Transport optimal pour la complétion de données manquantes dans des séries temporelles

LÉO LAFFEACH, ENS Rennes
ROMAIN TAVENARD, LETG, Université Rennes 2

Dans ce papier, nous traitons du sujet des données manquantes et abordons des méthodes pour effectuer une complétion. Nous nous limitons au domaine des séries temporelles et utilisons deux jeux de données provenant de différents domaines et donc qui ont des différences temporelles. Nous présentons une méthode de complétion de données permettant, en plus d'une association des données, un alignement temporel entre les différents domaines. Cette méthode se nomme *Match-And-Deform* (MAD), et nous comparons cette méthode à une méthode utilisant la divergence de Sinkhorn.

1 INTRODUCTION

Collecter des données peut facilement s'avérer complexe et il en résulte un grand nombre de données manquantes dans les jeux de données. C'est un problème qui existe depuis que les scientifiques collectent des données, et c'est un problème de plus en plus récurrent connaissant la large quantité de données collectées.

Il est donc utile d'avoir des méthodes pour compléter ces valeurs manquantes, plusieurs approches peuvent être possibles comme compléter avec une valeur constante bien choisie, ou alors apprendre à imputer. On va s'inspirer des méthodes présenté dans *Missing Data Imputation using Optimal Transport* [1] qui utilise le transport optimal pour faire des correspondances entre les données d'une même distribution pour les compléter. Cependant, nous nous intéressons ici à la complétion de jeux de données de séries temporelles et utilisons plusieurs domaines pour effectuer la complétion. Prenons un exemple :

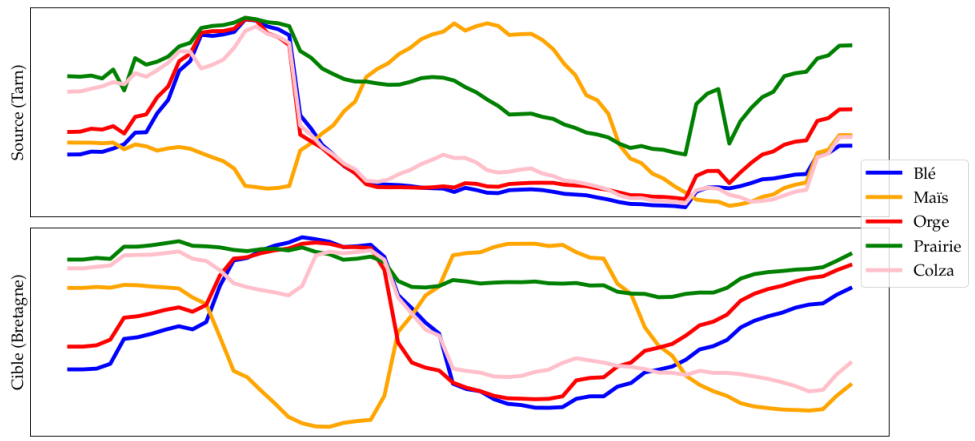


Fig. 1 – Moyennes par classe d'occupation du sol d'un indicateur de croissance de végétation pour deux zones géographiques différentes.

Sur la figure **??**, nous avons l'évolution des moyennes de croissance de végétation dans le Tarn et en Bretagne, on remarque que les moyennes semblent avoir une évolution similaire sur les deux domaines, mais ces variations apparaissent à des temps différents. Par exemple, le maïs semble

Authors' addresses: Léo LAFFEACH, ENS Rennes; Romain TAVENARD, LETG, Université Rennes 2.

pousser plus tôt dans le Tarn que en Bretagne alors que le blé et l'orge ont une deuxième croissance plus tôt en Bretagne que dans le Tarn. Ainsi, en cas de données manquantes, on pourrait vouloir utiliser ces données similaires pour effectuer une complétion. Mais il nous faut alors une méthode pour aligner d'un point de vue temporel les différents *dataset* et ensuite faire correspondre les données pour faire la complétion. On fait alors face à deux problèmes pour la complétion de données, l'alignement temporel et une assossiation entre les données des différents jeux. Une des pistes de résolution peut se trouver avec le transport optimal (*Optimal Transport* (OT)), qui permet d'avoir une distance géométriquement sensée pour comparer des distributions discrètes, et donc des données. L'autre peut voir une solution avec l'alignement temporel dynamique (*Dynamic Time Warping* (DTW)), permettant l'alignement de données temporelles.

Dans la suite du document nous allons d'abord présenter le transport optimal qui sert à la divergence de Sinkhorn, puis nous allons présenter un algorithme permettant la complétion de données en utilisant cette fonction de coût. Nous présenterons ensuite notre méthode *Match-And-Deform* (MAD), se basant sur le transport optimal mais aussi sur l'alignement temporel dynamique. Et enfin nous comparons les différentes méthodes sur quelques couples de domaines.

2 TRANSPORT OPTIMAL ET IMPUTATION DE DONNÉES MANQUANTES

Ici nous introduisons le transport optimal. Il définit une distance entre deux distributions. Pour coller aux conventions habituelles, on suppose que $\mathbf{X}$ est de dimension (n, d).

2.1 Transport Optimal

Le transport optimal est un problème d'optimisation qui permet de définir une distance entre deux mesures de probabilités. Soit $\mathbf{X}$ et $\mathbf{X}$', deux ensembles d'échantillons avec des poids, dans $\mathbb{R}^d$: $\{(x_i, w_i)\}_{i=1}^{n}$ avec $\sum_i w_i = 1$ et $\{(x'_j, w'_j)\}_{j=1}^{n'}$ avec $\sum_j w'_j = 1$. En l'absence d'information supplémentaire, les poids sont souvent fixés uniformément. $\mathbf{OT}$ définit une distance entre $\mathbf{X}$ et $\mathbf{X}$' en trouvant le plan de transport $\gamma \in \mathbf{\Gamma}(\mathbf{w}, \mathbf{w}')$ qui minimise le coût de transport :

$$\mathbf{OT}(\mathbf{X}, \mathbf{X}') = \underset{\gamma \in \Gamma(\mathbf{w}, \mathbf{w}')}{\arg\min} \langle \mathbf{C}(\mathbf{X}, \mathbf{X}'), \gamma \rangle$$

où $\Gamma(\mathbf{w}, \mathbf{w}')$ est l'ensemble de transports linéaires contraints, de telle sorte que toute la masse de $\mathbf{X}$ est transportée vers toute la masse de $\mathbf{X}$' :

$$\Gamma(\mathbf{w}, \mathbf{w}') = \{\gamma | \gamma \geq 0, \gamma \mathbb{1}_{n'} = \mathbf{w}, \gamma^\top \mathbb{1}_n = \mathbf{w}'\}$$

$\mathbf{C}(\mathbf{X}, \mathbf{X}') = \{d(\mathbf{X}_{ij}, \mathbf{X}'_{i'j'})\}$, où $d(\mathbf{X}_{ij}, \mathbf{X}'_{i'j'})$ est la distance entre deux éléments de $\mathbf{X}$ et $\mathbf{X}$'.

La solution à ce problème s'appelle le *plan de transport* dont l'élément γ_{ij} indique la quantité de masse transportée de x^i vers x'^j.

2.2 OT avec régularisation

Une limite du problème de transport optimal est qu'il n'est pas différentiable, on peut alors ajouter une régularisation pour résoudre ce problème. Le transport optimal régularisé compare $\mathbf{X}$ et $\mathbf{X}$' en considérant le transport de masse entre $\mathbf{w}$ et $\mathbf{w}$' le plus efficient, d'après un coût de départ entre les supports.

$$\mathbf{OT}_\epsilon(\mathbf{X}, \mathbf{X}') = \underset{\gamma \in \mathbf{\Gamma}(\mathbf{w}, \mathbf{w}')}{\arg\min} \langle \mathbf{C}(\mathbf{X}, \mathbf{X}'), \gamma \rangle + \epsilon h(\gamma)$$

où $\epsilon > 0$ et $h(\gamma) = \sum_{ij} \gamma_{ij} \log \gamma_{ij}$ est l'entropie négative. Ainsi, $\mathbf{OT}_\epsilon$ peut être résolue en utilisant les itérations de Sinkhorn [2] cela fournit une solution différentiable. Cependant, dû au terme de l'entropie, $\mathbf{OT}_\epsilon$ n'est plus forcément positif.

Ce qui peut se résoudre via un débiaisement, en soustrayant les termes d'auto-correction. Soit

$$S_\epsilon(\mathbf{X}, \mathbf{X}') = \mathbf{OT}_\epsilon(\mathbf{X}, \mathbf{X}') - \frac{1}{2}(\mathbf{OT}_\epsilon(\mathbf{X}, \mathbf{X}) + \mathbf{OT}_\epsilon(\mathbf{X}', \mathbf{X}'))$$

L'équation précédente définit la divergence de Sinkhorn [3], qui est positive, convexe, et qui peut être calculée avec un faible coût additionel comparé à $\mathbf{OT}_\epsilon$ [4].

2.3 complétion de données en utilisant le transport optimal

Maintenant plaçons nous dans le cadre où l'on a des données manquantes dans notre jeu de données. Si l'on a peu de données manquantes, on peut se servir des données encore présentes dans notre jeu de données pour faire l'imputation et compléter notre jeu de données. On cherche alors à effectuer une complétion en associant les données d'une même distribution entre elles. On peut alors avoir intuitivement l'idée d'utiliser le transport optimal. Dans *Missing Data Imputation using Optimal Transport* [1], Muzellec et al. présentent trois algorithmes permettant la complétion de données utilisant la divergence de Sinkhorn. Qui, selon eux, peut être une bonne méthode pour compléter un faible pourcentage de données manquantes en utilisant les données restantes et puisque les données sont issues de la même distributions, on peut s'attendre à des similarités et donc chercher à associer les données avec du transport pour retrouver les données manquantes. Le premier algorithme est une implémentation directe visant à compléter les valeurs manquantes en minimisant la distance du transport optimal entre des lots. D'abord les valeurs manquantes sont initialisées avec la moyenne des valeurs observées modifiées via un léger bruit. Puis, des lots sont échantillonés et la divergence de Sinkhorn entre les lots est minimisée par rapport aux valeurs imputées en utilisant une actualisation du gradient (ici utilisant RMSprop (Tieleman & Hinton, 2012)).

Soit $\Omega = (\omega_{ij})_{ij} \in \{0,1\}^{n \times d}$ un masque binaire qui indique si la donnée est observée ou non, ie : $\omega_{ij} = 1$ (resp. 0) si et seulement si l'entrée (i,j) est observée (resp. manquante). On observe alors la matrice de données incomplètes suivante :

$$\mathbf{X} = \mathbf{X}^{(obs)} \odot \Omega + \mathbf{NA} \odot (\mathbb{1} - \Omega)$$

où $\mathbf{X}^{(obs)} \in \mathbb{R}^{\mathbf{n} \times \mathbf{d}}$ contient les données observées, $\odot$ est le produit élément par élément et $\mathbb{1}$ est la matrice ne contenant que des uns. Connaissant $\mathbf{X}$, notre objectif est de construire $\hat{\mathbf{X}}$ une estimation complétant les données manquantes de $\mathbf{X}$, ce qui peut être écrit comme :

$$\hat{\mathbf{X}} = \mathbf{X}^{(obs)} \odot \Omega + \hat{\mathbf{X}}^{(imp)} \odot (\mathbb{1} - \Omega)$$

où $\hat{\mathbf{X}} \in \mathbb{R}^{n \times d}$ contient les valeurs imputées.

Algorithm 1: Imputation avec Sinkhorn par lots

Input: $\mathbf{X} \in (\mathbb{R} \cup \{NA\})^{n \times d}, \Omega \in \{0,1\}^{n \times d}, \alpha, \eta, \epsilon > 0, n \geq m > 0,$

Initialisation : pour j = 1, ..., d,

- pour i t.q. $\omega_{ij} = 0, \hat{x}_{ij} \leftarrow \bar{x}^{obs}_{:j} + \epsilon_{ij}$ *avec* $\epsilon \sim \mathcal{N}(0, \eta)$ *et* $\bar{x}^{obs}_{:j}$ correspondant à la moyenne des données observées dans la j-ème variable (données manquantes)
- pour i t.q. $\omega_{ij} = 1, \hat{x}_{ij} \leftarrow x_{ij}$ (entrées observés)

Pour iter = 1, 2, ..., $iter_{max}$ **faire**

Echantilloner deux ensembles K et L de m indices

$\mathcal{L}(\hat{\mathbf{X}}_K, \hat{\mathbf{X}}_L) \leftarrow S_\epsilon(\hat{\mathbf{X}}_K, \hat{\mathbf{X}}_L)$

$\hat{\mathbf{X}}^{(imp)}_{K \cup L} \leftarrow \hat{\mathbf{X}}^{(imp)}_{K \cup L} - \alpha \mathbf{RMSprop}(\nabla_{\hat{\mathbf{X}}^{(imp)}_{K \cup L}} \mathcal{L})$

Fin Pour

Output: $\hat{\mathbf{X}}$

3 MATCH-AND-DEFORM (MAD)

Maintenant on revient à notre hypothèse de départ, $\mathbf{X} \in \mathbb{R}^{n \times t \times d}$ les données, avec n le nombre de séries, t le nombre d'horodatage et d le nombre de *features* et $\Omega \in \{0,1\}^{n \times t \times d}$.

3.1 Alignement temporel dynamique

L'alignement temporel dynamique est un autre problème d'optimisation qui cherche à aligner des séries temporelles, x et $x' \in \mathbb{R}^{t \times d}$.

$$\mathbf{DTW}(x, x') = \underset{\pi \in \mathcal{A}(T,T')}{\arg\min} \langle \mathbf{C}(x, x'), \pi \rangle$$

où $\mathcal{A}(T, T')$ est l'ensemble des alignements admissibles entre x et x'. Un alignement admissible $\pi \in \mathcal{A}(T, T')$ est une matrice binaire telle que $\pi_{1,1} = \pi_{T,T'} = 1$, et pour chaque couple d'horodatage $(l; m)$ tel que $\pi_{l,m} = 1$, il y a soit $\pi_{l-1,m} = 1$ ou $\pi_{l,m-1} = 1$ ou $\pi_{l-1,m-1} = 1$. Les autres valeurs de π valent 0. Cette définition d'ensemble d'alignement admissible permet une calculabilité en temps quadratique en utilisant de la programmation dynamique [5]. Et $\mathbf{C}(x, x') = \{d(x_{jk}, x'_{j'k'})\}$, où $d(x_{jk}, x'_{j'k'})$ est la distance entre deux éléments de x et x'.

3.2 MAD

D'un côté, le DTW est limité à trouver un alignement temporel entre deux séries, sans considération pour l'alignement d'ensembles de séries. D'un autre côté, l'OT permet d'associer différents éléments dans l'ensemble, mais est incapable de gérer des séries temporelles qui ne partageraient pas un horizon temporel commun. Cependant notre problème de complétion de données dans des séries temporelles nous demande de résoudre à la fois un alignement temporel et une association entre des échantillons de provenance différente. Pour résoudre ces deux problèmes on introduit une nouvelle mesure, dénommée Match-And-Deform (MAD), qui optimise à la fois un DTW global et un plan de transport basé sur OT pour faire correspondre deux jeux de données de séries temporelles : $\mathbf{X}$ et $\mathbf{X}$'.

Définisons MAD comme le problème d'optimisation suivant :

$$\begin{aligned} MAD(\mathbf{X}, \mathbf{X}') &= \underset{\substack{\gamma \in \Gamma(w,w') \\ \pi \in \mathcal{A}(T,T')}}{\arg\min} \langle \mathbf{L(X, X')} \otimes \pi, \gamma \rangle \\ &= \underset{\substack{\gamma \in \Gamma(w,w') \\ \pi \in \mathcal{A}(T,T')}}{\arg\min} \sum_{i,j} \sum_{l,m} d(x^i_l, x'^j_m) \pi_{lm} \gamma_{ij} \end{aligned}$$

où $\otimes$ est la multiplication tenseur-matrice, γ est le plan de transport entre les échantillons et π est le chemin DTW global qui aligne les horodatages entre $\mathbf{X}$ et $\mathbf{X}$'. Ici $\mathbf{L(X, X')}$ est un tenseur en 4 dimensions dont les éléments sont $\boldsymbol{L}^{i,j}_{l,m} = d(x^i_l, x'^j_m)$, avec $d : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}^+$ une distance.

Ce qui signifie que si $MAD(\mathbf{X}, \mathbf{X}') = 0$ ça implique, qu'à un alignement temporel près, on peut trouver une association exacte entre les données des deux jeux de données.

3.3 MAD pour l'imputation de données dans des séries temporelles

On suppose que l'on a deux jeux de données temporels et on veut faire de l'imputation de données en utilisant l'information présente dans les deux, ce qui peut être le cas pour la télédétection où l'on peut avoir des images satelites de deux sites différents. Il peut donc y avoir une déformation temporelle entre les deux jeux de données. On souhaite évaluer l'efficacité de notre méthode MAD dans la complétion de donnés dans les séries temporelles en se comparant à la divergence de Sinkhorn utilisée dans l'algorithme 1. On compare la complétion des données cibles pour les différentes méthodes.

L'algorithme suivant reprend l'algorithme 1 et remplace Sinkhorn par MAD en guise de fonction de coût, et on tire un lot de données par jeu de données, contrairement à l'agorithme **??** qui tire deux lots de données dans le même jeu de données.

Algorithm 2: Imputation avec MAD par lots

Input: $\mathbf{X} \in (\mathbb{R} \cup \{NA\})^{n \times t \times d}, \Omega \in \{0,1\}^{n \times t \times d}, \mathbf{X'} \in (\mathbb{R} \cup \{NA\})^{n' \times t' \times d}, \mathbf{\Omega'} \in \{0,1\}^{n' \times t' \times d}, \alpha, \eta, \epsilon > 0, n \geq m > 0,$

Initialisation : for k = 1, ..., d,

- pour i pour j t.q. $\omega_{ijk} = 0, \hat{x}_{ijk} \leftarrow \bar{x}^{obs}_{::k} + \epsilon_{ijk}$ *avec* $\epsilon \sim \mathcal{N}(0, \eta)$ *et* $\bar{x}^{obs}_{::k}$ correspondant à la moyenne des données observées dans la k-ème variable (données manquantes)
- pour i pour j t.q. $\omega_{ijk} = 1, \hat{x}_{ijk} \leftarrow x_{ijk}$ (données observées)

faire la même chose pour la donnée cible.

Pour iter = 1, 2, ..., $iter_{max}$ **faire**

Echantilloner deux ensembles K et L de m indices

$\mathcal{L}(\hat{\mathbf{X}}_K, \hat{\mathbf{X}}'_L) \leftarrow \mathbf{MAD}(\hat{\mathbf{X}}_K, \hat{\mathbf{X}}'_L)$

$\hat{\mathbf{X}}_K^{(imp)} \leftarrow \hat{\mathbf{X}}_K^{(imp)} - \alpha \mathbf{RMSprop}(\nabla_{\hat{\mathbf{X}}_K^{(imp)}} \mathcal{L})$

$\hat{\mathbf{X}'}_L^{(imp)} \leftarrow \hat{\mathbf{X}'}_L^{(imp)} - \alpha \mathbf{RMSprop}(\nabla_{\hat{\mathbf{X}'}_L^{(imp)}} \mathcal{L})$

Fin Pour

Output: $\hat{\mathbf{X}}, \hat{\mathbf{X}}'$

Puisque l'algorithme de Muzellec ne prend pas en compte l'alignement temporel, on peut s'attendre à ce que l'on ait de meilleur résultat pour compléter les données.

Cependant on peut penser que parfois Sinkhorn peut apporter quelque chose, car il détermine un alignement au sein du même domaine. Donc les données sont plus homogènes et la complétion est plus facile s'il y a peu de données manquantes. Ainsi on peut s'en servir en définissant la loss comme la somme de Sinkhorn et de MAD.

Algorithm 3: Imputation avec MAD et Sinkhorn par lots

Input: $\mathbf{X} \in (\mathbb{R} \cup \{NA\})^{n \times t \times d}, \Omega \in \{0,1\}^{n \times t \times d}, \mathbf{X'} \in (\mathbb{R} \cup \{NA\})^{n' \times t' \times d}, \mathbf{\Omega'} \in \{0,1\}^{n' \times t' \times d}, \alpha, \eta, \epsilon > 0, n \geq m > 0,$

Initialisation : for k = 1, ..., d,

- pour i pour j t.q. $\omega_{ijk} = 0, \hat{x}_{ijk} \leftarrow \bar{x}^{obs}_{::k} + \epsilon_{ijk}$ *avec* $\epsilon \sim \mathcal{N}(0, \eta)$ *et* $\bar{x}^{obs}_{::k}$ correspondant à la moyenne des données observées dans la k-ème variable (données manquantes)
- pour i pour j t.q. $\omega_{ijk} = 1, \hat{x}_{ijk} \leftarrow x_{ijk}$ (données observées)

faire la même chose pour la donnée cible.

Pour iter = 1, 2, ..., $iter_{max}$ **faire**

Echantilloner quatre ensembles K, L, K' et L' de m indices

$\mathcal{L}(\hat{\mathbf{X}}_K, \hat{\mathbf{X}}'_L) \leftarrow \mathbf{MAD}(\hat{\mathbf{X}}_K, \hat{\mathbf{X}}'_L) + S_\epsilon(\hat{\mathbf{X}}'_{K'}, \hat{\mathbf{X}}'_{L'})$

$\hat{\mathbf{X}}_K^{(imp)} \leftarrow \hat{\mathbf{X}}_K^{(imp)} - \alpha \mathbf{RMSprop}(\nabla_{\hat{\mathbf{X}}_K^{(imp)}} \mathcal{L})$

$\hat{\mathbf{X}'}_{K' \cup L \cup L'}^{(imp)} \leftarrow \hat{\mathbf{X}'}_{K' \cup L \cup L'}^{(imp)} - \alpha \mathbf{RMSprop}(\nabla_{\hat{\mathbf{X}'}_{K' \cup L \cup L'}^{(imp)}} \mathcal{L})$

Fin Pour

Output: $\hat{\mathbf{X}}, \hat{\mathbf{X}}'$

Ici on a un abus de langage, la fonction Sinkhorn ne prenant en paramètre que des données en 2 dimensions, alors que là $\hat{\mathbf{X}}'$ est en 3 dimensions, on a donc eu a redimensionner les données pour considérer une série temporelle comme un vecteur de taille $t \times d$.

4 EXPÉRIENCE

Pour évaluer les performances de MAD sur la complétion de données, nous allons comparer les différentes méthodes sur des séries temporelles, en les adaptant au besoin comme expliqué auparavant, pour l'algorithme 1 entre autre. On définit RMSE comme mesure pour évaluer la complétion, qui est aussi utilisée par Muzellec [1].

$$\sqrt{\frac{1}{m_0} \sum_{(i,j)|\omega_{ij}=0} (x_{i,j}^{true} - \hat{x}_{i,j})^2}. \tag{RMSE}$$

4.1 Génération synthétique de données manquantes

Maintenant que l'on a des méthodes pour compléter nos données manquantes, il nous reste à définir comment choisir ces données manquantes. La méthode la plus simple serait de choisir des couples $(j,k) \in \{0,1,...,t\} \times \{0,1,...,d\}$ et de les considérer comme manquant pour tout $i \in \{0,1,...,n\}$, il s'agit de considérer les données comme **manquante au hasard**. Une autre méthode, plus proche de la réalité pour notre jeu de données par exemple, serait de choisir des horodatages et d'y supprimer les données pour tout $i \in \{0,1,...,n\}$, ce qui revient à considérer la présence d'un nuage ou de son ombre au niveau de la zone de capture satellite ce qui résulte en une observation pour laquelle toutes les *features* (bandes spectrales) sont manquantes. Il s'agit de données **manquantes avec un biais**.

4.2 Manquantes au hasard

Nous comparons ici les valeurs de RMSE pour le domaine cible, des différentes méthodes au bout de 1000 itérations des algorithmes **??** (Sinkhorn), **??** (MAD) et **??** (MAD + Sinkhorn). Donc par exemple, pour $DK1 \longrightarrow FR1$, il s'agit des résultats pour la complétion de FR1. On se compare aussi à la moyenne qui est la méthode d'initialisation des données manquantes. La plus petite valeur par ligne est mise en **gras** et la seconde meilleure valeur est soulignée.

Pourcentage de données manquantes	Problèmes	Moyenne	Sinkhorn	MAD	MAD + Sinkhorn
50%	DK1 → FR1	1.2469	1.2528	1.1015	**1.0843**
	DK1 → FR2	1.1294	1.1305	1.0366	**1.0345**
	DK1 → AT1	1.1920	1.1944	**1.0343**	1.0346
	FR1 → DK1	1.1862	1.1894	1.0537	**1.0501**
	FR1 → FR2	1.1294	1.1305	1.0540	**1.0525**
60%	DK1 → FR1	1.2575	1.2614	1.1128	**1.1061**
	DK1 → FR2	1.1134	1.1141	1.0407	**1.0390**
	DK1 → AT1	1.1699	1.1716	1.0321	**1.0286**
	FR1 → DK1	1.1772	1.1791	**1.0567**	1.0568
	FR1 → FR2	1.1134	1.1141	1.0486	**1.0480**
70%	DK1 → FR1	1.2577	1.2599	1.1036	**1.1004**
	DK1 → FR2	1.0620	1.0627	1.0109	**1.0055**
	DK1 → AT1	1.1894	1.1904	1.0645	**1.0624**
	FR1 → DK1	1.2021	1.2031	1.1075	**1.1038**
	FR1 → FR2	1.0620	1.0627	1.0303	**1.0299**
80%	DK1 → FR1	1.2614	1.2626	1.1256	**1.1233**
	DK1 → FR2	1.1067	1.1072	1.0305	**1.0298**
	DK1 → AT1	1.2003	1.2016	1.0820	**1.0806**
	FR1 → DK1	1.2267	1.2311	1.1165	**1.1151**
	FR1 → FR2	1.1067	1.1072	1.0748	**1.0746**
90%	DK1 → FR1	1.2635	1.2605	1.1405	**1.1389**
	DK1 → FR2	1.0932	1.0937	1.0441	**1.0436**
	DK1 → AT1	1.1977	1.1961	**1.0863**	1.0900
	FR1 → DK1	1.2266	1.2247	1.1252	**1.1240**
	FR1 → FR2	1.0932	1.0937	**1.0549**	1.0588

TABLE 1 – Valeurs de RMSE pour différents pourcentages de données manquantes à 1000 itérations

On remarque que dans la grande majorité MAD donne de meilleur résultat que Sinkhorn tout seul et qu'outre de rares occasions indépendamment du couple de données, la combinaison de MAD et de Sinkhorn performe mieux que MAD tout seul.

4.3 Manquantes avec un biais

De même qu'à la section précédente, nous comparons les différents résultats de la même façon.

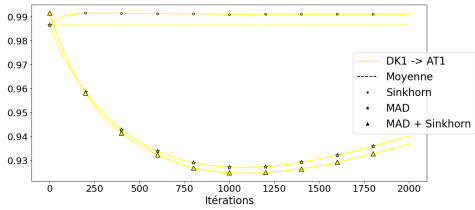
Pourcentage de données manquantes	Problèmes	Moyenne	Sinkhorn	MAD	MAD + Sinkhorn
50%	DK1 → FR1	1.1238	1.1313	**1.0579**	1.0604
	DK1 → FR2	1.0147	1.0167	**0.9488**	0.9510
	DK1 → AT1	0.9867	0.9911	0.9273	**0.9248**
	FR1 → DK1	1.1694	1.1725	1.1412	**1.1390**
	FR1 → FR2	1.0147	1.0167	0.9885	**0.9477**
60%	DK1 → FR1	1.2083	1.2144	1.1503	**1.1139**
	DK1 → FR2	1.0465	1.0478	0.9877	**0.9853**
	DK1 → AT1	0.9534	0.9574	**0.9424**	0.9445
	FR1 → DK1	**1.1259**	1.1286	1.1374	1.1390
	FR1 → FR2	**1.0465**	1.0478	1.1040	1.1013
70%	DK1 → FR1	1.2415	1.2454	1.1111	**1.1087**
	DK1 → FR2	0.9793	0.9810	0.9772	**0.9767**
	DK1 → AT1	0.9734	0.9757	0.9538	**0.9536**
	FR1 → DK1	1.2088	1.2104	**1.1429**	1.1505
	FR1 → FR2	**0.9793**	0.9810	0.9902	1.0016
80%	DK1 → FR1	1.2400	1.2429	1.1791	**1.1710**
	DK1 → FR2	1.0273	1.0283	1.0234	**1.0223**
	DK1 → AT1	1.0578	1.0587	0.9984	**0.9978**
	FR1 → DK1	1.2590	1.2599	1.2187	**1.2180**
	FR1 → FR2	1.0273	1.0283	**1.0122**	1.0140
90%	DK1 → FR1	1.2329	1.2352	1.1453	**1.1442**
	DK1 → FR2	**1.0631**	1.0639	1.1570	1.1580
	DK1 → AT1	**1.0473**	1.0484	1.0538	1.0542
	FR1 → DK1	1.2394	1.2402	**1.1556**	1.1626
	FR1 → FR2	**1.0631**	1.0639	1.1294	1.1296

TABLE 2 – Valeurs de RMSE pour différents pourcentages de données manquantes à 1000 itérations

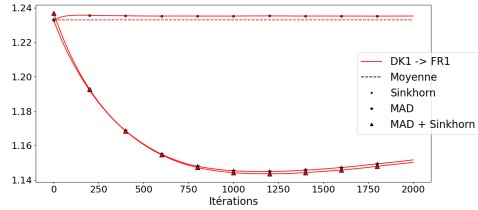
On remarque que notre méthode a de plutôt bon résultat en moyenne. Mais que certains couples de données (FR1 ← FR2) ont de moins bon résultats lors d'une augmentation du pourcentage de données manquantes. De plus comme on le verra par la suite, il peut arriver que notre méthode soit plus performantes que la moyenne avant ou après la 1000-ième itération.

4.4 Evolution de la RMSE en fonction du nombre d'itérations

Ci dessus, on a fixé un nombre d'itérations fixes, regardons l'impact de ce nombre d'itérations sur la valeur de RMSE, dans le cadre de données manquantes avec un biais.



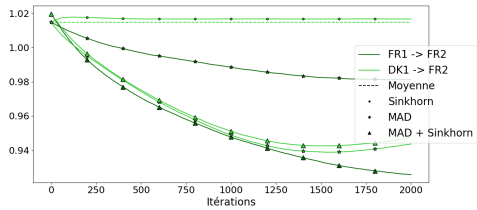
(a) 50% de données manquantes



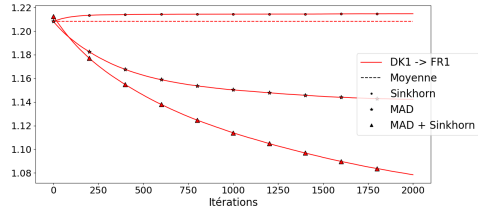
(b) 90% de données manquantes

FIG. 2 – RMSE en fonction du nombre d'itérations

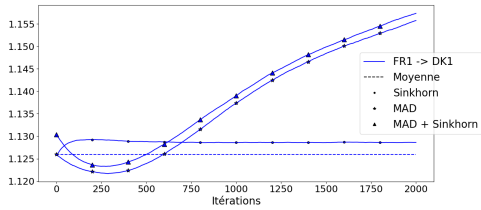
Mais ce n'est pas forcément le cas pour tous, par exemple :



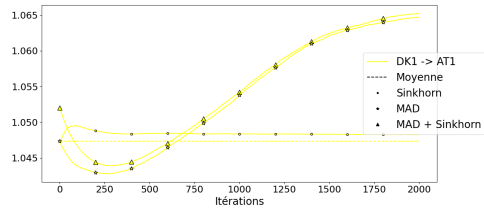
(a) 50% de données manquantes



(b) 60% de données manquantes



(c) 60% de données manquantes



(d) 90% de données manquantes

FIG. 3 – RMSE en fonction du nombre d'itérations

Dans les graphes de la figure **??** et **??**, on remarque que la combinaison de MAD et Sinkhorn n'a pas atteint son minimum. De plus, on peut avoir des cas de figures où la RMSE atteint un minimum plus bas que la moyenne, puis remonte au dessus de la moyenne avant les 1000 itérations, comme pour la figure **??** et **??**. Ce qui pourrait inciter à faire un *early stopping*.

5 CONCLUSION

Pour conclure, on observe que de façon générale, notre méthode a de meilleurs résultats que celle n'utilisant que le transport optimal. Mais il peut arriver que l'on fasse pire après un trop grand nombre d'itérations, ce qui peut nous pousser à chercher à faire du *early stopping*, pour estimer un nombre d'itérations raisonnable. En outre, dans *Missing Data Imputation using Optimal Transport*, il y a une autre méthode qui consiste à entrainer un réseau de neuronne pour faire la complétion de données, en ayant toujours Sinkhorn pour fonction de coût, qui avait de meilleur résultat que l'imputation direct utilisé comme comparatif dans ce papier. On pourrait alors imaginer en faire de même en utilisant MAD comme fonction de coût.

RÉFÉRENCES

[1] Muzellec, M., Josse, J., Boyer, C., Cuturi, M. Missing Data Imputation using Optimal Transport. 2020.

[2] Cuturi, M. Sinkhorn distances : Lightspeed computation of optimal transport. In *Proceedings of the 26th International Conference on Neural Information Processing Systems - Volume 2*, pp. 2292–2300, Red Hook, NY, USA, 2013. Curran Associates Inc.

[3] Genevay, A., Peyre, G., and Cuturi, M. Learning generative models with Sinkhorn divergences. In Storkey, A. and Perez-Cruz, F. (eds.), *Proceedings of the Twenty-First International Conference on Artificial Intelligence and Statistics*, pp. 1608-1617, Playa Blanca, Lanzarote, Canary Islands, 09-11 Apr 2018. PMLR.

[4] Feydy, J., Séjourné, T., Vialard, F., Amari, S., Trouvé, A., and Peyré, G. Interpolationg between optimal transport and MMD using Sinkhorn divergences. *In The 22nd International Conference on Artificial Intelligence and Statistics, AISTATS 2019, 16-18 April 2019, Naha, Okinawa, Japan,* pp. 2681-2690, 2019.

[5] Sakoe, H., Chiba, S. : Dynamic programming algorithm optimization for spoken word recognition. IEEE transactions on acoustics, speech, and signal processing 26(1), 43–49 (1978)

6 ANNEXES

6.1 Manquantes au hasard

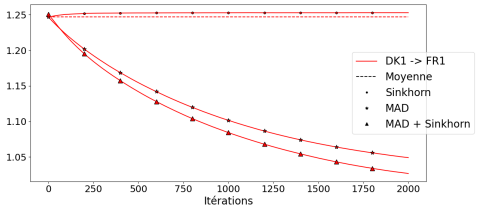


FIG. 4 – RMSE en fonction du nombre d'itération avec 50% de données manquantes

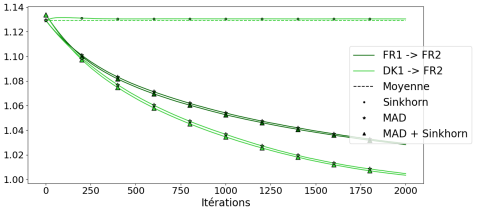


FIG. 5 – RMSE en fonction du nombre d'itération avec 50% de données manquantes

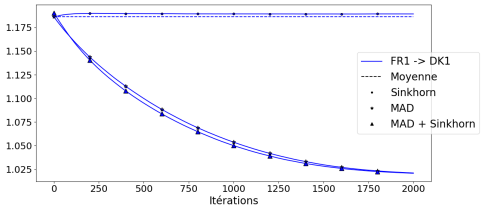


FIG. 6 – RMSE en fonction du nombre d'itération avec 50% de données manquantes

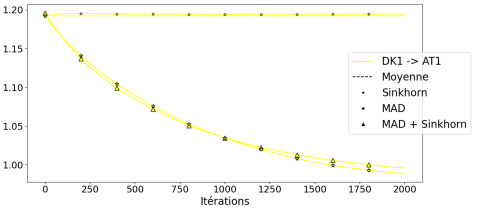


FIG. 7 – RMSE en fonction du nombre d'itération avec 50% de données manquantes

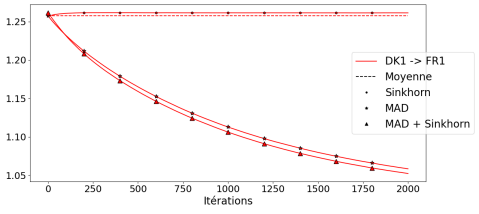


FIG. 8 – RMSE en fonction du nombre d'itération avec 60% de données manquantes

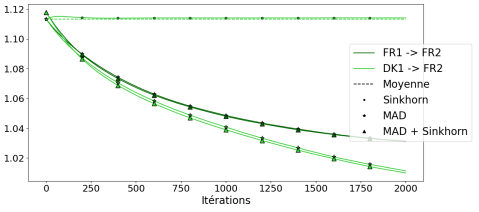


FIG. 9 – RMSE en fonction du nombre d'itération avec 60% de données manquantes

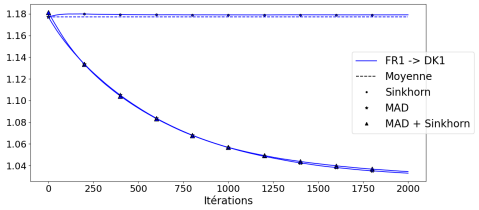


FIG. 10 – RMSE en fonction du nombre d'itération avec 60% de données manquantes

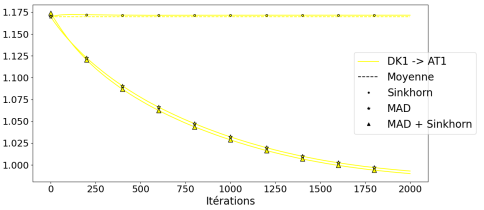


FIG. 11 – RMSE en fonction du nombre d'itération avec 60% de données manquantes

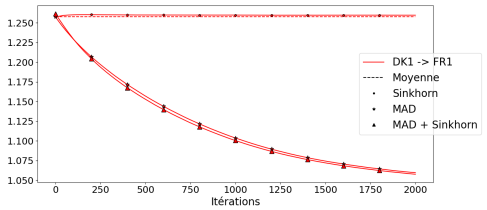


FIG. 12 – RMSE en fonction du nombre d'itération avec 70% de données manquantes

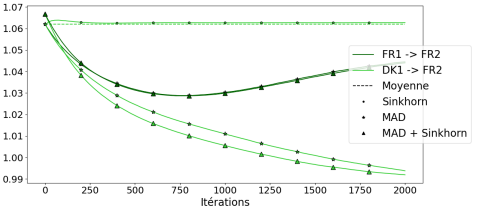


FIG. 13 – RMSE en fonction du nombre d'itération avec 70% de données manquantes

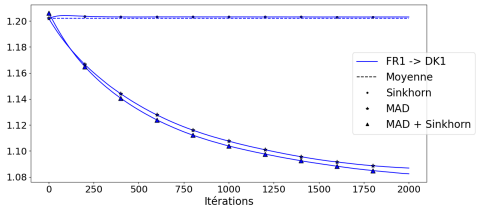


FIG. 14 – RMSE en fonction du nombre d'itération avec 70% de données manquantes

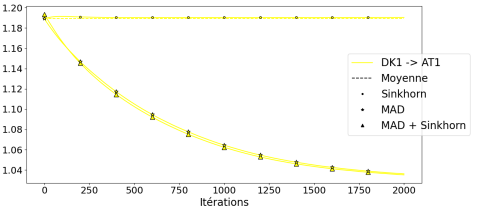


FIG. 15 – RMSE en fonction du nombre d'itération avec 70% de données manquantes

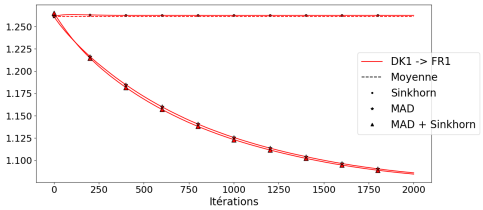


FIG. 16 – RMSE en fonction du nombre d'itération avec 80% de données manquantes

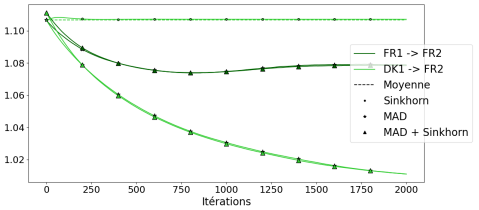


FIG. 17 – RMSE en fonction du nombre d'itération avec 80% de données manquantes

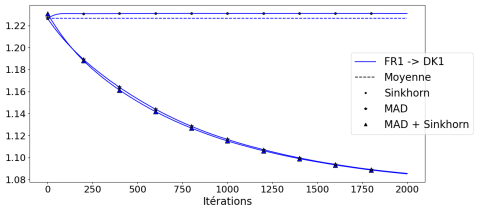


FIG. 18 – RMSE en fonction du nombre d'itération avec 80% de données manquantes

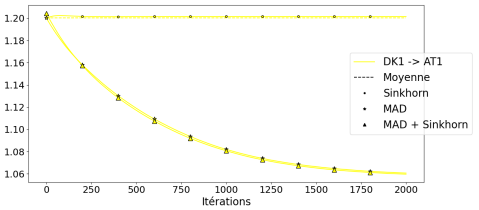


FIG. 19 – RMSE en fonction du nombre d'itération avec 80% de données manquantes

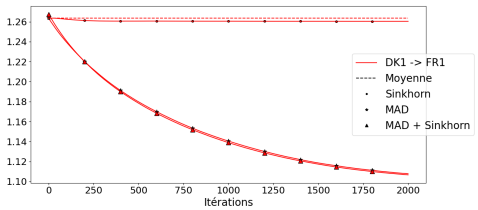


FIG. 20 – RMSE en fonction du nombre d'itération avec 90% de données manquantes

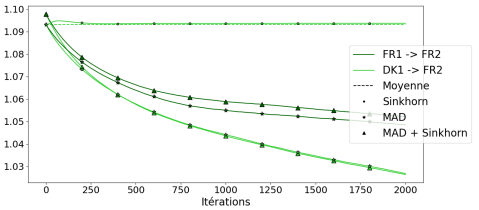


FIG. 21 – RMSE en fonction du nombre d'itération avec 90% de données manquantes

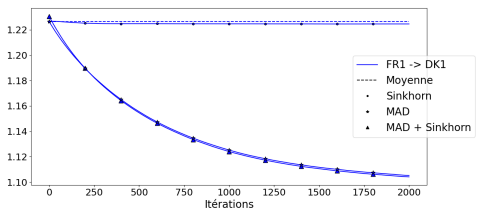


FIG. 22 – RMSE en fonction du nombre d'itération avec 90% de données manquantes

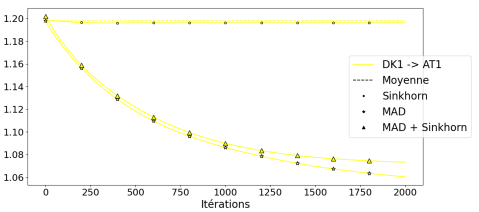


FIG. 23 – RMSE en fonction du nombre d'itération avec 90% de données manquantes

6.2 Manquantes avec un biais

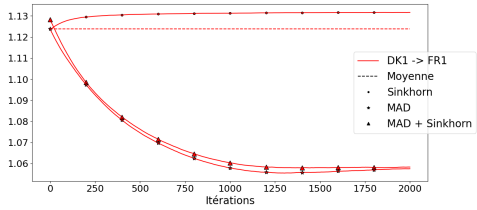


Fig. 24 – RMSE en fonction du nombre d'itération avec 50% de données manquantes

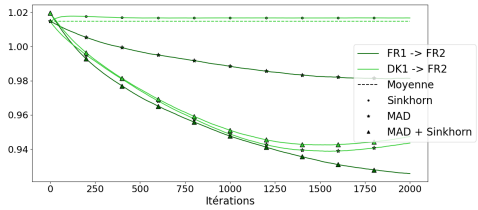


Fig. 25 – RMSE en fonction du nombre d'itération avec 50% de données manquantes

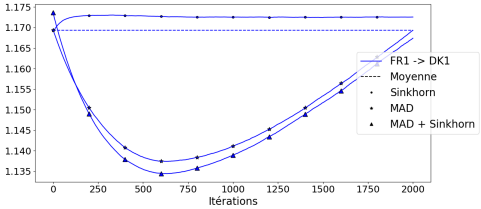


Fig. 26 – RMSE en fonction du nombre d'itération avec 50% de données manquantes

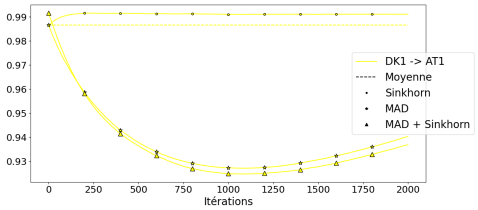


Fig. 27 – RMSE en fonction du nombre d'itération avec 50% de données manquantes

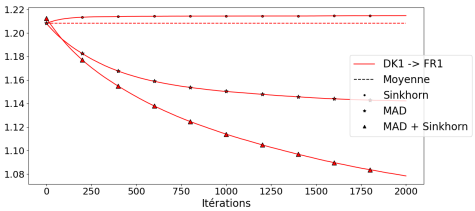


Fig. 28 – RMSE en fonction du nombre d'itération avec 60% de données manquantes

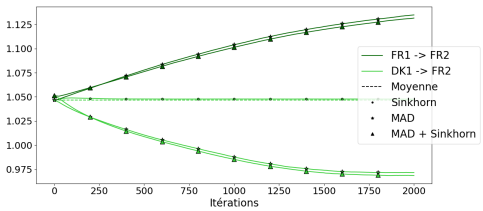


Fig. 29 – RMSE en fonction du nombre d'itération avec 60% de données manquantes

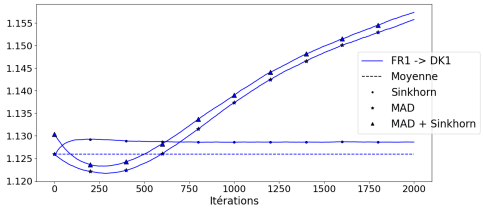


FIG. 30 – RMSE en fonction du nombre d'itération avec 60% de données manquantes

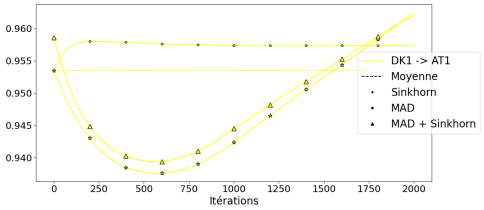


FIG. 31 – RMSE en fonction du nombre d'itération avec 60% de données manquantes

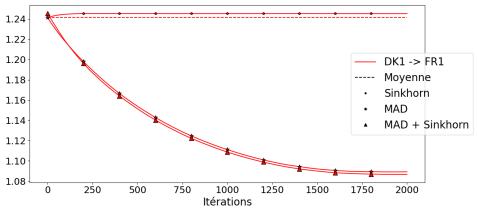


FIG. 32 – RMSE en fonction du nombre d'itération avec 70% de données manquantes

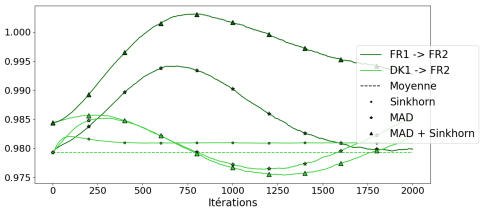


FIG. 33 – RMSE en fonction du nombre d'itération avec 70% de données manquantes

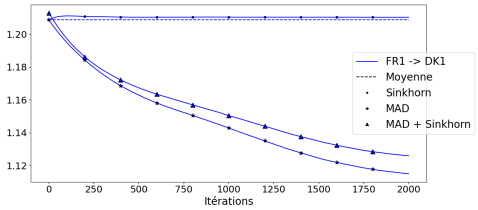


FIG. 34 – RMSE en fonction du nombre d'itération avec 70% de données manquantes

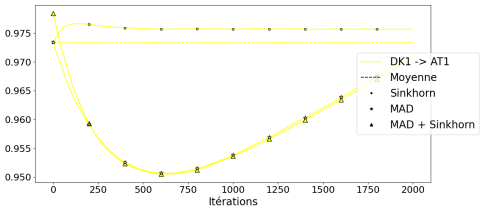


FIG. 35 – RMSE en fonction du nombre d'itération avec 70% de données manquantes

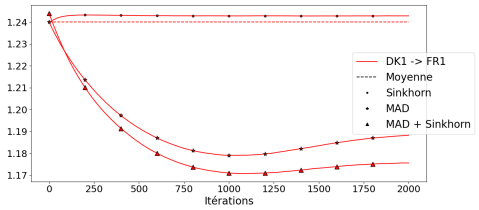


FIG. 36 – RMSE en fonction du nombre d'itération avec 80% de données manquantes

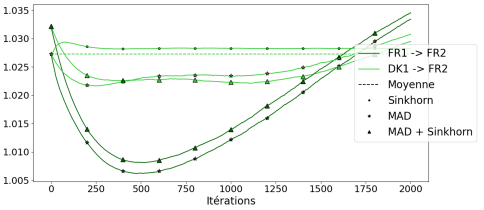


FIG. 37 – RMSE en fonction du nombre d'itération avec 80% de données manquantes

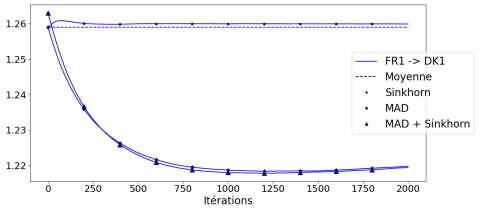


FIG. 38 – RMSE en fonction du nombre d'itération avec 80% de données manquantes

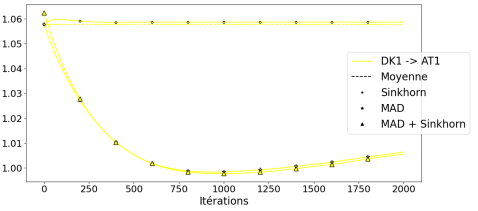


FIG. 39 – RMSE en fonction du nombre d'itération avec 80% de données manquantes

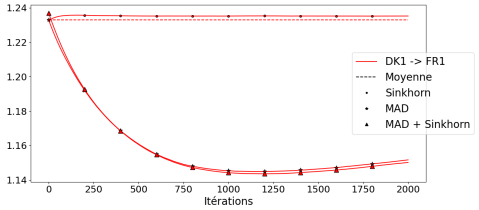


FIG. 40 – RMSE en fonction du nombre d'itération avec 90% de données manquantes

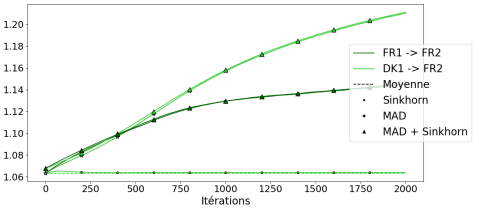


FIG. 41 – RMSE en fonction du nombre d'itération avec 90% de données manquantes

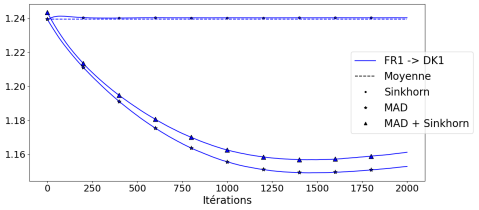


FIG. 42 – RMSE en fonction du nombre d'itération avec 90% de données manquantes

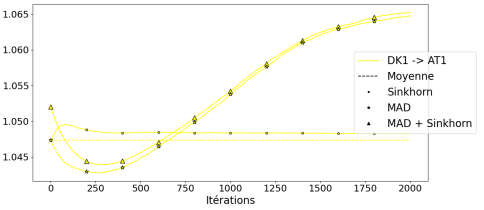


FIG. 43 – RMSE en fonction du nombre d'itération avec 90% de données manquantes