

Modelo de Machine Learning usando Azure ML Studio

Manuel Zúñiga, Alejandra Martínez, Wilmer David Guarín y Juan José Bedoya

{manuel.znniga@uao.edu.co, alejandra.martinez_h@uao.edu.co, wilmer.guarin@uao.edu.co y
juan.bedoya@uao.edu.co}

Universidad Autónoma de Occidente

Resumen— Machine Learning permite crear una máquina de aprendizaje continuo que solucione problemas en específico, así que esta disciplina es indispensable para hacer sistemas capaces de identificar patrones entre los datos, y hacer predicciones además de análisis estadísticos. En el servicio Azure que ofrece Microsoft en la nube mediante esta disciplina (Machine Learning) se brinda un servicio de análisis predictivo que permite crear e implementar rápidamente tanto modelos predictivos como soluciones de análisis.

Índice de Términos— Azure, Machine Learning, Inteligencia Artificial, la nube, optimización y patrones.

I. INTRODUCCIÓN

En los años 50's gracias a Arthur Samuel, conocido por ser el pionero de la inteligencia artificial, escribió el primer programa de aprendizaje informático. Samuel contribuyó que mediante inteligencia artificial en el juego de “damas” la computadora mejorará en el juego conforme a las partidas que iba jugando, en donde se incorporan las jugadas ganadoras en el programa.



Figura 1, A.L Samuel

<https://img.unocero.com/2015/03/samuel00.jpg>

En los años 90's la disciplina de Machine Learning tuvo un avance significativo gracias a la intersección de la informática con las técnicas de probabilidad y estadísticas, dando la posibilidad de trabajar con más datos. Esto permitió que esta tecnología se comenzará a usar en aplicaciones web,

aprendizaje de idiomas, áreas comerciales para la minería de datos, entre otros.

En la actualidad, el machine learning se encuentra en todas las áreas, como en el medio ambiente, como lo es para el proyecto del que se estará hablando. En el que mediante Machine Learning identifica 3 flores gracias a los parámetros que se van ingresando de acuerdo a las características de las flores.

Para la entrega de este proyecto se tiene en cuenta conceptos claves para iniciar a analizar las herramientas y métodos que debemos usar para realizar el modelo de Machine Learning adecuadamente, Azure Machine Learning Studio es el principal recurso de Machine Learning. Esta funcionalidad proporciona un lugar centralizado para que los científicos de datos y desarrolladores trabajen con todos los artefactos para crear, entrenar e implementar modelos de Machine Learning.

II. Descripción del problema a solucionar

Muchas veces al identificar una flor de todo un campo de flores, e identificar el tipo de planta se hace tedioso por la gran cantidad de flora que existe y más si se necesita identificar no solo 1 sino más de 100 especies de flores, lo que resultaría un trabajo arduo y monótono.

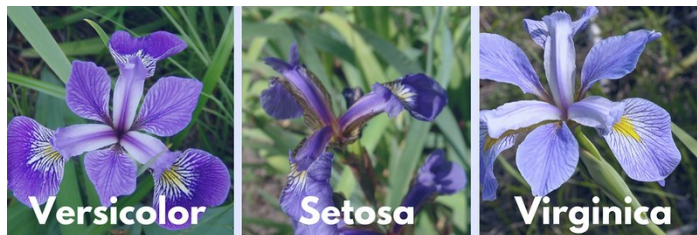


Figura 2, Campo de flores iris

https://live.staticflickr.com/7866/47075467621_85ab810139_c.jpg

Al tener en cuenta que con solo bases de datos y programación no basta para reconocer patrones y mucho menos analizarlos, Se hace uso del modelo machine learning, que ha sido utilizado por varios años, siendo una de las tecnologías a emplear diferentes usos en la nueva era de la infraestructura de la nube, gracias al modelo predictivo se logra no solo un

resultado casi automático, si no también asertivo, en el que la máquina va a almacenar esta respuesta para seguir mejorando su actividad con el tiempo. En conclusión, es necesario elaborar varios pasos para crear una máquina que haga un análisis probabilístico repetidas veces, que además este programa aprenda de estos errores, logrando analizar con una exactitud del casi 100% los tipos de flores que se necesita conocer y lo haga de manera casi instantánea.

III. Descripción de la Solución

Planteamiento de objetivos a solucionar usando el modelamiento de la máquina de aprendizaje con azure

1. Capacidad de identificar patrones de las características de una flor, según el tamaño del tallo y del pétalo.
2. Capacidad de identificar patrones entre los datos que se desea identificar para hacer predicciones continuamente de las flores.
3. Probar el correcto funcionamiento de la máquina según la exactitud del reconocimiento de la flor, usando porcentajes.

Para resolver el problema anterior se necesita crear un grupo de recursos para usar las apps de Azure he implementar machine learning que solucione el reconocimiento de 3 tipos de flores iris (Setosa, Virginica y Versicolor) ingresando varios detalles de las flores, ya sea el tallo, tipo de pétalos, y que la máquina de aprendizaje pueda responder asertivamente cual es cada flor según las características antes mencionadas.

Inicialmente se necesitan de los recursos de Azure Apps, específicamente el que se llama “Azure Machine Learning” que se encuentra en la infraestructura en la nube de Azure, para hacer uso de esta plataforma hay que tener una cuenta y pagar por sus servicios, pero nosotros al ser estudiantes de una universidad dan 100 créditos para usarlos en todas las plataformas que se desee aprender, por consiguiente para empezar a realizar la práctica hay configurar el recurso y entrar al workspace, que facilita el acceso a los DataStores y los DataSets.

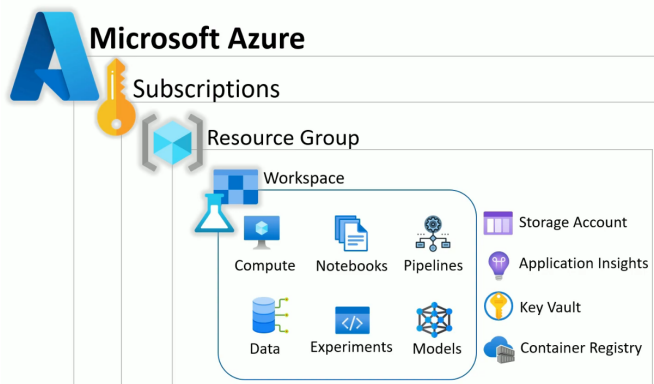


Figura 3, Mapa Azure

<https://azure.microsoft.com/es-es/services/monitor/#overview>

Tal cual como se puede observar en la figura 3, se necesita de una suscripción para acceder a los recursos que maneja Azure y crear nuestro espacio de trabajo que en este caso tendrá el servicio de machine learning, que tendrá el siguiente icono que se muestra en la siguiente figura.

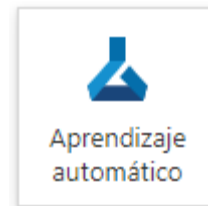


Figura 4, Icono Azure

<https://azure.microsoft.com/es-es/services/monitor/#overview>

Luego de crear los nombres y configuraciones utilizar el Workspace, lo cual se debe crear una instancia para que los códigos que vamos a usar y subirlos a la carpeta, con el fin de tener un cluster y se entrene la máquina de aprendizaje.

Pero para lograr esto debemos tener un DataSet que nos servirá inicialmente para tener datos con los que la máquina va a empezar a entrenar y testear para así aprender de los errores y aciertos, por esto mismo usamos un repositorio en github que contiene los datos necesarios para que funcione el servicio, ese archivo debe estar en un formato csv que se separa por comas y contiene una tabla de datos que se usa como entrenamiento, luego de lograr esto vamos a los directorios de Notebooks de jupyter para enlazar los scripts de Python, para ello se necesita crear un cluster dentro del Machine Learning de 0 a 3 nodos, esto permite que se enlacen todas las máquinas en una sola y si el programa requiere usar todos los nodos use para resolver los scripts de la Machine learning.

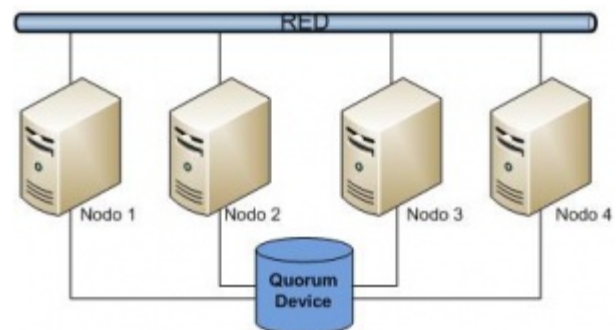


Figura 5, Ejemplo de cómo funciona un Cluster

https://www.ecured.cu/images/thumb/4/4f/Quorum_devices.JPG/300px-Quorum_devices.JPG

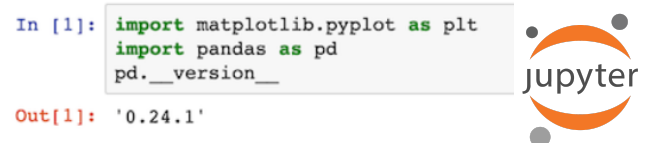


Figura 6, Ejemplo estructura de un notebook

https://upload.wikimedia.org/wikipedia/commons/thumb/3/38/Jupyter_logo.svg/1200px-Jupyter_logo.svg.png

Ahora bien, una vez se montan todos los archivos necesarios

en los notebooks, se empieza hacer todas la pruebas e iniciar código por código.

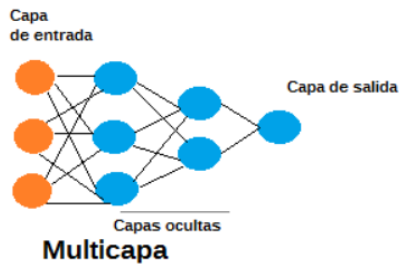


Figura 7, Ejemplo Capa neuronal

<https://inteligencia-artificial.dev/wp-content/uploads/2021/01/tipos-redes-neuronales-capas.png>

Para realizar los códigos se hace uso del concepto de las redes neuronales para realizar un modelo que aprenda de las entradas que le damos al sistema, las analice y pueda sacar un resultado probabilístico que nos diga con certeza qué tipo de flor iris es cada una, además a este tipo de red neuronal se le llama multicapa.

IV. Pruebas realizadas

Para realizar las pruebas usamos todo el servicio que creamos anteriormente e iniciamos código por código en el notebook, lo que facilita lo que se hace en cada parte del código. Pero antes de ello se prueba con un código de testeo que este funcionando correctamente el servicio, para ello se usan librerías de Azure y Python y se cambia el id del Dataset en el código de testeo para probar el funcionamiento, extrayendo el archivo del Dataset.

```
You have logged in. Now let us find all the subscriptions to which you have access...
Interactive authentication successfully completed.
Index(['sepal.length', 'sepal.width', 'petal.length', 'petal.width',
      'variety'],
      dtype='object')
   sepal.length  sepal.width  petal.length  petal.width  variety
0           5.1           3.5           1.4           0.2  Setosa
1           4.9           3.0           1.4           0.2  Setosa
2           4.7           3.2           1.3           0.2  Setosa
3           4.6           3.1           1.5           0.2  Setosa
4           5.0           3.6           1.4           0.2  Setosa
   sepal.length  sepal.width  petal.length  petal.width  variety
145           6.7           3.0           5.2           2.3  Virginica
146           6.3           2.5           5.0           1.9  Virginica
147           6.5           3.0           5.2           2.0  Virginica
```

Figura 8, Resultado de los datos analizados por el código

En la anterior imagen se muestra el resultado de la autenticación del id con los datos que teníamos en el Dataset, lo cual muestra los parámetros de las flores iris, como lo es el ancho y largo del pétalo y tallo de las flores, teniendo en sí una estadística de cada una de las 3 flores.

Según el modelo que creamos para la red neuronal nos muestra una cantidad de 512 neuronas en la capa oculta, que interpreta las 4 entradas para obtener 3 salidas es decir 3 tipos de flores iris, tal como se puede observar en la siguiente imagen.

Model: "sequential"

Layer (type)	Output Shape	Param #
flatten (Flatten)	(None, 4)	0
dense (Dense)	(None, 512)	2560
dropout (Dropout)	(None, 512)	0
dense_1 (Dense)	(None, 3)	1539
Total params: 4,099		
Trainable params: 4,099		
Non-trainable params: 0		

Figura 8, Resultado del modelo de la máquina

```
1 model.evaluate(x_test, y_test)
✓
50/50 [=====] - 0s 2ms/sample - loss: 0.0419 - accuracy: 0.9800
[0.04190216675400734, 0.98]

50/50 [=====] - 0s 84us/sample - loss: 0.0419 - accuracy: 0.9800
[[1.  0.  0. ]
 [0.  0.93 0.07]
 [0.  0.  1.  ]]
```

Figura 9, Prueba de la evaluación después de lo aprendido

Figura 10, Resultado de las flores que reconoció el modelo

Después de haber establecido las salidas y entradas del modelo se hace un testeo del modelo, para saber qué tanto aprendió, con lo anterior se puede observar que el modelo tiene una exactitud del 98% lo cual es un puntaje muy bueno, además de ello se puede observar la matriz de las flores apreciando cuales son las flores según esas entradas, aunque en la segunda flor que es la versicolor no tuvo una exactitud del 100% como lo fue con las demás flores, es un porcentaje muy acercado del 93%, y nos da una seguridad de que esa es la flor que se espera, según las entradas que se ingresaron al modelo.

V. CONCLUSIÓN

Hoy en día las bases de datos y funciones utilizadas como recursos para crear plataformas digitales para diferentes fines, nos ha servido para administrar servidores que estén creados virtualmente, físicamente o usando alguna infraestructura en la nube, por lo tanto se sugiere, crear plataformas digitales enlazadas en la nube, en las que se usan configuraciones y programación necesaria que logren generar una o varias acciones al mismo tiempo logrando cumplir satisfactoriamente la meta planteada por el equipo de trabajo.

De los 3 objetivos que se plantearon en el ítem de pruebas realizadas se puede evidenciar como la máquina logra ser asertiva con el reconocimiento de cada flor, haciendo que el modelo logre un 98% de exactitud, aunque se sugiere variar los porcentajes de las variables de testeo y aprendizaje para obtener un resultado mucho más exacto.

Muchas veces se cree que las máquinas de inteligencia artificial funcionan tal cual como un cerebro interpretativo, lo cual no es

así, ya que estas pueden aprender de forma errónea si los datos siempre se los entregamos ordenados, lo que hace que tengamos un resultado falso, para solucionar lo anterior se sugiere ingresar los datos aleatoriamente para que la máquina aprenda correctamente y por consiguiente nos muestre un resultado verdadero.

Este tema al ser un poco complejo se requiere de herramientas de programación tales como los notebooks de Júpiter, con ellos podemos dividir el código en secciones y probarlo uno por uno, lo que concluye una forma idónea de realizar este tipo de modelos que requieren de un análisis más exhaustivo para culminar correctamente con el modelo.

VI. BIBLIOGRAFÍA

- [1] Machine Learning. (s. f.). IBM - Deutschland | IBM.
<https://www.ibm.com/co-es/analytics/machine-learning>
- [2] Azure Machine Learning documentation. (s. f.).
Developer tools, technical documentation and coding examples | Microsoft Docs.
<https://docs.microsoft.com/en-us/azure/machine-learning>
- [3] Elearners365. (2021, 6 de agosto). Azure Machine Learning | Hands-on-lab on Azure Machine Learning Studio| Workspaces |by Jaya Gupta [Video]. YouTube.
<https://www.youtube.com/watch?v=w3vUPldu2TY>
- [4] DataPlatformGeeks & SQLServerGeeks. (2019, 11 de enero). Azure Machine Learning SDK by Parashar Shah [Video]. YouTube.
<https://www.youtube.com/watch?v=N7pN8xAnKXw>
- [5] IBM Docs. (s. f.). IBM - Deutschland | IBM.
<https://www.ibm.com/docs/es/spss-modeler/SaaS?topic=networks-neural-model>
- [6] Clusters de servidores. (s. f.). Seguridad Informática.
<https://infosegur.wordpress.com/unidad-2/clusters-de-servidores/>