



Mots clés : Réduction de dimension, codage parcimonieux, apprentissage d'un dictionnaire, compressive sensing...

Dans le procédé du compressive Sensing, les méthodes de reconstruction permettent de retrouver le signal à partir de sa forme mesurée car l'information portée par le signal se retrouve intégralement dans la forme compressée (mesurée).

Nous nous intéresserons dans ce travail à la validation des méthodes de codage parcimonieux et la possibilité d'exploitation du dictionnaire appris pour reconstruire un signal dont des données sont manquantes en utilisant la méthode (de codage parcimonieux) validée.

Après une phase initiale d'apprentissage de dictionnaire, le travail s'organisera autour de trois phases:

1. La première phase de ce travail consistera à développer des méthodes de codage parcimonieux.
2. Générer des matrices de mesures et valider vos choix de nombre de mesure.
3. Appliquer le procédé du compressive sensing afin de compléter des données manquantes.

Rendu

Une archive qui contient toutes les sources de votre code augmenté de commentaires. Un rapport reprenant le travail d'analyse initial augmenté des aspects de tests sera fourni (**Les questions ont pour objectif d'orienter votre travail, vous ne devez en aucun cas fournir une copie de réponses aux questions**). Des graphiques et des tableaux comparatifs permettront de mettre en évidence les résultats des simulations pour la reconstruction parcimonieuse et le procédé du compressive sensing et avoir un avis critique sur ces résultats.

Ce micro projet doit être réalisé par groupes de 3 élèves. La qualité du compte rendu, du code et le bon fonctionnement de celui-ci contribueront à la note finale de votre micro projet.

Modalités

Ce mini projet doit être réalisé par groupes de 3 élèves. La qualité du rapport, du code et le bon fonctionnement de celui-ci contribueront à la note finale de votre projet.

1 Contexte

Les données qu'on va utiliser seront les mêmes représentées dans le fichier "data" et qui correspondent à la production des puits en gaz, la production d'huile et la production en eau. Ces données seront considérées comme une matrice. Cette matrice est de taille $\mathbf{N} \times \ell$, $\mathbf{N} = 98$ et $\ell = 108$.

Les colonnes \mathbf{X}_i , $i = 1, \dots, \ell$ représentent des vecteurs d'apprentissage et nous permettrons d'apprendre un dictionnaire $\mathbf{D} \in \mathcal{M}_{\mathbf{N} \times \mathbf{K}}(\mathbb{R})$ dans lequel les signaux \mathbf{X}_i sont parcimonieux.

Le dictionnaire appris par la méthode K-SVD utilise la méthode de l'OMP est une matrice $\mathbf{D} \in \mathcal{M}_{\mathbf{N} \times \mathbf{K}}(\mathbb{R})$ où $\mathbf{N} = 98$, $\mathbf{K} = 100$, $\mathbf{L} = 10$ et $\epsilon = 10^{-6}$.

2 Apprentissage d'un dictionnaire

**Travail à faire**

En utilisant la méthode d'apprentissage du KSVD, apprendre un dictionnaire \mathbf{D} sur les vecteurs d'apprentissage. Expliquer l'intérêt d'un tel apprentissage par rapport aux autres dictionnaires usuels.

3 Codage parcimonieux

La résolution du problème de minimisation (ℓ_0)

$$\min_{\tilde{\alpha}} \|\alpha\|_0 \text{ sous la contrainte } \mathbf{D}\tilde{\alpha} = x \quad (1)$$

nécessite une recherche exhaustive de la solution la plus parcimonieuse $\tilde{\alpha}$ et est très complexe à mettre en œuvre. Plusieurs méthodes ont été proposées dans la littérature pour contourner ce problème. La famille des algorithmes gloutons est large et elle repose sur des sélections d'atomes qui contribuent le plus au résiduel pris lorsqu'on note du signal original la contribution des atomes sélectionnés. Il existe des algorithmes permettant

- ♣ la sélection d'un seul atome par itération (Matching pursuit et Orthogonal Matching Pursuit),

- ♣ la sélection de plusieurs atomes par itération (Stagwise Orthogonal Matching Pursuit),
- ♣ la sélection et le rejet d'atomes par itération (Compressive Sampling du Matching Pursuit).

Description de la méthode StOMP

L'algorithme de **Stagwise Orthogonal Matching Pursuit** (StOMP) consiste à la **sélection** de plusieurs atomes (au lieu d'un seul atome pour l'OMP) dépassant un seuil calculé à chaque itération. Le StOMP s'initialise comme suit:

$$\alpha = 0_{K,1}, \quad \mathbf{R} = \mathbf{x}; \quad \mathbf{R}: \text{ le résiduel}$$

À l'itération $k \geq 1$, l'algorithme StOMP procède selon les étapes suivantes:

1. Calcul du vecteur des contributions de tous les atomes \mathbf{d}_j , $j = 1, \dots, K$ au résiduel courant,

$$\mathbf{C}_j = \frac{|\langle \mathbf{d}_j, \mathbf{R}^{(k-1)} \rangle|}{\|\mathbf{d}_j\|_2}.$$

2. **La sélection:**

(a) On calcule le seuillage $\mathbf{S}^{(k)}$: $\mathbf{S}^{(k)} = \mathbf{t} \frac{\|\mathbf{R}^{(k-1)}\|_2}{\sqrt{K}}$, $2 \leq \mathbf{t} \leq 3$.

(b) On sélectionne l'ensemble Λ_k des indices des atomes dont la contribution est supérieure au seuillage $\mathbf{S}^{(k)}$

$$\Lambda_k = \left\{ j \in \{1, \dots, K\}, \quad \mathbf{C}_j > \mathbf{S}^{(k)} \right\}.$$

3. **La mise à jour:** On met à jour l'ensemble des indices $\mathbf{P}_k = \mathbf{P}_{k-1} \cup \Lambda_k$.

4. On construit la matrice Φ_k en considérant les colonnes \mathbf{d}_p , $p \in \mathbf{P}_k$.

$$\Phi_k = [\mathbf{d}_p, p \in \mathbf{P}_k].$$

5. On résout le problème d'optimisation $\alpha_{\mathbf{P}_k}^{(k)} = \arg \min \|x - \Phi_k \alpha_{\mathbf{P}_k}\|_2$ par la méthode des moindres carrées.

6. On met à jour le résiduel, $\mathbf{R}^{(k)} = \mathbf{x} - \Phi_k \alpha_{\mathbf{P}_k}^{(k)}$.

**Travail à faire**

Implémenter la méthode StOMP en expliquant son principe et la différence avec les méthodes MP et OMP.

Description de la méthode CoSaMP

L'algorithme du Compressive Sampling du Matching Pursuit (CoSaMP) consiste à la **sélection** des atomes et au **rejet** de quelques atomes déjà sélectionnés en une itération. Le CoSaMP s'initialise comme suit:

$\alpha = 0_{K,1}$, $\mathbf{R} = \mathbf{x}$; \mathbf{R} : le résiduel
 $\text{Supp} = \emptyset$ Supp : le support de la solution parcimonieuse

Le CoSaMP suit cinq étapes principales et repose sur la connaissance a priori de l'ordre s de parcimonie de la solution:

Tant que le critère d'arrêt n'est pas satisfait faire, $k \geq 1$,

1. La sélection: Déterminer les $2s$ atomes de plus grande contribution au résiduel et en déduire le vecteur des positions de ces atomes, on le notera le support de la sélection **supp1**.
2. Mise à jour du support: Fusionner le support de la sélection **supp1** avec le support de la solution de l'itération précédente **supp** :

$$\text{supp} = \text{supp} \cup \text{supp1}$$

On note $\mathbf{AS} = \mathbf{D}(:, \text{supp})$ la matrice des atomes actifs sélectionnés.

3. Estimation: Utiliser la méthode des moindres carrés pour estimer les coefficients \mathbf{z}_i de la solution parcimonieuse α correspondant au support mis à jour.
4. Rejet: Considérer les s plus grands coefficients de \mathbf{z} .
5. Mise à jour du résiduel: $\mathbf{R} = \mathbf{x} - \mathbf{D}\alpha$;

**Travail à faire**

1. En se basant sur vos résultats de l'apprentissage du dictionnaire par KSVD, déduire une valeur pour l'ordre de parcimonie des vecteurs d'apprentissage.
2. Implémenter la méthode CoSaMP.

3.1 Relaxation ℓ_p et optimisation convexe

La résolution du problème d'optimisation ℓ_0 au sens de la pseudo norme ℓ_0 est très difficile, pour palier ce problème, on peut relâcher cette contrainte en la remplaçant par une autre pseudo norme ℓ_p avec $0 < p < 1$.

Soit un vecteur $\alpha \in \mathbb{R}^K$ et $p > 0$, la pseudo norme ℓ_p se définit par

$$\ell_p(\alpha) = \|\alpha\|_p = \left(\sum_{i=1}^K |\alpha_i|^p \right)^{\frac{1}{p}}.$$

La pseudo norme ℓ_p avec $0 < p < 1$ est un bon indicateur de la parcimonie d'un vecteur. On s'intéresse donc à la possibilité de résoudre le problème (P_p) suivant :

$$(\mathbf{P}_p) : \min \|\alpha\|_p, \text{ tel que } \mathbf{x} = \mathbf{D}\alpha$$



Travail à faire

Montrer que le problème (\mathcal{P}_p) s'écrit sous forme d'un problème de moindres carrés pondérés :

$$(\mathbf{P}_2) : \min \|W\alpha\|_2, \text{ tel que } \mathbf{x} = \mathbf{D}\alpha$$

où W est une matrice qui dépend de α .

Il s'agit d'un problème non linéaire. On peut alors procéder de manière itérative, en fixant la valeur de W à partir de la solution estimée $\hat{\alpha}$ précédente, ce qui permet de calculer une nouvelle estimation de α et ainsi de suite.

L'algorithme IRLS, pour Iteratively Reweighted Least-Squares, cherche la solution de :

$$(\mathbf{P}_2) : \min \sum_{i=1}^K w_i^2 \alpha_i^2, \text{ tel que } \mathbf{x} = \mathbf{D}\alpha$$

où les poids sont tels que $w_i = |\alpha_i|^{\frac{p}{2}-1}$.



Travail à faire

Etablir les conditions d'existence de solutions du problème (\mathcal{P}_2) .

On cherche à obtenir la solution de (\mathcal{P}_2) par une méthode itérative. On note le poids à l'itération n , $w_i^{(n)} = |\alpha_i^{(n-1)}|^{\frac{p}{2}-1}$ où $\alpha_i^{(n-1)}$ est l'estimée du signal à l'itération précédente $n-1$.

**Travail à faire**

Supposons l'itéré $\alpha^{(k-1)}$ calculé, montrer que l'itéré courant peut s'obtenir selon la formule suivante

$$\alpha^{(k)} \leftarrow QD' (DD')^{-1} \mathbf{x}.$$

où Q est une matrice diagonale.

Comme on s'intéresse au cas où $0 < p < 1$, il faut prendre en considération le fait que les poids w_i ne sont plus définis si à un moment donné, $\exists \alpha_i = 0$. On propose alors une régularisation des poids, on ajoute un $\varepsilon > 0$ dans le poids :

$$w_i^{(n)} = \left(|\alpha_i^{(n-1)}|^2 + \varepsilon \right)^{\frac{p}{2}-1}.$$

En procédant ainsi, on peut garder les termes nuls de α dans la matrice Q (lorsque w_i est nul, il reste le terme en ε). Ce terme de régularisation est variable : en effet, au début des itérations, on le choisit relativement élevé pour que son influence soit minime. Ensuite, ce terme est diminué lorsque l'évolution entre deux itérations successives n'est plus suffisante, car on suppose que le signal devient parcimonieux et le recours à la régularisation devient nécessaire. On fixe un seuil minimal à ce paramètre afin de définir un critère d'arrêt pour l'algorithme. On remarque que ce seuil minimal conditionne en partie l'erreur de reconstruction.

Algorithme IRLS

1. On initialise le compteur d'itération $k = 0$, le nombre maximum d'itérations k_{\max} , le coefficient de régularisation à $\varepsilon = 0,1$, et la solution de départ:

$$\alpha^{(0)} = D' (DD')^{-1} \mathbf{x}.$$

2. On calcule les poids $w_i = \left(|\alpha_i^{(k-1)}|^2 + \varepsilon \right)^{\frac{p}{2}-1}$;

3. On calcule la prochaine itération $\alpha^{(k)} = QD' (DD')^{-1} \mathbf{x}$.

(a) Si $\left| \|\alpha^{(k)}\|_2 - \|\alpha^{(k-1)}\|_2 \right| > \frac{\sqrt{\varepsilon}}{100}$ et $k < k_{\max}$ on retourne à l'étape 2 après avoir incrémenté le compteur k .

(b) Si $\left| \|\alpha^{(k)}\|_2 - \|\alpha^{(k-1)}\|_2 \right| < \frac{\sqrt{\varepsilon}}{100}$ et $\varepsilon > 10^{-8}$ alors on modifie $\varepsilon = \frac{\varepsilon}{10}$:

i. Si $k < k_{\max}$, on incrémente k et on retourne à l'étape 2,

ii. sinon, on termine.

**Travail à faire**

I Implémenter l'algorithme IRLS.

4 Procédé du compressive sensing

On considère les matrices de mesures proposées dans le cours (chapitre 3): Φ_1, Φ_2, Φ_3 et Φ_4 . On rappelle que le nombre M de mesures considérées nous fournit un ratio de mesures

$$r = \frac{M}{N}.$$

On considère le pourcentage P de mesures souhaitées par rapport à la taille N du vecteur d'origine, $P = r \times 100$ et le nombre de mesures M comme un arrondi supérieur de $\frac{P \times N}{100}$.

**Travail à faire**

1. On considère le Pourcentage de mesures P égal à 15, 20, 25, 30, 50 et 75. Déterminer les vecteurs de mesures $y_i, i = 1, \dots, 4$ en utilisant les 5 matrices de mesure.
2. Calculer la cohérence mutuelle entre le dictionnaire D et les différentes matrices de mesure pour chacun des pourcentages en remplissant le tableau suivant:

| Cohérence mutuelle | $P = 15$ | $P = 20$ | $P = 25$ | $P = 30$ | $P = 50$ | $P = 75$ |
|--------------------|----------|----------|----------|----------|----------|----------|
| Φ_1 | | | | | | |
| Φ_2 | | | | | | |
| Φ_3 | | | | | | |
| Φ_4 | | | | | | |

**Travail à faire**

I Commenter vos résultats.

On considère 3 vecteurs pour validation des résultats (fichier : vecteurs pour valider).

**Travail à faire**

1. Utiliser le procédé du compressive pour reconstruire les vecteurs d'origine à l'aide des méthodes de de codage parcimonieux : MP, OMP, StOMP, CoSaMP et IRLS.
2. Calculer l'erreur relative de reconstruction en remplissant le tableau suivant pour chaque vecteur $X_j, j = 1, 2, 3$:

| Erreurs relatives | P = 15 | P = 20 | P = 25 | P = 30 | P = 50 | P = 75 |
|-------------------|---------------|---------------|---------------|---------------|---------------|---------------|
| MP | | | | | | |
| OMP | | | | | | |
| StOMP | | | | | | |
| CoSaMP | | | | | | |
| IRLS | | | | | | |

**Travail à faire**

Commenter vos résultats.