HOUMMADY Enzo HOUE04029900
AMSELLEM Nathan AMSN25080004

Rapport TP2

8INF867 FONDAMENTAUX DE L'APPRENTISSAGE AUTOMATIQUE

1. Préambule

Le modèle de régression linéaire multiple est le modèle le plus largement mis en place afin d'étudier des variables multidimensionnelles. Cette méthode est une méthode de régression mathématique étendant celle de la régression linéaire simple.

Si la régression linéaire simple décrit les variations d'une variable endogène (variable cible) associées aux variations d'une seule variable exogène (variable indépendante : souvent imposée par la nature).
La régression linéaire multiple décrit alors les variations d'une variable endogène associées aux variations de plusieurs variables exogènes.

Le but de ce TP sera donc de mettre en place un pipeline (ensemble de traitements de données acheminant des données afin d'exploiter un modèle d'apprentissage machine) permettant d'entraîner et tester un modèle de régression, afin de pouvoir évaluer les performances de ce dernier et de le comparer à d'autres méthodes.

2. Sélection des données

Pour ce tp, nous devions trouver un dataset dont l'objectif serait de déterminer une variable cible. Nous avons fait le choix de nous porter vers un dataset se nommant : **"Seoul Bike Data"**, qui comprend le nombre de vélos qui ont été loués par heure, sur une durée d'un an. De nombreux paramètre sont présents :

- Jour de l'année (type date)
- Nombre de vélo loué (type entier)
- Heure de la journée (type entier)
- température (flottant)
- l'humidité (flottant)
- Vitesse du vent (flottant)
- visibilité à 10m (flottant)
- Température de rosée (flottant)
- Radiation solaire (flottant)
- Quantité de pluie (flottant)
- Quantité de neige (flottant)
- Saisons en cours (String)
- Vacances ou non (String)
- Service en fonctionnement (String)

HOUMMADY Enzo HOUE04029900
AMSELLEM Nathan AMSN25080004

Ainsi l'objectif de ce TP serait pour nous d'évaluer un modèle de régression linéaire multiples pouvant prédire le nombre de vélo loué à une heure donnée d'une journée selon les autres paramètres (notamment climatique)

3. Traitement des données

Notre dataset étant conséquent et possédant de nombreuses variables il était nécessaire de faire un traitement des données préalables à la régression. Plusieurs problèmes étaient face à nous :

1. Chaque ligne de la variable de date, ne correspond qu'à un jour précis de l'année, cela risque de biaiser et d'apporter que peu à notre régression. (Il en sera plus tard de même pour nos heures cf. stratégie de régression)
2. Nous avons de multiples variables catégoriques
3. Nous avons des données qui correspondent à un service non en fonctionnement ce qui signifie que 0 vélo ont été loués. Ces données risquent d'influencer les résultats de notre régression.

Afin de parer à ces problèmes nous avons mis en place plusieurs solutions :

1. Au lieu de traiter la première colonne comme des dates uniques, nous avons transformé chaque variable en jour de la semaine représenté par des entiers de 0 à 6 (du lundi au dimanche). Cela est plus pertinent car un modèle de régression sera ainsi capable de percevoir si un dimanche de nombreux vélos sont loués, au lieu d'attacher cela à une date fixe.

2. La présence des variables catégoriques à être traitée. Nous avons transformé les saisons et les autres variables de la manière suivante :

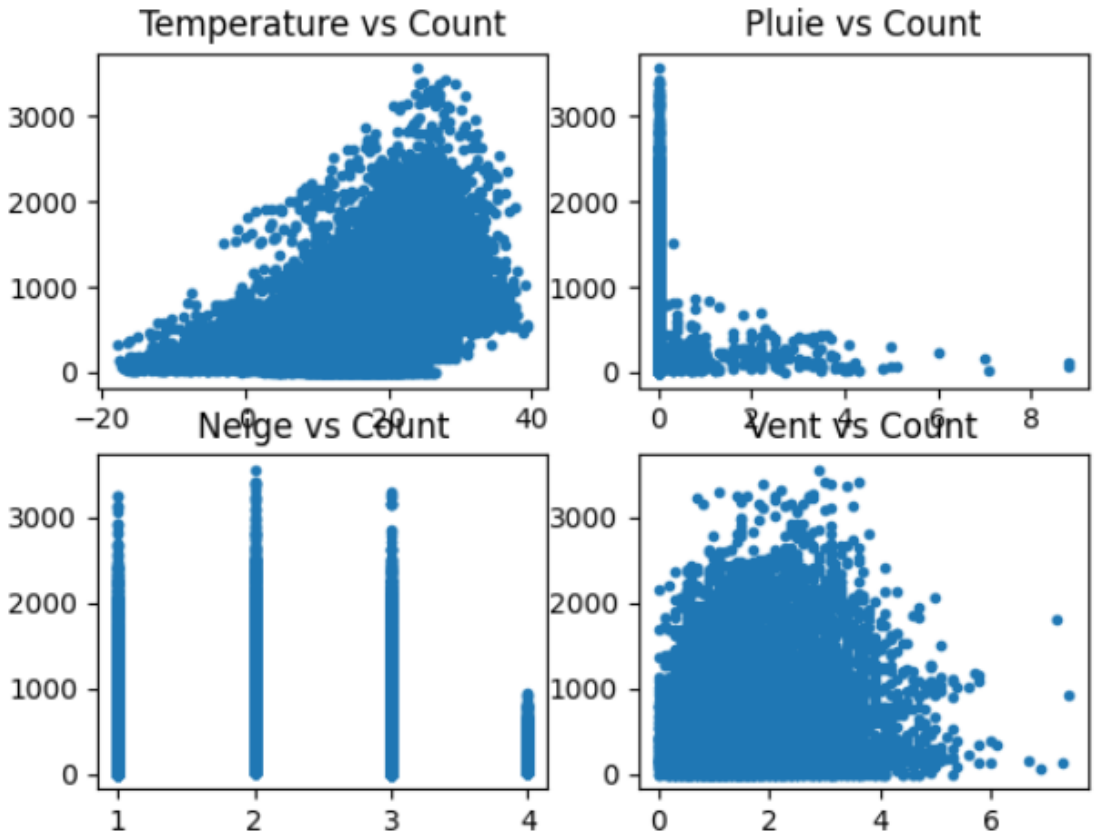
Season	
Winter	4
Spring	1
Summer	2
Autumn	3
Holiday	
Holiday	1
No holliday	0
Functioning day	
Yes	1
No	0

HOUMMADY Enzo HOUE04029900
AMSELLEM Nathan AMSN25080004

Toutes les variables catégoriques ont ainsi été transformées en entier, et sont maintenant utilisables pour notre régression.

3. Le dernier point a été assez facile à traiter, nous ne pouvions pas laisser les occurrences où le service de location de vélo n'était pas en fonctionnement , car le fait que 0 vélo soit loué ne dépendait pas du temps et des autres variables, et cela aurait influencé notre régression.
   Ainsi nous avons juste décidé de supprimer toutes les occurrences, qui correspondaient à un service en non fonctionnement.

Nous avons décidé de visualiser quelques données avec le nombre de vélo loué selon d'autres paramètres, nous constatons que les données sont très disparates.



On imagine donc que la mesure de la variable cible est assez complexe, ce qui implique qu'une régression linéaire simple ne suffirait sûrement pas à expliquer notre variable cible. Nous allons voir maintenant si la mise en place de notre régression linéaire multiple va suffire à pouvoir prédire une telle variable.

HOUMMADY Enzo HOUE04029900
AMSELLEM Nathan AMSN25080004

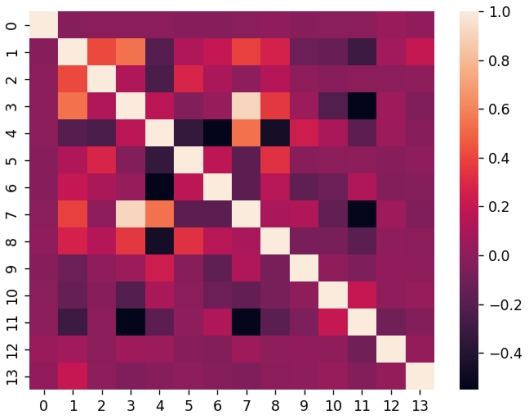
4. Stratégie de régression.

Pour notre régression linéaire nous avons décidé de séparer notre set de données, en deux set distincts, un set d'entraînement comprenant 70% des valeurs et un set de test comprenant 30%. Nous avons utilisé la fonction shuffle plutôt que la fonction train test split de scikit learn.

Puis nous avons appliqué une régression linéaire via sklearn, nous obtenons alors un score assez faible (toujours autour de 0.50) lorsque que nous voulons regarder la précision de notre modèle sur les set de test.. Par curiosité nous avons créé 4 subset, correspondant à chaque saison puis nous avons réalisé une régression pour chaque saison afin de voir si le modèle de régression linéaire pouvait être plus efficace pour une prédiction sur une saison en particulier. Nous avons obtenu des résultats similaires pour chaque saison. Sur la capture ci dessous, les set sont les suivants : total, hiver,printemps,été,automne

Total	Score de la regression : 0.5111529678489418
Hiver	Score de la regression : 0.5332371494624766
Printemps	Score de la regression : 0.5067755705299211
Été	Score de la regression : 0.5305741126309591
Automne	Score de la regression : 0.5255150657066592

Au vu de la visualisation des données qui sont très disparates, cela explique en partie nos premiers résultats d'une régression faite de manière "naïve", sans traiter les données potentiellement corrélées entre elles. Nous avons donc décidé de visualiser la matrice de corrélation de notre dataset.



HOUMMADY Enzo HOUE04029900
AMSELLEM Nathan AMSN25080004

Notre variable cible est la variable notée “1” , on constate qu' assez peu de variable vont être productives pour améliorer le score de notre régression, sauf la variable 3 et 7 qui semble d’ailleurs être colinéaire, il faudra donc sans doute faire une sélection d’une des deux colonnes.

Ce résultat vient du fait que notre MSE est très élevé à cause de la répartition de nos données. Nous avons essayé de normaliser entre 0 et 1 les données “cible” afin de voir si cela pouvait améliorer notre modèle de manière artificielle. Cependant l’effet inverse s’est produit. Ce n’était donc pas la bonne solution à appliquer.

Total	MEAN SQARED ERROR : 203499.51820276014 Score de la regression : 0.32749672460612
Hiver	MEAN SQARED ERROR : 195191.72244664258 Score de la regression : 0.32959033208812
Printemps	MEAN SQARED ERROR : 199765.05645844407 Score de la regression : 0.31972481481987
Été	MEAN SQARED ERROR : 191832.54005698208 Score de la regression : 0.31393826853583
Automne	MEAN SQARED ERROR : 196364.2026305675 Score de la regression : 0.32238598321532

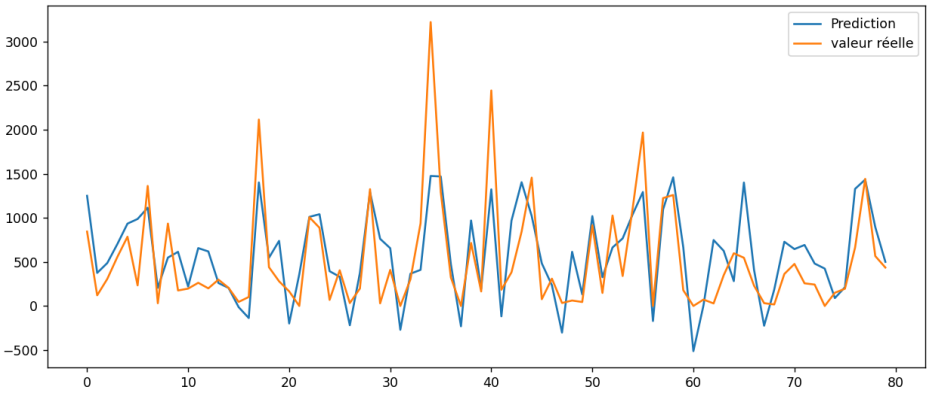
Nous avons ensuite décidé comme expliqué plus tôt de supprimer la colonne 7 due à la colinéarité avec la colonne 3. Mais cela n’a pas amélioré les résultats.

Dans un second temps, au lieu d'utiliser les 24 heures de la journée nous avons décidé de les regrouper en 4 catégories : Matin, journée, Soir, Nuit afin de regrouper par grosse tranche horaire. Nous obtenons alors une légère amélioration de nos résultats de l'ordre de 5%.

Total	Score de la regression : 0.5712883968071654
Hiver	Score de la regression : 0.5538549117700631
Printemps	Score de la regression : 0.5706436205400365
Été	Score de la regression : 0.5913679095226272
Automne	Score de la regression : 0.5711948844895581

Par curiosité nous avons voulu visualiser les données entre les données prédites sur notre set de test, et les valeurs effectives réelles.

HOUMMADY Enzo HOUE04029900
AMSELLEM Nathan AMSN25080004



Nous constatons que malgré nos résultats qui semblent d'un premier lieu assez faible, que notre modèle prédit une tendance assez correcte concernant la location des vélos.
Le problème provient surtout des valeurs de 'pics' réels, qui montent au dessu de 1500 locations environ qui posent effectivement un problème pour le modèle puisqu'il n'arrive pas à gérer ces cas de forte affluence, on observe donc un fort écart entre la prédiction et la valeur réelle pour ces valeurs ci.

5. Comparaison avec lasso et Ridge

Nous avons mis en place avec les mêmes conditions les algorithmes de régression que sont Lasso et Ridge. Encore une fois, les résultats sont exprimés dans l'ordre suivant :total, hiver,printemps,été,automne

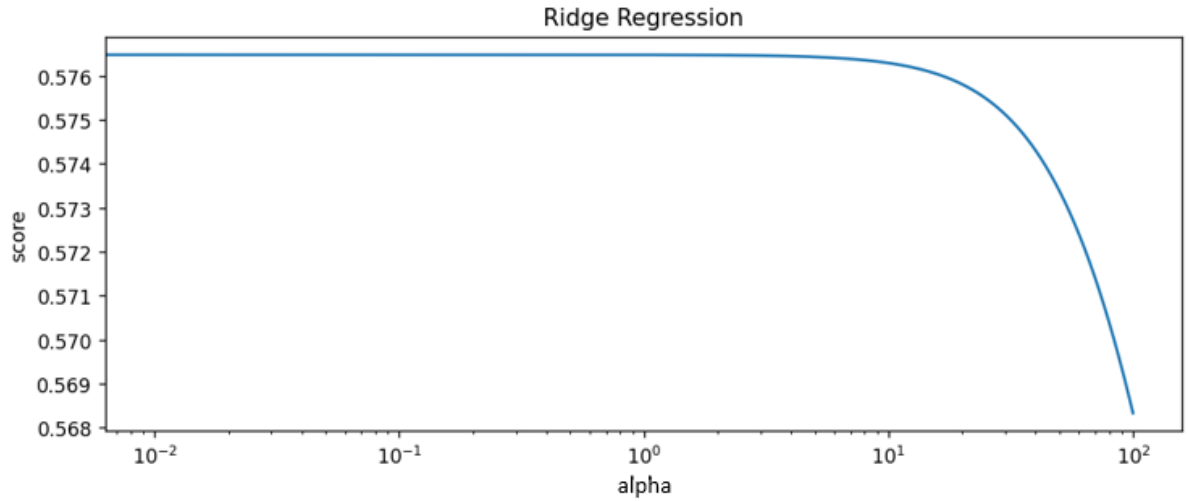
```
----------NORMAL ----------


Score de la regression :  0.5681739054037396
Score de la regression :  0.5870346126159768
Score de la regression :  0.5822331049234969
Score de la regression :  0.5710347559944838
Score de la regression :  0.5849655294638658


----------LASSO ----------


Score de la regression :  0.5681765485255696
Score de la regression :  0.5870327314169623
Score de la regression :  0.5822317659971762
Score de la regression :  0.5710338908134838
Score de la regression :  0.5849639406855541


----------RIDGE----------


Score de la regression :  0.568174044982142
Score de la regression :  0.5870342854642644
Score de la regression :  0.5822317659971762
Score de la regression :  0.5710338908134838
Score de la regression :  0.5849639406855541


---------- MSE ----------


MEAN SQARED ERROR :  178723.11493924636
MEAN SQARED ERROR :  173300.10899568175
MEAN SQARED ERROR :  175338.66020931114
MEAN SQARED ERROR :  182159.99608313054
MEAN SQARED ERROR :  174502.94862984822
```

On observe que les résultats sont les mêmes à l'erreur numérique près. Le coefficient alpha de lasso et ridge ont été placés à 0.01.

Ceci ne représentait qu'une valeur de alpha, nous avons donc voulu nous intéresser à d'autres valeurs. Et donc voir dans quelle mesure la précision du modèle variait en fonction de alpha.

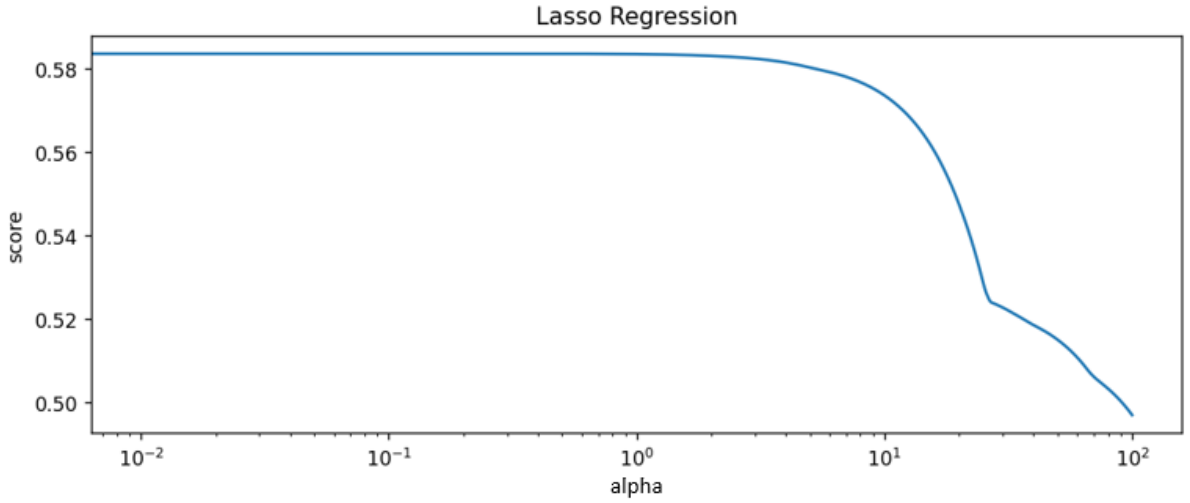
HOUMMADY Enzo HOUE04029900
AMSELLEM Nathan AMSN25080004

Dans un premier temps nous allons donc représenter la variation de la précision du modèle sur une régression effectuée avec ridge et un paramètre alpha variable. Nous obtenons le graphe suivant



L'axe alpha est placé selon une échelle logarithmique et on observe une décroissance pure de la précision du modèle à mesure que alpha augmente.

Dans un premier temps nous allons donc représenter la variation de la précision du modèle sur une régression effectuée avec ridge et un paramètre alpha variable. Nous obtenons le graphe suivant



L'axe alpha est placé selon une échelle logarithmique et on observe une décroissance pure de la précision du modèle à mesure que alpha augmente, on peut préciser qu'une non linéarité apparaît pour alpha = 30 environ.

Après cela nous avons utilisé une régression avec la méthode elastic net de scikit learn en laissant des paramètres de base :

```
----------ELASTIC NET-------------------

Score de la regression :  0.5278405328471231
Score de la regression :  0.5246569367970068
Score de la regression :  0.5306005454185663
Score de la regression :  0.5228549664530404
Score de la regression :  0.5204332397865651
```

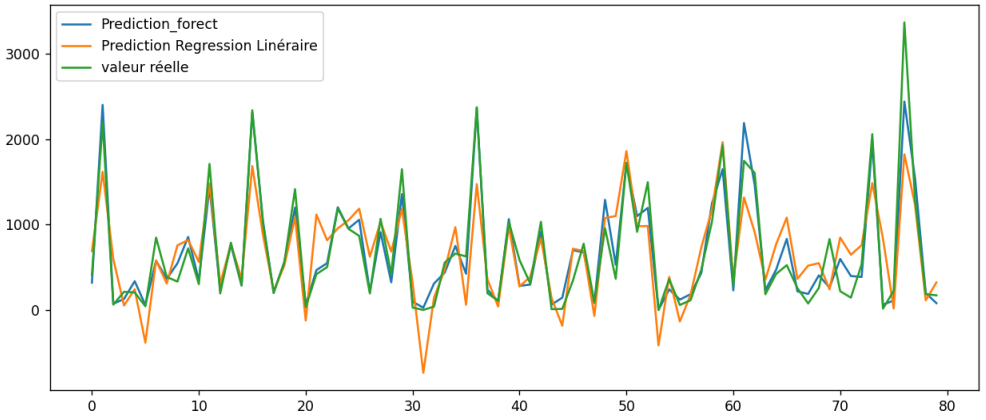
De la même manière, on reste dans les mêmes ordres de grandeur observés précédemment.

En parallèle nous avons voulu avec le random forest de scikit learn :

```
---------- NORMAL DECISION TREE ----------


Score de la regression :  0.8509217033563545
Score de la regression :  0.8515819857878673
Score de la regression :  0.8496934477580672
Score de la regression :  0.8359160932867108
```

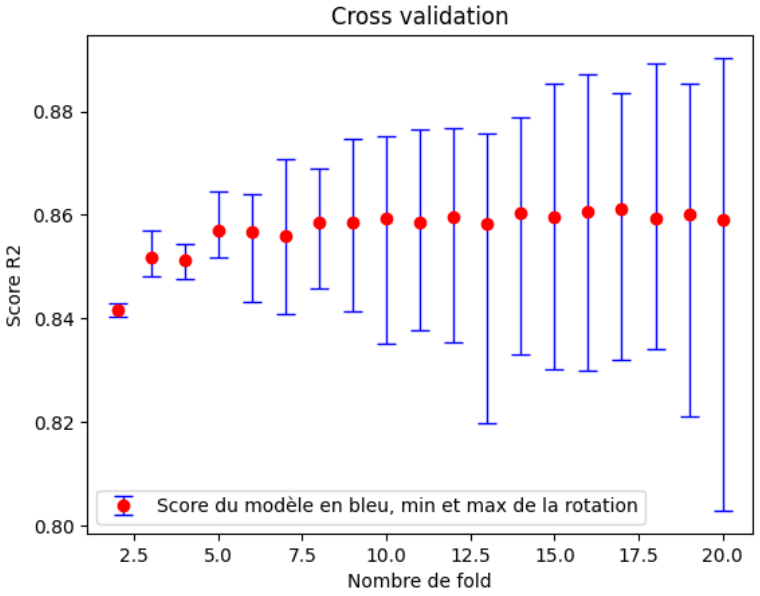
Le normal decision tree quant à lui à ramené des résultats mesurés à un score de 0.85 en moyenne, une amélioration notable par rapport aux modèles observés précédemment. Nous avons donc choisi de poursuivre avec une amélioration des paramètres liés aux paramètres du décision tree plutôt qu'à l'optimisation de l'elastic net.

La dernière étape à donc consisté en une comparaison entre le décision et la régression pour observer si la différence de précision annoncée se répercute bien sur l'observation des prédictions.



On observe donc ici grâce à la visualisation qu'avec random forest (courbe bleue), on suit beaucoup mieux notre valeur réelle pour de fortes valeurs (>1500 vélos), la tendance générale est bien suivie par les deux modèles. En revanche on observe quelques valeurs aberrantes avec la régression (prédictions inférieures à 0, impossible dans notre cas)Nous avons également essayé de calculer les résultats avec la cross validation dans notre dernier cas de RandomForest. Ne sachant pas combien de part découper notre dataset nous avons décidé de calculer le score de la cross validation pour chaque itération de 2 à 21 parts. Nous avons calculé un résultat de R2_score dans un cas théorie qui sera notre score pour comparer chaque itération. Nous avons également récupéré le score max et minimum pour chaque itération ainsi que le score moyen qui lui nous intéresse. Nous avons obtenus le graphique suivant :

HOUMMADY Enzo HOUE04029900
AMSELLEM Nathan AMSN25080004



> folds=2, accuracy=0.842 (0.840,0.843)
> folds=3, accuracy=0.852 (0.848,0.857)
> folds=4, accuracy=0.851 (0.848,0.854)
> folds=5, accuracy=0.857 (0.852,0.864)
> folds=6, accuracy=0.857 (0.843,0.864)
> folds=7, accuracy=0.856 (0.841,0.871)
> folds=8, accuracy=0.859 (0.846,0.869)
> folds=9, accuracy=0.858 (0.841,0.875)
> folds=10, accuracy=0.859 (0.835,0.875)
> folds=11, accuracy=0.859 (0.838,0.877)
> folds=12, accuracy=0.860 (0.835,0.877)
> folds=13, accuracy=0.858 (0.820,0.876)
> folds=14, accuracy=0.860 (0.833,0.879)
> folds=15, accuracy=0.860 (0.830,0.885)
> folds=16, accuracy=0.861 (0.830,0.887)
> folds=17, accuracy=0.861 (0.832,0.884)
> folds=18, accuracy=0.859 (0.834,0.889)
> folds=19, accuracy=0.860 (0.821,0.885)
> folds=20, accuracy=0.859 (0.803,0.890)

Nous constatons que pour un découpage en 8 parties nous obtenons un résultat très satisfaisant. De plus, l'écart entre les valeurs max et min lors de la rotation de l'entraînement est assez faible. Ainsi pour un entraînement du modèle nous utiliserons un Kfold = 8.
De plus, il semble que nous améliorons un tout petit peu la précision de notre modèle.

6. Conclusion

Ce travail nous a permis de mettre en œuvre une première stratégie d'apprentissage automatique avec un pipeline complet. Nous somme passer d'une stratégie de régression "naïve", a des stratégie plus complexe afin d'améliorer nos résultats. D'un traitement des données plus avancé jusqu'à un changement de modèle de régression pour passer sur du random Forest. Explorer des choses que nous avions pas encore pratiquées comme Grid Search CV afin de trouver les meilleurs hyperparamètres de nos modèles, les randomForest et les cross validation fut très plaisant. Atteindre des résultats satisfaisants et pouvoir les visualiser fut très plaisant.

Bibliographie :

https://github.com/syedsharin/Seoul-Bike-Sharing-Demand-Prediction/blob/main/Seol_Bike_Sharing_Demand_Prediction_Capstone_Project.ipynb

https://machinelearningmastery.com/how-to-configure-k-fold-cross-validation/