

Universidad de Guanajuato

División de Ciencias Económico-Administrativas

Lic En sistemas de información administrativas

Aprendizaje maquina

Profesor: Andrés Espinal Jiménez

Alumno: Braulio Eugenio Camacho Loya

Tema: Reporte clasificación supervisada usando Python y scikirlearn

Contenido

[Introducción 4](#_Toc36747885)

[Python 4](#_Toc36747886)

[Scikit learn 4](#_Toc36747887)

[Numpy 4](#_Toc36747888)

[Datasets 5](#_Toc36747889)

[Irisplant 5](#_Toc36747890)

[Generado 7](#_Toc36747891)

[Dígitos 8](#_Toc36747892)

[Clasificadores 8](#_Toc36747893)

[Perceptrón paralelo 9](#_Toc36747894)

[Máquinas de soporte vectorial 10](#_Toc36747895)

[KNN(K- nearest neighbor) 12](#_Toc36747896)

[validación cruzada 12](#_Toc36747897)

[Experimentos 15](#_Toc36747898)

[Irisplant 15](#_Toc36747899)

[Pruebas Irisplant – SVM – sin PCA 15](#_Toc36747900)

[Pruebas Irisplant – SVM – con PCA 16](#_Toc36747901)

[Pruebas Irisplant – KNN – sin PCA 16](#_Toc36747902)

[Pruebas Irisplant – KNN – con PCA 17](#_Toc36747903)

[Pruebas Irisplant – Parallel perceptron – sin PCA 17](#_Toc36747904)

[Pruebas Irisplant – Parallel perceptron – con PCA 18](#_Toc36747905)

[Generado 18](#_Toc36747906)

[Pruebas Generado – SVM – sin PCA 18](#_Toc36747907)

[Pruebas Generado – SVM – con PCA 19](#_Toc36747908)

[Pruebas Generado – KNN – sin PCA 20](#_Toc36747909)

[Pruebas Generado – KNN – con PCA 20](#_Toc36747910)

[Pruebas Generado – Parallel perceptron – sin PCA 21](#_Toc36747911)

[Pruebas Generado – Parallel perceptron – con PCA 21](#_Toc36747912)

[Dígitos 22](#_Toc36747913)

[Pruebas dígitos – SVM – sin PCA 22](#_Toc36747914)

[Pruebas dígitos – SVM – con PCA 22](#_Toc36747915)

[Pruebas dígitos – KNN – sin PCA 23](#_Toc36747916)

[Pruebas dígitos – KNN – con PCA 23](#_Toc36747917)

[Pruebas dígitos – Parallel perceptron – sin PCA 24](#_Toc36747918)

[Pruebas dígitos – Parallel perceptron – con PCA 24](#_Toc36747919)

[Titanic 25](#_Toc36747920)

[Pruebas titanic – SVM – sin PCA 25](#_Toc36747921)

[Pruebas titanic – SVM – con PCA 25](#_Toc36747922)

[Pruebas titanic – KNN – sin PCA 26](#_Toc36747923)

[Pruebas titanic – KNN – con PCA 27](#_Toc36747924)

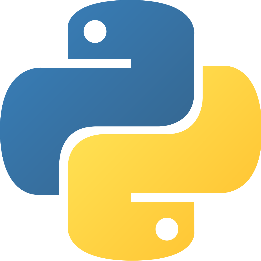
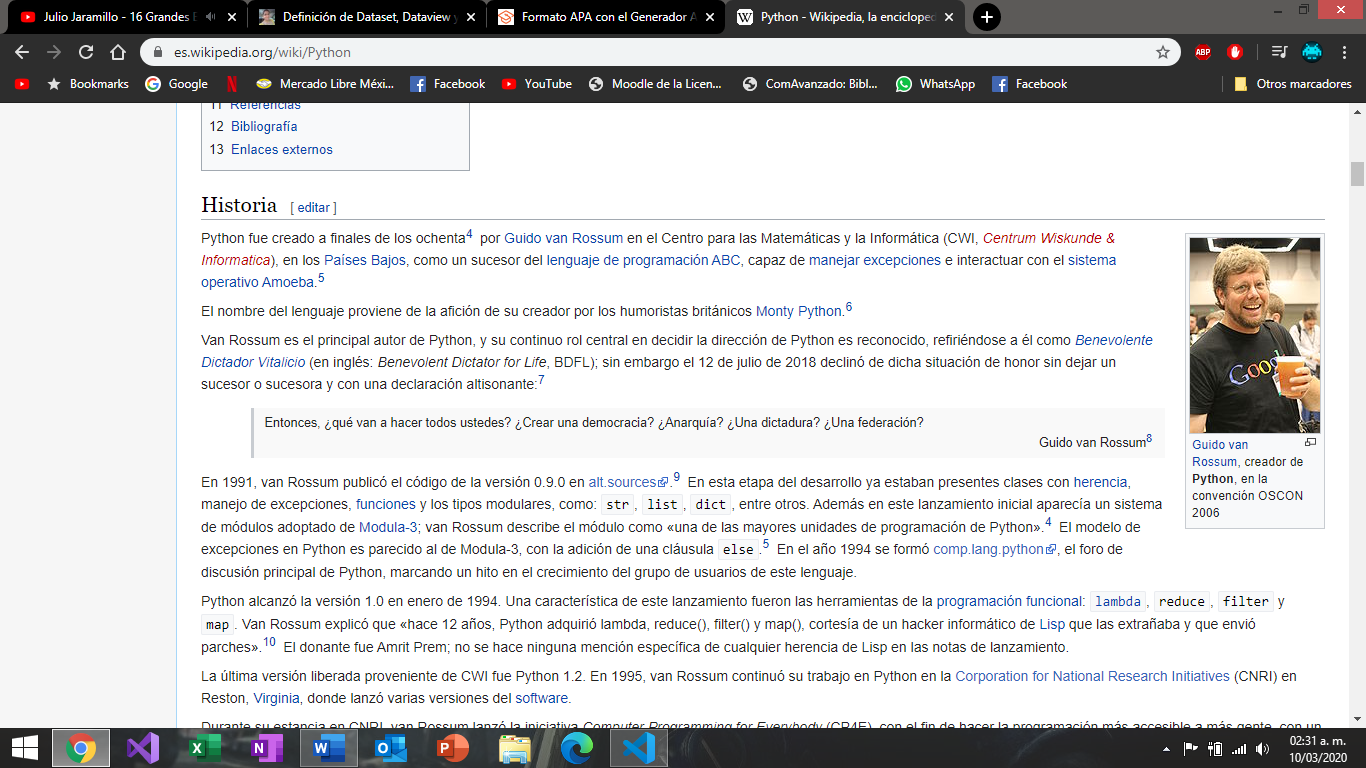
[Pruebas titanic – Parallel perceptron – sin PCA 27](#_Toc36747925)

[Pruebas titanic – Parallel perceptron – con PCA 27](#_Toc36747926)

[Referencias 28](#_Toc36747927)

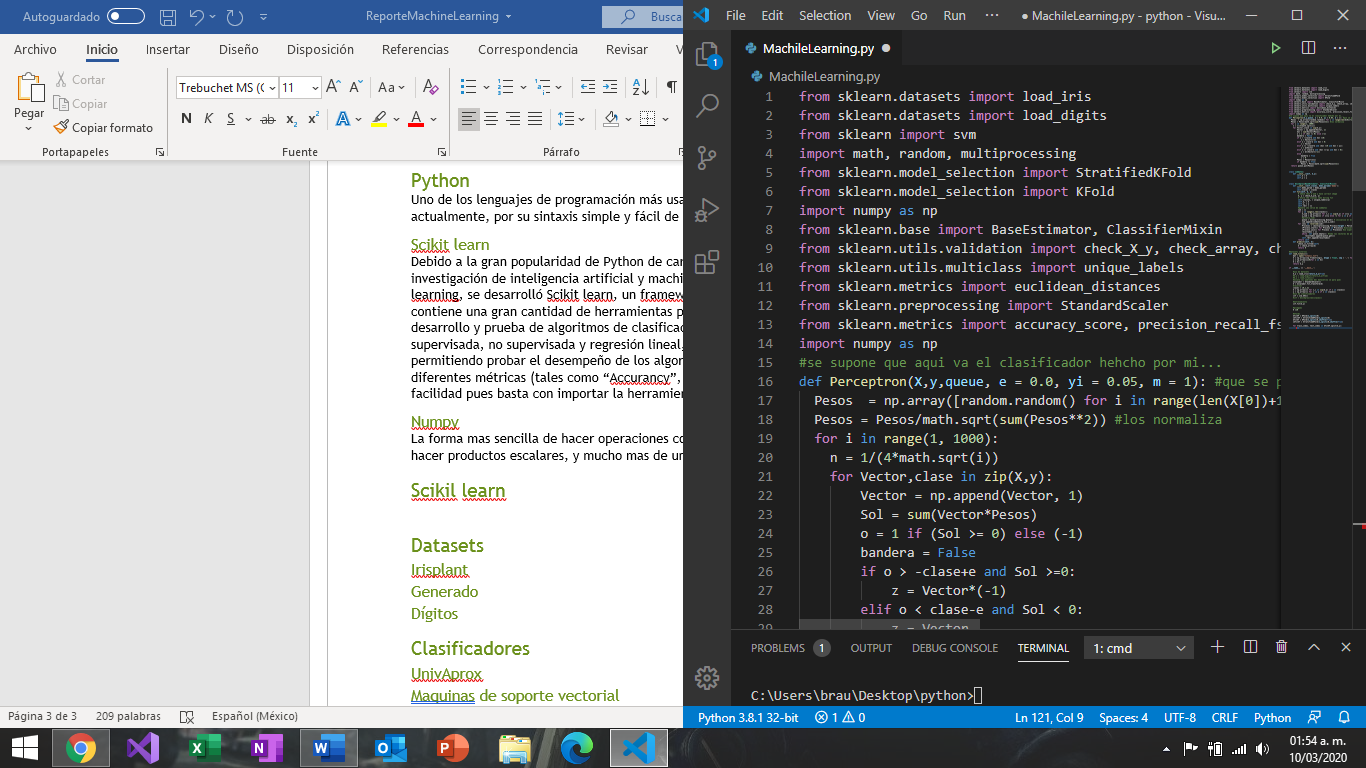
# Introducción

# Python

Uno de los lenguajes de programación más usados actualmente, por su sintaxis simple y fácil de aprender, a pesar de esto, no esta dentro de mis gustos, pues por lo simple que es, a veces puede resultar ambiguo y confuso, sobre todo de cara al paradigma orientado a objetos pues comparándolo con otro lenguaje, como por ejemplo “JAVA”, el cual a pesar de que cuenta con una sintaxis más rebuscada, lo cierto es que posee una estructura orientada a objetos mucho mas robusta y clara, pues es mucho mas estricta, lo que la hace mas estable., a pesar de esto “Python”, junto con sus frameworks “scikit learn” y numpy, serán herramientas a usar para este experimento.

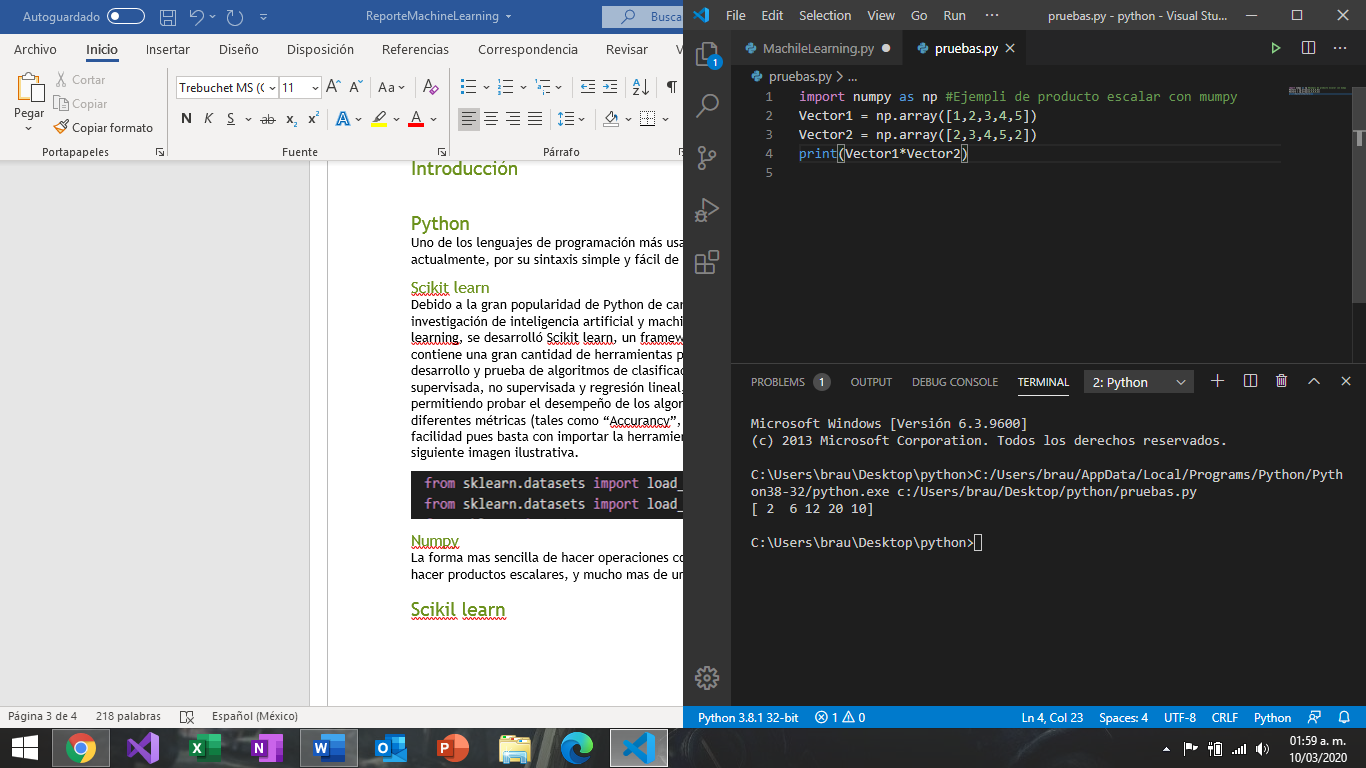
## Scikit learn

Debido a la gran popularidad de Python de cara a la investigación de inteligencia artificial y machine learning, se desarrolló Scikit learn, un framework que contiene una gran cantidad de herramientas para el desarrollo y prueba de algoritmos de clasificación supervisada, no supervisada y regresión lineal, permitiendo probar el desempeño de los algoritmos con diferentes datasets (incluidos) y diferentes métricas (tales como “Accurancy”, “Presision” y “Recall”) con mucha facilidad pues basta con importar la herramienta necesaria, y lo podemos ver en la siguiente imagen ilustrativa.

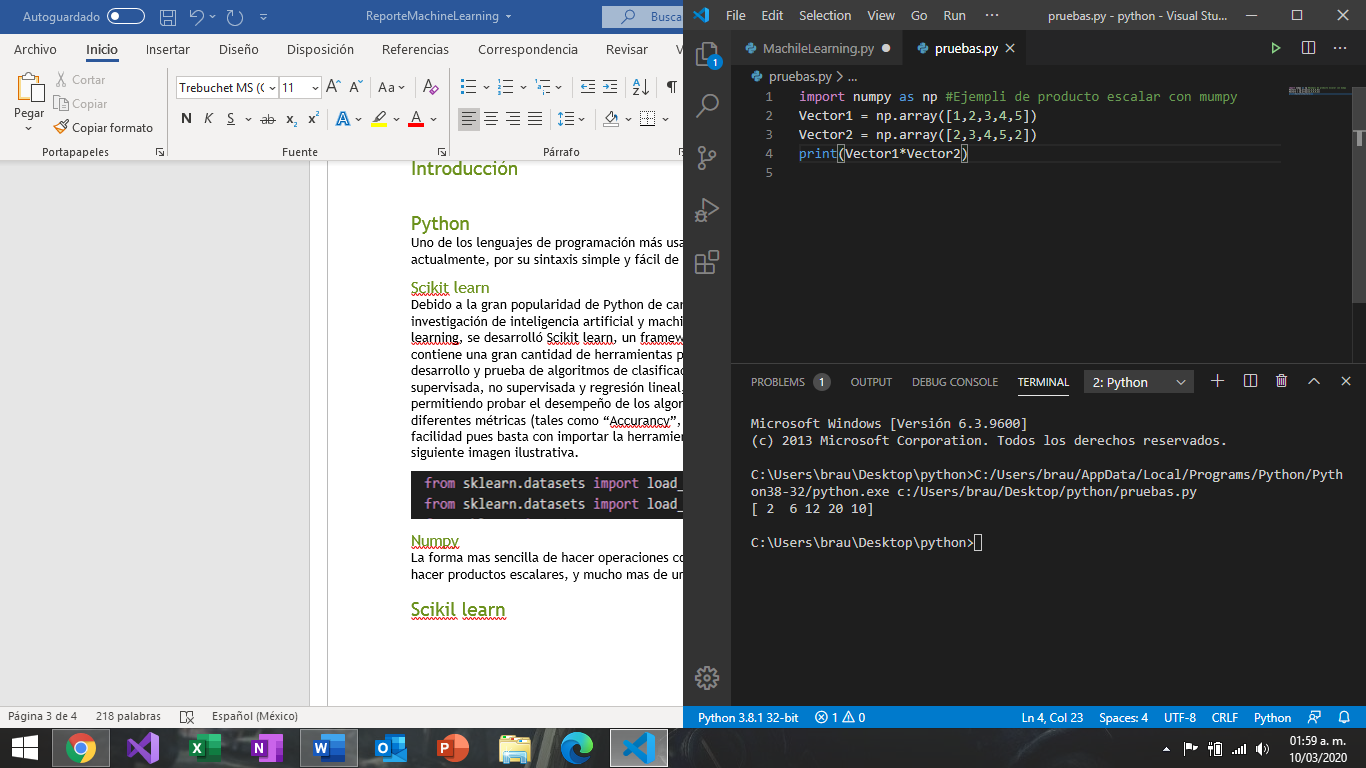


## Numpy

La forma mas sencilla de hacer operaciones con vectores es usando numpy, pues permite hacer productos escalares, y mucho mas de una forma muy sencilla.



El anterior código nos da como resultado:



# Datasets

Todo experimento de inteligencia artificial y machine learning, requiere de algún tipo de conocimiento previo acumulado, y algo muy común es usar datasets(“una representación de datos residente en memoria que proporciona una modelo de programación relacional coherente independientemente del origen de datos que contiene”, 2011).

Los datasets se componen principalmente de 2 partes:

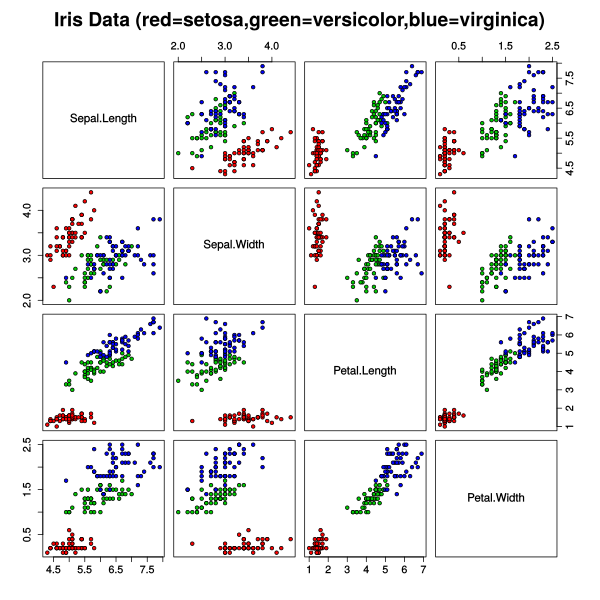
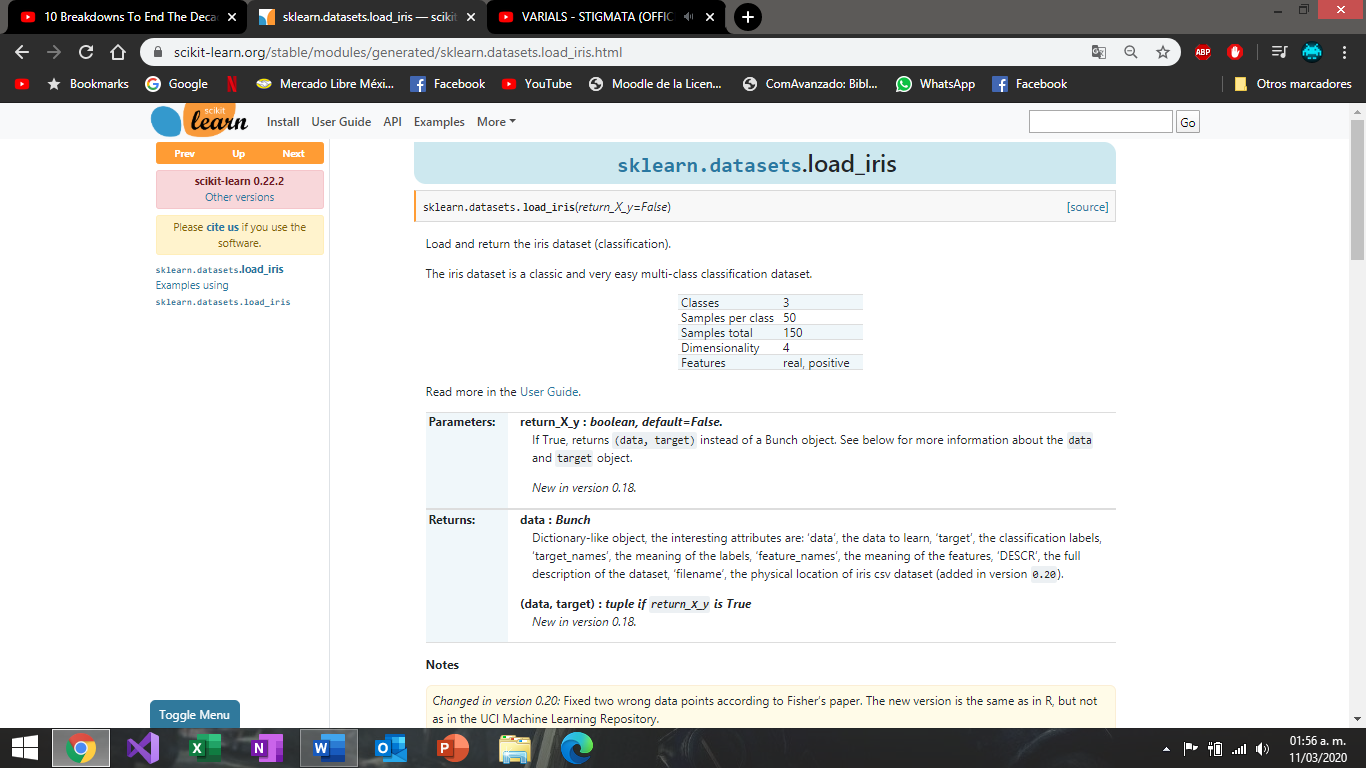
* **Clase:**

Es el indicador de a que conjunto de datos pertenece, y se usa tanto para entrenar los algoritmos, como punto de referencia de los algoritmos para sabes si su predicción es correcta o no.

* **Atributos:**

Es el conjunto de características que describen a la clase.

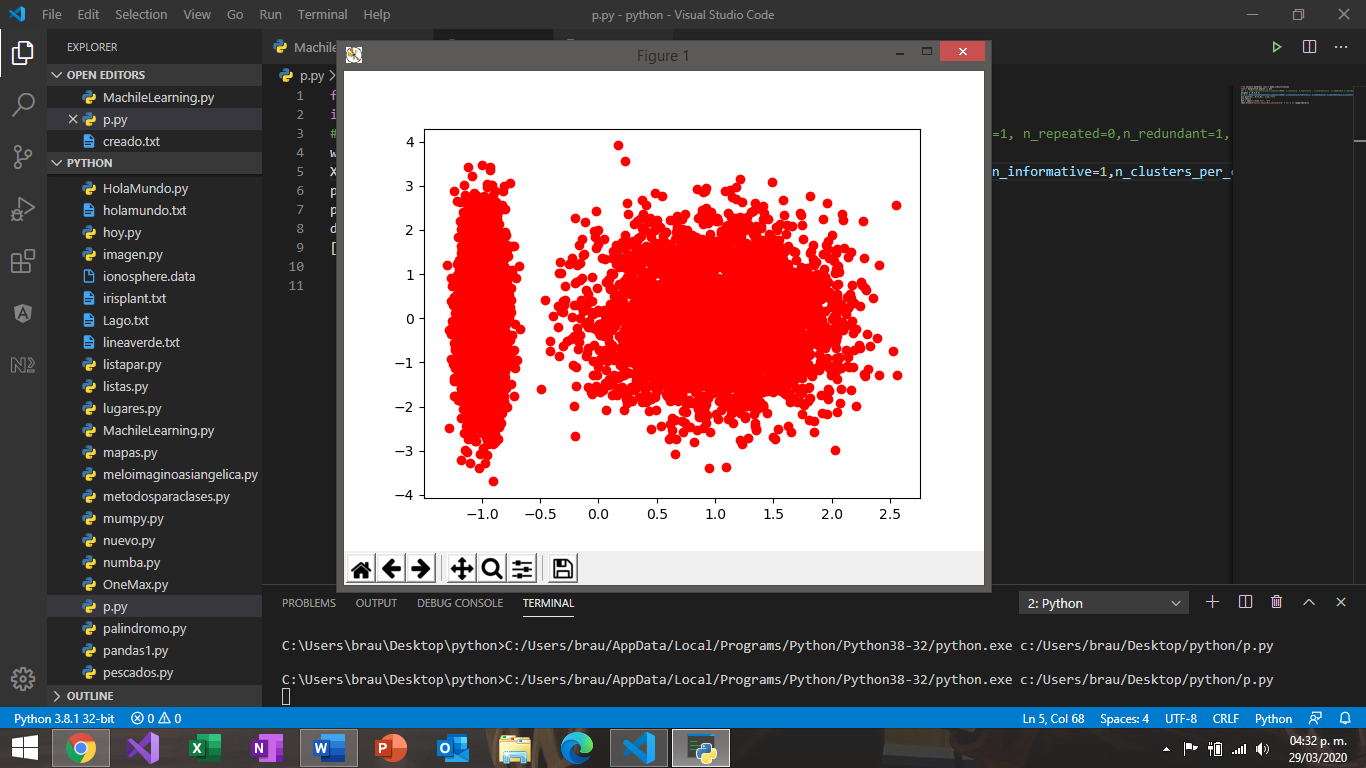
## Irisplant

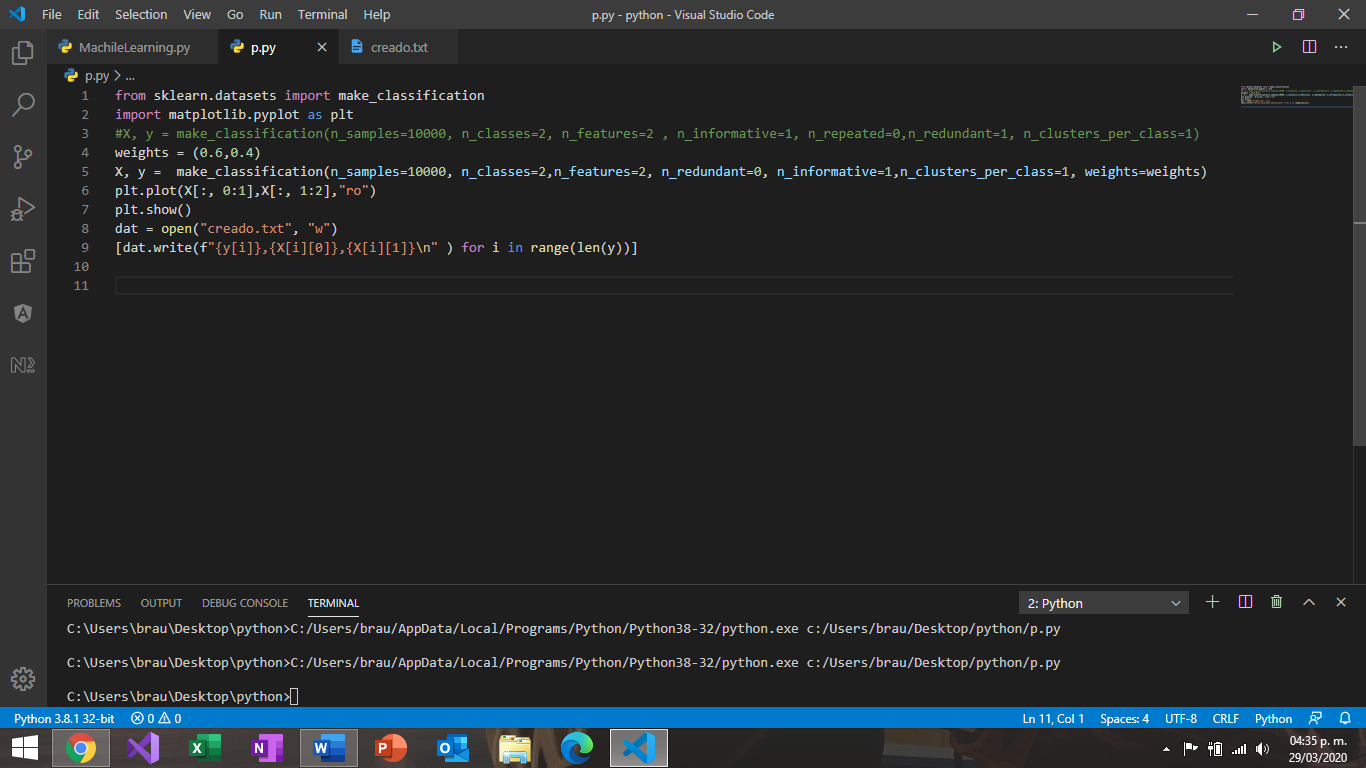
Para experimentación con estimadores, el dataset más usado es “irisplant”, pues es multiclase (que contiene más de 2 clases), dos de las cuales son separables linealmente. 



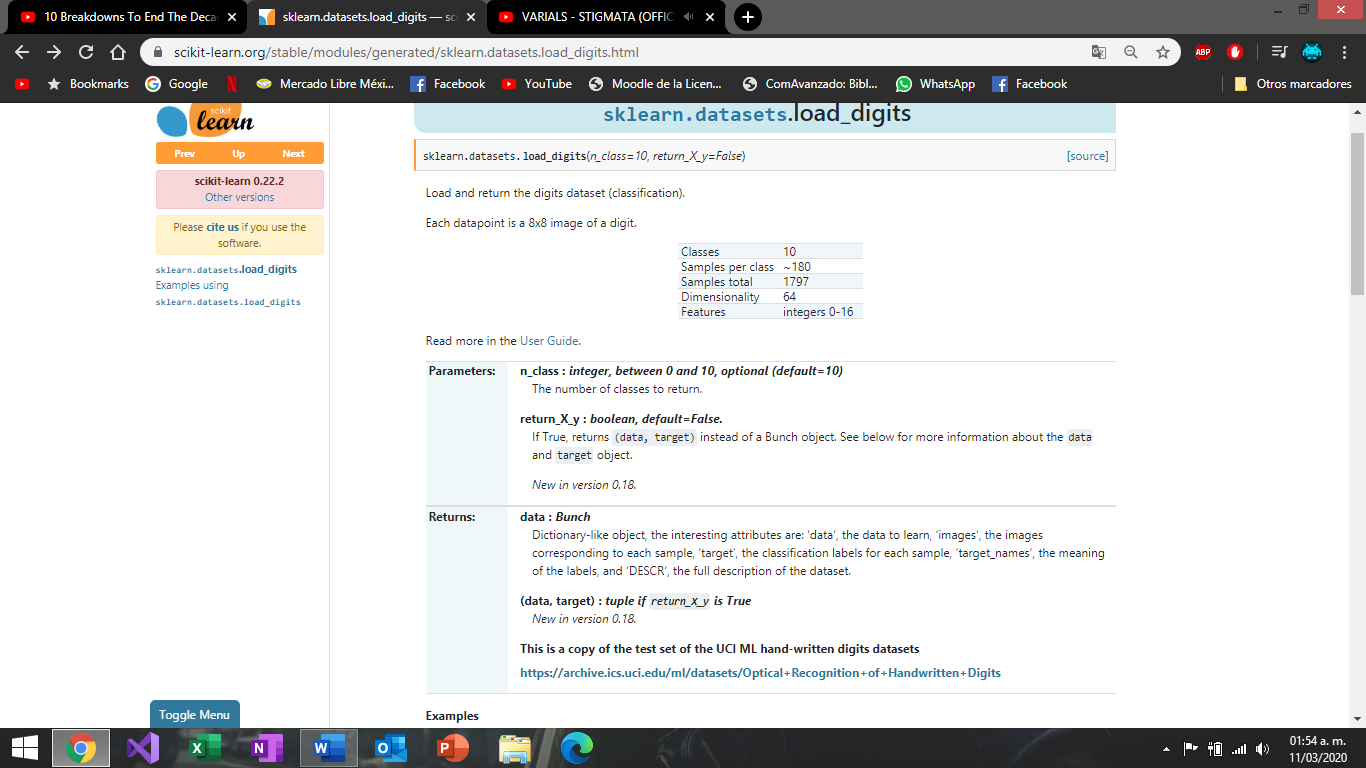
## Generado

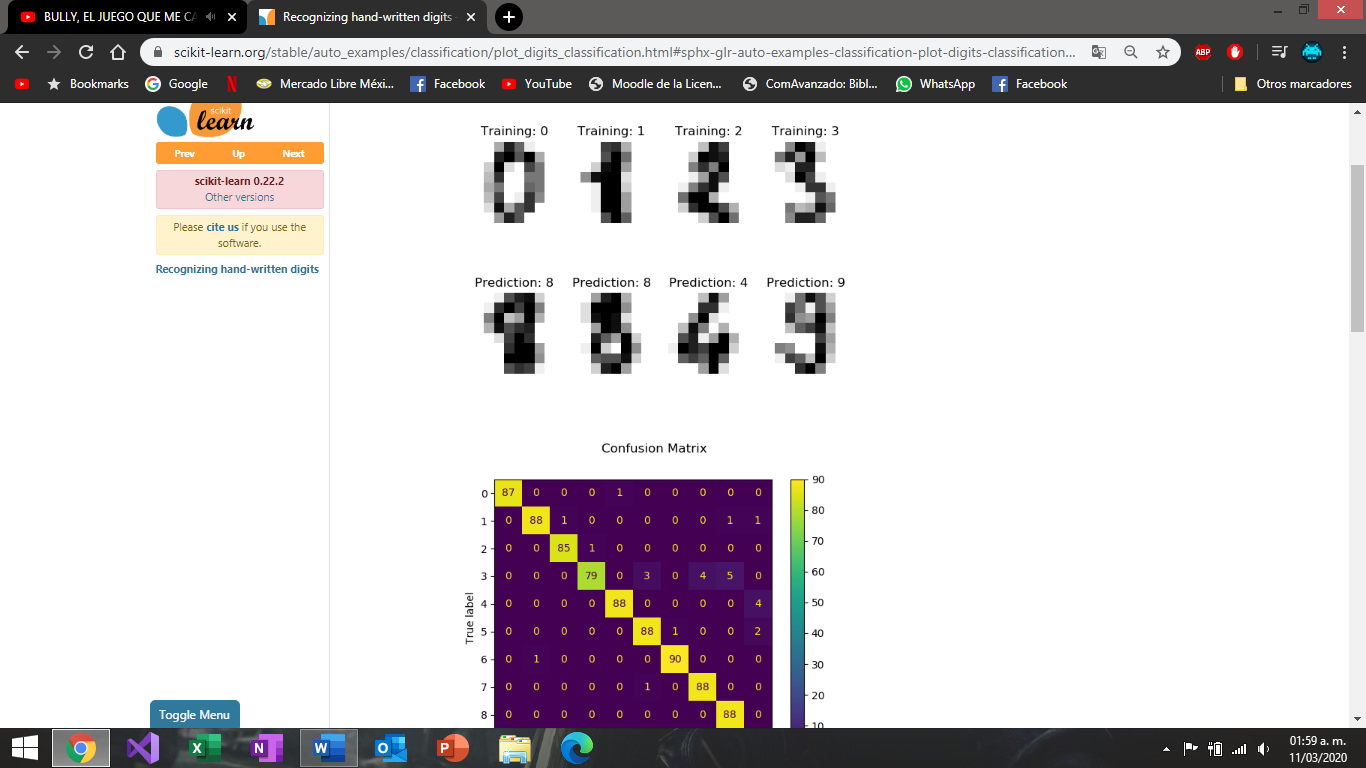
Scikit Learn dentro de todas las herramientas que contiene, nos permite crear un dataset, con sus atributos y clases, que sea linealmente separable o no, o que tanto lo sea o no, e incluso para regresión o clusterización(clasificación no supervisada), por lo que para poner a prueba los algoritmos, se va a generar un dataset con 2 clases, 2 atributos y 10000 atrubutos no balanceados 60/40





## Dígitos





# Clasificadores

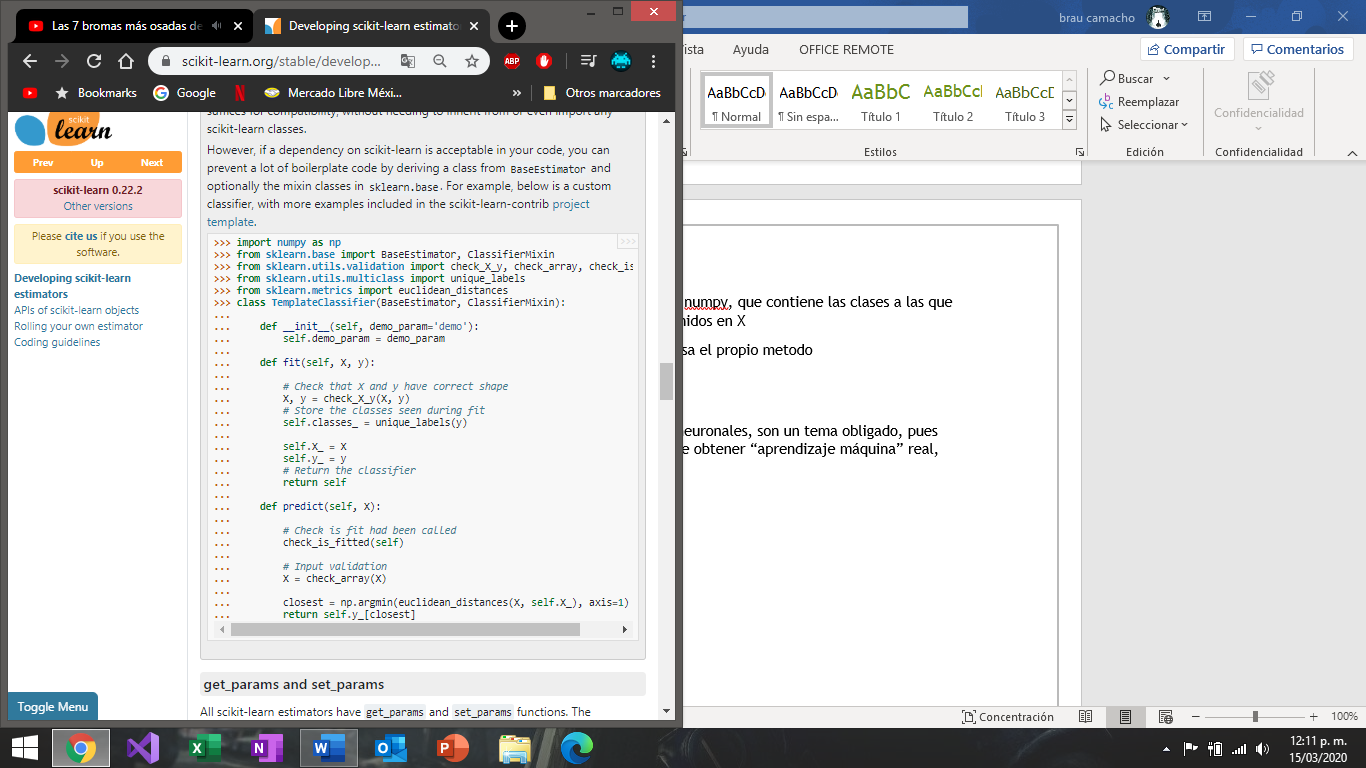
Las herramientas que provee scikit learn permiten probar estimadores (incluidos) y adaptar clasificadores externos a la estructura predefinida

Por lo general, la estructura de sikit learn en cuanto a clasificadores esta basada en 2 métodos básicos:

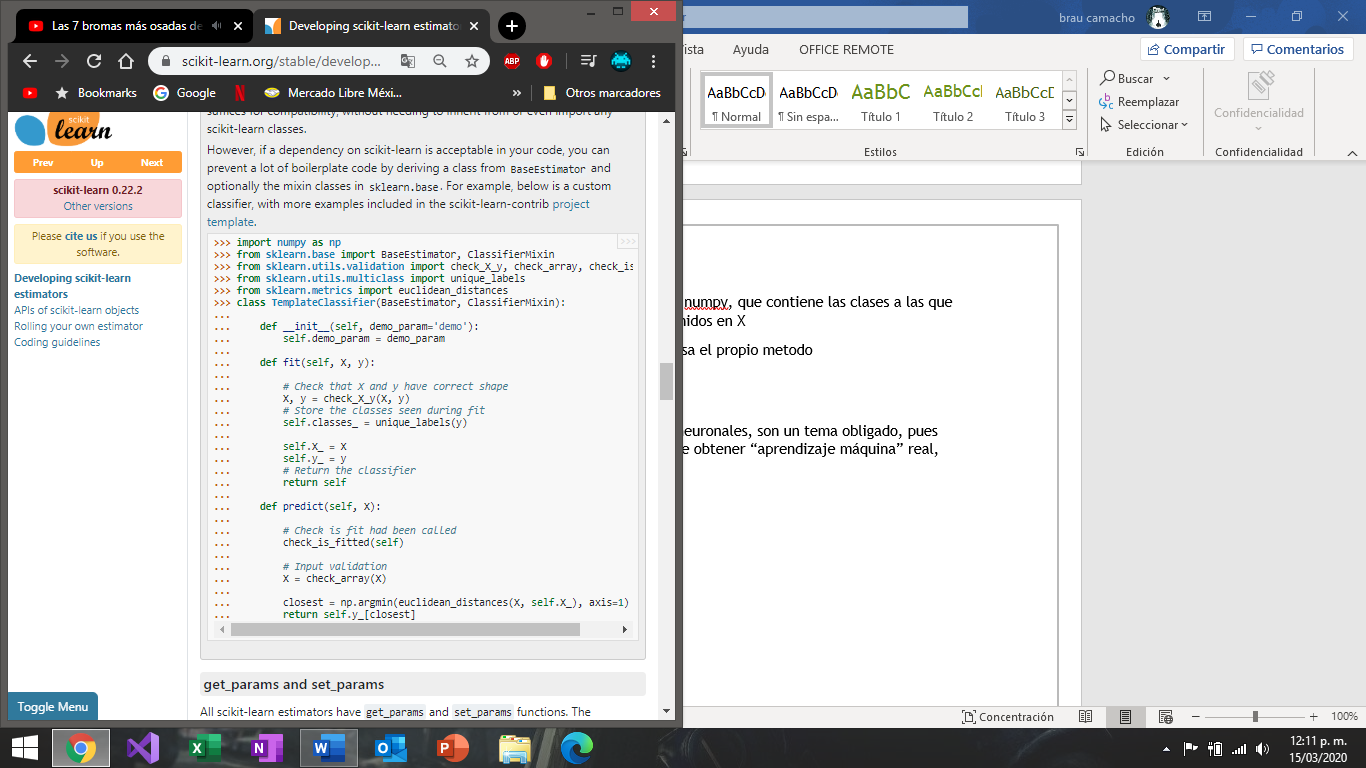
* Fit().- método usado para facilitarle a el clasificador, los datos con lo que entenar. Los atributos de entrada son 2 dada la estructura de scikit learn:

1. X.- el conjunto de datos, sin la clase, contenidos en una matriz bidimensional de tipo “numpy.array()”
2. y.- el arreglo, también de tipo numpy, que contiene las clases a las que pertenecen los vectores contenidos en X

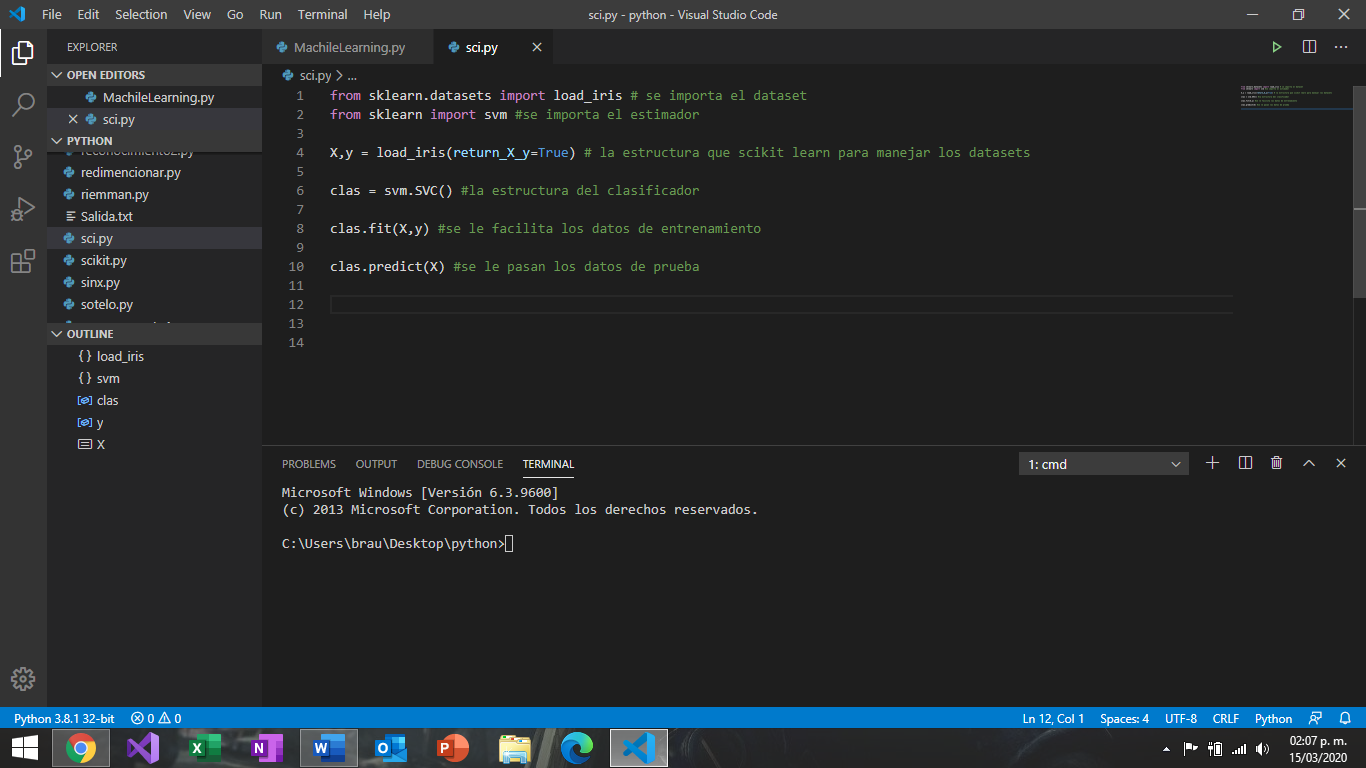
Este método, según la estructura básica regresa el propio método



* Predict().- Este método permite probar el entrenamiento del clasificador, pues como único atributo de entrada, es el conjunto de datos para experimentar, y regresa un arreglo que contiene las clases a las que pertenecen



para utilizar los clasificadores contenidos scikit learn se debe seguir la siguiente estructura



## Perceptrón paralelo

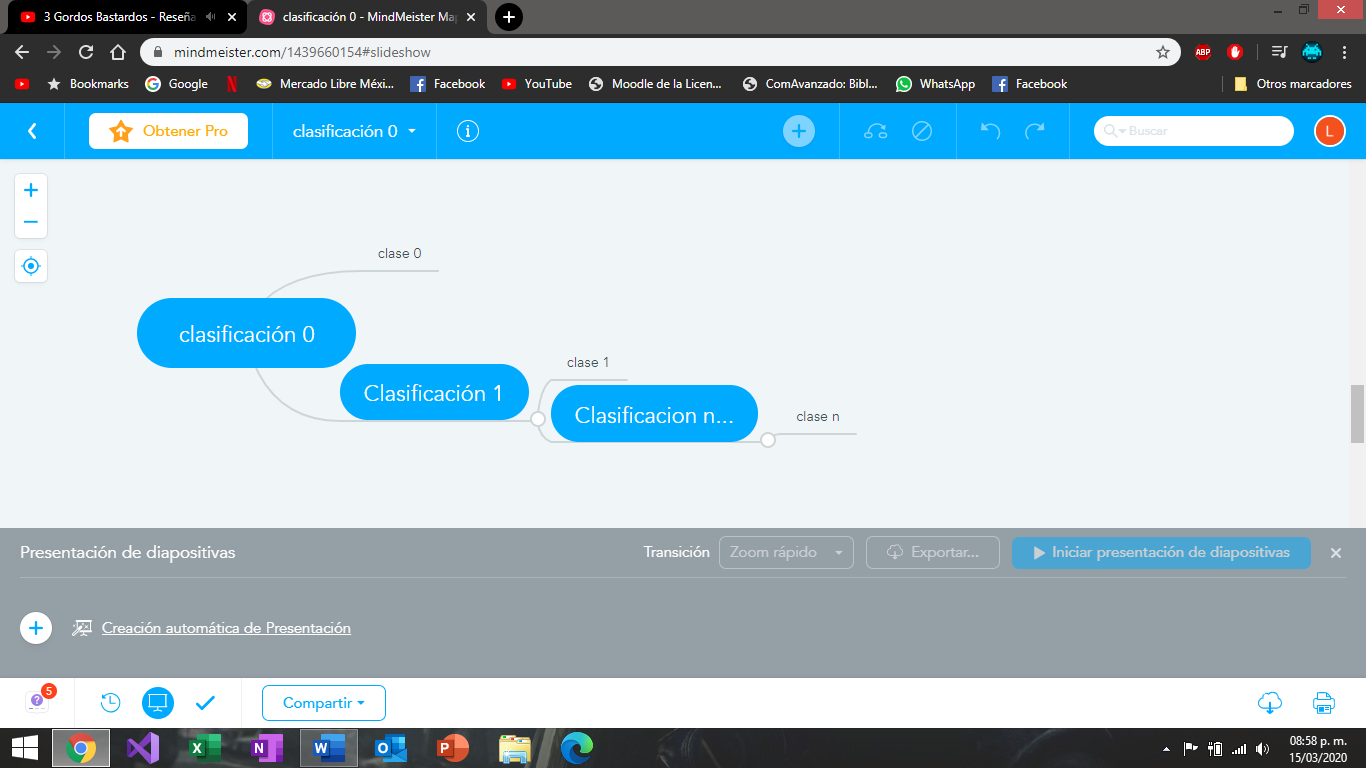
Hablando de inteligencia artificial, las redes neuronales, son un tema obligado, pues según muchos expertos, es la mejor manera de obtener “aprendizaje máquina” real, a pesar de que el perceptrón unicapa es la representación más básica de una red neuronal, lo cierto es que, esta versión del perceptrón tiene una estructura muy robusta, pues su regla re aprendizaje es la regla pdelta, que permite actualizar el vector de peso en función de la comparación de el resultado obtenido, comparado con el esperado, pero ya hablando de el funcionamiento, este cuenta con la siguiente estructura:

* **Input**: es el vector de entrada, el cual debe ir acompañado de una clase y debe estar normalizado.
* **Vector de pesos**: vector con el que se hará un producto escalar.
* **P-delta rule**: se basa en un principio muy sencillo, solo actualizar, si la salida obtenida es diferente a la esperada:



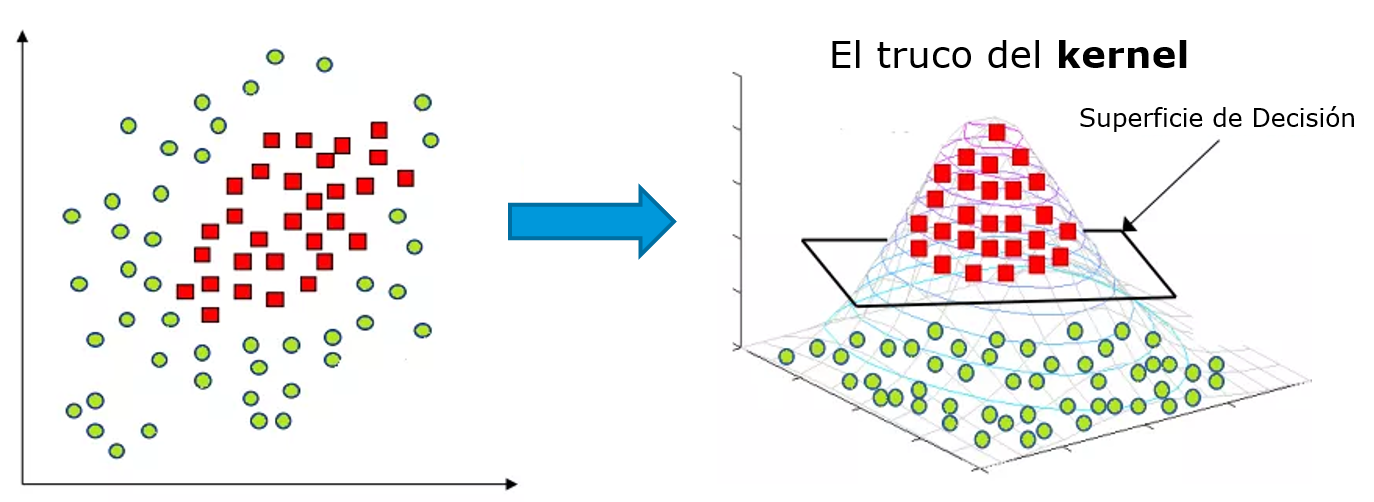
Se debe iterar la cantidad de veces que sea necesario, pues cada iteración, el ajuste en el vector de pesos será menor.

A pesar de lo robusta que sea la estructura de el clasificador, se limita solo a 2 clases, es decir, solo puede diferenciar entre 2 clases, pues para poder extender su capacidad, se debe usar la estrategia de clasificación en cascada



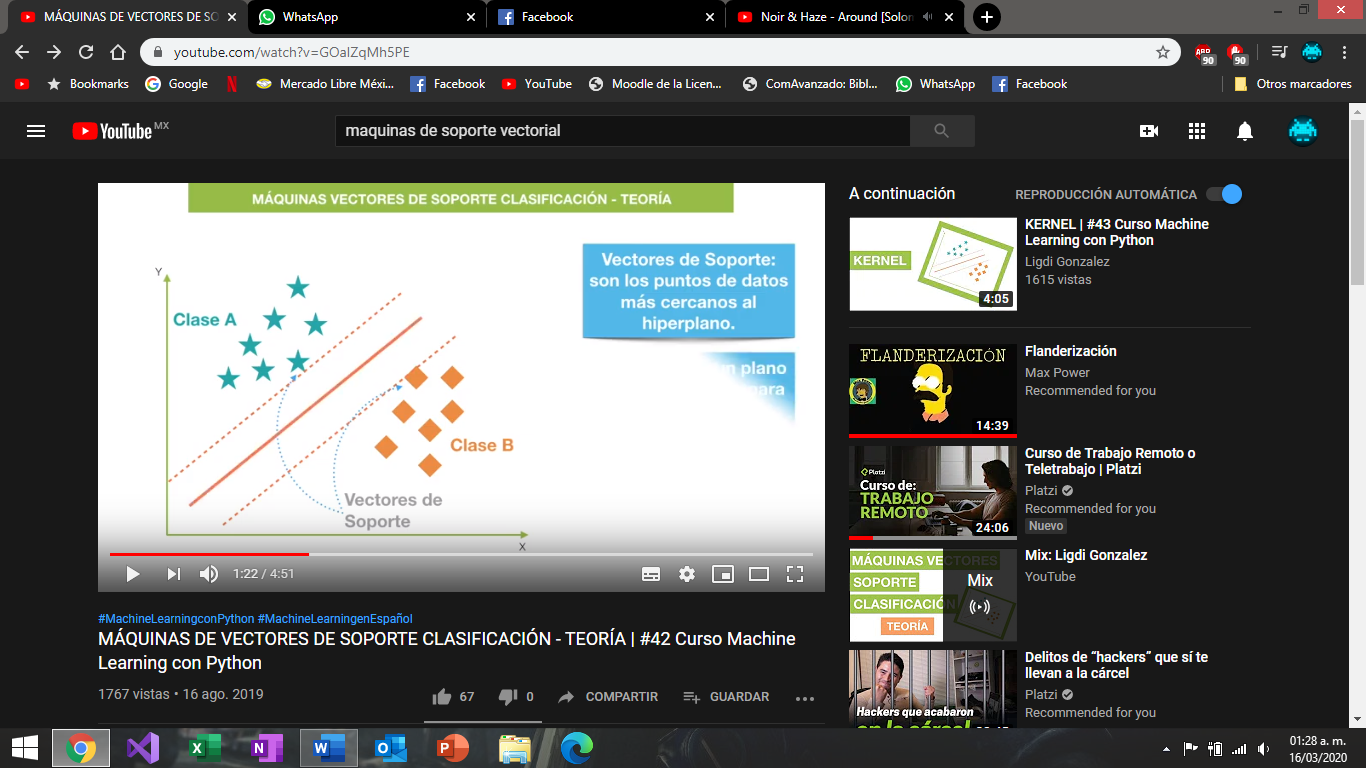
## Máquinas de soporte vectorial

Dentro del mundo de los clasificadores supervisados, las maquinas de soporte vectorial es uno de los mas curiosos, pues a pesar de que funcionan con el principio de la separabilidad lineal, lo cierto es que agrega una dimensión extra a los vectores, para generar mayor margen entre las distintas clase, haciendo uso de un kerlel(explicado más adelante), lo que lo hace muy útil para resolver solo cierto tipo de problemas, y haciéndolo completamente inútil para otros.



Para entender el funcionamiento básico del modelo matemático se deben tener en cuenta los siguientes conceptos

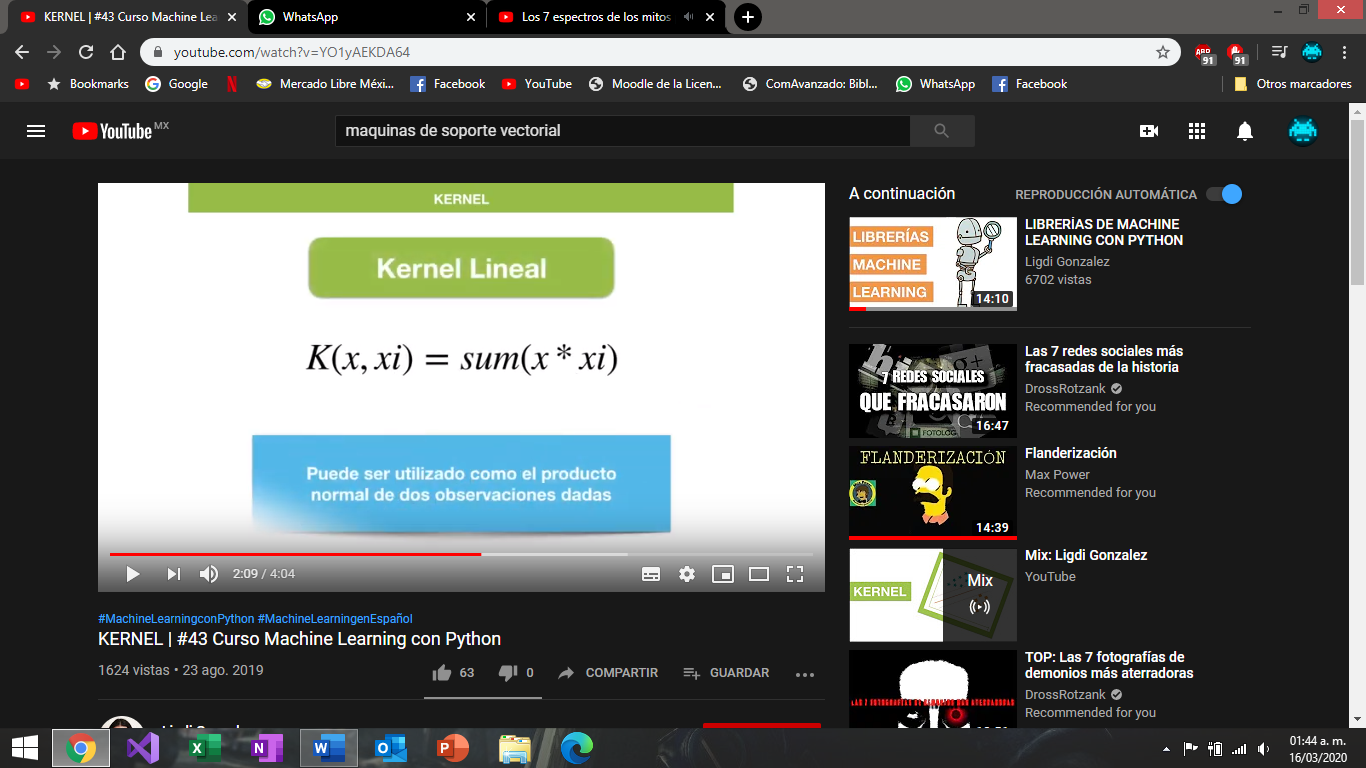
* **Vectores de soporte**: son los vectores entre clases, más separados entre si



* **Hiperplano:** plano de separación o decisión para discernir entre una clase y otra
* **Margen:** espacio entre los vectores de soporte
* **Kernel:** truco matemático, usado para agregar una dimensión y así maximizar la distancia entre los conjuntos de vectores

Lo mas importante y por lo tanto lo que lo diferencia entre otros clasificadores es el kernel, y existen varios tipos de ellos:

* **Kernel lineal**: Puede ser usado como el producto normal de dos observaciones dadas



* **Kernel polinomial**: no solo observa las características dadas por las muestras de entrada para determinar su similitud, si no también las combinaciones de estas

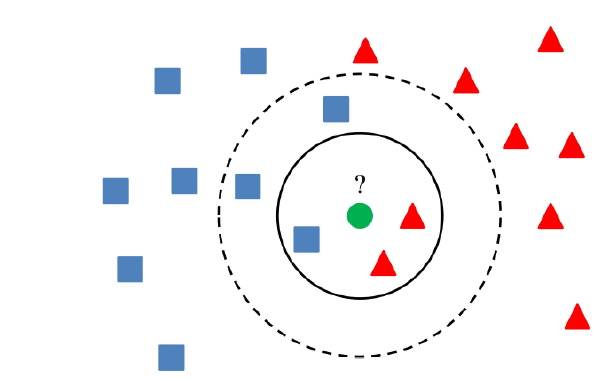


* **Kernel RBF:** puede mapear un espacio de entrada en un espacio dimensional infinito



## KNN(K- nearest neighbor)

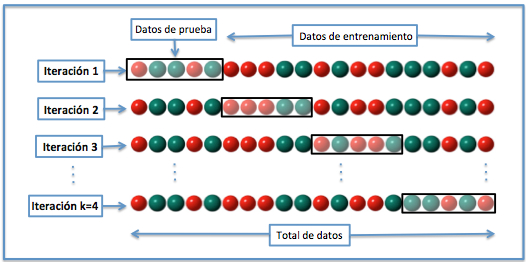
Uno de los estimadores más usados y fáciles de entender y programar, también es muy útil sobre todo para datasets no linealmente separables, pus está basado en la distancia (euclídea, Manhattan, o Minkowski) entre un vector no identificado, o de prueba y sus vecinos más cercanos (de ahí el nombre)(número totalmente seleccionable), y asigna el vector a la clase de vectores que tenga mayor incidencia en los vecinos.



No solo se usa la distancia, pues existen variables que usan un radio de distancia alrededor de el vector.

# validación cruzada

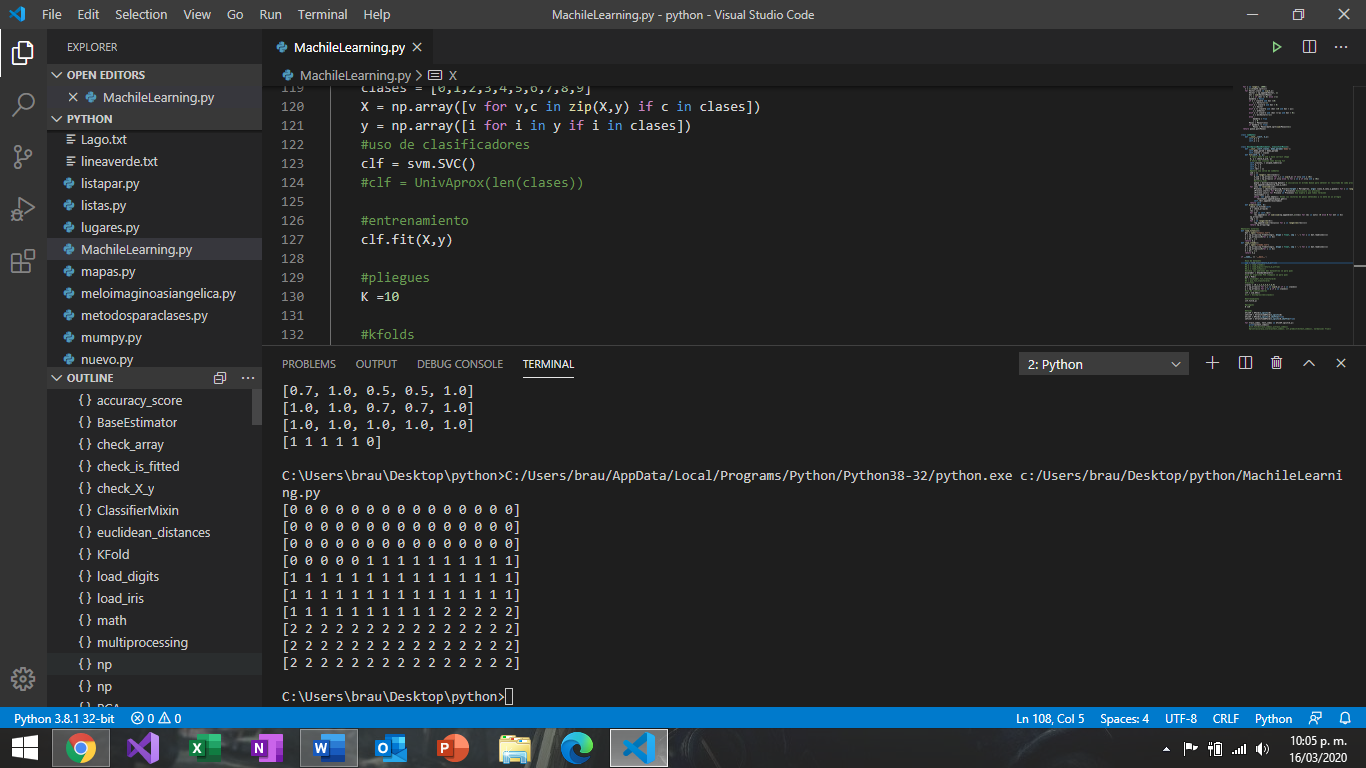
la validación cruzada o Kfolds cross validation, es un método para probar algoritmos de clasificación supervisada, reservando datos(del dataset), tanto para experimentación, como para entrenamiento, a pesar de que existen varias modalidades de validación cruzada, la funcionalidad básica es la siguiente:



En donde K, es la cantidad de iteraciones.

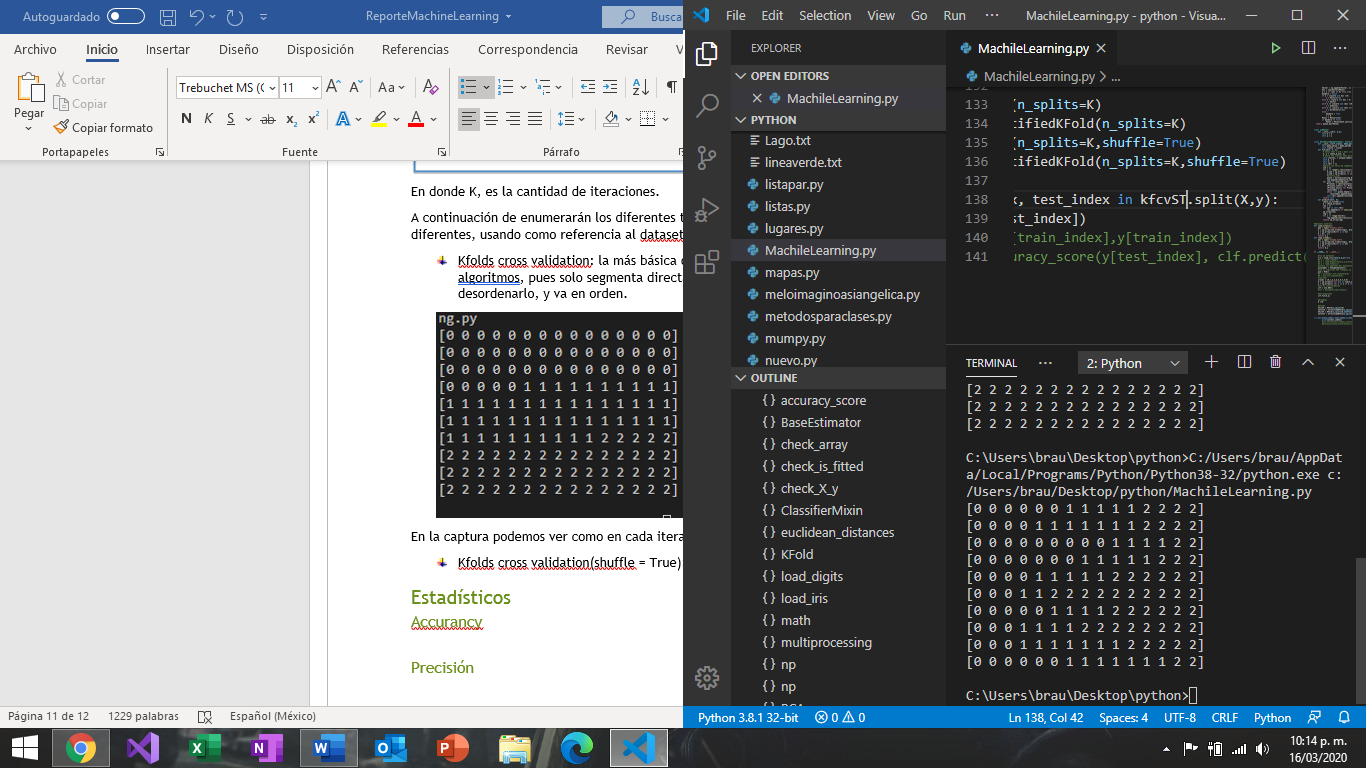
A continuación de enumerarán los diferentes tipos de validación cruzada y que las hace diferentes, usando como referencia al dataset irisplant

* Kfolds cross validation: la más básica de las implementaciones de este algoritmo, pues solo segmenta directamente el dataset como viene, sin desordenarlo, y va en orden.

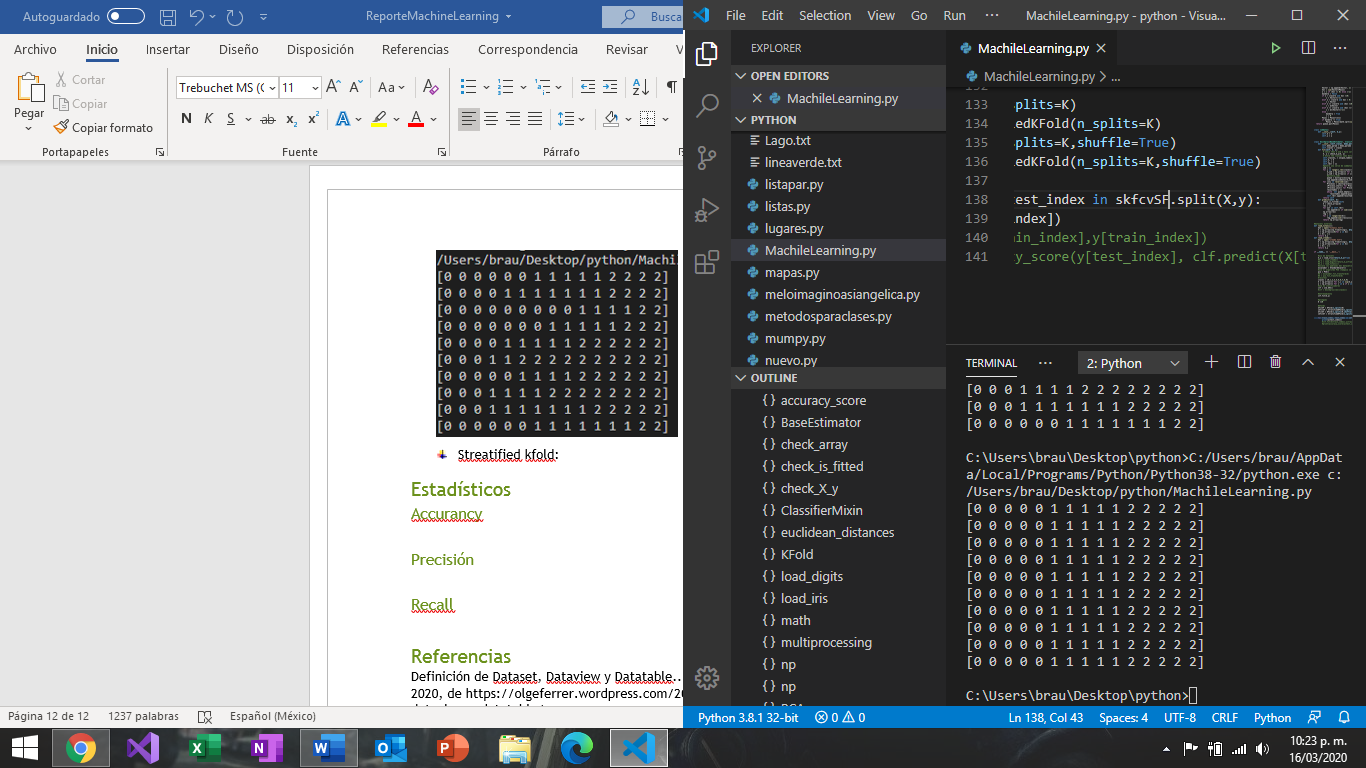


En la captura podemos ver como en cada iteración va en orden de principio a fin

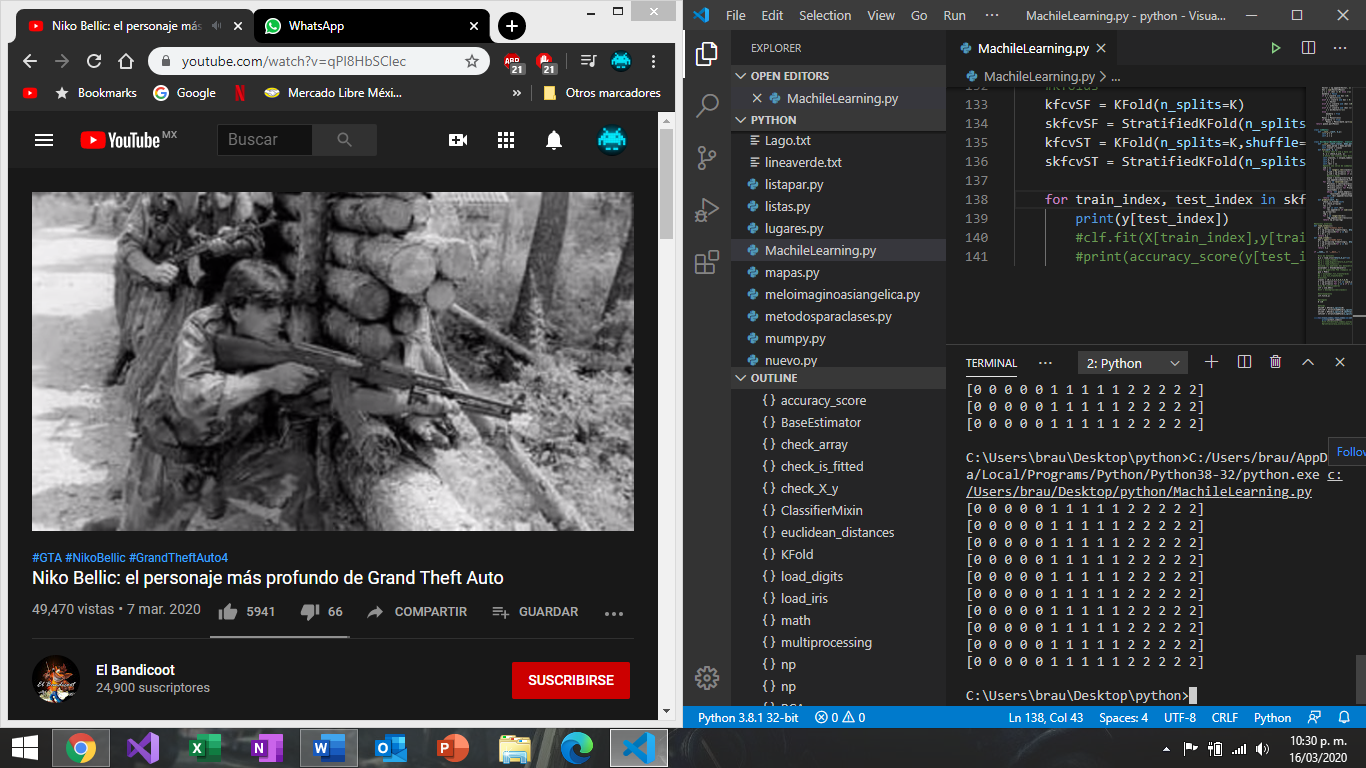
* Kfolds cross validation(shuffle = True): este toma aleatoriamente de todas las clases



* Streatified kfold: trata(en la medida de lo posible) de tomar la misma cantidad de cada clase



* Stretified kfold(shuffle = True): toma aleatoriamente cantidades diferentes por clase, pero lo misma por iteración



# Experimentos

Descripción de experimentos

* PCA de 9.0
* SVM(support vector machine)
* Kernel = RBF
* Parallel perceptrón
* Épocas =1000
* E = 0
* Y = 0.5
* M = 1
* KNN
* Neibors = 3

# Irisplant

## Pruebas Irisplant – SVM – sin PCA

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| No.prueba | Entrenamiento | | | Prueba | | |
| Exactitud | Precisión | Sensibilidad | Exactitud | Precisión | Sensibilidad |
| 1 | 1 | 1 | 1 | 0.96296296 | 0.96310935 | 0.96296296 |
| 2 | 0.93333333 | 0.94444444 | 0.93333333 | 0.97777778 | 0.97793149 | 0.97777778 |
| 3 | 1 | 1 | 1 | 0.97777778 | 0.97793149 | 0.97777778 |
| 4 | 0.93333333 | 0.94444444 | 0.93333333 | 0.98518519 | 0.98518519 | 0.98518519 |
| 5 | 1 | 1 | 1 | 0.96296296 | 0.96310935 | 0.96296296 |
| 6 | 1 | 1 | 1 | 0.95555556 | 0.95612733 | 0.95555556 |
| 7 | 0.93333333 | 0.94444444 | 0.93333333 | 0.97777778 | 0.97793149 | 0.97777778 |
| 8 | 1 | 1 | 1 | 0.97037037 | 0.97037037 | 0.97037037 |
| 9 | 0.93333333 | 0.94444444 | 0.93333333 | 0.97777778 | 0.97793149 | 0.97777778 |
| 10 | 1 | 1 | 1 | 0.97777778 | 0.97793149 | 0.97777778 |
| M | 0.970370369 | 0.97530864 | 0.97037037 | 0.97366255 | 0.97382774 | 0.97366255 |
| desv | 0.03513642 | 0.029280351 | 0.03513642 | 0.0091558 | 0.00903699 | 0.0091558 |

## Pruebas Irisplant – SVM – con PCA

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| No.prueba | Entrenamiento | | | Prueba | | |
| Exactitud | Precisión | Sensibilidad | Exactitud | Precisión | Sensibilidad |
| 1 | 0.93333333 | 0.94444444 | 0.93333333 | 0.93333333 | 0.93452381 | 0.93333333 |
| 2 | 0.86666667 | 0.9047619 | 0.86666667 | 0.94074074 | 0.94292351 | 0.94074074 |
| 3 | 0.93333333 | 0.94444444 | 0.93333333 | 0.93333333 | 0.93346509 | 0.93333333 |
| 4 | 0.86666667 | 0.9047619 | 0.86666667 | 0.94074074 | 0.94292351 | 0.94074074 |
| 5 | 0.93333333 | 0.94444444 | 0.93333333 | 0.93333333 | 0.93666667 | 0.93333333 |
| 6 | 0.93333333 | 0.94444444 | 0.93333333 | 0.93333333 | 0.93452381 | 0.93333333 |
| 7 | 0.93333333 | 0.94444444 | 0.93333333 | 0.93333333 | 0.93452381 | 0.93333333 |
| 8 | 0.86666667 | 0.9047619 | 0.86666667 | 0.93333333 | 0.93666667 | 0.93333333 |
| 9 | 0.86666667 | 0.9047619 | 0.86666667 | 0.93333333 | 0.93666667 | 0.93333333 |
| 10 | 1 | 1 | 1 | 0.92592593 | 0.92643906 | 0.92592593 |
| m | 0.913333333 | 0.93412698 | 0.91333333 | 0.93333333 | 0.93666667 | 0.93407407 |
| de | 0.044996568 | 0.030406023 | 0.04499657 | 0.00420479 | 0.00473207 | 0.00420479 |

## Pruebas Irisplant – KNN – sin PCA

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| No.prueba | Entrenamiento | | | Prueba | | |
| Exactitud | Precisión | Sensibilidad | Exactitud | Precisión | Sensibilidad |
| 1 | 0.86666667 | 0.8968254 | 0.8968254 | 0.97777778 | 0.97709655 | 0.97709655 |
| 2 | 1 | 1 | 1 | 0.95555556 | 0.95396825 | 0.95396825 |
| 3 | 1 | 1 | 1 | 0.95555556 | 0.9555336 | 0.9555336 |
| 4 | 0.93333333 | 0.88888889 | 0.93333333 | 0.96296296 | 0.96407647 | 0.96435185 |
| 5 | 0.93333333 | 0.96666667 | 0.88888889 | 0.97037037 | 0.96955544 | 0.96955544 |
| 6 | 0.93333333 | 0.93333333 | 0.88888889 | 0.96296296 | 0.96435185 | 0.96407647 |
| 7 | 1 | 1 | 1 | 0.95555556 | 0.9555336 | 0.9555336 |
| 8 | 1 | 1 | 1 | 0.95555556 | 0.9555336 | 0.9555336 |
| 9 | 1 | 1 | 1 | 0.95555556 | 0.9555336 | 0.9555336 |
| 10 | 1 | 1 | 1 | 0.95555556 | 0.95650118 | 0.95650118 |
| M | 0.966666666 | 0.967619048 | 0.96079365 | 0.96074074 | 0.95895418 | 0.96076841 |
| de | 0.047140452 | 0.045544462 | 0.05207838 | 0.00784703 | 0.00775793 | 0.00775793 |

## Pruebas Irisplant – KNN – con PCA

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| No.prueba | Entrenamiento | | | Prueba | | |
| Exactitud | Precisión | Sensibilidad | Exactitud | Precisión | Sensibilidad |
| 1 | 0.8 | 0.85 | 0.84126984 | 0.96296296 | 0.96190476 | 0.96159267 |
| 2 | 0.86666667 | 0.89166667 | 0.89166667 | 0.94074074 | 0.93862434 | 0.93862434 |
| 3 | 0.93333333 | 0.93333333 | 0.94444444 | 0.94814815 | 0.94972623 | 0.94762846 |
| 4 | 1 | 1 | 1 | 0.93333333 | 0.93697479 | 0.93472222 |
| 5 | 0.93333333 | 0.96666667 | 0.88888889 | 0.94074074 | 0.94005168 | 0.94118665 |
| 6 | 0.86666667 | 0.88888889 | 0.77777778 | 0.94074074 | 0.9449113 | 0.94233734 |
| 7 | 1 | 1 | 1 | 0.93333333 | 0.93353126 | 0.9331357 |
| 8 | 0.93333333 | 0.95238095 | 0.91666667 | 0.93333333 | 0.93479343 | 0.93280632 |
| 9 | 0.93333333 | 0.93333333 | 0.94444444 | 0.94074074 | 0.94104084 | 0.94104084 |
| 10 | 0.8 | 0.75 | 0.75555556 | 0.96296296 | 0.96539683 | 0.96327817 |
| M | 0.906666666 | 0.916626984 | 0.89607143 | 0.9437037 | 0.94469555 | 0.94363527 |
| de | 0.071664512 | 0.07589376 | 0.08400969 | 0.01115219 | 0.0110762 | 0.01090733 |

## Pruebas Irisplant – Parallel perceptron – sin PCA

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| No.prueba | Entrenamiento | | | Prueba | | |
| Exactitud | Precisión | Sensibilidad | Exactitud | Precisión | Sensibilidad |
| 1 | 0.93333333 | 0.95238095 | 0.94444444 | 0.98518519 | 0.98550725 | 0.98484848 |
| 2 | 1 | 1 | 1 | 0.97777778 | 0.97793149 | 0.97777778 |
| 3 | 1 | 1 | 1 | 0.97777778 | 0.97708872 | 0.97757074 |
| 4 | 1 | 1 | 1 | 0.97777778 | 0.97806366 | 0.97760211 |
| 5 | 1 | 1 | 1 | 0.97777778 | 0.97864262 | 0.97790404 |
| 6 | 0.93333333 | 0.88888889 | 0.94444444 | 0.98518519 | 0.9854798 | 0.9854798 |
| 7 | 1 | 1 | 1 | 0.97777778 | 0.97809984 | 0.97809984 |
| 8 | 0.93333333 | 0.94444444 | 0.94444444 | 0.99259259 | 0.99275362 | 0.99242424 |
| 9 | 0.93333333 | 0.96296296 | 0.88888889 | 0.96296296 | 0.96279461 | 0.9636947 |
| 10 | 1 | 1 | 1 | 0.97777778 | 0.97809984 | 0.97809984 |
| M | 0.973333332 | 0.974867724 | 0.972222221 | 0.97925926 | 0.97944615 | 0.97935016 |
| De | 0.03442652 | 0.037646163 | 0.039283711 | 0.00765034 | 0.00775699 | 0.00743671 |

## Pruebas Irisplant – Parallel perceptron – con PCA

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| No.prueba | Entrenamiento | | | Prueba | | |
| Exactitud | Precisión | Sensibilidad | Exactitud | Precisión | Sensibilidad |
| 1 | 0.93333333 | 0.95238095 | 0.94444444 | 0.91111111 | 0.90959282 | 0.90909091 |
| 2 | 0.93333333 | 0.94444444 | 0.93333333 | 0.91111111 | 0.91305791 | 0.91111111 |
| 3 | 1 | 1 | 1 | 0.9037037 | 0.90198105 | 0.90234645 |
| 4 | 1 | 1 | 1 | 0.91111111 | 0.91333333 | 0.91040843 |
| 5 | 0.8 | 0.8 | 0.83333333 | 0.93333333 | 0.93497864 | 0.93434343 |
| 6 | 1 | 1 | 1 | 0.91111111 | 0.91538462 | 0.91161616 |
| 7 | 0.73333333 | 0.72222222 | 0.71666667 | 0.94074074 | 0.94331066 | 0.94170692 |
| 8 | 0.86666667 | 0.87777778 | 0.87777778 | 0.91851852 | 0.91886856 | 0.91734007 |
| 9 | 0.86666667 | 0.93333333 | 0.77777778 | 0.92592593 | 0.92491582 | 0.92570078 |
| 10 | 1 | 1 | 1 | 0.91111111 | 0.9138322 | 0.91239936 |
| M | 0.913333333 | 0.923015872 | 0.908333333 | 0.91777778 | 0.91892556 | 0.91760636 |
| de | 0.094542433 | 0.095819069 | 0.103214786 | 0.01181579 | 0.01232261 | 0.01242278 |

# Generado

## Pruebas Generado – SVM – sin PCA

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| No.prueba | Entrenamiento | | | Prueba | | |
| Exactitud | Precisión | Sensibilidad | Exactitud | Precisión | Sensibilidad |
| 1 | 0.993 | 0.99382779 | 0.99146959 | 0.99477778 | 0.99518192 | 0.99398475 |
| 2 | 0.993 | 0.99332715 | 0.99220752 | 0.99477778 | 0.9952345 | 0.99391366 |
| 3 | 0.994 | 0.99416436 | 0.99332358 | 0.99466667 | 0.99513889 | 0.99379011 |
| 4 | 0.992 | 0.99290338 | 0.99057697 | 0.99488889 | 0.99527893 | 0.99409946 |
| 5 | 0.992 | 0.99286931 | 0.99067363 | 0.99488889 | 0.99528007 | 0.99409345 |
| 6 | 0.997 | 0.99708476 | 0.99669519 | 0.99433333 | 0.99481734 | 0.99341182 |
| 7 | 0.995 | 0.99502481 | 0.99467614 | 0.99455556 | 0.99505199 | 0.99363047 |
| 8 | 0.997 | 0.99748744 | 0.99630542 | 0.99433333 | 0.99477008 | 0.99345684 |
| 9 | 0.996 | 0.99672668 | 0.99491094 | 0.99444444 | 0.99485914 | 0.99361161 |
| 10 | 0.997 | 0.99707799 | 0.99661623 | 0.99433333 | 0.99481226 | 0.99343392 |
| M | 0.9946 | 0.995049367 | 0.993745521 | 0.9946 | 0.99504251 | 0.99374261 |
| de | 0.002065591 | 0.001874263 | 0.002424835 | 0.00022951 | 0.00020805 | 0.00027012 |

## Pruebas Generado – SVM – con PCA

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| No.prueba | Entrenamiento | | | Prueba | | |
| Exactitud | Precisión | Sensibilidad | Exactitud | Precisión | Sensibilidad |
| 1 | 0.993 | 0.99382779 | 0.99146959 | 0.99477778 | 0.99518192 | 0.99398475 |
| 2 | 0.993 | 0.99332715 | 0.99220752 | 0.99477778 | 0.9952345 | 0.99391366 |
| 3 | 0.994 | 0.99416436 | 0.99332358 | 0.99466667 | 0.99513889 | 0.99379011 |
| 4 | 0.992 | 0.99290338 | 0.99057697 | 0.99488889 | 0.99527893 | 0.99409946 |
| 5 | 0.992 | 0.99286931 | 0.99067363 | 0.99488889 | 0.99528007 | 0.99409345 |
| 6 | 0.997 | 0.99708476 | 0.99669519 | 0.99433333 | 0.99481734 | 0.99341182 |
| 7 | 0.995 | 0.99502481 | 0.99467614 | 0.99455556 | 0.99505199 | 0.99363047 |
| 8 | 0.997 | 0.99748744 | 0.99630542 | 0.99433333 | 0.99477008 | 0.99345684 |
| 9 | 0.996 | 0.99672668 | 0.99491094 | 0.99444444 | 0.99485914 | 0.99361161 |
| 10 | 0.997 | 0.99707799 | 0.99661623 | 0.99433333 | 0.99481226 | 0.99343392 |
| M | 0.9946 | 0.995049367 | 0.993745521 | 0.9946 | 0.99504251 | 0.99374261 |
| de | 0.002065591 | 0.001874263 | 0.002424835 | 0.00022951 | 0.00020805 | 0.00027012 |

## Pruebas Generado – KNN – sin PCA

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| No.prueba | Entrenamiento | | | Prueba | | |
| Exactitud | Precisión | Sensibilidad | Exactitud | Precisión | Sensibilidad |
| 1 | 0.995 | 0.99546821 | 0.99389239 | 0.99455556 | 0.9949026 | 0.99381237 |
| 2 | 0.998 | 0.99832776 | 0.99752475 | 0.99433333 | 0.99472263 | 0.99350441 |
| 3 | 0.992 | 0.9925 | 0.99089641 | 0.99488889 | 0.99532493 | 0.99405934 |
| 4 | 0.995 | 0.99461562 | 0.99500797 | 0.99466667 | 0.99514049 | 0.99378348 |
| 5 | 0.995 | 0.99538505 | 0.99432602 | 0.99466667 | 0.99504878 | 0.99385985 |
| 6 | 0.991 | 0.99241147 | 0.98918269 | 0.99511111 | 0.99537096 | 0.99445775 |
| 7 | 0.997 | 0.99670899 | 0.99708532 | 0.99444444 | 0.99486281 | 0.99359225 |
| 8 | 0.994 | 0.99503311 | 0.99253731 | 0.99477778 | 0.99509204 | 0.99405838 |
| 9 | 0.993 | 0.99332819 | 0.99219548 | 0.99488889 | 0.99527916 | 0.99409826 |
| 10 | 0.994 | 0.99415471 | 0.99323245 | 0.99477778 | 0.99518217 | 0.99398352 |
| M | 0.995 | 0.99546821 | 0.99389239 | 0.99455556 | 0.9949026 | 0.99381237 |
| de | 0.998 | 0.99832776 | 0.99752475 | 0.99433333 | 0.99472263 | 0.99350441 |

## Pruebas Generado – KNN – con PCA

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| No.prueba | Entrenamiento | | | Prueba | | |
| Exactitud | Precisión | Sensibilidad | Exactitud | Precisión | Sensibilidad |
| 1 | 0.995 | 0.99546821 | 0.99389239 | 0.99455556 | 0.9949026 | 0.99381237 |
| 2 | 0.998 | 0.99832776 | 0.99752475 | 0.99433333 | 0.99472263 | 0.99350441 |
| 3 | 0.992 | 0.9925 | 0.99089641 | 0.99488889 | 0.99532493 | 0.99405934 |
| 4 | 0.995 | 0.99461562 | 0.99500797 | 0.99466667 | 0.99514049 | 0.99378348 |
| 5 | 0.995 | 0.99538505 | 0.99432602 | 0.99466667 | 0.99504878 | 0.99385985 |
| 6 | 0.991 | 0.99241147 | 0.98918269 | 0.99511111 | 0.99537096 | 0.99445775 |
| 7 | 0.997 | 0.99670899 | 0.99708532 | 0.99444444 | 0.99486281 | 0.99359225 |
| 8 | 0.994 | 0.99503311 | 0.99253731 | 0.99477778 | 0.99509204 | 0.99405838 |
| 9 | 0.993 | 0.99332819 | 0.99219548 | 0.99488889 | 0.99527916 | 0.99409826 |
| 10 | 0.994 | 0.99415471 | 0.99323245 | 0.99477778 | 0.99518217 | 0.99398352 |
| M | 0.9944 | 0.994793311 | 0.993588079 | 0.99471111 | 0.99509266 | 0.99392096 |
| de | 0.0021187 | 0.001842212 | 0.00258637 | 0.00022951 | 0.00021194 | 0.00027437 |

## Pruebas Generado – Parallel perceptron – sin PCA

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| No.prueba | Entrenamiento | | | Prueba | | |
| Exactitud | Precisión | Sensibilidad | Exactitud | Precisión | Sensibilidad |
| 1 | 0.994 | 0.99417156 | 0.99348857 | 0.99377778 | 0.99398112 | 0.99306938 |
| 2 | 0.991 | 0.99165227 | 0.9897505 | 0.99422222 | 0.99439826 | 0.99358582 |
| 3 | 0.994 | 0.99460555 | 0.99288803 | 0.99388889 | 0.99407475 | 0.9932287 |
| 4 | 0.996 | 0.9968 | 0.99472296 | 0.994 | 0.99425807 | 0.99329919 |
| 5 | 0.992 | 0.99290338 | 0.99057697 | 0.99477778 | 0.99509287 | 0.99405163 |
| 6 | 0.992 | 0.99175023 | 0.99175023 | 0.99455556 | 0.99495617 | 0.99372403 |
| 7 | 0.997 | 0.99754098 | 0.99618321 | 0.99388889 | 0.99416595 | 0.99314554 |
| 8 | 0.993 | 0.99331375 | 0.99232348 | 0.99455556 | 0.99491009 | 0.99376096 |
| 9 | 0.999 | 0.9991511 | 0.99878641 | 0.99344444 | 0.99370374 | 0.99266039 |
| 10 | 0.996 | 0.99575552 | 0.99575552 | 0.994 | 0.99434835 | 0.99321088 |
| M | 0.9944 | 0.994764434 | 0.993622588 | 0.99411111 | 0.99438894 | 0.99337365 |
| de | 0.00254733 | 0.00251205 | 0.002768912 | 0.00041243 | 0.0004585 | 0.00040604 |

## Pruebas Generado – Parallel perceptron – con PCA

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| No.prueba | Entrenamiento | | | Prueba | | |
| Exactitud | Precisión | Sensibilidad | Exactitud | Precisión | Sensibilidad |
| 1 | 0.994 | 0.99417156 | 0.99348857 | 0.99377778 | 0.99398112 | 0.99306938 |
| 2 | 0.991 | 0.99165227 | 0.9897505 | 0.99422222 | 0.99439826 | 0.99358582 |
| 3 | 0.994 | 0.99460555 | 0.99288803 | 0.99422222 | 0.99444441 | 0.99355265 |
| 4 | 0.996 | 0.9968 | 0.99472296 | 0.99388889 | 0.99412024 | 0.99320574 |
| 5 | 0.992 | 0.99290338 | 0.99057697 | 0.99477778 | 0.99509287 | 0.99405163 |
| 6 | 0.99 | 0.98935228 | 0.99004665 | 0.99455556 | 0.99495617 | 0.99372403 |
| 7 | 0.997 | 0.99754098 | 0.99618321 | 0.994 | 0.99430443 | 0.99323875 |
| 8 | 0.993 | 0.99331375 | 0.99232348 | 0.99455556 | 0.99491009 | 0.99376096 |
| 9 | 0.999 | 0.9991511 | 0.99878641 | 0.99333333 | 0.99356466 | 0.9925675 |
| 10 | 0.996 | 0.99575552 | 0.99575552 | 0.99411111 | 0.99448656 | 0.99330432 |
| M | 0.9942 | 0.994524639 | 0.99345223 | 0.99414445 | 0.99442588 | 0.99340608 |
| de | 0.002820559 | 0.002914038 | 0.002943793 | 0.00042569 | 0.00047294 | 0.00042066 |

# Dígitos

## Pruebas dígitos – SVM – sin PCA

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| No.prueba | Entrenamiento | | | Prueba | | |
| Exactitud | Precisión | Sensibilidad | Exactitud | Precisión | Sensibilidad |
| 1 | 0.98888889 | 0.98766667 | 0.99166667 | 0.995671 | 0.99570933 | 0.9955166 |
| 2 | 0.98888889 | 0.98848684 | 0.98848684 | 0.99690785 | 0.99691955 | 0.996916 |
| 3 | 0.98888889 | 0.9902381 | 0.98935574 | 0.99690785 | 0.99689181 | 0.99685232 |
| 4 | 0.98888889 | 0.98809524 | 0.98856209 | 0.99690785 | 0.99693899 | 0.99691344 |
| 5 | 0.98333333 | 0.98454106 | 0.98464027 | 0.99628942 | 0.99630924 | 0.99628389 |
| 6 | 0.97777778 | 0.97690476 | 0.97847339 | 0.99690785 | 0.99694209 | 0.99690885 |
| 7 | 0.99444444 | 0.99545455 | 0.99333333 | 0.99628942 | 0.99636269 | 0.99627583 |
| 8 | 0.99441341 | 0.996 | 0.99411765 | 0.99690977 | 0.99696139 | 0.99690559 |
| 9 | 0.99441341 | 0.99545455 | 0.99444444 | 0.99505562 | 0.99509687 | 0.99508291 |
| 10 | 0.98882682 | 0.98968254 | 0.98989899 | 0.99690977 | 0.99691614 | 0.99682942 |
| M | 0.988876475 | 0.989252431 | 0.989297941 | 0.99647564 | 0.99650481 | 0.99644849 |
| de | 0.005230552 | 0.005799623 | 0.004853865 | 0.00065468 | 0.00064573 | 0.00066282 |

## Pruebas dígitos – SVM – con PCA

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| No.prueba | Entrenamiento | | | Prueba | | |
| Exactitud | Precisión | Sensibilidad | Exactitud | Precisión | Sensibilidad |
| 1 | 0.99444444 | 0.99166667 | 0.99583333 | 0.99752628 | 0.99754266 | 0.99743779 |
| 2 | 0.99444444 | 0.99375 | 0.99473684 | 0.99876314 | 0.99877296 | 0.99878788 |
| 3 | 0.98333333 | 0.98571429 | 0.98459384 | 0.99752628 | 0.99747506 | 0.99754601 |
| 4 | 0.98888889 | 0.99054622 | 0.98856209 | 0.99876314 | 0.99875772 | 0.99875772 |
| 5 | 0.98888889 | 0.98976982 | 0.98940217 | 0.99814471 | 0.99816586 | 0.99815156 |
| 6 | 0.98333333 | 0.98309524 | 0.98435574 | 0.99814471 | 0.99817296 | 0.99813973 |
| 7 | 0.99444444 | 0.99545455 | 0.99333333 | 0.99814471 | 0.99818903 | 0.99818908 |
| 8 | 0.98882682 | 0.991 | 0.98697479 | 0.99814586 | 0.99816653 | 0.9981517 |
| 9 | 0.99441341 | 0.9952381 | 0.99411765 | 0.99814586 | 0.99816185 | 0.99820359 |
| 10 | 0.98882682 | 0.98968254 | 0.98989899 | 0.99876391 | 0.99876468 | 0.99876103 |
| M | 0.989984481 | 0.990591743 | 0.99583333 | 0.99820686 | 0.99821693 | 0.99821261 |
| de | 0.00438236 | 0.003916115 | 0.004175001 | 0.00045639 | 0.00046159 | 0.00046986 |

## Pruebas dígitos – KNN – sin PCA

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| No.prueba | Entrenamiento | | | Prueba | | |
| Exactitud | Precisión | Sensibilidad | Exactitud | Precisión | Sensibilidad |
| 1 | 0.98888889 | 0.98898551 | 0.9875 | 0.99319728 | 0.99330815 | 0.99322099 |
| 2 | 0.98888889 | 0.98976982 | 0.98935574 | 0.99257885 | 0.99265144 | 0.99252012 |
| 3 | 0.98333333 | 0.98280702 | 0.98571429 | 0.99381571 | 0.99386217 | 0.99380289 |
| 4 | 0.98333333 | 0.98375 | 0.98562711 | 0.99443414 | 0.99452395 | 0.99437869 |
| 5 | 0.98333333 | 0.98391813 | 0.98671024 | 0.99381571 | 0.99385825 | 0.99372679 |
| 6 | 0.98333333 | 0.98583333 | 0.98078431 | 0.99505257 | 0.99507246 | 0.99493636 |
| 7 | 0.99444444 | 0.99375 | 0.99444444 | 0.99319728 | 0.99336495 | 0.99318994 |
| 8 | 0.99441341 | 0.99444444 | 0.996 | 0.99134734 | 0.99128567 | 0.99136202 |
| 9 | 0.98882682 | 0.98848684 | 0.99065217 | 0.99258344 | 0.99256306 | 0.99242645 |
| 10 | 0.99441341 | 0.99411765 | 0.99333333 | 0.99258344 | 0.99270738 | 0.99260547 |
| M | 0.988320918 | 0.988586274 | 0.989012163 | 0.99326058 | 0.99331975 | 0.99321697 |
| De | 0.004855005 | 0.004463531 | 0.004683746 | 0.00106745 | 0.00108731 | 0.00104537 |

## Pruebas dígitos – KNN – con PCA

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| No.prueba | Entrenamiento | | | Prueba | | |
| Exactitud | Precisión | Sensibilidad | Exactitud | Precisión | Sensibilidad |
| 1 | 0.98333333 | 0.98315217 | 0.98125 | 0.99257885 | 0.99267105 | 0.99258807 |
| 2 | 0.99444444 | 0.99444444 | 0.9952381 | 0.99196042 | 0.99200711 | 0.99190662 |
| 3 | 0.98333333 | 0.98280702 | 0.98571429 | 0.99319728 | 0.99327708 | 0.99317396 |
| 4 | 0.98333333 | 0.98375 | 0.98562711 | 0.99381571 | 0.9938931 | 0.99375757 |
| 5 | 0.97777778 | 0.97894737 | 0.98300654 | 0.99257885 | 0.99274719 | 0.99243302 |
| 6 | 0.98333333 | 0.98583333 | 0.98078431 | 0.99381571 | 0.99378595 | 0.99367048 |
| 7 | 0.99444444 | 0.99375 | 0.99444444 | 0.99257885 | 0.99270161 | 0.99256882 |
| 8 | 0.99441341 | 0.99444444 | 0.996 | 0.99196539 | 0.9919557 | 0.99199493 |
| 9 | 0.99441341 | 0.99473684 | 0.995 | 0.99196539 | 0.99202155 | 0.9917642 |
| 10 | 0.98324022 | 0.98160714 | 0.98041667 | 0.99258344 | 0.99272204 | 0.99258606 |
| M | 0.987206702 | 0.987347275 | 0.987748146 | 0.99270399 | 0.99277824 | 0.99264437 |
| de | 0.00644026 | 0.006265362 | 0.006643392 | 0.00070084 | 0.00069702 | 0.00069684 |

## Pruebas dígitos – Parallel perceptron – sin PCA

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| No. prueba | Entrenamiento | | | Prueba | | |
| Exactitud | Precisión | Sensibilidad | Exactitud | Precisión | Sensibilidad |
| 1 | 0.31666667 | 0.48688453 | 0.29371981 | 0.30550402 | 0.49459358 | 0.3083171 |
| 2 | 0.26111111 | 0.56795031 | 0.25453339 | 0.3135436 | 0.40185003 | 0.31526811 |
| 3 | 0.3 | 0.58593527 | 0.3197076 | 0.3283859 | 0.40470545 | 0.32665032 |
| 4 | 0.31666667 | 0.49887668 | 0.30662142 | 0.33147805 | 0.45045117 | 0.33320041 |
| 5 | 0.28333333 | 0.49303268 | 0.32918207 | 0.35250464 | 0.50991794 | 0.34591539 |
| 6 | 0.3 | 0.60638053 | 0.34865497 | 0.35683364 | 0.50866736 | 0.3512614 |
| 7 | 0.35555556 | 0.50388573 | 0.30167903 | 0.29560915 | 0.49022483 | 0.30228498 |
| 8 | 0.35195531 | 0.50456493 | 0.32858148 | 0.31829419 | 0.49127362 | 0.32145386 |
| 9 | 0.31284916 | 0.47825657 | 0.27965226 | 0.29666255 | 0.38326394 | 0.30063056 |
| 10 | 0.30167598 | 0.50079328 | 0.33686869 | 0.32509271 | 0.50111502 | 0.32097066 |
| M | 0.309981379 | 0.522656051 | 0.309920072 | 0.32239085 | 0.46360629 | 0.32259528 |
| De | 0.028486029 | 0.04584652 | 0.02860994 | 0.02102483 | 0.0493598 | 0.01716242 |

## Pruebas dígitos – Parallel perceptron – con PCA

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| No.prueba | Entrenamiento | | | Prueba | | |
| Exactitud | Precisión | Sensibilidad | Exactitud | Precisión | Sensibilidad |
| 1 | 0.31111111 | 0.56769353 | 0.28816425 | 0.29993816 | 0.47959563 | 0.30312718 |
| 2 | 0.32222222 | 0.48365117 | 0.30717971 | 0.30117502 | 0.48153952 | 0.30299351 |
| 3 | 0.28888889 | 0.56181624 | 0.30611111 | 0.33209647 | 0.49176627 | 0.33003717 |
| 4 | 0.31666667 | 0.5110831 | 0.30521702 | 0.30426716 | 0.48776485 | 0.30597848 |
| 5 | 0.2 | 0.44935821 | 0.24375781 | 0.29004329 | 0.47686402 | 0.28449444 |
| 6 | 0.23333333 | 0.5654286 | 0.2725731 | 0.29931973 | 0.47591581 | 0.29554548 |
| 7 | 0.33888889 | 0.49342732 | 0.28962766 | 0.3030303 | 0.58232362 | 0.30975738 |
| 8 | 0.30726257 | 0.47838845 | 0.28100197 | 0.30160692 | 0.48024625 | 0.30537702 |
| 9 | 0.31843575 | 0.47895928 | 0.29436404 | 0.31829419 | 0.48860048 | 0.32047062 |
| 10 | 0.30167598 | 0.50899412 | 0.34141414 | 0.32818294 | 0.49419768 | 0.32363582 |
| M | 0.293848541 | 0.509880002 | 0.292941081 | 0.30779542 | 0.49388141 | 0.30814171 |
| De | 0.04122176 | 0.041762549 | 0.025579369 | 0.0136724 | 0.0317045 | 0.01356434 |

# Titanic

## Pruebas titanic – SVM – sin PCA

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| No.prueba | Entrenamiento | | | Prueba | | |
| Exactitud | Precisión | Sensibilidad | Exactitud | Precisión | Sensibilidad |
| 1 | 0.69662921 | 0.7001634 | 0.63101604 | 0.67919799 | 0.70321089 | 0.60190167 |
| 2 | 0.69662921 | 0.69810127 | 0.58704116 | 0.68295739 | 0.69128951 | 0.61765055 |
| 3 | 0.6741573 | 0.72816781 | 0.63666667 | 0.68170426 | 0.6947355 | 0.6019802 |
| 4 | 0.70786517 | 0.69991438 | 0.62801535 | 0.68045113 | 0.69097744 | 0.61165389 |
| 5 | 0.66292135 | 0.80519481 | 0.64285714 | 0.68045113 | 0.68146341 | 0.60084337 |
| 6 | 0.73033708 | 0.74804382 | 0.66951872 | 0.68170426 | 0.68960169 | 0.61178108 |
| 7 | 0.62921348 | 0.59099099 | 0.5540107 | 0.68671679 | 0.69475684 | 0.61887755 |
| 8 | 0.59090909 | 0.56803653 | 0.53948066 | 0.68961202 | 0.70372258 | 0.61539457 |
| 9 | 0.65909091 | 0.55701754 | 0.53095238 | 0.68210263 | 0.68794232 | 0.62136713 |
| 10 | 0.73863636 | 0.74305556 | 0.65846067 | 0.67459324 | 0.67705167 | 0.60823428 |
| M | 0.678638916 | 0.683868611 | 0.607801949 | 0.68194908 | 0.69147519 | 0.61096843 |
| De | 0.045414775 | 0.083774891 | 0.050862498 | 0.0040658 | 0.00839094 | 0.00752068 |

## Pruebas titanic – SVM – con PCA

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| No.prueba | Entrenamiento | | | Prueba | | |
| Exactitud | Precisión | Sensibilidad | Exactitud | Precisión | Sensibilidad |
| 1 | 0.6741573 | 0.67117117 | 0.60160428 | 0.66165414 | 0.66767311 | 0.58279221 |
| 2 | 0.65168539 | 0.58333333 | 0.53003337 | 0.6716792 | 0.68432682 | 0.59969496 |
| 3 | 0.65168539 | 0.72835498 | 0.60820513 | 0.6641604 | 0.66143436 | 0.58207821 |
| 4 | 0.70786517 | 0.69991438 | 0.62801535 | 0.6679198 | 0.67664385 | 0.59434823 |
| 5 | 0.61797753 | 0.74113924 | 0.59650456 | 0.66666667 | 0.66044096 | 0.58251004 |
| 6 | 0.71910112 | 0.73651961 | 0.65481283 | 0.66917293 | 0.67374382 | 0.59494434 |
| 7 | 0.64044944 | 0.61634199 | 0.55748663 | 0.67669173 | 0.68551587 | 0.60408163 |
| 8 | 0.59090909 | 0.56923077 | 0.53577107 | 0.67459324 | 0.68483449 | 0.59509524 |
| 9 | 0.68181818 | 0.6025641 | 0.54761905 | 0.6795995 | 0.6848793 | 0.61818242 |
| 10 | 0.72727273 | 0.74461538 | 0.63497453 | 0.66082603 | 0.66691638 | 0.58588438 |
| M | 0.666292134 | 0.669318495 | 0.58950268 | 0.66929636 | 0.6746409 | 0.59396117 |
| de | 0.044284586 | 0.070271172 | 0.044177091 | 0.00632398 | 0.01005486 | 0.01148276 |

## Pruebas titanic – KNN – sin PCA

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| No.prueba | Entrenamiento | | | Prueba | | |
| Exactitud | Precisión | Sensibilidad | Exactitud | Precisión | Sensibilidad |
| 1 | 0.73033708 | 0.71408991 | 0.70882353 | 0.84210526 | 0.83641748 | 0.82741187 |
| 2 | 0.69662921 | 0.67701149 | 0.66470588 | 0.84210526 | 0.83817568 | 0.825 |
| 3 | 0.71910112 | 0.70189099 | 0.69411765 | 0.83834586 | 0.83477691 | 0.82012987 |
| 4 | 0.71910112 | 0.70258621 | 0.68850267 | 0.839599 | 0.83496206 | 0.82295918 |
| 5 | 0.71910112 | 0.70430108 | 0.6828877 | 0.84461153 | 0.8395501 | 0.82945269 |
| 6 | 0.76404494 | 0.7924812 | 0.70802139 | 0.83709273 | 0.83364568 | 0.81850649 |
| 7 | 0.65168539 | 0.63079096 | 0.62248677 | 0.83709273 | 0.83399267 | 0.81756967 |
| 8 | 0.69662921 | 0.68107345 | 0.66957672 | 0.84210526 | 0.83850135 | 0.82408433 |
| 9 | 0.63636364 | 0.61655773 | 0.61655773 | 0.84105131 | 0.83625058 | 0.82468855 |
| 10 | 0.76136364 | 0.75561627 | 0.72930283 | 0.84605757 | 0.84083733 | 0.83118206 |
| M | 0.709435647 | 0.697639929 | 0.678498287 | 0.84101665 | 0.83671098 | 0.82409847 |
| de | 0.041363491 | 0.052128816 | 0.036483485 | 0.00301529 | 0.00245676 | 0.00449602 |

## Pruebas titanic – KNN – con PCA

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| No.prueba | Entrenamiento | | | Prueba | | |
| Exactitud | Precisión | Sensibilidad | Exactitud | Precisión | Sensibilidad |
| 1 | 0.70786517 | 0.68954802 | 0.67941176 | 0.79448622 | 0.78622717 | 0.77416512 |
| 2 | 0.58426966 | 0.55762987 | 0.55695187 | 0.79699248 | 0.78752042 | 0.77982375 |
| 3 | 0.68539326 | 0.6752551 | 0.68368984 | 0.79824561 | 0.79019827 | 0.77843228 |
| 4 | 0.59550562 | 0.57573375 | 0.57727273 | 0.80701754 | 0.79838644 | 0.79039889 |
| 5 | 0.65168539 | 0.62518797 | 0.58903743 | 0.79699248 | 0.78898727 | 0.77680891 |
| 6 | 0.62921348 | 0.60773026 | 0.60396825 | 0.80075188 | 0.79176029 | 0.7825451 |
| 7 | 0.59550562 | 0.56332139 | 0.55608466 | 0.80576441 | 0.79787645 | 0.78661841 |
| 8 | 0.68181818 | 0.66666667 | 0.66993464 | 0.80725907 | 0.79983141 | 0.78811794 |
| 9 | 0.63636364 | 0.61383929 | 0.61111111 | 0.78723404 | 0.77730412 | 0.76819438 |
| 10 | 0.63636364 | 0.61655773 | 0.61655773 | 0.80600751 | 0.79939625 | 0.78528447 |
| M | 0.640398366 | 0.619147005 | 0.614402002 | 0.80007512 | 0.79174881 | 0.78103893 |
| de | 0.041843779 | 0.046367876 | 0.048257022 | 0.00655157 | 0.00724542 | 0.00689236 |

## Pruebas titanic – Parallel perceptron – sin PCA

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| No. prueba | Entrenamiento | | | Prueba | | |
| Exactitud | Precisión | Sensibilidad | Exactitud | Precisión | Sensibilidad |
| 1 | 0.83146067 | 0.8470696 | 0.80079365 | 0.80576441 | 0.80028161 | 0.78295641 |
| 2 | 0.74157303 | 0.73401468 | 0.73206861 | 0.77318296 | 0.76005492 | 0.76454295 |
| 3 | 0.84269663 | 0.84267677 | 0.84267677 | 0.80325815 | 0.79566231 | 0.77589262 |
| 4 | 0.80898876 | 0.79338154 | 0.78919956 | 0.80701754 | 0.80306956 | 0.78455843 |
| 5 | 0.76404494 | 0.74397275 | 0.75889878 | 0.76441103 | 0.75256188 | 0.74946024 |
| 6 | 0.76404494 | 0.74843505 | 0.68247126 | 0.82330827 | 0.82592593 | 0.79912058 |
| 7 | 0.74157303 | 0.72037422 | 0.73954802 | 0.78822055 | 0.77965252 | 0.77047721 |
| 8 | 0.68181818 | 0.65625 | 0.65625 | 0.7747184 | 0.76366823 | 0.75750049 |
| 9 | 0.77272727 | 0.76914085 | 0.74824798 | 0.76470588 | 0.75138036 | 0.75198949 |
| 10 | 0.77272727 | 0.77790766 | 0.75198728 | 0.81101377 | 0.80772758 | 0.78507334 |
| M | 0.772165472 | 0.763322312 | 0.750214191 | 0.7915601 | 0.78399849 | 0.77215718 |
| De | 0.047156318 | 0.056984282 | 0.054335689 | 0.02122542 | 0.02612699 | 0.01624713 |

## Pruebas titanic – Parallel perceptron – con PCA

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| No.prueba | Entrenamiento | | | Prueba | | |
| Exactitud | Precisión | Sensibilidad | Exactitud | Precisión | Sensibilidad |
| 1 | 0.80898876 | 0.80677966 | 0.78730159 | 0.77568922 | 0.76442703 | 0.75607515 |
| 2 | 0.66292135 | 0.65487179 | 0.65696466 | 0.77318296 | 0.75990707 | 0.75891664 |
| 3 | 0.83146067 | 0.83307927 | 0.83106061 | 0.76566416 | 0.75037244 | 0.74250336 |
| 4 | 0.79775281 | 0.7819209 | 0.77357456 | 0.76817043 | 0.75730994 | 0.74867795 |
| 5 | 0.80898876 | 0.78919956 | 0.79338154 | 0.79197995 | 0.78391712 | 0.77320956 |
| 6 | 0.75280899 | 0.72303922 | 0.68304598 | 0.78571429 | 0.77744619 | 0.76762623 |
| 7 | 0.75280899 | 0.73008386 | 0.7480226 | 0.78320802 | 0.77473327 | 0.76406695 |
| 8 | 0.76136364 | 0.74366577 | 0.75223214 | 0.78222778 | 0.77316268 | 0.76186424 |
| 9 | 0.80681818 | 0.828125 | 0.77169811 | 0.77972466 | 0.7690577 | 0.75928537 |
| 10 | 0.81818182 | 0.81818182 | 0.80604134 | 0.77847309 | 0.76747167 | 0.75624876 |
| M | 0.780209397 | 0.770894685 | 0.760332313 | 0.77840346 | 0.76778051 | 0.75884742 |
| De | 0.04983764 | 0.057039048 | 0.053842496 | 0.0080198 | 0.01010369 | 0.00884976 |

# Conclusiones

En este experimento, se basó prácticamente en medir el rendimiento de varios clasificadores supervisados, junto con varios datasets, en diferentes condiciones, es decir con y sin reducción de características(PCA), con y sin escalador(normalizar), y si bien existen diferencias en los resultados, con estas variables, lo cierto es que la diferencia es mínima no me enfocaré en esto, si no en los clasificadores en si, que es donde está la verdadera diferencia, no solo en resultados de las métricas, si no en tiempo de ejecución. Lo primero en tomar en cuenta, es que los mejores clasificadores en mi experimento, son las maquinas de soporte vectorial, y el clasificador K- nearest-neibor, pues ofrecen resultados bastante buenos, con un bajo consumo de tiempo y recursos de procesamiento, cosa con la que el perceptrón paralelo no cuenta, pues para ofrecer resultados aceptables, en los datasets mas exigentes, requieren de una cantidad insana de recursos de cómputo, cosa que se refleja en los resultados, considerablemente menores, pues para realizar éste experimento se utilizaron solo 4 neuronas, y 1000 épocas.

Para terminar, y en conclusión, recomiendo el uso del perceptrón paralelo, solo en casos donde las decimas y las milésimas, sean de extrema importancia, pues si se requieren resultados aceptables, no tan precisos es mejor usar cualquiera de los otros dos clasificadores.

# Referencias

Definición de Dataset, Dataview y Datatable... (2011, 18 marzo). Recuperado 10 marzo, 2020, de https://olgeferrer.wordpress.com/2011/03/18/definicion-de-dataset-dataview-y-datatable/