Rapport de Projet IA

Louis Descamps

Sujet

Le but du sujet est de vérifier l'hypothèse suivante

> Les performances des algorithmes de Machine Learning permettent de modéliser une tendance d'une manière efficace.

Pour cela nous tenterons de modéliser les tendances du covid19 et de prédire son évolution au sein d'une population.

Autrement dit notre but est de créer une IA qui peut :

- Prédire le nombre de décès à venir liés aux Covid.
- Prédire le nombre de nouveaux cas positifs.

Nous travaillerons avec un ΔT d'une journée (24h).

Choix et Analyse des données

On travaillera sur l'ensemble du territoire français en exploitant les données de synthèses des indicateurs du ministères de la santé.

On pourra lire la description des données sur le site de téléchargement, voici un rapide survol :

Les bases de données diffère par la prise en compte de la région et du département.

Contexte : Chaque données est entourée de son contexte avec les champs `date`, `dep` (Département) et `reg` (Région).

Situation Hospitalière : On note 3 catégories majeures `hosp` (pour les hospitalisations), `rea` (pour les patients en réanimations), `rad` (pour les retours à domicile suite à une entrée à hôpital). Ces trois catégories sont complétées par `incid_***` Le nombre de nouveaux admis/relâchés en 24h.

Décès : Notamment `dc_tot` (le nombre de décès totaux dû au Covid19 dans des établissement de santé).

Tests : Qui inclue notamment les cas confirmés `conf`.

Indicateurs mathématiques : Ceux qui pourraient nous être particulièrement utile étant `R` le facteur de reproduction du virus, `TO` La proportion d'occupation des services d'urgences par des cas Covid.

On travaillera sur la base de donnée qui ne prends pas en compte les départements et les régions. On mettra au point notre algorithme sur cette base de données. On pourra utiliser les données par région pour vérifier la capacité de prédiction de l'algorithme dans un second temps.

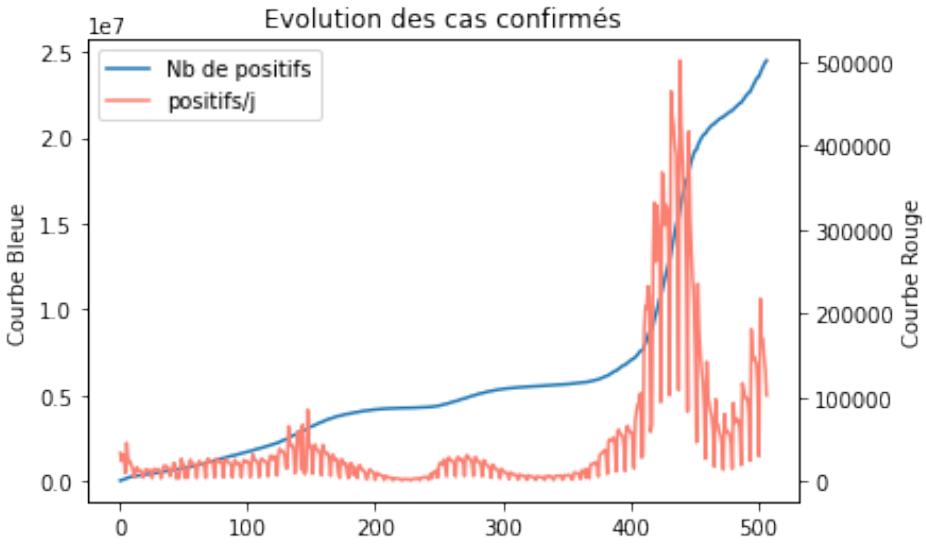
On gardera uniquement huit colonnes. `date`, `TO`, `R`, `hosp`, `rad`, `conf`, `conf_j1` et `dc_tot`. On travaillera sur ce jeu de données plus petit pour simplifier les calculs et permettre un meilleur apprentissage de l'algorithme. Beaucoup de ses données sont absentes en parties de la base de données. On utilisera `conf_j1` pour calculer `conf` avec la formule `conf(n) = conf(n-1)+conf_j1(n)`. On utilisera `dchosp` et `esms_dc` pour calculer `dc_tot` avec la formule `dc_tot = dchosp + esms_dc`.

Les manquements dans la base de données ne permettent pas de commencer l'étude avant novembre 2020.

Visualisation des données

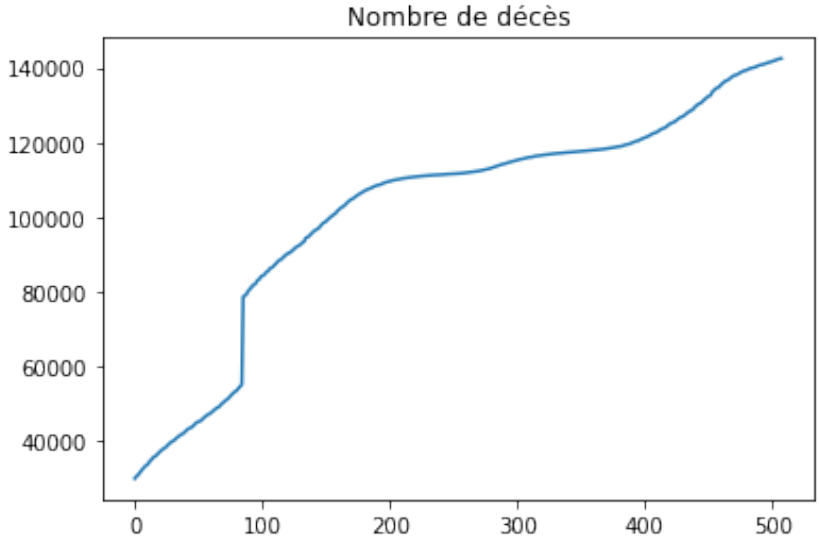
On regarde l'allure des données afin de mieux préparer nos modèles mathèmatiques.

Cas Confirmé

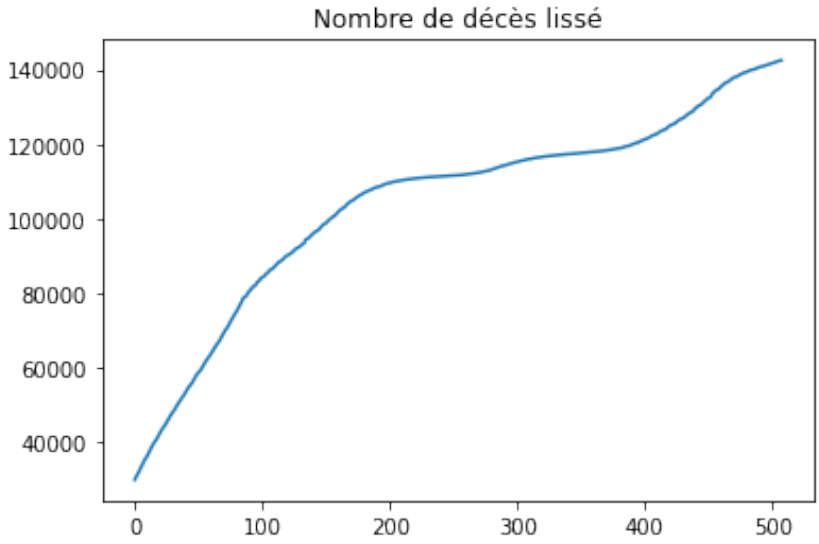


On voit que la courbe qui sera probablement la plus simple à modéliser et prédire sera le nombre total de cas confirmé depuis le début de l'épidémie (en bleue).

Nombre de décès



Le nombre de décès total saute brusquement entre le 4 et le 5 février 202, c'est la prise en compte des dècès en établissement externe aux hopitaux (exemple : ehpad). On va lisser leurs ajouts sur les 100 premiers jours pour éviter un saut si violent qui pourrait altérer l'apprentissage. Une fois lissé on obtient :



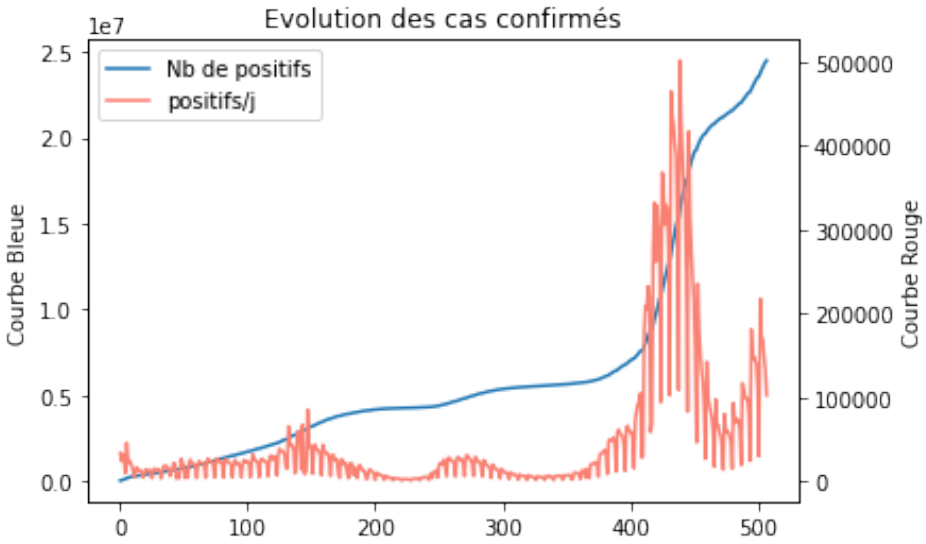
La courbe comprends encore un point de discontinuité (de la dérivée) au même endroit que précément mais cette discontinuité est bien sera bien plus négligeable lors de l'apprentissage.

On crée une nouvelle base de donnée `dataframe_clean` avec ces modifications que l'on utilisera dans la suite. Elle contient 508 lignes et 8 colonnes.

Régression

La première modélisation que nous allons réaliser sera une régression. On travaille ici sur des données temporelles : le but est de prédire une partie de la courbe sur laquelle nous n'avons aucune valeurs expérimentale (le futur ne s'étant pas encore produit). La régression est très efficace pour créer une fonction continue a partir de donnée discrète hors ce n'est pas notre cas du tout.

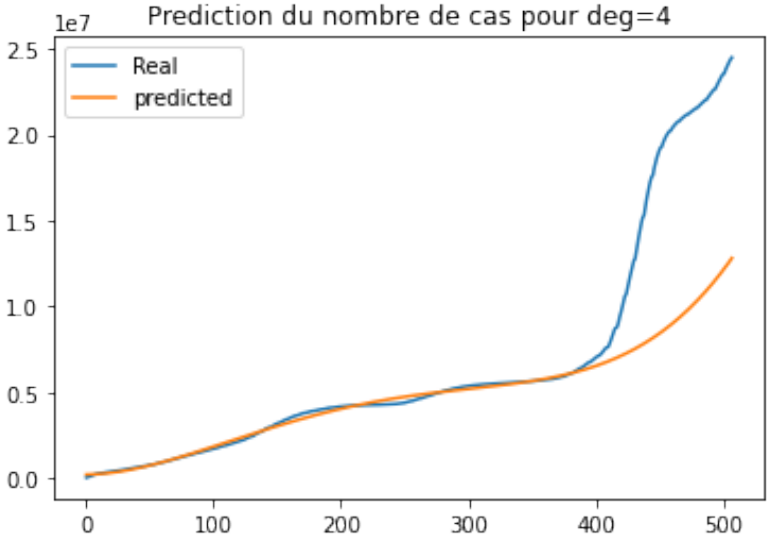
Nombre de Cas



Comme dit précédemment nous utiliserons la courbe en bleue afin d'entrainer notre modèle. On voit ici que l'on a un changement violent de tendance de la courbe bleue lors de la vague de janvier 2022 (qui corresponds à la sortie de la logique des confinements répétés du gouvernement).

Regression Naive

On met en place une régression polynomiale (avec RIDGE) pour tenter de prédire la courbe. On utilise les 2/3 de premières valeurs pour tenter de prédire toute la fin de la courbe.

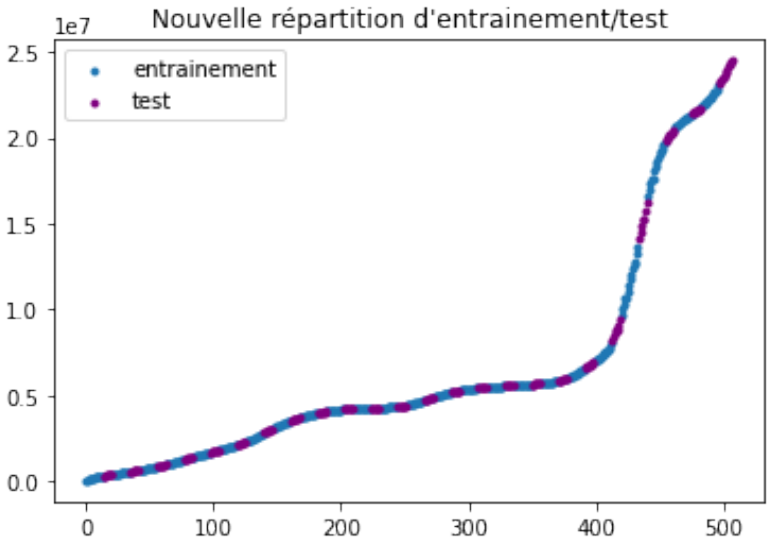


On a du mal à avoir une prédiction correcte comme on l'avait prévue. La courbe ci-dessus a le meilleur score de toutes les courbes de prédiction possible (en changeant les degrés des polynomes) avec un score de test de -0,679. C'est une très mauvaise prédiction.

En revanche, on peut probablement estimer le nombre de cas par régression sur de petites durées.

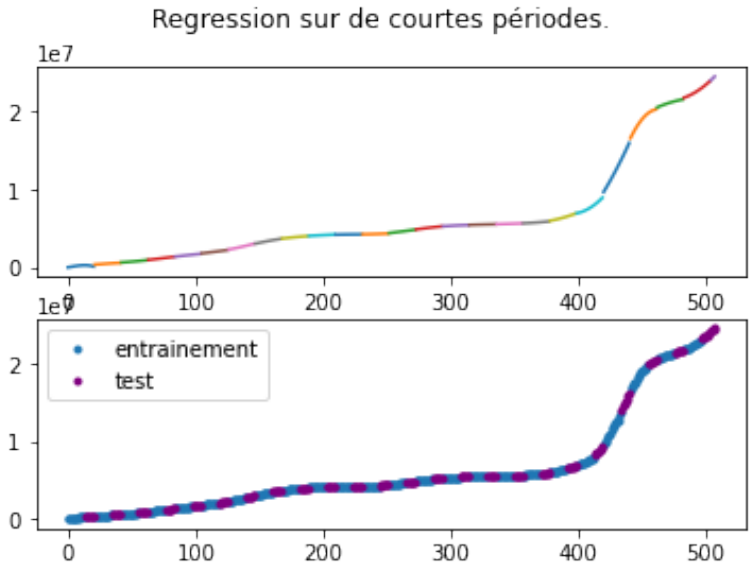
Regressions sur de courtes durées

Changeons de paradygme et tentons de prédire pour une semaine (7jours) le nombre de cas en utilisant le nombre de cas des deux semaines passées. (On garde les proportions 2/3 d'entrainement et 1/3 de test.)



Voici la nouvelle répartition de nos batchs d'entrainement et de test des données. On utilisera uniquement les deux semaines (bleues) précédant une phase de test (violette) pour mettre en place un modèle et le tester sur la dite phase de test d'une semaine.

On met donc en pratique la même solution que précèdemment et on obtient :



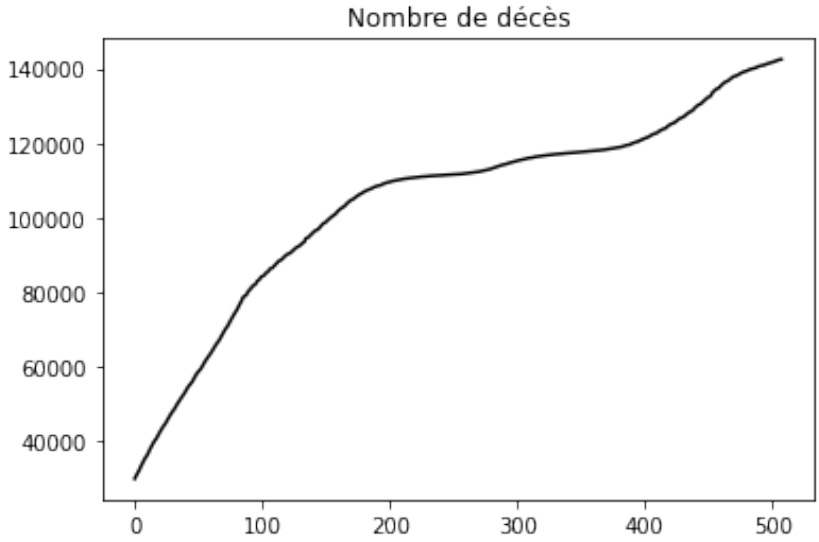
Avec un score de précision de 0.9999/1.

Conclusion du nombre de cas

On peut créer un bon modèle qui peut prédire l'évolution des cas sur une courte période (1semaine) mais on est incapable de créer un système qui prédit l'évolution du nombre de cas sur le long terme.

Nombre de décès

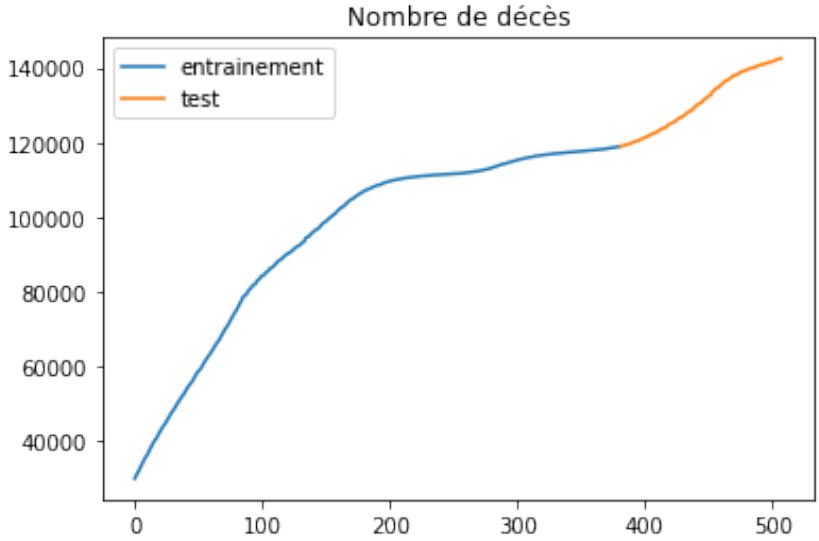
Nous allons maintenant travailler sur le nombre de décès sur la même période.



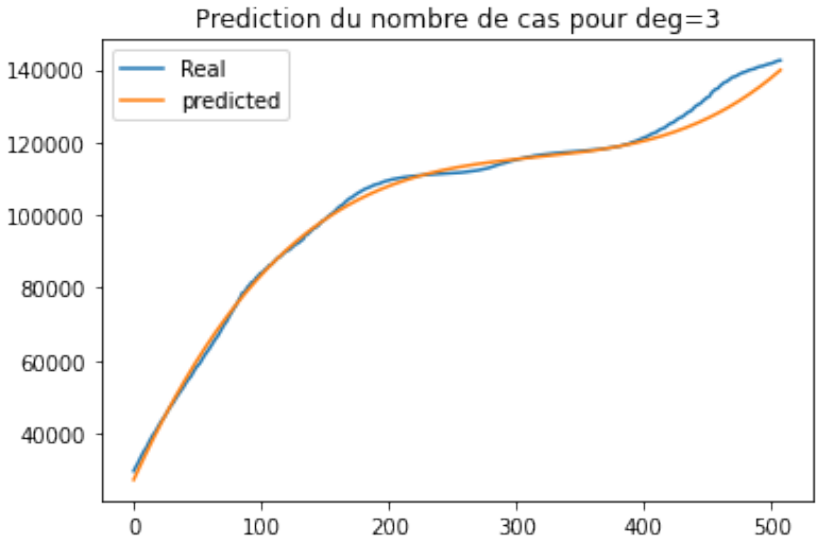
On risque d'être confronter à la même prolématique que précédemment, la prédiction risque d'être très approximative si l'on tente d'entrainer le modèle sur une longue période pour prédire une longue période. Vérifions le de suite.

Regression Naïve

Nous allons donc comme précèdemment utiliser les 2/3 premières valeurs pour prédire le reste.



On crée un modèle polynomial avec Ridge. On obtient des résultats correcte :



On a un score de test de 0,64/1. Le problème étant qu'on peut imaginer qu'on a une mauvaise généralisation des données (puisque qu'on a une courbe qui tends vers l'infini).

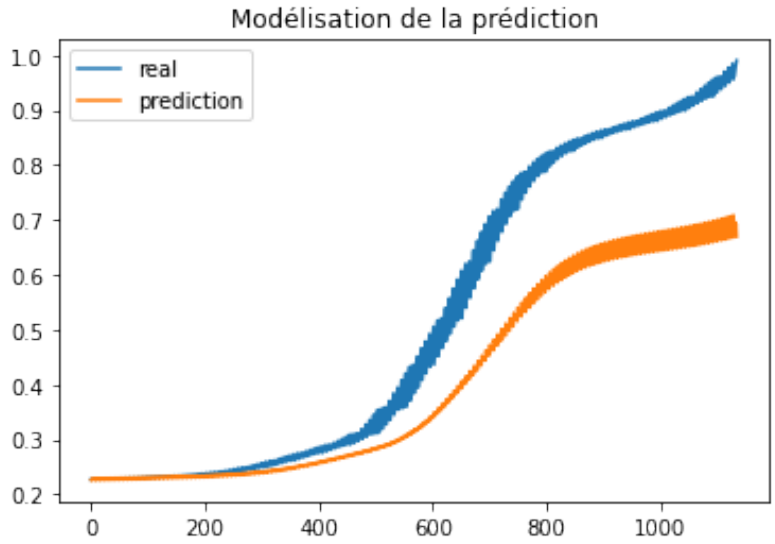
Reseaux LSTM

Monovariable

On met en place un réseaux avec une seule variable qui va tenter (comme précédemment) de deviner l'évolution du nombre de cas pour les 7 jours à venir selon le nombre de cas des 14 jours précèdent.

On met donc en place un réseaux à trois couche avec en entrée 14 neuronnes et en sortie 7 neuronnes différents.

Avec un peu de temps de calcul on obtient :

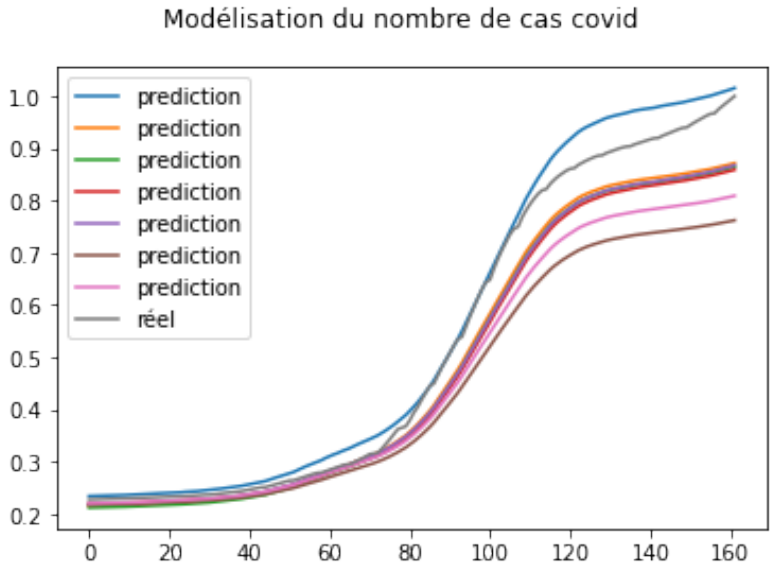


On voir que la prédiction n'est pas bonne et n'arrive pas à rattraper la réalité.

Multivariable

Nous allons cette fois ci utiliser plusieurs variables afin de tenter de prédire le nombre de cas

On utilisera Le nombre d'hospitalisation `hosp`, le nombre de retour à domicile `rad`, l'indicateur de reproduction du virus `T0` et le nombre de cas `conf` des deux dernières semaines pour tenter de prédire le nombre de décès de la semaine suivante.



On obtient une prédiction relativement correcte avec une erreur de 0,07 sur la phase de test. On voit aussi que le premier jour (en bleu) est extrêmement bien prédit et les jours suivants le sont avec plus d'approximation.

On ne met en place que 7 epoch pour éviter l'overfitting.