CHALLENGE VEOLIA 2022: PRÉDICTION DE LA CONCENTRATION DE DIOXYDE DE SOUFRE

Master 2 Modélisations Économiques Statistiques et Financières

Deep Learning

Avril 2022

Fancello Marie Clara, Germini Eva

UNIVERSITÉ PARIS 1
PANTHÉON SORBONNE

1. Introduction

L'objectif de Veolia étant d'apporter une réponse technique aux nuisances olfactives autour de certaines stations d'épuration, le sujet de ce projet est d'étudier la concentration de dioxyde de soufre à la station du Havre, nommée MAS. Le SO2 est un gaz toxique dont l'inhalation est fortement irritante. Notre objectif est donc de répondre à problématique suivante : compte tenu des données mesurées à partir d'un réseau de capteurs ATMO Normandie, des données météorologiques et des données d'occupation des sols, quelles seront les concentrations horaires de SO2 en µg / m³ à la station du Havre au cours des 12 prochaines heures ?

Afin de répondre au mieux à cette question, nous commencerons par présenter nos données et les prétraitements effectués dans une première partie, nous effectuerons dans une seconde partie une analyse exploratoire des données pour finir par proposer un modèle linéaire, un modèle de boosting et deux modèles de Deep Learning, dans une troisième partie.

2. Prétraitements de la base de données

2.1. Présentation des données

La base de données est composée de 1776 variables explicatives numériques, de 288 variables explicatives catégorielles et de 6089 observations. Toutes ces variables sont observées avec 48 lags, ce qui signifie que nous avons en réalité 37 variables numériques et 6 variables catégorielles.
Nous disposons de plusieurs types d'informations : premièrement, des informations temporelles telles que le jour de la semaine ou l'heure, chacunes ayant des lags allant de 1 à 48 heures. Ensuite, nous disposons des moyennes horaires exprimées en µg / m³ de concentration de dioxyde de soufre (SO2) dans l'air dans la région de Normandie pour plusieurs stations (HIR, HAR, CAU, GOR, HVH et STA). Pour ces mêmes stations, nous avons des informations sur la classe d'occupation du sol (classes et polluants). Le dataset contient également des informations météorologiques telles que la température, la vitesse et la direction du vent, la pression atmosphérique, le taux de précipitations, le point de rosée ou encore l'humidité relative.

2.2. Gestion des valeurs manquantes

Le jeu de données des variables expliquées n'a pas de valeur manquante. Cependant, le dataset d'entraînement à 0.03% de valeurs manquantes. Elles sont toutes numériques. Nous décidons de les imputer par la méthode backward car les valeurs extrêmes se produisent dans un intervalle de temps restreint. Effectivement, lorsque l'émission de soufre est élevée à l'heure h, alors elle l'est également pour les heures suivantes.

3. Analyse Exploratoire des Données (EDA) et Datavisualisation

3.1. Etude des variables d'intérêt

L'ensemble des variables d'output ont des médianes inférieures aux moyennes, indiquant la présence d'outliers. La moyenne du dioxyde de soufre est aux alentours de 1.32 µg/m³ dans le dataset d'entraînement, son maximum est à 156 µg / m³. Nous analysons ensuite trois variables du dataset d'entraînement : "SO2_MAS+0", "SO2_MAS+5" et "SO2_MAS+11" car les autres variables sont des déclinaisons temporelles de ces dernières. L'émission de soufre ne présente pas de saisonnalité, ni de tendance. Les outliers constatés précédemment se situent en début et en fin de période.

Enfin, les violin plot confirment que les distributions de ces variables sont fortement concentrées autour de 0, leurs minimums. En effet, avec 61% des valeurs de SO2_MAS fixées à 0, nous sommes confrontés à un problème de "data inflated zero". La variable à expliquer étant une variable continue, cela complique l'adaptation des modèles aux données. Effectivement, ce dernier doit apprendre à prédire des valeurs de dioxyde de soufre entre 0 et 156, en ayant une forte majorité de 0 et uniquement 10% des observations ayant un niveau de dioxyde de soufre supérieur à 2.

3.2. Analyse de la concentration de SO2 aux autres stations

Permettant d'analyser les distributions et la concentration des observations en chaque point, nous utilisons des violin plot afin de comparer la distribution de l'émission de SO2 aux différentes stations avec celle de la station MAS. Tout comme "SO2_MAS", l'ensemble des variables ont une forte concentration de 0. Les variables "SO2_HVH" et "SO2_HAR" ont des distributions similaires à celle de "SO2_MAS". Ensuite, nous regardons si les valeurs extrêmes de "SO2_MAS" se produisent dans un même intervalle de temps que pour les autres stations. Une fois de plus, les variables de HVH présentent le même comportement que les variables de MAS. Également, la variable "SO2_MAS-1" est la plus corrélée à la variable d'intérêt.

La classe d'occupation des sols des stations ne semble pas avoir d'impact sur l'émission de dioxyde de soufre, contrairement à la localisation des stations. A titre d'exemples, la station HAR, qui a la même occupation des sols que MAS « Continuous urban fabric » et qui se situe plus loin de MAS, n'est pas corrélée à cette dernière. Également, la station CAU, corrélée avec un coefficient de 34% à MAS, a une classe « Green urban areas ». Cette station est par extension la plus proche de MAS.

3.3. Analyse des variables explicatives restantes

Nous supprimons premièrement les variables ne prenant qu'une seule valeur, soit 672 variables numériques et 288 variables catégorielles. Les données relatives à la température, ou la direction du vent ont des distributions concentrées autour de leurs moyennes. D'autre part, les variables de pression atmosphérique et d'humidité relative ont des distributions concentrées vers le haut. Finalement, les autres variables telles que le rayonnement solaire descendant ou le rayonnement solaire normal ont des distributions concentrées vers le bas. Nous observons également deux variables ayant la quasi-totalité de leurs valeurs égales à 0 : les chutes de neige par centimètre et les précipitations au centimètre. Ces variables n'étant pas discriminantes, nous décidons de les supprimer. Au niveau des corrélations entre les autres variables explicatives continues et "SO2_MAS" à t+0, elles sont comprises entre -0.19 et 0.16, soit très faibles.

4. Prédiction de l'émission de dioxyde de soufre à la station MAS

4.1. Train/Validation Split

Avant de procéder à la modélisation, nous séparons la base de données en deux parties : un dataset d'entraînement représentant 90% du dataset fournit, ainsi qu'un dataset de validation représentant les 10% restants. Finalement, le dataset de test est celui fournit par l'équipe de Veolia et servira à la prédiction. Cette étape est primordiale pour entraîner nos modèles sur un jeu de données indépendant au reste. Le jeu de validation permet d'ajuster les hyper-paramètres des modèles. Finalement, l'échantillon de test permet d'appliquer le modèle à de nouvelles données entrantes, inconnues au modèle.

Nous procédons à ce découpage de manière aléatoire. Ce choix est motivé par deux facteurs. D'une part, bien qu'ayant des données temporelles, nous n'avons pas de date à notre disposition pour choisir une date de cutoff. D'autre part, l'échantillon de test a été mélangé de façon aléatoire par l'équipe de Veolia organisant le challenge. Ainsi, la séparation aléatoire des échantillons d'entraînement et de validation permet une adaptation du modèle à la structure de l'échantillon test.

4.2. Modélisation

Lors de la première étape de la modélisation, nous entraînons le modèle sur le dataset d'entraînement. Deuxièmement, nous régulons les hyper-paramètres des modèles car sans régularisation, le modèle peut être confronté à un problème d'overfitting. L'overfitting, également appelé le sur-apprentissage, peut être présent au sein d'un modèle lorsque ce dernier est sans biais, c'est-à-dire qu'il a un bon ajustement, cependant, sa variance importante génère de mauvaises prédictions. Intuitivement, cette étape consiste à sélectionner le modèle présentant le meilleur compromis biais-variance. Ainsi, si la

métrique d'évaluation est plus importante sur le train que sur l'échantillon de validation, indiquant la non présence d'overfitting, nous sélectionnons le modèle minimisant l'erreur. Enfin, nous appliquons le modèle sur le jeu de test afin d'obtenir les prédictions. La métrique d'évaluation utilisée dans ce projet est la MSE (Mean Squared Error). Elle mesure l'erreur moyenne au carré entre le réel output et l'output prédit. Dans le cas de multi-régresseurs, la MSE est agrégée sur l'ensemble des variables expliquées.

Dans cette partie, nous utilisons quatre modèles afin de prédire les concentrations horaires de SO2 en µg / m³ au cours des 12 prochaines heures. Tout d'abord, nous présentons deux modèles de Machine Learning : une régression linéaire avec une pénalité "ElasticNet" et un modèle de Boosting. Deuxièmement, nous effectuons deux modèles de Deep Learning : le LSTM (Long Short Term Memory) et le GRU (Gated Reccurent Unit). Enfin, nous comparons les résultats de chaque modèle afin de sélectionner celui avec la meilleure performance.

4.3. Modèles de Machine Learning

Nous effectuons un modèle linéaire et un modèle de boosting car chacun de ces modèles à un avantage que l'autre ne possède pas. D'une part, le modèle linéaire capture mal les données s'il n'existe pas de relation linéaire entre ces dernières, contrairement aux modèles par arbre. D'autre part, les modèles par arbres ne se généralisent pas sur les données qu'il ne voit pas lors de l'entraînement du modèle. Il n'y a donc pas d'interpolation ou d'extrapolation, contrairement aux modèles linéaire.

4.3.1. Régression Linéaire

La régression linéaire, consistant à trouver une relation linéaire entre les variables explicatives et les variables à expliquer, permet une régularisation par pénalisation. Intuitivement, elle consiste à imposer une pénalité sur le vecteur de poids des coefficients. Ainsi, un biais est introduit au sein du modèle, cependant, la variance étant plus faible, le modèle est généralisable sur un nouveau jeu de données. Nous choisissons d'appliquer une pénalité "ElasticNet" (hyperparamètre "l1_ratio) car cette dernière utilise la pénalité "L1" (Lasso) pour supprimer les variables qui sont inutiles dans l'analyse et la pénalité "L2" (Ridge) pour diminuer le poids des variables moins importantes. L'hyperparamètre alpha régule le poids pour la régularisation s'appliquant à la pénalité. Plus les hyperparamètres "alpha" et "l1_ratio" sont élevés, plus la pénalisation est importante.

4.3.2. Modèle de Boosting: CatBoost

Le boosting consiste à entraîner le modèle séquentiellement à partir d'arbres de décision. Le premier arbre construit est un régresseur faible. Les arbres suivants sont entraînés sur les résidus de l'arbre précédent, permettant un apprentissage des erreurs passées. Il existe plusieurs variantes de modèle de boosting comme l'XGBoost ou le LightGBM.

Nous choisissons d'effectuer un CatBoost pour deux raisons majeures. Premièrement, contrairement au LightGBM qui utilise la stratégie du Leaf-wise growth, le CatBoost utilise celle du Level-wise. Ayant un échantillon d'entraînement composé de seulement 5480 observations, la stratégie Leaf-wise a tendance à overfitter, tandis que celle du Level-wise agit comme une régularisation pour limiter la complexité de l'arbre. Deuxièmement, le Catboost échantillonne les observations de manière à maximiser la précision du score de split (Mininal Variance Sampling), tandis que le XGBoost n'utilise aucune optimisation.

Le premier hyperparamètre que nous régulons est "depth" déterminant la profondeur des arbres. Plus un arbre est profond, plus la probabilité d'overfitting est importante. Le second est "learning_rate", paramètre qui permet de contrôler la pondération des corrections ajoutées par un nouvel arbre.

4.4. Modèles de Deep Learning: LSTM et GRU

Le LSTM (Long Short Term Memory) et le GRU (Gated Reccurent Unit) sont deux types de réseaux de neurones récurrents qui pallient les problèmes d'extinction ou d'explosion du gradient et qui sont connus pour être performants sur des datasets séquentiels. Ces modèles sont composés de ce que l'on appelle des portes (Gated) qui aident à capturer les dépendances dans les séquences. Les portes de réinitialisation (reset gate) aident à capturer les dépendances à court terme tandis que les portes de mise à jour (update gate) aident à capturer les dépendances à long terme. Le GRU est une variante simplifiée du LSTM donnant des performances comparables tout en augmentant la vitesse de calcul.

Au sein des modèles de Deep Learning, nous utilisons un Dropout et un Early Stopping. Le Dropout permet de ne prendre en compte qu'une partie des neurones au sein du modèle. L'Early Stopping apporte une vérification quant à la diminution de la fonction de perte à la fin de chaque apprentissage ("epochs" que nous faisons varier). Nous faisons également varier le nombre de neurones dans les couches ("units") et la grandeur de pas ("learning").

4.5. Présentation des résultats

Le benchmark de Veolia correspond à un LSTM avec Dropout et optimisateur Adam. La performance de ce modèle sur l'échantillon de test est de 46.34.

Tableau des résultats:

MSE	ElasticNet	CatBoost	LSTM	GRU
Train	20.62	10.64	21.93	24.82
Valid	14.51	10.57	14.10	14.99
Test	45.78	47.25	43.02	49.03

Le GRU performe moins bien que les autres modèles sur l'ensemble des échantillons. Les modèles ayant une performance plus faible sur l'échantillon d'entraînement que sur l'échantillon de validation performent davantage sur l'échantillon test. Étonnement, le CatBoost ayant une meilleure performance sur le train et sur le valid à une performance plus faible sur l'échantillon de test. Ceci est d'autant plus étonnant que ce modèle ne présente pas d'overfitting (la MSE de l'échantillon "train" est proche de celle de l'échantillon "valid"). Finalement, le modèle retenu est le LSTM qui surperforme le benchmark de 2.

5. Conclusion

En conclusion, l'objectif de Veolia est d'étudier et prédire la concentration de dioxyde de soufre à la station MAS (dans le Havre) afin d'apporter une réponse technique aux nuisances olfactives. Dans cet objectif, nous commençons par pré-traiter nos données, puis dans une deuxième partie, nous effectuons une analyse exploratoire de notre dataset. Cette analyse exploratoire nous permet d'identifier deux problématiques dans ce sujet. D'une part, le dataset d'output présente une forte concentration de 0. D'autre part, il est difficile d'identifier une relation claire entre la matrice de design et les variables expliquées. Enfin, nous effectuons plusieurs modèles de Machine Learning et les comparons afin de maximiser les performances prédictives. Finalement, le modèle retenu est un modèle de Deep Learning LSTM ayant une meilleure performance prédictive que l'ensemble des modèles effectués et que le benchmark de Veolia.