**PREWORK**

**SESIÓN 02**

**Introducción:**

En esta sesión veremos algoritmos, herramientas y librerías que nos permitirán evaluar nuestros algoritmos de Machine Learning que veremos en el resto de las sesiones.

**Pre-requisitos técnicos:**

Para esta sesión utilizaremos las siguientes herramientas (además de las que utilizamos en la sesión 01):

1. Scikit Learn.

Scikit Learn nos servirá para manejar algoritmos de separación de clases, evaluación de matrices de confusión, y muchas otras cosas para que no tengamos que hacer los algoritmos desde cero. Es una herramienta muy útil que nos simplificará muchos procesos, sin llegar a resolvernos todo.

Lo instalas con ***pip3 install sklearn***

**Pre-requisitos de conocimientos:**

En esta sesión nos enfocaremos en crear bases sólidas sobre como entrenar, dividir tu dataset, preparar entrenamientos, y realizar métricas de desempeño de tus algoritmos de Machine Learning, para ello necesitaremos que tengas en mente varios conceptos:

* **Overfitting**: También conocido como sobre-entrenamiento. Ocurre cuando un algoritmo de Machine Learning ha sido expuesto numerosamente a los mismos datos una y otra vez durante muchas iteraciones. Cuando esto ocurre, un algoritmo de Machine Learning no es capaz de generalizar y su desempeño baja considerablemente.
* **Underfitting**: También conocido como sub-entrenamiento, ocurre cuando un algoritmo de Machine Learning tiene muy poco entrenamiento o es demasiado simple para resolver un problema. Por lo general tratamos de entrenar un algoritmo de Machine learning para estar en el punto óptimo entre Overfitting y Underfitting.
* **Falacia de Cherry Picking**: Una falacia de Cherry Picking ocurre cuando previamente seleccionas datos con la finalidad de probar un punto, mientras que ignoras o retiras aquellos datos que contradicen el punto que quieres defender. Un ejemplo de Cherry Picking ocurre con el movimiento anti-vacunas: Se seleccionan solamente aquellos casos que consideran peligrosos, pero ignoran todos los casos que no lo son.
* **Validación Cruzada:** La validación cruzada es un mecanismo que hace que todos los datos “roten” entre entrenamiento y prueba, de tal manera que no existan casos de *“Cherry Picking”*
* **Matriz de confusión:** Una matriz de confusión es una matriz cuadrada de N x N donde evalúas N clases. Los valores de la diagonal principal de la matriz de confusión indican que tan bueno ha sido tu algoritmo, versus los valores que se encuentran fuera de la diagonal principal. Es uno de los mecanismos más utilizados para evaluar algoritmos de machine learning.

**Recursos adicionales:**

Te recomiendo echar un vistazo a los siguientes artículos:

* <https://machinelearningmastery.com/overfitting-and-underfitting-with-machine-learning-algorithms/>
* <https://towardsdatascience.com/train-validation-and-test-sets-72cb40cba9e7>
* <https://www.researchgate.net/post/How_many_ways_exist_to_separate_train_and_test_data_sets_in_ANN_which_one_is_the_best>
* <https://stats.stackexchange.com/questions/19048/what-is-the-difference-between-test-set-and-validation-set>
* <https://en.wikipedia.org/wiki/Cross-validation_(statistics)>
* <https://en.wikipedia.org/wiki/Confusion_matrix>

**Quiz:**

**¿Qué diferencia hay entre overfitting y underfitting?**

1. Overfitting hace que un algoritmo aprenda menos, y underfitting que aprenda más.
2. **Overfitting hace que un algoritmo no generalice, y underfitting que no aprenda.**
3. Overfitting hace que un algoritmo se entrene, y underfitting hace que olvide.
4. Overfitting hace que un algoritmo se reinicie y underfitting hace que se preserve.

**¿Cuál de estos enunciados es una falacia de Cherry Picking?**

1. **Mi hermano y mi primo se cayeron de la bicicleta. Las bicicletas son peligrosas.**
2. Si no tengo la razón, la obtendré por la fuerza.
3. El experto dijo que el programa se hace de esta manera, debe tener razón.
4. Si no te gustan los tacos no eres un verdadero mexicano.

**¿Qué tienes que observar en la matriz de confusión?**

1. Los valores de toda la matriz.
2. **Los valores que se encuentran en la diagonal principal**
3. Los valores que se encuentran en las diagonales.
4. Que la matriz no tenga valores negativos.

**¿Cuál de estos evita el Cherry Picking?**

1. Evitar el Overfitting
2. Evitar el Underfitting
3. Usar una matriz de confusión
4. **Utilizar validación cruzada**