

Universidad Católica de Santa María
Facultad de Ciencias e Ingenierías Físicas y
Formales
Escuela Profesional de Ingeniería de Sistemas



**PREDICCIÓN DEL ESTADO DE CARGA DE BATERIAS PARA SISTEMAS
FOTOVOLTAICOS EN EL SECTOR MINERO UTILIZANDO TÉCNICAS DE
APRENDIZAJE AUTOMÁTICO SUPERVISADO**

Tesis presentada por el Bachiller:

Apaza Pinto, Alexa Xiomara

Para optar por el título
profesional:

**Ingeniero de Sistemas con
Especialidad en Sistemas de
Información**

Asesor:

Dr. Sulla Torres, José Alfredo

Arequipa – Perú

2021

DICTAMEN APROBATORIO

UCSM-ERP

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

INGENIERIA DE SISTEMAS

TITULACIÓN CON TESIS

DICTAMEN APROBACIÓN DE BORRADOR

Arequipa, 01 de Septiembre del 2021

Dictamen: 001661-C-EPIS-2021

Visto el borrador del expediente 001661, presentado por:

2013200212 - APAZA PINTO ALEXA XIOMARA

Titulado:

**PREDICCIÓN DEL ESTADO DE CARGA DE BATERIAS PARA SISTEMAS FOTOVOLTAICOS EN EL
SECTOR MINERO UTILIZANDO TÉCNICAS DE APRENDIZAJE AUTOMÁTICO SUPERVISADO**

Nuestro dictamen es:

APROBADO

**1561 - GUEVARA PUENTE DE LA VEGA KARIM
DICTAMINADOR**



**1635 - SULLA TORRES JOSE ALFREDO
DICTAMINADOR**



**2820 - ESQUICHA TEJADA JOSE DAVID
DICTAMINADOR**



PRESENTACIÓN

Señor Decano de la Facultad de Ciencias e Ingenierías Físicas y Formales.

Señor Director de la Escuela Profesional de Ingeniería de Sistemas.

Señores Miembros del Jurado Dictaminador de la Tesis.

De conformidad con las disposiciones del Reglamento de Grados y Títulos de la Escuela Profesional de Ingeniería de Sistemas, pongo a vuestra consideración el Presente trabajo de investigación titulado: **“PREDICCIÓN DEL ESTADO DE CARGA DE BATERIAS PARA SISTEMAS FOTOVOLTAICOS EN EL SECTOR MINERO UTILIZANDO TÉCNICAS DE APRENDIZAJE AUTOMÁTICO SUPERVISADO”**, el mismo que al ser aprobado me permitirá optar por el Título Profesional de Ingeniería de Sistemas.

APAZA PINTO ALEXA XIOMARA.

DEDICATORIA

El presente trabajo de investigación se lo agradezco a Dios por guiarme en mi camino y por permitirme concluir con mi objetivo.

A mi madre Yeny y a mi hermanita Samantha por ayudarme y apoyarme incondicionalmente en cada meta, sueño y objetivo planteado, las amo mucho y gracias por ser mi fuerza y alentarme a seguir cuando pensaba que no podría más. Ustedes fueron el pilar para llegar hasta aquí y convertirme en lo que soy.

A mi padre José del cual estoy agradecida y valoro el apoyo brindado en mi educación.

A mis hermosas mascotas Molly y Zoé que me hacen feliz todos los días, también a ti Melissa mi fiel compañera en mi etapa universitaria gracias pequeña sé que siempre seguirás presente cuidándome desde el cielo.

A mi enamorado José por ser un gran apoyo en mi vida, por darme los ánimos para que siga adelante con este proyecto y no me rinda, siempre confiando en mí. Y como me dijiste esfuérate y trázate un objetivo; créeme que este es el inicio de más metas a cumplir.

A mis mejores amigos Christian y Valia, que se convirtieron en más que amigos mi familia, a quienes los valoro y aprecio mucho.

A mis amigos de mi etapa universitaria aquellos que serán mis colegas, gracias por todo esos momentos vividos llenos de alegría.

AGRADECIMIENTOS



Le agradezco al Dr. Paredes Marchena por guiarme al inicio de este proyecto, al Dr. Sulla Torres por sus conocimientos y asesoría que me brindó, al Lic. Juan Pasten Barahona por su gran apoyo y confianza; y a todas las personas que me ayudaron durante toda la elaboración de este trabajo de investigación.

RESUMEN

En el presente trabajo de investigación propuesto tiene como objetivo la predicción del Estado de Carga de Baterías de equipos energizados por paneles solares en sector minero, mediante técnicas de aprendizaje supervisado automático.

Se cuenta con un sistema de monitoreo que registra cada variable de energía programado en el sistema fotovoltaico, por el cual se realizó un análisis de los datos extraídos del sistema de monitoreo. Los datos fueron evaluados mediante técnicas de aprendizaje supervisado automático por medio de la herramienta RapidMiner cuyo promedio de predicción fue un 90.12 %.

Se eligió la técnica de aprendizaje supervisado automático más adecuada para la predicción del estado de carga de las baterías para sistemas fotovoltaicos, cuya técnica elegida fue desarrollada utilizando la librería Tensorflow y el lenguaje de programación Python.

Se hizo un análisis y discusión de los resultados del entrenamiento del modelo que finalmente con el modelo correctamente entrenado se procedió a realizar una validación que conllevó a la comparación de los datos predictivos con los datos en tiempo real obteniendo una buena relación y resultados satisfactorios.

Palabras Claves

Predicción, Estado de Carga, Sistemas Fotovoltaicos, Aprendizaje Automático Supervisado.

ABSTRACT

The objective of this proposed research work is to predict the Battery Charge Status of equipment powered by solar panels in the mining sector, using automatic supervised learning techniques.

There is a monitoring system that records each energy variable programmed in the photovoltaic system, through which an analysis of the data extracted from the monitoring system was carried out. The data were evaluated using supervised automatic learning techniques using the RapidMiner tool, whose average prediction was 90.12%.

The most suitable supervised automatic learning technique was chosen for the prediction of the state of charge of batteries for photovoltaic systems, whose chosen technique was developed using the Tensorflow library and the Python programming language.

An analysis and discussion of the results of the training of the model was made, and finally, with the correctly trained model, a validation was carried out that led to the comparison of the predictive data with the data in real time, obtaining a good relationship and satisfactory results.

Keywords

Prediction, State of Charge, Photovoltaic systems, Supervised Machine Learning.

INTRODUCCIÓN

El sector energético es indispensable para el ser humano de tal forma que su ausencia provocaría inconvenientes tales como la realización de diferentes actividades en la vida diaria. Sin embargo, en varias partes del mundo no disponen mencionado servicio.

Hoy en día una de las soluciones más viables tanto en lo técnico y financiero para satisfacer las necesidades del uso de la energía, se les designe a las generadas por los sistemas de energía fotovoltaico.

Las energías fotovoltaicas son tecnologías que con el pasar del tiempo ha tenido disminución en sus costos de mantenimiento, creadoras de ningún tipo de emisiones de carácter perjudicial para el medio ambiente, con gran facilidad y enlace en fuentes ya existentes de energía en el lugar de instalación. (Bitar S. & Chamas B., 2017)

A la interconexión de varias celdas fotovoltaicas se le llama panel fotovoltaico cuya función es transformar la radiación solar en energía eléctrica. Una agrupación de paneles da lugar a un módulo, que en conjunto con las baterías que reservan la energía eléctrica, el inversor, el conmutador y los cables, conforman el sistema fotovoltaico.

Las baterías solares almacenan la energía eléctrica generada por las placas solares fotovoltaicas para así poderla emplear durante la noche e incluso en días nublados, no obstante, resulta fundamental poder estimar con precisión el estado de carga en las baterías (SOC).

La estimación del estado de carga (SOC) es un enlace clave para el desarrollo de sistemas de almacenamiento de energía de baterías, en este trabajo de investigación se hará el uso de aprendizaje automático mediante la evaluación de los resultados efectuados por diferentes algoritmos de aprendizaje supervisado para la predicción del estado de carga en las baterías en sistemas fotovoltaicos.

El trabajo de investigación está constituido por 4 capítulos en los que se tratan los siguientes temas:

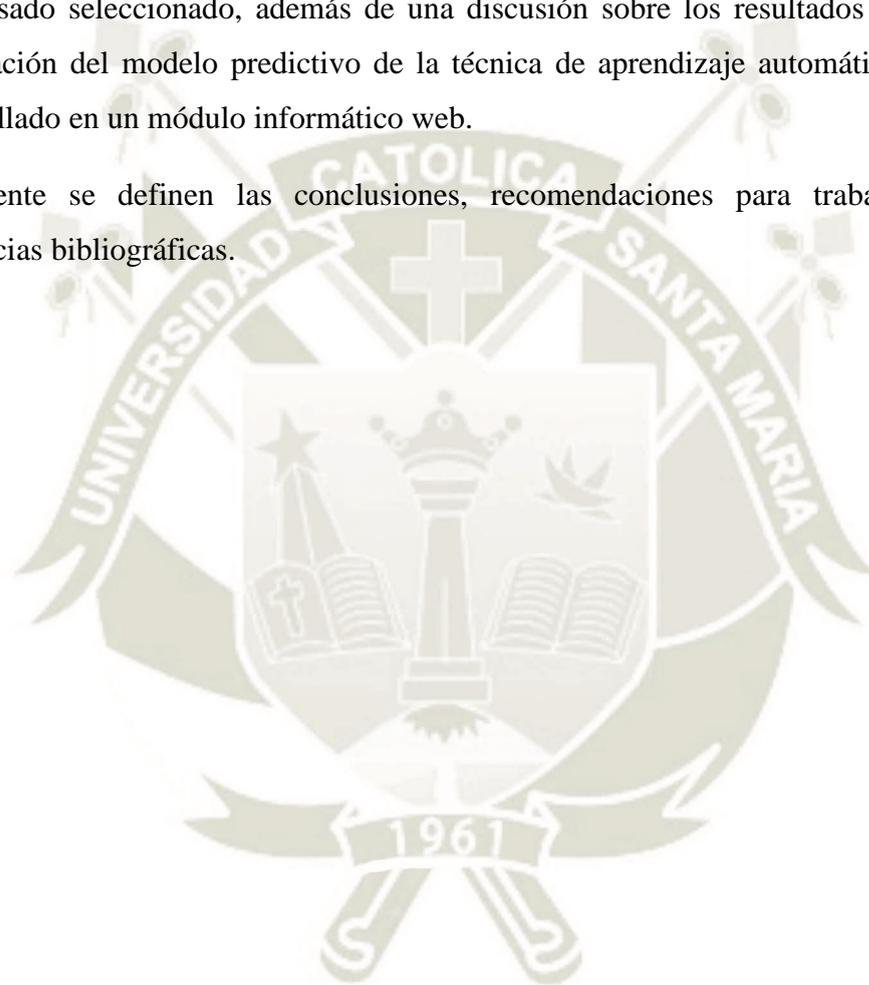
En el capítulo 1, se inicia con el desarrollo del marco referencial que explica el planteamiento del problema, objetivos del trabajo de investigación, las bases teóricas de la investigación además de una solución propuesta para el problema a tratar.

En el capítulo 2, el marco teórico del proyecto, describiendo el estado del arte y las bases teóricas del proyecto.

En el capítulo 3, se presenta el proceso de construcción, evaluación de las técnicas de aprendizaje automático supervisado y la selección de la técnica de aprendizaje automático supervisado más adecuada a los objetivos planteados

En el capítulo 4, se muestran los resultados de la técnica de aprendizaje automático supervisado seleccionado, además de una discusión sobre los resultados obtenidos y la importación del modelo predictivo de la técnica de aprendizaje automático supervisado desarrollado en un módulo informático web.

Finalmente se definen las conclusiones, recomendaciones para trabajos futuros y referencias bibliográficas.



ÍNDICE

DICTAMEN APROBATORIO.....	I
PRESENTACIÓN	II
DEDICATORIA	III
AGRADECIMIENTOS	IV
RESUMEN	V
ABSTRACT	VI
INTRODUCCIÓN.....	VII
CAPÍTULO I	1
1. PLANTEAMIENTO TEÓRICO	1
1.1 PLANTEAMIENTO DE LA INVESTIGACIÓN	1
1.1.1 Planteamiento del Problema.....	1
1.1.2 Objetivos de la Investigación	2
1.1.3 Preguntas de Investigación.....	3
1.1.4 Línea y Sub-Línea de Investigación.....	3
1.1.5 Palabras Claves	3
1.1.6 Solución Propuesta.....	3
1.2 FUNDAMENTOS TEÓRICOS.....	5
1.2.1 Estado del Arte	5
1.3 MARCO METODOLÓGICO.....	8
1.3.1 Bases Teóricas de la Investigación.....	8
1.4 ALCANCES Y LIMITACIONES	10
1.4.1 Alcances	10
1.4.2 Limitaciones	11
1.4.3 Tipo y Nivel de Investigación	11
1.4.4 Población y Muestra Metodológica.....	12
1.4.5 Métodos, Técnicas e Instrumentos	12
1.4.5.1 Métodos.....	12
1.4.5.2 Técnicas	12
1.4.5.3 Instrumento y Recolección de Datos	13
1.4.6 Metodología de desarrollo del trabajo.....	13
CAPÍTULO II.....	15
2. PLANTEAMIENTO TEÓRICO	15

2.1	INTELIGENCIA ARTIFICIAL	15
2.1.1	Funcionamiento de la Inteligencia Artificial.....	18
2.2	APRENDIZAJE AUTOMÁTICO	19
2.2.1	Fundamentos del Aprendizaje Automático	20
2.2.1.1	El aprendizaje.....	20
2.2.1.2	El error y los problemas frecuentes en el Aprendizaje Automático	21
2.2.2	Importancia del Aprendizaje Automático	24
2.2.3	Aprendizaje Automático Supervisado.....	25
2.2.4	Aprendizaje Automático No Supervisado.....	27
2.2.5	Aprendizaje Automático por Refuerzo	28
2.3	TENSORFLOW.....	30
2.3.1	Características del Tensorflow	31
2.3.1.1	Compilación sencilla de Modelos	31
2.3.1.2	Producción de Aprendizaje Automático sólido en cualquier parte.....	32
2.3.1.3	Importante experimentación para la investigación	32
2.3.2	Tensorflow frente a otras bibliotecas	33
2.4	LENGUAJE DE PROGRAMACIÓN PYTHON	34
2.4.1	Características del Lenguaje de Programación Python	34
2.4.2	Inteligencia Artificial en Python	35
2.5	RAPIDMINER.....	35
2.5.1	Fortalezas de la herramienta RapidMiner	36
2.5.1.1	Gestión del ciclo de vida de la ciencia de datos aumentada de extremo a extremo.....	36
2.5.1.2	Flexibilidad, agilidad y escalabilidad moderna	36
2.5.1.3	Gobernanza de modelos	37
2.6	SISTEMAS FOTOVOLTAICOS	38
2.6.1	Tipos de Sistemas Fotovoltaicos	39
2.6.1.1	Sistema Fotovoltaico aislado (SFA)	39
2.6.1.2	Sistema Fotovoltaicos conectados a red (SFRC).....	40
2.6.1.3	Sistema Híbridos Fotovoltaicos	40
2.6.2	Baterías en Sistemas Fotovoltaicos	40
2.7	ESTADO DE CARGA (SoC) %.....	43
2.7.1	Estimación del SoC basada en la Intensidad.....	43
2.8	CRISP-DM.....	44
2.8.1	Comprensión del negocio.....	45
2.8.2	Comprensión de los datos	45
2.8.3	Preparación de los datos	45
2.8.4	Modelamiento.....	45
2.8.5	Evaluación.....	46
2.8.6	Despliegue.....	46
CAPÍTULO III		47
3. CONSTRUCCIÓN Y EVALUACIÓN DE LAS TÉCNICAS DE APRENDIZAJE AUTOMÁTICO SUPERVISADO		47
3.1	FASES.....	48

3.1.1	Comprensión del negocio.....	48
3.1.2	Comprensión de los datos	51
3.1.3	Preparación de datos.....	53
3.1.4	Modelos.....	54
3.1.4.1	Decision Tree	57
3.1.4.2	Random Forest	58
3.1.4.3	Gradient-boosted.....	60
3.1.4.4	Neuronal Network.....	62
3.1.4.5	Support Vector Machine	63
3.1.5	Evaluación.....	65
3.1.5.1	Resultados de las técnicas de aprendizaje automático supervisado.....	65
3.1.5.2	Análisis y discusión de los Resultados de las técnicas supervisadas de aprendizaje automático.....	66
3.1.6	Desarrollo.....	68
3.1.6.1	Despliegue a producción de la Red Neuronal (Neuronal Network)	69
3.1.6.1.1	Lectura de los Datos	69
3.1.6.1.2	Limpieza de los Datos	70
3.1.6.1.3	División de los Datos.....	70
3.1.6.1.4	Inspección de los datos	71
3.1.6.1.5	Separación del valor objetivo	72
3.1.6.1.6	Normalización	74
3.1.6.1.7	Construcción de la red neuronal	74
3.1.6.1.8	Entrenamiento de la red neuronal	76
3.1.6.1.9	Predicción de la red neuronal.....	77
3.1.6.1.10	Importación del Modelo en Tensorflow.js.....	77
3.1.6.1.11	Módulo Informático Web con Importación del Modelo en Tensorflow.js	79
CAPÍTULO IV		81
4.	RESULTADOS.....	81
4.1	Resultados del Entrenamiento del Modelo de la Red Neuronal	81
4.2	Análisis y discusión de los resultados del entrenamiento del modelo	84
4.3	Pruebas del módulo informático web con importación del modelo en Tensorflow.JS	86
CONCLUSIONES.....		88
RECOMENDACIONES Y TRABAJOS FUTUROS.....		91
ANEXO.....		95
Anexo A: Plan de Tesis		95

ÍNDICE DE TABLAS

Tabla 1 Lista de algoritmos de Aprendizaje Automático Supervisado.....	6
Tabla 2 Fases de la metodología CRISP-DM	14
Tabla 3 Trabajos recientes en el sector de energía para los sistemas fotovoltaicos aplicando inteligencia artificial	16
Tabla 4 Tensorflow frente a otras bibliotecas para el aprendizaje Automático	33
Tabla 5 Tipos de baterías en sistemas fotovoltaicos	41
Tabla 6 Comparación de Resultados de las Técnicas de Aprendizaje Automático Supervisado seleccionadas	66
Tabla 7 Aplicaciones de Google Colaboratory	68
Tabla 8 Total de atributos del conjunto de datos.	69
Tabla 9 Categorías de la Evaluación del Modelo.....	85



ÍNDICE DE FIGURAS

Figura 1 Relación entre artificial intelligence, machine learning y deep Learning	9
Figura 2 Aprendizaje Automático	10
Figura 3 La implementación de la IA en diferentes sectores	16
Figura 4 Funcionamiento de la IA.....	18
Figura 5 Esquema generalizado de los tres subconjuntos existentes de Machine Learning	20
Figura 6 La programación clásica frente al Machine Learning	21
Figura 7 Nivel de complejidad del modelo en función del error	22
Figura 8 Distintos ajustes para los problemas de regresión y clasificación	24
Figura 9 Modelos de Regresión y Clasificación	26
Figura 10 Aprendizaje automático no supervisado	28
Figura 11 Aprendizaje automático por refuerzo	29
Figura 12 Niveles de APIs en Tensorflow	31
Figura 13 Cuadrante mágico para plataformas de ciencia de datos y aprendizaje automático	38
Figura 14 Esquema equivalente al banco de baterías en un sistema fotovoltaico.....	42
Figura 15 Fórmula para hallar el Estado de Carga (SoC)	44
Figura 16 Fases de la metodología CRISP – DM	44
Figura 17 Fases para la construcción y evaluación de técnicas de aprendizaje automático supervisado basado en la metodología CRISP-DM	47
Figura 18 Actividades de la Fase para la comprensión del negocio	48
Figura 19 Repetidor móvil	50
Figura 20 Hoja de Cálculo - VE Data Export	51
Figura 21 Ventana de Inicio de la herramienta RapidMiner	55
Figura 22 Importación de los Datos en la herramienta RapidMiner	56
Figura 23 Datos importados correctamente en la Herramienta RapidMiner	57
Figura 24 Decision Tree	58
Figura 25 Técnica de aprendizaje automático supervisado Decision Tree modelado en la herramienta RapidMiner.....	58
Figura 26 Random Forest.....	59
Figura 27 Técnica de aprendizaje automático supervisado Random Forest modelado en la herramienta RapidMiner.....	60
Figura 28 Gradient-boosted.....	61
Figura 29 Técnica de aprendizaje automático supervisado Gradient Boosted modelado en la herramienta RapidMiner	61
Figura 30 Neuronal Network	63
Figura 31 Técnica de aprendizaje automático supervisado Neuronal Network modelado en la herramienta RapidMiner	63
Figura 32 Support Vector Machine	64
Figura 33 Técnica de aprendizaje automático supervisado Support Vector Machine modelado en la herramienta Rapidminer	65

Figura 34 Visualización de las primeras 4 filas con uso de la función dataframe.head ()	70
Figura 35 Inspección del Conjunto de Datos.	71
Figura 36 Información estadística del valor objetivo con uso de la función describe().....	73
Figura 37 Información estadística de las características con uso de la función describe().....	73
Figura 38 Datos Normalizados.....	74
Figura 39 Construcción de la red neuronal	76
Figura 40 Entrenamiento de la red neuronal	76
Figura 41 Predicción de la red neuronal.....	77
Figura 42 Biblioteca Tensorflow JS.....	77
Figura 43 Conversión del modelo	78
Figura 44 Modelo Convertido	78
Figura 45 Modelo cargado en JS.....	79
Figura 46 Módulos informáticos de la plataforma web	80
Figura 47 Módulo informático web para el estado de carga (SoC)	80
Figura 48 Resultados del entrenamiento del modelo	82
Figura 49 Curvas de error.....	82
Figura 50 Evaluación del modelo para el conjunto de datos de prueba.....	83
Figura 51 Resultado de las predicciones	83
Figura 52 Datos reales vs datos predictivos	84
Figura 53 Primera prueba con datos reales para la predicción del estado de carga (SoC)	87
Figura 54 Segunda prueba con datos reales para la predicción del estado de carga (SoC)	87

CAPÍTULO I

1. PLANTEAMIENTO TEÓRICO

1.1 PLANTEAMIENTO DE LA INVESTIGACIÓN

1.1.1 Planteamiento del Problema

En esta 4ta etapa de la revolución industrial la tecnología va evolucionando rápidamente y para empresas que brindan servicios a industrias mineras deben desarrollar nuevos modelos de negocios basados en innovaciones tecnológicas.

La empresa de rubro de telecomunicaciones donde se presentó la problemática es una empresa cuyo enfoque es el diseño y la implementación de soluciones tecnológicas que permiten dar soporte a las necesidades de conectividad operacional en minería, añadiendo valor al cliente mediante el manejo de la información. Siendo el enfoque de la empresa las soluciones tecnológicas, por lo cual implementaron equipos que brindan cobertura para el área operacional minero mediante la red Airpro (Red Multiservicio).

Para cada equipo que se encuentra instalado en varias zonas del área operacional su flujo de energía es mediante baterías las cuales son energizadas por paneles solares. Sin embargo, en los últimos meses fue creciendo el número de incidencias por falla des energización en las baterías de los equipos, esto debido a que no se cuenta con un sistema que analice el estado de carga de las baterías y solo era gestionado manualmente por el personal. Las consecuencias que puede traer este problema son varias, pero

la más crítica de todas, es en el factor económico porque cuando un equipo se encuentra desenergizado ocasiona el paro de operación en mina.

La importancia de este proyecto es dar a conocer la predicción del estado de carga de las baterías de los equipos haciendo uso de la Inteligencia Artificial mediante el subcampo del Aprendizaje Automático para así prevenir incidencias a largo plazo que perjudiquen la operación en mina y planificar mantenimientos preventivos si es necesario.

1.1.2 Objetivos de la Investigación

a) General

Predecir el estado de carga de las baterías para sistemas fotovoltaicos en el sector minero utilizando técnicas de aprendizaje automático supervisado.

b) Específicos

- Analizar el proceso de la toma de datos de las variables de energía de los equipos actuales de la empresa.
- Aplicar técnicas de aprendizaje automático supervisado que permitan predecir con gran exactitud el estado de carga de las baterías de los equipos.
- Determinar la técnica de aprendizaje automático supervisado más adecuada para la predicción del estado de carga de las baterías de los equipos.
- Validación de los resultados en la predicción del estado de carga de las baterías para sistemas fotovoltaicos.
- Desarrollar un módulo informático web que integre el aprendizaje automático supervisado propuesto.

1.1.3 Preguntas de Investigación

- ¿Puede Machine Learning mejorar la precisión de resultados para el estado de carga?
- ¿Los algoritmos de aprendizaje automático supervisado pueden predecir el estado de carga a partir de sus datos de ingreso?
- ¿Qué algoritmo de aprendizaje automático supervisado facilitará el análisis de los datos?

1.1.4 Línea y Sub-Línea de Investigación

- **Línea de Investigación:** Inteligencia Artificial.
- **Sub-línea de Investigación:** Aprendizaje Automático.

1.1.5 Palabras Claves

Predicción, Estado de Carga, Sistemas Fotovoltaicos, Aprendizaje Automático Supervisado.

1.1.6 Solución Propuesta

a) **Justificación e Importancia**

El sector fotovoltaico en los últimos años se ha convertido en una de las mayores industrias a nivel mundial utilizando la energía solar para la reducción de costos de energía.

La Empresa de rubro de telecomunicaciones donde se presentó la problemática. es participe en la implementación de los sistemas fotovoltaicos para la energización de sus equipos que brindan cobertura para

la operatividad en sectores mineros. Cada equipo cuenta con baterías energizadas por paneles solares y son monitoreadas en un Sistema de Monitoreo que registra cada variable de energía programado en el sistema fotovoltaico; sin embargo, las personas encargadas de la toma de datos no realizan un buen análisis de las variables de energía, por lo que no se puede tener un adecuado control del estado de carga de las baterías en los equipos. Teniendo como consecuencia la desenergización de las baterías implicando así que la operación en mina pare por un lapso de tiempo hasta que las baterías pasen por un mantenimiento y se vuelvan a energizar.

Hay que considerar que cuando se para la operación en mina por un lapso se generan grandes pérdidas en lo económico.

Por tal motivo la predicción del estado de carga de las baterías para los sistemas fotovoltaico sería de gran utilidad para la operación ya que determinaría resultados con más precisión y un mejor análisis para las variables de energía.

b) Descripción de la Solución

Con las técnicas de aprendizaje automático se podrá saber la predicción del estado de carga de las baterías en sistemas fotovoltaicos obteniendo resultados en tiempo real. Con esta información personal del área de monitoreo y personal de terreno en mina podrá visualizar y analizar las tendencias en el estado de carga de las baterías de sus equipos.

1.2 FUNDAMENTOS TEÓRICOS

1.2.1 Estado del Arte

En esta 4ta etapa de la revolución industrial en las que nos encontramos actualmente las tecnologías de la información tienen como propósito llevar a cabo un cambio digital que las haga inteligentes.

El aprendizaje automático es una de las áreas de la informática de más rápido crecimiento, con aplicaciones de gran alcance. Se refiere a la detección automatizada de patrones significativos en los datos. Las herramientas de aprendizaje automático se preocupan por dotar a los programas de la capacidad de aprender y adaptarse. (Shai Shalev-Shwartz, Shai Ben-David, 2014)

El aprendizaje automático entrega la garantía de extraer importantes fuentes de datos grandes y distintas a través de una menor dependencia periférica programada en una pista individual, ya que se determinan los datos y se dispara a escala de máquina.

El aprendizaje automático es adecuado para la complejidad del manejo a través de un origen de datos diferente y la amplia gama de variables, así como la cantidad de datos involucrados donde el aprendizaje automático prospera en conjuntos de datos crecientes. Cuanto más se suministren datos a una estructura de aprendizaje automático, más se podrá capacitar y se tendrán en cuenta las consecuencias del valor superior de los conocimientos.

Al estar libre de los confines del pensamiento y el estudio a nivel individual,

Machine Learning es inteligente para descubrir y mostrar los patrones ocultos en los datos. (Pradeep K R, Dr.Naveen N C, 2017)

Partiendo de estos conceptos sobre el aprendizaje automático es posible extraer y almacenar una gran cantidad de datos. No obstante, se requiere la aplicación de algoritmos de aprendizaje automático para crear modelos que den valor a esos datos y hagan de mayor facilidad la toma de decisiones.

Las técnicas de aprendizaje supervisado se enfocan en utilizar una combinación etiquetada de datos para entrenar un algoritmo que sirva al propósito de la mejor función para describir la selección de datos de entrada.

La técnica de aprendizaje no supervisado utiliza datos sin etiquetar para entrenar el algoritmo. Con el tiempo, ayuda a detectar un patrón al describir un modelo. En tal caso, los datos de entrada ayudan al algoritmo a extraer reglas y regulaciones junto con la detección de patrones relevantes. Los puntos importantes se resumen y combinan con el tiempo para proporcionar significado y una mejor comprensión. (Shanthamallu, Uday Shankar, Andreas Spanias, Cihan, 2017)

ML Algorithms	Supervised Learning Method	<i>Tabla 1 Lista de algoritmos de Aprendizaje Automático Supervisado</i>
K-Nearest Neighbors	Supervised ML	
Decision Tree and Random Forest	Supervised ML	
Support Vector Machines (SVM)	Supervised ML	
Recurrent Neural Networks (RNN)	Supervised ML	

Fuente: Dalal (2020)

Cabe concluir que las técnicas de aprendizaje supervisado son empleados en modelos predictivos utilizando algoritmos que aprenden de los datos de entrada generando datos de salida esperados siendo la entrada etiquetada y clasificada por alguien y por otro lado las técnicas de aprendizaje no supervisado donde los algoritmos aprenden de datos no etiquetados buscando patrones o relación entre ellos.

La tecnología de estimación del estado de carga (SoC) es importante para el desarrollo de sistemas de almacenamiento de energía de baterías de iones de litio (H. Rahimi-Eichi, U. Ojha, F. Baronti, and M. Chow, 2013).

Un SoC preciso es el elemento principal para el funcionamiento seguro de los paquetes de baterías. También es una base de referencia para el control de ecualización celular. La función de un equilibrador de celdas es mantener todas las celdas igualadas durante la carga y descarga para maximizar la capacidad del paquete y proteger las celdas de sobrecarga o descarga. Puede provocar daños permanentes cuando una celda se sobrecarga o se descarga en exceso. Por lo tanto, mejorar la precisión del SoC estimado hace una diferencia en la mitigación del daño de la celda, prolongando el ciclo de vida y disminuyendo el costo de mantenimiento. (Zhihao Wang, 2016)

La toma de data de las variables de energía para los equipos que brindan cobertura radial las cuales se encuentran en mina pueden ser muy útiles para estimar la predicción del estado de carga de las baterías utilizando técnicas de aprendizaje automático supervisado teniendo como datos de entradas el

voltaje de las baterías, temperatura de las baterías, corriente consumo, corriente carga y corriente paneles.

Este proyecto estará enfocado en el desarrollo de la predicción del estado de carga de las baterías para sistemas fotovoltaicos utilizando técnicas de aprendizaje automático supervisado e integrarlo en un módulo informático web.

1.3 MARCO METODOLÓGICO

1.3.1 Bases Teóricas de la Investigación

a) Inteligencia Artificial

La Inteligencia Artificial tiene por objetivo el estudio y el análisis del comportamiento humano en los ámbitos de la comprensión, la percepción, la resolución de problemas y la toma de decisiones con el fin de poder reproducirlos con la ayuda de un computador, esta simulación de inteligencia humana incluye procesos de aprendizaje de razonamiento y de mejoramiento.

La Inteligencia Artificial ha estructurado una plataforma tecnocientífica mediante la puesta en marcha de una disciplina transversal que, tiene por finalidad entender, modelar y replicar los procesos cognitivos típicos de la inteligencia humana integrando en su esfuerzo principios propios de la matemática, la lógica, la biología y la mecánica, en aras de incorporarlos a máquinas auto controladas. (Villalba Gómez, 2016)

El diagrama de la Figura 1 muestra cómo se subdivide el campo de la IA, incluido el aprendizaje automático y sus subsecuentes divisiones de aprendizaje de representación y aprendizaje profundo.

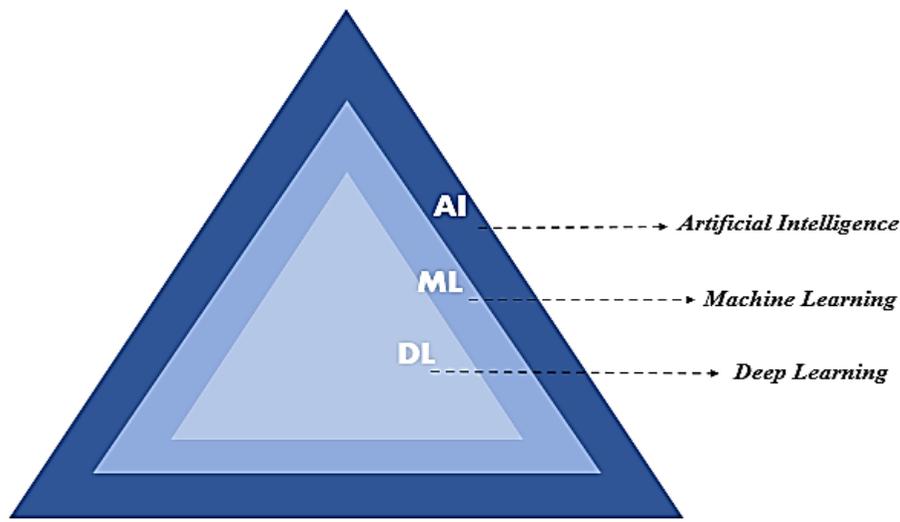


Figura 1 Relación entre artificial intelligence, machine learning y deep Learning

Fuente: Elaboración Propia

b) Aprendizaje Automático

El aprendizaje automático consiste en programar computadoras para optimizar un criterio de rendimiento utilizando datos de ejemplo o experiencias pasadas. Tenemos un modelo definido hasta algunos parámetros y el aprendizaje es la ejecución de un programa informático para optimizar los parámetros del modelo utilizando los datos de entrenamiento o la experiencia pasada. (Alpaydin, 2020)

En la Figura 2 se muestra un ejemplo del proceso del aprendizaje automático donde primero se iniciará con las entradas de los datos, segundo se encontrará las características para describir las instancias, tercero se evaluará la técnica de aprendizaje automático más adecuado y cuarto y último se obtendrá una salida el cual serán los resultados de la predicción.

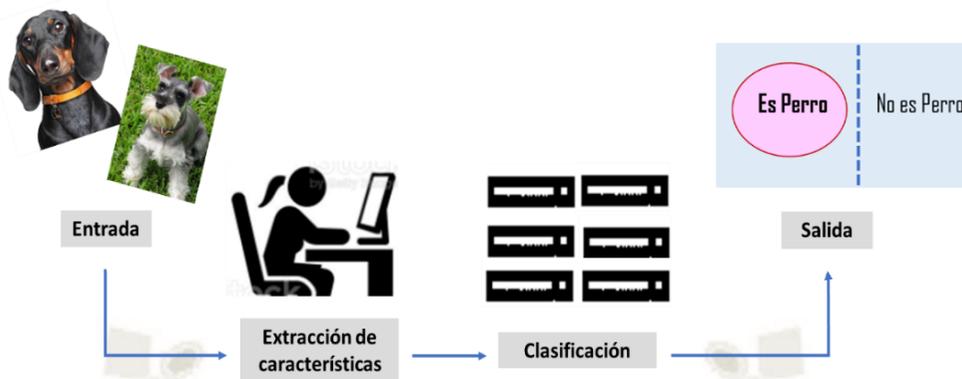


Figura 2 Aprendizaje Automático

Fuente: Elaboración Propia

c) Tensorflow

Tensorflow es una plataforma de código abierto de extremo a extremo para el aprendizaje automático. Cuenta con un ecosistema integral y flexible de herramientas, bibliotecas y recursos de la comunidad que les permite a los investigadores impulsar un aprendizaje automático innovador y, a los desarrolladores, compilar e implementar con facilidad aplicaciones con tecnología de Aprendizaje Automático. (MartIn Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, 2015)

1.4 ALCANCES Y LIMITACIONES

1.4.1 Alcances

El trabajo de investigación tiene la finalidad de predecir el estado de carga de las baterías para los sistemas fotovoltaicos utilizando un análisis continuo con el uso de algoritmos de aprendizaje automático supervisado para la predicción de fallas futuras.

La investigación tomará en cuenta los datos de ingreso a partir del Sistema Victron Energy quien se encarga de registrar las variables de energía de los Sistemas Fotovoltaicos, el cual permitirá que se pueda extraer la información necesaria.

1.4.2 Limitaciones

La investigación está limitada al análisis de la data histórica de las variables de energía registrada por el Sistema Victron Energy, el periodo comprendido de la data es a partir del mes de mayo del 2020, con el objetivo de poder realizar el modelo de Machine Learning con esta información.

1.4.3 Tipo y Nivel de Investigación

a) Tipo de Investigación

La presente investigación es de tipo aplicada, ya que el propósito principal es el de actuar sobre el estado de carga de las baterías analizando constantemente la información necesaria proveniente del sistema fotovoltaico, aplicando los algoritmos de aprendizaje automático necesarios para encontrar algún patrón.

b) Nivel de Investigación

La presente investigación es experimental, ya que se quiere probar cual técnica de aprendizaje automático supervisado es el más adecuado.

1.4.4 Población y Muestra Metodológica

a) Población

Registro de variables de energía de equipos fotovoltaicos extraídos del sistema Victron Energy.

b) Muestra

Un conjunto de 19063 a más registros provenientes del Sistema Victron Energy extraídos del equipo de infraestructura fotovoltaica que actualmente se encuentra operando, siendo esta data exportada mediante un archivo en formato csv.

1.4.5 Métodos, Técnicas e Instrumentos

1.4.5.1 Métodos

Observación experimental: elaborar la relación entre variables mediante el análisis y procesamiento de los datos mediante las técnicas de aprendizaje automático supervisado.

1.4.5.2 Técnicas

a) Análisis Documental: revisión de documentación del servicio de operaciones específicamente del área de ISM (Infraestructure Support Management), para así poder evaluar el proceso de gestión y análisis de las variables de energía en equipos con sistemas fotovoltaicos.

b) La observación: esta técnica se empleó para observar documentos, registros del sistema de monitoreo de energía, variables de

energía de los equipos, actividades realizadas y pericia del personal en la atención de incidencia ocurridas por fallas de energía.

1.4.5.3 Instrumento y Recolección de Datos

- a) **Discusión y análisis:** Nos permitirá analizar la técnica de aprendizaje automático supervisado adecuado para la predicción del estado de carga.
- b) **Hoja o ficha de registro de datos:** Nos permitirá guardar todos los datos extraídos de las variables seleccionadas por la persona que va a analizar el caso.
- c) **Diccionario de datos:** Nos permitirá dar a conocer el listado de las variables que forman parte del flujo de los datos, evitando así malas interpretaciones o ambigüedades.

1.4.6 Metodología de desarrollo del trabajo

La metodología realizada para este trabajo de investigación se basa en la metodología CRISP-DM (Cross-Industry Standard Process for Data Mining) siendo una metodología libre, gratuita y diseñada para proporcionar normalización en todo el ciclo de vida de un proyecto de análisis de datos.

Esta metodología consta de 6 fases no unidireccionales lo que indica que el resultado final de cada fase determina con que fase se deba seguir, siendo cada fase de gran utilidad para nuestro trabajo de investigación.

A continuación, en la siguiente tabla se muestra las 6 fases correspondientes a la metodología CRISP-DM:

Tabla 2 Fases de la metodología CRISP-DM

Fases	Objetivo
Comprensión del negocio	Entender el problema que se está presentando
Comprensión de los datos	Análisis preliminar de los datos que se dispone.
Preparación de los datos	Preprocesamientos de los datos, transformación y limpieza
Modelos	Selección de las técnicas de aprendizaje automático supervisado.
Evaluación	Evaluación de los resultados de las técnicas de aprendizaje automático supervisado elegidas.
Desarrollo	Despliegue a producción de la técnica de aprendizaje automático supervisada más adecuado según la evaluación de los resultados

Fuente: *Elaboración Propia*

CAPÍTULO II

2. PLANTEAMIENTO TEÓRICO

En este capítulo se desarrollará los conceptos teóricos en base a la Inteligencia Artificial, Machine Learning, Sistema Fotovoltaicos, entre otros temas; el cual permitirá al lector un mejor entendimiento del trabajo de investigación presentado en este documento.

2.1 INTELIGENCIA ARTIFICIAL

La inteligencia artificial ha ido evolucionando durante los últimos años captando la atención de millones de usuarios como ejemplos tenemos los asistentes virtuales Cortana, Siri, Alexa, entre otros hasta videojuegos en los cuales la máquina lograba actuar de manera realista en una partida.

La inteligencia artificial por concepto es el intento de hacer a una computadora tan o más inteligente que un humano darle la capacidad de razonar o de pensar basado en los datos que se le está transmitiendo.

La IA tiene dos objetivos principales, uno es tecnológico: usar los ordenadores para hacer cosas útiles (a veces empleando métodos muy distintos a los de la mente). El otro científico: usar conceptos de IA que ayuden a resolver cuestiones sobre los seres humanos y demás seres vivos. (Boden, 2017)

En la siguiente Figura 3 se muestra la implementación de la inteligencia artificial para diferentes sectores.



Figura 3 La implementación de la IA en diferentes sectores

Fuente: Elaboración Propia

A continuación, se nombrará algunos trabajos realizados en el rubro de energía para los sistemas fotovoltaicos aplicando inteligencia artificial:

Tabla 3 Trabajos recientes en el sector de energía para los sistemas fotovoltaicos aplicando inteligencia artificial

Trabajos realizados	Autor	Año	Enlace de Publicación
Diseño óptimo de sistema de generación fotovoltaico con almacenamiento en baterías para vivienda unifamiliar y electrolineras basado en técnicas de Machine Learning	-Rafael Amadeo García Pérez.	2020	https://idus.us.es/handle/11441/107045
Predicción de radiación solar en	-Luis Eduardo	2020	https://revis

Trabajos realizados	Autor	Año	Enlace de Publicación
sistemas fotovoltaicos mediante técnicas de aprendizaje automático	Ordoñez-Palacios. -Daniel Andrés León Vargas. -Víctor Andrés Bucheli Guerrero. -Hugo Armando Ordoñez-Eraso.		tas.uptc.edu .co/index.ph p/ingenieria /article/vie w/11751
Aplicación del aprendizaje máquina para la pronóstica de temperatura de módulos fotovoltaicos solares	-Rafel Antunes Campos, Giuliano. -Luchetta Martins, Ricardo Ruther. -Luis Ordoñez Palacios.	2020	https://anais cbens.emnu vens.com.br /cbens/articl e/view/877
Predicción de radiación solar en sistemas fotovoltaicos utilizando técnicas de aprendizaje automático	-Daniel León Vargas. -Victor Buchell Guerrero. -Hugo Ordoñez Erasó.	2020	https://dialn et.unirioja.e s/servlet/arti culo?codigo =7712427

Fuente: Elaboración Propia

La inteligencia artificial no es un campo monolítico, sino que está dividido en varias ramas, tales como: el aprendizaje automático, el procesamiento del lenguaje natural, los sistemas expertos, la visión por computadoras, el reconocimiento automático del habla, la planificación y la robótica. (Maikel Leyva Vázquez, 2018)

2.1.1 Funcionamiento de la Inteligencia Artificial

La inteligencia artificial "funciona" combinando varios enfoques para la resolución de problemas de matemáticas, estadística computacional, aprendizaje automático y análisis predictivo.

En la Figura 4 se muestra el funcionamiento de la inteligencia artificial donde se tomará un conjunto de datos como entrada y los procesará utilizando algoritmos inteligentes que aprenden y mejoran cada vez que se procesa un nuevo conjunto de datos. Terminado el proceso de entrenamiento se genera un modelo el cual podrá predecir información o revelar información específica a partir de nuevos datos.

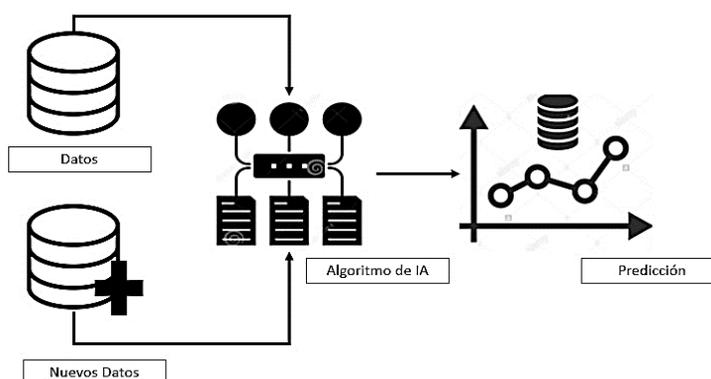


Figura 4 Funcionamiento de la IA

Fuente: Elaboración Propia

2.2 APRENDIZAJE AUTOMÁTICO

Los seres humanos necesitan comprender el mundo que los rodea y buscar patrones para predecir comportamientos específicos. El aumento en el número de interacciones entre usuarios y sistemas está generando grandes volúmenes de datos de los cuales se esperan obtener patrones que permitan predecir resultados o comportamientos con el mínimo error. Para lograrlo, se emplean diferentes técnicas computacionales y en los últimos años están destacando aquellas que se engloban bajo el nombre de Machine Learning (Aprendizaje Automático), un área de la inteligencia artificial que es cada vez más importante en todos los sectores. (Orozco, 2018)

El aprendizaje automático (Machine Learning) utilizan algoritmos que hacen que un sistema sea artificialmente inteligente, por ejemplo, los filtros de correo no deseado de los correos electrónicos utilizan este tipo de técnica de aprendizaje para que así captar aquellos mensajes que son spam y aislarlos de los que no son. Haciendo mención al ejemplo anterior las técnicas de aprendizaje automático por medio de un análisis aprenden patrones y toman decisiones en base al conocimiento adquirido.

En la Figura 5 se muestra los tres subconjuntos existentes del aprendizaje automático que pueden utilizarse.

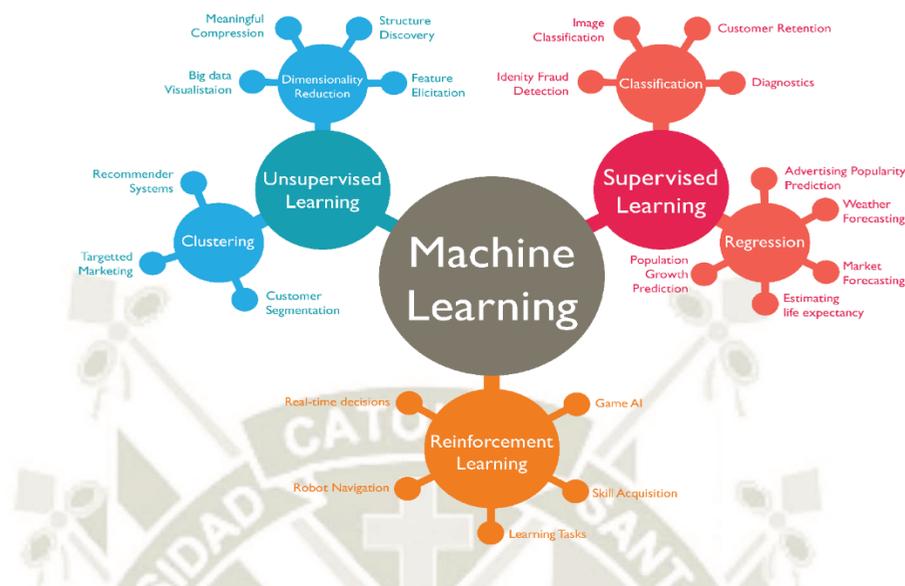


Figura 5 Esquema generalizado de los tres subconjuntos existentes de Machine Learning

Fuente: Hidalgo (2019)

2.2.1 Fundamentos del Aprendizaje Automático

2.2.1.1 El aprendizaje

El entrenamiento en un sistema de aprendizaje automático consiste en ajustar los parámetros del modelo en función de los datos recibidos. Este conjunto de datos se denomina “data set” y contiene variables independientes y dependientes. Las variables independientes (características) son columnas de conjuntos de datos que utiliza el algoritmo para construir un modelo que predice mejor la variable dependiente. Por otro lado, las variables dependientes (labels) son las columnas de la data set resultado de una correlación entre variables independientes, por lo que deben ser predichas por el modelo implementado. (Esteve, 2017)

En la Figura 6 se muestra que la programación clásica es un conjunto de reglas diseñadas por una persona con el fin de lograr respuestas que cumplan dichas reglas; sin embargo, el Aprendizaje Automático se ocupa de obtener el conjunto de reglas más efectivas que relacionan los datos de entrada con las respuestas que esperamos obtener por medio del aprendizaje.

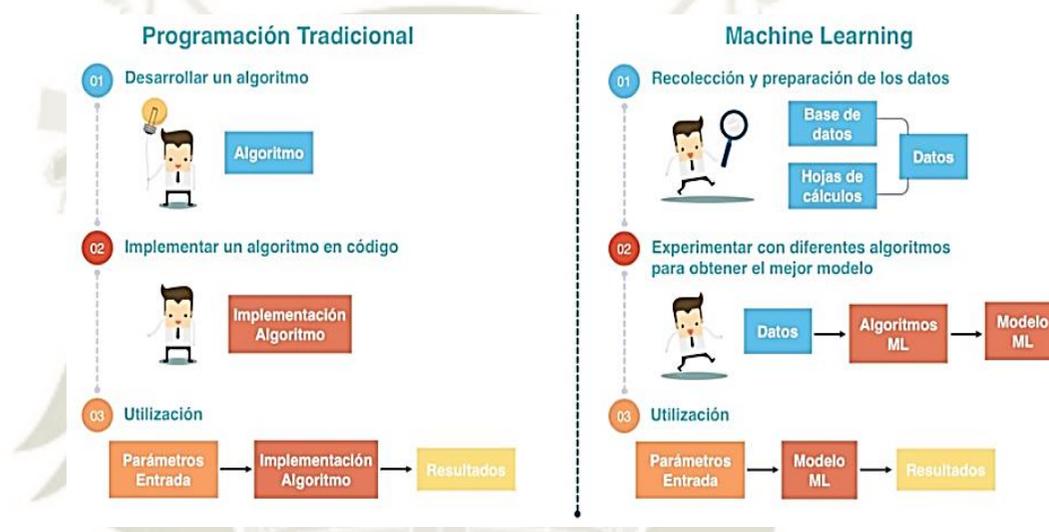


Figura 6 La programación clásica frente al Machine Learning

Fuente: González L. (2019)

2.2.1.2 El error y los problemas frecuentes en el Aprendizaje Automático

a) Los principales errores de predicción en un modelo y que están relacionados, son la varianza y el sesgo (bias).

Existe un tercer tipo de error relacionado a la cantidad de ruido en los datos de entrada, sobre el cual no se puede actuar porque es irreducible e independiente de lo bueno que sea el modelo (González L. , 2018).

El error de varianza está relacionado con el grado en el que la función objetivo varía según los datos de entrenamiento proporcionados y, en el

caso de un error de sesgo, es la diferencia entre los valores reales y la predicción que espera el modelo. No obstante, la estrecha relación entre la varianza y el sesgo hace que disminuya uno de ellos e implica el aumento del otro. Por lo cual, los algoritmos de baja varianza y alto sesgo resultan más simples y rígidos, como por ejemplo la Regresión Lineal o la Regresión Logística. Por otro lado, los algoritmos de alta varianza y bajo sesgo como los Árboles de Decisión o los Support Vector Machine son más complejos y flexibles, pero más inconsistentes. (González L. , 2018)

En la Figura 7 se muestra una gráfica explicando el nivel de complejidad de un modelo de predicción en función al error total, por lo cual si se quiere construir un modelo de predicción bueno se necesitará encontrar un buen equilibrio entre el bias y varianza de modo que minimice el error.

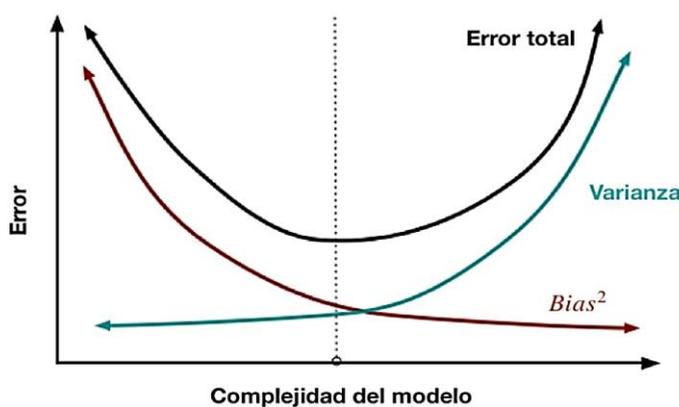


Figura 7 Nivel de complejidad del modelo en función del error

Fuente: González (2018)

Los problemas más comunes que pueden surgir en el proceso de entrenar *algoritmos*:

- **Underfitting (Subajustar)**, se presenta cuando un modelo no puede capturar la tendencia de los datos. Es generalmente el resultado de un modelo extremadamente simple. Como recomendación para resolver este problema, agregar más información representativa al modelo para mejorar la separación de las características. Esto ayudará a enfocar el aprendizaje hacia lo que desea predecir.
- **Overfitting (Sobreajustar)**, se presenta cuando un algoritmo está perfectamente adaptado a los datos con los que lo entrenamos, pero si tratamos de predecir nuevos datos, lo más probable es que nos de error. Como recomendación para resolver este problema, agregar información representativa y no representativa. Esto le permitirá al modelo mejorar la predicción al momento de enfrentar todo tipo de valores.

En la Figura 8 se visualiza diferentes gráficas representando diferentes modelos de predicción si presentasen un subajuste o un sobreajuste para cada tipo de problema ya sea de regresión o de clasificación.

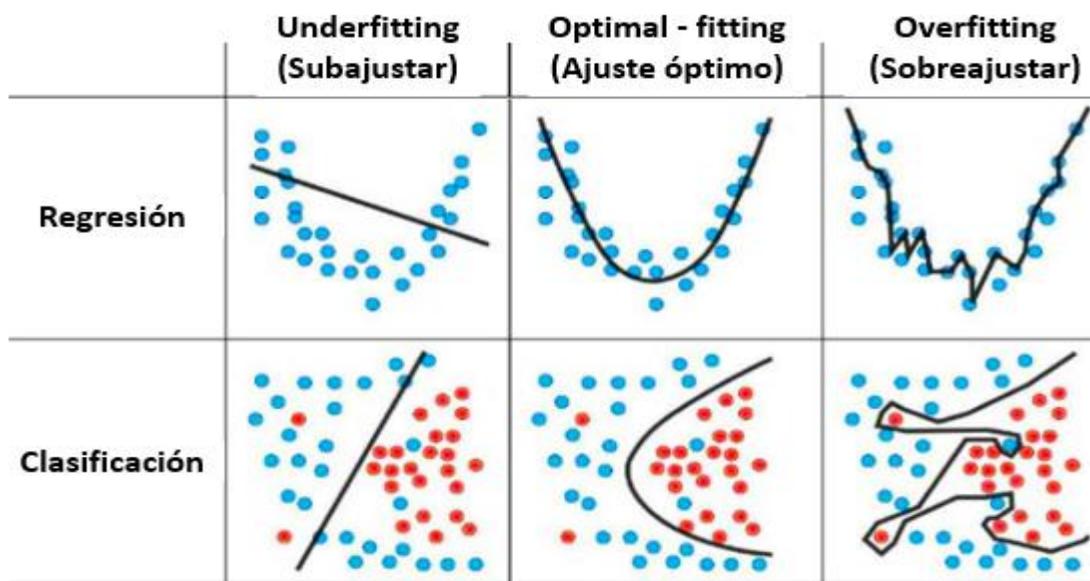


Figura 8 Distintos ajustes para los problemas de regresión y clasificación

Fuente: Minhas (2021)

2.2.2 Importancia del Aprendizaje Automático

Dado a las nuevas tecnologías informáticas, el aprendizaje automático actual no es como el aprendizaje automático del pasado. Empezó con el reconocimiento de patrones y la teoría de que las computadoras sin ser programadas pueden aprender para así poder realizar trabajos en específico. Los investigadores interesados en la inteligencia artificial querían ver si las computadoras podían aprender de los datos. El aspecto iterativo del aprendizaje automático es importante porque a medida que los modelos están expuestos a nuevos datos, y el modelo se puede ajustar individualmente. Aprenden de cálculos pasados y toman decisiones y resultados que son confiables y repetibles. No es nuevo, pero es una ciencia que ha tomado un nuevo ímpetu. (Statistical, 2020)

Si bien muchos algoritmos de aprendizaje automático han existido durante mucho tiempo, la capacidad de aplicar automáticamente cálculos matemáticos complejos a big data, una y otra vez, cada vez más rápido, es un desarrollo reciente (Statistical, 2020).

2.2.3 Aprendizaje Automático Supervisado

El Aprendizaje Supervisado un subconjunto del Aprendizaje Automático donde se generan modelos para predecir el resultado de datos de salida esperado, modelos construidos a base de algoritmos de Aprendizaje Automático y cuyas entradas hayan sido etiquetadas y clasificadas por alguien.

Los algoritmos se entrenan utilizando ejemplos etiquetados, como una entrada cuya salida deseada se conoce. Por ejemplo, una pieza de equipo podría tener puntos de datos etiquetados como "F" (fallidos) o "R" (ejecuciones). El algoritmo de aprendizaje recibe un conjunto de entradas junto con las salidas correctas correspondientes, y el algoritmo aprende comparando su salida real con las salidas correctas para encontrar errores. Luego modifica el modelo en consecuencia mediante métodos como la clasificación, la regresión, la predicción y el aumento de gradiente. El aprendizaje supervisado utiliza patrones para predecir los valores de las etiquetas en datos adicionales no etiquetados. (Statistical, 2020)

El aprendizaje supervisado se usa comúnmente en aplicaciones donde los datos históricos predicen probables eventos futuros. Por ejemplo, puede anticipar cuándo es probable que las transacciones con tarjeta de crédito sean fraudulentas o qué cliente de seguros es probable que presente una reclamación (Statistical, 2020).

En la Figura 9 se visualiza en la primera gráfica un modelo de regresión cuyo objetivo es predecir el valor de una variable continua como por ejemplo se visualiza la predicción de los ingresos de una empresa a partir de una nueva campaña de marketing; sin embargo, en la siguiente gráfica se muestra un modelo de clasificación cuyo objetivo es predecir una etiqueta como por ejemplo predecir si un paciente se encuentra enfermo.

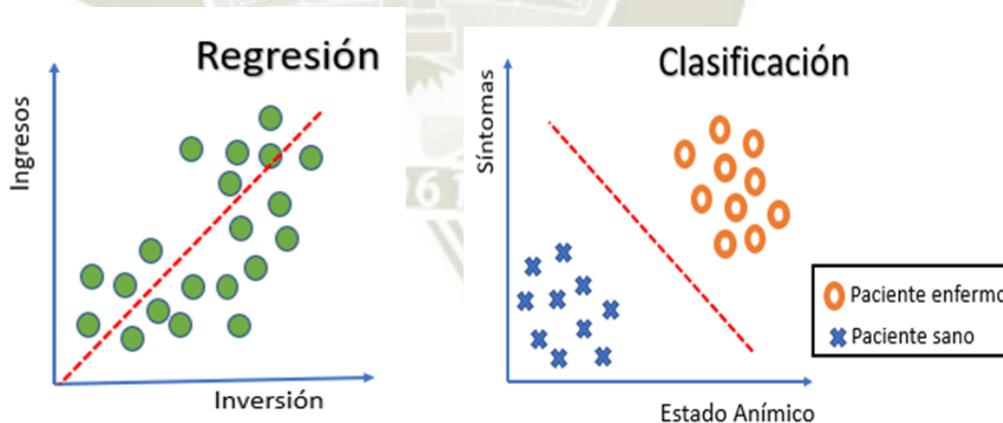


Figura 9 Modelos de Regresión y Clasificación

Fuente: Elaboración Propia

Existen diferentes técnicas de aprendizaje automático supervisado para modelos de regresión entre ellos tenemos la regresión lineal y la regresión

polinomial; y para los modelos de clasificación resalta la regresión logística, Support Vector Machine, K-Nearest Neighbors y Decision Tree.

2.2.4 Aprendizaje Automático No Supervisado

El Aprendizaje No Supervisado un subconjunto del Aprendizaje Automático en donde el algoritmo detecta patrones y agrupa la información con estos patrones.

Se usa contra datos que no tienen etiquetas históricas. La "respuesta correcta" no se notifica al sistema. El algoritmo debe averiguar qué se muestra. El objetivo es explorar los datos y encontrar alguna estructura dentro. El aprendizaje no supervisado funciona bien con datos transaccionales. Como, por ejemplo, el aprendizaje automático no supervisado puede identificar segmentos de clientes con atributos similares que luego pueden ser tratados de manera similar en campañas de marketing. O también puede encontrar los atributos principales que separan los segmentos de clientes entre sí. (Statistical, 2020)

Las técnicas comunes incluyen mapas autoorganizados, el mapeo del vecino más cercano, el agrupamiento de k-medias y la descomposición de valores singulares. Estos algoritmos también se utilizan para segmentar temas de texto, recomendar elementos e identificar datos atípicos (Statistical, 2020).

En la Figura 10 se muestra un ejemplo de agrupamiento del aprendizaje no supervisado en donde se tienen diferentes imágenes de perros y que a través de una técnica de aprendizaje automático no supervisado pretende descubrir patrones en donde logró agrupar las imágenes de perros que se encuentra sentados y perros que se encuentran parados.



Figura 10 Aprendizaje automático no supervisado

Fuente: Elaboración Propia

2.2.5 Aprendizaje Automático por Refuerzo

El Aprendizaje por Refuerzo es un subconjunto del Aprendizaje Automático en el cual se utiliza frecuentemente para la robótica, juegos y navegación, en donde el algoritmo descubre mediante pruebas y errores qué acciones producen las mayores recompensas.

Este tipo de aprendizaje tiene tres componentes principales: el agente (el aprendiz o tomador de decisiones), el entorno (todo con lo que interactúa el agente) y las acciones (lo que el agente puede hacer). El objetivo es que el

agente elija acciones que maximicen la recompensa esperada durante un período de tiempo determinado. El agente alcanzará el meta mucho más rápido si sigue una buena política. Por tanto, el objetivo del aprendizaje por refuerzo es aprender la mejor política. (Statistical, 2020)

En la Figura 11 se muestra un ejemplo para el aprendizaje automático por refuerzo en donde cada iteración constante en base a prueba y error, condiciones y teniendo un objetivo la máquina logrará vencer una partida. Considerando que las condiciones serían las reglas de un juego y el objetivo sería vencer una partida; de esta esta forma se pueden obtener resultados, patrones, correlaciones, caminos y conclusiones basadas en experiencia previa generada por la propia máquina.

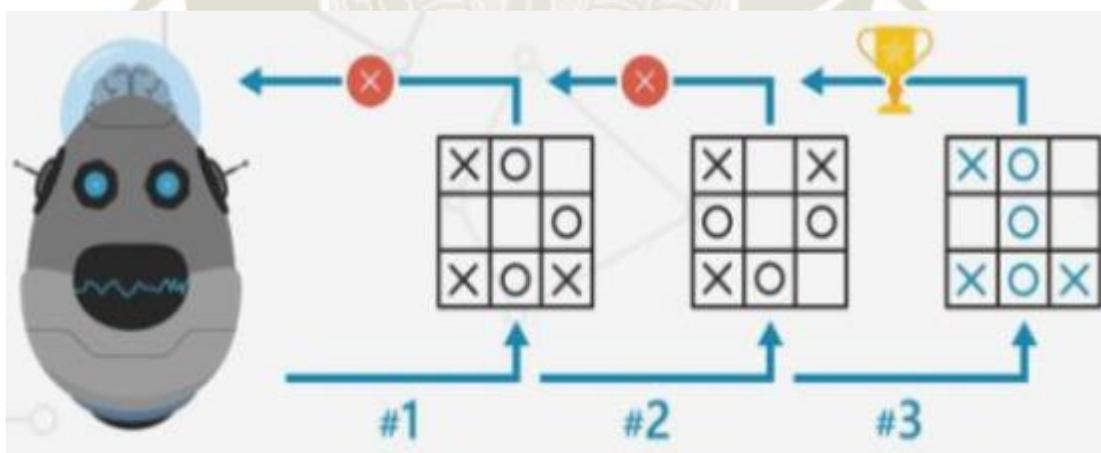


Figura 11 Aprendizaje automático por refuerzo

Fuente: Kern (2019)

2.3 TENSORFLOW

Tensorflow es una biblioteca de código abierto de extremo a extremo para Aprendizaje Automático, el cual fue desarrollado por Google.

Cuenta con un ecosistema integral y flexible de herramientas, bibliotecas y recursos de la comunidad que les permite a los investigadores impulsar un aprendizaje automático innovador y, a los desarrolladores, compilar e implementar con facilidad aplicaciones con tecnología de AA (Google Brain, 2019).

Tensorflow fue escrito en el lenguaje C++, que se ejecuta extremadamente rápido. No obstante, la API está disponible en C++ y en Python, el lenguaje más utilizado para el aprendizaje automático, y también en otros lenguajes como Java, Go, Haskell, JavaScript, Swift y más.

En la Figura 12 se visualiza una gráfica que muestra la evolución del Tensorflow y una de las características principales de esta evolución es que se han ido agregando capas de mayor nivel de abstracción teniendo como ventaja que a mayor nivel de abstracción mayor facilidad de crear programas, también se visualiza que se pueden agregar otros diferentes lenguajes de programación como Java, Go, C++, Python.

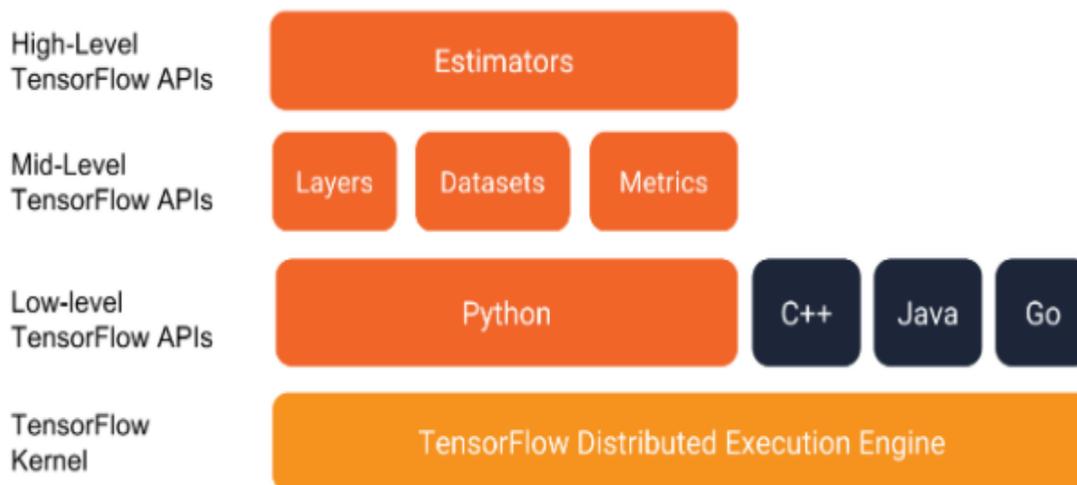


Figura 12 Niveles de APIs en Tensorflow

Fuente: Verhulsdonck (2018)

Las aplicaciones de Tensorflow son increíbles, ha marcado gran impacto en la comunidad, desde startups de Silicon Valley hasta grandes corporaciones que, como Google, Uber, Twiter, IBM, CocaCola, Airbnb entre otros están usando Tensorflow en sus proyectos.

2.3.1 Características del Tensorflow

2.3.1.1 Compilación sencilla de Modelos

Tensorflow brinda varios niveles de abstracción para que así el usuario pueda elegir según sus necesidades el que mejor se le adecue.

Compila y entrena modelos mediante la API de alto nivel de Keras, que ayuda a que comenzar a usar Tensorflow y el aprendizaje automático sea sencillo (Google Brain, 2019).

Si el usuario necesitase más flexibilidad, la ejecución inmediata permite llevar a cabo una iteración inmediata y una depuración intuitiva.

Para tareas grandes de Aprendizaje Automático, usa la API de estrategia de distribución para el entrenamiento distribuido en diferentes configuraciones de hardware sin cambiar la definición del modelo (Google Brain, 2019).

2.3.1.2 Producción de Aprendizaje Automático sólido en cualquier parte

Tensorflow ofrece un camino directo a la producción, el cual permite el entrenamiento y la implementación de tu modelo tenerlo en una estructura más sencilla, ya sea en servidores, dispositivos perimetrales o en la web, sin importar el lenguaje o la plataforma que utilices. (Google Brain, 2019)

Para realizar la inferencia en dispositivos móviles y perimetrales, usa Tensorflow Lite y si se entrena e implementa modelos en entornos de JavaScript es llevado con Tensorflow.js (Google Brain, 2019).

2.3.1.3 Importante experimentación para la investigación

Se pueden compilar y entrenar modelos innovadores sin sacrificar la velocidad ni el rendimiento. Tensorflow ofrece la flexibilidad y el control con funciones como la API funcional de Keras y la API de

subclases de modelos para la creación de topologías complejas. Para lograr un prototipado fácil y una depuración rápida, usa la ejecución inmediata. (Google Brain, 2019)

2.3.2 Tensorflow frente a otras bibliotecas

En la siguiente tabla se mostrará una comparativa frente dos bibliotecas conocidas para el aprendizaje automático frente a la biblioteca Tensorflow.

Tabla 4 Tensorflow frente a otras bibliotecas para el aprendizaje Automático

Tensorflow	PyTorch	Scikit-learn
Biblioteca de código abierto para el aprendizaje automático desarrollado por Google.	Biblioteca de código abierto para el aprendizaje automático basado en Python.	Biblioteca de código abierto para el aprendizaje construida sobre SciPy.
Brinda varios niveles de abstracción que se adecuen a las necesidades del usuario gracias a esto se puede construir con facilidad modelos simples como una regresión lineal o redes neuronales convolucionales con millones de parámetros.	Tiene un gran soporte para gráficos de cálculo dinámico (DCG), siendo una ventaja como por ejemplo para el análisis lingüístico. Pytorch para mejorar la ejecución de su entrenamiento puede usar GPU; sin embargo, en Tensorflow también puede	Es un conjunto de herramientas simples y eficaces para el análisis de datos y minera de datos. Scikit-learn no es la mejor elección para el desarrollo de aprendizaje profundo, puede desarrollar redes neuronales básicas, pero no al nivel más

Tensorflow integró el API de usar GPU, pero este usa su alto y complejo de KERAS, cuya API plantea el propia aceleración de GPU personalización que se tiene desarrollo rápido y sencillo en incorporada. con Tensorflow. redes neuronales.

Fuente: Elaboración Propia

2.4 LENGUAJE DE PROGRAMACIÓN PYTHON

Python tiene la funcionalidad de poder usarse en cualquier plataforma a la vez que hace que el código sea fácil de entender e intuitivo, también es lo suficientemente versátil como para interactuar con otros lenguajes de programación (Walker, 2018).

2.4.1 Características del Lenguaje de Programación Python

Python además es un lenguaje de código abierto, pudiendo ser utilizado por diferentes compañías, empresa o institución y contando con una fuerte biblioteca de herramientas que considere a Python un lenguaje de programación único.

Entre sus principales características del lenguaje de programación Python tenemos:

- Interpretado.
- Multiparadigma.
 - Imperativo.
 - Funcional.
 - Orientado a objetos (POO).
- De tipado dinámico.

2.4.2 Inteligencia Artificial en Python

Para la Inteligencia Artificial el lenguaje de programación Python es un buen aliado, uno por ser un lenguaje de código abierto y por tener una gran cantidad de bibliotecas útiles para su desarrollo, como por ejemplo Scikit-learn una biblioteca de aprendizaje automático que contiene algoritmos de regresión, clasificación y agrupamiento. Asimismo, tenemos a Numpy una biblioteca que brinda soporte para datos genéricos que comprenden un objetivo de matriz N-dimensional y Pandas utilizado para la manipulación y análisis de datos.

También existen bibliotecas como “Keras” y “TensorFlow” que abarcan información sobre funcionalidades del Aprendizaje Automático.

La implementación de soluciones de Inteligencia Artificial con Python seguirá siendo uno de los grandes pilares para su desarrollo y es que dentro de sus grandes ventajas se debe a sus sintaxis simple y facilidad de lectura.

2.5 RAPIDMINER

RapidMiner Studio es una herramienta fundamental para el desarrollo de procesos de análisis de datos, disponible como edición gratuita y como edición comercial. Esta herramienta también cuenta con la gran mayoría de modelos de Aprendizaje Automático, las funciones de RapidMiner Studio son una interfaz visual de arrastrar y soltar que marcan la diferencia dado que la preparación de los datos para la visualización y salida final sea tan sencilla como arrastrar bloques y conectarlos.

2.5.1 Fortalezas de la herramienta RapidMiner

2.5.1.1 Gestión del ciclo de vida de la ciencia de datos aumentada de extremo a extremo

RapidMiner facilita la gestión de la canalización de los datos de extremo a extremo, desde el inicio hasta la creación de modelos y la producción.

El proveedor ofrece específicamente un programa de certificación a través de su Academia RapidMiner para que los científicos que no son de datos comprendan el producto, el desarrollo del modelo y la operacionalización. Turbo Prep, Auto Model y Model Ops forman una visión convincente de la ciencia de datos aumentada de extremo a extremo. (Gartner, 2020)

2.5.1.2 Flexibilidad, agilidad y escalabilidad moderna

RapidMiner es compatible con una amplia gama de tecnologías como, por ejemplo: R, Python, Scala, Java, MATLAB, Octave, HiveQL, Pig, SQL y Groovy). Para el aprendizaje profundo se tiene a Keras, Tensorflow, Eclipse Deeplearning4j y Theano los cuales están integrados, mencionar que también se cuenta con todos los algoritmos de la biblioteca Weka estando disponibles como parte de la plataforma. RapidMiner también permite la creación de contenedores con Docker y Kubernetes para ejecutar y escalar modelos de forma transparente. (Gartner, 2020)

2.5.1.3 Gobernanza de modelos

RapidMiner Studio con su nueva solución Automated Model Ops, permite el cumplimiento y la auditoría de modelos tanto en las instalaciones como en la nube. Esta gestión cubre no solo el linaje de datos, sino también el entrenamiento del modelo, el uso del modelo, el reentrenamiento del modelo, el control de versiones del modelo y el acceso al modelo. (Gartner, 2020)

En la Figura 13 se muestra el Cuadrante Mágico de Gartner para plataformas de ciencia de datos y aprendizaje automático, en donde el cuadrante mágico es una culminación de la investigación en un mercado específico que la consultoría Gartner realiza sobre los principales subsectores de la industria tecnológica, siendo Gartner una de las empresas consultora de investigación y tecnología de la información más importante a nivel mundial. El Cuadrante Mágico de Gartner se encuentra dividido entre 4 cuadrantes, el eje X indica su capacidad para ejecutar y su eje Y indica la propuesta de valor. De izquierda a derecha y de arriba hacia abajo se muestra diferentes empresas ubicadas dentro de los 4 cuadrantes las cuales son: challenger, leaders, niche players y visionaires. La herramienta RapidMiner es parte del Cuadrante Mágico de Gartner encontrándose en una buena posición para el año 2020.



Figura 13 Cuadrante mágico para plataformas de ciencia de datos y aprendizaje automático
Fuente: Gartner (2020)

2.6 SISTEMAS FOTOVOLTAICOS

Un sistema Fotovoltaicos es un conjunto de equipos que utilizan la energía solar para la generación de energía y consta de los siguientes componentes:

a) Panel Fotovoltaico

Formado por celdas solares, que normalmente reciben los rayos del sol y que por medio del efecto fotoeléctrico se obtiene la conversión directa de la energía procedente del sol a energía eléctrica continua (Rodríguez, 2004).

b) Baterías

Operan como almacenadores de la energía generada, la cual es distribuida a sus cargas cuando la generación es baja o cuando no existe que en este caso sería la ausencia del sol (Rodríguez, 2004).

c) Regulador

Es un dispositivo electrónico que trabaja en conjunto con las baterías para controlar su estado de carga y asegurar un llenado óptimo de energía (Rodríguez, 2004).

d) Inversor

Este dispositivo permite la conversión de Corriente Continua (DC) en Corriente Alterna (AC), permitiendo que cargas que trabajen con AC puedan funcionar sin ningún problema (Rodríguez, 2004).

2.6.1 Tipos de Sistemas Fotovoltaicos

2.6.1.1 Sistema Fotovoltaico aislado (SFA)

Es un sistema que alimenta ciertas cargas a través de la energía solar y que no está conectado a la red eléctrica. Su objetivo principal es satisfacer total o parcialmente la demanda de energía eléctrica en zonas fuera del perímetro de la red.

Los SFA tienen una gran aplicabilidad a distintas escalas, desde pequeños sistemas fotovoltaicos que alimentan lámparas, hasta redes capaces de suministrar energía eléctrica para poblaciones completas. Habitualmente cuentan con un sistema de acumulación de energía que les permite funcionar cuando no existe recurso solar, momento en el cual suele haber mayor demanda de energía. (Abella, 2016)

2.6.1.2 Sistema Fotovoltaicos conectados a red (SFRC)

Un sistema fotovoltaico conectado a red (SFRC) se basa en una tecnología dual, el cual tiene a su disposición un generador de tipo fotovoltaico, asimismo de estar conectado a una red de energía convencional tal como un sistema de reserva.

En estos sistemas el inversor realiza la gestión entre las cargas y ambos generadores; el fotovoltaico y la red convencional. De esta forma, permite que el inversor demande energía de la red cuando el generador fotovoltaico no produzca energía suficiente, e incluso que se inyecte energía en la red eléctrica cuando la generación fotovoltaica sea superior al consumo. Por lo general no cuentan con acumulador, hecho que provoca una disminución de los gastos relacionado a la adquisición de equipos. (Mena, 2013)

2.6.1.3 Sistema Híbridos Fotovoltaicos

En algunas situaciones los sistemas fotovoltaicos carecen de energía solar y pueden combinarse con otras fuentes de energía. Estos sistemas a menudo se aplican a sistemas fotovoltaicos aislados, lo que le permite utilizar otras fuentes disponibles como por ejemplo energía eléctrica y así poder combinarla con la escasa de energía solar.

2.6.2 Baterías en Sistemas Fotovoltaicos

Las baterías son aplicadas como sistemas de almacenamiento energético, esto debido a la generación de energía que puede existir durante el día y los

consumos que se dan por las noches. Por consiguiente, las baterías posibilitan la operación cuando el generador fotovoltaico por sí mismo no puede producir energía capaz de suministrar consumo y continuar con la operación con normalidad.

En la siguiente tabla se muestra una comparativa de las diferentes baterías existentes en el mercado, comparando determinados aspectos importantes.

Tabla 5 Tipos de baterías en sistemas fotovoltaicos

Aspecto	Plomo ácido	AGM/GEL	LITIO
Importantes			
Tecnología	Plomo	Plomo	Litio
Vida útil	4-5 años	5-8 años	Más de 10 años
Profundidad de descarga máxima	No debería sobrepasar el 50- 60%	No debería sobrepasar el 50-60%	95-100%
Impacto medio ambiental	Alto	Alto	Moderado-Bajo

Fuente: Paule (2021)

En la Figura 14 se muestra el esquema en función al banco de baterías en un sistema fotovoltaico. Conformado por un panel solar que genera electricidad a partir de la luz solar, siendo esta energía almacenada en el banco de baterías y controlado constantemente por un regulador el cual regula la intensidad de carga con la finalidad de alargar la vida útil de las baterías.

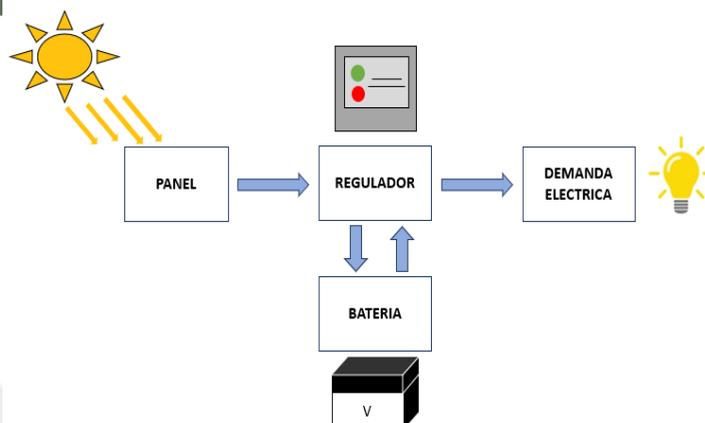


Figura 14 Esquema equivalente al banco de baterías en un sistema fotovoltaico

Fuente: Elaboración Propia

En los sistemas Fotovoltaicos se suele emplear dos tipos de baterías: Plomo ácido y Níquel-Cadmio. Siendo por motivos de coste la más habitual el uso de baterías de plomo ácido; sin embargo, las de Níquel-Cadmio en ocasiones son usadas en aplicaciones profesionales donde el coste no es un parámetro decisivo. (CIEMAT, 2005)

En comparación con las baterías de plomo-ácido, las baterías de níquel-cadmio tienen la ventaja de poder descargarse durante mucho tiempo y mantener una carga baja sin degradar la calidad. Cabe resaltar que hay menos auto descarga y menos mantenimiento

Las baterías suelen estar formadas por elementos de dos voltios que conectados en serie proporcionan tensiones de trabajo de 12 V, 24 V, 48 V, etc. La capacidad (Ah) de un grupo de baterías conectadas en es igual a la capacidad de cada uno de los elementos que lo componen. Si se conectan baterías en paralelo se suma la capacidad de sus elementos. La capacidad

necesaria de las baterías en un sistema FV se calcula en función a los consumos y al número de días de autonomía del sistema. (CIEMAT, 2005)

2.7 ESTADO DE CARGA (SoC) %

Se define como la capacidad disponible de una batería y podría ser comparada con el depósito de combustible de un vehículo. Se representa como un porcentaje de una referencia de carga completa, el cual debe ser actualizado periódicamente. Otras unidades pueden ser Culombios (C), Amperios-hora (Ah) o kilovatiohora. (kWh) (Maldonado S. Edison, 2018)

Uno de los factores más importantes que afectan a la estimación del SoC de una batería es el envejecimiento. Debido a los ciclos de carga y descarga, la capacidad de las celdas que forman la batería decrece con el tiempo. Este hecho induce a actualizar el máximo estado de carga disponible para la batería periódicamente ya que es la referencia para calcular el porcentaje antes mencionado. (Maldonado S. Edison, 2018)

2.7.1 Estimación del SoC basada en la Intensidad

Conocido como contador amperios-hora de Coulomb (Coulomb Counting), el método consiste en integrar la intensidad suministrada y absorbida de la batería, método fácil de implementar y preciso, y es una de las maneras más prácticas de estimar el SoC en baterías de plomo-ácido selladas, puesto que básicamente lo que realiza es un balance energético. (Zabaleta, Biurru Idoia San, & Pascual Miqueleiz, 2016)

SoC_0 : Estado de carga inicial

n : Rendimiento de la batería

$I(t)$: Corriente de la batería

t_0 : Instante Inicial

SoC: Estado de carga

C: Capacidad de la batería

$$SoC(t) = SoC_0 + \frac{n}{C} \cdot \int_{t_0}^t I(t) \cdot dt$$

Figura 15 Fórmula para hallar el Estado de Carga (SoC)

Fuente: Zabaleta, Biurrun Idoia San, & Pascual Miqueleiz (2016)

2.8 CRISP-DM

CRISP-DM (Cross Industry Standard Process for Data Mining), metodología basada en procesos estándares que proporciona una descripción normalizada del ciclo de vida de un proyecto de análisis de datos, de forma análoga a como se hace en la ingeniería del software con los modelos de ciclo de vida de desarrollo de software. El modelo CRISP-DM cubre las fases de un proyecto, sus tareas respectivas, y las relaciones entre estas tareas. (Sngular, 2016)

Como se mencionó en el párrafo anterior es una metodología cíclica, en la siguiente imagen se mostrará las seis fases del ciclo de vida del proyecto:

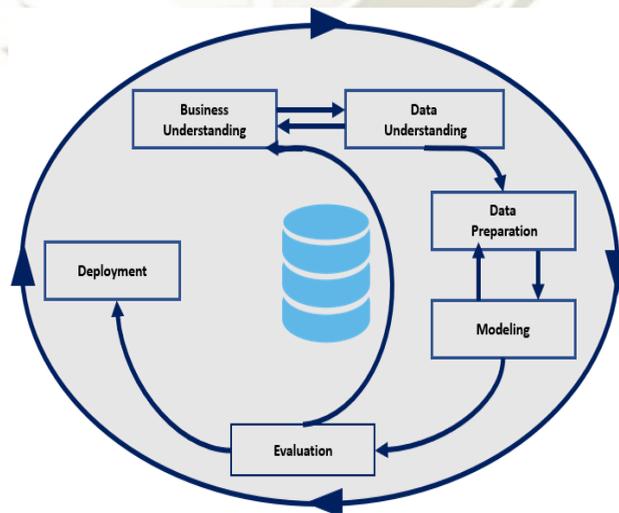


Figura 16 Fases de la metodología CRISP – DM

Fuente: Chapman (2000)

2.8.1 Comprensión del negocio

Es la fase inicial en donde se enfoca el comprender los objetivos y requisitos del proyecto desde un punto de vista comercial, posteriormente convirtiendo este conocimiento de los datos en la descripción de un problema de minería de datos para luego diseñar un plan preliminar para lograr los objetivos deseados.

2.8.2 Comprensión de los datos

En esta fase se comienza la recolección de los datos en donde se continuará con actividades para que pueda familiarizarse con los datos, identificar los problemas abarcado a la calidad de los datos, descubrir los primeros conocimientos en los datos o detectar subconjuntos interesantes para formar hipótesis en cuanto a la información oculta.

2.8.3 Preparación de los datos

En esta fase se preparan los datos para que así la fase de modelamiento los reciba sin ningún problema alguno y se puedan realizar las operaciones necesarias. Las tareas para la preparación de los datos abarcaran la selección de tablas, registros, y atributos, así como la transformación y la limpieza de datos para las herramientas de modelado y es posible que mencionadas tareas se lleven a cabo varias veces y no en ningún orden prescrito

2.8.4 Modelamiento

En esta fase del modelamiento, varias técnicas de modelado serán elegidas y aplicadas a los datos previamente preparados, ahora si se aplican más

técnicas será más viable el resultado final. En general existen diferentes técnicas que proveerán diferentes resultados, de tal modo que el resultado que se aproxime a lo deseado será elegido para así poder continuar con la evaluación.

2.8.5 Evaluación

En la fase de la evaluación, se han debido construir uno o varios modelos que han alcanzado una calidad suficiente desde una perspectiva de análisis de datos.

Antes de continuar a la fase del despliegue del modelo final, es fundamental evaluar a fondo y verificar los pasos efectuados para crearlo y comparar el modelo obtenido con los objetivos de negocio. En parte a los objetivos es clave determinar si hay alguna cuestión importante de negocio que no ha sido considerada lo suficiente. En el final de esta fase se debería lograr una decisión en el uso de los resultados del proceso de análisis de datos.

2.8.6 Despliegue

Por lo general, la creación del modelo no es la etapa final. Ahora si el objetivo del modelo es de aumentar el conocimiento de los datos, el conocimiento obtenido tendrá que reevaluar las operaciones a realizar. La fase del despliegue puede ser tan fácil como la realización de un informe de los resultados logrados o tan compleja como implementación automática de un proceso de análisis de datos en la organización.

CAPÍTULO III

3. CONSTRUCCIÓN Y EVALUACIÓN DE LAS TÉCNICAS DE APRENDIZAJE AUTOMÁTICO SUPERVISADO

En este capítulo se desarrolla la construcción y evaluación de las técnicas de aprendizaje automático supervisado.

Para el paso a paso de este trabajo de investigación se apoyará en base a las 6 fases de la metodología CRISP-DM.

En la Figura 17 se muestra un esquema de entradas y salidas para el desarrollo y la evaluación de técnicas de aprendizaje automático supervisada basado en las 6 fases de la metodología CRISP-DM.



Figura 17 Fases para la construcción y evaluación de técnicas de aprendizaje automático supervisado basado en la metodología CRISP-DM

Fuente: Elaboración Propia

3.1 FASES

3.1.1 Comprensión del negocio

Esta fase inicial se identificará el problema que ocasiona la perplejidad en la organización y se proyectará una solución para la resolución del problema presente.

En la Figura 18 se muestra un gráfico con las diferentes actividades de la Fase para la comprensión del negocio que a continuación se describirán cada una.

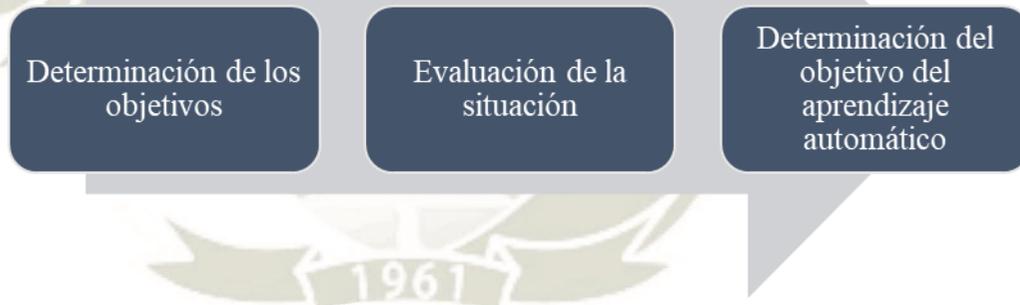


Figura 18 Actividades de la Fase para la comprensión del negocio

Fuente: Elaboración Propia

a) Determinación de los objetivos

El objetivo principal se enfoca en la solución de llevar un monitoreo correcto del estado de carga de las baterías en equipos energizados por paneles solares en sector minero.

b) Evaluación de la situación

Se cuentan con repetidores móviles que tienen como objetivo brindar cobertura a diferentes equipos de infraestructura fija y flota en el sector minero y estos se encuentran ubicados en diferentes zonas a tajo abierto en la mina.

Siendo ubicados en diferentes zonas los repetidores móviles trabajan con energía solar por medio de un sistema fotovoltaico, ahora esta energía solar es acumulada en sus baterías para que así estas suministren energía y también puedan ser usada en los momentos en los que no haya luz como por ejemplo en la noche o días nublados.

Sin embargo, se ha vuelto una problemática las incidencias ocasionadas por energía en los repetidores móviles ocasionando que los equipos tanto de infraestructura fija y de flota en algunas zonas del tajo no presentasen comunicación y era debido a que los repetidores móviles que le brindaban cobertura se encontraban desenergizados y como solución que se daba era llamar a personal de terreno encargado de los repetidores móviles para así ejecutar un actividad correctiva y conseguir que las baterías vuelvan a recargarse. No obstante, para que las baterías de los repetidores vuelvan a

recargarse de energía se demoraban unas cuantas horas al caso y mientras tanto la operación en mina paraba.

c) Determinación del objetivo del aprendizaje automático

En esta etapa se tomará la determinación de aplicar una correcta técnica de aprendizaje automático para así evitar que las baterías de los repetidores móviles se queden sin energía y puedan ser revisados previamente, teniendo como finalidad mejorar la calidad del servicio.

Teniendo en claro los objetivos y la problemática podemos pasar a la siguiente fase de la metodología.

En la Figura 19 se visualiza una imagen referencial del repetidor móvil conectado a un sistema fotovoltaico.



Figura 19 Repetidor móvil

Fuente: Protab (2018)

3.1.2 Comprensión de los datos

a) Recolección de los datos iniciales

Los datos iniciales fueron extraídos por medio del Sistema Victron Energy, el cual tiene como función monitorear las variables de energía de los paneles solares.

La toma de las variables de energía está programada a que se registren en una relación de 15 a 25 minutos en el periodo de horas desde 00:00 AM hasta las 23:55 PM de lunes a domingos, siendo almacenado cada registro en la base de datos del mismo sistema.

En la Figura 20 se muestra los datos del Sistema Victron Energía que fueron exportados en formato de una Hoja de Cálculo Excel.

A	G	H	I	J	K	L	M	N
timestamp	VE.Bus Syst	VE.Bus Syst	VE.Bus Syst	VE.Bus Syst	VE.Bus Syst	VE.Bus Syst	VE.Bus Syst	VE.Bus Syst
America/Santiago (-03:00)	Input frequen	Input power	Output volta	Output curre	Output frequ	Output powe	Voltage	Current
	Hz	W	VAC	A	Hz	W	V	A
2020-12-0100:10:47	0	0	230.2	3	50.03	606	49.95	-13.1
2020-12-0100:25:47	0	0	230.2	3	50.03	611	49.89	-13.2
2020-12-0100:40:47	0	0	229.8	3	50.03	604	49.89	-12.9
2020-12-0100:55:47	0	0	230.2	3	49.9	608	49.83	-13
2020-12-0101:10:47	0	0	229.8	3	50.03	604	49.8	-12
2020-12-0101:25:47	0	0	229.8	3	50.03	615	49.76	-13.2
2020-12-0101:40:48	0	0	229.8	3	49.9	608	49.73	-13
2020-12-0101:55:47	0	0	230.2	3	50.03	611	49.69	-13
2020-12-0102:10:47	0	0	229.8	3	50.03	610	49.63	-13.1
2020-12-0102:25:48	0	0	230.2	3	50.03	607	49.63	-13.1
2020-12-0102:40:48	0	0	230.2	3	50.03	623	49.57	-13.5
2020-12-0102:55:47	0	0	230.2	3	50.03	607	49.54	-13.1
2020-12-0103:10:47	0	0	230.2	3	50.03	608	49.51	-13.2
2020-12-0103:25:47	0	0	229.8	3.1	49.9	621	49.46	-13.5
2020-12-0103:40:47	0	0	230.2	3	50.03	606	49.44	-13.2
2020-12-0103:55:47	0	0	230.2	3	50.03	608	49.38	-13.1
2020-12-0104:10:48	0	0	230.2	3	50.03	607	49.38	-13.1
2020-12-0104:25:47	0	0	229.8	2.9	50.03	597	49.32	-12.9
2020-12-0104:40:47	0	0	230.2	3	49.9	603	49.31	-13.1
2020-12-0104:55:47	0	0	230.2	3	50.03	603	49.25	-13.3
2020-12-0105:10:47	0	0	229.8	2.9	49.9	600	49.23	-13.1
2020-12-0105:25:47	0	0	230.2	3	50.03	605	49.18	-13
2020-12-0105:40:47	0	0	230.2	3	50.03	604	49.13	-13.2
2020-12-0105:55:47	0	0	230.2	3	50.03	607	49.12	-13.2
2020-12-0106:10:47	0	0	229.8	3	50.03	602	49.06	-13.3
2020-12-0106:25:47	0	0	230.2	3	50.03	606	49.05	-13.2
2020-12-0106:33:59	0	0	229.8	3	50.03	607	49.04	-13.2
2020-12-0106:40:47	0	0	229.8	3	50.03	613	48.99	-13.2
2020-12-0106:55:47	0	0	229.8	3	50.03	609	48.99	-13.3

Figura 20 Hoja de Cálculo - VE Data Export

Fuente: Victron E. (2020)

b) Describir los Datos

Los datos extraídos del Sistema Victron Energy fueron exportados en formato Hoja de Cálculo Excel con un total de 19062 registros correspondientes a partir de la fecha del 10 de abril hasta el 31 de octubre del año 2020.

Todos los campos tienen un formato de campo del tipo Numérico y Texto, ya que al parecer los datos exportados fueron obtenidos de diferentes tablas las cuales se juntaron en una sola Hoja de Cálculo.

c) Explorar los Datos

Cuando se realizó la exploración de los datos, se apreció que dentro de los campos que fueron exportados solo tres aportarían valor alguno para el análisis con respecto al estado de carga, siendo esta observación corroborada por Ingeniero en Campo quien estaría a cargo del mantenimiento de los paneles solares de los equipos.

Los 3 campos corresponderían al voltaje, consumo y la corriente.

- Voltaje

Es la fuerza por la que se mueven los electrones debido a una diferencia de potenciales, dicha diferencia de potencial se mide en voltios (V).

- Consumo

Es la cantidad de energía que se utiliza, cuya unidad de carga eléctrica es (Ah) (Amperio-hora).

- Corriente

Flujo de electrones dentro de un conductor y se representa con la unidad de intensidad de corriente eléctrica Amperio (A).

3.1.3 Preparación de datos

En esta fase se construirá la base de datos final que alimentará a la herramienta de modelamiento, para esto se procedió a migrar la Hoja de Cálculo a un Archivo CSV Delimitado por Comas.

a) Selección de Datos

Mediante la sección de datos se permitió destacar aquellos campos que aportarían valor alguno al análisis del Estado de Carga de las Baterías los cuales fueron el Voltaje, el Consumo y la Corriente; omitiendo así los demás campos exportados del Sistema Victron Energy.

No obstante, se consideró agregar un campo más cuyo nombre sería SoC, al cual le agregaríamos una fórmula para poder hallar el Estado de Carga, ver Ecuación 1. Siendo esta fórmula aplicada por medio de los valores de los Campos Voltaje, Consumo y Corriente que se mencionaron anteriormente.

La fórmula y los resultados que se obtuvieron fueron validados por Ingeniero de Campo quien estaría a cargo del mantenimiento de los paneles solares de los equipos.

b) Limpieza de Datos

Las tareas de limpieza de datos a ejecutar son:

- Conversión de tipo para los campos Voltaje, Corriente, Consumo y SoC de tipo de dato General a Numérico.
- Modificar datos erróneos como por ejemplos algunos símbolos especiales por el alfabeto.
- Eliminación de espacios en blanco.

3.1.4 Modelos

Al realizar un análisis al detalle de la fuente de datos a utilizar, se concluye que los datos nominales son de tipo numérico y por tal motivo se tomó la decisión que el tipo de predicción más adecuado será el de regresión.

Por ello se realizará la comparación de algunas técnicas de aprendizaje automático supervisado para un modelo de tipo regresión que dentro de los cuales tenemos:

- Decision Tree
- Random Forest
- Gradient-boosted
- Neuronal Network
- Support Vector Machine

Para ejecutar la comparación de las técnicas de aprendizaje automático supervisado, vamos a llevarlo a cabo mediante el Software RapidMiner.

RapidMiner es una plataforma de ciencia de datos que unifica la preparación de datos, el aprendizaje automático y las operaciones del modelo, posicionándose dentro del “Cuadrante Mágico para plataformas de ciencia de datos y aprendizaje automático”, según la Gartner Inc en su evaluación del año 2020.

En la Figura 21 se visualiza la ventana de inicio al ejecutar la herramienta RapidMiner.



Figura 21 Ventana de Inicio de la herramienta RapidMiner

Fuente: RapidMiner (2020)

Teniendo una buena preparación de nuestros datos de entrada, estos serán importados en un formato específico en este caso el tipo será un CSV, cuyo formato permitirá que nuestros datos se representen en forma de una tabla donde las columnas se separan por comas y las filas por saltos de línea.

En la Figura 22 se visualiza la ventana de configuración cuando se está importando los datos en la herramienta RapidMiner.

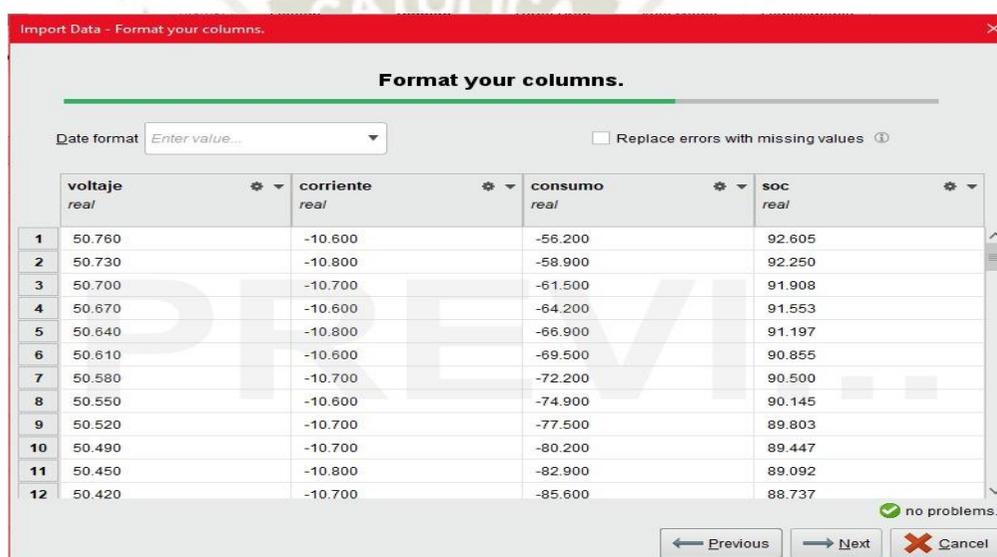


Figura 22 Importación de los Datos en la herramienta RapidMiner

Fuente: RapidMiner (2020)

En la Figura 23 se visualiza los datos correctamente importados en la herramienta RapidMiner con un total de 19062 registros y 4 atributos entre ellos se tiene el voltaje, la corriente, el consumo y el SoC.

Open in  Turbo Prep  Auto Model

Row No.	voltaje	corriente	consumo	soc
1	50.760	-10.600	-56.200	92.605
2	50.730	-10.800	-58.900	92.250
3	50.700	-10.700	-61.500	91.908
4	50.670	-10.600	-64.200	91.553
5	50.640	-10.800	-66.900	91.197
6	50.610	-10.600	-69.500	90.855
7	50.580	-10.700	-72.200	90.500
8	50.550	-10.600	-74.900	90.145
9	50.520	-10.700	-77.500	89.803
10	50.490	-10.700	-80.200	89.447
11	50.450	-10.800	-82.900	89.092
12	50.420	-10.700	-85.600	88.737
13	50.390	-10.700	-88.300	88.382
14	50.350	-10.700	-90.900	88.039

ExampleSet (19,062 examples, 0 special attributes, 4 regular attributes)

Figura 23 Datos importados correctamente en la Herramienta RapidMiner

Fuente: RapidMiner (2020)

de

regresión o clasificación en forma de estructura de árbol.

Divide un conjunto de datos en subconjuntos cada vez más pequeños en donde se desarrolla de forma incremental un árbol de decisiones asociado, la construcción del árbol sigue un enfoque de división binaria recursiva. Cuyo resultado es un árbol con nodos de decisión y nodos de hoja.

En la Figura 24 se tiene la estructura de la técnica de aprendizaje automático supervisado Decision Tree conformada con nodos de decisión y nodos de hoja.

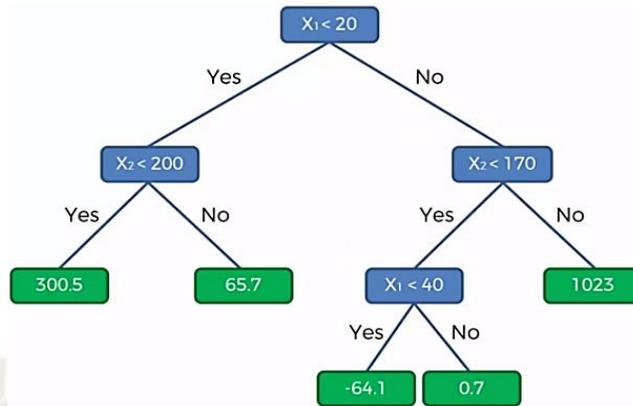


Figura 24 Decision Tree

Fuente: Girgin (2019)

En la Figura 25 se modeló la técnica de aprendizaje automático supervisado Decision Tree en la plataforma RapidMiner para los 19062 registros en el archivo CSV. importado.

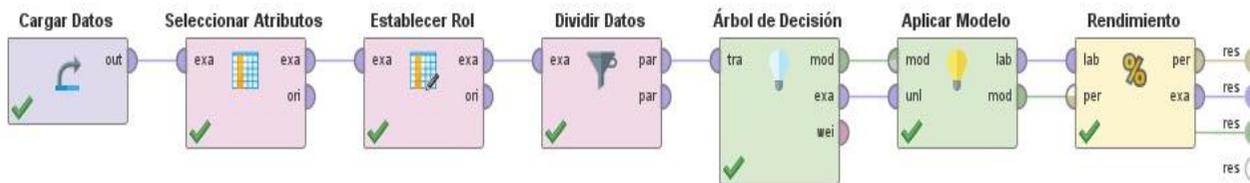


Figura 25 Técnica de aprendizaje automático supervisado Decision Tree modelado en la herramienta RapidMiner

Fuente: Elaboración Propia

3.1.4.2 Random Forest

Técnica de aprendizaje automático supervisado utilizado en modelos de regresión o clasificación y están formados por un conjunto de árboles de decisión individuales.

El bosque aleatorio o Random Forest es una técnica de embolsado y no una técnica de impulso. Los árboles en bosques aleatorios se ejecutan en

paralelo. No hay interacción entre estos árboles mientras se construyen (Chakure, 2019).

En la Figura 26 se muestra la estructura de la técnica de aprendizaje automático supervisado Random Forest conformada por un conjunto de árboles de decisión individuales que se ejecutan en paralelo. Tomado de.

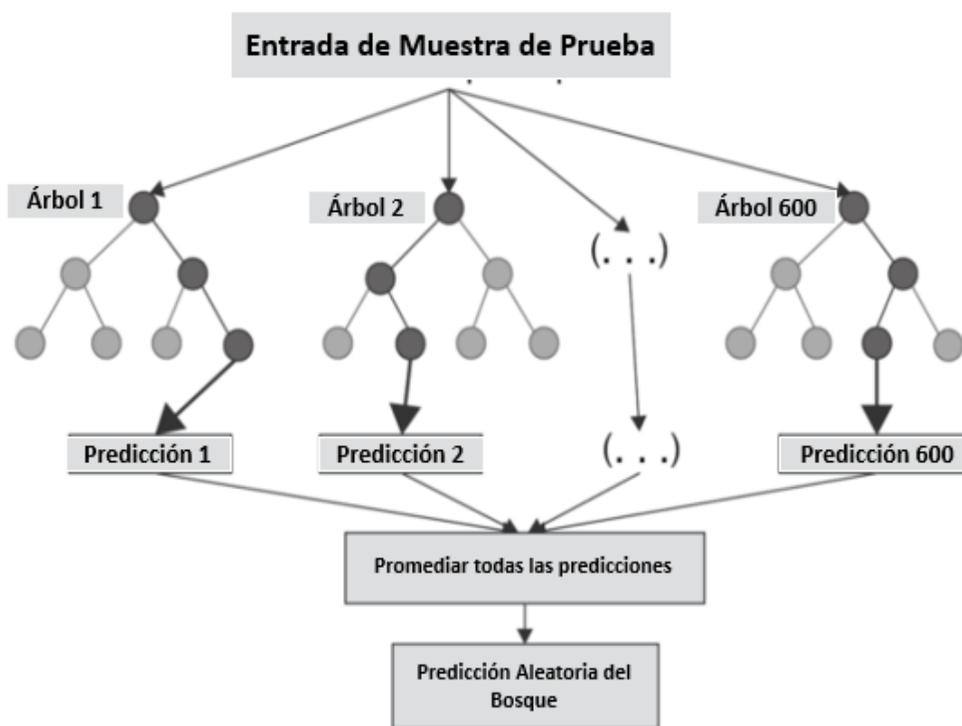


Figura 26 Random Forest

Fuente: Chakure (2019)

En la siguiente imagen se modeló la técnica de aprendizaje automático supervisado Random Forest en la plataforma RapidMiner para los 19062 registros en el archivo CSV. importado.

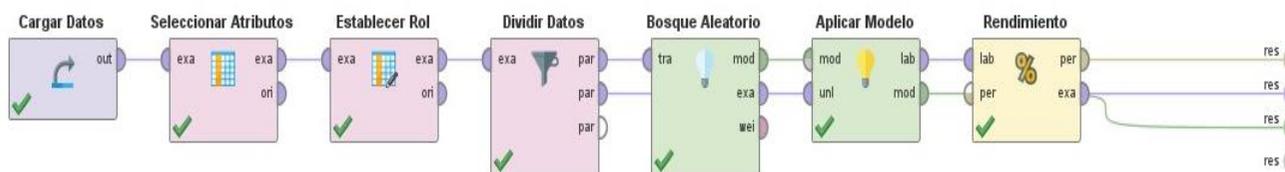


Figura 27 Técnica de aprendizaje automático supervisado Random Forest modelado en la herramienta RapidMiner

Fuente: Elaboración Propia

3.1.4.3 Gradient-boosted

Técnica de aprendizaje automático supervisado utilizado en modelos de regresión o clasificación y están formados por un conjunto de árboles de decisión en forma secuencial.

Cada árbol intenta minimizar los errores del árbol anterior. Los árboles en el impulso son aprendices débiles y cada uno de ellos se centra en los errores del anterior. Cada vez que se agrega un árbol nuevo, se ajusta a una versión modificada del conjunto de datos inicial.

En la Figura 28 se muestra la estructura de la técnica de aprendizaje automático supervisado Gradient-boosted que genera cada árbol de forma gradual donde cada árbol intenta minimizar los errores del árbol anterior.

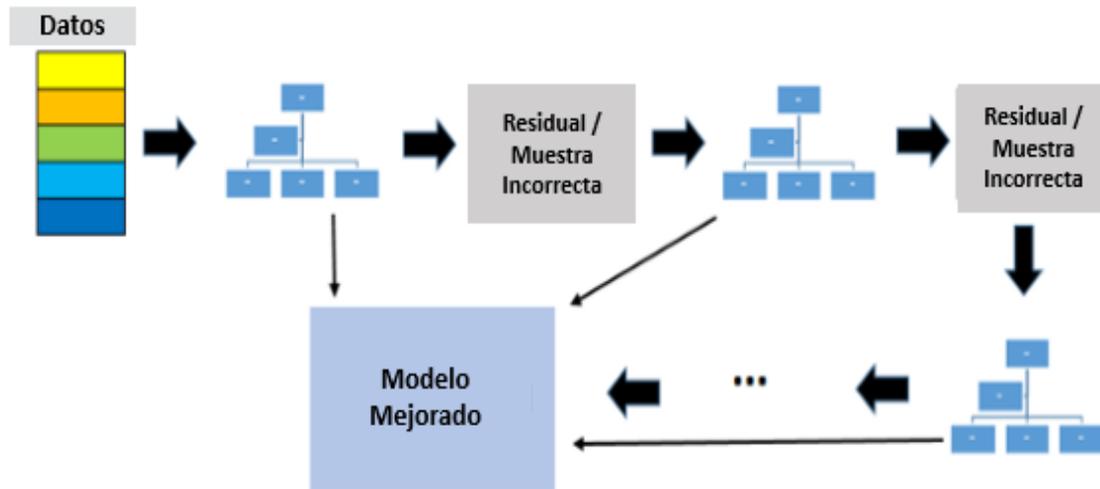


Figura 28 Gradient-boosted

Fuente: Yildirim (2020)

En la siguiente imagen se modeló la técnica de aprendizaje automático supervisado Gradient Boosted en la plataforma RapidMiner para los 19062 registros en el archivo CSV. importado.

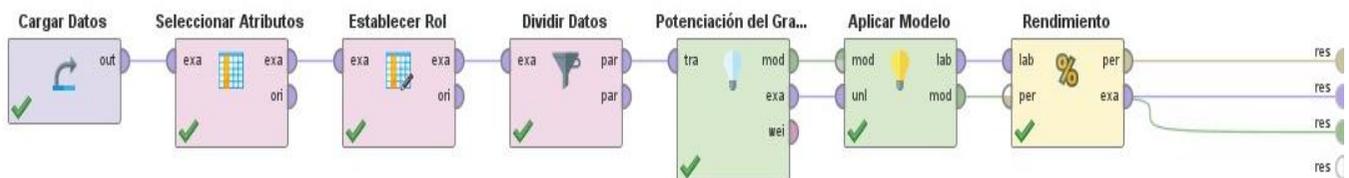


Figura 29 Técnica de aprendizaje automático supervisado Gradient Boosted modelado en la herramienta RapidMiner

Fuente: Elaboración Propia

3.1.4.4 Neuronal Network

Técnica de aprendizaje automático supervisado utilizado en modelos de regresión o clasificación, basado en la red neuronal biológica, como el cerebro.

En un sistema con RNA los nodos se conectan por medio de sinapsis, esta estructura de conexión determina el comportamiento de la red. La estructura más utilizada es el Perceptrón Multicapa. La Ilustración 33 muestra una estructura de este tipo con dos capas ocultas donde X_i representan las entradas, O_i las salidas y φ la función de activación. Las capas de entrada dependen de la información disponible para ser clasificada mientras que en las capas de salida se tiene un número de nodos igual a la cantidad de variables de respuesta al medio exterior. Las neuronas en una capa se conectan con las de la capa siguiente mediante sinapsis, cuyo valor es diferente para cada una de las conexiones y se determina a través del proceso de entrenamiento. (Villada, Muñoz, & García, 2016)

En la Figura 30 se muestra la estructura de la técnica de aprendizaje automático supervisado Neuronal Network conformada por una capa de entrada, una o varias capas ocultas y una capa de salida.

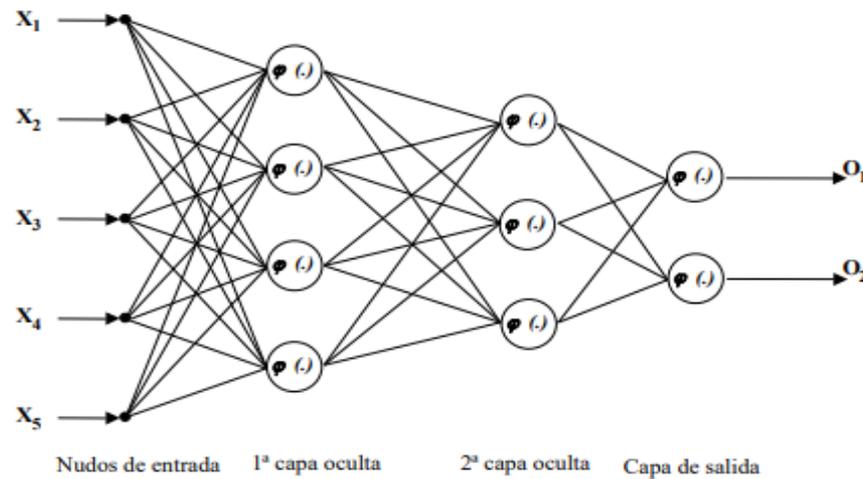


Figura 30 Neuronal Network

Fuente: Villada, Muñoz, & García (2016)

En la siguiente imagen se modeló la técnica de aprendizaje automático supervisado Neuronal Network en la plataforma RapidMiner para los 19062 registros en el archivo CSV. importado.

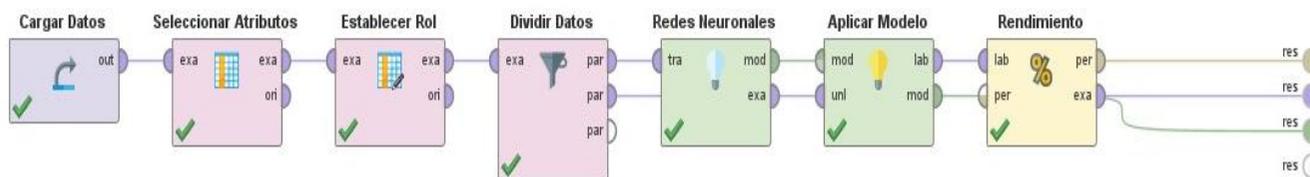


Figura 31 Técnica de aprendizaje automático supervisado Neuronal Network modelado en la herramienta RapidMiner

Fuente: Elaboración Propia

Técnica de aprendizaje automático supervisado utilizado en modelos de regresión o clasificación, cuyo objetivo del algoritmo es encontrar un hiperplano que crea un límite entre los tipos de datos.

SVM (Support Vector Machine) pertenecen a una clase de algoritmos de Machine Learning denominados métodos kernel, donde se puede utilizar una función de kernel para transformar las características. El

entrenamiento del algoritmo se asemeja a resolver un problema de optimización cuadrática para ajustar un hiperplano que minimice el margen flexible entre las clases. El número de características transformadas está determinado por el número de vectores de soporte. (MathWorksInc, 2020)

En la Figura 32 se muestra la estructura de la técnica de aprendizaje automático supervisado Support Vector Machine que hallará un hiperplano que crea un límite entre los tipos de datos.

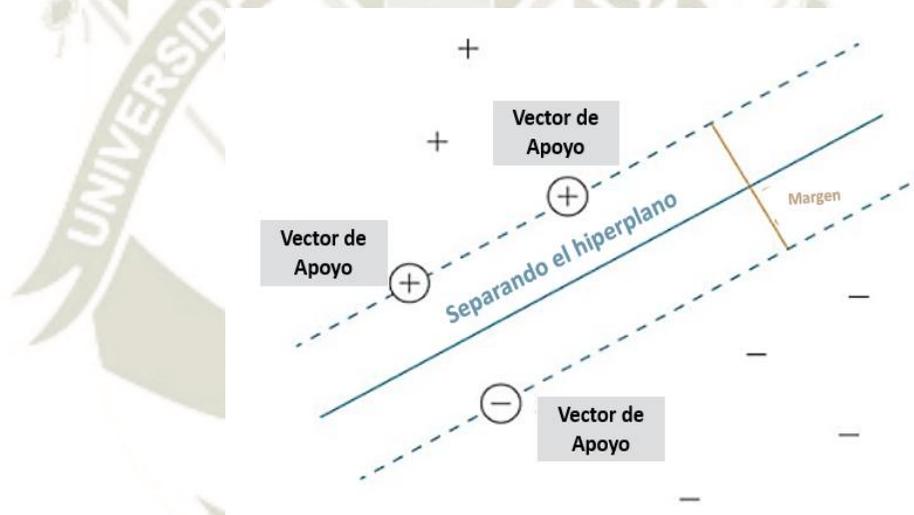


Figura 32 Support Vector Machine

Fuente: MathWorksInc (2020)

En la siguiente imagen se modeló la técnica de aprendizaje automático supervisado Support Vector Machine en la plataforma RapidMiner para los 19062 registros en el archivo CSV. importado.

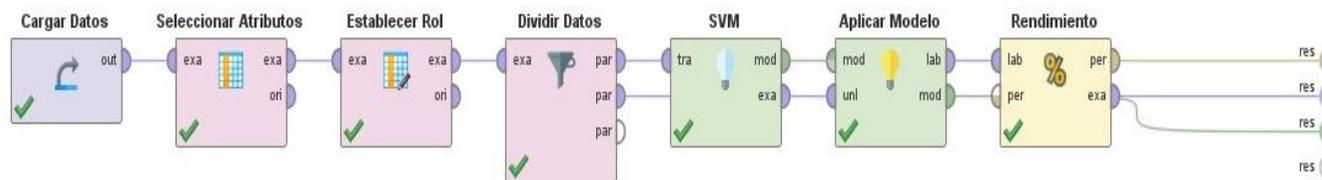


Figura 33 Técnica de aprendizaje automático supervisado Support Vector Machine modelado en la herramienta Rapidminer

Fuente: Elaboración Propia

3.1.5 Evaluación

3.1.5.1 Resultados de las técnicas de aprendizaje automático supervisado

Para la plataforma RapidMiner en el cual se modelaron diferentes técnicas de aprendizaje automático supervisado que fueron Decision Tree, Random Forest, Gradient-Boosted, Neurnal Network y Support Vector Machine, se utilizaron datos de la toma de variables de Energía del Sistema Victrom que fueron un total de 19062 registros. De la cantidad de datos para modelar se utilizó un 70% para entrenar y un 30% para realizar las pruebas.

A continuación, se mostrará un cuadro de los resultados obtenidos del modelamiento que se ejecutó en la plataforma RAPDMINER para las diferentes técnicas que fueron elegidos previamente, utilizando como datos de entrada el archivo CSV. conformada por las variables de Energía.

Tabla 6 Comparación de Resultados de las Técnicas de Aprendizaje Automático Supervisado seleccionadas

PERFORMANCE VECTOR	DECISION TREE	RANDOM FOREST	GRADIENT BOOSTED	NEURONAL NETWORK	SUPPORT VECTOR MACHINE
RMSE	0.265%	0.11%	6.70%	0.37%	3.23%
SQUARED CORRELATION	0.99%	1.00%	1.00%	0.99%	0.98%
PREDICTION AVERAGE	90.12%	90.12%	90.13%	90.12%	90.12%
EXUCUTION TIME	0.4 segundos	0.4 segundos	4 segundos	0.2 segundos	18 segundos

Fuente: Elaboración Propia

3.1.5.2 Análisis y discusión de los Resultados de las técnicas supervisadas de aprendizaje automático

En la tabla 3 se puede identificar que para las técnicas elegidas los que dieron mejores resultados con respecto al RMSE (Error cuadrático Medio) fueron: Decision Tree, Random Forest y Neuronal Network cuyos valores fueron los más bajos indicando un mejor ajuste a la predicción estimada. Para las métricas de SQUARED CORRELATION y PREDICTION AVERAGE los algoritmos dieron valores similares variando en algunos por décimas. Sin embargo, para el tiempo en ejecución se observó una gran diferencia entre los algoritmos viendo que Neuronal Network tuvo el menor tiempo con 0.2 segundos de ejecución y para Support Vector Machine fue el algoritmo que tomo más tiempo en ejecución con un total de 18 segundos.

Considerando los resultados obtenidos e identificando que la data nominal utilizada en el modelamiento es tipo Numérica se procede con la elección del algoritmo de Neuronal Network (Redes Neuronales) que es la que más se adecua al tipo de predicción que se va a ejecutar en este proyecto de investigación, esto con el sustento que las redes neuronales crean su propia interpretación de su información en su interior y son más robustas a la tolerancia a fallos y flexible cuando los datos de entrada puedan presentar cambios no tan significativos como por ejemplo el ruido, las RNA pueden manejar estos cambios adecuadamente (EcuRED, 2012) (Mahanta, 2017) que a diferencia de los algoritmos como Decision Tree y Random Forest son árboles de decisión que puedan presentar inestabilidad ante cualquier mínimo cambio en los datos de entrada puede suponer un árbol de decisión totalmente diferente, los árboles de decisión si bien son más especializados en tipo de data categórica más que en tipo de data numérica y su capacidad predictiva en un único árbol es bastante inferior a la conseguida con otros modelos, esto debido a su tendencia al overfitting y alta varianza. (Rodrigo, 2020)

Es factible que para trabajos de investigación futuros se piense utilizar otras técnicas de aprendizaje automático, esta propuesta no obliga a usar las técnicas aplicadas en el proceso anterior.

3.1.6 Desarrollo

En esta fase final permite desarrollar la técnica de aprendizaje automático supervisado elegida el cual fue Redes Neuronales (Neuronal Network), cuya técnica cumplió los objetivos que fueron planteados en un principio.

Para el desarrollo de la técnica elegida utilizaremos el lenguaje de programación Python y la librería de Tensorflow y emplearemos el desarrollo de la Red Neuronal por medio de la herramienta Google Colaboratory.

Google Colaboratory es una herramienta muy utilizada en la comunidad de aprendizaje automático, con Colab puedes importar un conjunto de datos de imágenes, entrenar un clasificador de imágenes con dicho conjunto de datos y evaluar el modelo con tan solo usar unas pocas líneas de código. Los cuadernos de Colab ejecutan código en los servidores en la nube de Google, lo que te permite aprovechar la potencia del hardware de Google, incluidas las GPU y TPU, independientemente de la potencia de tu equipo. Lo único que se necesita es un navegador. (Google, 2021)

Tabla 7 Aplicaciones de Google Colaboratory

Ejemplo de las aplicaciones que tiene Google Colaboratory

Dar los primeros pasos con Tensorflow
Desarrollar y entrenar redes neuronales
Experimentar con TPUs
Divulgar datos de investigación sobre IA
Crear tutoriales

Fuente: Google (2021)

3.1.6.1 Despliegue a producción de la Red Neuronal (Neuronal Network)

Como se mencionó en líneas anteriores se desarrollará la Red Neuronal en la herramienta Google Colaboratory utilizando la librería Tensorflow y el lenguaje de programación Python.

3.1.6.1.1 Lectura de los Datos

- Los datos serán cargados en un formato específico en este caso el tipo será un CSV.

Tabla 8 Total de atributos del conjunto de datos.

Número de Columna	Parámetro
1	Voltaje
2	Consumo
3	Corriente
4	SoC

Fuente: Elaboración Propia

- El archivo CSV será importado a través de la función Pandas `read_csv()` y almacenado en la variable “dataframe”, se importará un total de 19062 registros.
- Con el archivo importado se utilizará la función Pandas `dataframe.head()` para obtener las primeras n filas.

```
dataframe.head()
```

	voltaje	corriente	consumo	soc
0	50.76	-10.6	-56.2	92.61
1	50.73	-10.8	-58.9	92.25
2	50.70	-10.7	-61.5	91.91
3	50.67	-10.6	-64.2	91.55
4	50.64	-10.8	-66.9	91.20

Figura 34 Visualización de las primeras 4 filas con uso de la función `dataframe.head()`

Fuente: Elaboración Propia

3.1.6.1.2 Limpieza de los Datos

- Con la visualización de las primeras 4 filas de nuestra data importa continuaremos con una limpieza de la data donde se utilizará la función Pandas `dataframe.dropna()` que permitirá analizar y eliminar filas y columnas con valores nulos en diferente formas.

3.1.6.1.3 División de los Datos

- Se creará una nueva variable con el nombre “entrenardataset” en donde se almacenará la fracción aleatoria de los datos en un 80% para entrenamiento y 20 % para pruebas por medio de la función Pandas `dataframe.sample()`.

`entrenardataset=dataframe.sample(frac=0.8,random_state=0)`

- Se creará una segunda nueva variable con el nombre “testdataset” y llamaremos a la función Pandas `dataframe.drop()` donde enviaremos dentro de la función a nuestra variable “entrenardataset” para que así no se dupliquen los datos durante el entramiento.

```
testdataset=dataframe.drop(entrenardataset.index)
```

3.1.6.1.4 Inspección de los datos

- Para visualizar como es que se están relacionando los datos se importará la librería Seaborn, el cual es una librería de visualización de datos para Python y llamaremos a la función pairplot, función que crea una cuadrícula de ejes, de tal modo que se ejecutará de la siguiente forma `sns.pairplot()`.

En la Figura 35 se muestra diferentes gráficas donde se visualiza la relación del Conjunto de Datos entre cada variable que son el voltaje, el consumo, la corriente y el SoC.

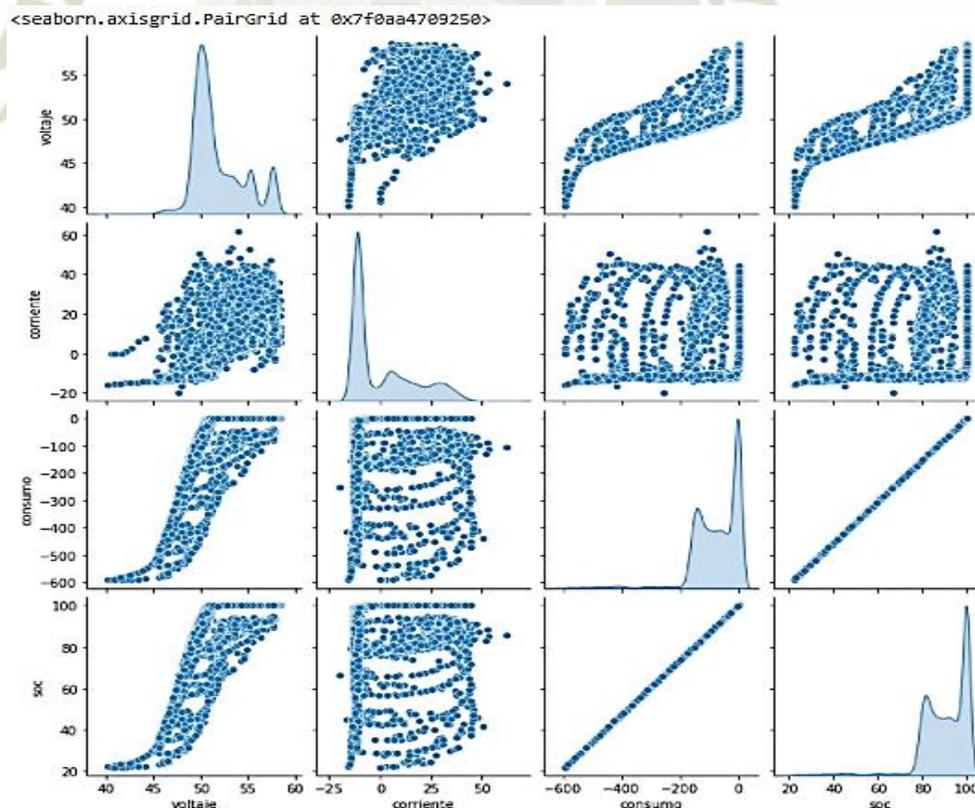


Figura 35 Inspección del Conjunto de Datos.

Fuente: Elaboración Propia

3.1.6.1.5 Separación del valor objetivo

- Se procederá a separar el valor objetivo o etiqueta, de las características.
- El valor objetivo o etiqueta separado es “SoC”, y sus características son “voltaje”, “corriente” y “consumo”.
- “SoC”, esta etiqueta es el valor que entrenará el modelo para predecir.
- Para llevar a cabo esta separación se utilizó la función `.pop()` y será almacenado en una nueva variable que tiene como nombre “train_stats”
- Una vez separado el valor objetivo o etiqueta de mi conjunto de datos se verificará sus estadísticas generales, los cuales serán visualizados en la siguiente tabla y fueron obtenidos gracias a la función `.describe()`.

En la Figura 36 para un buen análisis de los datos se debe conocer la información estadística, primero se muestra la información del valor objetivo donde se visualiza el número de muestras, el valor medio, la desviación estándar, el mínimo, el máximo, la mediana y los valores bajo los percentiles 25%, 50 y 75%.

```
train_stats = entrenardataset.describe()
train_stats.pop('soc')
```

```
count    15250.000000
mean      90.119239
std       11.117949
min       22.030000
25%      83.360000
50%      92.030000
75%      100.000000
max       100.000000
Name: soc, dtype: float64
```

Figura 36 Información estadística del valor objetivo con uso de la función describe()

Fuente: Elaboración Propia

- Con la información estadística del valor objetivo también es requerido revisar las estadísticas generales de las características (Voltaje, Corriente y Consumo).

En la Figura 37 ahora se mostrará la información de las características que son el voltaje, la corriente y el consumo donde se visualiza de igual forma el número de muestras, el valor medio, la desviación estándar, el mínimo, el máximo, la mediana y los valores bajo los percentiles 25%, 50 y 75%.

	count	mean	std	min	25%	50%	75%	max
voltaje	15250.0	51.717047	2.687710	40.17	49.84	50.72	53.28	58.47
corriente	15250.0	0.486446	15.649104	-19.90	-11.20	-10.40	10.60	61.80
consumo	15250.0	-75.096715	84.497365	-592.60	-126.50	-60.60	0.00	0.00

Figura 37 Información estadística de las características con uso de la función describe()

Fuente: Elaboración Propia

3.1.6.1.6 Normalización

- Después de haber revisado la información estadística se identificó cuan diferente son los rangos de cada característica.
- Con uso de la normalización reduciremos el rango de los datos para tener una sola longitud y así poder facilitar el entrenamiento de la red neuronal.

En la Figura 38 se muestra algunas filas de forma aleatoria cuyos datos se encuentra normalizados para cada característica (el voltaje, la corriente y el consumo).

```
normtest.head()
```

	voltaje	corriente	consumo
10	-0.471422	-0.72122	-0.092349
13	-0.508629	-0.71483	-0.187027
19	-0.579321	-0.71483	-0.378749
26	-0.661175	-0.72122	-0.602425
28	-0.683499	-0.72122	-0.666332

Figura 38 Datos Normalizados

Fuente: Elaboración Propia

3.1.6.1.7 Construcción de la red neuronal

- Para la construcción de la red neuronal se usará un modelo de tipo secuencial con una capa de entrada, una capa de salida y una capa de salida que devuelve un único valor continuo.

En la Figura 39 se visualiza el código que se utilizó para la construcción de la red neuronal, donde se implementó KERAS un API de alto nivel para compilar y entrenar modelos de aprendizaje automático.

KERAS tiene dos modos de construir redes neuronales y en este caso se utilizó el modelo secuencial (Kera.Sequential), una vez definido el tipo de modelo se determinarán cuántas capas conformaran la red neuronal y cuáles serán sus funciones de activación para cada capa, definiendo que se tendrá una capa de entrada con 4 neuronas y su función de activación “sigmoid”, una capa oculta de 4 neuronas con su función de activación “sigmoid” y una capa de salida con 1 neurona de salida con su función de activación “linear”.

Con la red neuronal definida debe completarse agregando una función de coste, un optimizador y las métricas de rendimiento, utilizando como optimizador el algoritmo RMSPROP con una tasa de aprendizaje de 0.001 y las métricas “mae, mse y accuracy” las cuales miden la magnitud promedio de los errores en un conjunto de predicciones.

```
def construir_model():  
    model=keras.Sequential([layers.Dense(4,activation='sigmoid',input_shape=[len(entrenardataset.keys())]),  
                            layers.Dense(4, activation='sigmoid'),  
                            layers.Dense(1,activation='linear')])  
    optimizador= tf.keras.optimizers.RMSprop(0.001)  
    model.compile(loss='mse',optimizer=optimizador,metrics=['mae','mse',tf.keras.metrics.CategoricalAccuracy()])  
    return model  
model = construir_model()
```

Figura 39 Construcción de la red neuronal

Fuente: Elaboración Propia

3.1.6.1.8 Entrenamiento de la red neuronal

- Se iniciará con el entrenamiento del modelo donde definiremos que la cantidad de iteraciones de aprendizaje será 1000 (epochs) de entrenamiento y se registrará la precisión del entrenamiento y la validación en la variable “History”.

En la Figura 40 se visualiza el código para el entrenamiento de la red neuronal donde se utilizó la función fit() que recibe el conjunto de datos normalizados (normentrenar), el conjunto de resultados esperados (temp_entrenar) y la cantidad de 1000 iteraciones de aprendizaje (epochs) de entrenamiento.

```
history = model.fit(normentrenar, temp_entrenar, epochs=1000,  
                    validation_split = 0.2, verbose=0, callbacks=[early_stop, PrintDot()])
```

Figura 40 Entrenamiento de la red neuronal

Fuente: Elaboración Propia

3.1.6.1.9 Predicción de la red neuronal

- Se tomará parte de los datos de prueba normalizados y procederemos a predecir el Estado de Carga (SoC).

En la Figura 41 se muestra la función predict() que permitió predecir las etiquetas de los valores de los datos sobre la base del modelo de la red neuronal entrenada.

```
predicciones=model.predict(normtest).flatten()
predicciones
array([89.11727 , 88.05361 , 85.88909 , ..., 98.235725, 95.18685 ,
       94.372154], dtype=float32)
```

Figura 41 Predicción de la red neuronal

Fuente: Elaboración Propia

3.1.6.1.10 Importación del Modelo en Tensorflow.js

- Para poder implementar el modelo junto a sus pesos en un módulo informático web se deberá instalar la biblioteca Tensorflow.JS.

En la Figura 42 se muestra la instalación de la biblioteca Y.JS

```
pip install tensorflowjs
Collecting tensorflowjs
  Downloading https://files.pythonhosted.org/packages/d4/1f/632d04bec71d4736a4e0e512cf41236b3416ac0d0a532f6511a829d18c9/tensorflowjs-3.3.0-py3-none-any.whl (63kB)
    |#####| 71kB 6.0MB/s
Requirement already satisfied: tensorflow<3,>=2.1.0 in /usr/local/lib/python3.7/dist-packages (from tensorflowjs) (2.4.1)
Requirement already satisfied: six<2,>=1.12.0 in /usr/local/lib/python3.7/dist-packages (from tensorflowjs) (1.15.0)
Requirement already satisfied: h5py<3,>=2.8.0 in /usr/local/lib/python3.7/dist-packages (from tensorflowjs) (2.10.0)
Collecting tensorflow-hub<0.10,>=0.7.0
  Downloading https://files.pythonhosted.org/packages/ac/83/a7df82744a794107641dad1decaad017d82e25f0e1f761ac9204829eef96/tensorflow-hub-0.9.0-py2.py3-none-any.whl (103kB)
    |#####| 112kB 28.4MB/s
Requirement already satisfied: grpcio==1.32.0 in /usr/local/lib/python3.7/dist-packages (from tensorflow<3,>=2.1.0->tensorflowjs) (1.32.0)
Requirement already satisfied: protobuf==3.9.2 in /usr/local/lib/python3.7/dist-packages (from tensorflow<3,>=2.1.0->tensorflowjs) (3.12.4)
Requirement already satisfied: wheel==0.35 in /usr/local/lib/python3.7/dist-packages (from tensorflow<3,>=2.1.0->tensorflowjs) (0.36.2)
Requirement already satisfied: gast==0.3.3 in /usr/local/lib/python3.7/dist-packages (from tensorflow<3,>=2.1.0->tensorflowjs) (0.3.3)
Requirement already satisfied: tensorboard==2.4 in /usr/local/lib/python3.7/dist-packages (from tensorflow<3,>=2.1.0->tensorflowjs) (2.4.1)
Requirement already satisfied: absl-py==0.10 in /usr/local/lib/python3.7/dist-packages (from tensorflow<3,>=2.1.0->tensorflowjs) (0.12.0)
Requirement already satisfied: opt-einsum==3.3.0 in /usr/local/lib/python3.7/dist-packages (from tensorflow<3,>=2.1.0->tensorflowjs) (3.3.0)
Requirement already satisfied: tensorflow-estimator<2.5.0,>=2.4.0 in /usr/local/lib/python3.7/dist-packages (from tensorflow<3,>=2.1.0->tensorflowjs) (2.4.0)
Requirement already satisfied: google-pasta==0.2 in /usr/local/lib/python3.7/dist-packages (from tensorflow<3,>=2.1.0->tensorflowjs) (0.2.0)
Requirement already satisfied: numpy==1.19.2 in /usr/local/lib/python3.7/dist-packages (from tensorflow<3,>=2.1.0->tensorflowjs) (1.19.5)
Requirement already satisfied: typing-extensions==3.7.4 in /usr/local/lib/python3.7/dist-packages (from tensorflow<3,>=2.1.0->tensorflowjs) (3.7.4.3)
Requirement already satisfied: keras-preprocessing==1.1.2 in /usr/local/lib/python3.7/dist-packages (from tensorflow<3,>=2.1.0->tensorflowjs) (1.1.2)
```

Figura 42 Biblioteca Tensorflow JS

Fuente: Elaboración Propia

- Una vez que se encuentre instalado la biblioteca, se procederá a importarla.
- Cuando la biblioteca se encuentra importada, el Modelo junto a sus pesos se convertirán en base a la biblioteca Tensorflow.JS y será guardado en una nueva ubicación en un archivo con la extensión JS.

En la Figura 43 se muestra el código para la conversión del modelo de la red neuronal a Tensorflow.JS.

```
import tensorflowjs as tfjs
tfjs.converters.save_keras_model(model, '/content/drive/My Drive/Go tesis/PREDICCION SOC/SOC/ModelJsultimate/probar_js')
```

Figura 43 Conversión del modelo

Fuente: Elaboración Propia

En la Figura 44 se muestran las carpetas generadas luego de la conversión a Tensorflow.JS. siendo almacenadas en una nueva ubicación.

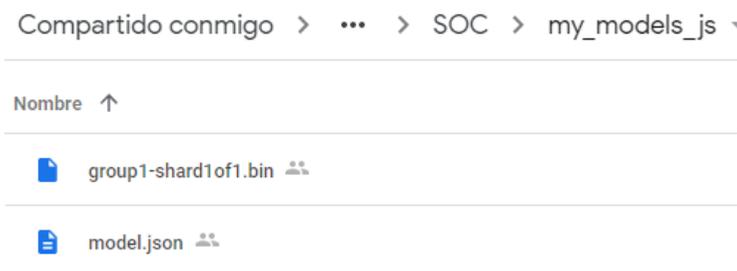


Figura 44 Modelo Convertido

Fuente: Elaboración Propia

- Con el Modelo Guardado se continuará en importarlo en el código JS del módulo informático web que se implementará para este trabajo de investigación.
- Dentro del código del módulo informático web a implementar se cargará el Modelo junto a sus pesos.

En la Figura 45 se visualiza la línea de código que permitirá cargar el modelo de la red neuronal en JS.

```
tf.loadLayersModel('https://localhost:44341/MTensor/model.json', { weightPathPrefix: 'https://localhost:44341/MTensor/' }).
```

Figura 45 Modelo cargado en JS

Fuente: Elaboración Propia

3.1.6.1.11 Módulo Informático Web con Importación del Modelo en Tensorflow.js

- A continuación, se mostrará el módulo Informático Web para predecir el Estado de Carga de las Baterías (SoC) que se propuso en parte a los objetivos específicos en este trabajo de investigación. Cabe mencionar solo se integró un nuevo módulo informático web esto debido a que la empresa en donde se está realizando la investigación ya contaba con su propia Plataforma Web y solo se optó por integrar un módulo más.

En la Figura 46 se visualizan los módulos informáticos de la plataforma web destacando el módulo nuevo que se desarrolló para el estado de carga de las baterías en sistemas fotovoltaicos.

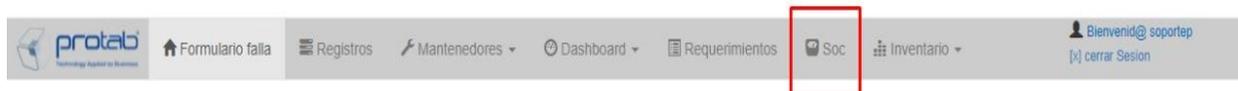
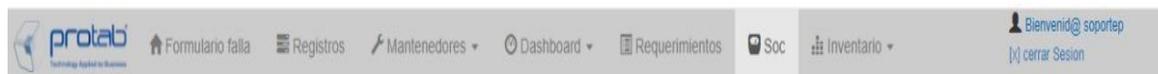


Figura 46 Módulos informáticos de la plataforma web

Fuente: Elaboración Propia

- El Módulo Informático Web (SoC) contará con 3 Textbox para ingresar los datos como “Voltaje”, “Corriente” y “Consumo” fundamentales para la predicción del objetivo principal de este trabajo de investigación, con los 3 datos ingresados también se observa un botón con el nombre “Ejecutar”, mencionado botón se encargará del análisis de la predicción del Estado de Carga.



Estado de Carga

Voltaje (V) Corriente (A) Consumo (AH)

El grafico de la prediccion se cargara aqui!

Figura 47 Módulo informático web para el estado de carga (SoC)

Fuente: Elaboración Propia

CAPÍTULO IV

4. RESULTADOS

En este capítulo se presentarán los resultados obtenidos de la técnica de aprendizaje automático supervisado elegido.

Además de estos resultados, se mostrará las pruebas desarrolladas del módulo informático web con la importación del modelo de la técnica de aprendizaje seleccionado en Tensorflow.JS.

4.1 Resultados del Entrenamiento del Modelo de la Red Neuronal

Se utilizó la librería Tensorflow para encontrar el mejor resultado de los parámetros.

Como se mencionó en líneas anteriores la cantidad de iteraciones de aprendizaje que se utilizó en el entrenamiento del modelo fueron 1000 (epochs), en la siguiente imagen se mostrarán los resultados obtenidos:

En la Figura 49 se observa que a partir de la iteración 282 logra el 100% de aciertos tanto de la data en entrenamiento como para la data de validación, No obstante, se observa que los valores de las Métricas “MAE” y “MSE” por cada iteración del entrenamiento del modelo de la red neuronal el error va disminuyendo dando así una buena respuesta para la relación entre los datos de entrada y los resultados.

	loss	mae	mse	categorical_accuracy	val_loss	val_mae	val_mse	val_categorical_accuracy
282	0.004266	0.029661	0.004266	1.0	0.004209	0.035097	0.004209	1.0
283	0.004181	0.029329	0.004181	1.0	0.009370	0.073105	0.009370	1.0
284	0.004424	0.029528	0.004424	1.0	0.005868	0.025485	0.005868	1.0
285	0.004304	0.029633	0.004304	1.0	0.004069	0.030677	0.004069	1.0
286	0.004034	0.029239	0.004034	1.0	0.006402	0.032521	0.006402	1.0

Figura 48 Resultados del entrenamiento del modelo

Fuente: Elaboración Propia

En la Figura 49 se tiene una gráfica con la tasa del nivel de error para el conjunto de datos de entrenamiento (Train Error) y el nivel de error para el conjunto de datos de validación (Val Error), siendo esto ejecutado en base a la métrica MAE por cada iteración que realizó el modelo de la red neuronal.

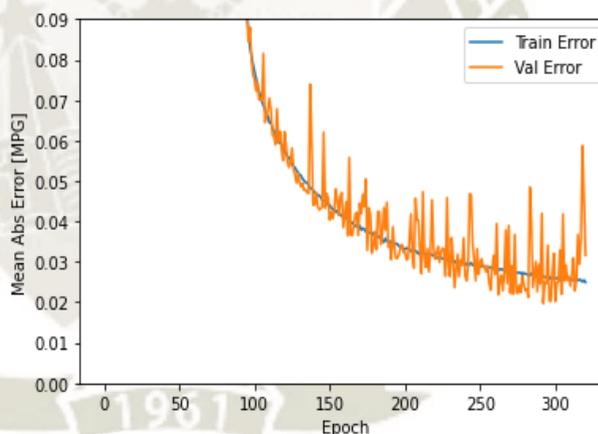


Figura 49 Curvas de error

Fuente: Elaboración Propia

Con buenos resultados en el entrenamiento del modelo de la Red Neuronal, a continuación se procederá con la evaluación del modelo pero con el conjunto de datos de prueba.

En la Figura 50 se muestra los resultados obtenidos en la evaluación del modelo con el conjunto de datos de prueba que para ello se tiene el valor de

pérdida y todas las métricas que se definieron cuando se ejecutó el modelo, cuyos valores son bajos lo que quiere decir que el modelo tuvo una buena relación entre los datos de entrada y los resultados.

```
loss, mae, mse, accuracy = model.evaluate(normtest, temp_test, verbose=2)

print("Testing set Mean Abs Error: {:.5.2f} MPG".format(mae))

120/120 - 0s - loss: 0.0032 - mae: 0.0316 - mse: 0.0032 - categorical_accuracy: 1.0000
Testing set Mean Abs Error: 0.03 MPG
```

Figura 50 Evaluación del modelo para el conjunto de datos de prueba

Fuente: Elaboración Propia

En la Figura 51 se mostrarán los resultados predcidos con uso del conjunto de datos prueba.

```
predicciones

array([89.11727 , 88.05361 , 85.88909 , ..., 98.235725, 95.18685 ,
       94.372154], dtype=float32)

temp_test

10      89.09
13      88.04
19      85.91
26      83.42
28      82.71
...
19037   100.00
19044   99.80
19049   98.22
19057   95.22
19059   94.41
Name: soc, Length: 3812, dtype: float64
```

Figura 51 Resultado de las predicciones

Fuente: Elaboración Propia

Por último, en la siguiente gráfica Figura 52 se muestra la relación entre los datos reales del valor objetivo “SoC” y los datos predictivos del valor objetivo “SoC”; en donde se puede apreciar que el modelo predice razonablemente bien.

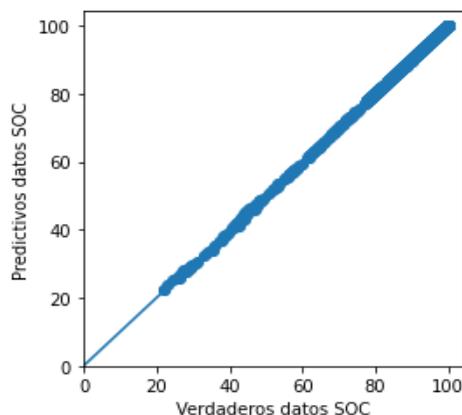


Figura 52 Datos reales vs datos predictivos

Fuente: Elaboración Propia

4.2 Análisis y discusión de los resultados del entrenamiento del modelo

En el entrenamiento del modelo de la red neuronal las métricas que monitoreó el aprendizaje del modelo fueron el “MAE” (Error Absoluto Medio) y el “MSE” (“Error al Cuadrado Medio”) en donde se pudo observar que el error de entrenamiento por cada “epoch” fue cada vez más bajo, de esta manera se obtuvo una buena respuesta para la relación entre los datos de entrada y los resultados.

Sin embargo, en la gráfica mostrada anteriormente ver Figura 48 se eligió la métrica de regresión “MAE” por ser la más robusta y menos sensible a los valores atípicos como el error cuadrático medio.

Es importante que se evalúe el modelo de la red neuronal y si no se evalúan los datos para el entrenamiento adecuadamente es posible que los resultados con llevarán a conclusiones confusas o incorrectas. Por consiguiente, cuando se mostró la gráfica anterior ver Figura 48 se consideró la evaluación del modelo en cuatro categorías las cuales mostraremos en la siguiente tabla:

Tabla 9 Categorías de la Evaluación del Modelo

Underfitting	Error de validación y error de entrenamiento alto
Sobreajuste	Error de validación es alto, el error de entrenamiento es bajo.
Buen Ajuste	Error de validación bajo, levemente superior al error de entrenamiento.
Ajuste desconocido	Error de validación bajo, error de entrenamiento alto.

Fuente: Koehrsen (2018)

No obstante, en la gráfica ver Figura 50 se mostró que el error de entrenamiento y validación se acercan entre si a medida que continúe el aprendizaje; pero si se observará que el entrenamiento comienza a ser más bajo que el error de validación se comenzaría a sobre ajustar el modelo.

De manera que para evitar ese sobreajuste en el modelo se utilizó la función “EarlyStopping” que detendrá el entrenamiento cuando el puntaje de validación no mejore, quiere decir que si se transcurre una determinada cantidad de “Epochs” sin mostrar mejoría entonces el entrenamiento se detiene automáticamente. Como se observó en el entrenamiento del modelo se declaró en un principio 1000 “Epochs” pero en los resultados solo se muestra hasta 300

“Epochs” dando a concluir que a partir de los 300 “Epochs” no se va a presentar más mejoras o degradaciones con respecto a la métrica “MAE”.

Por último, se hizo una última validación de los datos predictivos y los datos reales del valor objetivo “SoC” donde se muestra una buena relación entre ambos datos, así como la demostración de una parte de los datos que se predicieron y la comparación con los datos reales la diferencia fue mínima.

Los atributos que se utilizaron para esta investigación fueron muy relevantes para llegar a buenos resultados de exactitud porque se obtuvo una alta precisión tanto para los datos de entrenamiento como también para los datos reales de prueba.

4.3 Pruebas del módulo informático web con importación del modelo en Tensorflow.JS

Para las pruebas realizadas en el módulo informático web se utilizaron datos reales extraídos del Sistema Victron Energy y los resultados fueron validados por el jefe del área de ISM, a continuación, se presentarán las siguientes pruebas realizadas:

1. Se hizo una primera prueba con los siguientes datos reales de entrada de un equipo en ejecución:

Voltaje: 53.03

Corriente: 31.3

Consumo: 0

El resultado de la Predicción fue que nuestro equipo está cargado al 100 %.



Figura 53 Primera prueba con datos reales para la predicción del estado de carga (SoC)

Fuente: Elaboración Propia

2. Se hizo una segunda prueba con los siguientes datos reales de entrada de un equipo en ejecución:

Voltaje: 49.53

Corriente: -12.3

Consumo: -126.3

El resultado de la Predicción fue que nuestro equipo está cargado al 84 %.



Figura 54 Segunda prueba con datos reales para la predicción del estado de carga (SoC)

Fuente: Elaboración Propia

CONCLUSIONES

1. Se estableció que la técnica supervisada de aprendizaje automático más adecuada sería Redes Neuronales con un promedio de predicción al 90,20% en un tiempo de ejecución de 0.02 segundos, además fue la que más se adecuó al tipo de predicción que se ejecutó en este proyecto de investigación, dado que las redes neuronales crean su propia interpretación de su información en su interior y son más robustas a la tolerancia a fallos y flexible cuando los datos de entrada puedan presentar cambios no tan significativos.
2. Se obtuvo la recolección de todos los datos fundamentales por medio del Sistema Victron Energy, el cual tiene como función monitorear las variables de energía de los paneles solares, se realizó una preparación de los datos donde se apreció que dentro de los campos que fueron exportados solo tres aportarían valor alguno para el análisis con respecto al estado de carga, siendo esta observación corroborada por ingeniero en campo quien estaría a cargo del mantenimiento de los paneles solares de los equipos. No obstante, se consideró agregar un campo más cuyo nombre sería SoC, al cual le agregaríamos una fórmula para poder hallar el Estado de Carga.
3. Se realizó una limpieza de los datos y con ello, se pudo consolidar toda esta información en un archivo CSV que alimentará a la herramienta de modelamiento. Al realizar un análisis al detalle de la fuente de datos a utilizar, se concluye que los datos nominales son de tipo numérico y por tal motivo se tomó por la decisión que

el tipo de predicción más adecuado será el de regresión. Por ello se efectuó la comparación de algunas técnicas de aprendizaje automático supervisado para un modelo de tipo regresión, y estas son Decision Tree, Random Forest, Gradient-boosted, Neuronal Network y Support Vector Machine.

4. Se comprobó la efectividad del entrenamiento del modelo de la red neuronal para la predicción del Estado de Carga de las Baterías, observando que a partir de la iteración 282 se logra el 100% de aciertos tanto para la data entrenada y la data de validación; no obstante también se evaluó las Métricas elegidas en el Entrenamiento del Modelo que en este caso fue el (“MAE”) por ser la más robusta y menos sensible a los valores atípicos como el error cuadrático medio.
5. Se mostró que el error de entrenamiento y validación se acercan entre sí a medida que continúe el aprendizaje; para evitar un sobreajuste en el modelo se utilizó la función “EarlyStopping” que detendrá el entrenamiento cuando el puntaje de validación no mejore, quiere decir que si se transcurre una determinada cantidad de “Epochs” sin mostrar mejoría entonces el entrenamiento se detiene automáticamente.
6. Gracias a la técnica de aprendizaje supervisado automático elegido se logró predecir el estado de carga de las baterías haciendo una comparación entre los datos predictivos y los datos en tiempo real obteniendo una buena relación y resultados satisfactorios.

7. Además, se logró importar el modelo junto a sus pesos de la red neuronal al módulo informático web propuesto en parte a los objetivos específicos del trabajo de investigación, conformado de un formulario en donde el usuario pueda colocar los datos de las variables de energía monitoreadas y predecir el estado de carga que bota sus baterías.



RECOMENDACIONES Y TRABAJOS FUTUROS

1. Ampliar los atributos determinantes como por ejemplo las temperaturas de las baterías, entre otros para así no solo poder hallar el estado de carga de las baterías si no también poder hallar el ciclo de vida de las baterías.
2. Realizar la aplicación de otros algoritmos de regresión para comparar los resultados y comprobar la eficiencia.
3. Para identificar si el modelo este presentando problemas como el underfitting o el overfitting se recomienda dividir el conjunto de datos de entrada en dos subconjuntos, siendo la repartición un 70 % para el entrenamiento y un 30 % para evaluación. Asimismo, esta proporción puede variar según sea el caso, como en el modelamiento de la red neuronal presentado en este trabajo de investigación donde la repartición fue un 80 % para el entrenamiento y 20 % para la evaluación. Lo importante es que hay que estar consciente y evitar que en el modelo obtenido se presenten estos dos problemas.
4. Considerar tener una cuenta de Google con la agregación de la aplicación Google Colaboratory para poder desarrollar el entrenamiento del Modelo de la Red Neuronal.
5. Automatizar el proceso de la extracción de la data para evitar errores de ingreso de datos a la predicción.
6. Utilizar reportes dinámicos (Power BI, entre otros) para tener una mejor visualización de la data que se está prediciendo y a futuro tomar mejores decisiones de esta misma.

REFERENCIAS BIBLIOGRÁFICAS

- Abella, M. (2016). *Dimensionado de Sistemas fotovoltaicos*. Madrid: CIEMAT. Centro de Investigaciones Energética.
- Alpaydin, E. (2020). *Introduction to Machine Learning*. The MIT Press.
- Bitar S., S. M., & Chamas B., F. (2017). *Estudio de factibilidad para la implementación de sistemas fotovoltaicos como fuente de energía en el sector industrial de Colombia*. Obtenido de <https://repository.cesa.edu.co/handle/10726/1572>
- Boden, M. A. (2017). *Inteligencia Artificial*. TURNER NOEMA.
- Chakure, A. (29 de Junio de 2019). *Random Forest Regression Along with its implementation in Python*. Obtenido de <https://medium.com/swlh/random-forest-and-its-implementation-71824ced454f>
- Chapman, P. C. (2000). *Crisp-dm 1.0 step-by-step data mining guide*.
- CIEMAT. (2005). *Sistemas Fotovoltaicos Miguel Alonso Abella CIEMAT Centro de Investigaciones Energéticas, Medioambientales y Tecnológicas Departamento de Energías Renovables Laboratorio de Energía Solar Fotovoltaica*. Madrid.
- DeepAi. (2017). *Artificial Intelligence*. Obtenido de <https://deepai.org/machine-learningglossary-and-terms/artificial-intelligence>
- EcuRED. (Agosto de 2012). <https://dspace.ups.edu.ec/bitstream/123456789/4098/1/UPS-CT002584.pdf>. Obtenido de https://www.ecured.cu/index.php/Redes_neuronales_artificiales
- Energy, V. (2020). Victron Energy. Chile.
- Esteve, J. (2017). *Introducción al Machine Learning*. Obtenido de <https://es.slideshare.net/JavierEsteveMeli/introduccion-al-machine-learning-77766343>
- Gartner. (2020). *Cuadrante mágico para plataformas de ciencia de datos y aprendizaje automático*. Obtenido de <https://www.gartner.com/en>
- Girgin, S. (22 de Mayo de 2019). *Decision Tree Regression in 6 Steps with Python*. Obtenido de <https://medium.com/pursuitnotes/decision-tree-regression-in-6-steps-with-python-1a1c5aa2ee16>
- González, L. (Noviembre de 2018). *Aprende todo sobre inteligencia artificial*. Obtenido de <http://ligdigonzalez.com/bias-y-varianza-enmachine-learning/>
- González, L. (2019). *aprendeIA*. Obtenido de <https://aprendeia.com/diferencia-entre-machine-learning-y-la-programacion-tradicional/>
- Google. (2021). *Te damos la bienvenida a Colaboratory*. Obtenido de <https://colab.research.google.com/notebooks/welcome.ipynb?hl=es#scrollTo=OwuxHmxllT wN>
- Google Brain, J. D. (2019). *TENSORFLOW*. Obtenido de <https://www.tensorflow.org/?hl=es-419>
- H. Rahimi-Eichi, U. Ojha, F. Baronti, and M. Chow. (2013). Battery Management System: An Overview of Its Application in the Smart Grid and Electric Vehicles in Industrial Electronics Magazine . *IEEE*, 4-16.
- Hidalgo, O. T. (2019). *Esquema generalizado sobre Machine Learning*. Obtenido de <https://otech.uaeh.edu.mx/noti/index.php/machine-learning/machine-learning-que-es-y-cuales-su-relacion-con-la-ia/>
- Ian Goodfellow, Yoshua Bengio, and Aaron. (2016). *Deep Learning*. Cambridge: The MIT Press.
- Jason Brownlee. (2016). *Master Machine Learning Algorithms: Discover How They Work and Implement Them From Scratch*. Machine Learning Mastery.
- John D. Kelleher, B. M. (2017). *Fundamentals of Machine learning for predictive data analyTIC's algorithms, worked, and case of studies*. London.

- Kern, I. (2019). *Tipos de algoritmos de Inteligencia Artificial y Machine Learning*. Obtenido de <http://pe.globedia.com/tipos-algoritmos-inteligencia-artificial-machine-learning>
- Koehrsen, W. (Junio de 2018). *Overtting vs. Undertting: A Complete Example*. Obtenido de <https://towardsdatascience.com/overfitting-vs-underfitting-a-complete-example-d05dd7e19765>
- Kushal Rashmikant Dalal. (2020). *Analysing the Role of Supervised and Unsupervised Machine Learning in IoT*. *IEEE*.
- Mahanta, J. (Julio de 2017). *Introduction to Neural Networks, Advantages and Applications*. Obtenido de <https://towardsdatascience.com/introduction-to-neural-networks-advantages-and-applications-96851bd1a207>
- Maikel Leyva Vázquez, F. S. (2018). *Neutrosofía: Nuevos avances en el tratamiento de la incertidumbre: Pons Publishing House*. INFINITE STUDY.
- Maldonado S. Edison, M. T. (2018). *Investigación del proceso de Carga (SOC) y descarga (DSOC) de las baterías de alta tensión para estimar su eficiencia y tiempo de vida útil a través de un módulo de corriente constante*.
- Marr, B. (s.f.). *A short history of Machine learning*. Obtenido de <https://www.forbes.com/sites/bernardmarr/2016/02/19/a-short-history-of-machine-learning-every-manager-should-read/?sh=7d2dffe715e7>
- Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro. (2015). *TensorFlow*. Obtenido de TensorFlow: <https://www.tensorflow.org/>
- MathWorksInc. (2020). *Hiperplanos óptimos como límites de decisión*. Obtenido de <https://es.mathworks.com/discovery/support-vector-machine.html>
- Mena, P. O. (2013). *Diseño de un sistema solar fotovoltaico para el suministro de energía eléctrica en el nuevo campus de la escuela politécnica del ejército extensión Latacunga*. Latacunga, Ecuador: Universidad de las Fuerzas Armadas ESPE. Latacunga.
- Minhas, M. S. (Junio de 2021). *Técnicas para manejar el desajuste y el sobreajuste en el aprendizaje automático*. Obtenido de <https://towardsdatascience.com/techniques-for-handling-underfitting-and-overfitting-in-machine-learning-348daa2380b9>
- Orozco, J. A. (2018). *Machine Learning y su importancia en la actualidad*. Obtenido de <https://www.ipade.mx/2018/08/30/machine-learning-y-suimportancia-en-la-actualidad/>
- Paule, R. (2021). *Energías Renovables*. El obscuro precio de la luz. Obtenido de <https://www.energias-renovables.com/ficheroenergias/productos/pdf/Revista670.pdf>
- Peter Krensky, P. d. (2020). *Cuadrante mágico para plataformas de ciencia de datos y aprendizaje automático*. Obtenido de <https://www.gartner.com/doc/reprints?id=1-248D2OTP&ct=200924&st=sbhttps://www.gartner.com/doc/reprints?id=1-248D2OTP&ct=200924&st=sb%20Rapdiminer%202020>
- Pradeep K R, Dr.Naveen N C. (3 de Mayo de 2017). *International Journal of Computer Trends and Technology (IJCTT)*. Obtenido de <https://www.ijcttjournal.org/http://www.ijcttjournal.org/2017/Volume47/number-3/IJCTT-V47P121.pdf>
- Protab. (2018). *Protab Technology Applied to Business*. Obtenido de <http://www.protab.com/>
- RapidMiner. (2020). *Studio RapidMiner 9.8*.
- Rodrigo, J. A. (Febrero de 2020). *Árboles de decisión, random forest, gradient boosting y C5.0*. Obtenido de https://www.cienciadedatos.net/documentos/33_arboles_decision_random_forest_gradient_boosting_c50
- Rodriguez, L. (2004). *COMPONENTES DE UN SISTEMA FOTOVOLTAICOS*. Jaén.
- S. Zuboff. (2019). *The Age of Surveillance Capitalism: The Fight for a Human Future at the New Frontier of Power*. New York: PublicAffairs.

- Shai Shalev-Shwartz, Shai Ben-David. (2014). *Understanding Machine Learning From Theory to Algorithms*. Cambridge University Press.
- Shanthamallu, Uday Shankar, Andreas Spanias, Cihan. (2017). A brief survey of machine learning methods and their sensor and IoT applications. *8th International Conference on Information, Intelligence, Systems & Applications (IISA)* (págs. 1-8). IEE.
- Sngular. (2016). *CRISP-DM: La metodología para poner orden en los proyectos*. Obtenido de <https://www.sngular.com/es/data-science-crisp-dm-metodologia/>
- Statistical, A. S. (2020). *Machine Learning what it is and what it matters*. Obtenido de https://www.sas.com/en_us/insights/analytics/machine-learning.html
- Verhulsdonck, T. (2018). *Un ejemplo avanzado de la clase Estimador de Tensorflow*. Obtenido de <https://towardsdatascience.com/an-advanced-example-of-tensorflow-estimators-part-1-3-c9ffba3bff03>
- Villada, F., Muñoz, N., & García, E. (2016). *Redes Neuronales Artificiales aplicadas a la Predicción del Precio del Oro*.
- Villalba Gómez. (2016). Problemas bioéticos emergentes de la inteligencia. *REVISTA DIVERSITAS PERSPECTIVAS EN PSICOLOGÍA*, 138.
- Walker, J. S. (2018). *Python: La Guía Definitiva para Principiantes para Dominar Python*. Babelcube Inc.
- Yildirim, S. (25 de April de 2020). *Random Forest vs Gradient Boosted Decision Tree*. Obtenido de <https://ai.plainenglish.io/random-forests-vs-gradient-boosted-decision-trees-dd4f0ef86554>
- Zabaleta, J., Biurrum Idoia San, M., & Pascual Miqueleiz, J. (2016). *Cálculo del estado de carga en baterías de plomo-ácido: diseño y validación experimental*.
- Zhihao Wang. (2016). State-of-charge estimation of lithium iron phosphate battery using extreme learning machine. *IEEE*.

ANEXO

Anexo A: Plan de Tesis

1. CARACTERIZACION DEL PROBLEMA

En esta 4ta etapa de la revolución industrial la tecnología va evolucionando rápidamente y para empresas que brindan servicios a industrias mineras deben desarrollar nuevos modelos de negocios basados en innovaciones tecnológicas. La empresa de rubro de telecomunicaciones donde se presentó la problemática es una empresa de integración de proyectos tecnológicos para la automatización de la minería, enfocada al diseño e implementación de Soluciones Tecnológicas Aplicadas que permiten apoyar necesidades de conectividad operacional en minería, en tiempo real, de alta disponibilidad y agregando valor al cliente mediante el uso de la información. Siendo el enfoque de la empresa las Soluciones Tecnologías implementaron equipos que brindan cobertura para el área operacional minero mediante la red Airpro (Red Multiservicio). Para cada equipo que se encuentra instalado en varias zonas del área operacional su flujo de energía es mediante baterías las cuales son energizadas por Paneles Solares. El problema que se viene presentando es los debidos incidentes ocasionados por fallas en energía de los equipos que brindan cobertura por ende si un equipo falla la operación en mina para y esto se ve perjudicado por el tiempo que se brinda para darle un mantenimiento correctivo en energizar a las baterías.

La importancia de este proyecto es dar a conocer la predicción del estado de carga de las baterías de los equipos haciendo uso de la Inteligencia Artificial mediante el subcampo del Aprendizaje Automático para así prevenir incidencias a largo plazo

que perjudiquen la operación en mina y planificar mantenimientos preventivos si es necesario.

1.1.LINEA Y SUBLINEA

- **Línea:** Inteligencia Artificial.
- **Sub-línea:** Aprendizaje Automático.

1.2.PALABRAS CLAVES

Predicción, Estado de Carga, Sistemas Fotovoltaicos, Aprendizaje Automático Supervisado.

2. JUSTIFICACION

El sector fotovoltaico en los últimos años se ha convertido en una de las mayores industrias a nivel mundial utilizando la energía solar para la reducción de costos de energía.

La Empresa de rubro de telecomunicaciones donde se presentó la problemática es participe en la implementación de los sistemas fotovoltaicos para la energización de sus equipos que brindan cobertura para la operatividad en sectores mineros. Cada equipo cuenta con baterías energizadas por paneles solares y son monitoreadas en un Sistema de Monitoreo que registra cada variable de energía programado en el sistema fotovoltaico; sin embargo no se lleva un buena toma de datos ya que las personas encargadas no realizan un buen análisis de las variables de energía provocando así que sucedan incidentes recurrentes por la descarga de las baterías conllevando así que la operatividad de la mina pare y se tenga pérdidas de dinero

por el tiempo que se brinde para darle un mantenimiento correctivo en energizar las baterías.

Con las técnicas de aprendizaje automático se podrá saber la predicción del estado de carga de las baterías en sistemas fotovoltaicos obteniendo resultados en tiempo real. Con esta información personal del Área de Monitoreo y personal de Terreno en Mina podrá visualizar y analizar las tendencias en el estado de carga de las baterías de sus equipos.

3. OBJETIVOS DEL PROYECTO

3.1 GENERAL

- Desarrollar la predicción del estado de carga de baterías para sistemas fotovoltaicos en el sector minero utilizando técnicas de aprendizaje automático supervisado.

3.2 ESPECÍFICOS

- Analizar el proceso de la toma de datos de las variables de energía de los equipos actuales de la empresa.
- Aplicar técnicas de aprendizaje automático supervisado que permitan predecir con gran exactitud el estado de carga de las baterías de los equipos.
- Determinar la técnica de aprendizaje automático supervisado más adecuada para la predicción del estado de carga de las baterías de los equipos.
- Validación de los resultados en la predicción del estado de carga de las baterías para sistemas fotovoltaicos.

- Desarrollar un módulo informático web que integre el aprendizaje automático supervisado propuesto.

4. VARIABLES

4.1 INDEPENDIENTE

- Aprendizaje Automático.

4.2 DEPENDIENTE

- Predicción del estado de carga de baterías para sistemas fotovoltaicos en el sector minero.

5. OPERACIONALIZACION DE VARIABLES

Tabla A1 Operacionalización de Variables

VARIABLE	SUB VARIABLE	INDICADORES
Aprendizaje Automático. (Variable Independiente)	Aprendizaje Supervisado.	Predicción del Estado de Carga de Baterías en Sistemas Fotovoltaicos.
Predicción del estado de carga de baterías para sistemas fotovoltaicos en el sector minero. (Variable Dependiente)	Clasificación	Precisión Exactitud Especificidad Sensibilidad Curva ROC

Fuente: Elaboración Propia

6. MARCO TEORICO

6.1 ESTADO DEL ARTE

En esta 4ta etapa de la revolución industrial en las que nos encontramos actualmente las tecnologías de la información tienen como propósito llevar a cabo un cambio digital que las haga inteligentes.

El aprendizaje automático es una de las áreas de la informática de más rápido crecimiento, con aplicaciones de gran alcance. Se refiere a la detección automatizada de patrones significativos en los datos. Las herramientas de aprendizaje automático se preocupan por dotar a los programas de la capacidad de aprender y adaptarse. (Shai Shalev-Shwartz, Shai Ben-David, 2014)

El aprendizaje automático entrega la garantía de extraer importantes fuentes de datos grandes y distintas a través de una menor dependencia periférica programada en una pista individual, ya que se determinan los datos y se dispara a escala de máquina

El aprendizaje automático es adecuado para la complejidad del manejo a través de un origen de datos diferente y la amplia gama de variables, así como la cantidad de datos involucrados donde el aprendizaje automático prospera en conjuntos de datos crecientes. Cuanto más se suministren datos a una estructura de aprendizaje automático, más se podrá capacitar y más se podrá capacitar y se tendrán en cuenta las consecuencias del valor superior de los conocimientos. Al estar libre de los confines del pensamiento y el estudio a nivel individual, Machine Learning es inteligente para descubrir y mostrar los patrones ocultos en los datos. (Pradeep K R, Dr.Naveen N C, 2017)

Partiendo de estos conceptos sobre el aprendizaje automático es posible extraer y almacenar una gran cantidad de datos no obstante se requiere la aplicación de algoritmos de Aprendizaje Automático para crear modelos que den valor a esos datos y hagan de mayor facilidad la toma de decisiones.

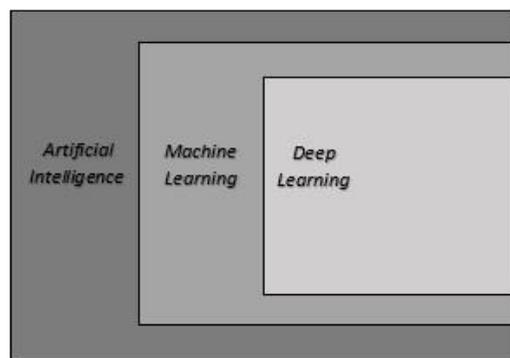


Figura A1. Estructura de los métodos de aprendizaje.

Fuente: Elaboración Propia

Las técnicas de aprendizaje supervisado se enfocan en utilizar una combinación etiquetada de datos para entrenar un algoritmo que sirva al propósito de la mejor función para describir la selección de datos de entrada. Como resultado, los resultados correspondientes y las entradas conocidas se presentan para el aprendizaje. Con el tiempo, estos conjuntos de datos ayudan a las máquinas a identificar una salida específica para una entrada suministrada. Puede existir el desempeño del aprendizaje supervisado cuando se han definido ciertos objetivos para alcanzar un conjunto específico de insumos. Para tal aprendizaje, es importante enfrentar los datos de la etiqueta y luego proporcionar capacitación a la máquina a través de los datos etiquetados de los resultados deseados y las entradas reales. (Jason Brownlee, 2016)

La técnica de aprendizaje no supervisado utiliza datos sin etiquetar para entrenar el algoritmo. Con el tiempo, ayuda a detectar un patrón al describir un modelo. En tal caso, los datos de entrada ayudan al algoritmo a extraer reglas y regulaciones junto con la detección de patrones relevantes. Los puntos importantes se resumen y combinan con el tiempo para proporcionar significado y una mejor comprensión. (Shanthamallu, Uday Shankar, Andreas Spanias, Cihan, 2017)

Tabla A 2: Lista de clasificaciones de algoritmos de Aprendizaje Automático Supervisado

ML Algorithms	Supervised Learning Method
K-Nearest Neighbors	Supervised ML
Decision Tree and Random Forest	Supervised ML
Support Vector Machines (SVM)	Supervised ML
Recurrent Neural Networks (RNN)	Supervised ML

Fuente: Dalal (2020)

Cabe concluir que las técnicas de aprendizaje supervisado son empleados en modelos predictivos utilizando algoritmos que aprenden de los datos de entrada generando datos de salida esperados siendo la entrada etiquetada y clasificada por alguien y por otro lado las técnicas de aprendizaje no supervisado donde los algoritmos aprenden de datos no etiquetados buscando patrones o relación entre ellos.

La tecnología de estimación del estado de carga (SOC) es importante para el desarrollo de sistemas de almacenamiento de energía de baterías de iones de litio.” (H. Rahimi-Eichi, U. Ojha, F. Baronti, and M. Chow, 2013), (K. W. E. Cheng, B. P. Divakar, H.

Wu, K. Ding, and H. F. Ho., 2011). “Un SOC preciso es el elemento principal para el funcionamiento seguro de los paquetes de baterías. También es una base de referencia para el control de ecualización celular. La función de un equilibrador de celdas es mantener todas las celdas igualadas durante la carga y descarga para maximizar la capacidad del paquete y proteger las celdas de sobrecarga o descarga. Puede provocar daños permanentes cuando una celda se sobrecarga o se descarga en exceso. Por lo tanto, mejorar la precisión del SOC estimado hace una diferencia en la mitigación del daño de la celda, prolongando el ciclo de vida y disminuyendo el costo de mantenimiento. (Zhihao Wang, 2016)

La toma de data de las variables de energía para los equipos que brindan cobertura radial las cuales se encuentran en mina pueden ser muy útiles para estimar la predicción del estado de carga de las baterías utilizando técnicas de Aprendizaje Automático Supervisado teniendo como datos de entradas el voltaje de las baterías, temperatura de las baterías, corriente consumo, corriente carga y corriente paneles. “Los enfoques basados en datos de aprendizaje automático para la estimación del estado de la batería han sido impulsados por avances recientes en inteligencia artificial (IA)” (S. Zuboff, 2019) , “en campos como la visión por computadora y los vehículos autónomos. El diagrama de Venn de la Fig. 2 muestra cómo se subdivide el campo de la IA, incluido el aprendizaje automático y sus subsecuentes divisiones de aprendizaje de representación y aprendizaje profundo”. (Ian Goodfellow, Yoshua Bengio, and Aaron, 2016)

Este proyecto estará enfocado en desarrollar e implementar un módulo informático web a partir de técnicas de aprendizaje automático supervisado con la finalidad de predecir el estado de carga de las baterías de sistemas fotovoltaicos empleados en equipos que brindan cobertura radial.

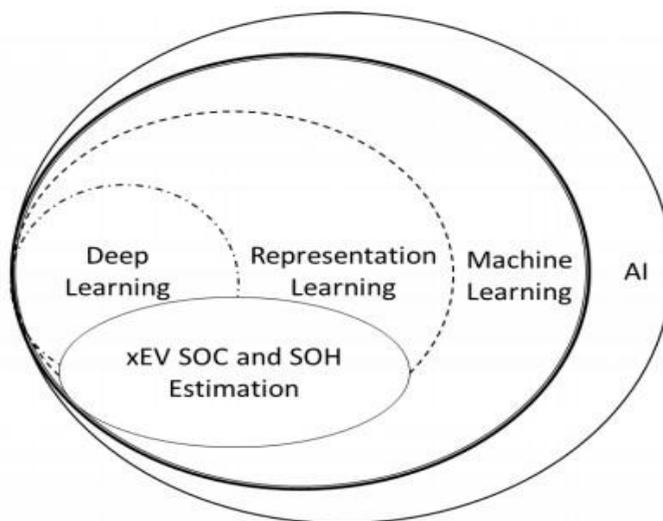


Figura A2. Diagrama de Venn que muestra la relación de xEV SOC y SOH Estimación al campo de la inteligencia artificial y el aprendizaje automático.

Fuente: Goodfellow, Bengio, & Aaron (2016)

7. BASES TEORICAS

7.1 INTELIGENCIA ARTIFICIAL

La Inteligencia Artificial tiene por objetivo el estudio y el análisis del comportamiento humano en los ámbitos de la comprensión, la percepción, la resolución de problemas y la toma de decisiones con el fin de poder reproducirlos con la ayuda de un computador, esta simulación de inteligencia humana incluye procesos de aprendizaje, de razonamiento y de mejoramiento.

La Inteligencia Artificial ha estructurado una plataforma tecnocientífica mediante la puesta en marcha de una disciplina transversal que, tiene por finalidad entender, modelar y replicar los procesos cognitivos típicos de la inteligencia humana integrando en su esfuerzo principios propios de la matemática, la lógica, la biología y la mecánica, en aras de incorporarlos a máquinas auto controladas. (Villalba Gómez, 2016).

7.2. APRENDIZAJE AUTOMÁTICO

El aprendizaje automático consiste en programar computadoras para optimizar un criterio de rendimiento utilizando datos de ejemplo o experiencias pasadas. Tenemos un modelo definido hasta algunos parámetros y el aprendizaje es la ejecución de un programa informático para optimizar los parámetros del modelo utilizando los datos de entrenamiento o la experiencia pasada. (Alpaydin, 2020)

7.3. Tensorflow

Tensorflow es una plataforma de código abierto de extremo a extremo para el aprendizaje automático. Cuenta con un ecosistema integral y flexible de herramientas, bibliotecas y recursos de la comunidad que les permite a los investigadores impulsar un aprendizaje automático innovador y, a los desarrolladores, compilar e implementar con facilidad aplicaciones con tecnología de Aprendizaje Automático. (MartIn Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, 2015)

8. CRONOGRAMA DE TRABAJO

Tabla A3 Cronograma de Trabajo

CRONOGRAMA ACTIVIDADES	MESES			
	Octubre	Noviembre	Diciembre	Enero
Presentación de plan de tesis	x			
Calificación y aprobación de plan de tesis	x			
Recolección de información	x			
Procesamiento de la información	x			
Elaboración del primer avance		x		
Presentación del primer avance calificado y firmado		x		
Desarrollo del proyecto		x		
Análisis e interpretación de resultados		x		
Elaboración del segundo avance			x	
Presentación del segundo avance calificado y firmado			x	
Correcciones y recomendaciones			x	
Elaboración del tercer avance				x
Conclusiones				x
Trabajo final de tesis				x
Sustentación				x

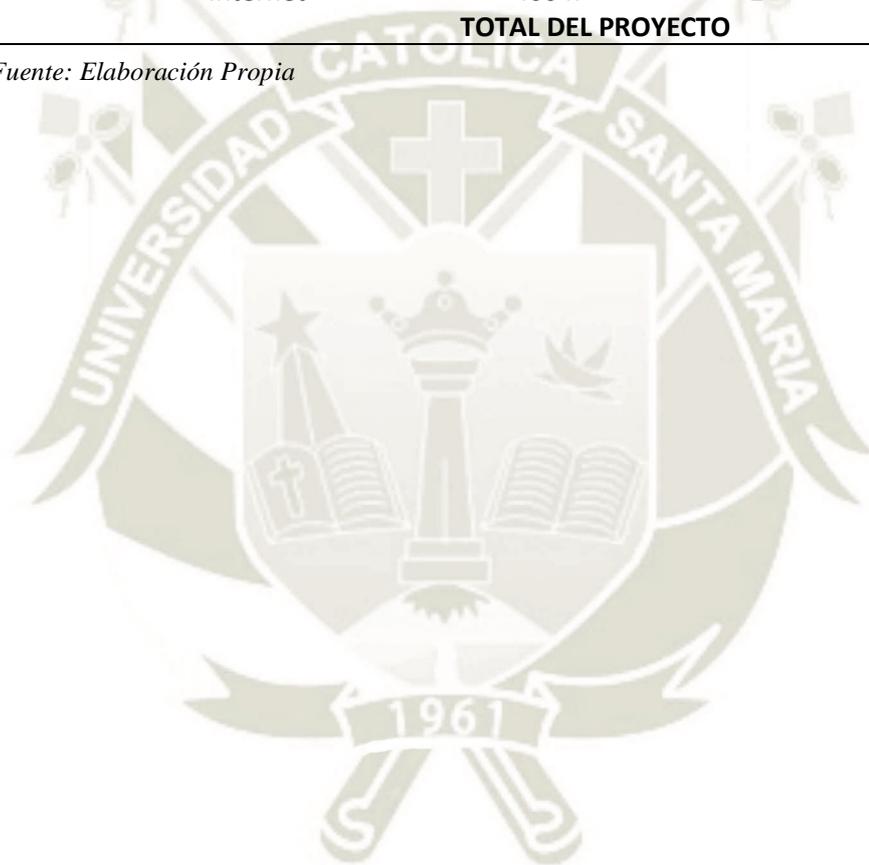
Fuente: Elaboración Propia

9. PRESUPUESTO

Tabla A4 Presupuesto

	Recursos	Cantidad	Costo	Monto
Personal	Tiempo	400 h		
	Computador	1	2000	2000
	Impresora	1	700	700
Bienes	Gs. Google	1	20	20
	Útiles Escritorio	2	2.5	5
	Disco Externo	1	200	200
	SW Licencias	0	0	0
Servicios	Luz	90 días	120	360
	Internet	400 h	1	400
TOTAL DEL PROYECTO				3685

Fuente: Elaboración Propia



10. REFERENCIAS BIBLIOGRAFICAS

- Alpaydin, E. (2020). *Introduction to Machine Learning*. The MIT Press.
- H. Rahimi-Eichi, U. Ojha, F. Baronti, and M. Chow. (2013). Battery Management System: An Overview of Its Application in the Smart Grid and Electric Vehicles in *Industrial Electronics Magazine . IEEE*, 4-16.
- Ian Goodfellow, Yoshua Bengio, and Aaron. (2016). *Deep Learning*. Cambridge: The MIT Press.
- Jason Brownlee. (2016). *Master Machine Learning Algorithms: Discover How They Work and Implement Them From Scratch*. Machine Learning Mastery.
- K. W. E. Cheng, B. P. Divakar, H. Wu, K. Ding, and H. F. Ho. (2011). Battery-Management System (BMS) and SOC Development for Electrical Vehicles," in *Vehicular Technology. IEEE*, 76-78.
- Kushal Rashmikant Dalal. (2020). Analysing the Role of Supervised and Unsupervised Machine Learning in IoT. *IEEE*.
- Martin Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro. (2015). *Tensorflow*. Obtenido de Tensorflow: <https://www.tensorflow.org/>
- Pradeep K R, Dr.Naveen N C. (3 de Mayo de 2017). *International Journal of Computer Trends and Technology (IJCTT)* . Obtenido de <https://www.ijcttjournal.org/>: <http://www.ijcttjournal.org/2017/Volume47/number-3/IJCTT-V47P121.pdf>
- S. Zuboff. (2019). *The Age of Surveillance Capitalism: The Fight for a Human Future at the New Frontier of Power*. New York: PublicAffairs.
- Shai Shalev-Shwartz, Shai Ben-David. (2014). *Understanding Machine Learning From Theory to Algorithms*. Cambridge University Press.

Shanthamallu, Uday Shankar, Andreas Spanias, Cihan. (2017). A brief survey of machine learning methods and their sensor and IoT applications. *8th International Conference on Information, Intelligence, Systems & Applications (IISA)* (págs. 1-8). IEE.

Villalba Gómez. (2016). Problemas bioéticos emergentes de la inteligencia.

REVISTA DIVERSITAS PERSPECTIVAS EN PSICOLOGÍA, 138.

Webfleet Solutions B.V. (2006). *Solución de gestión de flotas de WEBFLEET*.

Obtenido de https://www.webfleet.com/es_es/webfleet/

Zhihao Wang. (2016). State-of-charge estimation of lithium iron phosphate battery using extreme learning machine. *IEEE*.

