Construcción de un espacio virtual con material de apoyo y ejercicios reproducibles para las prácticas de Machine Learning e Inteligencia Artificial

Autor
Raul Alejandro Buitrago Castellanos

Universidad Distrital Francisco José de Caldas
Maestría en Ciencias de la Información y las Comunicaciones
Énfasis en Ingeniería de Software
Bogotá, Colombia
Junio 2021

Construcción de un espacio virtual con material de apoyo y ejercicios reproducibles para las prácticas de Machine Learning e Inteligencia Artificial

Informe de resultados de la pasantía

Autor
Raul Alejandro Buitrago Castellanos

Director
Cesar Andey Perdomo Charry
Msc. Ciencias de La Informacion y Las Comunicaciones

Universidad Distrital Francisco José de Caldas
Maestría en Ciencias de la Información y las Comunicaciones
Énfasis en Ingeniería de Software
Bogotá, Colombia
Junio 2021

TABLA DE CONTENIDO

LISTA DE FIGURAS

LISTA DE TABLAS

RESUMEN

En este documento se encuentra una descripción detallada de los pasos que se siguieron para la ejecución de la pasantía académica bajo la dirección del ingeniero ***Cesar Andrey Perdomo Charry*** que busca realizar y/o consolidar material de apoyo a los docentes del grupo de investigación ***LASER*** adscrito a la *Universidad Distrital Fransisco José de Caldas* [1] para las practicas y/o laboratorios de *Machine learning e Inteligencia Artificial* por medio de ejemplos que aborden las siguientes temáticas:

1. Introducción al lenguaje de programación Python.
2. Estadística descriptiva.
3. Preparación de datos.
4. Algoritmos clásicos de Machine Learning
5. Redes Neuronales.

Para la realización de esta pasantía se utilizaran diversos conjuntos de datos disponibles en UC Irvine Machine Learning Repository, Kaggle, entre otros.

PALABRAS CLAVE

Python, CMS, WordPress, Drupal, Jupyter, Google Colab, Algoritmo, Machine Learning, Inteligencia Artificial, Redes neuronales.

1 RESULTADOS OBTENIDOS

1.1 Definición de tecnologías

Para la definición de tecnologías se tuvieron en cuenta los siguientes criterios:

1. En lo posible manejar la misma tecnología en todas las fases.
2. Escoger tecnologías populares, que contaran con una comunidad robusta y con soporte a múltiples librerías.
3. Usar la menor cantidad de infraestructura posible puesto que implica menor complejidad y esfuerzo para tareas de soporte, mantenimiento y actualización.
4. Las restricciones que debe tener el sitio web, en cuanto a ser interactivo y tener código reproducible.

En cuanto a la definición del sitio web la decisión fue el ***Colab de Google*** es un servicio de ***Jupyter*** que opera bajo la infraestructura de Google [2] evita el tema de gestionar e invertirle tiempo, esfuerzo y recursos a temas de servidores, maquinas, dominios, contenedores y todo aquello que pueda requerir a nivel de infraestructura. Ya que para efectos de la guía los ejercicios son sencillos.

Característica	Google Colab	Jupyter Hub
Infraestructura	Requiere un repositorio en ***Github*** o tener en un ***Google Drive*** los contenidos que se quieren publicar [3].	Debe contar con un servidor que contenga el Jupyter Hub, además requiere un repositorio en ***Github*** [4].

Tabla 1: Comparación Google Colab y Jupyter Hub

Teniendo en cuenta todo lo anterior y tras consultar diversas fuentes web y en textos académicos. Se tomo la decisión de usar Python como lenguaje de programación para el montaje de los ejemplos, algoritmos y las guías debido a que su sintaxis es sencilla de entender, soporta múltiples paradigmas de programación, tiene bastantes librerías que se pueden agregar fácilmente [5] [6]. Además los proyectos Jupyter Hub y Google Colab se integran muy bien con Python, por no decir que están diseñados para trabajar con Python.

Además otra cosa bien particular y agradable de usar estas herramientas como lo son Jupyter o el Google Colab, es que permite utilizar markdown y codigo html para escribir contenidos estaticos permitiendo de una forma sencilla e intuitiva incluir diseños como se puede evidenciar en la figura 1

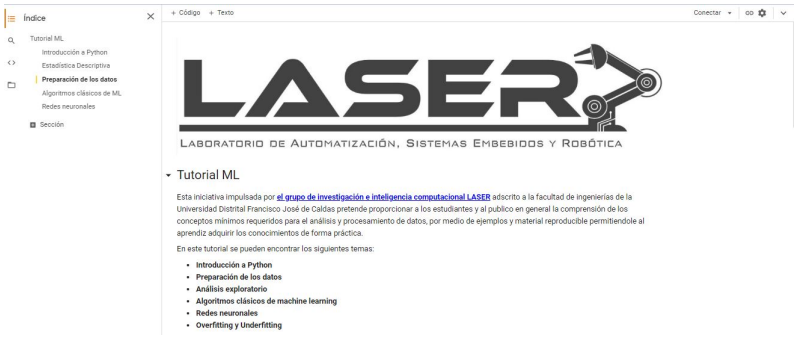

Figura 1: Visualización pantalla de inicio del blog haciendo uso de Markdown

1.2 Estadística Descriptiva

Para realizar el análisis exploratorio de los datos se utilizo la librería *pandas* de Python, ya que incorpora el uso del *Dataframe* habilitando las funciones *describe(), y corr()* las cuales sirven para obtener las medidas de tendencia central, y correlación entre variables como se puede observar en las imágenes 2 . Adicionalmente al añadir las librerías *seaborn y matplotlib.pyplot* nos permite realizar las gráficas asociadas a los diagramas de dispersión, histogramas, correlación, entre otros como se puede observar en las figuras .

```
[ ]  train_data.describe().transpose()
```

	count	mean	std	min	25%	50%	75%	max
MYCT	125.0	216.616	271.592074	17.0	50.0	125.0	240.0	1500.0
MMIN	125.0	2849.760	3801.615680	64.0	768.0	2000.0	3100.0	16000.0
MMAX	125.0	11977.248	12222.417312	64.0	4000.0	8000.0	16000.0	64000.0
CACH	125.0	23.944	40.031702	0.0	0.0	8.0	32.0	256.0
CHMIN	125.0	4.440	6.042057	0.0	1.0	3.0	6.0	52.0
CHMAX	125.0	18.176	27.237919	0.0	5.0	8.0	24.0	176.0
PRP	125.0	109.448	169.638489	6.0	28.0	50.0	106.0	1150.0
ERP	125.0	101.144	157.430000	15.0	28.0	44.0	102.0	978.0

Figura 2: Visualización comando para mostrar las medidas de tendencia central

```
# Coeficientes de correlación
correlation_matrix = train_data.corr()
correlation_matrix
```

	MYCT	MMIN	MMAX	CACH	CHMIN	CHMAX	PRP	ERP
MYCT	1.000000	-0.353649	-0.376010	-0.343344	-0.320118	-0.257330	-0.325442	-0.302930
MMIN	-0.353649	1.000000	0.731526	0.602469	0.527348	0.264624	0.737935	0.749771
MMAX	-0.376010	0.731526	1.000000	0.575013	0.528606	0.586418	0.863111	0.921245
CACH	-0.343344	0.602469	0.575013	1.000000	0.546109	0.386039	0.700521	0.691900
CHMIN	-0.320118	0.527348	0.528606	0.546109	1.000000	0.508565	0.563861	0.563363
CHMAX	-0.257330	0.264624	0.586418	0.386039	0.508565	1.000000	0.687921	0.673815
PRP	-0.325442	0.737935	0.863111	0.700521	0.563861	0.687921	1.000000	0.960478
ERP	-0.302930	0.749771	0.921245	0.691900	0.563363	0.673815	0.960478	1.000000

Figura 3: Visualización comando para mostrar la correlación de variables

Lo anterior es bien importante porque con la interpretación correcta se pueden identificar comportamientos a simple vista de los datos y se hace referencia a ellos en la guía para inducir al estudiante/usuario a asimilar los conceptos rápidamente.

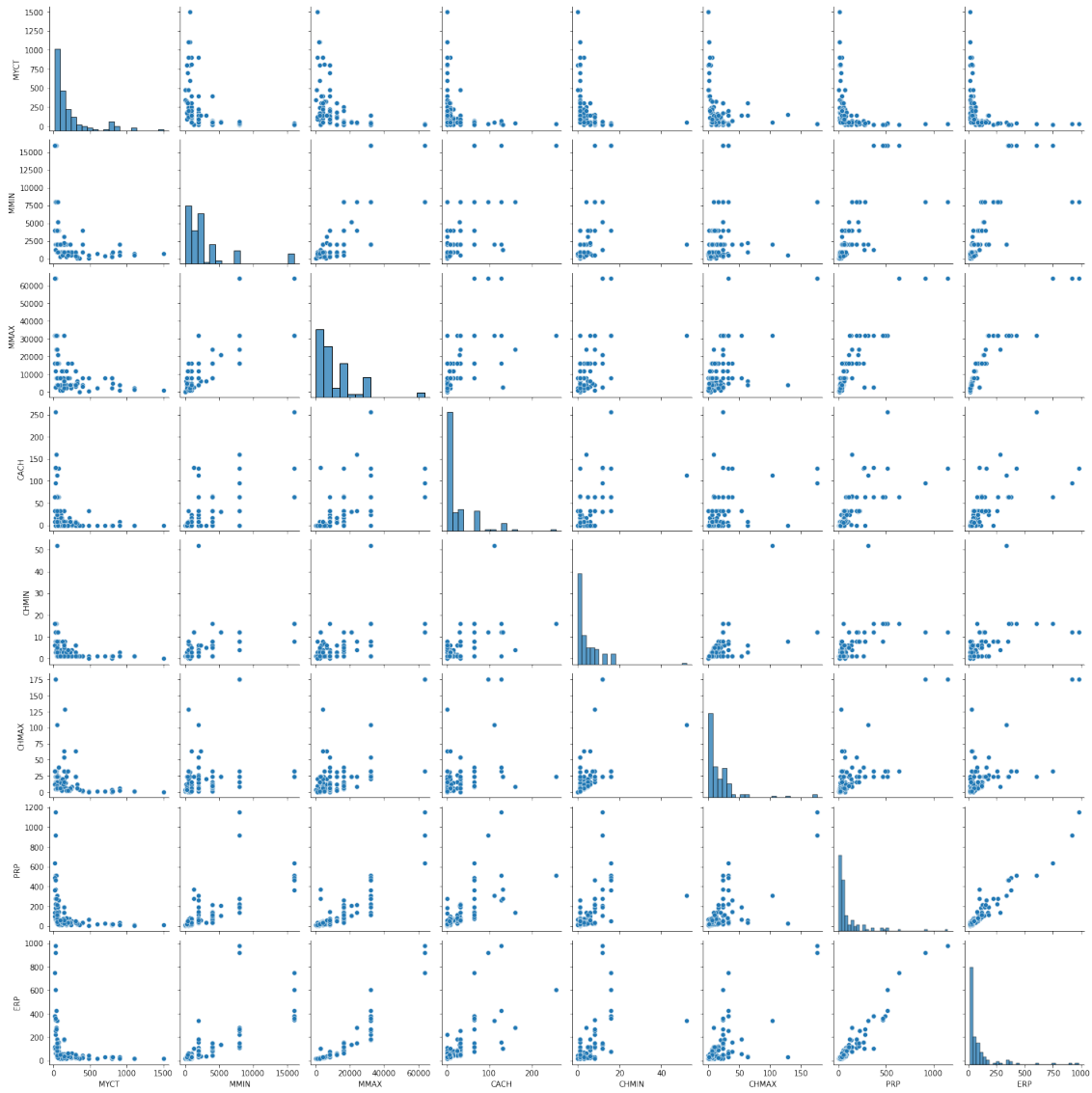

Figura 4: Gráfica que relaciona todas las variables e histograma de cada variable

1.3 Preparación de los Datos

Se mencionan los tipos de variable que se pueden encontrar en las bases de datos, y las respectivas transformaciones que se podrían llegar a realizar mediante el uso de diccionarios y la función *transform()* de pandas para la generación de la base de datos ampliada con información valida que pueda ser procesada correctamente por los

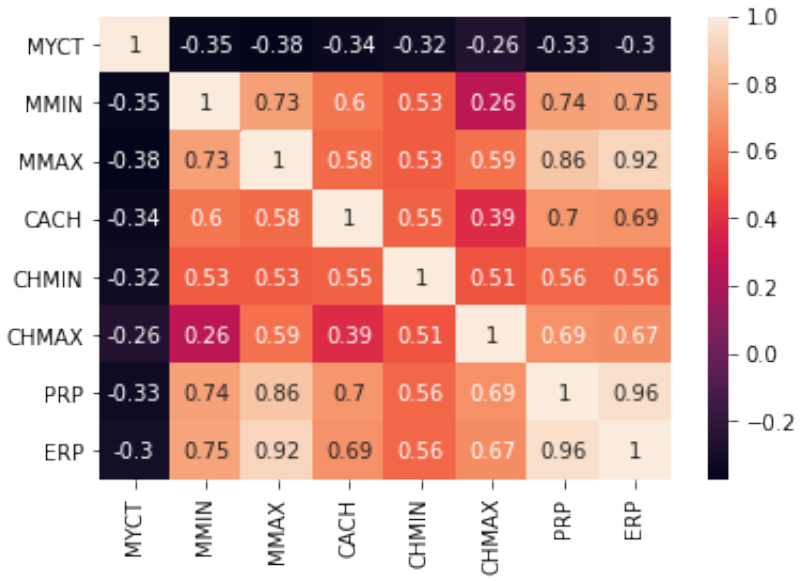

Figura 5: Gráfica para facilitar el entendimiento de la correlación de variables

algoritmos.

Adicionalmente se mencionan las diferentes técnicas para el tratamiento de datos faltantes o nulos como lo son:

- Asignar un valor constante.
- Asignar el valor promedio de la variable.
- Remover o ignorar el registro.
- Plantear una reducción de dimensiones e incluso proponer una reducción de variables.

1.4 Selección de las Bases de Datos

Para efectos de esta pasantía tomamos como fuente de datos los repositorios del *UCI Machine Learning Repository* en *Kaggle* puesto que sobre ellos se han realizado varios concursos y artículos académicos indicando que sobre esos datos son viables para experimentar y asimilar conceptos rápidamente.

1.5 Implementación de Algoritmos

Los algoritmos implementados en los ejemplos se basan en la documentación de Python asociada a la librería *Sklearn* que contiene la implementación de los algoritmos clásicos de

Machine Learning [7].

Para la implementación de redes neuronales se utilizo la librería *TensorFlow*. Ya que es una tecnología desarrollada por Google para el procesamiento de datos utilizando redes neuronales haciendo uso de la tecnología Keras para la definición de las capas que componen la red; permitiendo la consolidación de un marco de trabajo claro, sencillo, portable y con el potencial para desarrollar modelos a gran escala [8].

2 IMPACTO

Este proyecto es un producto mínimo viable que tiene el potencial para ser no solo un material de apoyo, sino además un espacio para que el publico en general con interés en desarrollar sus habilidades en el proceso de *Machine Learning* haga sus pruebas, desarrolle sus practicas y proyectos de investigación como lo son el desarrollo de artículos científicos, tesis de grado, entre otros.

Adicionalmente con este proyecto se muestra como crear artículos reproducibles de forma sencilla y transparente; puesto que en ocasiones cuando un investigador requiere interactuar con un trabajo previo se encuentra con dificultades para replicar la prueba, por lo tanto la realización de artículos con este enfoque contribuyen a la sinergia y desarrollo de la sociedad del conocimiento.

3 CONCLUSIONES

- Según la naturaleza de los datos y teniendo en cuenta el objeto de estudio es necesario evaluar dichos datos desde diferentes puntos de vista (incluyendo la implementaciíon de uno o múltiples algoritmos), ya que de esa forma se evitan sesgos y se obtienen buenos resultados.
- La construcción del repositorio y el material alojado en el sitio web esta diseñada para guiar al usuario (estudiante, profesor, investigador, pasante, tesita) en los diferentes temas que debe conocer para comprender de forma adecuada la información asociada a las prácticas que contienen la introducción al uso de los diferentes algoritmos.
- Al interactuar con tecnologías como los repositorios de Git almacenados en *Github*, los *Notebook* que permite crear *Google Colabs* y la simplicidad de *Python*. El esfuerzo es para escribir artículos reproducibles de forma sencilla, elegante y robusta es mínimo; permitiendo al investigador enfocarse en los elementos que realmente competen al desarrollo de su proyecto de investigación.

4 PROBLEMAS IDENTIFICADOS Y POSIBLES TRABAJOS FUTUROS

La tecnología va en cambio constante y evoluciona todos los días así que siempre vendrán retos nuevos y cosas muy interesantes por hacer; y para entrar un poco en materia de las cosas que se podría pensar que son factibles para mejorar, crear o profundizar se pueden encontrar:

1. La iniciativa del grupo de investigación ***LASER*** no solo tiene un carácter social interesante, sino que además pretende abrir puertas para la introducción de métodos alternativos apoyados en herramientas técnologicas para transmitir el conocimiento. En ese orden de ideas podría ser un trabajo futuro implementar mas ejemplos con otros algoritmos, habilitar una sección para trabajos de grado donde futuros investigadores, pasantes, o tesistas incluyan los conjuntos de datos, algoritmos utilizados y como los abordaron en el desarrollo de su proyecto de investigación.
2. En el caso de las redes neuronales, TensorFlow es una tecnología robusta que tiene soporte para Python, Javascript, a nivel de dispositivos móviles y de IoT como lo menciona la documentación oficial [8]. Entonces podría pensarse en realizar captura de datos, procesamiento, análisis, y toma de decisiones en tiempo real utilizando cualquier dispositivo que tenga un sensor y una conexión a Internet.

REFERENCIAS

[1] G. de Investigación LASER, "Laser," *Sitio web consultado el 01 de septiembre de 2020.* [Online]. Available: https://comunidad.udistrital.edu.co/laser/

[2] Google, "Colaboratory," *Sitio web consultado el 01 de septiembre de 2020.* [Online]. Available: https://research.google.com/colaboratory/faq.html

[3] M. Ismail P, "How to use google colab with github via google drive," *Sitio web consultado el 24 de junio de 2021.* [Online]. Available: https://medium.com/analytics-vidhya/how-to-use-google-colab-with-github-via-google-drive-68efb23a42d

[4] Jupyter, "Jupyter," *Sitio web consultado el 01 de septiembre de 2020.* [Online]. Available: https://jupyter.org

[5] P. Gupta, *Practical Data Science with Jupyter: Explore Data Cleaning, Pre-processing, Data Wrangling, Feature Engineering and Machine Learning using Python and Jupyter*. BPB Publications, 2021, *Sitio web consultado el 24 de junio de 2021.* [Online]. Available: https://books.google.com.co/books?id=EaMgEAAAQBAJ&pg=PP20&dq=top+programming+languages+for+data+science+2021&hl=es-419&sa=X&ved=2ahUKEwjf3fvgprLxAhX8RDABHUgxAuIQ6AEwAXoECAQQAg#v=onepage&q=top%20programming%20languages%20for%20data%20science%202021&f=false

[6] A. Cuevas Alvarez, *Programar con Python 3*. RA-MA Editorial, 2016.

[7] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[8] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng, "TensorFlow: Large-scale machine learning on heterogeneous systems," 2015, software available from tensorflow.org. [Online]. Available: https://www.tensorflow.org/