OZBAY Yasin 3T2
LHOEST Simon 3T1
VAN RENTERGHEM Tom 3T1

Rapport du Bureau d'Etude.

Partie 1 :

1) On cherche ici à réécrire l'erreur de prédiction de f sur les données d'entraînement :

$$\min\left(\sum_{i=1}^{p}(f(u_i)-1)^2 + \sum_{j=1}^{q}(f(v_j)+1)^2\right)$$, sous forme matricielle.

On remarque premièrement que $\sum(x\pm1)^2$ est une norme deux au carré. On peut donc réécrire les sommes en normes :

$$\min\left(\sum_{i=1}^{p}(f(u_i)-1)^2 + \sum_{j=1}^{q}(f(v_j)+1)^2\right) = \min\left(\|f(u_i)-1\|_2^2 + \|f(v_j)+1\|_2^2\right)$$

La norme 2 étant une somme, on peut écrire la somme des deux normes comme la norme d'un seul vecteur :

$$\|f(u_i)-1\|_2^2 + \|f(v_j)+1\|_2^2 = \left\|\begin{pmatrix} f(u_1)-1 \\ \vdots \\ f(u_p)-1 \\ f(v_1)+1 \\ \vdots \\ f(v_q)+1 \end{pmatrix}\right\|_2^2$$

On remplace f par son expression : $f(x)=w^Tx+b$:

$$= \left\|\begin{pmatrix} (w^Tu_1+b)-1 \\ \vdots \\ (w^Tu_p+b)-1 \\ (w^Tv_1+b)+1 \\ \vdots \\ (w^Tv_q+b)+1 \end{pmatrix}\right\|_2^2 = \left\|\begin{pmatrix} w^Tu_1+b \\ \vdots \\ w^Tu_i+b \\ w^Tv_1+b \\ \vdots \\ w^Tv_j+b \end{pmatrix} - \begin{pmatrix} 1 \\ \vdots \\ 1 \\ -1 \\ \vdots \\ -1 \end{pmatrix}\right\|_2^2$$

On transpose les termes du premier vecteur pour pouvoir l'écrire sous forme de multiplication matricielle :

$$= \left\| \begin{pmatrix} u_1^T w + b \\ \vdots \\ u_p^T w + b \\ v_1^T w + b \\ \vdots \\ v_q^T w + b \end{pmatrix} - \begin{pmatrix} 1 \\ \vdots \\ 1 \\ -1 \\ \vdots \\ -1 \end{pmatrix} \right\|_2^2$$

On remarque ici que $b^T = b$, cela est dû au fait que b est un réel. On peut le remarquer dans l'expression : $w^T \cdot + b \pm 1$. On a le premier terme qui est un produit vectoriel alors que le second est une somme ou une différence, b doit donc être un réel pour se sommer à un autre réel.

On peut maintenant écrire la norme avec un produit d'une matrice par un vecteur :

$$= \left\| \begin{pmatrix} u_1^T & 1 \\ \vdots & \\ u_p^T & 1 \\ v_1^T & 1 \\ \vdots & \\ v_q^T & 1 \end{pmatrix} \begin{pmatrix} w \\ b \end{pmatrix} - \begin{pmatrix} 1 \\ \vdots \\ 1 \\ -1 \\ \vdots \\ -1 \end{pmatrix} \right\|_2^2 = \| Ax - y \|_2^2$$

On a bien $\min \left(\sum_{i=1}^{p} (f(u_i) - 1)^2 + \sum_{j=1}^{q} (f(v_j) + 1)^2 \right) = \min_{x \in \mathbb{R}^{n+1}} \| Ax - y \|_2^2$.

Par identification, on trouve :

$$A \in M_{p+q,2}(\mathbb{R}) = \begin{pmatrix} u_1^T & 1 \\ \vdots & \vdots \\ u_p^T & 1 \\ v_1^T & 1 \\ \vdots & \vdots \\ v_q^T & 1 \end{pmatrix}, \quad x = \begin{pmatrix} w \\ b \end{pmatrix} \quad \text{et} \quad y = \begin{pmatrix} 1 \\ \vdots \\ 1 \\ -1 \\ \vdots \\ -1 \end{pmatrix}$$ (1 de 1 à p et -1 de p+1 à q).

2) On note $\phi(x) = \|Ax-y\|_2^2$.

En dérivant ϕ selon x et en développant les conditions de minimum pour le 1er ordre, on peut écrire :

$\phi'(x) = 2A^TAx - 2A^Ty = 0$.

On peut ici reconnaître l'équation normale $A^TAx - A^Ty = 0$ multipliée par 2.

On peut ainsi dire que les points critiques de ϕ sont solutions de l'équation normale.

On vérifie que les points critiques de ϕ sont bien solution de l'équation normale en dérivant une nouvelle fois cette équation :

$\phi''(x) = 2A^TA \neq 0$.

Les points critiques de ϕ sont bien non-nuls donc pas dégénérés. De plus, on étudie le minimum d'une norme, on peut dire que les points critiques de ϕ (dérivée seconde) sont bien solution de l'équation normale.

3) Pour trouver le rang de la matrice A à partir de son itération QR, on range par ordre décroissant la matrice trouvée par itération QR. Ainsi, on a une matrice de la forme :

$$QR(A^TA) = \begin{pmatrix} a_{1,1} & & & & \\ & \ddots & & & \\ & & a_{i,i} & & \\ & & & 0 & \\ & & & & \ddots & \\ & & & & & 0 \end{pmatrix}_{785,785}$$

Le rang de A correspond donc au nombre de coefficients non-nuls de la diagonale.

A^TA n'est donc pas inversible car $\det(A^TA)=0$

4) De la même façon, on estime le rang de A avec SVD en comptant le nombre de coefficients non nuls de Σ.

Dans notre cas, on estime le rang à 716 pour l'itération QR et 713 avec les valeurs singulières.

5.a) On vérifie qu'une matrice est symétrique définie positive, pour cela il faut :

- $A = A^T$
- Que tous les termes de la matrice soient positifs.
- Qu'aucun terme de la diagonale soit nul.

5.b) On trouve les taux de réussite et matrices de confusions suivants pour $\varepsilon = 1$:

0 : 98,43 % $\begin{pmatrix} 866 & 114 \\ 43 & 977 \end{pmatrix}$ 1 : 98,34 % $\begin{pmatrix} 1035 & 100 \\ 66 & 8799 \end{pmatrix}$

2 : 95,84 % $\begin{pmatrix} 644 & 388 \\ 28 & 8940 \end{pmatrix}$ 3 : 96,02 % $\begin{pmatrix} 655 & 355 \\ 43 & 8947 \end{pmatrix}$

4 : 96,65 % $\begin{pmatrix} 685 & 297 \\ 38 & 8980 \end{pmatrix}$ 5 : 94,65 % $\begin{pmatrix} 413 & 479 \\ 56 & 9052 \end{pmatrix}$

6 : 97,40 % $\begin{pmatrix} 772 & 186 \\ 74 & 8968 \end{pmatrix}$ 7 : 96,46 % $\begin{pmatrix} 735 & 293 \\ 61 & 8911 \end{pmatrix}$

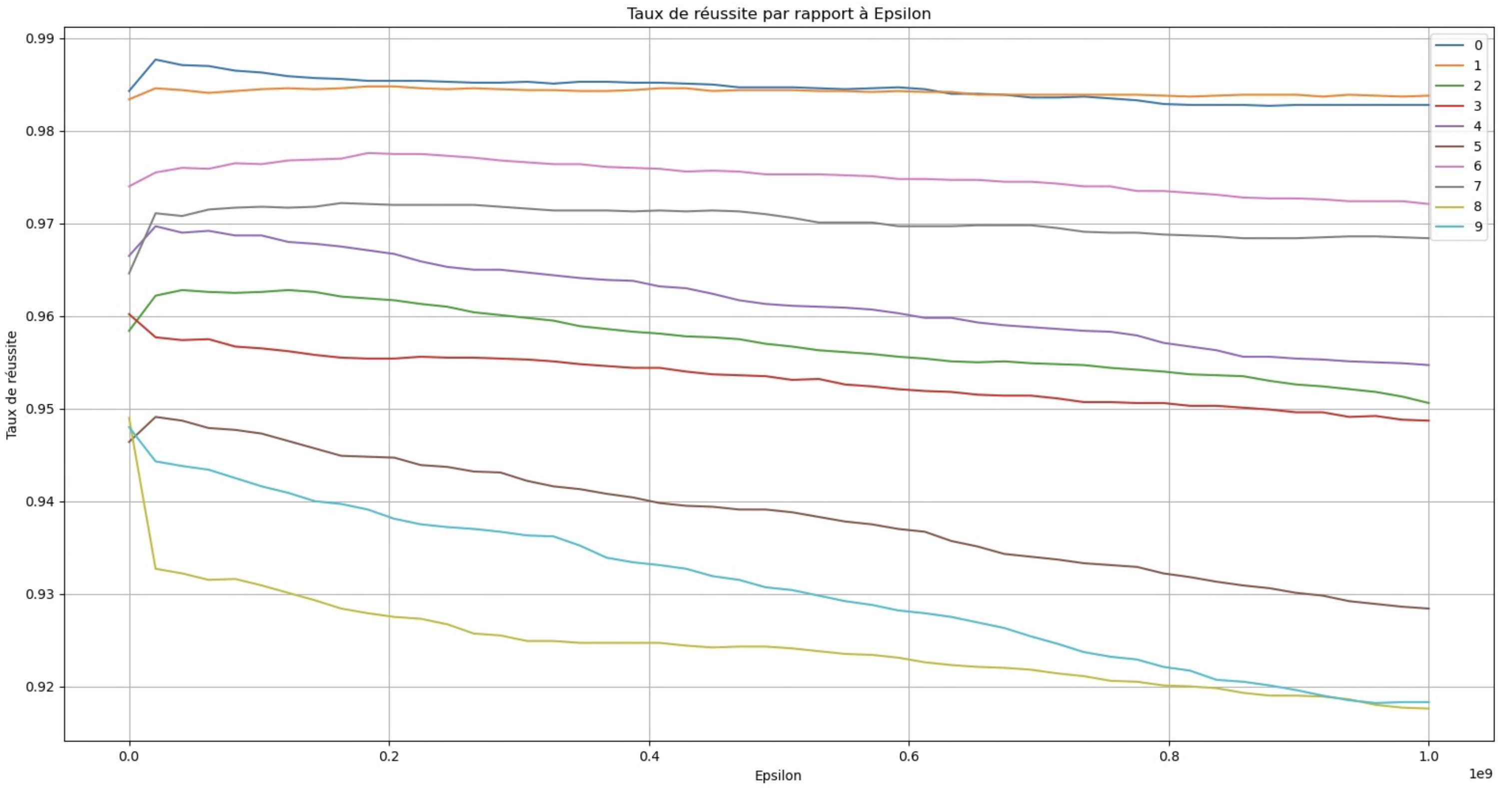
8 : 94,90 % $\begin{pmatrix} 504 & 470 \\ 40 & 8986 \end{pmatrix}$ 9 : 94,80 % $\begin{pmatrix} 560 & 449 \\ 71 & 8920 \end{pmatrix}$

5.c) On calcule ε_i qui maximise le taux de réussite pour chaque chiffre :

0 : $2{,}04 \cdot 10^{7}$	5 : $2{,}04 \cdot 10^{7}$
1 : $1{,}84 \cdot 10^{8}$	6 : $1{,}84 \cdot 10^{8}$
2 : $4{,}08 \cdot 10^{7}$	7 : $1{,}63 \cdot 10^{8}$
3 : $1 \cdot 10^{-10}$	8 : $1 \cdot 10^{-10}$
4 : $2{,}04 \cdot 10^{7}$	9 : $1 \cdot 10^{-10}$

On pourrait ré-afficher les taux de réussite et matrices de confusion de chaque chiffre avec le nouveau ε optimal. On retiendra que les taux de réussite sont meilleurs.

On peut afficher le taux de réussite de chaques chiffres en fonction de ε :



6) On calcule le taux de réussite de notre programme avec le ε optimal.
On trouve un taux de 86,45%.

C'est un meilleur taux que celui en TP qui est de 86,03%.
Les 0,42% ne sont pas négligeables lors d'algorithme de ce genre qui vont être utilisés sur des millions voire milliards d'images.
Notre méthode est donc meilleure que celle du pseudo-inverse.

7) L'interface python se trouve dans le fichier interface.

8) a) On remplace A^TA par $A^TA + \epsilon I_n$ dans l'équation normale :

$$(A^TA + \epsilon I_n)x = A^Tb \Leftrightarrow A^TAx - A^Tb + \epsilon I_n.x = 0 \Leftrightarrow A^T(Ax-b) + \epsilon I_n.x = 0$$

On cherche la solution à cette nouvelle équation de la même manière qu'auparavant, en minimisant la norme.
Les solutions de la nouvelle équation sont donc de la forme :

$\|Ax-y + \epsilon I_n x\|_2^2 = \|Ax-y\|_2^2 + \|\epsilon I_n.x\|_2^2$, ϵ étant un réel on peut le sortir de la norme :

$$\|Ax-y\|_2^2 + \epsilon\|I_n.x\|_2^2 = \|Ax-y\|_2^2 + \epsilon\|x\|_2^2$$

En manipulant les normes, on voit bien que résoudre l'équation normale régularisée revient à chercher les points critiques de $\phi(x) = \|Ax-y\|_2^2 + \epsilon\|x\|_2^2$.

b) On travaille ici sur des matrices non-inversibles. Le but de la solution proposée est de régulariser l'équation normale des moindres carrés. En ajoutant ϵI_n à A^TA, on crée une matrice inversible car on sera sûr qu'il n'y aura pas de zéros sur la diagonale.
Cette solution régularisée permet également et surtout d'optimiser la solution x.
On cherche à minimiser $\|Ax-y\|_2^2 + \epsilon\|x\|_2^2$ donc plus ϵ sera grand, plus x sera petit.
On ne cherche donc plus une solution minimale donnée par $\min\|Ax-y\|_2^2$ mais une solution optimale où x est minimal.
La forme de x sera donc modifiée par ϵ qui diminuera x.

Partie 2 :

Pour la construction de la fonction Proust, le sujet était plutôt ambigu sur la forme et les calculs de chaque variable. Après consultation avec les professeurs et après avoir essayé plusieurs façon différente de définir la fonction, nous avons trouvé qu'il est meilleur d'avoir les matrices images et cm sous forme de ligne. C'est pour cela que l'on reshape les matrices en (1,784). Cette méthode est la méthode la plus rapide et la plus précise que nous ayons trouvé.

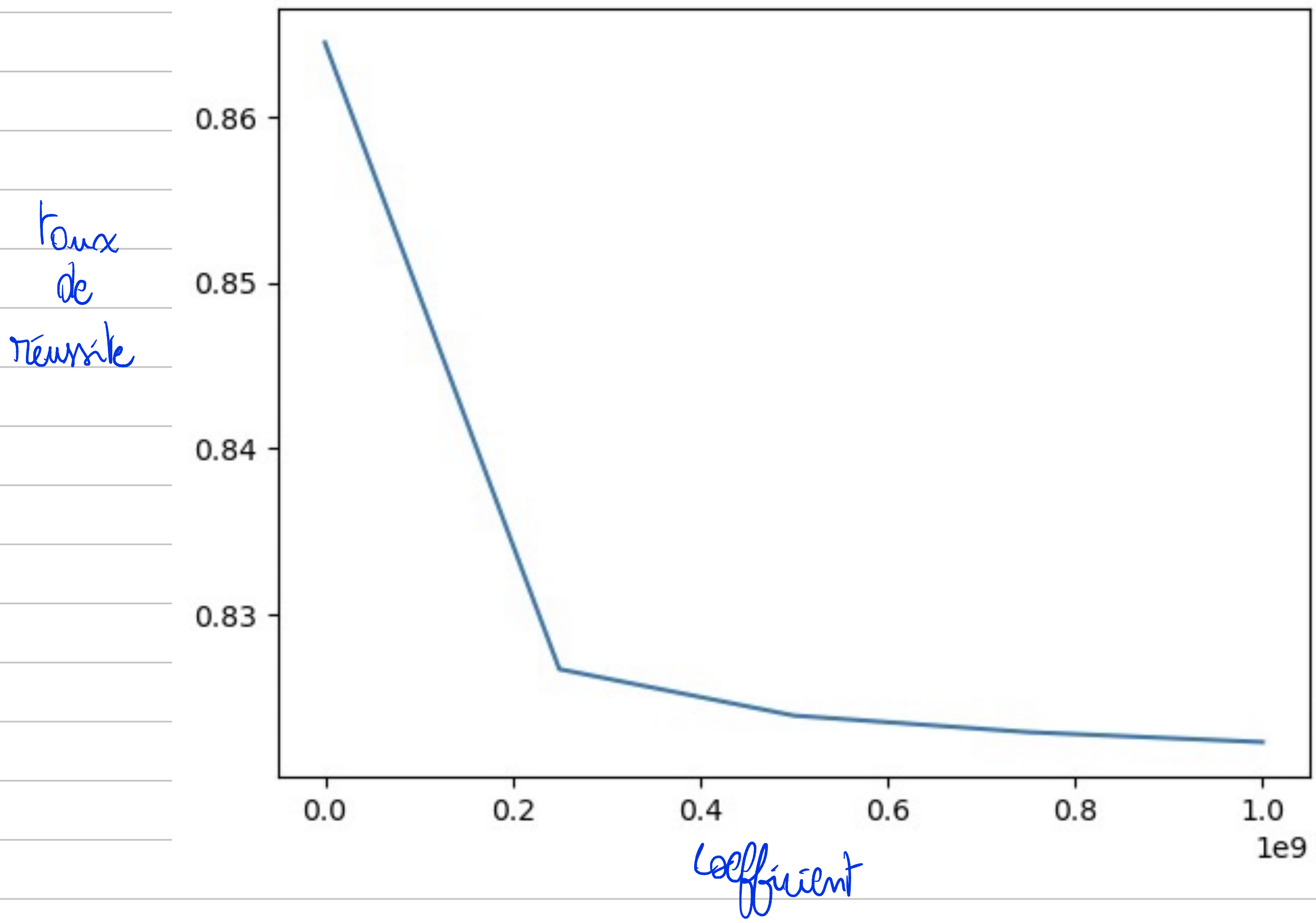
Nous avons considéré, comme indiqué par M. Couffignal, de rentrer directement dix fois l'image et CM complet dans la fonction. Après tests, nous avons préféré faire une boucle dix fois pour chaque composante de CM.

Ainsi, on trouve un taux de réussite de 82,08%. C'est donc la première méthode qui l'emporte. De plus, la méthode avec Cholesky est plus rapide : le taux de réussite se calcule en moins d'une seconde, là où l'analyse proustéenne prend plus de quatre minutes.

Partie 3 :

En mélangeant les deux méthodes, on peut espérer que les erreurs d'une méthode soit compensée par l'autre.

On trace donc le taux de réussite en fonction du coefficient :



Avec un coefficient de 0, on a le taux de réussite maximum, qui en réalité celui de la première méthode (86,45%)

Encore une fois, le taux est très long à calculer à cause de la deuxième méthode.