IHDCM037 - Machine Learning
Travaux pratiques
Séance 1 : Introduction au Machine Learning

Remarque : Pour réaliser les travaux pratiques de ce cours, nous travaillerons avec le langage `Python` sur des `Jupyter Notebook`. Ceux-ci peuvent être utilisés en ouvrant la plateforme `Anaconda Navigator` et en lançant `Jupyter Notebook`.

Première partie

La première partie de cette séance est dédiée à la découverte d'un *dataset* qui sera manipulé dans la deuxième partie. Celui-ci est composé de données concernant le taux de diabète de plusieurs patients. Voici les différentes étapes à suivre :

1. Importez le *dataset* `diabetes` de la librairie `sklearn`. Pour cela, vous devez d'abord importer le module `datasets` de `sklearn`.
2. Visualisez le *dataset* puis utilisez l'argument `as_frame = True` pour obtenir une meilleure visualisation de celui-ci.
   - Combien de patients sont repris dans ce *dataset* ? .....
   - Quelles sont les *features* considérées pour mesurer la valeur de la *target* (mesure quantitative de la progression du diabète) ? .....
     .....

   Remarque : La librairie `Pandas` peut également être utilisées pour obtenir une représentation plus claire d'un *dataset*.
3. Sauvegardez la partie *data* du *dataset* dans un premier tableau et la partie *target* dans un deuxième.
4. À partir du tableau contenant la partie *data*, extrayez la colonne qui correspond au BMI des patients.
5. Affichez un nuage de points représentant le taux de progression du diabète de chaque patient en fonction de leur BMI.

Deuxième partie

La deuxième partie de cette séance est consacrée à la création d'un tout premier modèle de Machine Learning à partir des données contenues dans le *dataset* `diabetes` de la librairie `sklearn`. Le modèle utilisé dans cette partie est un modèle de régression linéaire. Voici les différentes étapes à suivre :

1. Importez le *dataset* `diabetes` de la librairie `sklearn`.
2. Sauvegardez la partie *data* du *dataset* dans un premier tableau et la partie *target* dans un deuxième.
3. À partir du tableau contenant la partie *data*, extrayez la colonne qui correspond au BMI des patients.

   Remarque : Pour pouvoir entraîner un modèle de Machine Learning sur des données, il faut qu'elles aillent une certaine structure. Lorsqu'une seule *feature* est considérée, il faut utiliser la fonction `reshape` pour spécifier que les données ne forment qu'un seul vecteur colonne (vous pouvez utiliser la fonction `.reshape(-1,1)`).
4. À partir de chaque tableau créé (un pour la partie *data* et un pour la partie *target*), formez un ensemble de test et un ensemble d'entraînement. L'ensemble d'entraînement doit contenir 80% des patients.
   - Quels sont les moyens d'y arriver (au moins 2) ? .....

     .....
5. Entraînez un modèle de régression linéaire sur l'ensemble d'entraînement grâce à la fonction `LinearRegression()` du module `linear_model` de la librairie `sklearn`.
   - Quelle est la forme générale d'une droite de régression ? .....
   - Pour cet exercice, quelle est l'équation de la droite de régression ? .....
6. Calculez l'erreur empirique du modèle de régression sur les données d'entraînement.
   - Quelle est la forme générale de l'erreur empirique d'un modèle de régression ?

     .....
   - Quelle fonction de la librairie `sklearn` pouvez-vous utiliser pour calculer cette erreur ?

     .....
   - Quelle est la valeur de cette erreur ? .....
7. Calculez l'erreur empirique du modèle de régression sur les données de test.
   - Quelle est la valeur de cette erreur ? .....
   - Que pouvez-vous remarquer par rapport à l'erreur empirique calculée sur les données d'entraînement ? .....
   - Comment pouvez-vous expliquer cette dernière constatation ? .....

     .....
8. Vérifiez visuellement que le modèle créé colle bien aux données d'entraînement et aux données de test. Pour cela, vous pouvez afficher le nuage de points représentant les données d'entraînement ainsi que celui représentant les données de test sur un même graphique, puis y ajouter la droite de régression (grâce à son équation au point 5).

Lors de l'exercice précédent, uniquement la *feature* concernant le BMI des patients a été considérée et l'erreur empirique obtenue sur l'ensemble de test est assez élevée. Pour produire de meilleurs résultats, un modèle de Machine Learning a donc besoin de plus d'informations. Dès lors, entraînez un deuxième modèle de régression en considérant l'ensemble des *features* fournies par le *dataset* `diabetes`.

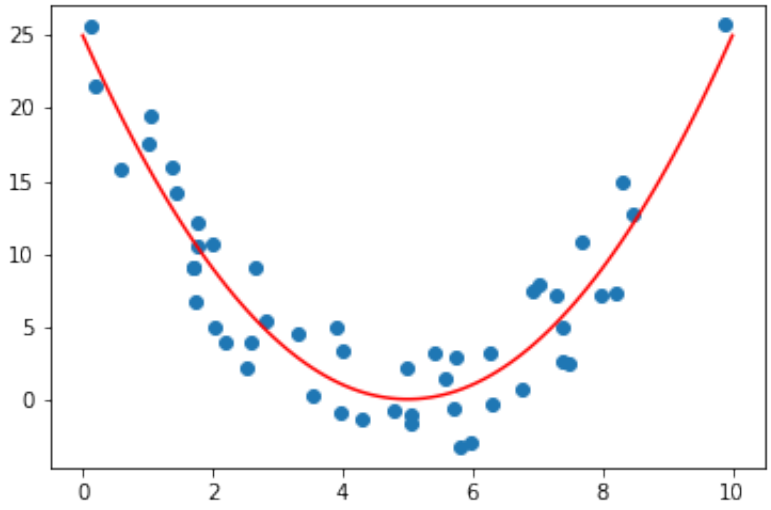
- Quelle est l'erreur empirique de ce modèle de régression sur l'ensemble des données de test ?

  .....

- En affichant les coefficients de la droite de régression, que pouvez-vous conclure ? .....

  .....

Remarque : Un modèle de régression linéaire est approprié pour approximer des données qui ont un comportement linéaire mais ce n'est pas toujours le cas. Si les données suivent un autre comportement, des modèles polynomiaux de plus haut degrés peuvent être plus adaptés. Par exemple, les données affichées sur le graphique suivant sont bien approximée par un modèle polynomial du second degré.



Troisième partie

La troisième partie de cette séance concerne les phénomènes *d'overfitting* et *d'underfitting*.

Voici le nuage de points d'un certain jeu de données. Tracez un modèle de régression (linéaire ou polynomial) qui *underfit* les données, un qui les *overfit*, puis un modèle qui les généralise bien.

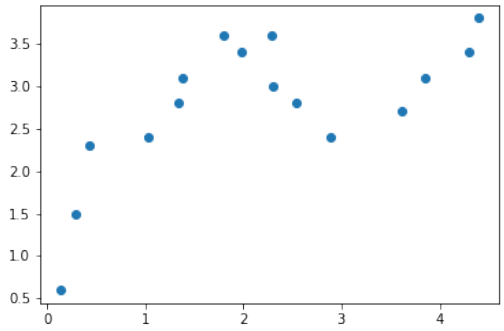


FIGURE 1 – Modèle qui *underfit* les données.

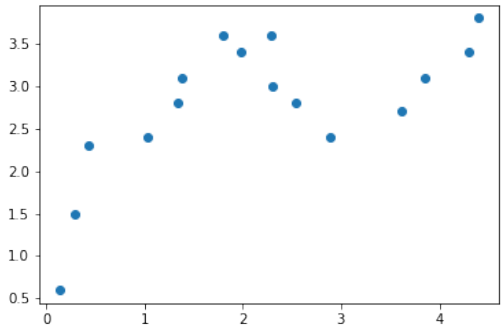


FIGURE 2 – Modèle qui *overfit* les données.

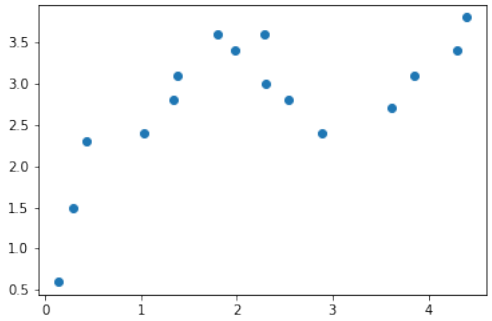


FIGURE 3 – Modèle qui généralise bien les données.

Imaginez calculer les erreurs empiriques de ces modèles par rapport aux données d'entraînement et par rapport aux données de test. Que pouvez-vous attendre ? (l'erreur est élevée ou l'erreur est faible)

	Erreur empirique sur l'ensemble d'entrainement	Erreur empirique sur l'ensemble de test
Underfitting		
Overfitting		
Bonne généralisation		