Singular Value Decomposition

Johan Astborg

Computational Programming with Python

Project Summary, 6/4 2015

Outline

1. Motivation
2. Theory
3. The project

Motivation

Matrix decomposition

Consider the matrix A

$$A = \begin{bmatrix} 4 & 11 & 14 \\ 8 & 7 & -2 \end{bmatrix}. \tag{1}$$

Since A is *singular*, the following (common eigen) decomposition

$$A = VDV^T \tag{2}$$

is inapplicable[1].

[1] Clearly, since A is non-invertible

Motivation

Matrix decomposition cont.

Instead there is a non-trivial possibility, using the absolute values of the eigenvalues instead

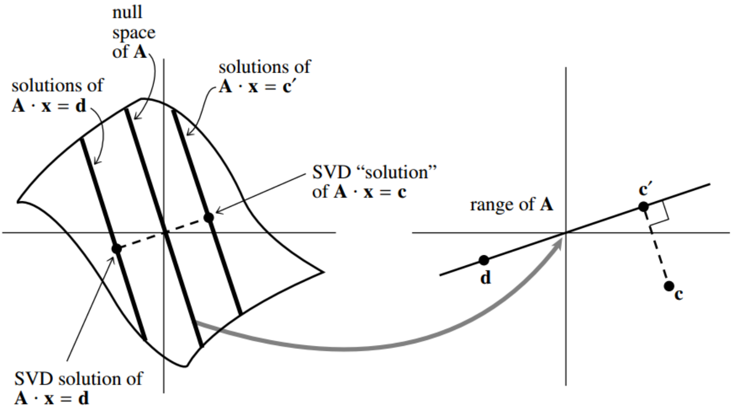
$$A = UDV^*. \tag{3}$$

These are the *singular values* of A, where A^* is *conjugate transpose*.

Motivation

Geometric representation

The following figure will learn you SVD for life



Motivation

The singular values of A

For the curious mind, the *singular* values of A is as follows

$$\begin{aligned} \sigma_1 &= 6\sqrt{10} \\ \sigma_2 &= 3\sqrt{10}. \\ \sigma_3 &= 0 \end{aligned}$$

They correspond to $||Av_i|| = \sigma_i$

Motivation

The singular values of A cont.

Explicitly for σ_2 we have

$$A\mathbf{v}_2 = \begin{bmatrix} 4 & 11 & 14 \\ 8 & 7 & -2 \end{bmatrix} \begin{bmatrix} -2/3 \\ -1/3 \\ 2/3 \end{bmatrix} = \begin{bmatrix} 3 \\ -9 \end{bmatrix}$$

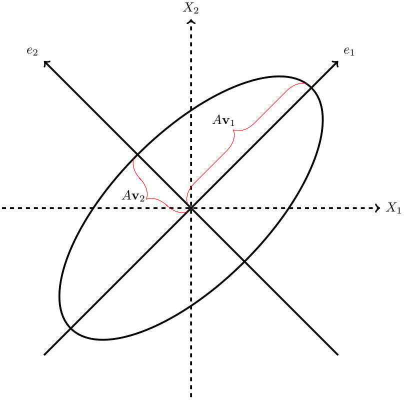
and then we take the norm

$$\implies \sigma_i = ||Av_i|| = 3\sqrt{10}$$

Motivation

Geometric interpretation of singular values

The two first singular values of A corresponds to the length of the semiaxis of an ellipse



Definition

Singular Value Decomposition

The decomposition of A involves an $m \times n$ matrix Σ of the form

$$\Sigma = \begin{bmatrix} D & 0 \\ 0 & 0 \end{bmatrix}. \tag{4}$$

where D is an $r \times r$ diagonal matrix (r is the rank of A)

Theorem

Singular Value Decomposition

U is $m \times n$, Σ is $n \times n$ and V is $n \times n$,

$$A = U \Sigma V^* \tag{5}$$

where U, V are orthogonal and its positive diagonal entries is called the *SVD* of A.

Proof

Singular Value Decomposition

catch me in the break if you are interested
... since V is orthogonal matrix, $U\Sigma V^* = AUV^T = A$

The Project
Singular Value Decomposition

Implement this in Python given constraints and a set of tasks.

The Project

Singular Value Decomposition

In theory we can follow the following steps

1. Find the orthogonal diagonalization of $A^T A$
2. Set up V and Σ
3. Construct U
4. Check singular values against eigenvalues ($||Av_i|| = \sigma_i$)

Reality check

Singular Value Decomposition

But numerical linear algebra is reality

...which means *IEEE-754* and 64-bit FPUs.

Reality check

Singular Value Decomposition

SVD is considered a robust and numerically stable method, but a *rank deficient*[2] matrix will not produce exact zeroes for singular values. This is mainly due to finite numerical precision (IEEE-754 etc).

[2] we have to guess more or less

The Project

Requirements

Requirements for the project

1. Givens rotations (to bring back c to the x-axis)
2. Use Python
3. Visualize the computation (decomposition)
4. Have fun

The Project

Implementation

How it was done

1. Paper and pen and a good reference[3]
2. A few keystrokes, i.e. Python is a reduced *Lisp*
3. Emacs, git and Latex
4. MATLAB (as a reference)

[3] Numerical Recipes In C, etc.

The Project

Challenges

Well, **software development** and **numerical analysis**

- will test your skills (to some extent at least)
- will learn you something new

The Project

Challenges

Now, just one thing remains...

The Project
Challenges

Go home and decompose some matrices!

The Project

Checkout the source

Checkout the project[4]

`http://github.com/joastbg/python-svd`

[4] there will be updates

The Project

Thanks and have a nice summer

Thanks!