**Partie II – Complétion de la matrix bruitée**

**5 sous-base**

|  |  |  |
| --- | --- | --- |
| **ml-100k** | **lambda** | **RMSE** |
| **Base test 1** | **30** | **0.974113** |
| **Base test 2** | **25** | **0.9570765** |
| **Base test 3** | **25** | **0.95019** |
| **Base test 4** | **25** | **0.9526237** |
| **Base test 5** |  |  |

**1/Introduction du problème**

Ayant pour l’objectif de prédire les notes inconnues, on considère l’ensemble de notes une matrice incomplète et on a envie de la compléter. Dans notre contexte, il faut combler les données manquantes dans la base d’apprentissage M. représente l’ensemble de l’information disponible dans cette base. On définit l’opérateur tel que :

Pour ce faire, il faut retenir au maximum les corrélations entre différentes lignes et colonnes mais retirer les bruits aléatoires. Sachant les corrélations sont autant plus important que le rang de la matrice est petit, on cherche une approximation X proche de la base originale **M** mais avec le rang le plus faible. En plus, on ne peut pas prédire les notes concernées s’il nous manque l’information de toute ligne ou colonne. L’opérateur nous permet de se focaliser sur l’information existante.

**2/Analyses mathématiques**

Il s’agit de résoudre le problème d’optimisation sous contrainte, ce qui s’exprime par la formule suivante :

|  |  |
| --- | --- |
| Minimise |  |
| S.C. |  |

Le fait que la fonction rang(X) n’est pas forcément convexe causera le problème NP-hard et ainsi on introduit une méthode alternative convexe selon (…)

|  |  |
| --- | --- |
| Minimise |  |
| S.C. |  |

Similaire au fait que le problème de minimisation est la meilleur alternative convexe du problème de minimisation , la norme nucléaire nous permet d’approcher le problème de minimisation de rang. Pour simplifier la résolution numérique, on le reformule comme ci-dessous :

**3/Méthode numérique**

On choisit d’utiliser l’algorithme du « gradient descente », qui est destiné à minimiser la fonction différentiable.

|  |
| --- |
| **Algorithme du gradient** — On se donne un point initial aléatoirement et un seuil de tolérance . L'algorithme du gradient définit une suite d'itérés , jusqu'à ce qu'un test d'arrêt soit satisfait. Il passe de  à  par les étapes suivantes.   1. Simulation : calcul de   d’où   1. Test d’arrêt : si 2. Calcul du pas (n est l’indice de l’itéré). 3. Nouvel itéré : |