# Single Value Decomposition

## Introduction and applications in software development

PID_00266369

Marcel Serra Julià

Universitat Oberta de Catalunya

**Marcel Serra Julià**

I got my degree in Mathematics at the Universitat Autònoma de Barcelona (UAB) in 2009 and did my eHealth Masters at the McMaster University (Canada) in 2011. While I did my graduate training I was also a teaching assistant in the Department of Computer Sciences at the McMaster University. In 2012, I developed my professional career as a Software Developer working for several companies in the digital sector in the Barcelona area, including Mascoteros (a marketplace for the pet industry) and YaEncontre (a real estate website). In these companies I specialized in web development with Symfony, a popular PHP framework based on MVC architecture. The rapidly changing digital industry was a catalyst which led me to look to new technologies such as Big Data, Computer Vision and Computational Modeling. As a result of my expertise and motivation for research I am currently working as a Data Manager at ICTA (Institut de Ciència i Tecnologia Ambiental, UAB).

The assignment and creation of this UOC Learning Resource have been coordinated by the lecturer: Cristina Cano Bastidas (2020)

First edition: February 2020

Av. Tibidabo, 39-43, 08035 Barcelona
Publishing: FUOC

## Introduction

In this module we will introduce two powerful methodologies for data analysis that rely on concepts in linear algebra we have already presented in this course. Both of them make an intensive use of eigenvalues, eigenvectors and the diagonalization theorem. We will first present Principal Component Analysis (PCA) and then Single Value Decomposition (SVD).

Working with large datasets formed by many variables can lead to the problem known as the curse of dimensionality. It refers to the challenge of interpreting and extracting knowledge from a large dataset. Although the amount of information grows as the dataset increases, the dataset also becomes more complex. The increase in complexity brings about other challenges, as for example the difficulty of extracting conclusions from plots. The two methods presented in this module, PCA and SVD, can help us to tackle scenarios with large datasets and overcome the effect of the curse of dimensionality.

PCA is a method used in statistics to study large datasets. The mathematical idea behind it is to apply a linear mapping to a dataset with the objective of transforming it in such a way that the new dataset has a diagonal covariance matrix. Given a dataset with $n$ samples with $m$ variables measured for each sample, Principal Component Analysis will transform the data into a new set of variables satisfying two conditions:

1- The covariance among the new variables is reduced
2- The number of new variables is also reduced

The second method presented in this module will be Single Value Decomposition (SVD). This method has many applications in data analysis and it has a relevant role in image compression. We will present several examples of data compression.

From a mathematical point of view, Singular Value Decomposition can be seen as an extension of the Diagonalization theorem. When applied to a matrix A, the Diagonalization theorem has some limitations. It can only be applied to a square matrix and it requires that the number of linearly independent eigenvectors of A is equal to the dimension of A. The SVD theorem has some advantages, one being that it relaxes these requirements on the matrix A and only needs to have an $m \times n$ matrix with real entries, and another being that the singular vectors (which are the equivalent to the eigenvectors) form an orthogonal base.

Since images are represented by matrices with $m \times n$ real and positive entries, the application of the SVD theorem on them is direct. In order to introduce the student to the manipulation of images represented by matrices, we propose the following example.

Let's suppose that the matrix A represents an image.

$$A = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 2 & 4 & 6 & 8 & 10 \\ 3 & 6 & 9 & 12 & 15 \\ 4 & 8 & 12 & 16 & 20 \\ 5 & 10 & 15 & 20 & 25 \end{pmatrix}$$

This image requires $5 \cdot 5 = 25$ entries to store the entire image. However, notice that this matrix can be obtained by the following scalar product of vectors

$$\begin{pmatrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{pmatrix} \begin{pmatrix} 1 & 2 & 3 & 4 & 5 \end{pmatrix} = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 2 & 4 & 6 & 8 & 10 \\ 3 & 6 & 9 & 12 & 15 \\ 4 & 8 & 12 & 16 & 20 \\ 5 & 10 & 15 & 20 & 25 \end{pmatrix}$$

These two vectors are represented by $5 + 5 = 10$ entries, which represents a 40% compression. With an original matrix of $200 \cdot 200 = 40000$ entries the reduction would be down to 400 entries, which uses 1% of the original space to store the same amount of information. This example provides some insight into how an image can be compressed and the fact that less space is required to store the same amount of information. This idea, which is easy to understand when applied to images, can also be translated to any other dataset represented by a matrix of real entries.

## Objectives

The main objectives of this module are: to understand the two methods presented (PCA and SVD); and to apply them in different scenarios to solve problems with real or realistic data.

1. To understand the curse of dimensionality problem when working with large datasets.
2. To apply the concepts related to linear algebra learned in the previous modules to practical problems.
3. To solve problems by applying the Principal Component Analysis method to real or realistic data.
4. To solve problems by using the Single Value Decomposition method. In particular, this method will be applied to image compression.
5. To practice and learn how to use the R programming language to solve problems with large datasets.