Rapport TP Algèbre linéaire pour le data mining

HUC-LHUILLERY Alexia

1 - Traitement d'ambiguïtées entières

2.2 - Si $\hat{I}$ est connu, le problème devient alors un problème aux moindres carrés linéaires en x, qui s'écrit $min_{x\in R^p}\|(b - G\hat{I}) - Ax\|$. La solution vérifie donc l'équation $A^TAx = A^T(b - G\hat{I})$ par résolution en dérivant l'expression, et on sait que A^TA est inversible donc la solution $\hat{x}$ existe et est unique.

Pour résoudre QNE dans le cas où Q est diagonale, on peut remarquer que QNE s'écrit $min_{I\in Z^n}\sum_{k=1}^{n}(I_k - \bar{I}_k)^2q_k$, ce qui revient à minimiser la distance entre I et $\bar{I}$ car pour tout k, $q_k > 0$. On choisit donc pour tout k l'arrondi au plus proche de $\bar{I}_k$ comme valeur de I_k pour minimiser la distance.

3.1 - On cherche à minimiser la fonction ϕ sur Z^n, χ est calculé d'après une solution possible $round(\bar{I})$ de ce problème, donc $\phi(\bar{I})$ est inférieur ou égal à χ, la solution réelle se trouve alors dans l'espace définit par $C(\chi)$. Dans le problème QNE(χ) l'espace de recherche a été réduit à $C(\chi)$, donc $\hat{I}$ est bien solution du problème QNE(χ).

On a $\chi_1 \leq \chi$ et la fonction ϕ est définie par une quadratique donc elle est strictement convexe. La solution réelle vérifie alors $\phi(\hat{I}) \leq \chi_1$ et l'espace $C(\chi_1)$ est plus petit que l'espace $C(\chi)$, il est donc plus intéressant de chercher la solution sur ce nouvel espace.

3.1.2 - On trouve comme borne pour i_1 :

$$g_1(i_3, i_2) = \lceil -\frac{\sqrt{\chi - r_{33}(i_3 - \bar{i}_3)^2 - (r_{22}(i_2 - \bar{i}_2) + r_{23}(i_3 - \bar{i}_3))^2}}{r_{11}} - \frac{r_{12}(i_2 - \bar{i}_2) + r_{13}(i_3 - \bar{i}_3)}{r_{11}} + \bar{i}_1 \rceil$$

$$d_1(i_3,i_2) = \lfloor \frac{\sqrt{\chi - r_{33}(i_3-\bar{i}_3)^2 - (r_{22}(i_2-\bar{i}_2)+r_{23}(i_3-\bar{i}_3))^2}}{r_{11}} - \frac{r_{12}(i_2-\bar{i}_2)+r_{13}(i_3-\bar{i}_3)}{r_{11}} + \bar{i}_1 \rfloor$$

La solution du problème QNE n'est en général pas unique, comme on peut le voir dans l'exemple du sujet, les I donnés sont deux solutions différentes du problème. Ces solutions ne sont pas des minimums de la fonction ce qui enlève l'unicité de la solution.

La solution généralisée qui est implantée ne fonctionne pas car elle ne converge pas assez rapidement. C'est une fonction récursive descendante, mais les bornes trouvées sont trop larges et les boucles prennent donc du temps à être calculées. Mais cela fonctionne sur l'exemple d'avant avec les Q et $\bar{I}$ donnés.

2 - Classification spectrale

En faisant varier σ on observe que les figures sont similaires pour des valeurs élevées supérieures à 10, avec des rayures qui correspondent environ aux images d'origine, entre 1 et 10 les images obtenues ont peu de détails, et pour des valeurs inférieures à 1, les images ne ressemblent plus à celles d'origine.

On observe comme résultats avec les profils temporels simulés une ressemblance avec l'image d'origine, les taches principales se trouvent au mêmes endroits bien que les formes ne soient pas exactement les mêmes.

3 - Algorithme de page ranking

2.2 - Pour vérifier que $e^T P = e^T$, on peut vérifier que pour tout j on a $\sum_{i=1}^n P_{ij} = 1$. Si la colonne j de Q est vide, on a alors pour tout i $P_{ij} = \frac{1}{n}$ par construction de d d'où le résultat. Sinon, $\sum_{i=1}^n P_{ij} = \sum_{i=1}^n Q_{ij}$ par construction de d aussi, donc

$$\sum_{i=1}^n P_{ij} = \sum_{i=1}^n \frac{1}{N_j} |O_j| = 1$$

par définition de Q car $|O_j|$ est le nombre de pages Web vers lesquelles j a un lien. On a donc bien le résultat.

Avec un graphe dont une page au moins n'en recommande aucune autre mais qui est recommandée, on obtient que Q n'a pas 1 pour valeur propre et P l'a.

2.3 - Après les calculs, on obtient que Q est une matrice creuse, P est une matrice creuse contenant les coefficients de Q avec en plus des coefficients sur les colonnes vides de Q, et $A_{\alpha,v}$ est une matrice pleine. Le vecteur propre que l'on cherche est un vecteur propre de $A_{\alpha,v}$, $A_{\alpha,v}$ étant pleine le calcul direct sera d'autant plus couteux que la taille de la matrice est importante.

3 - On a :

$$A_{\alpha,v}r = \alpha Pr + (1-\alpha)ve^T r = \alpha Qr + \frac{\alpha}{n}ed^T r + (1-\alpha)v$$

On peut donc calculer r en utilisant la matrice creuse Q, ainsi que d qui est un vecteur valant 1 aux indices des colonnes nulles de Q et le vecteur v. On a donc juste besoin de Q et de v pour pouvoir obtenir r par la méthode de la puissance itérée en remplaçant le calcul de $A_{\alpha,v}r$ par celui-ci. J'ai voulu utiliser dans l'implantation les indices des éléments non nuls de Q pour le calcul de Qr, mais je n'obtient pas le même résultat qu'avec le calcul direct, et je n'ai pas trouvé d'où venait l'erreur.