



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

ESCUELA NACIONAL DE ESTUDIOS SUPERIORES
UNIDAD MORELIA

MANUAL DE PRÁCTICAS PARA LA MEJORA DE LA
ENSEÑANZA DEL APRENDIZAJE MÁQUINA APLICADO
A LA CIENCIA DE DATOS A GRAN ESCALA

INFORME FINAL

QUE PARA OBTENER EL TÍTULO DE

LICENCIADA EN TECNOLOGÍAS PARA
LA INFORMACIÓN EN CIENCIAS

P R E S E N T A

MARIANA MICHELL FLORES MONROY

TUTOR

DR. SERGIO ROGELIO TINOCO MARTÍNEZ

CO-TUTOR

DR. HEBERTO FERREIRA MEDINA

MORELIA, MICHOACÁN FEBRERO 2023



Agradecimientos institucionales

Le agradezco principalmente a la Escuela Nacional de Estudios Superiores Unidad Morelia, a los institutos de investigación que forman parte de la UNAM campus Morelia y a la Universidad Nacional Autónoma de México, por haberme dado la oportunidad de adquirir las habilidades y conocimientos necesarios para desarrollarme en los ámbitos profesional, ético, académico y laboral.

Mi mayor y más profundo agradecimiento a todo el cuerpo docente de la Licenciatura en Tecnologías para la Información en Ciencias, por su tiempo, su paciencia y por todas sus enseñanzas tanto dentro como fuera del ámbito académico, especialmente a las Dras. Marisol Flores Garrindo y Adriana Menchaca Méndez por ayudar a no rendirme y siempre brindarme su apoyo en cada etapa de mi trayectoria universitaria.

Mi más sincero agradecimiento a las académicas y los académicos que tan amablemente aceptaron participar en la mesa sinodal.

Agradezco al proyecto **PAPIME PE106021** que me ayudó económicamente durante el desarrollo de este trabajo ya que, sin este apoyo, me hubiera sido imposible concluirlo.

Agradezco infinitamente al Dr. Sergio Rogelio Tinoco Martínez por su tiempo, comprensión, guía y paciencia durante este proyecto, por haber fungido como mi asesor y por sus observaciones.

De la misma manera agradezco muchísimo al Dr. Heberto Ferreira Medina por haber compartido su experiencia y conocimientos conmigo para poder aplicarlos en este proyecto, así como haber fungido como co-asesor en este trabajo.

Agradecimientos personales

Primero quiero agradecer a mi familia por el infinito sacrificio que hicieron todos y cada uno de ellos para que yo pueda llegar hasta aquí. Agradezco a mi mamá y a mi papá quienes, aunque no entendían bien de qué trata mi carrera, no dejaron de creer en mí.

A mis hermanos Jersain, Abigail y Monse por motivarme a seguir adelante y cumplir mis objetivos.

A mis mascotas, que a pesar de no entender qué pasa, me han motivado a ser mejor y a esforzarme cada día.

También agradezco a mis compañeros de clase que hicieron mi trayectoria universitaria más divertida y productiva de lo que podía esperar, pero en especial agradecer a Bruce que sin su amor y ayuda no hubiera podido concluir este proyecto.

Agradezco a mi amiga y compañera Ruth, que a pesar de haber tomado un camino diferente su influencia sigue presente en mí.

A mi amigo de toda la vida, Eric quien siempre ha creído en mí y en mi potencial.

A mi asesor, el Dr. Sergio Tinoco y a mi cos-asesor, el Dr. Heberto Ferreria por facilitar todos los recursos necesarios para llevar a cabo este proyecto.

Resumen

Este trabajo forma parte del proyecto PAPIME titulado “Propuesta de mejora a la enseñanza del aprendizaje automático aplicado a la Ciencia de Datos a gran escala”, con el número de proyecto PE106021. Se propone la elaboración de un manual de prácticas para estudiantes del nivel licenciatura, que ayude a mejorar el conocimiento del Machine Learning (ML) y su aplicación en la ciencia de datos a gran escala (Big Data).

El proyecto estará basado en diseñar, construir e implementar una guía de prácticas dirigida a los estudiantes a partir de sexto semestre en adelante de la Licenciatura en Tecnologías para la Información en Ciencias (LTICs) o de otras licenciaturas de la ENES Morelia que cuenten con los conocimientos básicos del ML.

Para realizar dicho proyecto, se tomará en cuenta la opinión de alumnos y docentes de las diferentes licenciaturas, dentro de la ENES Morelia, con respecto a cuáles temas son los que consideran de mayor importancia y que se deben impartir dentro de las materias que utilizan el aprendizaje automático dentro del plan de estudios de la LTICS.

El objetivo de este proyecto es el de mejorar la formación académica de los estudiantes dentro de la LTICS, así como mejorar la calidad de la enseñanza de este tema por parte de los docentes.

Abstract

This work is part of the PAPIME project called “Proposal to improve the teaching of machine learning applied to large-scale Data Science”, with PE106021 project number. The development of a computing laboratory manual for undergraduate students is proposed in order to improve the knowledge acquisition of Machine Learning (ML) and its application onto a large scale Data Science (Big Data).

The project is focused on designing, building and implementing a manual of computer laboratory practices aimed at students from the sixth semester onwards of the Bachelor of Information Technology in Sciences (LTICs) and/or other degrees from ENES Morelia with basic knowledge of the ML.

To carry out this project, opinion of students and teachers of different degrees, within the ENES Morelia, will be taken into account with respect to which topics are the ones that they consider most important and that should be taught within subjects based on ML of the LTICs curriculum.

The objective of this project is to improve the academic training of students, within the LTICs, as well as to improve the quality of teaching of this knowledge area by the academic staff.

Índice general

Agradecimientos institucionales	I
Agradecimientos personales	II
Resumen	III
Abstract	IV
1. Introducción	1
1.1. Justificación	2
1.2. Hipótesis	3
1.3. Objetivo	3
1.3.1. Objetivo General	3
1.3.2. Objetivos Particulares	3
1.4. Descripción general	5
2. Antecedentes	6
2.1. Tipos de Machine Learning	7
2.2. Uso del ML en la actualidad	12
3. Algoritmos de Machine Learning	14
3.1. Árbol de decisión	14
3.1.1. CHi-squared Automatic Interaction Detector (CHAID)	15
3.1.2. Classification and regression tree (CART)	16
3.1.3. C4.5	16

3.2. Modelos de Regresión	17
3.2.1. Regresión lineal	18
3.2.2. Regresión logística	18
3.3. k -vecinos más cercanos (KNN)	20
3.4. Clustering (K-means)	22
3.5. Dask	25
3.6. Sistema de archivos HDFS	26
4. Métricas para Evaluar Modelos	28
4.1. RMSE	28
4.2. MAE	29
4.3. Matriz de confusión	30
4.4. Exactitud	31
4.5. Precisión	31
4.6. Recall	33
4.7. $F\beta$	33
4.8. F1 score	34
5. Metodología de las Prácticas	35
6. Resultados y Discusión	37
Apéndices	42
.1. Prácticas	43
.2. Datos	43
.3. Artículo	44

Índice de figuras

1.1. Mapa mental del desarrollo del proyecto.	4
2.1. Diagrama del aprendizaje supervisado.	8
2.2. Un ejemplo de aprendizaje supervisado usado para clasificación de spam.	10
2.3. Un ejemplo de aprendizaje no supervisado usado para clustering, ba- sado en los atributos de los datos.	11
2.4. Diagrama de un algoritmo de aprendizaje por refuerzo.	12
2.5. Diagrama de los campos de estudio dentro de la IA.	13
3.1. Ejemplo de una recta calculada con regresión lineal donde $y = 2.2816 + 0.3464x$	19
3.2. Función Sigmoide aplicada a la regresión lineal	20
3.3. Nuevo punto a clasificar	20
3.4. Vecinos más cercanos (con menor distancia)	21
3.5. Vecinos más cercanos (con $k = 3$)	22
3.6. $k = 3$ centroides en un conjunto de datos	23
3.7. Buscando el centroide más cercano	24
3.8. etiquetando el punto actual de acuerdo a su grupo	24
3.9. k grupos formados	25
3.10. Sintaxis de Pandas (arriba) Sintaxis de Dask (abajo)	26
3.11. Sintaxis de Numpy (izq) Sintaxis de Dask (der)	26
3.12. Diagrama de un HDFS	27
4.1. Matriz de confusión simple de 2 valores (positivo/negativo)	30
4.2. Representación gráfica de la exactitud vs precisión	32

6.1. Matriz de Correlación de los resultados obtenidos en la encuesta . . .	38
6.2. Nivel de importancia de acuerdo a los encuestados	38
6.3. Niveles de desconocimiento	39
6.4. Resultados del Módulo I	40

Capítulo 1

Introducción

El uso de herramientas que permiten el análisis de grandes volúmenes de datos ha permitido que las ciencias exactas jueguen un papel importante para la toma de decisiones en las organizaciones. En la Licenciatura en Tecnologías para la Información en Ciencias (LTICs), de la Escuela Nacional de Estudios Superiores Unidad Morelia (ENES Morelia), perteneciente a la Universidad Nacional Autónoma de México (UNAM) existen asignaturas relacionadas con la Ciencia de Datos que se incluyen en el plan de estudios a partir del sexto semestre, conocidas como asignaturas del área de profundización y que representan un reto para los estudiantes a la hora de tratar de poner en práctica la teoría aprendida, además de carecer de las herramientas para su aplicación en problemas reales.

En virtud de lo anterior, se observa la necesidad de que docentes y estudiantes de la LTICs conozcan nuevas fronteras en Inteligencia Artificial (IA), específicamente en la aplicación de modelos matemáticos del aprendizaje automático (*ML – Machine Learning*).

El ML es la rama de la IA que se encarga de desarrollar técnicas, algoritmos y programas que brindan a las computadoras la capacidad de aprender. Una máquina aprende cada vez que cambia su estructura, programas o datos, en función de la entrada o en respuesta a información externa, de tal manera que mejora su rendimiento en el futuro.

Por otro lado, el Internet de las Cosas (*IoT – Internet of Things*) y la industria

4.0 han requerido la introducción de dispositivos autónomos e *inteligentes*, además del uso de maquinaria en el sector industrial.

En el sector bancario, se tiene el caso de bancos en los cuales han integrado un asistente de chat virtual vía WhatsApp. Su objetivo es facilitar la interacción entre los usuarios y el banco para dar respuesta rápida acerca de la localización de sucursales, abrir una cuenta, etc.

El asistente virtual del banco procesa texto y voz para mejorar la interacción con los clientes, hace uso de datos y del ML para procesar la información que convierte. El agente inteligente utiliza algoritmos de IA para entender y aprender los requerimientos y consultas de los clientes, lo que le permite ampliar sus capacidades para futuras consultas.

La principal contribución de este trabajo es presentar una propuesta para mejorar el aprendizaje de los temas de ML y Big Data con capacitación práctica enfocada en casos de uso de la vida real.

1.1. Justificación

En la actualidad existen diferentes métodos para el análisis de grandes volúmenes de datos, haciendo que las ciencias de la información tomen un papel relevante en nuestra sociedad. Debido a su importancia, dentro de la LTICs de la ENES Morelia existen materias orientadas a la ciencia de datos, en especial a los métodos del Machine Learning. Estas materias, al igual que las técnicas y herramientas que se utilizan en el ML, son de alta importancia para el estudiante ya que forman la base que se requiere para materias más complejas, como redes neuronales.

Así también, actualmente en la LTICs las asignaturas del área del ML se imparten de manera teórica y práctica, especialmente debido al cambio de paradigma motivado por la pandemia médica del SARS-CoV-2. No obstante lo anterior y debido a la complejidad de los temas abordados, el rendimiento estudiantil no es el ideal. Aunado a lo antes mencionado, la aplicación de los métodos del aprendizaje automático sobre grandes volúmenes de datos no es considerado dentro de los temarios de las

diferentes asignaturas en el área. Por todo lo anterior, una práctica complementaria del alumnado, aplicada a problemas reales sobre el Big Data, reforzará el estudio y la comprensión de estos temas difíciles.

1.2. Hipótesis

En la Licenciatura en Tecnologías para la Información en Ciencias (ENES Morelia, UNAM) existe una cantidad considerable de estudiantes cuyo desempeño en las asignaturas del área del ML ha sido bajo debido a la complejidad de sus temas. Con el uso del manual de prácticas se espera que su desempeño mejore después de la intervención de este proyecto.

1.3. Objetivo

1.3.1. Objetivo General

Desarrollar un manual de prácticas para la enseñanza del Machine Learning, aplicado a volúmenes de datos a gran escala, dirigido a estudiantes a partir de sexto semestre de la Licenciatura en Tecnologías para la Información en Ciencias o de otras licenciaturas de la ENES Morelia que cuenten con los conocimientos básicos del Machine Learning.

1.3.2. Objetivos Particulares

1. Desarrollar una encuesta para diagnosticar los conocimientos previos e intereses del alumnado.
2. Establecer los temas que se van a cubrir en el manual de prácticas, relacionados al ML y con fundamento en la encuesta aplicada.
3. Elaborar el marco teórico del manual de prácticas con base en los temas seleccionados.

4. Determinar los ejemplos prácticos del ML que se abordarán en el manual de prácticas, conciliando con los docentes de la LTICs de las asignaturas del área del ML, la pertinencia de las prácticas propuestas enfocadas al Big Data.
5. Implementar los ejemplos prácticos usando el lenguaje Python.
6. Realizar una prueba piloto del manual de prácticas.
7. Realizar el diagnóstico de los resultados de la intervención a través de una encuesta de salida.
8. Publicar los resultados en la página web del proyecto.

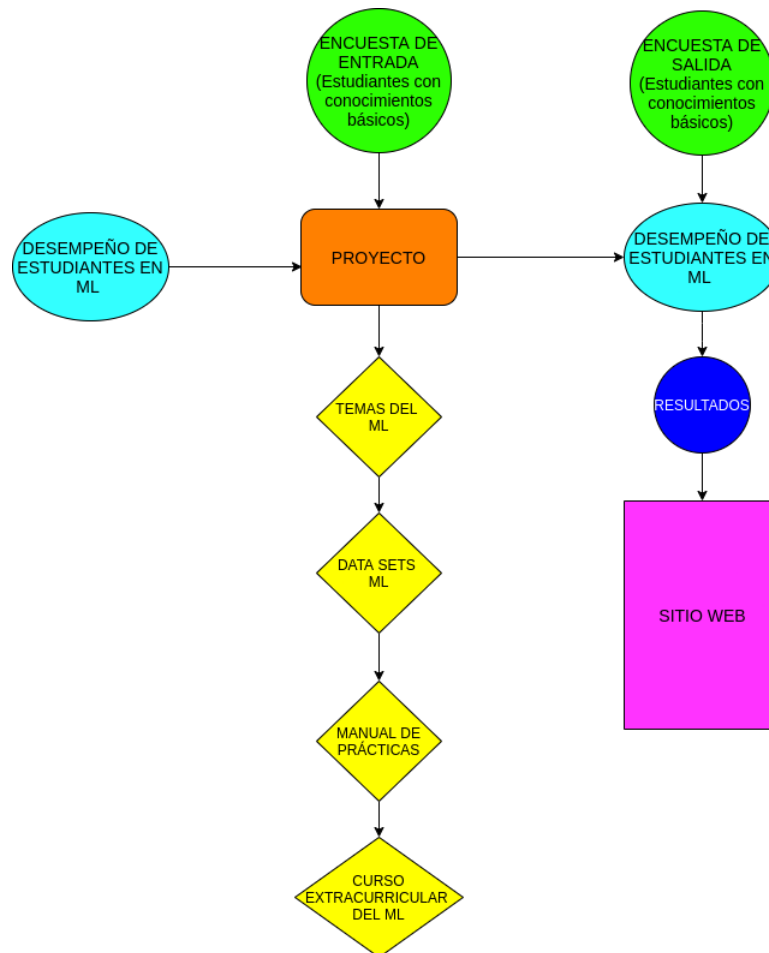


Figura 1.1: Mapa mental del desarrollo del proyecto.

1.4. Descripción general

Este documento está organizado de la siguiente manera:

- El Capítulo 2 es una recopilación de los antecedentes más relevantes que existen sobre la inteligencia artificial y el Machine Learning. Dentro de este capítulo también se abarcan temas como los tipos más comunes de ML, una breve descripción de sus algoritmos más populares así como su uso en la actualidad en diferentes áreas del conocimiento.
- En el Capítulo 3 se detalla de forma más técnica y teórica sobre el funcionamiento de los algoritmos que se usarán en las prácticas (Capítulo 5). En éste también se abarca la explicación sobre qué es y cómo funciona la librería Dask de Python, además de la aplicación y funcionamiento de los sistemas de archivos distribuidos (como el *HDFS – Hadoop File System*).
- El Capítulo 4 habla sobre las principales métricas de evaluación para modelos de ML. En éste se explica la razón de cada métrica, además de mostrar las formulas para calcularlas y dar la interpretación de las mismas dados los valores que pueden tomar.
- El Capítulo 5 enlista y da un resumen de la metodología de las prácticas. En éste se explica como se lleva a cabo la estructura de cada práctica, los datos que se van a usar, la métrica de evaluación a considerar, además de explicar cuál es el objetivo de cada una de ellas.
- Finalmente, en el Capítulo 6 se hace una recopilación de los datos obtenidos después de haber impartido el primer módulo del diplomado enfocado en el ML. Se muestran los resultados relacionados a la mejora del aprendizaje en ML en estudiantes de la LTICs y académicos con formación afín a éste.

Capítulo 2

Antecedentes

Según Kaplan (2016) se le llama Inteligencia Artificial (IA) a la ciencia que se encarga de crear sistemas inteligentes que sean capaces de imitar el comportamiento inteligente de los humanos para la toma de decisiones y lograr metas.

Haenlein y Kaplan (2019) describen que en el año 1942 se creía que la inteligencia artificial era un fenómeno futurista gracias al escritor Isaac Asimov y su obra *Runaround*.

Unos años después, el británico Alan Turing (1950) publicó su artículo “*Computing Machinery and Intelligence*” donde describe por primera vez cómo crear inteligencia artificial mediante las máquinas de Turing, además de explicar cómo se realiza la prueba de Turing para diferenciar IA de la inteligencia humana.

Posteriormente Mitchell (1997) establece que la IA se utiliza para resolver problemas complejos que la programación convencional no puede.

En su artículo, Jordan y Mitchell (2015), explican que dentro de la IA existe una rama llamada *aprendizaje automático* (*ML – Machine Learning*) cuyo fin es mejorar el rendimiento de los algoritmos a través de su experiencia. Esta técnica hace uso de herramientas como la estadística, la informática, las matemáticas y las ciencias computacionales, teniendo su fundamento en el análisis de los datos.

Por otra parte, la definición de Mitchell y cols. (1997) del ML es la siguiente:

Se dice que un programa de computadora aprende de una experien-

cia E , con respecto a una tarea T y una medida de desempeño P , si su desempeño con respecto a T , medido por P , mejora con la experiencia E .

Lee y cols. (2017) hacen mención a Samuel (1959), pionero en el estudio del ML, quien lo definió de la siguiente manera:

El aprendizaje automático es el campo de estudio que le da a la computadora la habilidad de aprender, sin que esté explícitamente programada.

El término Machine Learning se acuñó oficialmente alrededor del año 1960, según lo relatado por Liu (2020). Este nombre consiste en la palabra “Machine”, que hace alusión a cualquier dispositivo (robot, computadora, ...) y la palabra “Learning”, que hace referencia a la capacidad que se tiene de adquirir o descubrir patrones.

En la actualidad Mohri y cols. (2018) consideran al ML como la técnica de crear sistemas que sean capaces de aprender por sí mismos, utilizando grandes volúmenes de datos, haciendo que éstos sean aptos para realizar análisis y, con ello, poder predecir futuros comportamientos.

2.1. Tipos de Machine Learning

El ML ha ganado importancia en las últimas décadas debido a su habilidad de realizar predicciones a partir de un conjunto de datos. Murdoch y cols. (2019) mencionan que los diferentes modelos de ML tienen la capacidad de adquirir conocimiento, relacionando características contenidas en los datos. A esto se le conoce comúnmente como “interpretaciones”.

Géron (2019) explica que existen diferentes enfoques para el diseño de un sistema de ML. Tales enfoques se dividen en:

- Si están entrenados bajo supervisión humana o no. A los cuales se les denominan: **supervisado, no supervisado y por refuerzo.**
- Si pueden aprender sobre la marcha o no, denominados como **aprendizaje en línea.**

- Si detectan patrones de entrenamiento o si comparan nuevos datos con datos ya existentes. Este tipo de ML se cataloga como **aprendizaje basado en instancias o aprendizaje basado en modelos**.

Aprendizaje Supervisado

El aprendizaje supervisado suele usarse cuando se cuenta con datos de los cuales ya se sabe la respuesta que se desea predecir. Cunningham y cols. (2008) explican que este aprendizaje consiste en que el sistema pueda mapear entre los datos de entrada (*input*) y sus respectivas etiquetas (*output*) para después predecir las etiquetas dados nuevos datos no etiquetados, como se muestra en la Figura 2.1.

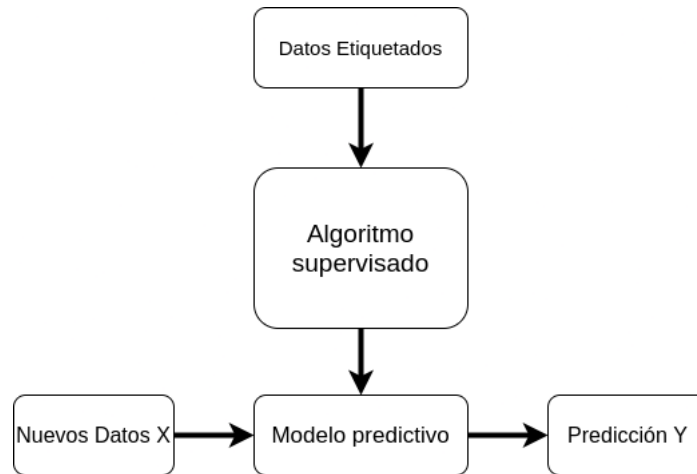


Figura 2.1: Diagrama del aprendizaje supervisado.

Según El Naqa y Murphy (2015) el principal objetivo es que el sistema aprenda a distinguir las características de una etiqueta de otra.

De acuerdo con Ayodele (2010) el aprendizaje supervisado tiene la tarea de resolver los siguientes problemas, los cuales no pueden resolverse con programación simple:

- **Regresión.**

Jagielski y cols. (2018) definen la regresión como un método en el cual se hace uso de variables numéricas, para realizar predicciones, las cuales se espera que cada vez tengan menor margen de error. Montgomery y cols. (2021) señalan que estas variables se estudian para encontrar correlación entre ellas y sus respectivas

etiquetas, para realizar predicciones de acuerdo a los patrones encontrados. Existen dos tipos principales de regresión: lineal y logística.

■ **Clasificación.**

Li y cols. (2001) indican que la clasificación también se usa para hacer predicciones utilizando un conjunto de datos etiquetados pero, a diferencia de la regresión, la clasificación realiza predicciones discretas (llamadas clases).

Para complementar lo anterior, Kotsiantis y cols. (2007) mencionan los siguientes algoritmos que se usan para clasificación (entre otros):

- Árboles de decisión.

Este es un método muy utilizado debido a que es un algoritmo simple, fácil de comprender y porque no requiere de parámetros. Su y Zhang (2006) explican que el funcionamiento de los árboles de decisión consiste en un algoritmo recursivo en el que en cada iteración se escoge el atributo cuyo valor es más adecuado para dividir el conjunto de datos, hasta que todos los datos sean clasificados.

- k -vecinos más cercanos.

Song y cols. (2007) afirman que este algoritmo tiene la tarea de predecir la etiqueta de un dato (x_0) dados los k datos más cercanos, es decir, aquéllos con menor distancia (euclidiana, distancia del coseno, etc.). Una vez que se tienen los k vecinos más cercanos, se revisan sus etiquetas y se le asigna a x_0 la etiqueta más repetida.

- Redes neuronales artificiales (RNA).

Wang (2003) define a las RNA como un modelo que consiste en una capa de neuronas de entrada, algunas capas de neuronas ocultas y una capa de neuronas de salida. Cada conexión entre capas está asociada a un valor numérico (*peso*). También cuentan con funciones de activación, siendo la más común la función sigmoide.

Una de las aplicaciones de la clasificación, por ejemplo, es la detección de correo

electrónico *spam* (correo no deseado), como se ve en la Figura 2.2.



Figura 2.2: Un ejemplo de aprendizaje supervisado usado para clasificación de spam.

Aprendizaje No Supervisado

En el caso del aprendizaje no supervisado los datos no están etiquetados, esto hace que el sistema tenga que aprender por sí mismo, sin que se le indique si la clasificación es correcta o no, según señalan Raschka y Mirjalili (2019).

El aprendizaje no supervisado, según Sathya y Abraham (2013), tiene la habilidad de aprender y organizar información, detectando patrones.

Para lograr clasificar los datos se utiliza una técnica de agrupamiento, mejor conocida como *clustering*. Dayan y cols. (1999) proponen que el objetivo del clustering es agrupar datos cuyas características sean similares entre sí, como se ve en la Figura 2.3.

Para implementar el clustering, Celebi y Aydin (2016) mencionan estas técnicas de aprendizaje no supervisado (entre otras):

- *k*-medias.

Na y cols. (2010) explican que este algoritmo consiste en seleccionar aleatoriamente k centros, después calcular la distancia euclidiana (u otra métrica de distancia) de los demás datos para determinar cuál de los k centros es el más cercano y de esa forma clasificarlo en uno de los k grupos.

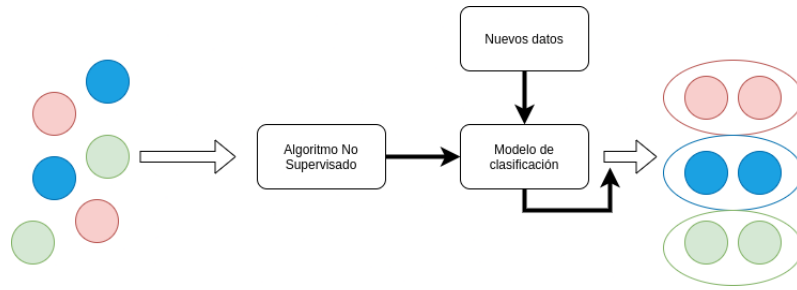


Figura 2.3: Un ejemplo de aprendizaje no supervisado usado para clustering, basado en los atributos de los datos.

- Visualización y Reducción de Dimensiones.

Vlachos y cols. (2002) mencionan que la reducción de dimensiones consiste en que un conjunto de datos reduzca su dimensionalidad sin la pérdida de información, para esto es común usar técnicas como Análisis de Componentes Principales (PCA en inglés) para que el conjunto de datos sea más fácil de procesar y de visualizar.

- Reglas de Asociación.

De acuerdo con Aher y Lobo (2012), las reglas de asociación se usan comúnmente en minería de datos para encontrar de forma eficiente patrones o correlación en un gran conjunto de datos, para posteriormente obtener información de éstos.

Aprendizaje Por Refuerzo

Wiering y Van Otterlo (2012) mencionan que el aprendizaje por refuerzo tiene como objetivo que el sistema aprenda en un entorno en el cual la única retroalimentación consiste en una recompensa escalar, la cual puede ser positiva o negativa (*castigo*).

La definición de Kaelbling y cols. (1996) es que el modelo recibe en cada iteración una recompensa r y el estado actual del entorno s , después el modelo toma una acción a de acuerdo con las entradas y eso es lo que se considera como la salida, la cual cambiará el estado s en la siguiente iteración. En la Figura 2.4 se puede ver, de manera muy general, el comportamiento del aprendizaje por refuerzo.

En los últimos años, este algoritmo ha ganado terreno en el campo de investigación debido a sus aplicaciones mencionadas en Sutton y Barto (2018) tales como:

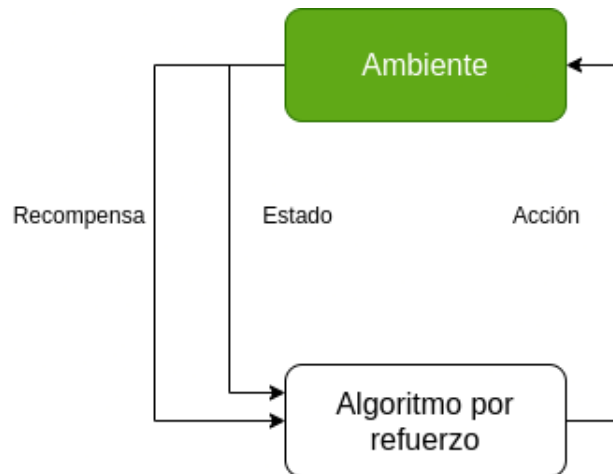


Figura 2.4: Diagrama de un algoritmo de aprendizaje por refuerzo.

- Algoritmos que juegan ajedrez (Alpha Zero).
- Controlador adaptable de parámetros.
- Toma de decisiones.
- *Felipe prepara su desayuno*, el cual es un proceso de subtareas (como abrir el refrigerador, caminar a la estufa, romper un huevo, etc.) para lograr una tarea grande (preparar el desayuno).

2.2. Uso del ML en la actualidad

En nuestros días, una de las principales aplicaciones del ML es usar métodos supervisados para el análisis de grandes volúmenes de datos para obtener información sobre ellos. Por ejemplo, en van Zoonen y Toni (2016), se muestra el caso de análisis de texto en redes sociales para entender la interacción entre usuarios.

Por otro lado, Siau y Wang (2018) apuntan la utilidad del aprendizaje supervisado en el campo de la computación y las matemáticas, como el desarrollo de Google DeepMind y Alpha Go. Además de que existen otros usos en diferentes campos de la ciencia, por ejemplo:

- En Abbasi y Goldenholz (2019) se ve que debido a la gran utilidad de los

algoritmos de ML para encontrar patrones, tienen un gran campo de aplicaciones tales como la detección de anomalías en electroencefalogramas.

- En biología, Libbrecht y Noble (2015) señalan que existen algoritmos encargados del análisis de grandes bases de datos de genomas, los cuales se entrenan para identificar potenciadores, nucleosomas, entre otras cosas.
- En medicina, Kourou y cols. (2015) indican que los algoritmos de ML se usan en la detección de tumores y su clasificación como benignos y malignos.
- Dentro de la psicología, Jiang y cols. (2020) los proponen como ayuda a la detección y predicción del riesgo de padecer algún trastorno mental.

En la Figura 2.5 se puede apreciar un diagrama a grandes rasgos de que el ML es un campo de estudio dentro de la Inteligencia Artificial.

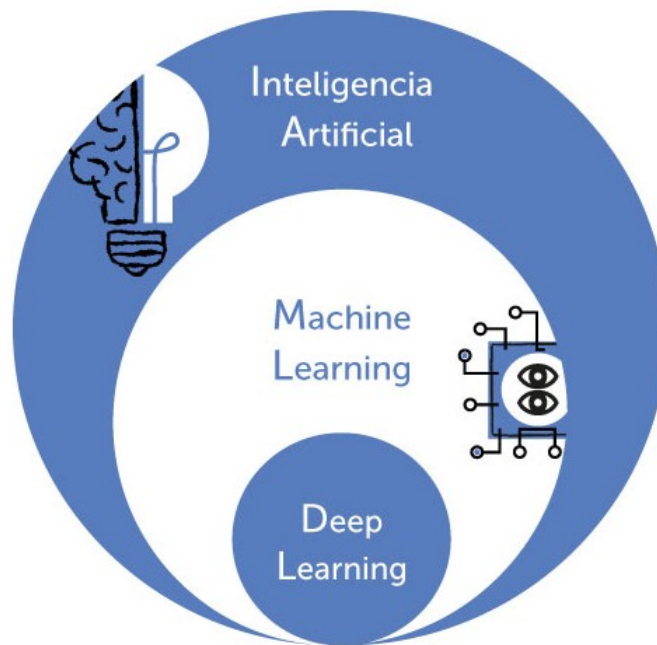


Figura 2.5: Diagrama de los campos de estudio dentro de la IA.

Capítulo 3

Algoritmos de Machine Learning

En la Sección 2.1 se habló sobre los tipos de algoritmos del ML. En esta sección se profundizará en los algoritmos que se usarán en el manual de prácticas.

3.1. Árbol de decisión

Myles y cols. (2004) definen a un árbol de decisión como un algoritmo del tipo “divide y vencerás”, usado comúnmente para hacer clasificación (aunque también puede usarse para regresión).

Su y Zhang (2006) proponen que el algoritmo inicia con un árbol vacío en el cual aún no hay nada de información acerca de los datos. Al ser un algoritmo avaricioso, éste busca cuál es el atributo que mejor divide el conjunto de datos y lo convierte en la raíz del árbol. Posteriormente el proceso se vuelve recursivo, dividiendo el conjunto de datos restante en subconjuntos que satisfacen la división de los datos.

A lo largo de los años, los investigadores han desarrollado diferentes algoritmos basados en árboles de decisión. Brijain y cols. (2014) explican que los modelos más importantes son los siguientes:

3.1.1. CHi-squared Automatic Interaction Detector (CHAID)

En su trabajo, Rodríguez y cols. (2016) mencionan que CHAID es un proceso que no hace suposiciones sobre los datos. El algoritmo determina cuál es la mejor forma de combinar las variables para predecir un resultado binario. Esto lo hace dividiendo cada variable en subconjuntos mutuamente excluyentes basados en la homogeneidad de los datos.

El criterio que se usa para determinar la división de los datos es la *prueba ji cuadrada* (χ^2). Pandis (2016) explica que esta prueba solo muestra si existe una asociación entre variables, es decir, mide qué tan dependiente es una variable de otra.

La forma de calcular χ^2 , mostrada en McHugh (2013), es la siguiente:

$$\chi^2 = \sum_i^j \frac{(O_i - E_i)^2}{E_i} \quad (3.1)$$

donde:

O = Los valores observados.

E = Frecuencia esperada.

Para calcular la frecuencia esperada (E) se emplea la fórmula siguiente:

$$E = \frac{M_R * M_C}{n} \quad (3.2)$$

donde:

M_R = Suma de la fila.

M_C = Suma de la columna.

n = Número total de datos.

La Ecuación 3.2 se aplica para cada uno de los datos y, con el resultado de cada uno de ellos, se calcula χ^2 también para cada dato. Este método ayuda mucho cuando se trata de análisis estadísticos.

3.1.2. Classification and regression tree (CART)

Según lo descrito por Loh (2011) estos modelos se obtienen mediante la división recursiva de los datos y ajustado un modelo simple de predicción en cada una de esas divisiones.

Este algoritmo usa una herramienta extra de aprendizaje, llamada "Poda". Loh (2014) explica que el método de poda es una herramienta bastante útil, ya que esta se basa en el concepto de eliminar al "eslabón más débil".

Estos valores de costo-complejidad se pueden medir usando los coeficientes de *Gini* y *Entropía*.

Gini: Este coeficiente es el más usado en árboles de clasificación, Alvaredo (2011) explica que este es más sensible a las transferencias en el centro de las distribuciones de datos.

Timofeev (2004) menciona que Gini utiliza la siguiente formula de impureza:

$$i(t) = \sum_{kl} p(k|t)p(k|l) \quad (3.3)$$

3.1.3. C4.5

El trabajo de Singh y Gupta (2014) menciona que el algoritmo C4.5 genera un árbol que divide recursivamente el conjunto de datos.

Este árbol de decisión considera todas las posibles divisiones de los datos para seleccionar la división que genere la mayor ganancia de información.

Cintra y cols. (2013) dicen que los árboles de decisión C4.5 usan la entropía y la ganancia de información como métricas para decidir cuales son los datos que mejor dividen el conjunto de datos.

Las definiciones matemáticas de la entropía y la ganancia de información, según

Ahmad y cols. (2020) es la siguiente:

$$Entropia(S) = \sum_{i=1}^n -p_i \times \log_2 p_i \quad (3.4)$$

Donde:

S es la entropía

p es la proporción de la clase (output)

$$Ganancia(S, A) = Entropia(S) \sum_{i=1}^n \frac{|S_i|}{|S|} \times Entropia(S) \quad (3.5)$$

Donde:

S es el caso a evaluar

A es un atributo a evaluar

$|S_i|$ es el caso actual

$|S|$ es el número total de casos

3.2. Modelos de Regresión

Los modelos de regresión, son definidos por Maulud y Abdulazeez (2020) como métodos matemáticos utilizados estimar la relación entre variables. Es uno de los métodos más comunes en ML para la predicción de datos.

Actualmente, la regresión es una herramienta importante que ya ayuda a los analistas y estadistas a entender las relaciones que existen entre los datos, Kumari y cols. (2018) enlista las siguientes razones por las cuales este método es importante:

- Descriptivo: Este método ayuda a analizar la fuerza que han entre las variables dependientes X y la variable independiente y .
- Ajuste: El método es capaz de ajustarse para minimizar el error.

Existen dos principales modelos de regresión, la lineal y logística.

3.2.1. Regresión lineal

La regresión lineal, a pesar de ser uno de los métodos más utilizados en ML, no es aplicable para cualquier problema, por ejemplo, no se puede usar la regresión lineal a los problemas donde las variables son categóricas.

Por lo general, las variables a predecir tienen que ser numéricas, para que la regresión lineal pueda ser aplicable exitosamente.

En Montgomery y cols. (2021) se define la fórmula de regresión lineal en \mathbb{R}^2 de la siguiente forma:

$$y = \beta_0 + \beta_1 x \quad (3.6)$$

Donde

β_0 es la intercepción de la recta

β_1 es la pendiente

x son las variables dependientes

Para \mathbb{R}^n la fórmula queda de la siguiente forma:

$$y = \beta_0 + \sum_{i=1}^n \beta_i x_i + \epsilon_i \quad (3.7)$$

Donde

ϵ es el error (distancia) entre la recta calculada y x_i

Un ejemplo simple de regresión lineal en \mathbb{R}^2 es el que se muestra en la 3.1:

3.2.2. Regresión logística

La regresión logística, a diferencia de la lineal, es usada comúnmente para realizar predicciones binarias.

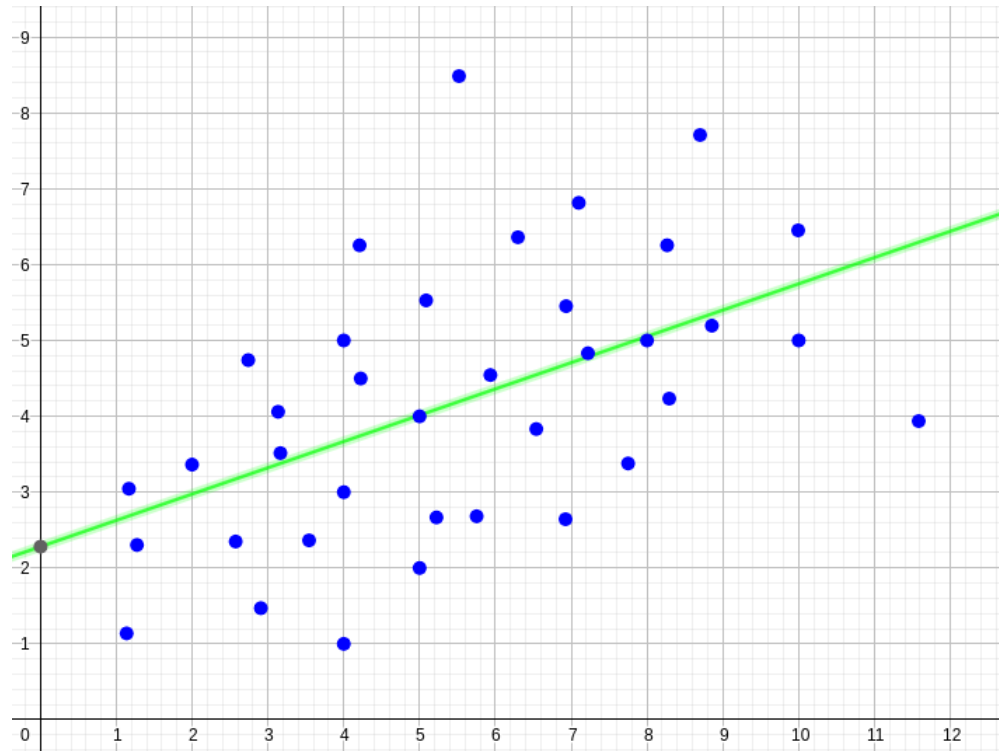


Figura 3.1: Ejemplo de una recta calculada con regresión lineal donde $y = 2.2816 + 0.3464x$

Harrell (2015) explica que el algoritmo consiste en generar la ecuación de la función sigmoide 3.8 que permita explicar la relación que existe entre las variables independientes y la variable dependiente (X y y).

$$y = \frac{1}{(1 + e^{-x})} \quad (3.8)$$

Donde:

x son todas las variables dependientes representadas como ecuaciones de una recta.

Para determinar la clasificación, el modelo de regresión logística se toma en cuenta la salida de la función de la recta (x) aplicada en la función sigmoide 3.8.

- Si la salida es ≤ 0.5 , entonces el algoritmo lo toma como 0
- Si la salida es > 0.5 , entonces el algoritmo lo toma como 1

Lo anterior se puede apreciar en la imagen 3.2

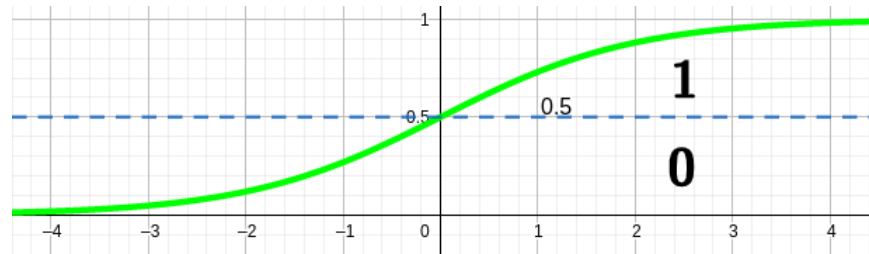


Figura 3.2: Función Sigmoide aplicada a la regresión lineal

3.3. k -vecinos más cercanos (KNN)

Como se mencionó en el capítulo 2, existen algoritmos de regresión y clasificación y uno de los algoritmos más importantes de clasificación es K -vecinos más cercanos o KNN por sus siglas en inglés (K -Nearest Neighbor).

En su trabajo, Zhang y cols. (2017) mencionan que KNN es un algoritmo supervisado que sirve para clasificar valores (y) buscando los puntos de datos "más similares" aprendidos en la etapa de entrenamiento.

El procedimiento consiste en calcular la distancia entre el "vecino" a clasificar y el resto de "vecinos" del dataset de entrenamiento. 3.3

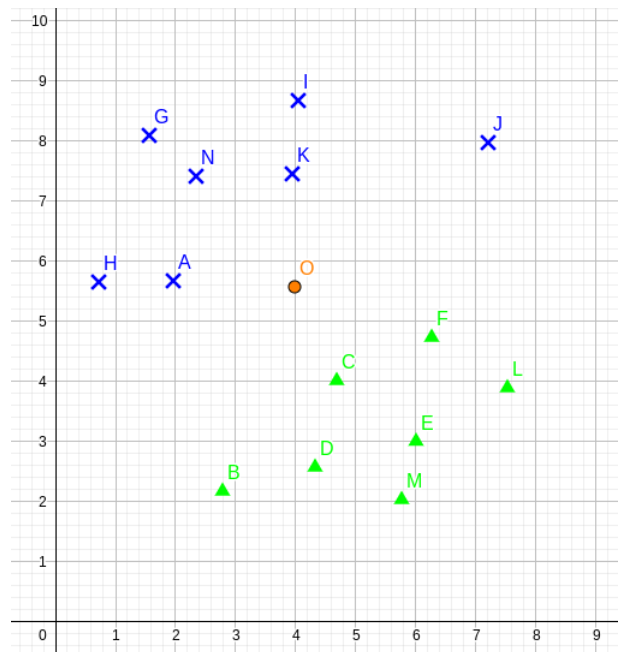


Figura 3.3: Nuevo punto a clasificar

Se seleccionan los "k vecinos" más cercanos, es decir, con menor distancia, según la función de distancia que se emplee (euclidiana, coseno, etc) 3.4.

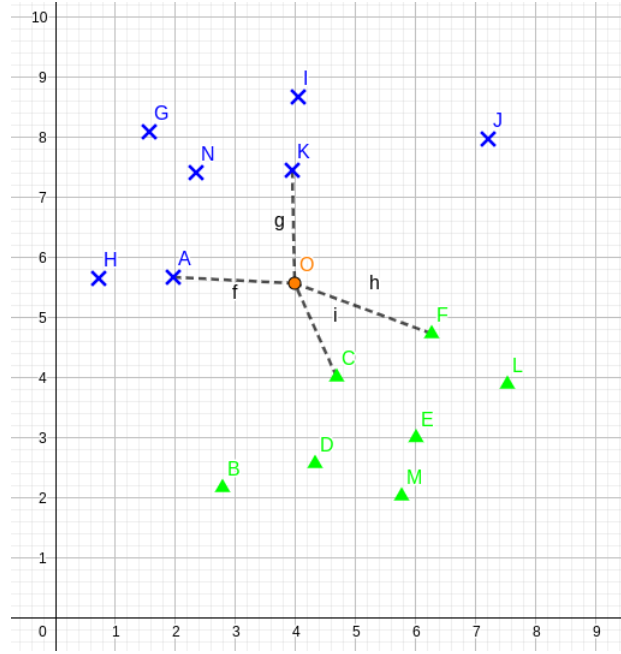


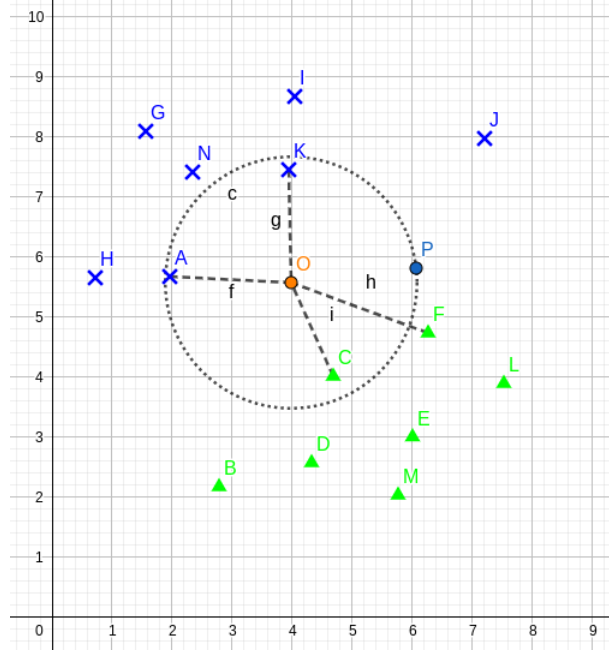
Figura 3.4: Vecinos más cercanos (con menor distancia)

En caso de usar la distancia euclidiana como métrica de distancia, Adithiyaa y cols. (2020) la definen de la siguiente manera 3.9:

$$D(x, y) = \sqrt{\sum_{i=1}^n (x_i - y_i)^2} \quad (3.9)$$

Una vez que se obtienen los puntos con menor distancia, se seleccionan los k vecinos más cercanos y la etiqueta que domine el conjunto (la etiqueta más frecuente) es la que decidirá la clasificación del punto actual.

Como se puede ver en la figura 3.5 hay 2 puntos etiquetados como **cruz** y solo uno etiquetado como **triángulo**, por lo tanto, el punto en cuestión se etiquetaría como una cruz azul.

Figura 3.5: Vecinos más cercanos (con $k = 3$)

3.4. Clustering (K-means)

K-means es un algoritmo de clasificación no supervisada (clustering) que agrupa objetos en k grupos basándose en sus características.

K-means es definido por Kanungo y cols. (2002) como un algoritmo iterativo que se encarga de buscar la solución mínima local.

Para encontrar el mínimo local, del conjunto de datos se seleccionan aleatoriamente k puntos 3.6, llamados centroides, los cuales tendrán su respectivo vecindario de n puntos.

Después, para cada punto en cada vecindario, se toma la distancia de este a su respectivo centroide (o al más cercano) 3.7. Por lo general se usa la distancia euclidiana 3.10 para obtener las distancias de cada punto al centro de su subconjunto:

$$\min_s E(\mu_i) = \min_s \sum_{i=1}^k \sum_{x_j \in s_i} \|x_j - \mu\| \quad (3.10)$$

Se consideran la menor distancia obtenida, para que el punto actual se etiquete de

acuerdo a su centroide más cercano 3.8.

Pérez y cols. (2007) mencionan los casos en que algoritmo converge, por ejemplo:

1. Cuando el algoritmo ha llegado al número de iteraciones especificado al inicio del algoritmo.
2. Cuando la diferencia entre los centroides de dos iteraciones consecutivas es más pequeño que un umbral dado.
3. Cuando no hay intercambio de elementos entre los k grupos.

Después de Que el algoritmo termina su entrenamiento, se tienen k grupos cuyos puntos comparten características 3.9

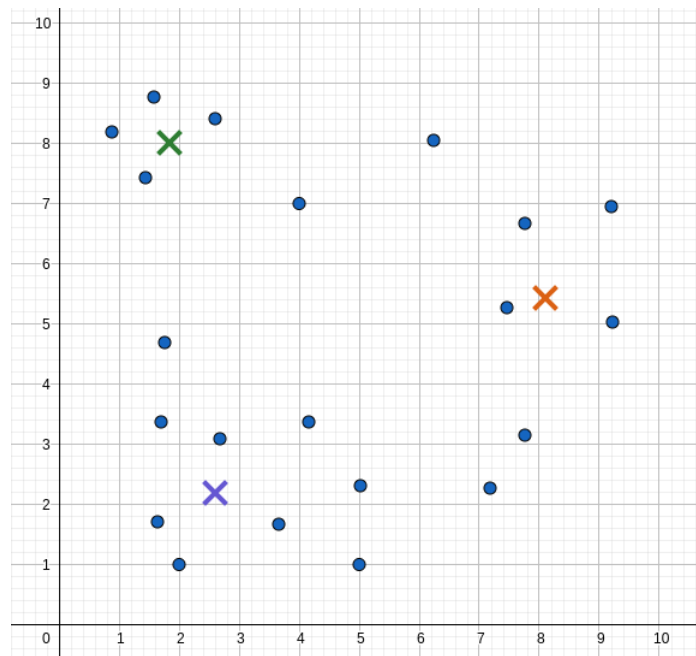


Figura 3.6: $k = 3$ centroides en un conjunto de datos

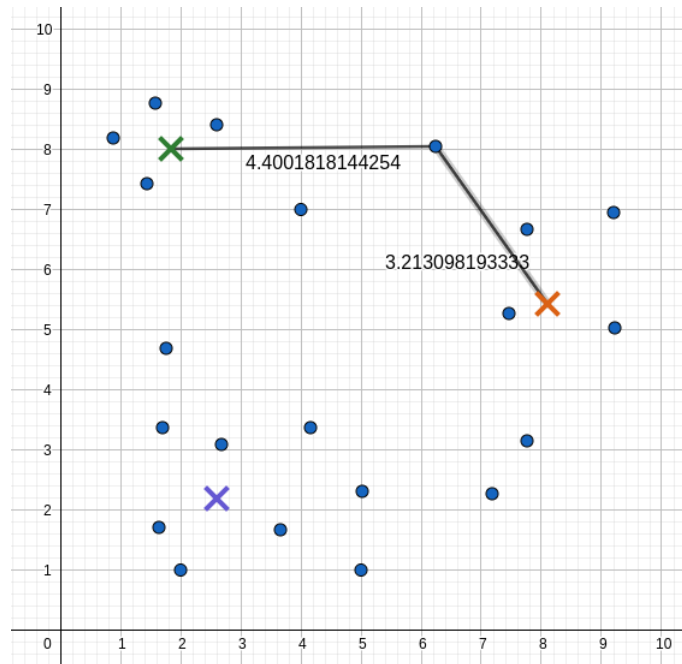


Figura 3.7: Buscando el centroide más cercano

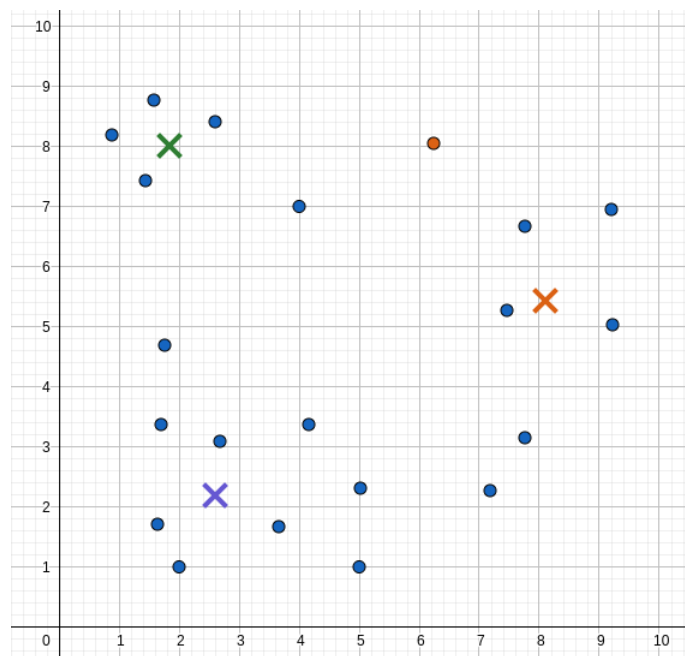


Figura 3.8: etiquetando el punto actual de acuerdo a su grupo

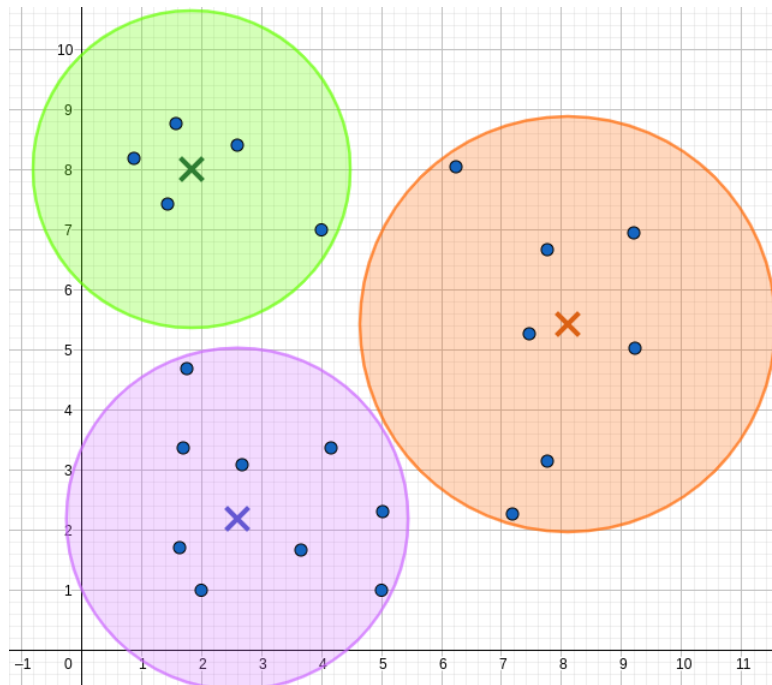


Figura 3.9: k grupos formados

3.5. Dask

En muchos de los casos, se usan herramientas de python como pandas, numpy, scikit-learn, etc para crear modelos de ML. Pero ¿qué se hace cuando el volumen de datos es más grande y pesado de lo que las librerías convencionales de python pueden procesar?

Para ese tipo de casos, python tiene una librería especial dedicada al procesamiento de grandes volúmenes de datos.

Como se muestra en su página <https://www.dask.org/>, dask simplifica las tareas de paralelización a la hora de hacer procedimientos de ML y DL.

Su característica principal es que dask está escrito sobre numpy y pandas, mientras que *dask-ml* está escrito sobre scikit-learn haciendo que la sintaxis de dask y pandas sea muy similar. Esto se puede ver en la figura 3.10 Ambos comandos dan el mismo resultado, es decir, regresan un dataframe con los datos cargados en un archivo csv.

Una propiedad especial que tiene `dask`, es que se pueden cargar varios archivos `csv` en una sola sentencia, siempre y cuando estos compartan esquema (mismas columnas).

```
import pandas as pd
df = pd.read_csv('2015-01-01.csv')
df.groupby(df.user_id).value.mean()
```

```
import dask.dataframe as dd
df = dd.read_csv('2015-*-.csv')
df.groupby(df.user_id).value.mean().compute()
```

Figura 3.10: Sintaxis de Pandas (arriba) Sintaxis de Dask (abajo)

De igual forma, tiene una sintaxis muy similar a la de `numpy`, justo como se ve en la imagen 3.11. De esta forma, `dask` puede realizar los mismos cálculos que puede hacer `numpy`.

```
import numpy as np
f = h5py.File('myfile.hdf5')
x = np.array(f['/small-data'])

x = x.mean(axis=1)
```

```
import dask.array as da
f = h5py.File('myfile.hdf5')
x = da.from_array(f['/big-data'],
                  chunks=(1000, 1000))
x = x.mean(axis=1).compute()
```

Figura 3.11: Sintaxis de Numpy (izq) Sintaxis de Dask (der)

3.6. Sistema de archivos HDFS

Un Sistema de Archivos Distribuidos de Hadoop o HDFS por su nombre en inglés Hadoop Distributed File System tiene como función principal almacenar grandes volúmenes de datos de manera distribuida.

De acuerdo a Karun y Chitharanjan (2013) un HDFS tiene una gran tolerancia a los fallos ya que este ha sido diseñado para ser implementado en sistemas cuyo hardware no requiera un gran costo de procesamiento.

El manual de HDFS escrito por Borthakur y cols. (2008) muestra que la arquitectura de este sistema consiste en el uso de clúster en los cuales se crean subconjuntos de datos, dando una arquitectura de Maestro y trabajadores.

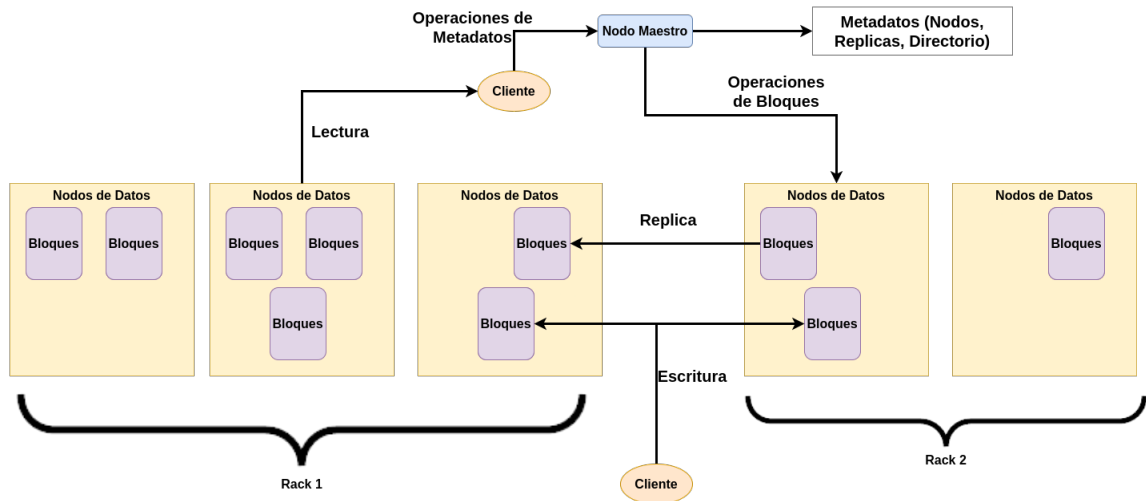


Figura 3.12: Diagrama de un HDFS

Como se puede ver en la figura 3.12 la arquitectura consiste en los siguientes elementos:

- Nodo Maestro, este almacena todos los datos del sistema de archivos en un clúster. Este también se encarga de almacenar todos los metadatos del clúster.
- Nodos de Datos son servidores de un solo archivo, lo que significa que si uno de estos nodos falla, el archivo aún se encuentra disponible en cualquier momento.
- Bloques de datos que representan un archivo. Cada bloque es replicado y añadido a un Nodo de Datos. Este proceso de replicación es rápido debido a que los bloques solo almacenan el nombre, ruta y permisos del archivo.

Capítulo 4

Métricas para Evaluar Modelos

Las métricas de evaluación ayudan a mejorar el rendimiento de modelos de ML. Esto se logra calculando la diferencia entre las variables predichas por el modelo y el valor real de estas.

El evaluar un modelo con más de una métrica puede dar mejores resultados en un modelo de ML.

4.1. RMSE

Error de Raíz Cuadrada Media o RMSE por sus siglas en inglés (Root Mean Square Error) es definida por Barnston (1992) de la siguiente manera:

$$RMSE = \sqrt{\sum_{i=1}^n \frac{(\hat{y}_i - y_i)^2}{n}} \quad (4.1)$$

Donde:

\hat{y}_i es el valor predicho por el modelo.

y_i es el valor real.

n el número de elementos en el conjunto de prueba.

El RMSE se puede interpretar como la desviación estándar de la varianza. Además de que esta nos da valores dentro de la misma escala de los datos.

Un RMSE bajo indica un mejor ajuste del modelo y un valor alto indican que el modelo requiere modificaciones.

4.2. MAE

Error absoluto medio (MAE por sus siglas en inglés) es una de las métricas más usadas para evaluar el desempeño de varios modelos de ML. Chai y Draxler (2014) menciona que MAE le da el mismo peso a todos los errores

$$\frac{\sum_{i=1}^n |\hat{y}_i - y_i|}{n} \quad (4.2)$$

Donde:

\hat{y}_i es el valor predicho por el modelo.

y_i es el valor real.

n el número de elementos en el conjunto de prueba.

Como se puede ver en la fórmula (**cambiar**) MAE mide qué tan cerca está la predicción en relación al valor real del conjunto de datos. Como mide el promedio de las distancias entre los valores reales y las predicciones, un MAE “perfecto” es cuando el promedio de las distancias es 0. Es decir, las predicciones fueron iguales a los valores reales.

Una modificación que se tiene de MAE es NMAE, Jannach y cols. (2010) muestran que consiste en normalizar los valores del MAE con respecto a los valores con los que se trabaja.

$$\frac{MAE}{r_{max} - r_{min}} \quad (4.3)$$

Donde:

r_{max} es el valor máximo del conjunto de datos.

r_{min} es el valor mínimo del conjunto de datos.

4.3. Matriz de confusión

Una Matriz de confusión, según López y cols. (2018) es una representación gráfica de los resultados de un modelo. Esta representación es algo similar a lo mostrado en la figura 4.1



Figura 4.1: Matriz de confusión simple de 2 valores (positivo/negativo)

Como se puede ver en la figura 4.1 existen 2 ejes, los valores de predicción y los valores reales. Dados estos ejes, la matriz de confusión se compone de:

- Verdaderos positivos: Son los valores clasificados como positivo y el valor real también es positivo.
- Falsos positivos: Son los valores clasificados como positivo y el valor real es negativo.
- Falsos negativos: Son los valores clasificados como negativo y el valor real es positivo.
- Verdaderos negativos: Son los valores clasificados como negativo y el valor real también es negativo.

Estos valores son útiles para calcular otras métricas de clasificación como la exactitud y la precisión.

4.4. Exactitud

Basados en la figura 4.1 existen verdaderos positivos (TP), verdaderos negativos (TN), falsos positivos (FP) y falsos negativos (FN) y estos se usan para calcular la exactitud y la precisión.

La exactitud es definida por Borja-Robalino y cols. (2020) como:

$$exactitud = \frac{TP + TN}{N} \quad (4.4)$$

Donde:

TP Son las predicciones positivas predichas correctamente.

TN Son las predicciones negativas predichas correctamente.

N Es el total de predicciones tanto correctas como incorrectas.

La exactitud se interpreta como la proporción de predicciones acertadas con respecto al total de datos.

Debido a esto, la exactitud representa qué tan cercano están los valores predichos con el valor real de los datos.

4.5. Precisión

La precisión mide el grado de proximidad o cercanía de los resultados entre sí y esta es definida también por Borja-Robalino y cols. (2020) como:

$$precision = \frac{TP}{TP + FP} \quad (4.5)$$

Donde:

TP Son las predicciones positivas predichas correctamente.

FP Son las predicciones negativas predichas como positivas.

La precisión se interpreta como la proporción de verdaderos positivos reales con respecto a los valores positivos predichos.

Kellman y Hansen (2014) mencionan que la exactitud se refiere a los errores sistemáticos del modelo, generando sesgo en los datos, mientras que la precisión se relaciona con algún componente aleatorio el cual genera ruido. En la figura 4.2 se da un ejemplo sobre predicciones/clasificaciones que carecen de exactitud y/o precisión.

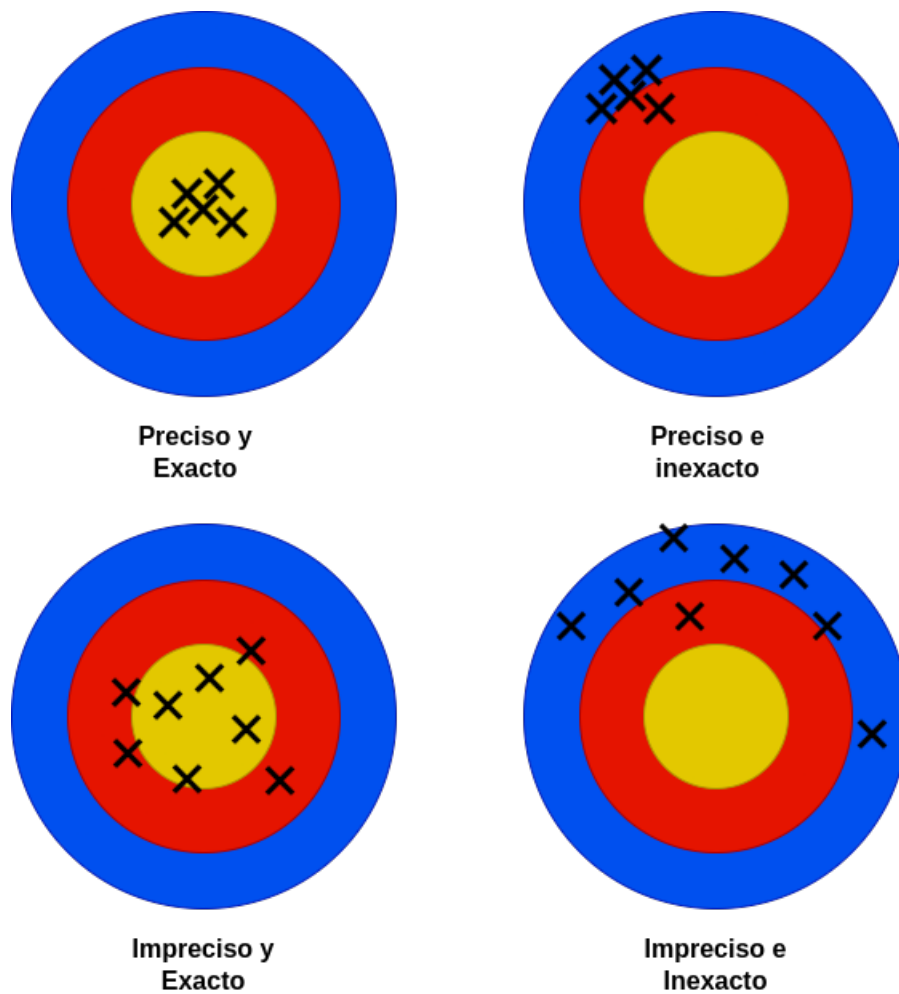


Figura 4.2: Representación gráfica de la exactitud vs precisión

4.6. Recall

El recall o sensibilidad muestra la proporción de verdaderos positivos predichos con respecto al número total de valores positivos.

En su trabajo, Davis y Goadrich (2006) definen el recall como

$$recall = \frac{TP}{TP + FN} \quad (4.6)$$

Donde:

TP Son las predicciones positivas predichas correctamente.

FN Son las predicciones positivas predichas como negativas.

De acuerdo a la ecuación 4.6 sabemos que si el recall da un valor cercano a 1, indica que el modelo tiene un buen rendimiento, en caso de dar un valor cercano a 0, indica que el modelo necesita ajustes

4.7. $F\beta$

Como se mencionó anteriormente, la precisión y el recall son métricas para cuantificar la calidad de clasificación de un modelo. De acuerdo con Derczynski (2016), estas métricas se pueden equilibrar de una forma proporcional, de acuerdo al objetivo deseado.

La $F\beta$ es definida por Goutte y Gaussier (2005) de la siguiente manera:

$$F\beta = (1 + \beta^2) \frac{PR}{\beta^2 P + R} \quad (4.7)$$

Donde:

P Es el valor de la Precisión.

R Es el valor del Recall.

β Determina el balance entre la precisión y el recall.

4.8. F1 score

De acuerdo a Chicco y Jurman (2020) el F1 es la métrica más común dentro del $F\beta$ score.

F1 es una media armónica entre la precisión y recall, es decir, ambas métricas tienen el mismo porcentaje de importancia. Teniendo esto en consideración Huang y cols. (2015) define F1 de la siguiente manera:

$$F1 = 2 \frac{PR}{P + R} \quad (4.8)$$

Como ambas métricas tienen la misma importancia, la única forma de tener un F1 alto es que P y R tengan un valor alto.

El problema de esto yace en que no siempre se pueden presentar casos en que la precisión y el recall sean altos debido a que si el valor de uno de ellos aumenta, el otro tiende a disminuir.

Para tratar este tipo de casos el valor de β puede variar de acuerdo al caso.

Los valores más utilizados en la práctica son:

- F1 con $\beta = 1$ representa el equilibrio entre precisión y recall.
- F0.5 con $\beta = 0,5$ le da más importancia a la precisión.
- F2 con $\beta = 2$ le da más importancia al recall.

Capítulo 5

Metodología de las Prácticas

Cada una de las prácticas presentadas en el anexo contienen las siguientes fases, en el orden en que se muestran:

1. Objetivo de la práctica.
2. Conceptos.
3. Herramientas a usar.
4. Desarrollo.
 - a)* Entender el Problema.
 - b)* Definir un criterio de evaluación.
 - c)* Preparar los datos.
 - d)* Construir el modelo.
 - e)* Análisis de errores.
 - f)* Implementación.

N° Práctica	NOMBRE	DATASET	CRITERIO DE EVALUACIÓN	DESCRIPCIÓN
1	Clasificación usando árboles de decisión	Pasajeros del Titanic	Exactitud y/o Métrica Fbeta	Construir un árbol de decisión para clasificar si los pasajeros del Titanic sobreviven o no dadas características como el sexo, edad, status, etc.
2	Predicción del costo de casa-habitación (Regresión Lineal)	California Housing	RMSE y/o MAE	Construir un modelo de predicción para el costo de las casa habitación en el este de California (década de 1990).
3	k -vecinos más cercanos	Pozos profundos del lago de Cuitzeo	Precisión	Usar un dataset de pozos para hacer un modelo de KNN que agrupe elementos conforme al volumen de extracción de pozos en Michoacán (supervisado).
4	Aprendizaje no supervisado (k-means)	Online Retail K-means & Hierarchical Clustering	No aplica	Diseñar un modelo de K-means para hacer clasificación las transacciones de los clientes de un banco y así poder identificar a los diferentes clientes que hay (no supervisado).
5	Instalación y uso de Dask	3 nodos virtuales con CPU y GPU c/u	No aplica	Mostrar cómo se realiza la instalación de Dask y cómo se usa para la manipulación de grandes volúmenes de datos.
6	Instalación y uso de HDFS	3 nodos virtuales con CPU y GPU c/u	No aplica	Enseñar paso por paso cómo se realiza la instalación de Hadoop y cuál es la utilidad de los comandos de este mismo.
7	Predicción del clima (Regresión Lineal)	RUOA de 2015 a 2021	RMSE y/o MAE	Usar los datos climáticos obtenidos por la RUOA para hacer un modelo de regresión lineal capaz de predecir el clima de los siguientes días.

Capítulo 6

Resultados y Discusión

Para conocer los temas más relevantes sobre ML para los alumnos, se elaboró una encuesta la cual cuantificaba diferentes temas y herramientas de interés para los estudiantes.

Usando la información obtenida de la encuesta aplicada con anterioridad, se generaron análisis descriptivos donde se realizó el estudio de confiabilidad aplicando el Alfa de Cronbach obteniendo como resultado 0.956 y demostrando que la información obtenida es consistente.

También se aplicó un estudio de correlaciones utilizando la bivariada de Pearson y seleccionando únicamente las correlaciones obtenidas en los niveles alto y muy alto $[0,7-0,93]$, que se muestran en la figura 6.1.

De acuerdo al análisis de correlaciones, se identificaron áreas de oportunidad según porcentaje de importancia que respondieron los encuestados. En la figura 6.2 se muestra esta importancia.

De acuerdo con la encuesta desarrollada, se observó que los encuestados tenían cierto desconocimiento en algunos temas. En la figura 6.3 los temas se muestran por eje, ordenados por nivel de desconocimiento: No sé (dnK), Nada importante (NImp), Menos importante (LImp), Neutro Importante (Imp), Muy importante (VImp) y para las herramientas, la escala es: No sé (dnK), Corta, Media y Alta.

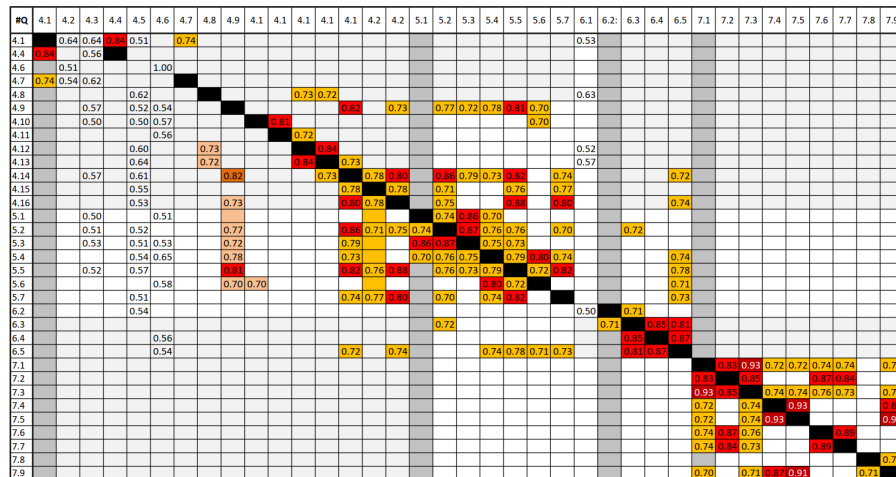


Figura 6.1: Matriz de Correlación de los resultados obtenidos en la encuesta

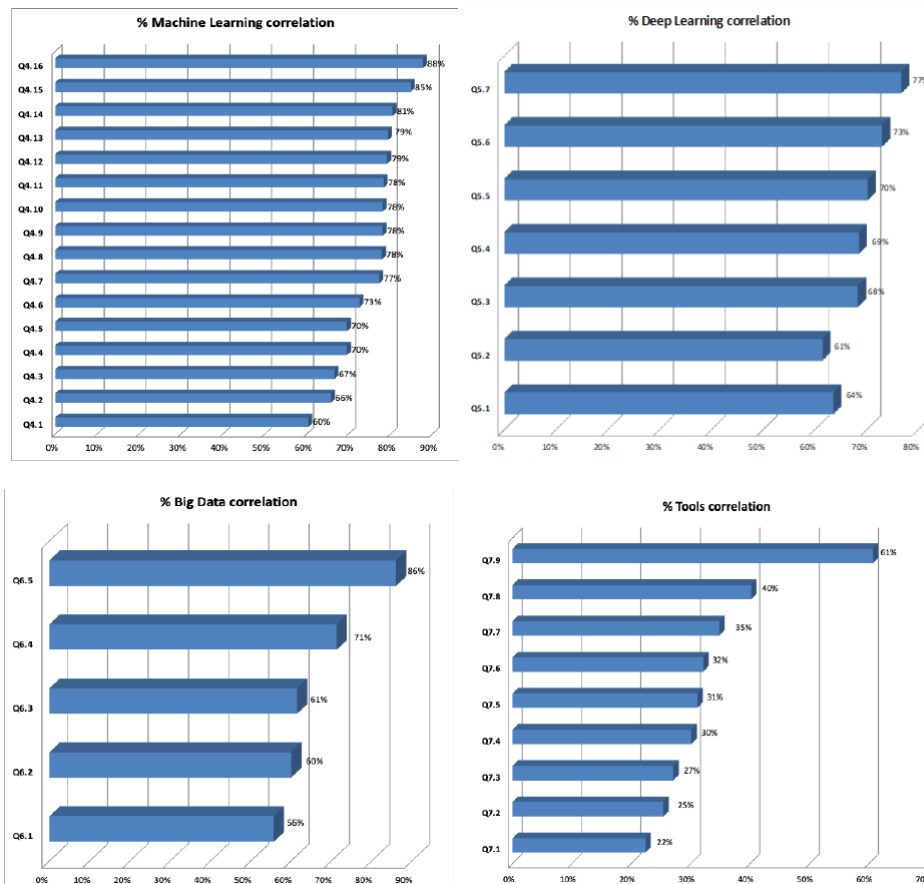


Figura 6.2: Nivel de importancia de acuerdo a los encuestados

Por la experiencia presentada durante el diplomado práctico dirigido a una mezcla de estudiantes y docentes de la Morelia campus de la universidad UNAM, divididos



Figura 6.3: Niveles de desconocimiento

en dos grupos heterogéneos, en cuanto a la aplicación de las prácticas propuestas, se ofrecieron dos cursos de acuerdo al diplomado descrito a continuación:

MÓDULO I. Aprendizaje automático (ML). "Teoría y Práctica para la Mejora de la Enseñanza del ML Aplicado a la Ciencia de Datos" Ver tabla 5

1. Árboles de decisión
2. Métodos de regresión
3. Métodos de clasificación
4. Métodos de predicción
5. Clustering
6. Sistemas de archivos distribuidos (HDFS)

Las herramientas utilizadas en el diplomado fueron: Anaconda Python, Scikit-Learn, Matplotlib, Dask, HDFS, entre otros.

Al finalizar el primer curso, donde se realizó la intervención en ML, se observó que el 50 % de los asistentes, de un total de 40, tenían diversos problemas de práctica resueltos.

Estos problemas se muestran como porcentajes de prácticas resueltas en la Fig 6.4. En esta figura, se comparan los resultados esperados (según nuestras experiencias en cursos anteriores) frente a los resultados reales, como prácticas resueltas y entregadas por los asistentes. Además, la figura muestra la eficiencia de la enseñanza según el siguiente criterio:

$$\%eficiencia = 100 * (prcticas\ resueltas - practicas\ esperadas) / practicas\ esperadas \quad (6.1)$$

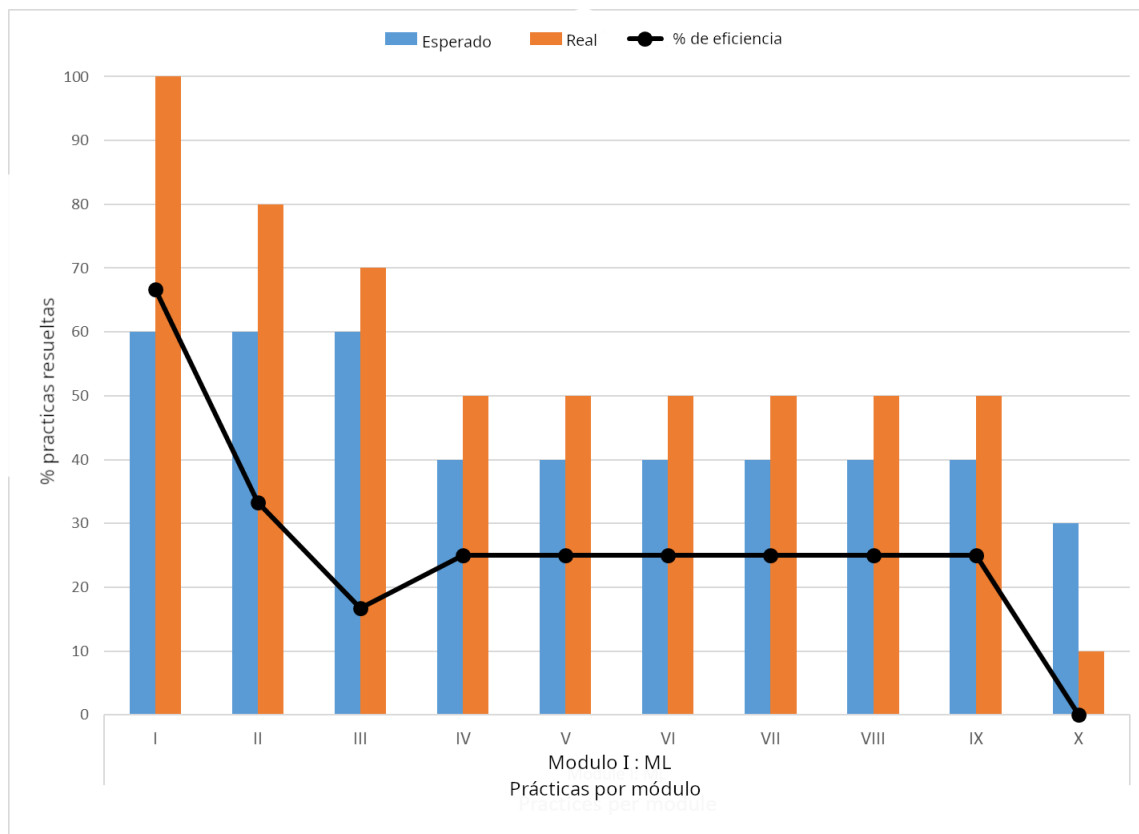


Figura 6.4: Resultados del Módulo I

También se observó que los estudiantes dejaron de trabajar en las prácticas más complejas de ML. Las principales razones que se identificaron fueron el aumento de las tareas de análisis de datos, además de tener que aplicar estadística y teoría matemática utilizando un lenguaje de programación (Python). La solución a estos problemas es dar mayor prioridad a la práctica con datos reales que a la teoría abstracta.

En trabajos similares a este, la enseñanza de ML se utiliza solo como un caso de uso para abordar educación en otros temas en casos reales; de hecho, con el enfoque de IA, pero no se realizan mejoras en la enseñanza de ML en sí, ni experiencias sobre cómo mejorar la enseñanza de la ciencia de datos.

La propuesta práctica en este trabajo permite establecer un currículo más completo y amplio, en cuanto a que incluye no solo el ML sino también el DL, el Big Data y las herramientas informáticas asociadas con la ciencia de datos.

Esta propuesta aún está en desarrollo y entre otras cuestiones, es necesario evaluar la eficiencia de la enseñanza de acuerdo con este enfoque práctico en un curso de DL, así como incluir otras herramientas que pueden ayudar a facilitar el aprendizaje de la ciencia de datos. Además de incluir plataformas online orientadas a la enseñanza así como otras herramientas que pueden ayudar facilitar el entendimiento de la ciencia de datos, como las plataformas de aprendizaje colaborativo en la nube.

Apéndice

Los anexos consisten en:

- Código fuente en lenguaje Python (como libretas de JupyterLab) de las prácticas desarrolladas.
- El listado de enlaces a las fuentes originales de las cuales se tomaron los conjuntos de datos para las prácticas.
- El texto original del artículo publicado en las memorias del Congreso Internacional *IntelliSys 2022*.

.1. Prácticas

Las prácticas (como libretas de JupyterLab) se encuentran en el siguiente repositorio en la plataforma **GitHub**: <https://github.com/MichellMonroy/Practicas-ML>

.2. Datos

Los conjuntos de datos que se usaron en las prácticas mostradas anteriormente, se pueden descargar de los enlaces siguientes:

- Pasajeros del Titanic.
<https://www.kaggle.com/c/titanic/data>
- California Housing.
<https://www.kaggle.com/fedesoriano/california-housing-prices-data-extra-features>
- Pozos profundos del lago de Cuitzeo
https://drive.google.com/file/d/19WVDYOM1xbF2hvbo75MDGe7OMN1clGhK/view?usp=share_link
- Online Retail K-means & Hierarchical Clustering.
<https://www.kaggle.com/hellbuoy/online-retail-customer-clustering>
- Anime Recommendation Database 2020.
<https://www.kaggle.com/hernan4444/anime-recommendation-database-2020>

- RUOA de 2015 a la actualidad.

<https://ruoa.unam.mx/index.php?page=estaciones&id=9>

- 100,000 UK Used Car Data set.

<https://www.kaggle.com/adityadesai13/used-car-dataset-ford-and-mercedes>



- Información de datos públicos.

Por decidirse

.3. Artículo



How to Improve the Teaching of Computational Machine Learning Applied to Large-Scale Data Science: The Case of Public Universities in Mexico

Sergio Rogelio Tinoco-Martínez¹, Heberto Ferreira-Medina^{2,3}() ,
José Luis Cendejas-Valdez⁴() , Froylan Hernández-Rendón¹,
Mariana Michell Flores-Monroy¹, and Bruce Hiram Ginori-Rodríguez¹

¹ Escuela Nacional de Estudios Superiores Unidad Morelia, UNAM Campus Morelia,
58190 Morelia, Michoacán, Mexico

{stinoco, fherandez}@enesmorelia.unam.mx

² Instituto de Investigaciones en Ecosistemas y Sustentabilidad, UNAM Campus Morelia,
58190 Morelia, Michoacán, Mexico

hferreir@iies.unam.mx

³ Tecnológico Nacional de México, Campus Morelia, DSC, Morelia, Michoacán, Mexico

⁴ Departamento de TI, Universidad Tecnológica de Morelia. Cuerpo académico

TRATEC-PRODEP. Morelia, 58200 Morelia, Michoacán, México

luis.cendejas@ut-morelia.edu.mx

Abstract. Teaching along with training on Machine Learning (ML) and Big Data in Mexican universities has become a necessity that requires the application of courses, handbooks, and practices that allow improvement in the learning of Data Science (DS) and Artificial Intelligence (AI) subjects. This work shows how the academy and the Information Technology industry use tools to analyze large volumes of data to support decision-making, which is hard to treat and interpret directly. A solution to some large-scale national problems is the inclusion of these subjects in related courses within specialization areas that universities offer. The methodology in this work is as follows: 1) Selection of topics and tools for ML and Big Data teaching, 2) Design of practices with application to real data problems, and 3) Implementation and/or application of these practices in a specialization diploma. Results of a survey applied to academic staff and students are shown. The survey respondents have already taken related courses along with those specific topics that the proposed courses and practices will seek to strengthen, developing needed skills for solving problems where ML/DL and Big Data are an outstanding alternative of solution.

Keywords: Machine learning · Deep learning · Big data · Data science · Teaching skills

1 Introduction

The use of tools that allow the analysis of large volumes of data has allowed exact sciences to play an important role for decision-making in organizations [1]. In the Bachelor

of Information Technologies for Sciences (ITCs) of the Escuela Nacional de Estudios Superiores (ENES) Morelia, Mexico, there are subjects related to Data Science (DS) [2] that are included in the curriculum starting from the 6th semester, known as subjects of the deepening area, and that represent a challenge for students when trying to put the theory learned into practice, in addition to lacking the necessary tools for its application in real problems. The need for teachers and students to know new frontiers in Artificial Intelligence (AI) is observed, specifically in the application of mathematical models of Machine Learning (ML). ML is the branch of AI that is responsible for developing techniques, algorithms, and programs that give computers the ability to learn. A machine learns each time it changes its structure, programs, or data, based on input or in response to external information, in such a way that better performance is expected in the future [3].

In [4] Deep learning (DL) is used to explain new architectures of Neural Networks (NN) that are capable of learning. DL is a class of ML techniques that exploit many layers of nonlinear processing for extraction and transformation of supervised and unsupervised features and pattern analysis and classification [5]. The 21st century has become the golden age for AI; this is due, in large part, to a greater computing capacity and the use of GPUs to speed up the training of these systems, with the ingestion of large amounts of data. Currently, numerous frameworks have ML and DL tools implemented, such as PyTorch [6], fast.ai [7], TensorFlow [8], Keras [9], DL4J [10], among others. Some of the main uses of DL today are, for example, identifying brand names and company logos in photos posted on social media, real-time monitoring of reactions on online channels during product launches, ad recommendation and prediction of preferences, as well as identification and monitoring of customer confidence levels, among others. The use of AI has allowed a better understanding of genetic diseases and therapies, the analysis of medical images (such as X-rays and magnetic resonance imaging) increasing the accuracy of diagnosis in less time and at a lower cost than traditional methods [11]. The DL forms a subcategory of the ML. To differentiate it from the rest of ML algorithms, it uses the fact that large-scale NNs allow a machine to learn to recognize complex patterns by itself, which is difficult to achieve with them [12]. A NN is made up of several layers or levels and a certain number of neurons in each of them, which constitute the processing unit, whose mathematical model allows having several data inputs and an output that is the weighting of their inputs [13, 14]. The connections of several neurons within a NN constitute a powerful parallel computation tool, capable of delivering approximate and non-definitive outputs. Furthermore, NNs can be structured in various ways and can be trained with various types of algorithms [15]. On the other hand, the Internet of Things (IoT) and industry 4.0 have required the introduction of autonomous and intelligent machinery in the industrial sector [16]. In this industry, Convolutional Neural Networks (CNN) are applied, which are a type of DL that is inspired by the functioning of the visual cortex of the human brain and differs from the other NNs by the fact that each of the neurons of the layers that compose it does not receive incoming connections from all the neurons of the previous layer, but only from some of them. This simplifies the learning of the network, generating lower computational and storage costs. All of the above mentioned makes DL models more accurate [12].

The main contribution of this paper is to present a proposal to improve the learning of ML, DL, and Big Data subjects with practical training focused on real-life use cases. The rest of the paper is organized as follows: Sect. 2 describes the work related to AI (ML, DL, and Big Data) and the implementation of its teaching in courses or diplomas oriented to data science. It shows the difficulty of the students in learning and applying the topics of ML and DL in data analysis. This problem must be solved with a practical approach or orientation. Section 3 describes the methodology used to develop the improvement proposal. Section 4 presents the results, as well as a brief discussion of them. Finally, Sect. 5 presents the conclusions obtained from the developed proposal.

2 Related Work

In current terms [17] explains that within AI there is a branch called ML whose purpose is to improve the performance of algorithms through their experience. The ML uses statistics, computer science, mathematics, and computational sciences having its foundation in data analysis. The definition coincides with other proposals that consider ML as the technique of creating systems that are capable of learning by themselves, using large volumes of data, making them suitable for analysis and, thus, being able to predict future behavior [18]. Regarding the ML as an area [12] points out that there are different approaches for the design of these systems. The approaches are divided into supervised, unsupervised and by reinforcement, if the system is trained under human supervision or not (the most used division); online or offline learning, whether the system can learn on the fly or not; and, finally, in instance-based learning or model-based learning, if the system detects training patterns or if it compares new data against existing data.

The supervised ML approach is used when you already have data, and you know the response you want to predict. This knowledge is then used to predict the labels of new data whose label is unknown. The main problems that are solved with this type of learning are regression (predicting the future value of a given element, whose values can only be numerical, from relevant characteristics and previous values [19]) and classification (assigning a label to a given element, from a discrete set of possibilities [20]).

In the unsupervised ML approach, the data is not labeled, this means that the system must learn by itself without being told if the classification is correct or not [21]. To classify the data, a grouping technique is used whose objective is to combine data whose characteristics are like each other [22].

Regarding the reinforcement ML approach, the goal is that the system learns in an environment in which the only feedback consists of a scalar reward, which can be positive or negative (punishment) [23]. That is, the system receives in each iteration a reward and the current state of the environment, then takes an action according to these inputs and what results, is considered as an output, which will change the state in the next iteration [24].

The arguably most used algorithms of supervised ML are linear or logistic regression, for regression [25]; and decision trees, k-nearest neighbors, support vector machines or artificial NNs, for classification [26]. The algorithms for unsupervised ML are k-means, visualization and dimensionality reduction or association rules [27].

In relation to artificial NNs, they serve to solve both regression and classification problems and even some unsupervised learning problems. Due to its versatility and

performance that have recently surpassed even human performance (at the cost of having example data in large quantities that the IoT has allowed to obtain).

The study of DL began in 1943 as a computer model inspired by the NNs of the human brain [28], however, it was not until 1985 that in [4] it was demonstrated that backpropagation in a NN could provide distribution representations of great utility for learning, generating a reborn interest in the area.

In [29] the first practical demonstration of backpropagation is provided. The team combined CNNs with backpropagation to read handwritten digits in a system that, finally, was used to read handwritten check numbers. The model was based on the hierarchical multi-layer design (CNNs) inspired by the human visual cortex of the Neocognitron, introduced in 1979 [30].

In the late 1990s, the problem of the fading or exploding gradient was detected in DL models. The problem originates in the activation functions of artificial neurons whose gradient (based on the derivative) decreased when calculated in each layer, until it reached practically zero (or tended to an infinite value); which implies loss of learning. The proposed solution is to store the gradient within the network itself [31] or to make it enter a layer and simultaneously avoid it through skip connections [32].

In [31] Recurrent NNs (RNNs) of the Long Short-Term Memory (LSTM) type are proposed, which specialize in the analysis and prediction of time series and problems that must recall previous states, such as Natural Language Processing (NLP). In addition, this architecture allows to solve the gradient problems mentioned above.

In 2009 ImageNet [33] was launched, a free, tagged database of more than 14 million images, which features a thousand different categories of objects as varied as 120 dog breeds. With this resource and coupled with the computing power that the evolution of GPUs already had by 2011, it became possible to train CNNs without the previous training (layer by layer) and considering architectures with an increasing number of these (hence the term DL).

The examples that can be mentioned of the efficiency and speed that DL algorithms have achieved are the computer vision algorithms that were winners in the ImageNet Large Scale Visual Recognition Challenge (LSVRC) [34] between the years of 2012 up to 2017 (all CNNs architectures), whose main challenge is based on the classification of images from the ImageNet database and that, as of 2017, it is considered solved in practice and with superior performance to that of the human being.

To our knowledge, there are very few works in the recent literature related to the improvement of AI teaching. Some of the most representative ones are mentioned below, as a review of the approaches they address and that are aimed at teaching AI as a secondary objective or use case.

According to [35] ML is a discipline that focuses on building a computer system that can improve itself using experience. ML models can be used to detect patterns from data and recommend strategic marketing actions, showing how educators can improve the teaching of these topics using the AI approach. The availability of Big Data has created opportunities and challenges for professionals and academics in the area. Therefore, study programs must be constantly updated to prepare graduates for rapidly changing trends and new approaches. In [36] it is described that the pandemic caused by the COVID-19 virus, the advent of Industry 4.0 confronts graduate students with the need

to develop competencies in ML, which are applied to solve many industrial problems that require prediction and classification, and the availability and management of large amounts of data. The proposal of how to apply AI practices in a virtual laboratory is shown, in addition to evaluating the performance of students in this type of environment. In the same way, in [37], an innovative practice of teaching applied ML to first-year multidisciplinary engineering university students is proposed, using a learning tool that consists of a public repository in the cloud and a course project. A set of practices for ML and how to apply it in real cases is offered as a use case for online collaborative work. The inclusion of DL and Big Data is mentioned as future work.

In Mexico, there are many academic degrees oriented to DS, ML, and DL. However, there are no uniform curricula on the areas of knowledge, topics, and tools that students require. In this paper, we offer an alternative way to solve this problem based on the experience of a public university such as the ENES Morelia - UNAM.

3 Methodology

Based on the review of the literature, a series of steps were generated, which allowed us to obtain the level of knowledge that the ENES Morelia population has about ML/DL and thus be able to define axes that support the design of the pedagogical strategy that will give rise to the proposal of a uniform curriculum through courses of practical experience. The present research is characterized by being a study of type: 1) exploratory, 2) descriptive, 3) correlational and 4) pre-experimental to have a case study through a single measurement. To this end, a survey was generated that was applied to a population made up of professors, students, and researchers of the UNAM (Morelia, Michoacan campus). The methodology followed for this work is shown in Fig. 1. Results of the analysis of the application of this survey and the monitoring of test groups are described in the following sections.

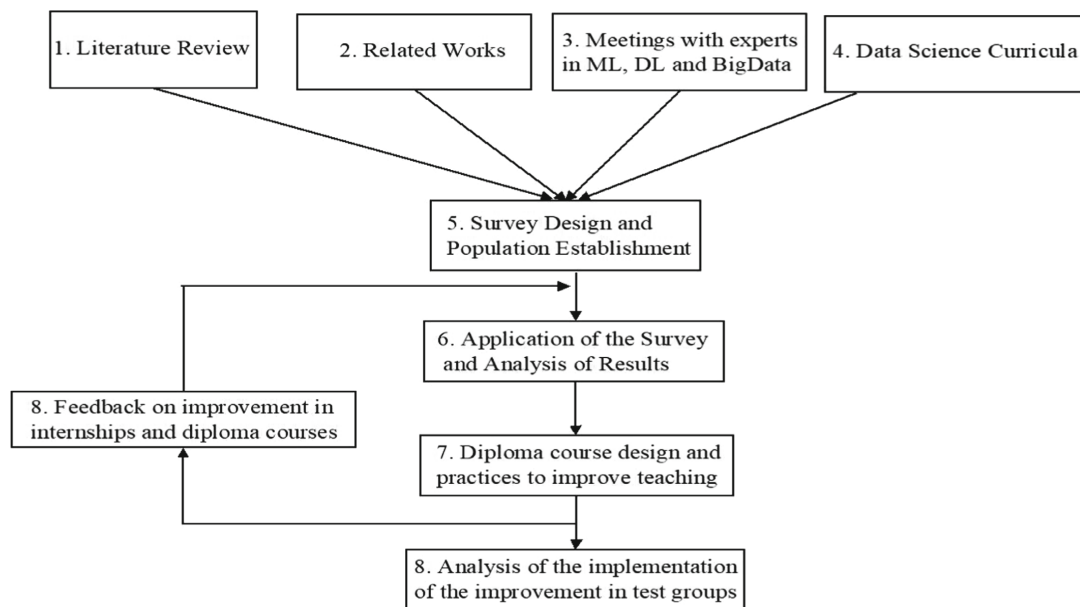


Fig. 1. Block diagram of the methodology used in this work.

3.1 Population

Primarily the survey was created through the “e-Encuesta®” Web platform and distributed via email and social media to randomly selected individuals affiliated with the campus mentioned before. In the first place, the sample was calculated using the finite population method based on 600 people. This sample has a confidence interval of 95% and a margin of error of 10%, as shown in Table 1. The link to the survey within the Web platform was distributed through the official email of students and teachers. It was validated that the information in each of the answers was consistent and complete.

Table 1. Population sample.

Description	Value
Population size	600
Trust level	95%
Margin of error	10%
Sample size	83

3.2 Survey

An eight-question survey was generated from a critical review of the literature related to DS of ML, DL and Big Data. Experts on the subject validated the selected questions. Its measurement was: a) different options, b) dichotomous responses, and c) Likert scale. The survey was refined by dividing it into four axes: Axis I: Machine Learning; Axis II: Deep Learning; Axis III: Big Data; and Axis IV. Tools, as shown in Table 2. Likert scale applied was: Very important, Important, Neutral, Less important, Nothing important, I do not know.

Table 2. Survey questions and format.

#	Question description	Axis	Type
Q1	UNAM account or employee number, your name if you do not have them	–	Options
Q2	Bachelor’s degree you are studying; 2.1 Sciences, 2.2 Agroforestry, 2.3 Environmental Sciences, 2.4 Sustainable Materials Science, 2.5 Ecology, 2.6 Social Studies and Local Management, 2.7 Geosciences, 2.8 Geohistory, 2.9 Information Technologies in Sciences, 2.10 Other	–	Options
Q3	Semester you are studying (1–12), does not apply to teachers and researchers	–	Number

(continued)

obtaining as a result 0.956 and demonstrating that the information obtained is consistent. Third, the study of correlations was applied using Pearson's bivariate and selecting only the correlations obtained at the high and very high levels [0.7–0.93], shown in Fig. 2.

According to the analysis of correlations, we identify areas of opportunity according to percentage of importance (scale) that the respondents answered. In Fig. 3 this importance is shown.

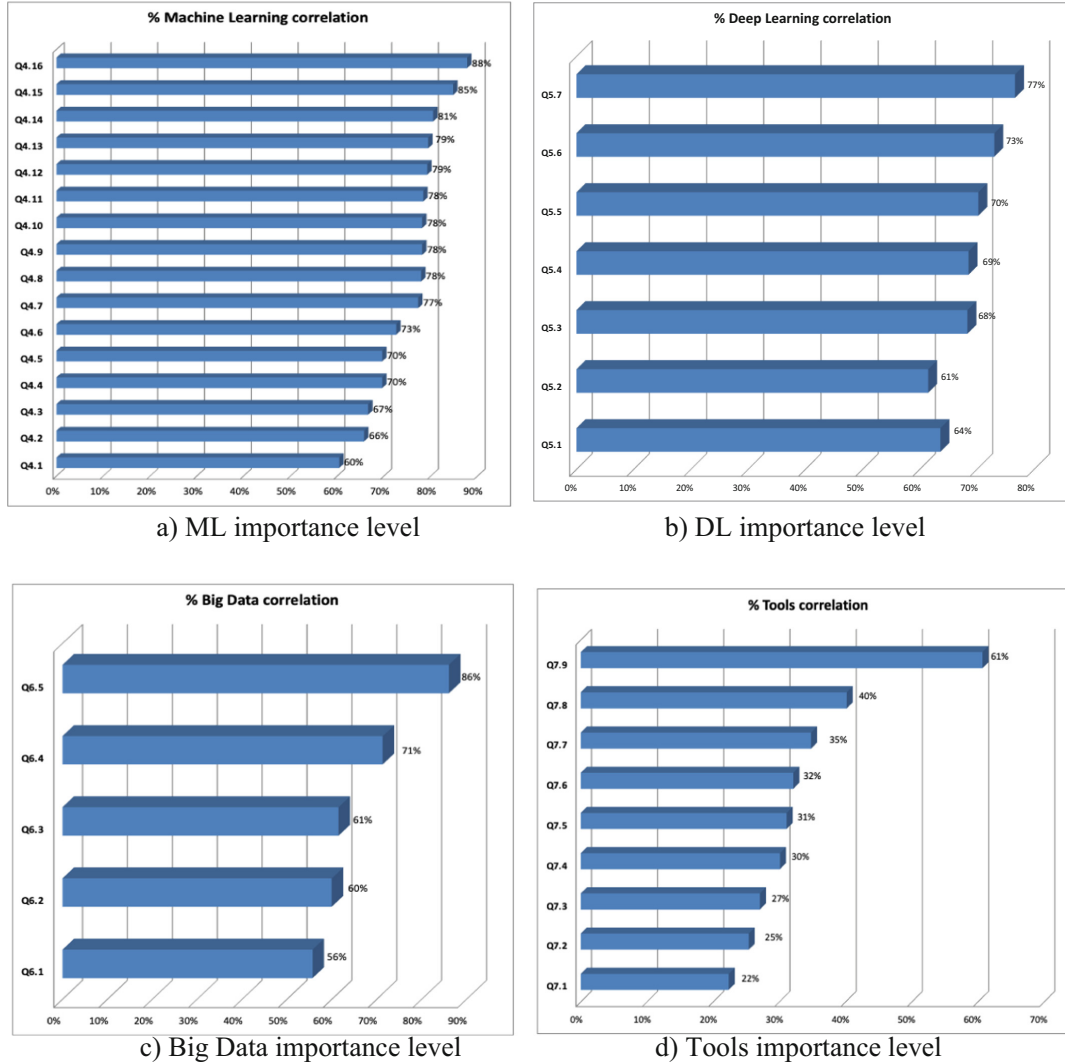


Fig. 3. Level of importance (Likert) according to survey respondents; a) ML, b) DL, c) big data and d) tools. See Table 1, for tag details.

4 Results and Discussion

According to the developed survey, a certain lack of knowledge of the respondents was observed in some topics. In Fig. 4 topics are shown by axis, ordered by level of unfamiliarity: I do not know (dnK), Nothing important (NImp), Less Important (LImp),

Neutral, Important (Imp), Very Important (VImp). And, for tools, scale is: I do not know (dnK), Short, Half, and High.

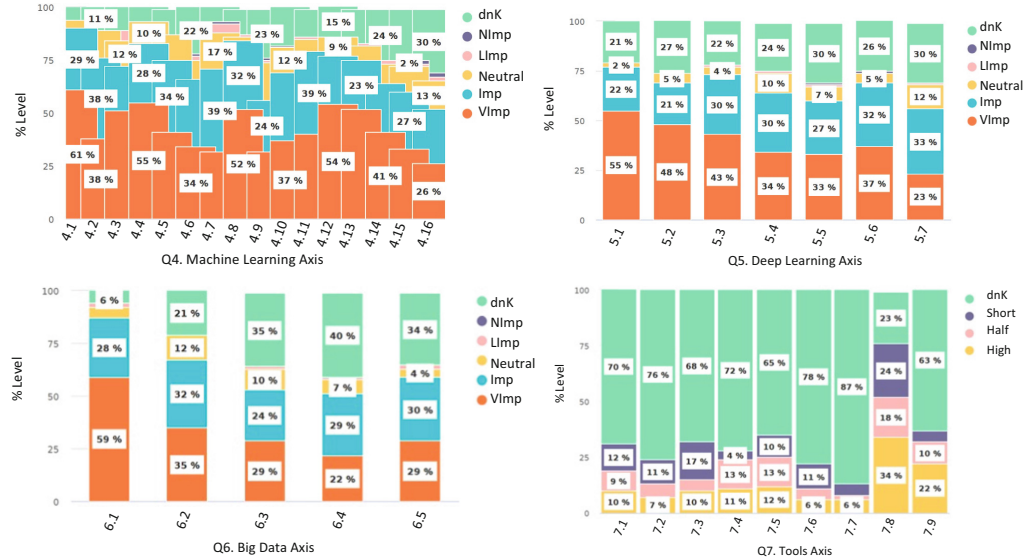


Fig. 4. Levels of unfamiliarity by axes. See Table 1, for tag details.

Based on the analysis of these results and considering the classic progression in recent literature, regarding the teaching of basic topics of ML, DL, and Big Data; coupled with our personal experience in teaching courses on these topics, at the undergraduate level and more of them aimed at teachers in the area of Information Technology, it is proposed to improve learning with practical training to strengthen those topics with the greatest lack of knowledge (dnK level of unfamiliarity in Fig. 4). This practical knowledge is shown in Tables 3 and 4 as a series of practices we recommended to take advantage of these areas for improvement. It was observed that respondents prefer an intervention oriented towards the practical application of knowledge.

Table 3. Proposed ML practices.

#	Name	Dataset	Evaluation metric	Description
1	Classification using decision trees	Titanic passengers [38]	Accuracy and/or Fbeta Metrics	Build a decision tree for the survival analysis of Titanic passengers (Classification)
2	Housing cost prediction	California Housing [39]	RMSE and/or MAE	Build a real estate cost prediction model. (Linear Regression/Logistic Regression)

(continued)

Table 3. (continued)

#	Name	Dataset	Evaluation metric	Description
3	k-Nearest Neighbors	Water wells [40]	Precision Score Fbeta	Build a prediction model of water well uses. (Supervised)
4	k-Means	Online retail K-means & Hierarchical clustering [41]	Not applicable	Design a model to classify the transactions of a bank's customers. (No supervised)
5	Installing and using Dask [42]	Not applicable	Not applicable	Show Dask installation and how it is used for Big Data manipulation
6	Installing and using HDFS [43]	Not applicable	Not applicable	Teaching how the installation of HDFS and its basic use is carried out
7	Weather forecasting	RUOA (UNAM, 2015) [44]	RMSE and/or MAE	Analyze climate data from the RUOA to predict weather on a daily horizon. (Linear Regression)
9	Car price prediction	100,000 UK Used Car Data set [45]	RMSE and/or MAE	Analyze car data to estimate prices. (Multiple regression)
10	Special case	Public data information	Several	Analyze data to apply the best strategy to solve a problem

Table 4. Proposed DL practices.

#	Name	Dataset	Evaluation metric	Description
1	Binary classification with CNN	800 images of mosquitoes, UNAM [46]	Accuracy	Differentiate between species <i>Aedes Albopictus</i> and <i>Aedes Aegypti</i> . (Visualization)
2	Binary classification with CNN	Covid-19 pneumonia screening [47]	Precision & confusion matrix	X-ray tomography analysis for identification of lungs affected by the SARS-CoV-2 virus

(continued)

Table 4. (continued)

#	Name	Dataset	Evaluation metric	Description
3	CNN & data augmentation	Ship Classification [48]	Precision & Confusion Matrix	Classification of 6,252 images of ships (5 categories)
4	RNN	Sarcasm Detection [49]	Accuracy and precision	Identify news titles that are sarcastic or satirical. (NLP)
5	Installing and using PyTorch over Dask	Not applicable	Not applicable	Installation and use of PyTorch in Dask
6	Transfer learning	Sports images [50]	Precision and accuracy	Classification of sports images

5 Conclusions

By the experience of practical teaching to a mix of students and teachers of the Morelia campus of the UNAM university, divided into two heterogeneous groups, concerning applying the proposed practices, two courses were offered according to the diploma described below [51]:

MODULE I. Machine Learning (ML). “Theory and Practice for the Improvement of the Teaching of ML Applied to Data Science”. Topics: 1. Artificial Intelligence and Machine Learning, 2. Phases of an ML Project, 3. Regression Methods, 4. Classification Methods, 5. Prediction Methods, 6. Supervised Learning, 7. Unsupervised Learning, 8. Metrics. Practices to be Developed: See Table 3.

MODULE II. Deep learning (DL). “Theory and Practice for the Improvement of the Teaching of DL Applied to Data Science”. Topics: 1. Artificial Intelligence and Deep Learning (DL), 2. Phases of a DL Project, 3. Convolutional Neural Networks (CNNs), 4. Learning Transfer, 5. Recurrent Neural Networks (RNNs), 6. Visualization and Treatment of Natural Language Processing (NLP). Practices: See Table 4.

Tools used in the diploma: Anaconda Python, Scikit-Learn, Matplotlib, Dask, PyTorch, fast.ai 2, TensorFlow, Keras, HDFS, among others.

At the end of the first course where the intervention was carried out, it was observed that 50% of the attendees, of a total of 40, had various problems solving practices. These problems are shown as percentages of solved practices in Fig. 5. In this figure, the expected results (according to our experiences in previous courses) are compared against the actual results, as practices solved and delivered by the attendees. In addition, the figure shows the efficiency of teaching according to:

$$\%efficiency = 100 * (solved\ practices - expected\ practices) / expected\ practices \quad (1)$$

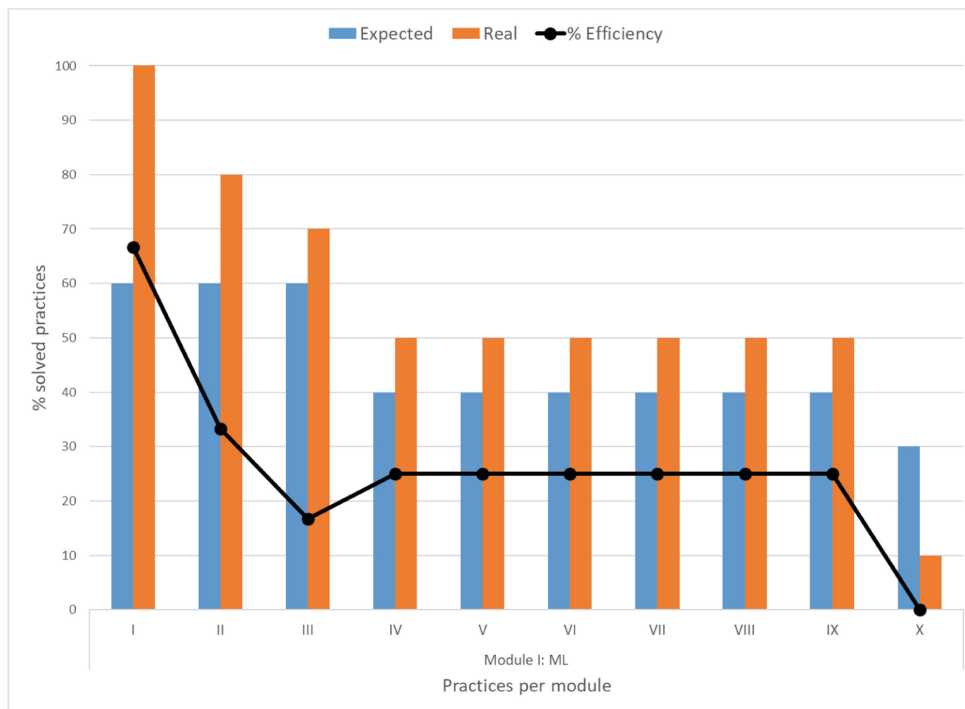


Fig. 5. ML teaching effectiveness.

We observed that students quit working on the most complex practices of ML. Reasons were the increase in data analysis tasks, on top of having to apply statistical and mathematical theory using a programming language (Python). The solution to these problems is to give higher priority to practice with real data than to abstract theory.

In works similar to this one, the teaching of ML is used only as a use case to address education in other topics in real cases; indeed, with the AI approach, but no improvements are made to the teaching of ML itself, nor experiences on how to improve the teaching of data science are included. The practical proposal in this work allows for establishing a more complete and broad curriculum, concerning the fact that it includes not only the ML but the DL, the Big Data, and the computer tools associated with data science as well.

Our proposal is still under development and, among other issues, it is necessary to evaluate the efficiency of teaching according to this practical approach in a DL course, as well as including other tools that may help facilitate the learning of data science. All this, in addition to including learning platforms as well as other tools that can help facilitate data science learning, such as collaborative learning platforms in the cloud.

Acknowledgment. We are grateful for the support of the Instituto de Investigaciones en Ecosistemas y Sustentabilidad (IIES), the CA TRATEC - PRODEP of the Universidad Tecnológica de Morelia (UTM), the TecNM campus Morelia, the Escuela Nacional de Estudios Superiores (ENES), UNAM Campus Morelia, and DGAPA UNAM PAPIIME PE106021. Especially thanks to MGTI. Atzimba G. López M., MTI. Alberto Valencia G., Eng. Oscar Álvarez, MTI. Pablo García C. and Eng. Javier Huerta S., for their technical support, comments, and analysis of the

statistical calculations. We thank Eng. Alfredo A. Aviña, for his help in applying the survey and Web page support.

References

1. Haenlein, M., Kaplan, A.: A brief history of artificial intelligence: on the past, present, and future of artificial intelligence. *Calif. Manage. Rev.* **61**(4), 5–14 (2019)
2. ENES-UNAM Homepage: <http://www.enesmorelia.unam.mx/>. Last accessed 18 Jan 2021
3. Nilsson, N.J.: Introduction to Machine Learning. Not published, Stanford, CA (1996)
4. Rumelhart, D., Hinton, G., Williams, R.: Learning representations by back-propagating errors. *Nature* **323**, 533–536 (1986)
5. Deng, L., Yu, D.: Deep learning: methods and applications. *FNT Sign. Process.* **7**, 197–387 (2014)
6. PyTorch Homepage: <https://pytorch.org/>. Last accessed 28 Jan 2021
7. Fast.ai Homepage: <https://www.fast.ai/>. Last accessed 28 Jan 2021
8. TensorFlow Homepage: <https://www.tensorflow.org/>. Last accessed 28 Jan 2021
9. Keras Homepage: <https://keras.io/>. Last accessed 28 Jan 2021
10. Deeplearning4j Homepage: <https://deeplearning4j.konduit.ai>. Last accessed 28 Jan 2021
11. El Naqa, I., Murphy, M.J.: What is machine learning? In: El Naqa, I., Li, R., Murphy, M.J. (eds.) *Machine Learning in Radiation Oncology*, pp. 3–11. Springer International Publishing, Cham (2015). https://doi.org/10.1007/978-3-319-18305-3_1
12. Géron, A.: *Hands-on Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems*. O'Reilly Media (2019)
13. Hagerty, J., Stanley R., Stoecker W.: Medical image processing in the age of deep learning. In: *Proceedings of the 12th international joint conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISIGRAPP)*, pp. 306–311 (2017)
14. Basheer, I., Hajmeer, M.: Artificial neural networks: fundamentals, computing, design, and application. *J. Microbiol. Methods* **43**(1), 3–31 (2000)
15. Lévy J., Flórez R., Rodríguez J.: *Las redes neuronales artificiales*, 1a. Edición. Tirant lo Blanch., Netbiblo (2008)
16. Lasi, H., Fettke, P., Feld, T., Hoffmann, M.: Industry 4.0. *Bus. Inform. Syst. Eng.* **6**(4), 239–242 (2014)
17. Jordan, M., Mitchell, T.: Machine Learning: Trends, Perspectives, and Prospects. *Science* **349**(6245), 255–260 (2015)
18. Mohri, M., Rostamizadeh, A., Talwalkar, A.: *Foundations of Machine Learning*. MIT Press (2018)
19. Jagielski, M., Oprea, A., Biggio, B., Liu, C., Nita-Rotaru, C., Li, B.: Manipulating machine learning: poisoning attacks and countermeasures for regression learning. In: *2018 IEEE Symposium on Security and Privacy (SP)*, pp. 19–35 (2018)
20. Li, W., Han, J., Pei, J.: CMAR: Accurate and efficient classification based on multiple class-association rules. In: *Proceedings 2001 IEEE. International Conference on Data Mining*, pp. 369–376 (2001)
21. Raschka, S., Mirjalili, V.: *Python Machine Learning: Machine Learning and Deep Learning with Python, Scikit-learn, and TensorFlow 2*. Packt Publishing Ltd. (2019)
22. Dayan, P., Sahani, M., Deback, G.: Unsupervised Learning. *The MIT Encyclopedia of The Cognitive Sciences*, pp. 857–859 (1999)
23. Wiering, M., van Otterlo, M. (eds.): *Reinforcement Learning*. Springer Berlin Heidelberg, Berlin, Heidelberg (2012)

24. Kaelbling, L., Littman, M., Moore, A.: Reinforcement learning: a survey. *J. Artif. Intell. Res.* **4**, 237–285 (1996)
25. Montgomery, D., Peck, E., Vining, G.: *Introduction to Linear Regression Analysis*. John Wiley & Sons (2021)
26. Kotsiantis, S., Zaharakis, I., Pintelas, P.: Supervised machine learning: a review of classification techniques. *Emerg Artif. Intell. Appl. Comput. Eng.* **160**(1), 3–24 (2007)
27. Celebi, M.E., Aydin, K. (eds.): *Unsupervised Learning Algorithms*. Springer, Cham (2016). <https://doi.org/10.1007/978-3-319-24211-8>
28. McCulloch, W.S., Pitts, W.: A logical calculus of the ideas immanent in nervous activity. *The Bull. Math. Biophys.* **5**(4), 115–133 (1943)
29. LeCun, Y., et al.: Handwritten digit recognition with a back-propagation network. In: Touretzky, D. (ed.) *Advances in Neural Information Processing Systems*, vol. 2. Morgan-Kaufmann (1990)
30. Fukushima, K.: Neocognitron: a self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. *Biol. Cybern.* **36**, 193–202 (1980)
31. Hochreiter, S., Schmidhuber, J.: Long short-term memory. *Neural Comput.* **9**(8), 1735–1780 (1997)
32. He, K., Zhang, X., Ren S., Sun J.: Deep residual learning for image recognition. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778 (2016)
33. Deng, J.: ImageNet: a large-scale hierarchical image database. In: *2009 IEEE conference on computer vision and pattern recognition*. pp. 248–255 (2009)
34. Russakovsky, O., et al.: ImageNet large scale visual recognition challenge. *Int. J. Comput. Vision* **115**(3), 211–252 (2015)
35. Thontirawong, P., Chinchanchokchai, S.: Teaching artificial intelligence and machine learning in marketing. *Mark. Educ. Rev.* **31**(2), 58–63 (2021)
36. Miller E., Ceballos H., Engelmann B., Schiffler A., Batres R., Schmitt J.: Industry 4.0 and International Collaborative Online Learning in a Higher Education Course on Machine Learning, *Machine Learning-Driven Digital Technologies for Educational Innovation Workshop*, pp. 1–8 (2021)
37. Huang L., Ma K.: Introducing machine learning to first-year undergraduate engineering students through an authentic and active learning labware. In: *IEEE Frontiers in Education Conference (FIE)*, pp. 1–4 (2018)
38. Kaggle Homepage: <https://www.kaggle.com/c/titanic/data>. Last accessed 28 Jan 2021
39. Kaggle Homepage: <https://www.kaggle.com/camnugent/california-housing-prices>. Last accessed 28 Jan 2021
40. RePDA Homepage: <https://app.conagua.gob.mx/consultarepda.aspx>. Last accessed 28 Jan 2021
41. Kaggle Homepage: <https://www.kaggle.com/hellbuoy/online-retail-k-means-hierarchical-clustering/data>. Last accessed 28 Jan 2021
42. Dask Homepage: <https://dask.org/>. Last accessed 28 Jan 2021
43. Hadoop Homepage: <https://hadoop.apache.org/>. Last accessed 28 Jan 2021
44. RUOA UNAM Homepage: <https://www.ruoa.unam.mx/index.php?page=estaciones>. Last accessed 28 Jan 2021
45. Kaggle Homepage: <https://www.kaggle.com/adityadesai13/used-car-dataset-ford-and-mercedes>. Last accessed 28 Jan 2021
46. Webmosquito Homepage: <http://basurae.ies.unam.mx/webmosquito/html/>. Last accessed 28 Jan 2021
47. Kaggle Homepage: <https://www.kaggle.com/khoongweihao/covid19-xray-dataset-train-test-sets>. Last accessed 28 Jan 2021
48. Kaggle Homepage: <https://www.kaggle.com/arpitjain007/game-of-deep-learning-ship-datasets>. Last accessed 28 Jan 2021

49. Kaggle Homepage: <https://www.kaggle.com/rmisra/news-headlines-dataset-for-sarcasm-detection>. Last accessed 28 Jan 2021
50. Kaggle Homepage: <https://www.kaggle.com/gpiosenska/sports-classification>. Last accessed 28 Jan 2021
51. EscuelaMLDL Homepage: <http://www.escuelamlldl.enesmorelia.unam.mx/index.php/es/>. Last accessed 28 Jan 2022

Referencias

- Abbasi, B., y Goldenholz, D. M. (2019). Machine Learning Applications in Epilepsy. En (Vol. 60, pp. 2037–2047). Wiley Online Library.
- Adithiyaa, T., Chandramohan, D., y Sathish, T. (2020). Optimal prediction of process parameters by gwo-knn in stirring-squeeze casting of aa2219 reinforced metal matrix composites. *Materials Today: Proceedings*, 21, 1000–1007.
- Aher, S. B., y Lobo, L. (2012). A Comparative Study of Association Rule Algorithms for Course Recommender System in E-learning. *International Journal of Computer Applications*, 39(1), 48–52.
- Ahmad, M., Al-Shayea, N. A., Tang, X.-W., Jamal, A., M Al-Ahmadi, H., y Ahmad, F. (2020). Predicting the pillar stability of underground mines with random trees and c4. 5 decision trees. *Applied Sciences*, 10(18), 6486.
- Alvaredo, F. (2011). A note on the relationship between top income shares and the gini coefficient. *Economics Letters*, 110(3), 274–277.
- Ayodele, T. O. (2010). Types of Machine Learning Algorithms. *New Advances in Machine Learning*, 3, 19–48.
- Barnston, A. G. (1992). Correspondence among the correlation, rmse, and heidke forecast verification measures; refinement of the heidke score. *Weather and Forecasting*, 7(4), 699–709.

- Borja-Robalino, R., Monleón-Getino, A., y Rodellar, J. (2020). Estandarización de métricas de rendimiento para clasificadores machine y deep learning. *Revista Ibérica de Sistemas e Tecnologías de Informação*(E30), 184–196.
- Borthakur, D., y cols. (2008). Hdfs architecture guide. *Hadoop apache project*, 53(1-13), 2.
- Brijain, M., Patel, R., Kushik, M., y Rana, K. (2014). A Survey on Decision Tree Algorithm for Classification.
- Celebi, M. E., y Aydin, K. (2016). *Unsupervised Learning Algorithms*. Springer.
- Chai, T., y Draxler, R. R. (2014). Root mean square error (rmse) or mean absolute error (mae)?—arguments against avoiding rmse in the literature. *Geoscientific model development*, 7(3), 1247–1250.
- Chicco, D., y Jurman, G. (2020). The advantages of the matthews correlation coefficient (mcc) over f1 score and accuracy in binary classification evaluation. *BMC genomics*, 21(1), 1–13.
- Cintra, M. E., Monard, M. C., y Camargo, H. A. (2013). A fuzzy decision tree algorithm based on c4. 5. *Mathware & soft computing*, 20(1), 56–62.
- Cunningham, P., Cord, M., y Delany, S. J. (2008). Supervised Learning. En *Machine Learning Techniques For Multimedia* (pp. 21–49). Springer.
- Davis, J., y Goadrich, M. (2006). The relationship between precision-recall and roc curves. En *Proceedings of the 23rd international conference on machine learning* (pp. 233–240).
- Dayan, P., Sahani, M., y Deback, G. (1999). Unsupervised Learning. *The MIT Encyclopedia of The Cognitive Sciences*, 857–859.
- Derczynski, L. (2016). Complementarity, f-score, and nlp evaluation. En *Proceedings of the tenth international conference on language resources and evaluation (lrec'16)* (pp. 261–266).

- El Naqa, I., y Murphy, M. J. (2015). What is Machine Learning? En *Machine Learning in Radiation Oncology* (pp. 3–11). Springer.
- Géron, A. (2019). *Hands-on Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems*. O'Reilly Media.
- Goutte, C., y Gaussier, E. (2005). A probabilistic interpretation of precision, recall and f-score, with implication for evaluation. En *European conference on information retrieval* (pp. 345–359).
- Haenlein, M., y Kaplan, A. (2019). A Brief History of Artificial Intelligence: On the Past, Present, and Future of Artificial Intelligence. *California Management Review*, 61(4), 5–14.
- Harrell, F. E. (2015). Ordinal logistic regression. En *Regression modeling strategies* (pp. 311–325). Springer.
- Huang, H., Xu, H., Wang, X., y Silamu, W. (2015). Maximum f1-score discriminative training criterion for automatic mispronunciation detection. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 23(4), 787–797.
- Jagielski, M., Oprea, A., Biggio, B., Liu, C., Nita-Rotaru, C., y Li, B. (2018). Manipulating Machine Learning: Poisoning Attacks and Countermeasures for Regression Learning. En *2018 IEEE Symposium on Security and Privacy (SP)* (pp. 19–35).
- Jannach, D., Zanker, M., Felfernig, A., y Friedrich, G. (2010). *Recommender systems: An introduction*. Cambridge University Press.
- Jiang, T., Gradus, J. L., y Rosellini, A. J. (2020). Supervised Machine Learning: A Brief Primer. *Behavior Therapy*, 51(5), 675–687.
- Jordan, M. I., y Mitchell, T. M. (2015). Machine Learning: Trends, Perspectives, and Prospects. *Science*, 349(6245), 255–260.

- Kaelbling, L. P., Littman, M. L., y Moore, A. W. (1996). Reinforcement Learning: A Survey. *Journal of Artificial Intelligence Research*, 4, 237–285.
- Kanungo, T., Mount, D. M., Netanyahu, N. S., Piatko, C. D., Silverman, R., y Wu, A. Y. (2002). An efficient k-means clustering algorithm: Analysis and implementation. *IEEE transactions on pattern analysis and machine intelligence*, 24(7), 881–892.
- Kaplan, J. (2016). *Artificial Intelligence: What Everyone Needs to Know*. Oxford University Press.
- Karun, A. K., y Chitharanjan, K. (2013). A review on hadoop—hdfs infrastructure extensions. En *2013 ieee conference on information & communication technologies* (pp. 132–137).
- Kellman, P., y Hansen, M. S. (2014). T1-mapping in the heart: accuracy and precision. *Journal of cardiovascular magnetic resonance*, 16(1), 1–20.
- Kotsiantis, S. B., Zaharakis, I., y Pintelas, P. (2007). Supervised Machine Learning: A Review of Classification Techniques. *Emerging Artificial Intelligence Applications in Computer Engineering*, 160(1), 3–24.
- Kourou, K., Exarchos, T. P., Exarchos, K. P., Karamouzis, M. V., y Fotiadis, D. I. (2015). Machine Learning Applications in Cancer Prognosis and Prediction. *Computational and Structural Biotechnology Journal*, 13, 8–17.
- Kumari, K., Yadav, S., y cols. (2018). Linear regression analysis study. *Journal of the practice of Cardiovascular Sciences*, 4(1), 33.
- Lee, A., Taylor, P., Kalpathy-Cramer, J., y Tufail, A. (2017). Machine Learning Has Arrived. *Ophthalmology*, 124(12), 1726–1728.
- Li, W., Han, J., y Pei, J. (2001). CMAR: Accurate and Efficient Classification Based on Multiple Class-Association Rules. En *Proceedings 2001 IEEE International Conference on Data Mining* (pp. 369–376).

- Libbrecht, M. W., y Noble, W. S. (2015). Machine Learning Applications in Genetics and Genomics. *Nature Reviews Genetics*, 16(6), 321–332.
- Liu, Y. H. (2020). *Build Intelligent Systems Using Python, TensorFlow2, PyTorch and Scikit-learn*. Packt Publishing Ltd.
- Loh, W.-Y. (2011). Classification and regression trees. *Wiley interdisciplinary reviews: data mining and knowledge discovery*, 1(1), 14–23.
- Loh, W.-Y. (2014). Fifty years of classification and regression trees. *International Statistical Review*, 82(3), 329–348.
- López, F. J. A., Avi, J. R., y Fernández, M. V. A. (2018). Control estricto de matrices de confusión por medio de distribuciones multinomiales. *Geofocus: Revista Internacional de Ciencia y Tecnología de la Información Geográfica*(21), 6.
- Maulud, D., y Abdulazeez, A. M. (2020). A review on linear regression comprehensive in machine learning. *Journal of Applied Science and Technology Trends*, 1(4), 140–147.
- McHugh, M. L. (2013). The chi-square test of independence. *Biochemia medica*, 23(2), 143–149.
- Mitchell, T. (1997). Does Machine Learning Really Work? *AI Magazine*, 18(3), 11–11.
- Mitchell, T., y cols. (1997). *Machine Learning*. McGraw-hill New York.
- Mohri, M., Rostamizadeh, A., y Talwalkar, A. (2018). *Foundations of Machine Learning*. MIT Press.
- Montgomery, D. C., Peck, E. A., y Vining, G. G. (2021). *Introduction to linear regression analysis*. John Wiley & Sons.
- Murdoch, W. J., Singh, C., Kumbier, K., Abbasi-Asl, R., y Yu, B. (2019). Definitions, Methods, and Applications in Interpretable Machine Learning. *Proceedings of the National Academy of Sciences*, 116(44), 22071–22080.

- Myles, A. J., Feudale, R. N., Liu, Y., Woody, N. A., y Brown, S. D. (2004). An Introduction to Decision Tree Modeling. *Journal of Chemometrics: A Journal of the Chemometrics Society*, 18(6), 275–285.
- Na, S., Xumin, L., y Yong, G. (2010). Research on K-means Clustering Algorithm: An Improved K-means Clustering Algorithm. En *2010 Third International Symposium on Intelligent Information Technology and Security Informatics* (pp. 63–67).
- Pandis, N. (2016). The Chi-square Test. *American journal of orthodontics and dentofacial orthopedics*, 150(5), 898–899.
- Pérez, J., Henriques, M., Pazos, R., Cruz, L., Reyes, G., Salinas, J., y Mexicano, A. (2007). Mejora al algoritmo de agrupamiento k-means mediante un nuevo criterio de convergencia y su aplicación a bases de datos poblacionales de cáncer. *Liver-2do Taller Latino Iberoamericano de Investigacion de Operaciones “la IO aplicada a la solución de problemas regionales*, 1–7.
- Raschka, S., y Mirjalili, V. (2019). *Python Machine Learning: Machine Learning and Deep Learning with Python, Scikit-learn, and TensorFlow 2*. Packt Publishing Ltd.
- Rodríguez, A. H., Avilés-Jurado, F. X., Díaz, E., Schuetz, P., Treffer, S. I., Solé-Violán, J., ... others (2016). Procalcitonin (PCT) Levels for Ruling-out Bacterial Coinfection in ICU Patients with Influenza: a CHAID Decision-Tree Analysis. *Journal of infection*, 72(2), 143–151.
- Samuel, A. L. (1959). Some Studies in Machine Learning Using the Game of Checkers. *IBM Journal of Research and Development*, 3(3), 210–229.
- Sathya, R., y Abraham, A. (2013). Comparison of Supervised and Unsupervised Learning Algorithms for Pattern Classification. *International Journal of Advanced Research in Artificial Intelligence*, 2(2), 34–38.
- Siau, K., y Wang, W. (2018). Building Trust in Artificial Intelligence, Machine Learning, and Robotics. *Cutter Business Technology Journal*, 31(2), 47–53.

- Singh, S., y Gupta, P. (2014). Comparative study id3, cart and c4. 5 decision tree algorithm: a survey. *International Journal of Advanced Information Science and Technology (IJAIST)*, 27(27), 97–103.
- Song, Y., Huang, J., Zhou, D., Zha, H., y Giles, C. L. (2007). Iknn: Informative k-Nearest Neighbor Pattern Classification. En *European Conference on Principles of Data Mining and Knowledge Discovery* (pp. 248–264).
- Su, J., y Zhang, H. (2006). A Fast Decision Tree Learning Algorithm. En *AAAI* (Vol. 6, pp. 500–505).
- Sutton, R. S., y Barto, A. G. (2018). *Reinforcement Learning: An Introduction*. MIT Press.
- Timofeev, R. (2004). Classification and regression trees (cart) theory and applications. *Humboldt University, Berlin*, 1–40.
- Turing, A. (1950). Computing Machinery and Intelligence. *Mind*, 59(236), 433.
- van Zoonen, W., y Toni, G. (2016). Social Media Research: The Application of Supervised Machine Learning in Organizational Communication Research. *Computers in Human Behavior*, 63, 132–141.
- Vlachos, M., Domeniconi, C., Gunopulos, D., Kollios, G., y Koudas, N. (2002). Non-linear Dimensionality Reduction Techniques for Classification and Visualization. En *Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 645–651).
- Wang, S.-C. (2003). Artificial Neural Network. En *Interdisciplinary Computing in Java Programming* (pp. 81–100). Springer.
- Wiering, M., y Van Otterlo, M. (2012). Reinforcement Learning. *Adaptation, Learning, and Optimization*, 12(3).

- Zhang, S., Li, X., Zong, M., Zhu, X., y Cheng, D. (2017). Learning k for knn classification. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 8(3), 1–19.