Project 2 NLA

Eddie Conti

December 2023

1 Application of SVD factorization

The aim of the project is to discuss some relevants applications of SVD factorization: LS problem, graphic compression and PCA.

1.1 LS problem

The method of least squares is a standard approach in regression analysis to approximate the solution by minimizing the sum of the squares of the residuals, i.e. the difference between and observed value and the one computed by the model, made in the results of each individual equation. The main application is in data fitting where we want to find x such that minimize $||Ax - y||_2$. In matrix form it can be formalized using the Vandermonde matrix:

$$\begin{pmatrix} 1 & x_0 & x_0^2 & \dots & x_0^n \\ 1 & x_1 & x_1^2 & \dots & x_1^n \\ 1 & x_2 & x_2^2 & \dots & x_2^n \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_m & x_m^2 & \dots & x_m^n \end{pmatrix} \cdot \begin{pmatrix} a_0 \\ a_1 \\ \vdots \\ a_n \end{pmatrix} = \begin{pmatrix} y_0 \\ y_1 \\ \vdots \\ y_m \end{pmatrix}.$$

In this case we are aiming at fiding a polynomial $p(x) = a_0 + a_1 x + a_2 x^2 + \dots + a_n x^n$, which satisfies $p(x_0) = y_0, \dots, p(x_m) = y_m$. The Vandermonde matrix A can be seen as a linear transformation. If we consider $Im(A)$, then our goal is to find the projection of y onto this space. By using this idea, we can deduce the normal equation

$$A^T Ax = A^T y. \tag{1}$$

The term $A^T A$ is clearly symmetric but numerically unstable, then to solve this equation one can use both QR factorization or SVD. In the case of QR factorization (1) becomes

$$Rx = Q^T y$$

which is an upper triangular system. On the other hand, with SVD factorization (1) becomes

$$\Sigma V^T x = U^T y.$$

However, if $rank(A) = r < n$ (typically $m >> n$) the solution is clearly not unique: $dim(ker(A)) = n - r$ and so if x is a solution to (1), then $x + x_0$ is also, if $x_0 \in ker(A)$.

However, one idea to get the uniqueness is to look for the solution with minimum l_2-norm. In the first part of the project we solve two problem of polynomial fitting both with QR and SVD. For the QR factorization we followed the procedure explained in pr4 that aims at minimizing the norm too. The solutions we obtain coincide for both methods and datasets.

1.2 Graphic compression

The SVD factorization has the important property of giving the best low rank approximation matrix with respect to the Frobenius and/or the 2-norm to a given matrix. The idea behind is that the singular values of Σ are ordered such that $s_1 \geq s_2 \geq \ldots \geq s_n \geq 0$. Since U and V^T are orthogonal (and so invertible) matrix of dimension, respectively, m and n, we have to work on Σ if we want to produce an approximation of the original matrix with a lower rank. Hence, the idea is to consider the first k singular values, and of course to adjust matrix U and V^T in order to be able to perform matrix product, to obtain the best approximation. Let us denote with A the original matrix and with $A_k = U_k \Sigma_k V_k^T$ where U_k corresponds to the first k columns of U, V_k^T to the first k rows of V^T and Σ_k corresponds to the first k singular values. In this notation, A_k has rank k and minimizes

$$||A - A_k||_F, \qquad ||A - A_k||_2.$$

By definition of Frobenius norm and l_2 norm and since orthogonal transformations are invariant w.r.t these norms,

$$||A - A_K||_F = \sqrt{s_{k+1}^2 + \ldots + s_n^2}, \qquad ||A - A_K||_2 = s_{k+1}.$$

Therefore, the relative error is either

$$\frac{s_{k+1}^2 + \ldots + s_n^2}{s_1^2 + \ldots + s_n^2}, \qquad \frac{s_{k+1}}{s_1}.$$

In the project we used three different images: one picture in grey scale, one simple text image and one elaborated and colored image. We used different ranks: $[5, 10, 20, 50, 100]$ and saved the name accordingly to the Frobenius norm captured in the compressed image. The filename, denoted as *compressed_rank_%*, provides insights into the efficiency of our file compression. Inspecting both the filename and the displayed images, it becomes evident that a substantial reduction in file size is achievable even with a lower rank, such as 5, resulting in a 99% approximation accuracy.
On the other hand, achieving a higher level of approximation for the first image demands a correspondingly higher rank. Notably, the last image, a colorful and tremendously detailed painting by Pollock, requires a rank of 100 to get an approximation of "only" 93%. This observation suggests that the percentage of approximation, determined by the rank, mirrors the complexity of the original image.

1.2.1 Results on SVD

As we stated previously the SVD factorization provides the best k rank approximation of the matrix A with respect to the Frobenius norm and $2-$norm. First of all, let us write the matrix A as

$$A = \sum_{i=1}^{r} s_i u_i v_i^T$$

where r is the rank of the matrix. Now, let us denote

$$A_k = \sum_{i=1}^{k} s_i u_i v_i^T \quad k \in \{0, 1, \dots, r\}.$$

Since the vectors v_i and u_i are orthonormal then A_k has rank k. Let us prove the result for the Frobenius norm.

Theorem 1.1. *Let $A \in \mathbb{R}^{m \times n}$ a real matrix. The matrix A_k is such that $||A - B_k||_F$ is minimized over the set of a matrices of rank up to k.*

Proof. First of all, with our notation and by definition of Frobenius norm, is immediate to note that

$$||A - A_K||_F = \sum_{i=k+1}^{n} s_i^2.$$

We have to prove that

$$\sum_{i=k+1}^{n} s_i^2 \leq ||A - B_K||_F.$$

Using the triangular inequality, splitting $A = A' + A''$ and denoting A'_k, A''_K respectively the rank k approximation to A' and A'', then for any $i, j \geq 1$

$$\begin{aligned} s_i(A') + s_j(A'') &= s_1(A' - A'_{i-1}) + s_1(A'' - A''_{j-1}) \\ &\geq s_1(A - A'_{i-1} - A''_{j-1}) \\ &\geq s_1(A - A_{i+j-2}) \\ &= s_{i+j-1}(A). \end{aligned}$$

In our case, since $s_{k+1}(B_k) = 0$, if we use as $A' = A - B_k$ and $A'' = B_k$, we conclude that for $i \geq 1$ and $j = k + 1$

$$s_i(A - B_k) \geq s_{k+i}(A).$$

To conclude,

$$\|A - B_k\|_F^2 = \sum_{i=1}^{n} s_i(A - B_k)^2 \geq \sum_{i=k+1}^{n} s_i(A)^2 = \|A - A_k\|_F^2.$$

□

Let us now prove the result for the 2−norm. Under the same notation, we firstly note that

$$\|A - A_k\|_2 = \left\| \sum_{i=1}^{n} s_i u_i v_i^\top - \sum_{i=1}^{k} s_i u_i v_i^\top \right\|_2 = \left\| \sum_{i=k+1}^{n} s_i u_i v_i^\top \right\|_2 = s_{k+1}$$

because u_i, v_i^T are othornomal vectors. We have to prove that if $B_k = XY^T$ then $s_{k+1} \leq ||A - B_K||_2$. Since Y has k columns there must exists a linear combination of the first $k+1$ columns of V

$$w = a_1 v_1 + \ldots + a_{k+1} v_{k+1}$$

such that $Y^T w = 0$. We can ask w to be unitary, so that $a_1^2 + \ldots a_{k+1}^2 = 1$, therefore

$$\|A - B_k\|_2^2 \geq \|(A - B_k)w\|_2^2 = \|Aw\|_2^2 = a_1^2 s_1^2 + \cdots + a_{k+1}^2 s_{k+1}^2 \geq s_{k+1}^2,$$

because $s_1 \geq \ldots \geq s_{k+1} \geq 0$. This concludes the proof.

1.3 PCA

Principal component analysis is a technique to detect the main components of a data set in order to reduce the dimensionality. The general idea is consider a linear combination of the original variables, called *principal components*, so that maximizes the variance. In other words, if we aim at maximizing the variance, in a sense we are capturing the range of the data and the space where they live. In this way, using a few components yields to a relevant reduction and approximation since the most important features are encoded in these few components. Consistently with what we stated and discussed in section 1.2, the SVD factorization works in this direction.
The third task of our project involves using this technique on two different datasets and discuss the number of principal components needed to explain the datasets. In our case we used the scree plot method: the idea is to find the "elbow" of the graph where the eigenvalues seem to level off. To be more clear, the graph shows the relation between the number of components and the eigenvalues associated and ordered. Once we detect the "elbow", the components to the left of this point should be retained as significant. The technique we are employing to determine the optimal number of factors is the scree plot method. It is important to note that, like all methods for selecting the appropriate number of factors, this technique is not universally applicable, but we widely adopt it in practice.

For the file *example.dat* which consists of 16 observations and 4 variables we performed a PCA on both covariance and correlation matrix. For the correlation matrix we denoted the values of example.dat transpose with the matrix A. Then, we substracted the mean of each row and performed the svd following the procedure described. We computed the variance explained by each component, the standard deviation and the new coordinates of the dataset and plotted the dataset with the first 2 components according to the scree plot rule. For the correlation matrix, we followed the same procedure described above but we divided also by the standard deviation of each row. The results are similar, but in this case the scree plot is not useful since there is no possibility to find the elbow of the graph. As a consequence, we used the Kaiser rule: since an eigenvalue of 1 means that the factor contains the same amount of information as a single variable, then you consider the components whose eigenvalue is greater or equal than 1. Even in this case, then, only 2 components are sufficient to explain the dataset.
To conclude we performed the a PCA analysis on the covariance matrix on the dataset RCsGoff.csv. What is possible to observe is that the elbow occurs at $n = 2$ and so we can confirm the decision of researcher to describe the data with the first two principal components.