Modélisation de la résistance du béton haute performance à l’aide de réseaux Neurone.

# Introduction :

Plusieurs études indépendantes ont montré que le développement de la résistance du béton est déterminé non seulement par le rapport eau-ciment, mais qu'il est également influencé par le contenu d'autres ingrédients du béton. Le béton à hautes performances est un matériau très complexe, ce qui rend la modélisation de son comportement très difficile.

Cette étude a conduit aux conclusions suivantes : 1) Un modèle de force basé sur ANN est plus précis qu'un modèle basé sur une analyse de régression ; et 2) Il est pratique et facile d'utiliser des modèles ANN pour des expériences numériques afin d'examiner les effets des proportions de chaque variable sur le mélange de béton.

Le béton à haute performance (HPC) est une nouvelle terminologie utilisée dans l'industrie de la construction en béton. En plus des trois ingrédients de base du béton conventionnel, à savoir le ciment Portland, les granulats fins et grossiers et l'eau, la fabrication du HPC doit incorporer des matériaux cimentaires supplémentaires, tels que des cendres volantes et du laitier de haut fourneau, et des adjuvants chimiques, tels que superplastifiant (1,2). Le béton à hautes performances est un matériau tellement complexe que la modélisation de son comportement est une tâche difficile.

Les réseaux de neurones sont une famille d'architectures massivement parallèles qui résolvent des problèmes difficiles via la coopération d'éléments de calcul hautement interconnectés mais simples. Fondamentalement, ils consistent en de nombreux éléments de calcul simples disposés en couches. L'intérêt pour les réseaux de neurones s'est rapidement développé ces dernières années. Une grande partie du succès des réseaux de neurones est due à des caractéristiques telles que le traitement non linéaire, le traitement parallèle, etc.

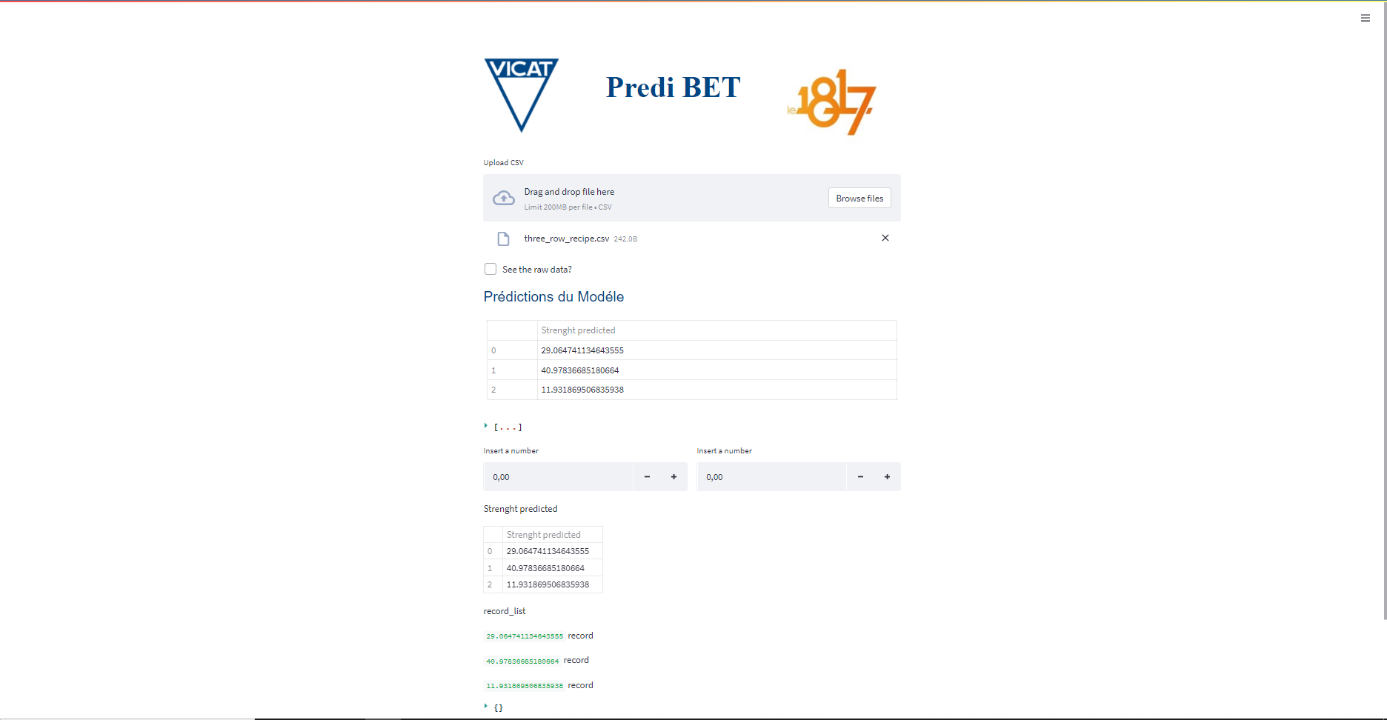
# Problématique :

Cet article vise à démontrer les possibilités d'adaptation des réseaux de neurones artificiels (ANN) pour prédire la résistance à la compression du béton à haute performance. Un ensemble de lots d'essai de HPC a été produit en laboratoire et a démontré des résultats expérimentaux satisfaisants.

# Application :

L’application permet de fournir des éléments qui compose le béton pour prédire sa résistance à la compression.

Pour tester la précision d'un réseau entraîné, le coefficient de détermination R2 est adopté. Le coefficient est une mesure de la mesure dans laquelle les variables considérées tiennent compte de la variable mesurée. Plus la valeur R2 est élevée, meilleure est la relation de prédiction !



L’application a été développé avec *Streamlit* qui transforme les scripts de données en applications Web partageables, le tout en pur Python.

# Mise en place du projet :

## Récupération des données :

Des données expérimentales provenant de 17 sources différentes ont été utilisées pour vérifier la fiabilité du modèle de résistance (1,13–28).

Les données d'essai ont été rassemblées pour le béton contenant du ciment plus des cendres volantes, du laitier de haut fourneau et du superplastifiant.

Une détermination a été faite pour s'assurer que ces mélanges étaient un groupe assez représentatif régissant tous les paramètres majeurs qui influencent la force de HPC et présentent les informations complètes requises pour une telle évaluation. Au total, environ 1000 échantillons de béton provenant des investigations ci-dessus ont été évalués.

Au cours de l'évaluation, certains échantillons de béton ont été supprimés des données en raison d'agrégats de plus grande taille (plus de 20 mm), de conditions de durcissement spéciales, etc. Environ 700 échantillons de béton fabriqués avec du ciment Portland ordinaire et durcis dans des conditions normales ont été évalués. Différentes études ont utilisé des spécimens de différentes tailles et formes. Tous ces types d'échantillons ont été convertis en cylindres de 15 cm conformément aux directives acceptées.

Les tableaux 1 et 2 présentent les détails généraux du béton évalué dans cette étude. La base de données contient souvent des inexactitudes inattendues, par exemple, la classe de cendres volantes n'est parfois pas signalée. La plus grande difficulté semble être liée à l'application des superplastifiants. Ils sont de fabricants différents, de compositions chimiques différentes, et sans précisions sur la teneur en solides de la suspension (8).

TABLE 1

Plages de composants d’ensembles de données.

|  |  |  |  |
| --- | --- | --- | --- |
| **Composant** | **Minimum (kg/m3)** | **Maximum (kg/m3)** | **Average (kg/m3 )** |
| Ciment Cendres volantes Laitier de haut fourneau Eau Superplastifiant Granulats grossiers Granulats fins |  |  |  |

Par conséquent, dans cette approche, la résistance à la compression du béton est fonction des huit caractéristiques d'entrée suivantes :

1. Ciment (kg/m3)

2. Cendres volantes (kg/m3)

3. Laitier de haut fourneau (kg/m3)

4. Eau (kg/m3)

5. Superplastifiant (kg/m3)

6. Granulats grossiers (kg/m3)

7. Granulats fins (kg/m3)

8. Âge du test (jours).

## Traitement des données :

Notre jeu de données de 1030 lignes, contenant chacun les huit composantes citées ci-dessus et la valeur de sortie (résistance à la compression), ont été scindée en deux ensembles de données :

* Un ensemble d’entrainement contenant 77% des données.
* Un ensemble de test contenant 33% des données.

Alternativement, tous les enregistrements ont été combinés et simplement mélangés à l'aide d'un échantillonnage aléatoire, en les divisant en groupes d'entraînement et de test. Les nombres d'exemples de formation et de test pour ces expériences sont répertoriés dans le tableau 3.

Les données sont très propres et n’ont pas nécessité beaucoup de pré-traitement mis à part une normalisation pour augmenter la rapidité d’exécution de notre algorithme.

## Modèle ANN :

**Données d’entrée**

Donnée tabulaire

**Sortie**

Linear

**Structure 6 neurones Dense**

ReLU

**Structure 8 neurones Dense**

ReLU

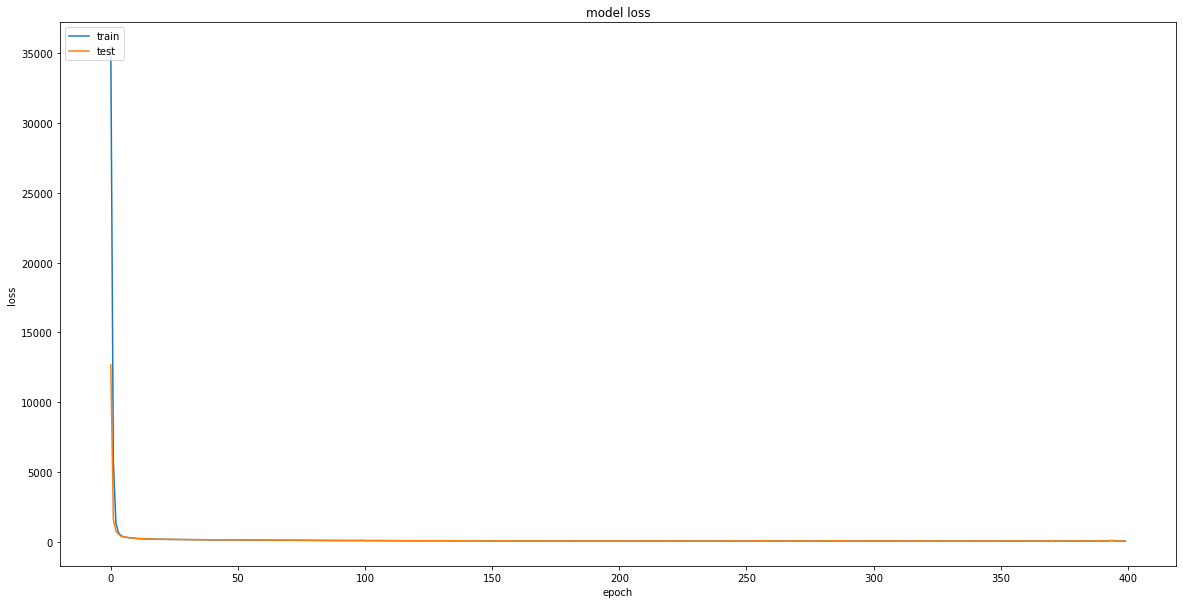
## Résultats et discussions :

Nous présentons ci-dessous les résultats de nos algorithmes pour la tâche de régression pour la prédiction de la résistance à la compression.

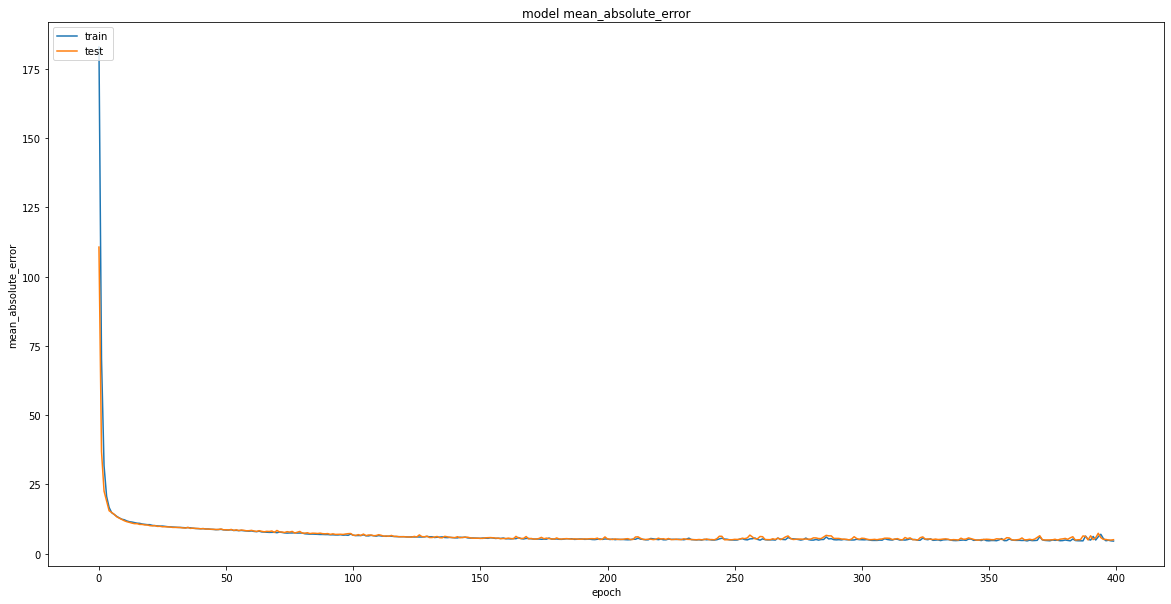
Dans chaque cas, nous représentons l’erreur moyenne absolue et le R2. Notre premier model CNN a pris toutes les données de notre dataset à savoir 1030 images et découpé de la manière suivante :

|  |  |
| --- | --- |
| Nombre de lignes : Train | Nombre de lignes : Test |
| 793 | 207 |

MAE :



LOSS :



|  |
| --- |
| Model : 1 |
| Mae: 4.56  R2: 0.83 |

# Test :

Divers tests fonctionnels ont été réalisé sur cette application notamment sur les fonctions de applications par exemple :

* Test sur le format des prédictions

Une image contenant texte

Description générée automatiquement

* Test sur la transformation de nos données

Une image contenant texte

Description générée automatiquement

* Test sur le format de notre modèle

Une image contenant texte

Description générée automatiquement

# Amélioration :

Pour évaluer notre modèle tout en continuant à ajuster ses hyperparamètres (tels que le nombre d’époques utilisées pour l’entraînement), nous pourrions diviser les données en un ensemble d’apprentissage et un ensemble de validation, comme nous l’avons fait dans les exemples précédents. Mais comme nous avons peu de points de données, l’ensemble de validation est très petit par exemple, environ 340 exemples). En conséquence, les scores de validation pourraient varier considérablement en fonction des points de données que nous avons choisis pour la validation et de ceux que nous avons choisis pour l’entraînement : les scores de validation pourraient présenter une forte variance suivant le nature du fractionnement de l’ensemble de validation.

**Cette variance importante nous empêcherait dévaluer notre modèle de façon fiable.**

La meilleur pratique dans cette situation est d’utiliser la validation croisée à K-fold (*K-fold cross validation*).

Cette pratique consiste à diviser les données disponibles en K partitions (typiquement K = 4 ou 5), à instancier K modèles identiques et à entraîner chacun sur des partitions K – 1 tout en évaluant sur la partition restante. Le score de validation pour le modèle utilisé est alors la moyenne des K scores de validation obtenus. En termes de code, c’est simple.

