1 20th of September 2018 — F. Poloni

1.1 A warm up

Do you recall?

Let $x, y \in \mathbb{R}^n$. The product between those two vectors is computed as follows $x^T y = \sum_{i=1}^{n} x_i y_i$ and $x^T y \in \mathbb{R}$.

Let $x \in \mathbb{R}^n$ and $\lambda \in \mathbb{R}$ we call **multiple** of vector x the following: $\lambda x = x\lambda = \begin{pmatrix} \lambda x_1 \\ \vdots \\ \lambda x_n \end{pmatrix}$.

Given a matrix $A \in M(n, m, \mathbb{R})$ and a vector $b \in \mathbb{R}^m$ the **matrix-vector product** $Ab = v \in \mathbb{R}^n$ is computed as follows:

$$v = Ab = \begin{pmatrix} A_1 b \\ A_2 b \\ \vdots \\ A_m b \end{pmatrix} = \sum_{j=1}^{m} A^j b_j$$

The computational complexity of this operation is $O(n^2)$.
We call **image** of a matrix A ($\mathrm{Im}(A)$) the set of vectors that can be obtained multiplying A by any vector in the domain of A.
On the other hand, we call **kernel** of a matrix A ($\ker(A)$) the set of vectors w in its domain such that $Aw = 0$.
Given two matrices $A \in M(n, m, \mathbb{R})$ and $B \in M(m, k, \mathbb{R})$ we call **matrix-matrix product** the following: $C = AB$ such that $C_{ij} = A_i B^j$, where ${A_i}^T \in \mathbb{R}^m$ is the i-th row of A, B^i is the i-th column of B ($B^i \in \mathbb{R}^m$) and $C \in M(n, k, \mathbb{R})$. The computational complexity of this operation is $O(n^3)$. Notice that this product is **not commutative**.
The matrix-matrix product also works on "matrices" made of one column only (vector of $\mathbb{R}^n$), but in this case a row of the right-side matrix is made by only one scalar...
Given a matrix $A \in M(n, \mathbb{R})$ we call **inverse** of A the matrix A^{-1} such that $A^{-1}A = AA^{-1} = I_n$.
The **transpose** of a matrix $A \in M(n, m, \mathbb{R})$ is A^T such that $A^T_{ij} = A_{ji}$
The **inverse of a product** (shoe-sock identity) is $(AB)^{-1} = B^{-1}A^{-1}$. Notice that this identity holds only for square matrices.
The **transpose of a product** (shoe-sock identity) is $(AB)^T = B^T A^T$.

The objective of this course, for the part concerning numerical methods, is solving linear systems efficiently.

Definition 1.1 (Linear system)**.** *Let $A \in M(n, m, \mathbb{R})$, $b \in \mathbb{R}^n$ and $x \in \mathbb{R}^m$. We term* ***linear system*** *the following:*

$$Ax = b$$

Our goal is to approximate such vector x, hence resulting in solving a minimum problem:

$$\min \|Ax - b\|$$

Something on Matlab ...

Notice that the machine precision is 10^{-16}, so we should pay attention when making computations, since we may incurr in some error (proportional to the size of the operands).
In Matlab a matrix is written as `A=[1, 2, 3; 4, 5, 6];`, where `[1, 2, 3]` is the first row of the matrix A.
The transpose of a matrix or a vector is denoted by `A'`.
If we are interested in only a part of our matrix `A` we may write `A[1:2, 1:3]` and obtain only the rows of `A` that go from 1 to 2 and those columns from 1 to 3.

Fact 1.1. *Let $A \in GL(n, \mathbb{R})$ (aka A is a real square matrix of size n and invertible), $B, C \in M(n, m, \mathbb{R})$.*
If $AB = AC$ then $B = C$.

Definition 1.2 (Block multiplications)**.** *Let $A \in M(n, m, \mathbb{R})$ and let $B \in M(m, k, \mathbb{R})$. We can compute the result of a block of the matrix AB as the product of the two blocks in A and B in the corresponding position.*

manca figura dei blocchi in latex

Fact 1.2 (Block triangular matrices)**.** *Let $M \in M(n, m, \mathbb{R})$ and $N \in M(m, k, \mathbb{R})$ such that they are* ***block triangular****. Their product is a block triangular matrix as well.*

$$MB = \begin{pmatrix} A & B \\ 0 & D \end{pmatrix} \begin{pmatrix} E & F \\ 0 & G \end{pmatrix} = \begin{pmatrix} AE & BF \\ 0 & DG \end{pmatrix}$$

Fact 1.3 (Properties of triangular matrices)**.**

1. *A block triangular matrix is invertible iff its blocks are invertible;*
2. *The eigenvalues of a block triangular matrix are the union of the eigenvalues of each block;*
3. *Let $M \in GL(n, m, \mathbb{R})$ such that $M = \begin{pmatrix} A & B \\ 0 & D \end{pmatrix}$ the inverse of M is*
$$M^{-1} = \begin{pmatrix} A^{-1} & -A^{-1}BD^{-1} \\ 0 & B^{-1} \end{pmatrix}.$$

Why are we interested in block triangular matrices? They depict a situation as shown in Figure 1.1.

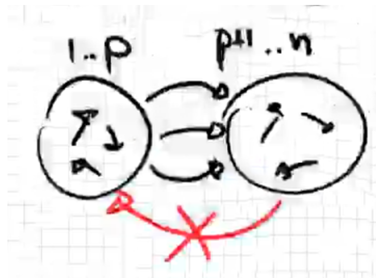

FIGURE 1.1: The adjacency matrix of a biparted graph has 0s in its bottom left part (Matlab syntax `A[p+1:n; 1:p]=0`).

1.2 Orthogonality

Definition 1.3 (Norms)**.** *Let $x \in \mathbb{R}^n$. We "measure" their magnitude using so-called "norms".*

EUCLIDEAN: $\|x\|_2 = x^T x = \sqrt{\sum_{i=1}^{n} {x_i}^2}$;

NORM 1: $\|x\|_1 = \sum_{i=1}^{n} |x_i|$;

p-NORM: $|x|_p = \left(\sum_{i=1}^{n} |x_i|^p \right)^{1/p}$;

0-NORM: $\|x\|_0 = |\{i \ : \ |x_i| > 0\}|$;

∞-NORM: $\|x\|_\infty = \max_{i=1,\dots,n} |x_i|$.

From now on in this part of th course we will refer to norm-2 only.

Definition 1.4 (Orthogonal matrix)**.** *Let $A \in M(n, m, \mathbb{R})$. We call A* ***orthogonal*** *if $\forall x \in \mathbb{R}^n$ $\|Ax\| = \|x\|$.*

Fact 1.4 (Equivalent definition of orthogonal matrix)**.** *Let $A \in M(n, \mathbb{R})$. A is orthogonal iff $A^T A = AA^T = I_n$, where I_n is the identity matrix of size n (1 on the diagonal, 0 elsewhere).*