SVD

Thierry Goudon

La “Décomposition en Valeurs Singulières” (SVD : Singular Value Decomposition) c'est

- un résultat fondamental d'algèbre linéaire

  Les premiers résultats remontent à Eugenio Beltrami et Camille Jordan, en 1873-74, puis à James Joseph Sylvester en 1889 pour les matrices réelles carrées [11]. Des propriétés similaires ont été mises en évidence pour des opérateurs intégraux (exemple typique d'opérateurs compacts) par Erhard Schmidt en 1907, sans connaissance des résultats en dimensions finies, et par Émile Picard en 1910, qui est l'origine du terme moderne de “valeurs singulières”. La première preuve de la décomposition en valeurs singulières pour les matrices rectangulaires et complexes est attribuée à Carl Eckart et à Gale Young, en 1936 [6].

- un algorithme très performant et très utile.

  En dépit du rôle central de la SVD pour les applications, aucune méthode efficace de calcul de cette décomposition n'était connue jusqu'en 1965-70, avec les percées de Gene H. Golub et ses collaborateurs William Kahan et Christian Reinsch notamment [8]. L'amélioration des perfomances de ces méthodes reste un sujet d'actualité [7, 12]

- de multiples applications.

  La SVD intervient naturellement dans la formulation de problèmes d'optimisation, comme on va le voir, en lien avec la méthode des moindres carrés. C'est aussi une méthode naturelle de compression de données (pour éliminer les informations redondantes) ou de filtrage (pour distinguer dans une image le signal du bruit). Pour mentionner des applications spectaculaires, les algorithmes SVD sont utilisés pour exploiter les données de l'interferomètre aLIGO, qui a permis la détection d'ondes gravitationnelles. Ces algorithmes interviennent aussi dans les problèmes de “ranking”. Entre 2006 et 2009, la compagnie Netflix avait lancé une compétition internationale avec un prix de un million de dollars pour une amélioration de plus de 10% des performances de leur système `Cinematch`. Les matrices à traiter pour ce problème ont une taille déterminée par plus 400 000 membres et 100 millions d'évaluations (1% du total des évaluations possibles). Le test consistait à reconstituer 1.4 millions d'évaluations manquantes. Les méthodes les plus performantes proposées reposent sur des variantes de la SVD et son calcul [1, 2].

1 Un énoncé fondamental d'algèbre linéaire

Théorème 1 *Soit A une matrice de taille $n \times m$ et de rang r ($r \leq \min(n,m)$). Il existe une unique famille de réels strictement positifs $\sigma_1 \geq \ldots \geq \sigma_r > 0$ et un couple de matrices orthogonales $(U,V) \in O_n(\mathbb{R}) \times O_m(\mathbb{R})$ telles que*

$$A = U\,\Sigma\,V^\intercal, \qquad \Sigma = \left(\begin{array}{ccc|c} \sigma_1 & 0 & 0 & \\ 0 & \sigma_2 & & 0 \\ & & 0 & \\ \hline 0 & & 0 & 0 \end{array}\right) \in \mathcal{M}_{n,m} \tag{1.1}$$

où Σ est la matrice de taille $n \times m$ dont les r premiers éléments diagonaux sont les σ_i et tous les autres éléments sont nuls.

Les σ_i sont appelées les valeurs singulières *de la matrice A. Ce sont également les racines carrées des valeurs propres non nulles de $A^\intercal A$ et de $A A^\intercal$. Par ailleurs, les r premières colonnes de U et de V sont des vecteurs propres de ces deux matrices.*

Preuve. On commence par remarquer que les matrices $A^\intercal A$ et de $A A^\intercal$ (qui sont respectivement de taille $m \times m$ et $n \times n$) sont des matrices symétriques et positives au sens où $A^\intercal Ax \cdot x = |Ax|^2 \geq 0$ et $A A^\intercal \xi \cdot \xi = |A^\intercal \xi|^2 \geq 0$ pour tous $x \in \mathbb{R}^m$, $\xi \in \mathbb{R}^n$. En particulier, ces matrices sont donc diagonalisables en base orthogonale. On note $\sigma_1^2 \geq \sigma_2^2 \geq \ldots \geq \sigma_r^2 > 0$ les r valeurs propres non nulles de $A^\intercal A$. En effet, si $A^\intercal Ax = 0$ alors $A^\intercal Ax \cdot x = |Ax|^2 = 0$ implique qu'aussi $Ax = 0$: $x \in \mathrm{Ker}(A)$; la réciproque est immédiate. Il y a donc bien $r = \mathrm{rank}(A) = m - \dim(\mathrm{Ker}(A))$ (d'après le théorème du rang[1]) valeurs propres strictement positives de $A^\intercal A$ (distinctes ou non).

Pour faire le lien avec la décomposition SVD, on peut commencer par supposer que la matrice A s'écrit sour la forme (1.1) et identifier ainsi les matrices Σ, U et V. En effet, on déduit de (1.1) que

$$A^\intercal A = \big(U\Sigma V^\intercal\big)^\intercal U\Sigma V^\intercal = V\Sigma^\intercal \underbrace{U^\intercal U}_{=\mathbb{I}} \Sigma V^\intercal = V\Sigma^\intercal\Sigma V^\intercal$$

puisque U est orthogonale. De plus, dans cette expression $\Sigma^\intercal\Sigma = \mathrm{diag}(\sigma_1^2, \ldots, \sigma_r^2, 0, \ldots, 0)$ est une matrice $m \times m$ diagonale. Autrement dit, on obtient une diagonalisation de la matrice $A^\intercal A$ où V est la matrice associée au changement de base : V apparaît comme la matrice de passage de la base canonique à la base des vecteurs propres orthonormés de $A^\intercal A$.

Cette observation conduit donc à définir V comme la matrice dont les colonnes sont des vecteurs propres de $A^\intercal A$:

$$v_i \in \mathbb{R}^m, \quad A^\intercal Av_i = \sigma_i^2 v_i \text{ pour } i \in \{1, \ldots, r\}, \quad A^\intercal Av_i = 0 \text{ pour } i \in \{r+1, \ldots, m\}, \quad v_i \cdot v_j = \delta_{ij}.$$

On pose ensuite

$$u_j = \frac{Av_j}{\sigma_j} \text{ pour } j \in \{1, \ldots, r\},$$

[1] si $A \in \mathcal{L}(E,F)$ alors $\dim(\mathrm{Ran}(A)) + \dim(\mathrm{Ker}(A)) = \dim(E)$

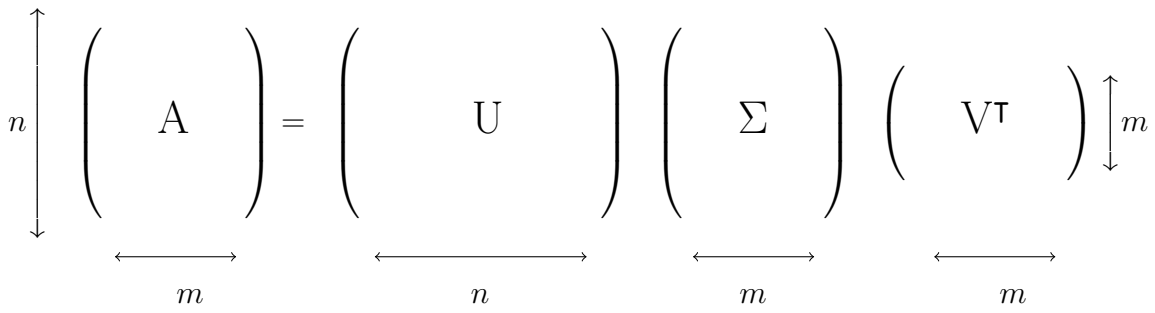
de sorte que

$$u_i \cdot u_j = \frac{Av_i \cdot Av_j}{\sigma_i \sigma_j} = \frac{A^\intercal A v_i \cdot v_j}{\sigma_i \sigma_j} = \sigma_i^2 \frac{v_i \cdot v_j}{\sigma_i \sigma_j} = \delta_{ij}.$$

On note aussi que, pour $j \in \{1, ..., r\}$, $AA^\intercal u_j = A\frac{A^\intercal A v_j}{\sigma_j} = \frac{A\sigma_j^2 v_j}{\sigma_j} = \sigma_j^2 u_j$. Les vecteurs $u_1, ..., u_r$ forment donc une famille orthonormée dans $\mathbb{R}^n$ (formée de vecteurs propres de $AA^\intercal$), qu'on complète pour obtenir une base orthonormée de $\mathbb{R}^n$; on note U la matrice correspondante, dont les colonnes sont données par ces vecteurs u_i. Par construction, les matrices U et V sont orthogonales. Il reste à calculer $\Sigma = U^\intercal AV$. Or, les colonnes de AV sont précisément les vecteurs Av_j qui valent donc $\sigma_j u_j$ si $j \in \{1, ..., r\}$ et 0 si $j \in \{r+1, ..., m\}$. Il vient finalement

$$U^\intercal AV = \begin{pmatrix} u_1^\intercal \\ u_2^\intercal \\ \vdots \\ u_n^\intercal \end{pmatrix} \begin{pmatrix} \sigma_1 u_1 & \sigma_2 u_2 & \cdots & \sigma_r u_r & 0 \end{pmatrix} = \left(\begin{array}{cccc|c} \sigma_1 & 0 & \cdots & 0 & \\ 0 & \sigma_2 & & 0 & \\ & & \ddots & & 0 \\ 0 & & 0 & \sigma_r & \\ \hline & & 0 & & 0 \end{array}\right) = \Sigma.$$

Dans ces formulations, il faut bien prendre garde aux dimensions des objets manipulés



La démonstration s'étend au cas des matrices à coefficients complexes. ∎

L'intérêt de cette décomposition pour les applications, et notamment pour le traitement du signal, provient de l'énoncé suivant : la SVD fournit naturellement la meilleure approximation de rang k de la matrice A au sens de la norme de Frobenius ou de la norme euclidienne[2].

Théorème 2 *Soit $A \in \mathcal{M}_{nm}$, de rang r. Soir ν un entier inférieur ou égal à r. Alors $A_\nu = \sum_{i=1}^{\nu} \sigma_i u_i v_i^\intercal$ minimise $\|A - B\|_F$ sur l'ensemble des matrices $B \in \mathcal{M}_{nm}$ de rang ν, où $\|M\|_F = \sqrt{\sum_{i=1}^n \sum_{j=1}^m M_{ij}^2}$ est la norme de Frobenius.*

Preuve. Pour établir ce résultat, on peut commencer par remarquer que la norme de Frobenius

$$\|M\|_F^2 = \mathrm{Tr}(M^\intercal M) \tag{1.2}$$

[2]la norme matricielle utilisée ici est définie par $\|M\|_2 = \sup_{\|x\|_2=1} \|Mx\|_2$

dérive d'un produit scalaire. Cependant, le théorème de projection ne s'applique pas sur l'ensemble $\mathcal{E}_\nu$ des matrices de rang ν, cet ensemble n'étant pas fermé[3]. On procède donc par une approche plus directe. En notant que la multiplication par une matrice orthogonale laisse la norme $\|\cdot\|_F$ invariante, on obtient

$$\|A-B\|_F^2 = \|\Sigma - U^\intercal B V\|_F^2 = \sum_{i=1}^{r} |\sigma_i - d_{ii}|^2 + \sum_{i>r} |d_{ii}|^2 + \sum_{i\neq j} d_{ij}^2 \tag{1.3}$$

où les d_{ij} sont les coefficients de la matrice $U^\intercal BV$. On en déduit qu'une matrice minimisante $B \in \mathcal{E}_\nu$ est telle que $d_{ij} = 0$ si $i \neq j$ et $d_{ii} = 0$ si $i > \nu$. On se ramène ainsi à minimiser

$$\mathscr{J} : (d_1, ..., d_\nu) \longmapsto \sum_{i=1}^{\nu} |\sigma_i - d_i|^2 + \sum_{i=\nu+1}^{r} \sigma_i^2. \tag{1.4}$$

Cette fonction est de classe $C^2(\mathbb{R}^\nu, \mathbb{R})$ avec $\partial_{d_i}\mathscr{J}(d) = 2(\sigma_i - d_i)$ et sa matrice hessienne est $\mathrm{diag}(\sigma_1, ..., \sigma_\nu)$: elle est convexe et atteint son minimum, qui vaut donc $\sum_{i=\nu+1}^{r}\sigma_i^2$, au point critique $(\sigma_1, ..., \sigma_\nu)$. ∎

Théorème 3 *Soit A une matrice comme au théorème précédent et $k \leq r$ un entier. On pose*

$$A_k = U\,\Sigma_k\,V^\intercal = \sum_{j=1}^{k} \sigma_j u_j v_j^\intercal, \tag{1.5}$$

où $U\,\Sigma\,V^\intercal$ est une SVD de A et Σ_k la matrice obtenue à partir de Σ en ne conservant que les k premiers éléments diagonaux et en annulant les autres. Alors cette matrice vérifie

$$\|A - A_k\|_2 = \inf_{B,\ \mathrm{rang}(B)=k} \|A-B\|_2 = \sigma_{k+1}. \tag{1.6}$$

Preuve. On rappelle plusieurs faits sur la norme $\|A\|_2 = \sup\big\{|Ax|,\ |x| = 1\big\}$, où $|\cdot|$ désigne la norme euclidienne. D'abord, cette norme est invariante par multiplication par des matrices orthogonales car, pour U matrice orthogonale, on a $|Ux|^2 = Ux \cdot Ux = U^\intercal Ux \cdot x = x \cdot x = |x|^2$. En conséquence, on a aussi, pour toute matrice A, $\|UA\|_2 = \|A\|_2$. Ensuite, comme $|Ax|^2 = Ax \cdot Ax = A^\intercal Ax \cdot x$, où $A^\intercal A$ est symmétrique, il vient $\|A\|_2 = \sigma_1$, la plus grande valeur singulière de A.

On en déduit aussitôt que

$$\|A - A_k\|_2 = \|\Sigma - \Sigma_k\|_2 = \sigma_{k+1}.$$

On va conclure en montrant que pour toute matrice B de rang $k < m$, on peut construire un vecteur $x \in \mathbb{R}^m$ tels que $|x| = 1$ et $|(A-B)x| \geq \sigma_{k+1}$. On en déduit que, pour toute matrice B de rang k, $\|A-B\|_2 \geq \sigma_{k+1} = \|A - A_k\|_2$ et conclure ainsi. Comme $\mathrm{rank}(B) = k$, le théorème du rang donne $\dim(\mathrm{Ker}(B)) = m - k > 0$. De plus, si on considère B comme une application linéaire sur

[3] comme le montre l'exemple de la matrice $\mathrm{diag}(1, \epsilon, ..., \epsilon)$ quand $\epsilon \to 0$

$\mathrm{Vect}(v_1,...,v_{k+1})$, comme $\dim\big(\mathrm{Vect}(v_1,...,v_{k+1})\big) = k+1$, le théorème du rang assure encore que le noyau de B sur $\mathrm{Vect}(v_1,...,v_{k+1})$ ne peut pas être trivial (sinon l'image de B serait de dimension au moins $k+1$). Autrement dit, il existe un vecteur non nul $x \in \mathrm{Ker}(B) \cap \mathrm{Vect}(v_1,...,v_{k+1})$, qu'on écrit $x = \sum_{j=1}^{k+1} \alpha_j v_j$. Pour ce vecteur, on a

$$|(A-B)x| = |Ax| = |\Sigma V^\intercal x| = \Big|\sum_{j=1}^{k+1} \alpha_j \Sigma V^\intercal v_j\Big|.$$

Or, les lignes de $V^\intercal$ sont précisément les vecteurs v_i. Ainsi, $V^\intercal v_j = e_j$, le $j$ème vecteur de la base canonique et $\Sigma V^\intercal v_j = \sigma_j e_j$. Il s'ensuit que

$$|(A-B)x|^2 = |(\sigma_1\alpha_1, \sigma_2\alpha_2, ..., \sigma_{k+1}\alpha_{k+1}, 0, ..., 0)|^2 = \sum_{j=1}^{k+1} \sigma_j^2\alpha_j^2 \geq \sigma_{k+1}^2 \sum_{j=1}^{k+1} \alpha_j^2 = \sigma_{k+1}^2 |x|^2 = \sigma_{k+1}^2.$$

■

Les résultats obtenus jusque là suggère une méthode de construction de la décomposition SVD. On commence par former la matrice $A^\intercal A$. On applique la méthode de la puissance pour trouver la valeur propre dominante σ_1^2, un vecteur propre normalisé v_1 et $u_1 = \frac{Av_1}{\sigma_1}$. Puis on recommence avec $A - \sigma_1 u_1 v_1^\intercal$, etc. Cette méthode permet de construire A_k, au moins quand les valeurs propres sont simples. Elle présente toutefois des faiblesses : coût du calcul des produits $A^\intercal A$, difficulté de convergence si les valeurs singulières sont proches, manque de robustesse... dont on verra des illustrations plus loin.

2 Application aux moindres carrées

On a vu qu'un filtre se traduit comme l'action d'un opérateur de convolution :

$$\mathcal{A}\phi(x) = \int_{-\pi}^{\pi} a(x-y)\phi(y)\,\mathrm{d}y. \tag{2.7}$$

On notera que $\mathcal{A}\phi$ est à valeurs réelles lorsque ϕ est à valeurs réelles. En posant $h = \frac{2\pi}{M+1}$, on approche cette formule par la relation

$$\mathcal{A}_M\phi(x) = h\sum_{m=0}^{M} a(x-x_m)\phi(x_m), \qquad x_m = -\pi + mh. \tag{2.8}$$

On dispose de N points de mesure $\widetilde{x}_1, ..., \widetilde{x}_N$ sur $[-\pi, \pi[$ sur lesquels on connaît les valeurs $v_1, ..., v_N$ de $\mathcal{A}\phi = v$: ce sont les valeurs du signal accessibles via l'action du filtre $\mathcal{A}$. On souhaiterait reconstituer le signal, c'est-à-dire trouver ϕ tel que $\mathcal{A}_M\phi(\widetilde{x}_n) = v_n$. En pratique ces mesures sont nombreuses : on a $N \gg M$ et en général le problème n'admet pas de solution. Ces considérations amènent donc à rechercher un minimiseur de

$$J : y = (y_1, ..., y_M) \in \mathbb{C}^M \longmapsto \frac{1}{2}\,|Ay - v|^2 \tag{2.9}$$

où $|\cdot|$ désigne la norme euclidienne, $v = (v_1, ..., v_N) \in \mathbb{C}^N$ et $A \in \mathcal{M}_{N,M}$ est la matrice (à N lignes et M colonnes) de composantes

$$A_{nm} = h\, a(\widetilde{x}_n - x_m).$$

De tels problèmes de minimisation sont extrêmement courants.

Lemme 1 *Les solutions du problème* (2.9) *vérifient*[4]

$$A^*Ay = A^*v \tag{2.10}$$

(qu'on appelle équation normale*) et le minimiseur est unique si la matrice A est de rang maximal.*

Preuve. La fonctionnelle J est continue, et même de classe C^2. Elle admet donc un minimiseur sur tout domaine compact. On a

$$\begin{aligned} J(y+h) &= \frac{1}{2}\,(A(y+h)-v)\cdot(A(y+h)-v) \\ &= \frac{1}{2}(Ay-v)\cdot(Ay-v) + \underbrace{\frac{1}{2}(Ay-v\cdot Ah)+\frac{1}{2}(Ah\cdot Ay-v)}_{=\frac{1}{2}A^*(Ay-v)\cdot h+\frac{1}{2}h\cdot A^*(Ay-v)} + \frac{1}{2}(Ah\cdot Ah) \\ &= J(y) + \mathrm{Re}\big(A^*(Ay-v)\cdot h\big) + \frac{1}{2}\|Ah\|^2. \end{aligned}$$

Si y minimise J, alors, avec $h = t\tilde{h}$, $t > 0$, il vient

$$0 \le \frac{J(y+t\tilde{h})-J(y)}{t} = \mathrm{Re}\big(A^*(Ay-v)\cdot\tilde{h}\big) + \frac{t}{2}\|A\tilde{h}\|^2$$

d'où on déduit en faisant tendre t vers 0

$$\mathrm{Re}A^*(Ay-v)\cdot\tilde{h} \ge 0.$$

Cette relation est satisfaite par tout vecteur $\tilde{h}$. En l'appliquant à $-\tilde{h}$ ainsi qu'à $\pm i\tilde{h}$, on en conclut qu'en fait

$$A^*(Ay-v) = 0.$$

Réciproquement, si y satisfait (2.10), alors on a $J(y+h) = J(y)+\frac{1}{2}\|Ah\|^2 \ge J(y)$ pour tout $h \in \mathbb{R}^M$ et y est un minimiseur de J. On peut interpréter (2.10) en disant que Ay est le projeté orthogonal de v sur $\mathrm{Ran}(A)$ puisque pour tout $h \in \mathbb{C}^m$, on a $(Ay-v)\cdot Ah = 0$.

L'équation normale (2.10) admet toujours au moins une solution. Ce fait est une conséquence des propriétés générales suivantes

$$\mathrm{Ker}(A^*A) = \mathrm{Ker}(A), \qquad \mathrm{Ran}(A^*A) = \mathrm{Ran}(A^*). \tag{2.11}$$

Ainsi $A^*v \in \mathrm{Ran}(A^*) = \mathrm{Ran}(A^*A)$ implique qu'il existe au moins un vecteur y tel que $A^*Ay = A^*v$.

[4] où le symbole $*$ désigne la matrice transposée-conjuguée.

Pour justifier (2.11), on commence par remarquer qu'il suffit de démontrer la première relation puisqu'alors

$$\mathrm{Ran}(A^*A) = (\mathrm{Ker}(A^*A)^*)^\perp = (\mathrm{Ker}(A^*A))^\perp = (\mathrm{Ker}(A))^\perp = \mathrm{Ran}(A^*).$$

Soit $x \in \mathrm{Ker}(A)$. Alors $A^*Ax = A^*0 = 0$, donc $x \in \mathrm{Ker}(A^*A)$. Réciproquement, si $x \in \mathrm{Ker}(A^*A)$, la relation $A^*Ax \cdot x = |Ax|^2$ montre que $x \in \mathrm{Ker}(A)$.

L'unicité en lien avec le rang de la matrice A est établie dans l'énoncé suivant. ∎

Lemme 2 *Soit $A \in \mathcal{M}_{n,m}$. L'équation* (2.10) *admet une unique solution, qui minimise* (2.9)*, si et seulement si A est de rang maximal.*

Preuve. Le théorème du rang implique qu'on a forcément ici $m = \dim(\mathrm{Ran}(A)) + \dim(\mathrm{Ker}(A)) \leq n + \dim(\mathrm{Ker}(A))$.

La matrice A^*A est hermitienne (de taille $n \times n$) et on remarque que

$$A^*Ay \cdot y = Ay \cdot Ay = |Ay|^2 \geq 0. \tag{2.12}$$

Aussi, si $y \in \mathrm{Ker}(A^*A)$, alors $y \in \mathrm{Ker}(A)$ (on peut aussi directement utiliser l'identification (2.11)). Lorsque A est injective, on en déduit que $y = 0$ et A^*A est donc inversible : l'équation (2.10) admet une unique solution. D'après le théorème du rang, cette situation est exclue si $n < m$ (car $m = \dim(\mathrm{Ran}(A)) + \dim(\mathrm{Ker}(A))$ devient $m = \dim(\mathrm{Ran}(A))$ lorsque A est injective et $\mathrm{Ran}(A) \subset \mathbb{C}^n$). Réciproquement, soit $y \in \mathbb{C}^m$ solution de (2.10) ; s'il existe un élément $x_0 \neq 0$ de $\mathrm{Ker}(A)$, alors pour tout $t \neq 0$, $y + tx_0 \neq y$ est aussi solution de (2.10).

∎

En pratique, les coefficients de A proviennent des mesures et rien n'assure que la condition de rang est satisfaite. Cependant on peut exploiter la décomposition en valeurs singulières de A, $A = U\Sigma V^*$, pour exprimer le problème sous une forme plus simple. En effet, comme U est orthogonale, on a $J(y) = \frac{1}{2}|\Sigma V^*y - U^*b|^2$. Dans ces variables, le problème de minimisation est facile à résoudre : on pose

$$\begin{aligned}
&\tilde{b} = U^*b = \begin{pmatrix} u_1^\intercal b \\ \vdots \\ u_n^\intercal b \end{pmatrix}, \\
&\text{si } \Sigma_{ii} > 0 \ (i \in \{1, ..., r\}) \text{ alors } \widetilde{y}_i = \frac{\tilde{b}_i}{\Sigma_{ii}}, \text{ et sinon } \widetilde{y}_i = 0 \ (i \in \{r+1, ..., m\}), \\
&y = V\widetilde{y} = \widetilde{y}_1 v_1 + ... + \widetilde{y}_m v_m \in \mathbb{R}^m.
\end{aligned} \tag{2.13}$$

On distingue donc deux situations : ou bien A est de rang maximal et dans ce cas cette formule fournit la solution recherchée, ou bien on obtient ainsi la solution du problème de minimisation dont la norme est minimale. En effet, comme on l'a vu, lorsque A n'est pas de rang maximal, l'équation (2.10) admet une infinité de solutions et l'ensemble des minimiseurs est convexe. Néanmoins, la solution obtenue à l'aide de la décomposition en valeurs singulières joue un rôle particulier en vertu de l'énoncé suivant.

Théorème 4 *Soit A une matrice de $\mathcal{M}_{n,m}$, de rang r, dont on note $A = U\Sigma V^\intercal$ sa décomposition en valeurs singulières, avec $\sigma_1 \geq \sigma_2 \geq ... \geq \sigma_r > 0$. Soit $b \in \mathbb{R}^n$. Alors le vecteur*

$$x_\star = \sum_{j=1}^{r} \frac{u_j^\intercal b}{\sigma_j} v_j \in \mathbb{R}^m \tag{2.14}$$

est le minimiseur de $|Ax - b|$ dont la norme ℓ^2 est minimale.

Preuve. Soit $x = Vy \in \mathbf{R}^m$. On a

$$|Ax - b|^2 = |\Sigma y - U^\intercal b|^2 = \sum_{j=1}^{r} |\sigma_j y_j - u_j^\intercal b|^2 + \sum_{j=r+1}^{n} |u_j^\intercal b|^2. \tag{2.15}$$

Les minimiseurs vérifient donc $y_j = u_j^\intercal b/\sigma_j$ pour tout $j \in \{1, ..., r\}$. Le vecteur pour lequel tous les autres coefficients y_j sont nuls est le minimiseur de norme ℓ^2 minimale.

On note

$$\Sigma^+ = \begin{pmatrix} \mathrm{diag}(1/\sigma_1, ..., 1/\sigma_r) & 0 \\ 0 & 0 \end{pmatrix} \in \mathcal{M}_{m,n}, \qquad A^+ = V\Sigma^+ U^\intercal \in \mathcal{M}_{m,n} \tag{2.16}$$

ce "pseudo-inverse" (appelé *pseudo-inverse de Moore-Penrose*) qui donne $x_\star = A^+ b$ défini par le Théorème 4. En particulier, on a toujours $A^+ Ax = x$ (car x minimise $|Az - Ax|$ sur les $z \in \mathbb{R}^m$). On pourrait montrer en reprenant l'argument donné au-dessus que $\|AA^+ - I_n\|_F = \min_Z \|AZ - I_n\|_F$ pour la norme de Frobenius. Enfin, comme $A^+ x$ satisfait l'équation normale, pour tous $x, y \in \mathbb{R}^n$, on a $A^\intercal(x - AA^+ x) \cdot y = 0 = (x - AA^+ x) \cdot Ay$, donc $A^\intercal(I_n - AA^+) = 0$ et AA^+ peut s'interpréter comme la projection orthogonale dans $\mathbb{R}^n$ sur $\mathrm{Ran}(A)$. ∎

3 Sensibilité aux données

Une difficulté pratique réside dans la très grande sensibilité aux variations des coefficients. Lorsqu'elle est possible (condition de rang maximal, dont on a vu qu'elle impose $\mathrm{rank}(A) = m \leq n$), la résolution du système linéaire (2.10) est affectée d'un conditionnement en σ_1^2/σ_m^2, très défavorable quand la matrice est proche d'être singulière. On va voir que la sensibilité aux données de la SVD est moins pénalisante. On suppose que A est de rang r et, par analogie avec le cas où A est une matrice inversible, on pose $\mathrm{cond}(A) = \frac{\sigma_1}{\sigma_r}$. On considère une variation δb du second membre : avec les notations introduites plus haut et en particulier la matrice pseudo-inverse A^+, on a $(x + \delta x) = A^+(b + \delta b) = x + A^+ \delta b$. Il s'ensuit que $|\delta x| \leq \|A^+\||\delta b|$, alors qu'on a toujours $|Ax| \leq \|A\||x|$. On en déduit

$$\frac{|\delta x|}{|x|} \leq \left(\|A\| \, \|A^+\| \, \frac{|b|}{|Ax|} \right) \frac{|\delta b|}{|b|}. \tag{3.17}$$

Or $\|A\| = \sigma_1$, $\|A^+\| = 1/\sigma_r$ et $|Ax| = |U\Sigma V^\intercal V\Sigma^+ U^\intercal b| = |\Sigma\Sigma^+ U^\intercal b| \leq |b|$. En posant $\cos(\theta) = \frac{|Ax|}{|b|}$, on en conclut que

$$\frac{|\delta x|}{|x|} \leq \frac{\mathrm{cond}(A)}{\cos(\theta)} \frac{|\delta b|}{|b|}. \tag{3.18}$$

Ainsi les erreurs sont amplifiées par deux effets : d'une part, le rapport des valeurs singulières, qui est cependant moins contraignant que son carré qui décrit le conditionnement de l'équation (2.10), et d'autre part le fait que $\cos(\theta) < 1$ lorsque $b \notin \mathrm{Ran}(A)$. ∎

L'étude de la sensibilité vis-à-vis des variations de la matrice A est, dans le cas où le rang n'est pas maximal, bien plus délicate. En particulier, la continuité de $A \mapsto A^+$ n'est même pas assurée. Pour s'en convaincre, on peut s'intéresser à l'exemple suivant. Avec

$$A = \begin{pmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 0 \end{pmatrix}, \qquad \delta A = \begin{pmatrix} 0 & 0 \\ 0 & \epsilon \\ 0 & 0 \end{pmatrix}. \tag{3.19}$$

Les matrices A et $A+\delta A$ sont déjà sous la forme "diagonale" et il n'y a pas à calculer de changements de bases. On obtient directement

$$A^+ = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}, \qquad (A+\delta A)^+ = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1/\epsilon & 0 \end{pmatrix}. \tag{3.20}$$

La difficulté et le défaut de continuité proviennent du fait que le rang de la matrice $A + \delta A$ change dans le régime $\epsilon \to 0$. Plus généralement, on peut montrer, lorsque $\mathrm{rank}(A+\delta A) > \mathrm{rank}(A)$, que

$$\|(A+\delta A)^+ - A^+\| \geq \frac{1}{\|\delta A\|}. \tag{3.21}$$

En effet, le fait que $\mathrm{rank}(A+\delta A) > \mathrm{rank}(A)$ entraîne qu'il existe $y \in \mathrm{Ran}(A+\delta A)$ tel que $|y| = 1$ et y est orthogonal à $\mathrm{Ran}(A)$. Il s'ensuit que

$$\begin{array}{lll} 1 & = y^\intercal y & \text{car } y \text{ est normalisé} \\ & = y^\intercal (A+\delta A)(A+\delta A)^+ y & \text{car } (A+\delta A)(A+\delta A)^+ \text{ est le projecteur sur } \mathrm{Ran}(A+\delta A) \\ & & \text{et } y \in \mathrm{Ran}(A+\delta A) \\ & = y^\intercal \delta A (A+\delta A)^+ y & \text{car } y \text{ est orthogonal à } \mathrm{Ran}(A) \\ & \leq \|\delta A\| |(A+\delta A)^+ y| & \text{par l'inégalité de Cauchy-Schwarz.} \end{array} \tag{3.22}$$

Comme $y \in (\mathrm{Ran}(A))^\perp = \mathrm{Ker}(A^\intercal)$, l'équation normale associée devient $A^\intercal A x = A^\intercal y = 0$; on a $A^+ y = 0$ et on en déduit l'inégalité annoncée

$$\|(A+\delta A)^+ - A^+\| \geq |(A+\delta A)^+ y - A^+ y| = |(A+\delta A)^+ y| \geq \frac{1}{\|\delta A\|}.$$

∎

Théorème 5 *Soit $N > M$ et $A \in \mathcal{M}_{N,M}$ qu'on suppose de rang maximal (*$\mathrm{rank}(A) = M$*). On désigne par x le minimiseur de J. On note $x+\delta x$ la solution du problème de minimisation associée à $A+\delta A$ et $b+\delta b$ où on suppose que $\epsilon = \max\left(\frac{\|\delta A\|}{\|A\|}, \frac{|\delta b|}{|b|}\right) \ll 1$. On pose $\sin(\theta) = \frac{|Ax-b|}{|b|}$ et $\kappa_A = \frac{\sigma_1}{\sigma_M}$. On a*

$$\frac{|\delta x|}{|x|} \leq \kappa_A\Big(\frac{\|\delta A\|}{\|A\|} + \frac{1}{\cos(\theta)}\frac{|\delta b|}{|b|}\Big) + \kappa_A^2 \tan(\theta)\frac{\|\delta A\|}{\|A\|} + \mathcal{R}(\epsilon), \tag{3.23}$$

où il existe une constante $C > 0$ telle que $|\mathcal{R}(\epsilon)| \leq C\epsilon^2$.

Preuve. En développant $(A + \delta A)^*(A + \delta A)(x + \delta x) = (A + \delta A)^*(b + \delta b)$, on obtient

$$A^*A\delta x = -A^*\delta A x + A^*\delta b + \delta A^*(b - Ax) + R(\epsilon). \tag{3.24}$$

Le terme de reste vérifie bien $|R(\epsilon)| \leq C\epsilon^2$. Or, la décomposition SVD permet d'établir que $\|A\| = \sigma_1$, $\|(A^*A)^{-1}\| = \frac{1}{\sigma_M^2}$ et $\|(A^*A)^{-1}A^*\| = \frac{1}{\sigma_M}$. On conclut en utilisant la définition de θ. En effet, on a, en exploitant l'équation normale, $|Ax - b|^2 = (Ax - b) \cdot (Ax - b) = A^*(Ax - b) \cdot x - (Ax-b)\cdot b = -(Ax-b)\cdot b \leq |Ax-b|\ |b|$ d'où on déduit que $\frac{|Ax-b|}{|b|} \leq 1$. Il est donc légitime de noter $\sin(\theta)$ cette quantité. Cette définition est cohérente avec ce qui précède car on peut décomposer $b = b - Ax + Ax$ avec $(Ax - b) \cdot Ax = (A^*Ax - A^*b) \cdot x = 0$: cette décomposition est orthogonale donc $|b|^2 = |Ax - b|^2 + |Ax|^2$ et on peut bien poser $\sin(\theta) = \frac{|Ax-b|}{|b|}$, $\cos(\theta) = \frac{|Ax|}{|b|}$. On déduit de (3.24) que

$$\frac{|\delta x|}{|x|} \leq \frac{|(A^*A)^{-1}A^*(\delta A x + \delta b)|}{|x|} + \frac{|(A^*A)^{-1}\delta A^*(b - Ax)|}{|x|} + \frac{|(A^*A)^{-1}R(\epsilon)|}{|x|}. \tag{3.25}$$

Le dernier terme produit le terme de reste d'ordre ϵ^2. Pour les termes dominants, on peut encore exploiter le fait que $\frac{1}{|x|} \leq \frac{\|A\|}{|Ax|}$. On obtient

$$\frac{|(A^*A)^{-1}\delta A^*(b - Ax)|}{|x|} \leq \frac{\|A\|}{|Ax|} \times \frac{\|\delta A\|}{\sigma_M^2} \times |b-Ax| = \frac{\|A\|^2}{\sigma_M^2} \times \frac{\|\delta A\|}{\|A\|} \times \frac{|b - Ax|}{|Ax|} = \frac{\sigma_1^2}{\sigma_M^2} \times \frac{\|\delta A\|}{\|A\|} \times \tan(\theta).$$

On procède de même pour estimer

$$\frac{|(A^*A)^{-1}A^*(\delta A x + \delta b)|}{|x|} \leq \frac{\|A\|}{\sigma_M} \times \left(\frac{\|\delta A\|}{\|A\|} + \frac{|\delta b|}{|Ax|}\right) = \frac{\sigma_1}{\sigma_M} \times \left(\frac{\|\delta A\|}{\|A\|} + \frac{|\delta b|}{\cos(\theta)|b|}\right)$$

■

Cette estimation montre que dans certaines situations, quand $\theta \to 0$, on peut obtenir des propriétés de stabilité par rapport aux fluctuations des données meilleures que celles attachées au système (2.10) et à la matrice A^*A.

4 Aspects numériques

La démonstration de la SVD et de ses propriétés fait intervenir la matrice $A^\intercal A$. Toutefois, en pratique, il vaut mieux éviter de former ce produit afin de préserver une précision numérique

satisfaisante. Des situations pathologiques sont décrites par les matrices de Läuchli comme

$$A = \begin{pmatrix} 1 & 1 & 1 \\ \epsilon & 0 & 0 \\ 0 & \epsilon & 0 \\ 0 & 0 & \epsilon \end{pmatrix}$$

où $0 < \epsilon \ll 1$. On trouve

$$A^\intercal A = \begin{pmatrix} 1+\epsilon^2 & 1 & 1 \\ 1 & 1+\epsilon^2 & 1 \\ 1 & 1 & 1+\epsilon^2 \end{pmatrix}.$$

Avec $a = (1+\epsilon^2)$, le polynôme caractéristique s'exprime

$$\begin{aligned}(a-\lambda)((a-\lambda)^2-1) - 2((a-\lambda)-1) &= ((a-\lambda)-1)((a-\lambda)((a-\lambda)+1)-2) \\ &= ((a-\lambda)-1)^2(a-\lambda+2) = (\epsilon^2-\lambda)^2(3+\epsilon^2-\lambda).\end{aligned}$$

Le spectre de cette matrice est $(3+\epsilon^2, \epsilon^2, \epsilon^2)$. Comme ϵ^2 est très petit, $3+\epsilon^2$ est numériquement considéré comme 1, et le spectre devient $(3, 0, 0)$. Un algorithme SVD performant est robuste pour traiter ces petites valeurs singulières. Pour illustrer cette difficulté on peut procéder au test suivant sur Matlab : la commande `A = gallery('lauchli',5,.001)` construit la matrice de Lauchli 6×5 de paramètre $\epsilon = .001$:

$$A = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 \\ \epsilon & 0 & 0 & 0 & 0 \\ 0 & \epsilon & 0 & 0 & 0 \\ 0 & 0 & \epsilon & 0 & 0 \\ 0 & 0 & 0 & \epsilon & 0 \\ 0 & 0 & 0 & 0 & \epsilon \end{pmatrix}.$$

La commande `svd(A)` renvoie le spectre 2.2361, 0.0010, 0.0010, 0.0010, 0.0010 alors que la commande `eigs(A'*A)` trouve 5, 0, 0 , 0 , 0. Pour exploiter au mieux la structure du problème, on a donc recours à des méthodes numériques qui construisent (une approximation de) la décomposition SVD, sans passer par la matrice $A^\intercal A$. Ces méthodes sont accessibles sur les logiciels de simulation (commande `svd` de `scilab` ou `np.linalg.svd` de `Python` par exemple). On détaille une approche possible ; plus de détails peuvent être trouvés dans [7, Chapter 5].

Rappel : décomposition QR d'une matrice.

La procédure d'orthonormalisation de Gram-Schmidt appliquée aux vecteurs colonnes d'une matrice $A \in \mathcal{M}_n$ fournit une décomposition sous la forme $A = QR$ où $Q \in O_n$ est une matrice orthogonale et R est une matrice triangulaire supérieure à coefficients diagonaux positifs. Dans le cas où A est inversible, une telle décomposition QR est alors unique.

Dans le cas particulier où A est bidiagonale inférieure (ses seuls coefficients non nuls sont sur et au dessous de la diagonale) ou supérieure, le calcul de sa décomposition QR peut s'effectuer en $\mathcal{O}(n)$ opérations seulement. Ce calcul à coût réduit s'effectue en cherchant la matrice Q sous la forme d'un produit $Q = C_1 \dots C_{n-1}$ où les matrices C_i, appelées *matrices de Givens*, sont des matrices

de rotation. Précisément C_i est une rotation dans le plan engendré par les i-ème et $(i+1)$-ème vecteurs de la base canonique de $\mathbb{R}^n$, avec un angle θ_i à choisir de manière pertinente.

Le calcul de la SVD d'une matrice exploite ces techniques. Dans un premier temps, on écrit une décomposition de la forme $A = U_1 B V_1^\intercal$ où U_1, V_1 sont des matrices orthogonales et B est une matrice bidiagonale. La deuxième étape calcule la SVD d'une matrice bidiagonale. Ces deux étapes sont à la fois robustes aux erreurs sur les données et rapides.

Calcul de la SVD

L'objectif est de déterminer la SVD d'une matrice A, sans calculer les produits matriciels $A\,A^\intercal$ et $A^\intercal\,A$ coûteux en temps de calcul et en place mémoire et aussi sources d'instabilité.

Pour simplifier les notations, on suppose dorénavant que $n \geq m$. La démarche procède en deux étapes : d'abord transformer A en une matrice bidiagonale, via des produits avec des matrices orthogonales, ensuite appliquer la décomposition QR à cette matrice bidiagonale.

• *Étape 1: bidiagonalisation.*

On suit la construction suivante :

Choisir $v_1 \in \mathbb{R}^m$ unitaire quelconque et poser $\beta_0 = 0$, $u_0 = 0 \in \mathbb{R}^n$

Pour $i = 1$ jusqu'à $m-1$, faire

$$\begin{array}{lll} \tilde{u}_i = Av_i - \beta_{i-1}u_{i-1}, & \alpha_i = |\tilde{u}_i|, & u_i = \frac{\tilde{u}_i}{\alpha_i}, \\ \tilde{v}_{i+1} = A^\intercal u_i - \alpha_i v_i, & \beta_i = |\tilde{v}_{i+1}|, & v_{i+1} = \frac{\tilde{v}_{i+1}}{\beta_i}. \end{array}$$

Poser $\tilde{u}_m = Av_m - \beta_{m-1}u_{m-1}$, calculer $\alpha_m = \|\tilde{u}_m\|_2$ puis $u_m = \frac{\tilde{u}_m}{\alpha_m}$.

Si $n > m$, compléter $(u_1, ..., u_m)$ en une base orthonormée.

On a alors le résultat suivant :

Lemme 3 *Si tous les coefficients α_i et β_i sont non nuls alors les matrices $\widetilde{U} = (u_1 \dots u_n)$ et $\widetilde{V} = (v_1 \dots v_m)$ données par l'algorithme ci-dessus sont orthogonales et de plus, on a*

$$\widetilde{U}\,A\,\widetilde{V}^\intercal = \begin{pmatrix} \alpha_1 & \beta_1 & 0 & \cdots & \cdots \\ 0 & \alpha_2 & \beta_2 & 0 & \cdots \\ \vdots & \ddots & \ddots & \ddots & \ddots \\ \vdots & & \ddots & \alpha_{m-1} & \beta_{m-1} \\ \vdots & & & \ddots & \alpha_m \\ 0 & \cdots & \cdots & \cdots & 0 \\ \vdots & & & & \vdots \end{pmatrix} \tag{4.26}$$

En effet, on établit par récurrence que u_i est orthogonal à tous les u_j précédents, et de même pour v_{i+1}. Le fait que $\widetilde{U}A\widetilde{V}^\intercal$ soit bidiagonale est alors une simple récriture des équations de l'algorithme.

Si $\alpha_i = 0$, il faut choisir u_i quelconque de norme 1 et orthogonal à tous les vecteurs $u_1, ..., u_{i-1}$. Si $\beta_i = 0$, on choisit v_{i+1} de norme 1 et orthogonal à tous les vecteurs $v_1, ..., v_{i-1}$.

• *Étape 2 : SVD d'une matrice B bidiagonale*

Par l'étape précédente on s'est ramené au cas d'une matrice bidiagonale supérieure B. Dans le terme de droite de (4.26), que l'on note maintenant B, les $n - m$ dernières lignes ne jouent aucun rôle dans la suite ; on se ramène donc au cas où B est bidiagonale carrée (i.e. $m = n$). La structure bidiagonale de B simplifie les calculs.

On pose $B_1 = B$

Jusqu'à convergence, faire :

Calculer la décomposition QR de $B_k^\intercal = Q_k R_k$.

Calculer la décomposition QR de $R_k^\intercal = \widetilde{Q}_k B_{k+1}$.

S'arrêter quand les coefficients sur-diagonaux de B_{k+1} sont assez petits.

La matrice B_{k+1} finale contient des approximations des valeurs singulières de B et les matrices orthogonales U et V correspondantes s'obtiennent à partir des matrices $Q_1, ..., Q_k$ et $\widetilde{Q}_1, ..., \widetilde{Q}_k$. Chaque étape de l'algorithme s'effectue en $\mathscr{O}(n)$ opérations.

5 Applications

5.1 Reconstruction

Exemple 1

On considère un signal

$$S(t) = \sum_{j=1}^{J} \varsigma_j e^{i\omega_j t}.$$

On effectue des mesures sur des temps discrets $t_1, .., t_L$. Ces mesures enregistrent en fait un signal perturbé $\tilde{S} = S + \epsilon$. On souhaite reconstruire le véritable signal à partir de ces évaluations perturbées. La stratégie (heuristique) est la suivante :

- On commence par construire une matrice $H \in \mathcal{M}_{NM}$ avec $M + N = L + 1$, en ordonnant les signaux enregistrés — que l'on note s_ℓ pour $\ell \in \{1, ..., L\}$, perturbations de $s(t_\ell) = s(\ell\delta t)$ — de manière particulière

$$H = \begin{pmatrix} s_1 & s_2 & s_3 & \cdots & & s_{M-1} & s_M \\ s_2 & s_3 & \cdots & \cdots & s_{M-1} & s_M & s_{M+1} \\ \vdots & \vdots & & & & & \vdots \\ \vdots & \vdots & & & & & \vdots \\ s_{N-1} & s_N & & \cdots & & s_{L-2} & s_{L-1} \\ s_N & s_{N+1} & & \cdots & & s_{L-1} & s_L \end{pmatrix}$$

Le signal enregistré est ainsi contenu sur l'ensemble (repéré en rouge) formé par la première ligne et la dernière colonne : la première ligne contient les M premières données et elle est suivie par $N-1$ autres lignes qui contiennent chacune une nouvelle donnée sur la dernière colonne. On obtient donc $L = M + N - 1$. Cette matrice a de plus une structure très particulière, dite de Hankel : le même terme est reproduit sur toutes les anti-diagonales.

- On détermine la SVD de cette matrice : $H = U\Sigma V^\intercal$.
- On introduit un seuil où on estime que les valeurs singulières ne sont plus significatives et on en déduit une SVD de rang r réduit $H_r = U_r\Sigma_r V_r^\intercal$, où la matrice Σ_r ne contient que r éléments (diagonaux) non nuls.
- La matrice H_r ainsi obtenue est la meilleure approximation de rang r de H (pour la norme de Frobenius) mais elle n'a plus la structure de Hankel. On la remplace par une nouvelle matrice $\tilde{H}_r$ dont les anti-diagonales sont obtenues en faisant la moyenne des termes des mêmes anti-diagonales de H_r. (C'est la méthode SSA, pour Singular Spectrum Analysis [9]. La matrice $\tilde{H}_r$ a en général un rang $\geq r$; de celle-ci on peut reconstruire une matrice SVD réduite de rang r, puis définir une nouvelle matrice de Hankel par moyennisation des termes anti-diagonaux, etc. Cette procédure itérative est connue comme la méthode de Cadzow [3].)
- À partir de la première ligne et de la dernière colonne de $\tilde{H}_r$, on dispose de L données corrigées, qui définissent un signal filtré : ces éléments sont compris comme les évaluations $s_r(t_\ell)$ du signal reconstruit aux instants t_ℓ.

La motivation de la méthode vient du fait que le bruit est censé être d'amplitude plus réduite que le signal brut : le tri procuré par la SVD devrait donc sélectionner plutôt les composantes du signal brut. Plus précisément, la matrice de Hankel associée au signal brut est de rang "petit", contrôlé par le nombre de modes J (qui en général n'est pas connu a priori, mais qu'on estime réduit par rapport aux paramètres N et M). On note

$$Z_j = e^{i\omega_j \delta t}.$$

Puis on introduit le polynôme P de degré J qui a pour racines $Z_1, ..., Z_J$, c'est-à-dire

$$P(z) = \prod_{j=1}^{J} (z - Z_j).$$

On écrit ce polynôme sous la forme

$$P(z) = z^J + \sum_{k=1}^{J} \pi_k z^{J-k},$$

pour certains coefficients $\pi_k \in \mathbb{C}$. En particulier, il résulte du fait que $P(Z_j) = 0$ que

$$Z_j^J = -\sum_{k=1}^{J} \pi_k Z_j^{J-k}$$

et même, pour tout $n \geq J$,

$$Z_j^n = -\sum_{k=1}^{J} \pi_k Z_j^{n-k}.$$

Il s'ensuit que

$$\begin{aligned} S(n\delta t) &= \sum_{j=1}^{J} \varsigma_j Z_j^n = \sum_{j=1}^{J} \varsigma_j \left(\sum_{k=1}^{J} \pi_k Z_j^{n-k} \right) \\ &= \sum_{k=1}^{J} \pi_k \underbrace{\left(\sum_{j=1}^{J} \varsigma_j Z_j^{n-k} \right)}_{S((n-k)\delta t)} \end{aligned}$$

qui prouve que pour $n > J$, $S(n\delta t)$ s'exprime comme combinaison linéaire des valeurs antérieures $S(\delta t), S(2\delta t), ..., S(J\delta t)$. En conséquence, les lignes et colonnes d'indice "assez grand" de la matrice de Hankel construites avec les $S(n\delta t)$ s'expriment comme combinaisons linéaires des précédentes : le rang de cette matrice doit donc, en pratique, être sensiblement inférieur à N ou M. Deux paramètres règlent la méthode : la répartition du signal dans la matrice de Hankel, contrôlé par le paramètre M, et le seuil qui fixe le rang des matrices réduites.

Pour les simulations reproduites dans la Fig. 1, le signal original est $S(t) = \frac{1}{5}\sum_{j=1}^{5} \sin(2\pi\omega_j t)$ où le vecteur des fréquences est $((\sqrt{5}-1)/2, 1, 2/(\sqrt{5}-1), e, \pi)$. Il y a $L = 2048$ points d'acquisition, avec un pas de temps de $\delta t = 0.05$. Le seuil retenu ne conserve que 8 modes de la SVD. L'erreur relative au signal original est de 0.5121, celle au signal bruité de 2.5151.

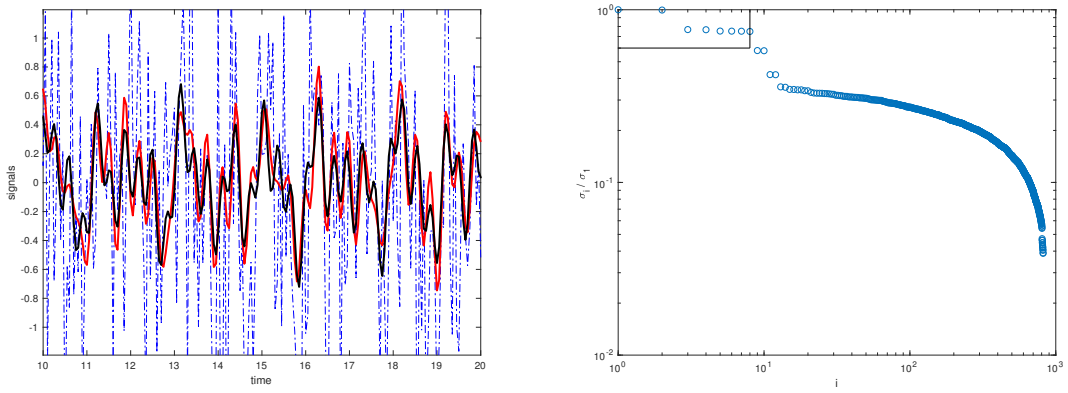


Figure 1: À gauche : signal original (rouge), bruité (pointillés) et filtré (noir) et répartition des valeurs singulières (à droite)

Exemple 2

On cherche à identifier la source $(t, x) \mapsto f(t, x)$ d'un signal qui se propage dans le tore $[-\pi, \pi]$.

Pour simplifier, on suppose connue la fréquence temporelle ω de cette source :

$$f(t,x) = e^{i\omega t} g(x)$$

et l'enjeu consiste donc à identifier la fonction g à partir de mesures de $(t,x) \mapsto u(t,x)$ solution de

$$\partial_{tt}^2 u - c^2 \partial_{xx}^2 u = f \quad \text{pour } t \geq 0,\ -\pi < x < \pi$$

avec les données initiales

$$(u, \partial_t u)\big|_{t=0} = 0.$$

En cherchant une solution qui oscille à la même fréquence ω que la source, c'est-à-dire $u(t,x) = e^{i\omega t} v(x)$ on trouve

$$-\omega^2 v - c^2 \partial_{xx}^2 v = g. \tag{5.27}$$

Il est possible de résoudre (5.27) par une méthode numérique (différences finies, éléments finis, etc). Mais ici on peut exploiter la périodicité du problème pour raisonner en termes de coefficients de Fourier. En effet, les coefficients de Fourier de v et de g sont liés par la relation

$$-(\omega^2 - c^2 k^2)\widehat{v}(k) = \widehat{g}(k),$$

qui fait apparaître une condition de *non-résonnance* : $\omega^2 \neq c^2 k^2$ lorsque $\widehat{g}(k) \neq 0$. Pour exprimer v sous forme d'une convolution avec g, il reste à identifier la fonction 2π-périodique a telle que

$$\widehat{a}(k) = \frac{1}{\omega^2 - c^2 k^2} = \frac{1}{\omega^2} \frac{1}{1 - c^2 k^2/\omega^2},$$

c'est-à-dire la solution élémentaire de $(\omega^2 + c^2 \partial_{xx}^2) T = \delta_{x=0}$ avec conditions de périodicité. À cette fin, on remarque d'abord que

$$T_0 : x \longmapsto \frac{e^{i|x|\omega/c}}{2i\omega c}$$

est localement intégrable sur $\mathbb{R}$ et vérifie (au sens des distributions)

$$\frac{\mathrm{d}}{\mathrm{d}x} T_0(x) = \frac{e^{i|x|\omega/c}}{2c^2} \mathrm{sgn}(x), \qquad \frac{\mathrm{d}^2}{\mathrm{d}x^2} T_0(x) = \frac{e^{i|x|\omega/c}}{2c^2} 2\delta_{x=0} + \frac{i\omega e^{i|x|\omega/c}}{2c^3} \mathrm{sgn}^2(x) = \frac{1}{c^2}(\delta_{x=0} - \omega^2 T_0(x)).$$

Toutefois cette fonction n'est pas 2π-périodique. On cherche donc a sous la forme $a = T_0 + T_h$ où T_h est solution de $\omega^2 T_h + c^2 T_h'' = 0$ de sorte que a soit 2π-périodique. On a

$$T_h(x) = \lambda e^{ix\omega/c} + \mu e^{-ix\omega/c}$$

et on choisit λ, μ pour que

$$\begin{aligned}
&a(-\pi) = a(\pi) = \frac{e^{i\pi\omega/c}}{2i\omega c} + \lambda e^{-i\pi\omega/c} + \mu e^{i\pi\omega/c} = \frac{e^{i\pi\omega/c}}{2i\omega c} + \lambda e^{i\pi\omega/c} + \mu e^{-i\pi\omega/c}, \\
&\frac{\mathrm{d}}{\mathrm{d}x} a(-\pi) = \frac{\mathrm{d}}{\mathrm{d}x} a(\pi) = -\frac{e^{i\pi\omega/c}}{2c^2} + \frac{i\omega}{c}\left(\lambda e^{-i\pi\omega/c} - \mu e^{i\pi\omega/c}\right) = \frac{e^{i\pi\omega/c}}{2c^2} + \frac{i\omega}{c}\left(\lambda e^{i\pi\omega/c} - \mu e^{-i\pi\omega/c}\right).
\end{aligned}$$

La résolution de ce système linéaire conduit à

$$\lambda = \mu = \frac{e^{i\pi\omega/c}}{4\omega c \sin(\pi\omega/c)}$$

et finalement

$$a(x) = \frac{e^{i|x|\omega/c}}{2i\omega c} + \frac{e^{i\pi\omega/c}\cos(x\omega/c)}{2\omega c \sin(\pi\omega/c)}.$$

Autrement dit, on considère le filtre de noyau $-a$:

$$A : g \longmapsto A[g](x) = -\int_{-\pi}^{\pi} a(x-y)g(y)\,\mathrm{d}y.$$

On raisonne en termes de signal discret, en remplaçant l'intégrale par une somme discrète

$$A_M[g](x) = h\sum_{m=0}^{M-1} a(x-\alpha_m)g(\alpha_m), \qquad \alpha_m = -\pi + mh, \qquad h = 2\pi/M.$$

On dispose de mesures de ce signal $Y_1, ..., Y_N$, réalisés aux points d'échantillonnage $x_1, ..., x_N$ et on cherche donc la fonction g telle que les échantillons $A_M[g](x_1), ..., A_M[g](x_N)$ soient au plus proche des mesures $Y_1, ..., Y_N$. Cette question se formule sous la forme de la minimisation de

$$J : (g_1, ..., g_M) \in \mathbb{C}^M \longmapsto \frac{|Ag - Y|^2}{2}$$

où $Y = (Y_1, ..., Y_N) \in \mathbb{C}^N$ et A est la matrice $N \times M$ de composantes

$$A_{nm} = -ha(x_n - \alpha_m).$$

La méthode SVD permet de résoudre ce problème.

En pratique, les mesures sont entâchées d'erreur. La méthode SVD permet aussi de filtrer ces erreurs en ne retenant que les modes les plus significatifs. On définit un seuil (par exemple en fonction d'un pourcentage de la plus grande valeur singulière) et on utilise la SVD d'un rang réduit en fonction de ce seuil.

Les figures suivantes illustrent cette démarche. Ici la fonction à reconstituer est (voir Fig. 2)

$$g(x) = \ln(2 + \sin(3x)).$$

La figure 2 montre le graphe de cette fonction et de la solution de (5.27) correspondante. La figure 3 montre les reconstructions obtenues par la méthode SVD à partir de la solution de l'EDP, et en perturbant celle-ci par des variations aléatoires d'amplitude maximale 10^{-4}. Les valeurs singulières sont aussi représentées à la figure 3 : elles décroissent rapidement et on peut espérer obtenir une reconstruction moins sensible aux perturbations en ne considérant qu'une partie de ses valeurs singulières. C'est ce que montre la figure 4 : le seuil est fixé en sélectionnant les valeurs singulières $\geq \sigma_1/(2 * c)$ avec $c \in \{1, 2, ..., 6\}$. Ces seuils correspondent à ne prendre que 2, 6, 9, 11, 11 ou 13 modes sur les 99 définis dans cette simulation.

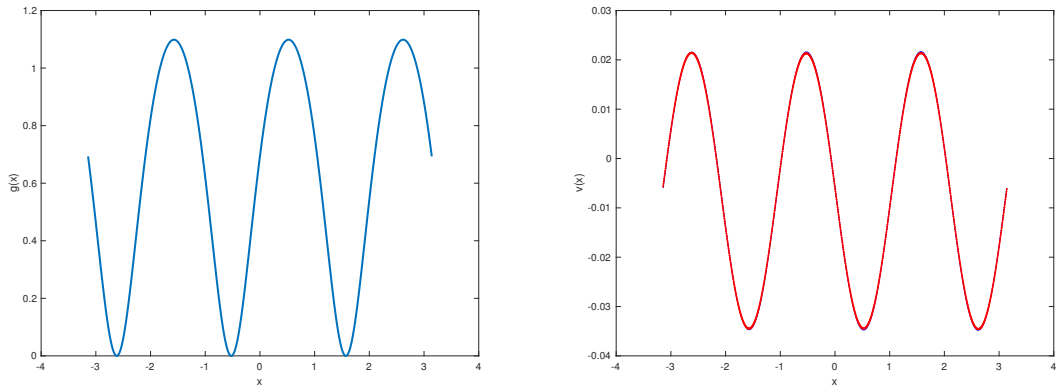


Figure 2: Donnée à reconstituer (à gauche) et solution de l'EDP correspondante (à droite)

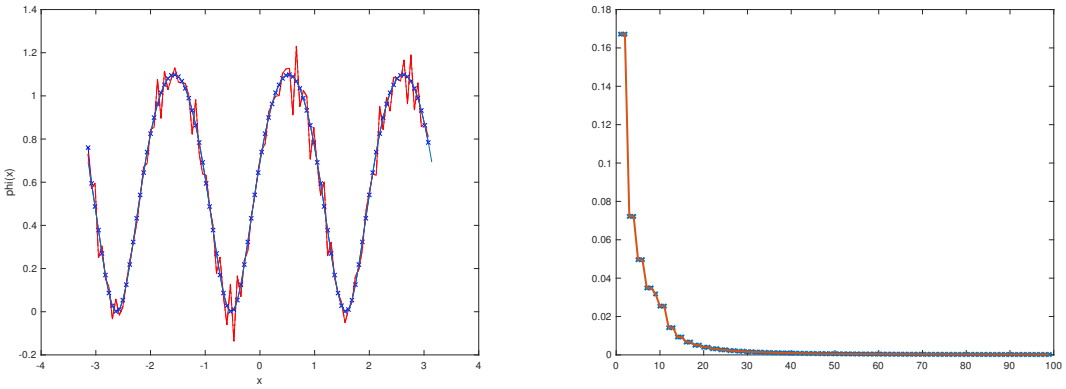


Figure 3: Reconstruction à partir de la solution de l'EDP (à gauche) et valeurs singulières (à droite)

5.2 Compression

Motivation.

Une image peut être naturellement représentée comme une matrice : l'élément a_{ij} décrit le niveau de coloration d'un élément de surface — ou *pixel* — dont le centre est caractérisé par les coordonnées discrètes (i, j). Pour une image en noir et blanc, a_{ij} prend ses valeurs dans $\{0, ..., 255\}$, correspondants aux niveaux de gris, codé sur 8 bits (un pixel totalement blanc est représenté par 11111111). Les images TV sont ainsi décrites par une matrice de taille 1080×1920 ; comme il y a 3 échelles de couleur et 24 images par secondes cela représente $3 \times 8 \times 1080 \times 1920 \times 24 \simeq 1.2 \cdot 10^9$ bits par seconde. La quantité d'information à transmettre devient vite vertigineuse et de nature à compromettre toute transmission en des temps assez réduits pour les applications visées. Les techniques de compression permettent de réduire substantiellement ces volumes. Elles reposent sur l'idée qu'il n'est pas nécessaire de transmettre toute l'information pour reconstruire une image

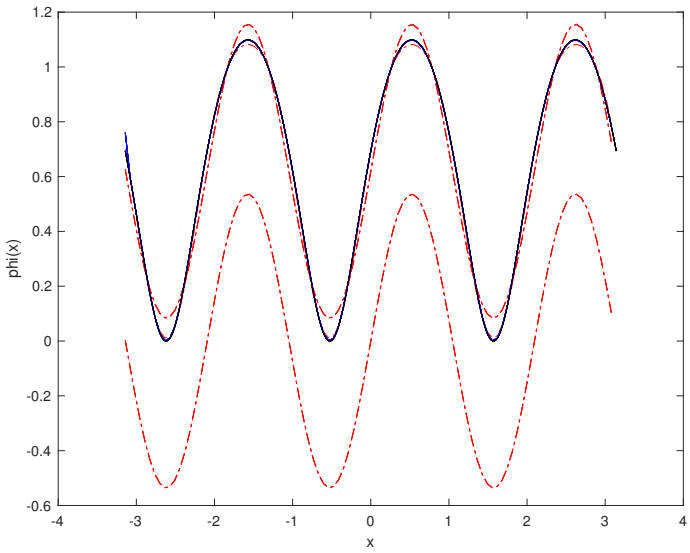


Figure 4: Reconstruction avec filtrage par rang partiel

satisfaisante, deux pixels voisins n'étant pas, en général, indépendamment remplis. De même, pour des images vidéos, les changements entre deux images successives restent modérés. L'enjeu consiste donc à repérer l'information redondante pour ne conserver que celle qui est véritablement pertinente. Par exemple, en désignant par e le vecteur dont toutes les coordonnées valent 1, le drapeau français peut être décrit sous la forme d'un simple produit colonne/ligne $e(b...bw...wr...r)$, qui réclame moins de stockage que l'ensemble de la matrice (pour une image de $n \times m$ pixels, comme le drapeau français est de rang 1 (!), il suffit de stocker les n coordonnées de e et les m coordonnées de $(b...bw...wr...r)$, c'est-à-dire $n+m$ valeurs seulement). L'idée consiste à interpréter le *rang* de la matrice comme une mesure de la redondance d'information et à décomposer une image sous forme de tels produits colonne/ligne.

Lemme 4 *La matrice $A \in \mathcal{M}_{nm}(\mathbf{R})$ est de rang 1 si et seulement si il existe deux vecteurs $u \in \mathbf{R}^n$, $v \in \mathbf{R}^m$, non nuls, tels que $A = uv^\intercal$.*

Preuve. La matrice $uv^\intercal$ est de rang un car $A\xi = v \cdot \xi \; u \in \mathrm{Span}(u)$. Réciproquement soit $A \in \mathcal{M}_{nm}$ de rang un. En particulier $\mathrm{Ran}(A)$ est engendré par un vecteur $u \in \mathbb{R}^n \setminus \{0\}$ et pour tout $\xi \in \mathbb{R}^m$, on a $A\xi = \ell(\xi)u$, $\ell(\xi) \in \mathbb{R}$. On définit le vecteur $v \in \mathbb{R}^m$ dont les coordonnées sont $\ell(e_j)$, $j \in \{1, ..., m\}$, les vecteurs e_j décrivant la base canonique de $\mathbb{R}^m$. Par définition, on a bien

$$A_{ij} = [Ae_j]_i = \ell(e_j)u_i = v_j u_i = [uv^\intercal]_{ij}.$$

■

Utilisation de la SVD.

La SVD peut se réinterpréter comme une écriture de la matrice A sous la forme d'une somme de matrices de rang 1

$$A = \sum_{i=1}^{r} \sigma_i u_i v_i^\intercal. \tag{5.28}$$

Cette expression suggère de compresser l'image en ne tenant pas compte des termes pour lesquels σ_i est "petit". Les théorème 2 et 3 donnent corps à cette approche : la SVD est "la meilleure" approximation de rang fixé.

Même si cette stratégie n'est pas la plus performante numériquement, cette observation conduit à mettre en œuvre la démarche suivante pour "compresser" une image.

1. On forme la matrice carrée $A^\intercal A$ (c'est le point faible de cette méthode) puis on détermine les éléments propres dominants de cette matrice, c'est-à-dire la plus grande valeur propre σ_1 et un vecteur propre normalisé v_1. L'*algorithme de la puissance* permet de trouver ces quantités. On pose alors $u_1 = \frac{Av_1}{\sigma_1}$. On dispose ainsi du premier terme de la décomposition $A_1 = \sigma_1 u_1 v_1^\intercal$.

2. On applique la même procédure à la matrice $A - A_1$...

3. On stoppe la construction lorsqu'on estime que la valeur de σ_i obtenue n'est plus significative.

Afin d'illustrer l'approche proposée, on présente les résultats de compression du portrait de la Figure 5 dans les Figures 6, la répartition des valeurs singulières étant donnée par la Figure 7.

Figure 5: Portrait, image de référence

La seconde illustration est encore plus spectaculaire et permet de montrer les limites de l'approche "naïve" présentée au-dessus. L'image originale, donnée dans la Fig. 8, a une taille 736×559. La Fig. 9 montre les compressions obtenues avec l'approche élémentaire qui calcule les éléments propres successifs de $A^\intercal A$ par la méthode de la puissance. On reconnaît des éléments marquants de l'image originale, mais la méthode ne parvient pas à progresser significativement. La raison provient des

Figure 6: Images compressées du portrait, avec 10, 30, 50 et 80 modes, soit un stockage de 3%, 9%, 15%, 24% de l'image originale

instabilités liées à la formation de $A^\intercal A$. En effet, la Fig. 10 reproduit les reconstitutions obtenues avec une routine de calcul de SVD optimisée, qui évite ce calcul d'un produit matriciel, fragilité de la méthode.

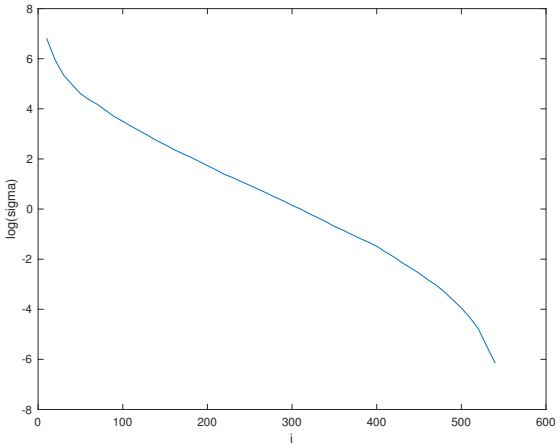


Figure 7: Evolution des valeurs singulières pour le portrait

Figure 8: Harry et Meghan : image originale

References

[1] The Netflix prize: Singular values for $1 million. `http://www.solverworld.com/the-netflix-prize-singular-values-for-1-million/`.

[2] M. Bhattacharyya. Beginners guide to creating the SVD recommender system, 2019.

[3] J. Cadzow. Signal enhancement: a composite property mapping algorithm. *IEEE Trans. Acoustics, Speech, and Signal Processing*, 36(2):49–82, 1988.

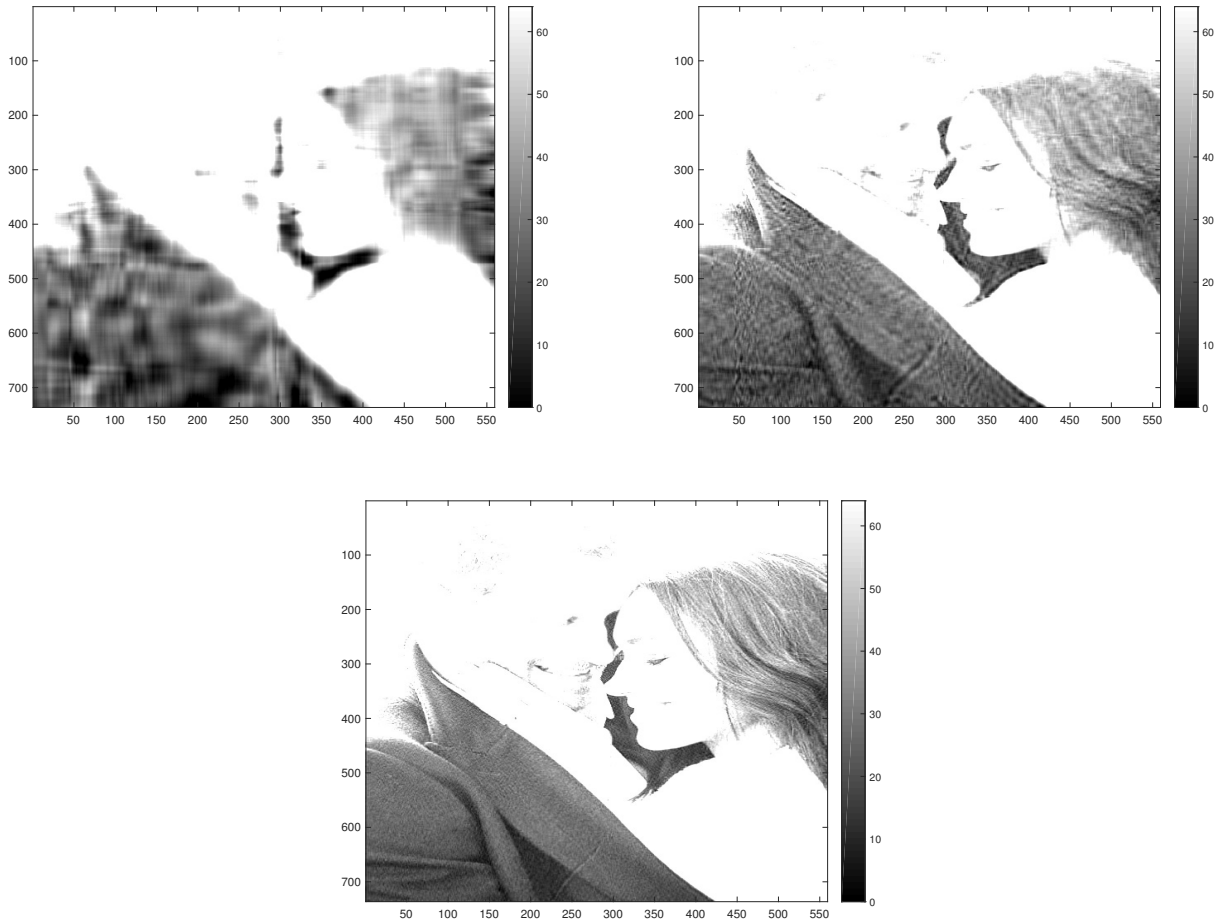


Figure 9: Images compressées du portrait de Harry et Meghan par l'approche naïve, avec 10, 50 et 150 modes, soit un stockage de 3%, 15%, 47% de l'image originale

[4] J. Cadzow. Signal processing via least squares error modeling. *IEEE ASSP Magazine*, pages 12–31, 1990.

[5] J. Cadzow. Least squares, modeling and signal processing. *Digital Signal Processing*, 4:2–20, 1994.

[6] C. Eckart. and G. Young. The approximation of one matrix by another of lower rank. *Psychometrika*, 1:211–218, 1936.

[7] G. H. Golub and C. F. Van Loan. *Matrix computations*. John Hopkins Studies in the Math. Sci. The John Hopkins Univ. Press, 1996. 3rd ed.

[8] G.H. Golub and C. Reinsch. Singular value decomposition and least squares solutions. *Numer. Math.*, 14:403–420, 1970.

Figure 10: Harry et Meghan : image reconstituée, avec 5, 10, 20, 40, 60 ou 100 modes

[9] N. Golyandina, V. Nekrutkin, and A. A. Zhigljavsky. *Analysis of Time Series Structure: SSA and related techniques*, volume 90 of *Monographs on Statistics and Applied Probability*. Champman-Hall, CRC, 2001.

[10] P. Lascaux and R. Théodor. *Analyse numérique matricielle appliquée à l'art de l'ingénieur. Tomes I & II*. Dunod, 2004.

[11] G. W. Stewart. On the early history of the singular value decomposition. *SIAM Rev.*, 35(4):551–566, 1993.

[12] G. Strang. *Linear Algebra and Its Applications*. Thomson Learning, Inc, 2006. 4th Ed.

A Data Scientist: The Sexiest Job of the 21st Century Meet the people who can coax treasure out of messy, unstructured data. by T. H. Davenport and D. J. Patil

Back in the 1990s, computer engineer and Wall Street "quant" were the hot occupations in business. Today data scientists are the hires firms are competing to make. As companies wrestle with unprecedented volumes and types of information, demand for these experts has raced...

When Jonathan Goldman arrived for work in June 2006 at LinkedIn, the business networking site, the place still felt like a start-up. The company had just under 8 million accounts, and the number was growing quickly as existing members invited their friends and colleagues to join. But users weren't seeking out connections with the people who were already on the site at the rate executives had expected. Something was apparently missing in the social experience. As one LinkedIn manager put it, "It was like arriving at a conference reception and realizing you don't know anyone. So you just stand in the corner sipping your drink — and you probably leave early." Goldman, a PhD in physics from Stanford, was intrigued by the linking he did see going on and by the richness of the user profiles. It all made for messy data and unwieldy analysis, but as he began exploring people's connections, he started to see possibilities. He began forming theories, testing hunches, and finding patterns that allowed him to predict whose networks a given profile would land in. He could imagine that new features capitalizing on the heuristics he was developing might provide value to users. But LinkedIn's engineering team, caught up in the challenges of scaling up the site, seemed uninterested. Some colleagues were openly dismissive of Goldman's ideas. Why would users need LinkedIn to figure out their networks for them? The site already had an address book importer that could pull in all a member's connections.

Luckily, Reid Hoffman, LinkedIn's cofounder and CEO at the time (now its executive chairman), had faith in the power of analytics because of his experiences at PayPal, and he had granted Goldman a high degree of autonomy. For one thing, he had given Goldman a way to circumvent the traditional product release cycle by publishing small modules in the form of ads on the site's most popular pages.

Through one such module, Goldman started to test what would happen if you presented users with names of people they hadn't yet connected with but seemed likely to knowfor example, people who had shared their tenures at schools and workplaces. He did this by ginning up a custom ad that displayed the three best new matches for each user based on the background entered in his or her LinkedIn profile. Within days it was obvious that something remarkable was taking place. The click-through rate on those ads was the highest ever seen. Goldman continued to refine how the suggestions were generated, incorporating networking ideas such as "triangle closing"the notion that if you know Larry and Sue, there's a good chance that Larry and Sue know each other. Goldman and his team also got the action required to respond to a suggestion down to one click.

It didn't take long for LinkedIn's top managers to recognize a good idea and make it a standard feature. That's when things really took off. "People You May Know" ads achieved a click-through rate 30% higher than the rate obtained by other prompts to visit more pages on the site. They generated millions of new page views. Thanks to this one feature, LinkedIn's growth trajectory shifted significantly upward.

A New Breed

Goldman is a good example of a new key player in organizations: the "data scientist". It's a high-ranking professional with the training and curiosity to make discoveries in the world of big data. The title has been around for only a few years. (It was coined in 2008 by one of us, D.J. Patil, and Jeff Hammerbacher, then the respective leads of data and analytics efforts at LinkedIn and Facebook.) But thousands of data scientists are already working at both start-ups and well-established companies. Their sudden appearance on the business scene reflects the fact that companies are now wrestling with information that comes in varieties and volumes never encountered before. If your organization stores multiple petabytes of data, if the information most critical to your business resides in forms other than rows and columns of numbers, or if answering your biggest question would involve a "mashup" of several analytical efforts, you've got a big data opportunity.

Much of the current enthusiasm for big data focuses on technologies that make taming it possible, including Hadoop (the most widely used framework for distributed file system processing) and related open-source tools, cloud computing, and data visualization. While those are important breakthroughs, at least as important are the people with the skill set (and the mind-set) to put them to good use. On this front, demand has raced ahead of supply. Indeed, the shortage of data scientists is becoming a serious constraint in some sectors. Greylock Partners, an early-stage venture firm that has backed companies such as Facebook, LinkedIn, Palo Alto Networks, and Workday, is worried enough about the tight labor pool that it has built its own specialized recruiting team to channel talent to businesses in its portfolio. "Once they have data", says Dan Portillo, who leads that team, "they really need people who can manage it and find insights in it". Who Are These People?

If capitalizing on big data depends on hiring scarce data scientists, then the challenge for managers is to learn how to identify that talent, attract it to an enterprise, and make it productive. None of those tasks is as straightforward as it is with other, established organizational roles. Start with the fact that there are no university programs offering degrees in data science. There is also little consensus on where the role fits in an organization, how data scientists can add the most value, and how their performance should be measured.

The first step in filling the need for data scientists, therefore, is to understand what they do in businesses. Then ask, What skills do they need? And what fields are those skills most readily found in?

More than anything, what data scientists do is make discoveries while swimming in data. It's their preferred method of navigating the world around them. At ease in the digital realm, they are able to bring structure to large quantities of formless data and make analysis possible. They identify rich data sources, join them with other, potentially incomplete data sources, and clean the resulting set. In a competitive landscape where challenges keep changing and data never stop flowing, data scientists help decision makers shift from ad hoc analysis to an ongoing conversation with data.

Data scientists realize that they face technical limitations, but they don't allow that to bog down their search for novel solutions. As they make discoveries, they communicate what they've learned and suggest its implications for new business directions. Often they are creative in displaying in-

formation visually and making the patterns they find clear and compelling. They advise executives and product managers on the implications of the data for products, processes, and decisions.

Given the nascent state of their trade, it often falls to data scientists to fashion their own tools and even conduct academic-style research. Yahoo, one of the firms that employed a group of data scientists early on, was instrumental in developing Hadoop. Facebook's data team created the language Hive for programming Hadoop projects. Many other data scientists, especially at data-driven companies such as Google, Amazon, Microsoft, Walmart, eBay, LinkedIn, and Twitter, have added to and refined the tool kit.

What kind of person does all this? What abilities make a data scientist successful? Think of him or her as a hybrid of data hacker, analyst, communicator, and trusted adviser. The combination is extremely powerfuland rare.

Data scientists' most basic, universal skill is the ability to write code. This may be less true in five years' time, when many more people will have the title "data scientist" on their business cards. More enduring will be the need for data scientists to communicate in language that all their stakeholders understandand to demonstrate the special skills involved in storytelling with data, whether verbally, visually, orideallyboth.

But we would say the dominant trait among data scientists is an intense curiositya desire to go beneath the surface of a problem, find the questions at its heart, and distill them into a very clear set of hypotheses that can be tested. This often entails the associative thinking that characterizes the most creative scientists in any field. For example, we know of a data scientist studying a fraud problem who realized that it was analogous to a type of DNA sequencing problem. By bringing together those disparate worlds, he and his team were able to craft a solution that dramatically reduced fraud losses.

Perhaps it's becoming clear why the word "scientist" fits this emerging role. Experimental physicists, for example, also have to design equipment, gather data, conduct multiple experiments, and communicate their results. Thus, companies looking for people who can work with complex data have had good luck recruiting among those with educational and work backgrounds in the physical or social sciences. Some of the best and brightest data scientists are PhDs in esoteric fields like ecology and systems biology. George Roumeliotis, the head of a data science team at Intuit in Silicon Valley, holds a doctorate in astrophysics. A little less surprisingly, many of the data scientists working in business today were formally trained in computer science, math, or economics. They can emerge from any field that has a strong data and computational focus.

It's important to keep that image of the scientist in mind – because the word "data" might easily send a search for talent down the wrong path. As Portillo told us, "The traditional backgrounds of people you saw 10 to 15 years ago just don't cut it these days." A quantitative analyst can be great at analyzing data but not at subduing a mass of unstructured data and getting it into a form in which it can be analyzed. A data management expert might be great at generating and organizing data in structured form but not at turning unstructured data into structured data – and also not at actually analyzing the data. And while people without strong social skills might thrive in traditional data professions, data scientists must have such skills to be effective.

Roumeliotis was clear with us that he doesn't hire on the basis of statistical or analytical capabilities. He begins his search for data scientists by asking candidates if they can develop

prototypes in a mainstream programming language such as Java. Roumeliotis seeks both a skill seta solid foundation in math, statistics, probability, and computer scienceand certain habits of mind. He wants people with a feel for business issues and empathy for customers. Then, he says, he builds on all that with on-the-job training and an occasional course in a particular technology.

Several universities are planning to launch data science programs, and existing programs in analytics, such as the Master of Science in Analytics program at North Carolina State, are busy adding big data exercises and coursework. Some companies are also trying to develop their own data scientists. After acquiring the big data firm Greenplum, EMC decided that the availability of data scientists would be a gating factor in its ownand customers'exploitation of big data. So its Education Services division launched a data science and big data analytics training and certification program. EMC makes the program available to both employees and customers, and some of its graduates are already working on internal big data initiatives.

Data scientists want to build things, not just give advice. One describes being a consultant as "the dead zone."

As educational offerings proliferate, the pipeline of talent should expand. Vendors of big data technologies are also working to make them easier to use. In the meantime one data scientist has come up with a creative approach to closing the gap. The Insight Data Science Fellows Program, a postdoctoral fellowship designed by Jake Klamka (a high-energy physicist by training), takes scientists from academia and in six weeks prepares them to succeed as data scientists. The program combines mentoring by data experts from local companies (such as Facebook, Twitter, Google, and LinkedIn) with exposure to actual big data challenges. Originally aiming for 10 fellows, Klamka wound up accepting 30, from an applicant pool numbering more than 200. More organizations are now lining up to participate. "The demand from companies has been phenomenal," Klamka told us. "They just can't get this kind of high-quality talent."

Even as the ranks of data scientists swell, competition for top talent will remain fierce. Expect candidates to size up employment opportunities on the basis of how interesting the big data challenges are. As one of them commented, "If we wanted to work with structured data, we'd be on Wall Street." Given that today's most qualified prospects come from nonbusiness backgrounds, hiring managers may need to figure out how to paint an exciting picture of the potential for breakthroughs that their problems offer.

Pay will of course be a factor. A good data scientist will have many doors open to him or her, and salaries will be bid upward. Several data scientists working at start-ups commented that they'd demanded and got large stock option packages. Even for someone accepting a position for other reasons, compensation signals a level of respect and the value the role is expected to add to the business. But our informal survey of the priorities of data scientists revealed something more fundamentally important. They want to be "on the bridge." The reference is to the 1960s television show Star Trek, in which the starship captain James Kirk relies heavily on data supplied by Mr. Spock. Data scientists want to be in the thick of a developing situation, with real-time awareness of the evolving set of choices it presents.

Considering the difficulty of finding and keeping data scientists, one would think that a good strategy would involve hiring them as consultants. Most consulting firms have yet to assemble many of them. Even the largest firms, such as Accenture, Deloitte, and IBM Global Services, are

in the early stages of leading big data projects for their clients. The skills of the data scientists they do have on staff are mainly being applied to more-conventional quantitative analysis problems. Offshore analytics services firms, such as Mu Sigma, might be the ones to make the first major inroads with data scientists.

But the data scientists we've spoken with say they want to build things, not just give advice to a decision maker. One described being a consultant as "the dead zoneall you get to do is tell someone else what the analyses say they should do." By creating solutions that work, they can have more impact and leave their marks as pioneers of their profession.

Data scientists don't do well on a short leash. They should have the freedom to experiment and explore possibilities. That said, they need close relationships with the rest of the business. The most important ties for them to forge are with executives in charge of products and services rather than with people overseeing business functions. As the story of Jonathan Goldman illustrates, their greatest opportunity to add value is not in creating reports or presentations for senior executives but in innovating with customer-facing products and processes.

LinkedIn isnt' the only company to use data scientists to generate ideas for products, features, and value-adding services. At Intuit data scientists are asked to develop insights for small-business customers and consumers and report to a new senior vice president of big data, social design, and marketing. GE is already using data science to optimize the service contracts and maintenance intervals for industrial products. Google, of course, uses data scientists to refine its core search and ad-serving algorithms. Zynga uses data scientists to optimize the game experience for both long-term engagement and revenue. Netflix created the well-known Netflix Prize, given to the data science team that developed the best way to improve the company's movie recommendation system. The test-preparation firm Kaplan uses its data scientists to uncover effective learning strategies.

Data scientists today are akin to the Wall Street "quants" of the 1980s and 1990s.

There is, however, a potential downside to having people with sophisticated skills in a fast-evolving field spend their time among general management colleagues. They'll have less interaction with similar specialists, which they need to keep their skills sharp and their tool kit state-of-the-art. Data scientists have to connect with communities of practice, either within large firms or externally. New conferences and informal associations are springing up to support collaboration and technology sharing, and companies should encourage scientists to become involved in them with the understanding that "more water in the harbor floats all boats."

Data scientists tend to be more motivated, too, when more is expected of them. The challenges of accessing and structuring big data sometimes leave little time or energy for sophisticated analytics involving prediction or optimization. Yet if executives make it clear that simple reports are not enough, data scientists will devote more effort to advanced analytics. Big data shouldn't equal "small math."

Hal Varian, the chief economist at Google, is known to have said, "The sexy job in the next 10 years will be statisticians. People think I'm joking, but who would've guessed that computer engineers would've been the sexy job of the 1990s?"

If "sexy" means having rare qualities that are much in demand, data scientists are already there. They are difficult and expensive to hire and, given the very competitive market for their services, difficult to retain. There simply aren't a lot of people with their combination of scientific

background and computational and analytical skills.

Data scientists today are akin to Wall Street "quants" of the 1980s and 1990s. In those days people with backgrounds in physics and math streamed to investment banks and hedge funds, where they could devise entirely new algorithms and data strategies. Then a variety of universities developed master's programs in financial engineering, which churned out a second generation of talent that was more accessible to mainstream firms. The pattern was repeated later in the 1990s with search engineers, whose rarefied skills soon came to be taught in computer science programs.

One question raised by this is whether some firms would be wise to wait until that second generation of data scientists emerges, and the candidates are more numerous, less expensive, and easier to vet and assimilate in a business setting. Why not leave the trouble of hunting down and domesticating exotic talent to the big data start-ups and to firms like GE and Walmart, whose aggressive strategies require them to be at the forefront?

The problem with that reasoning is that the advance of big data shows no signs of slowing. If companies sit out this trend's early days for lack of talent, they risk falling behind as competitors and channel partners gain nearly unassailable advantages. Think of big data as an epic wave gathering now, starting to crest. If you want to catch it, you need people who can surf.

T. H. Davenport is the President's Distinguished Professor in Management and Information Technology at Babson College, a research fellow at the MIT Initiative on the Digital Economy, and a senior adviser at Deloitte Analytics. He is the author of over a dozen management books, most recently Only Humans Need Apply: Winners and Losers in the Age of Smart Machines and The AI Advantage.

D.J. Patil is the data scientist in residence at Greylock Partners, was formerly the head of data products at LinkedIn, and is the author of Data Jujitsu: The Art of Turning Data into Product (O'Reilly Media, 2012).