Modèles et algorithmes des réseaux
DM2

à rendre le 19 janvier 2016

Exercise 1 Singular values and matrix completion (*10 points*)

We recall the matrix norms $\|A\|_2 = \sup_{x\neq 0}\|Ax\|_2/\|x\|_2$ and $\|A\|_F = \left(\sum_{i,j}|A_{i,j}|^2\right)^{1/2}$.

1. Prove that $\|A\|_2 \leq \|A\|_F \leq \sqrt{r}\|A\|_2$ where r is the rank of A.

We denote the SVD of A by $A = \sum_{i=1}^{\min(n,m)} \sigma_i(A)\mathbf{u}_i\mathbf{v}_i^T$. For a matrix A, we define A_k a *rang k approximation* of A by $A_k = \sum_{i=1}^{k}\sigma_i(A)\mathbf{u}_i\mathbf{v}_i^T$.

2. For any $i \leq \min(n,m)$ and vectors $\mathbf{w}_1,\ldots,\mathbf{w_{i-1}}$, show that
$$\sigma_i(A) \leq \max_{\mathbf{v}\perp\mathbf{w}_1,\ldots,\mathbf{w_{i-1}}} \frac{\|Av\|_2}{\|v\|_2}.$$
3. Let A,B be real matrices of dimension $m\times n$. Show that if $i+j \leq 1+\min\{m,n\}$, then
$$\sigma_{i+j-1}(A) \leq \sigma_i(B) + \sigma_j(A-B).$$
Deduce that
$$\max_{1\leq k\leq m\wedge n} |\sigma_k(A)-\sigma_k(B)| \leq \|A-B\|_2.$$
4. Show that
$$\|B-B_k\|_2 \leq \|A-A_k\|_2 + \|A-B\|_2.$$
Deduce that $\|A-B_k\|_2 \leq \|A-A_k\|_2 + 2\|A-B\|_2$.

We consider the following scenario : Let M be a $m\times n$ matrix with $m\leq n$ of rank r (modeling user rankings). Let $p\in[0,1]$. We assume that each entry of M is observed with probability p and not observed with probability $1-p$, independently of the other entries. For simplicity, we assume $M_{ij}\in[0,1]$ and that p and r are known. We construct an estimate $\hat{M}$ of M starting from the observed entries as follows :

a- Let X be the matrix with $x_{ij} = m_{ij}$ if the entry is observed and $x_{ij}=0$ else. Let $X = \sum_{i=1}^m s_i\mathbf{u}_i{\mathbf{v}_i}^T$ be its singular value decomposition.

b- We define
$$W = \frac{1}{p}\sum_{i\leq r} s_i\mathbf{u}_i{\mathbf{v}_i}^T.$$
and matrix $\hat{M}$ by $\hat{m}_{ij} = m_{ij}$ if the entry is observed and else by
$$\hat{m}_{ij} = \begin{cases} w_{ij} & \text{if } 0\leq w_{ij}\leq 1,\\ 1 & \text{if } w_{ij}>1,\\ 0 & \text{if } w_{ij}<0.\end{cases}$$

We define the mean squared error

$$\mathrm{MSE}(\hat{M}) = \mathbb{E}\left[\frac{1}{mn}\|M-\hat{M}\|_F^2\right].$$

5. Show that $\|M-W\|_F^2 \le 8r\|M-\frac{1}{p}X\|_2^2$. Deduce that

$$\mathrm{MSE}(\hat{M}) \le \frac{8r}{mnp^2}\mathbb{E}\left[\|pM-X\|_2^2\right]$$

To bound the last term, we use the matrix version of the Bernstein inequality : Let $Z_1,\dots,Z_k \in \mathbb{R}^{m\times n}$ be random independent matrices such that $\mathbb{E}[Z_t]=0$, $\|Z_t\|_2 \le 1$ and $\max\left\{\|\sum_{t=1}^k \mathbb{E}[Z_t Z_t^T]\|_2 ; \|\sum_{t=1}^k \mathbb{E}[Z_t^T Z_t]\|_2\right\} \le \sigma^2$. Then

$$\mathbb{P}\left(\|\sum_{t=1}^k Z_t\|_2 \ge s\right) \le (m+n)\exp\left(-\frac{s^2}{2(\sigma^2+s/3)}\right).$$

6. We define $Y = pM - X$. Find $\mathbb{E}[Y]$. Show that $\|Y\|_2^2 \le nm$.
7. Show that

$$\mathbb{P}(\|Y\| \ge s) \le (m+n)\exp\left(-\frac{s^2}{2(np+s/3)}\right)$$

8. Show that $\mathbb{E}\left[\|Y\|_2^2\right] \le nm\mathbb{P}(\|Y\|_2 \ge s) + s^2$. Deduce that for $np > \ln n$, there is a constant C such that :

$$\mathrm{MSE}(\hat{M}) \le \frac{Cr\log n}{mp}.$$

9. For $m = 100$, $n = 1000$ and $r = 10$, generate a matrix M (you can instead use the one available on my website `http://www.di.ens.fr/~lelarge/ani/matrix.txt`). For various values of $p \in [0,1]$, compute an empiric esitmate of $MSE(\hat{M})$ and plot the corresponding picture.

Exercise 2 Calcul distribué du nombre de nœuds dans un réseau (*5 points*)

On considère un réseau fortement connecté avec N nœuds et M liens bidirectionnels. Chaque nœud connait son identité et l'ensemble de ses voisins mais pas la topologie du réseau. Le nœud 1 veut déterminer le nombre total de nœuds dans le réseau. Dans une première étape, il initialise un algorithme pour trouver un arbre couvrant orienté, enraciné, ayant le nœud 1 pour racine.

1. Proposer un algorithme distribué utilisant l'échange des messages entre les nœuds voisins pour construire un tel arbre. L'algorithme est initié par le nœud 1 et doit utiliser au plus $O(M)$ messages. A la fin de l'algorithme, les nœuds qui sont des extrémités de chaque lien doivent connaitre si le lien fait partie de l'arbre couvrant, et si c'est le cas, quelle est la direction du lien.
2. Compléter le premier algorithme avec une deuxième étape qui utilise au plus $O(N)$ messages, et qui permet au nœud 1 de connaitre la valeur de N.
3. En supposant que chaque transmission du message prend le même temps T, donner une borne supérieure pour le temps nécessaire pour les deux étapes.

Exercise 3 Gossip à mémoire limitée (*5 points*)

Soit $G = (V,E)$ un graphe non-orienté. Nous allons considérer les algorithmes du type gossip pair-à-pair, mais avec une contrainte supplémentaire : les nœuds ont une mémoire qui consiste de 2 bits uniquement. Supposons que la valeur initiale des nœuds soit 0 ou 1. On cherche à calculer la valeur majoritaire.

Au temps t, les valeurs de nœuds $x_k(t), k \in V$, appartiennent à l'ensemble $\{0, 0.5-, 0.5+, 1\}$ (ces quatre valeurs différentes peuvent être codées sur 2 bits). Nous allons supposer un ordre total suivant : $0 \le 0.5- \le 0.5+ \le$

1. Un nœud est choisi de manière uniforme parmi tous les nœuds. Ce nœud choisit de manière uniforme un de ces voisins. Notons par i le nœud avec l'état le plus petit parmi ces deux nœuds et l'autre par j. Les nœuds i et j mettent à jour leurs valeurs selon la règle suivante (pour l'addition, on ignore les $-$ et les $+$ pour les valeurs $0.5-$ et $0.5+$) :

$$\begin{aligned} x_i(t+1) &= \left\lceil \frac{x_i(t)+x_j(t)}{2} \right\rceil \\ x_j(t+1) &= \left\lfloor \frac{x_i(t)+x_j(t)}{2} \right\rfloor \end{aligned}$$

Quand $x_i(t+1)$ ou $x_j(t+1)$ est égal à 0.5, alors on doit spécifier s'il s'agit de $0.5-$ ou $0.5+$. Nous allons considérer quatre cas différents :

— Si $x_i(t) = x_j(t)$, alors $x_i(t+1) = x_j(t+1) = x_i(t)$,

— Si $x_i(t) \neq x_j(t)$ et $x_i(t+1) = x_j(t+1) = 0.5$, alors $x_i(t+1) = 0.5+$ et $x_j(t+1) = 0.5-$.

— Si uniquement $x_i(t+1) = 0.5$, alors $x_i(t+1) = 0.5-$.

— Si uniquement $x_j(t+1) = 0.5$, alors $x_j(t+1) = 0.5+$.

Nous allons montrer quelques propriétés de cet algorithme.

1. Montrer les propriétés suivantes :

   (a) Préservation de la moyenne :

   $$x_i(t+1) + x_j(t+1) = x_i(t) + x_j(t).$$

   (b) Contraction : les valeurs de $x_i(t+1)$ et $x_j(t+1)$ sont soit égales soit consécutives. De plus, si $x_i(t) = x_j(t)$, alors $x_i(t+1) = x_j(t+1) = x_i(t) = x_j(t)$.

   (c) Si $x_i(t) \leq x_j(t)$, alors $x_i(t+1) \geq x_j(t+1)$. En particulier, si les valeurs de $x_i(t)$ et $x_j(t)$ sont consécutives, alors elles sont échangées : $x_i(t+1) = x_j(t)$ et $x_j(t+1) = x_i(t)$.

   (d) $\min(x_i(t), x_j(t)) \leq x_k(t+1) \leq \max(x_i(t), x_j(t))$, $k \in \{i, j\}$.

2. En utilisant les propriétés précédentes, montrer que cet algorithme converge au sens suivant : après un temps presque sûrement fini, les valeurs des nœuds sont soit toutes égales, soit il y a uniquement deux valeurs différentes et ces deux valeurs sont consécutives (au sens de l'ordre total $0 \leq 0.5- \leq 0.5+ \leq 1$). De plus, si au temps $t = 0$ la valeur majoritaire est 1, alors ces deux valeurs consécutives sont $0.5+$ et 1. Par symétrie, si la valeur majoritaire est 0, les seules valeurs seront 0 et $0.5-$. Si au temps $t = 0$ il y a autant de 0 que de 1, alors après un temps presque sûrement fini, on aura que des $0.5-$ et des $0.5+$.

3. Proposer un algorithme de type gossip à mémoire finie qui répond au problème suivant. Supposons que la valeur initiale des nœuds soit 0 ou 1. On cherche à vérifier si au moins 2/3 des valeurs initiales sont égales à 1. Quelle est la taille de mémoire nécessaire pour votre algorithme ? .