CentraleSupélec

Can you predict the tide ?

Challenge data by INRIA

Akiyo Worou, Théophile Louvet, Tiago Teixeira

14 Février 2021

1 Contexte

Le challenge est proposé en accès libre dans le cadre du Challenge Data, une initiative de l'ENS. Nous avons choisi de participer au challenge "Can you predict the tide" présenté en partenariat avec l'équipe FLUMINANCE de L'INRIA.

Ce challenge s'intéresse au phénomène de la houle, c'est-à-dire la différence entre le niveau de la mer prédit et celui réel, qui peut être dangereux dans les cas extrêmes. Le niveau de l'eau varie principalement en fonction de la marée, qui est un évènement déterministe influencé par la position de la Lune. Cependant, le niveau mesuré est différent de cette prédiction à cause du vent et de la pression atmosphérique. L'objectif de ce challenge est donc de prédire ce phénomène de houle pour 2 localisations différentes sur le flanc Atlantique Nord-Ouest.

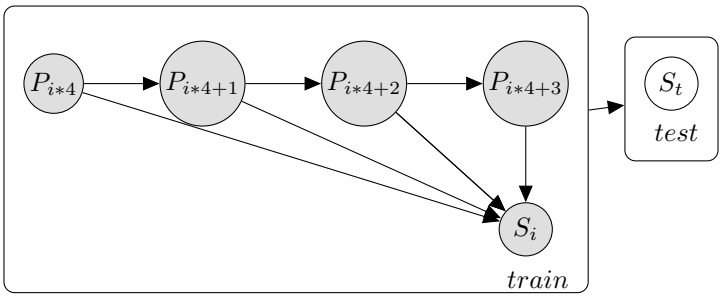
2 Présentation du problème

2.1 Le challenge

La prédiction de la houle se fait ici dans le cadre d'une série temporelle, on connaît les mesures de houle et de pression atmosphérique pendant les 5 derniers jours, et on cherche à en déduire la houle pendant les 5 journées suivantes pour les deux localisations différentes. Durant ces journées, aucune information sur la pression atmosphérique n'est disponible.

Les mesures de pression atmosphérique sont représentées sous la forme d'un champ de

pression, c'est-à-dire une grille de 41*41 points représentant l'Atlantique Nord-Ouest. Pour chaque entrée, ce champ de pression est mesuré toutes les 3 heures, c'est à dire qu'on a 40 observations pour les 5 jours de mesure. En parallèle, la houle est mesurée et est demandée toutes les 12 heures, on a donc 10 points en entrée et 10 valeurs à prédire pour chaque entrée.



Ci-dessus, se trouve une représentation du problème sous forme de réseaux bayésiens. Sur les mesures pour l'entraînement, il est supposé que chaque valeur de houle (notée P) dépend de la pression (P) lors des 4 derniers échantillonnages. Les valeurs de sorties sont influencées par l'ensemble des entrées du modèle.

Le dataset d'entraînement contient 5599 entrées ; dont chacune représente 5 jours de mesures et 5 jours à prédire. Ainsi, on a 5599*40 mesures du champ de pression, ainsi que 5599*10 mesures de houle et valeurs à prédire. Le dataset de prédiction contient quant à lui 509 entrées, donc 509*10 valeurs à prédire pour chaque port. La métrique de mesure du score est l'écart moyen au carré sommé pour les deux villes, avec un poids plus important pour les premières prédictions que pour les dernières.

2.2 L'état de l'art

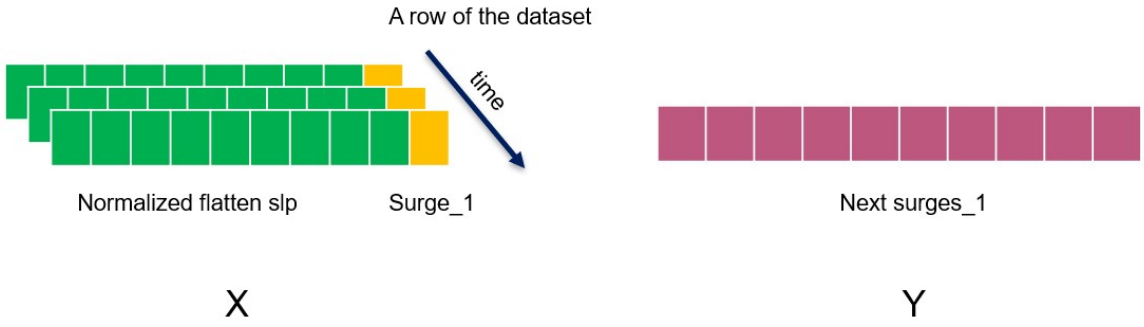
Avant d'essayer de résoudre le problème, nous nous sommes intéressés à l'état de l'art des méthodes de prédiction de la houle. Lors de la présentation du challenge, les organisateurs fournissent une méthode servant de benchmark. La prédiction utilise la méthode des K-Nearest-Neighbors pour déterminer les journées connues les plus proches du cas actuel, la prédiction est la moyenne des houles des 8 mesures les plus proches et obtient un score de 0.7.

Dans le cadre de la prédiction de série temporelle, une approche commune utilise des modèles auto-régressifs sur les mesures précédentes de houle (Weiss et al., 2012. Cependant, le modèle le plus commun est basé sur la physique, en appliquant les équations de Navier-Strokes au déplacement vertical de l'eau sur la côte. (Lazure, 2008). Récemment, les avancées dans le domaine des réseaux de neurones récurrents permettent une nouvelle approche (Braakmann-Folgmann et al., 2017)

3 Nos expérimentations

Nous avons d'abord essayé d'appliquer certaines méthodes de preprocessing. Pour se faire, on a supposer que l'écart de temps entre les t_{slp} reste le même. Ensuite on a interpolé les

données surges sur le même pas de temps que les données slp pour avoir une même base temporelle. Après on fait un flatten des données slp et on rajoute une dernière valeur qui est la valeur du surge associé. On obtient ainsi deux jeux de données associés à $surge_1$ et $surge_2$.



Le processus est expliqué dans le graphique ci-dessus. Les données de slp ont été normalisées pour certaines applications et non normalisées pour d'autre.

Hidden Markov Model (HMM)
Nous avons pensé à utiliser une HMM pour modéliser le problème. Nous avons des observations que sont surge et slp. Des moments de hausses et de baisses qui peuvent par exemple représenter les états cachés ainsi que la localisation de la zone considérée en fonction de l'index de surge($surge_1$ ou $surge_2$).
Nous avons utilisé la librairie hmmlearn de python afin de poser le problème sous forme de HMM. Nous avons uniquement utilisé les données X ainsi que l'algorithme de viterbi afin d'estimé les états cachés. Ensuite pour nos prédictions il suffit de déterminer le prochain état le plus probable et enfin de tirer l'observation sachant qu'on est dans l'état prédit. Ceci permet de prédire à la fois slp et surge.
Malheureusement nous n'avions pas eu le temps de faire nos prédictions car le modèle prenait énormément de temps pour réaliser les prédictions.

Une autre solution possible aurait été d'utiliser un filtre de Kalman, avec un vecteur d'état représentant la houle, et des observations grâce au champ de pression. Ce modèle à l'avantage de pouvoir fonctionner en l'absence de mesures, et intègre l'erreur liée à l'évolution de l'état. Cependant, il est difficile d'estimer les paramètres du modèle. L'influence des observations de pression peut être choisie comme linéaire et les coefficients estimés par une régression des moindre carrés, mais le modèle d'évolution entre deux états est impossible à estimer. Ainsi nous avons choisi de ne pas implémenter cette méthode.

3.1 Multi regression

Les modèles cités précédemment, sont basés sur la représentation bayésienne, ne permettent pas de résoudre le problème. Une première option est de considérer que les valeurs de houle à prédire sont une combinaison linéaire des mesures de pression dans les jours précédents. Étant donné que la relation est différente en fonction du temps et de la position, un estimateur est entraîné par localisation et par point temporel à prédire, soit un total de 20 modèles.

Cependant, les entrées du modèle comportent les 40 mesures de pression avec plus de

1600 points chacune, soit un total de plus de 65 000 variables. Ayant 5599 points pour apprendre, il y a un grand risque d'overfitting sur les données. Afin de réduire ce risque, il est nécessaire de réduire la dimension de l'entrée, cela est fait en utilisant l'analyse des composants principaux. En réduisant les cartes de 41*41 points à 15 valeurs, on garde plus de 95% de la variance du dataset.

Pour améliorer les performances du modèle et le temps de calcul, les données sont centrées et réduites. Premièrement, la régression des moindres carrés est utilisée pour estimer la relation entre les entrées et la sortie. Cependant, bien que le risque soit faible sur le set d'entraînement, le modèle ne généralise pas bien sur les données de test. Afin d'introduire de la régularisation on utilise la régression LASSO, qui permet d'obtenir un score de 0.75 sur les données de test du challenge.

Les prédictions faites précédemment n'utilisent pas les données de houle sur les jours précédents, qu'on peut rajouter en entrée du modèle. Cette information supplémentaire combinée avec des régresseurs LASSO nous permet d'obtenir un score de 0.4161 sur les données d'évaluation. Une amélioration supplémentaire est possible en utilisant Elastic-Net, ce qui permet de combiner une perte L1 et L2.

Il peut être intéressant d'intégrer une non-linéarité dans le modèle pour mieux prédire la houle, pour cela on choisit d'utiliser une NuSVR, qui permet de passer dans un espace de plus grande dimension grâce au kernel trick. Malgré une recherche extensive à travers l'espace des paramètres avec à la fois avec un noyau gaussien et polynomiale, ce modèle n'arrive pas à améliorer les performances de ceux précédents.

3.2 Réseau de neurones récurrents

Dans une perspective d'améliorer notre score nous avons essayé d'utiliser des réseaux de neurones récurrents, qui permettent de prédire de multiples valeurs à la fois en prenant en compte l'aspect temporel des mesures. Pour l'architecture de nos modèles, on utilise des couches de neurones GRU (Cho et al., 2014) ou LSTM (Hochreiter, Schmidhuber, 1997), suivi de multiples couches denses. Le modèle est entraîné avec une descende de gradient stochastique et une perte quadratique.

Les performances des modèles entraînés sont toutefois décevantes : d'un côté, le modèle obtient des scores faibles sur le set d'entraînement, mais les scores sur les données d'évaluation sont pires car le modèle ne généralise pas. L'architecture utilisée est de 24 couches de 512 unités et 20 couches denses. Le meilleur score sur le set d'entraînement est de 0.66.

Il serait possible de trouver une architecture plus pertinente, ou de mieux préparer les données pour obtenir un meilleur score. Mais en considérant que l'objectif de ce projet n'est pas de faire du deep learning nous avons choisi de ne pas poursuivre ce type de modèle.

4 Les résultats

Modèle	Risque
LASSO (sans houle)	0.6168
LASSO	0.4161
ElasticNet	0.4072
NuSVR	0.5973

Le tableau précédent permet de mettre en valeur l'importance d'utiliser l'information sur les houles en entrée du modèle. De plus, la combinaison des pénalisations L1 et L2 d'ElasicNet permet à ce modèle d'être le plus performant. Notre score final est de 0.5238, ce qui signifie que notre modèle garde environ 47.6% de la variance. Au moment d'écrire ce rapport nous sommes donc $6^{ème}$ sur les 15 participants.

5 Conclusion

Le challenge auquel nous avons participé a pour but de prédire les valeurs de houle sur une période de 5 jours, en connaissant les mesures sur les journées précédentes. Bien que les données de pression en entrée soient de grandes dimensions, la difficulté vient du fait qu'aucune information n'est disponible sur le champ de valeur à mesurer. D'après la représentation en réseau bayésien, les approches évidentes sont les chaînes de Markov cachées et les filtres de Kalman, bien qu'en pratique il est difficile d'appliquer ces solutions.

Nous avons donc choisi d'utiliser de multiples estimateurs pour les différents instants et localisation à prédire, avec une réduction de la dimension des données d'entrée par une APC. En ajoutant les informations de la houle lors des jours précédant la prédiction s'améliore grandement. Le meilleur prédicteur utilise l'algorithme d'ElasticNet.

Bien que les méthodes de deep learning soient prometteuses, l'utilisation d'une architecture simple ne permet pas d'obtenir des résultats concluants. Il est nécessaire d'augmenter la complexité du modèle, ce que nous avons choisi de ne pas faire afin de se focaliser sur une modélisation plus statistique.