\lambda \end{bmatrix}$$

$$det(A - \lambda I) = (2-\lambda)(2-\lambda) = 4 - 4\lambda + \lambda^2 = \lambda^2 - 4\lambda + 4$$

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

$$\lambda = 2$$

Step 2.2: Find eigenvectors for each eigenvalue

$$A - 2I = \begin{bmatrix} 2-2 & 0 \\ 0 & 2-2 \end{bmatrix} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}$$

← A zero matrix means every nen-zero vector is a eigenvector

$$V_1 = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \quad V_2 = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

Step 3: Find singular values

Singular values σ_i are the square roots of the eigenvalues of $A^T A$

Singular values are like eigenvalues they **only** stretche or compress the matrix. It does not rotate or change the direction of the matrix. The main difference is that eigenvalues are only applied to square matrices while a singular value can be applied to any matrix, square or rectangular.

$$\sigma_1 = \sigma_2 = \sqrt{2} \qquad \Sigma = \begin{bmatrix} \sqrt{2} & 0 \\ 0 & \sqrt{2} \end{bmatrix}$$

Step 4: Find V

we need the eigenvalues of $A^T A$ to get the columns of V

$$V = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix}$$

Eigenvector can be any value since every non-zero matrix can be a eigen vector

Step 5: find U

$$U = AV\Sigma^{-1} = \underset{A}{\begin{bmatrix} 1 & -1 \\ 1 & 1 \end{bmatrix}} \cdot \underset{\Sigma^{-1}}{\begin{bmatrix} \frac{1}{\sqrt{2}} & 0 \\ 0 & \frac{1}{\sqrt{2}} \end{bmatrix}} = \begin{bmatrix} \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{bmatrix}$$

V can have any eigenvector so it can be I

Question 2

Compute the left pseudoinverse matrix for matrix A when you have SVD decomposition for the matrix A

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

SVD factorization is given with the formula

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

The left pseudoinverse matrix A^+ is defined as $A^+ * A = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$, select the correct answer below

- [x] $A^+ = \begin{bmatrix} 0.5 & -0.5 & -0.5 \\ 0.5 & 0.5 & -0.5 \end{bmatrix}$
- [ ] $A^+ = \begin{bmatrix} 1 & -0.5 & 1 \\ 1 & 0.5 & -1 \end{bmatrix}$
- [ ] $A^+ = \begin{bmatrix} -0.5 & -0.5 & 0.5 \\ -0.5 & -0.5 & -0.5 \end{bmatrix}$

The method to compress an image in image processing is using convolution matrix where the number of outputs is reduced by replacing the matrix multiplication with much smaller kernel matrix. This is illustrated on the figure below. In this example a black and white photograph is scanned as a rectangular array of pixels and then stored as image matrix on the left by assigning each pixel a numerical value in accordance with its gray level, (0 = white to 255 = black), then the entries in the matrix are integers between 0 and 255.Then multiplication is employed with much smaller kernel matrix in the middle.

$$\underbrace{\begin{array}{|c|c|c|c|c|c|c|}\hline 0&0&0&0&0&0&0\\ \hline 0&0&21&0&0&0&0\\ \hline 0&85&71&0&0&0&0\\ \hline 0&250&231&127&63&3&0\\ \hline 0&250&252&250&209&56&0\\ \hline 0&250&252&250&250&83&0\\ \hline 0&0&0&0&0&0&0\\ \hline\end{array}}_{\textbf{Image}} \circ \underbrace{\begin{array}{|c|c|c|}\hline 0&0&1\\ \hline 0&1&0\\ \hline 1&0&0\\ \hline\end{array}}_{\textbf{Kernel}} \longrightarrow \underbrace{\begin{array}{|c|c|c|c|c|}\hline 0&106&&&\\ \hline &&&&\\ \hline &&&&\\ \hline &&&&\\ \hline &&&&\\ \hline\end{array}}_{\textbf{Feature map}}$$

Boxes in red and blue frames indicate how the upper left element is formed by applying the kernel to the corresponding upper left region of the input matrix. The elements of much smaller "feature map" matrix on the right can be calculated from:

$$\text{FeatureMap(m,n)} = \sum_{i=1}^{m} \sum_{j=1}^{n} Image(m+i-1, n+j-1)\ Kernel(i,j)$$

Calculate the entry in the FeatureMap(1,3) in the figure?

Boxes in red and blue frames indicate how the upper left element is formed by applying the kernel to the corresponding upper left region of the input matrix. The elements of much smaller "feature map" matrix on the right can be calculated from:

$$\text{FeatureMap(m,n)} = \sum_{i=1}^{m} \sum_{j=1}^{n} Image(m+i-1, n+j-1)\ Kernel(i,j)$$

Calculate the entry in the FeatureMap(1,3) in the figure?

- ○ 92
- ○ 21
- ○ 71