Machine Learning Code

Soft Computing

Autor: Manuel Felipe Valencia Ceballos

*IS&C, Universidad Tecnológica de Pereira, Pereira, Colombia*

Correo-e: felipe.valencia@utp.edu.co

***Resumen*— Este documento presenta una breve explicación de conceptos básico de una librería implementada en Python y el desarrollo de un pequeño y básico ejercicio de Machine Learning. El objetivo del documento es dar a entender de una manera más explicita lo que se quiere dar a entender con el código explicado en la clase.**

***Palabras clave—* Python, Machine Learning, Código.**

***Abstract*— This document presents a brief explanation of the basic concepts of a library implemented in Python and the development of a small and basic Machine Learning exercise. The objective of the document is to give explicitly understanding about what was understand with the core explained in the class**

***Key Word*— Python, Machine Learning, Code.**

1. INTRODUCCIÓN

El Aprendizaje Automático consiste en una disciplina de las ciencias informáticas, relacionada con el desarrollo de la Inteligencia Artificial, y que sirve, como ya se ha dicho, para crear sistemas que pueden aprender por sí solos.

Es una tecnología que permite hacer automáticas una serie de operaciones con el fin de reducir la necesidad de que intervengan los seres humanos. Esto puede suponer una gran ventaja a la hora de controlar una ingente cantidad de información de un modo mucho más efectivo.

En los siguientes apartados se presenta un resumen de dichas tendencias.

En los siguientes apartados se dará explicación de un código de machine Learning

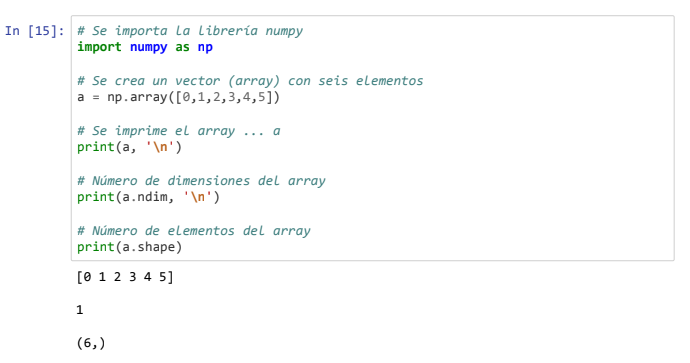


Figura 1.

Principalmente lo que se hizo fue agregar la librería de NumPy por que el trabajo que se va a realizar depende de ella. NumPy es una extensión de Python, que le agrega mayor soporte para vectores y matrices, constituyendo una biblioteca de funciones matemáticas de alto nivel para operar con esos vectores o matrices. Para empezar a entender como implementar diferentes aplicaciones de la librería se inició con cosas simples para ir entendiendo poco a poco, acá o que se hizo fue crear un vector de longitud 6, y se mostró como podíamos imprimir el vector, cuál es su dimensión y cuantos elementos contiene el vector.

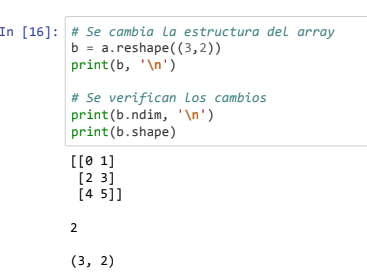


Figura 2.

Se cambia la estructura del vector con un “.reshape”, una nueva función convirtiéndolo en una matriz con la misma cantidad de elementos creados inicialmente con dos columnas y tres filas.

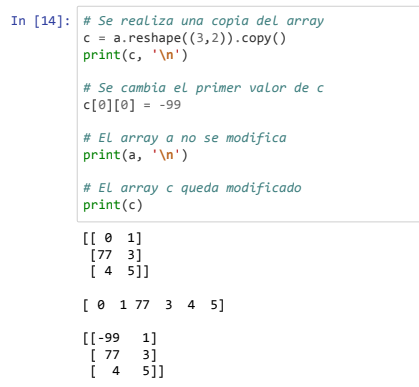


Figura 3.

Para realizar una copia se puede usar la función “.copy()”. Y como prueba se muestra que quedan los mismos datos de a en la variable c, el cual no se permite modificar por medio de misma.

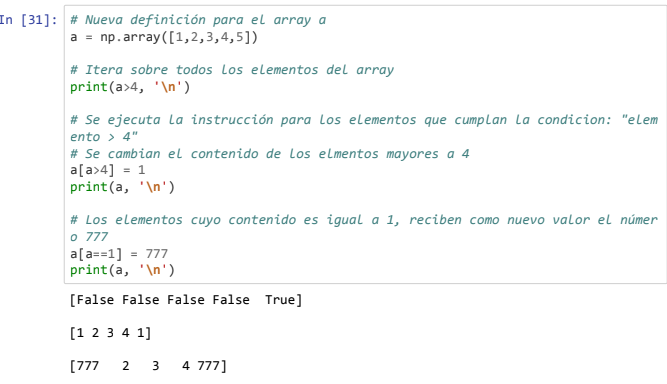


Figura 4.

Para este segmento de código se hace una nueva definición para el vector a, donde se puede verificar los elementos que sean mayor de un numero establecido, y con cambiar los valores dentro del vector para valores mayores al mismo.

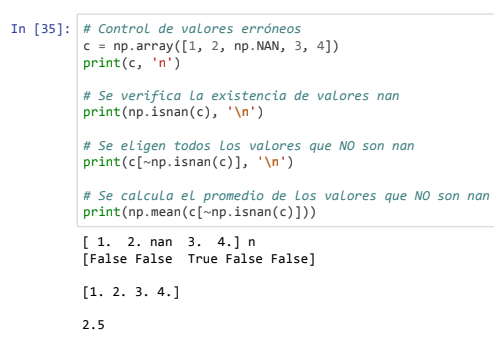


Figura 5.

Para verificar el control de errores se crea un nuevo vector donde unos de los elementos contengan un valor invalido (nan).

Se imprime para mostrar que efectivamente muestra el dato basura. Se hace una verificación de que existen valores “nan” y por medio de una función de NumPy se eligen todos los valores que no son nan.

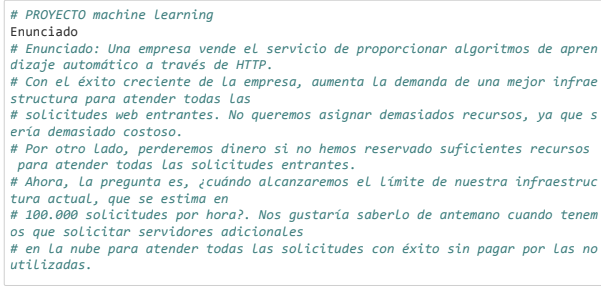


Figura 6.

Después de una pequeña introducción en cuanto al manejo de vectores y matrices, se hace un pequeño ejemplo. Para ello se va a desarrollar un pequeño programa de machine Learning atendiendo al enunciando de la figura 6.



Figura 7.

Se lee el archivo de “web\_traffic.tsv” que contiene todos los datos a analizar, el numero de horas y el numero de tareas ejecutadas. Se verifica por medio del “print” que la cantidad de datos analizados son 743.

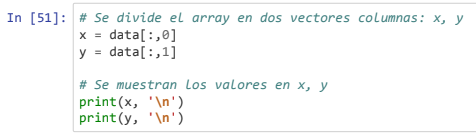


Figura 8.

Se divide el vector en dos vectores columna, los datos de la columna cero ponen en un vector x, y los datos de la columna una se ponen en otro vector y. Se transformó los datos que estaban en un vector, en dos vectores x, y.

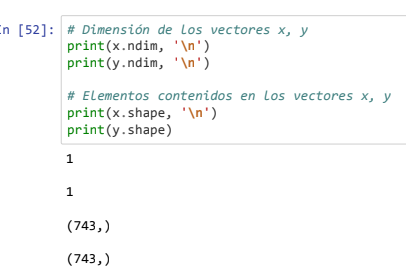


Figura 9.

Se verifica que ambos vectores (x, y) tiene la dimensión uno, que equivale a la de un vector, y que la cantidad de datos es la misma que habían en un principio.

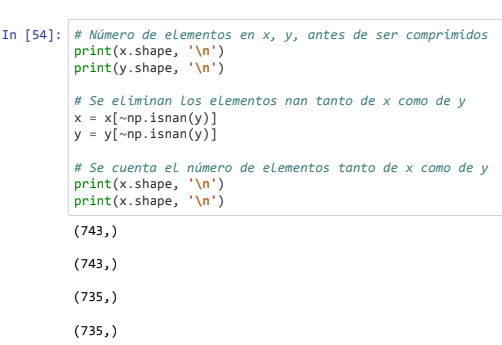


Figura 10.

Como es probable que el vector y contenta elementos basura, entonces se procede a eliminar la cantidad de elementos basura que contiene. Se recalcula lo vectores respecto a y ya que este es el vector que contiene los elementos basura. Se eliminan 8 elementos basura de los 743 que habían en un inicio.

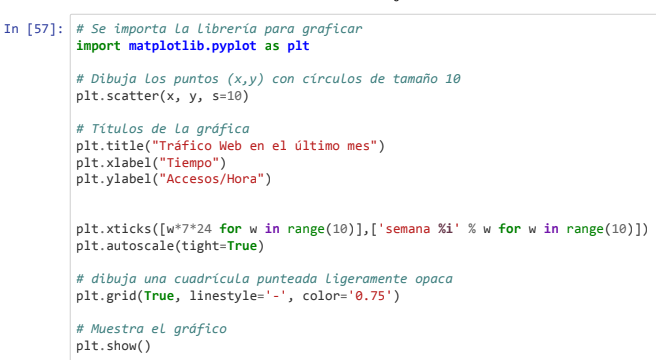
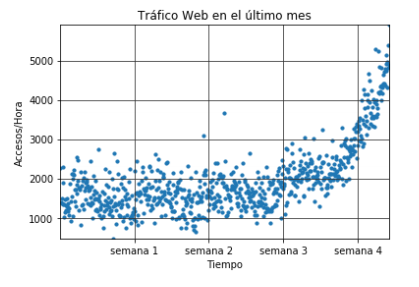


Figura 10.

Se importa una nueva librería, esta librería es equivalente a la que tiene Matlab. Se va a dibujar los puntos (x, y), con círculos de tamaño de 10 pixeles. Como se va a hacer una gráfica, se agregan las etiquetas que va a tener la gráfica. Se establece un rango, que este va a ser de 10 en 10. Para que la grafica no quede muy densa la información Se dibuja una cuadricula con un punto de opacidad y que estilo de línea se utiliza.



Esta grafica representa el grafico en un mes dependiendo las horas. Con esta grafica se puede analizar si el servidor aguanta la cantidad de datos o cuando se tiene que hacer un cambio de servidor

REFERENCIAS

Referencias en la Web:

[1]

<https://es.wikipedia.org/wiki/NumPy>

[2]

<https://www.apd.es/que-es-machine-learning/#:~:text=Machine%20Learning%20o%20Aprendizaje%20autom%C3%A1tico,de%20datos%20en%20su%20sistema.>