ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL

Facultad de Ingeniería en Electricidad y Computación

“Predicción y visualización de la demanda Eléctrica del Campus Gustavo Galindo ESPOL”

PROYECTO DE TITULACIÓN

Previo la obtención del Título de:

Magister en Ciencias de Datos

Presentado por:

Francisco Xavier Porras Carrión

GUAYAQUIL - ECUADOR

Año: 2024

AGRADECIMIENTO

A los profesores del programa de Maestría por su tiempo y dedicación, a mis compañeros de grupo por su ayuda durante el desarrollo de los cursos y a mi hermano por la recomendación del programa de maestría.

DEDICATORIA

A mi esposa, mis hijos José y Xavier, por su apoyo siempre incondicional en este proyecto su paciencia durante su desarrollo.

DECLARACIÓN EXPRESA

“Los derechos de titularidad y explotación, me corresponde conforme al reglamento de propiedad intelectual de la institución; *Francisco Xavier Porras Carrión* y doy mi consentimiento para que la ESPOL realice la comunicación pública de la obra por cualquier medio con el fin de promover la consulta, difusión y uso público de la producción intelectual"

Firmado electrónicamente por:
FRANCISCO XAVIER
PORRAS CARRION

Francisco Porras Carrión

COMITÉ EVALUADOR

Firmado electrónicamente por:
JOSE EDUARDO
CORDOVA GARCIA

.....

Ph.D José Córdova García, Ph.D

PROFESOR TUTOR

Firmado electrónicamente por:
LENIN EDUARDO
FREIRE COBO

.....

MSc. Lenin Freire Cobo

PROFESOR EVALUADOR

RESUMEN

La predicción de la demanda Eléctrica es esencial para la correcta planeación de recursos, adquisición de energía Eléctrica a bajo costo, dimensionamiento de sistemas de transmisión, adquisición de equipos con consumo más eficiente, inversión en nuevas alternativas de generación con un mínimo impacto en el medio ambiente.

El presente trabajo tiene como objetivo Desarrollar una herramienta de Visualización y de predicción de la demanda Eléctrica del campus Gustavo Galindo de la ESPOL usando técnicas de aprendizaje de máquina.

Para la predicción de la demanda de Energía Eléctrica, se evaluó modelos de aprendizaje profundo con redes recurrentes en sus tres variantes principales es decir Red neuronal recurrente o RNN, red LSTM o Modelo de largo y corto plazo y GRU o modelo de unidad de compuertas recurrentes para determinar cuál de estas ofrece un mejor comportamiento a la hora de predecir una serie temporal multivariante.

El prototipo de herramienta de visualización desarrollado cumple con los aspectos fundamentales de una herramienta de este tipo, es decir es interactivo y es fiel reflejo del comportamiento de los datos.

Al unir ambos componentes se obtuvo un prototipo de herramienta tanto para predecir como para visualizar e interactuar con la información.

Se concluye la factibilidad en el uso de modelos de aprendizaje de máquina profundo en sus variantes de redes recurrentes, el prototipo de herramienta de visualización propuesto es sencillo y cumple con dos preceptos principales los cueles son: fiel representación de los datos y permite la interacción con dichos datos.

ABSTRACT

The prediction of the Electricity demand is essential for the correct planning of resources, acquisition of Electric energy at low cost, dimensioning of transmission systems, acquisition of equipment with more efficient consumption, new generation alternatives with a minimum impact on the environment.

The objective of this work is to develop a tool for the Visualization and prediction of the Electricity demand of the Gustavo Galindo campus of ESPOL using machine learning techniques.

For the prediction of Electric Power demand, deep learning models were used with recurrent networks in their three main variants, that is, Recurrent Neural Network or RNN, LSTM network or Long and short-term model and GRU or recurrent gate unit model. to determine which of these offers the best performance when predicting a multivariate time series.

The proposed visualization tool prototype complies with the fundamental aspects of a tool of this type, that is, it is interactive and is a true reflection of the behavior of the data.

By joining both components, a prototype of a tool was obtained both to predict and to visualize and interact with the information.

The feasibility in the use of deep machine learning models in their recurrent network variants is concluded, the proposed visualization tool prototype is simple and complies with two main precepts, faithful representation of the data and allows interaction.

Keywords: Recurrent networks, predictions, Energy, visualization, model, consumption.

INDICE GENERAL

ABREVIATURAS

RRN……. Redes recurrentes

LSTM…….. Redes de corto y largo plazo

GRU……. Unidad de compuertas recurrentes

ESPOL……. Escuela Superior Politécnica del Litoral

ABS…….. Valor Absoluto

SIN…… Sistema Nacional Interconectado

CERA…….. Centro de energías Renovables y Alternativas

INDICE DE FIGURAS

INDICE DE TABLAS

CAPÍTULO 1

1 PLANTEAMIENTO DE LA PROBLEMÁTICA

La predicción de la demanda eléctrica permite realizar una adecuada planificación de recursos, integración con energías renovables, y evaluación de políticas de eficiencia energética (Mastronardi et al., 2016), por lo que es importante analizar las variables que influyen en el consumo de los sistemas de energía eléctrica como: clima, cantidad de usuarios, espacio físico. Como se menciona en (Mastronardi et al., 2016) la existencia de crecimiento del consumo durante picos estacionales de verano e invierno, producto de acondicionadores y calefactores eléctricos. Este aumento por ejemplo provoca un incremento en la inversión de mantenimientos programados en el sistema de suministro eléctrico, el cual además tiene que ser dimensionado para cubrir dichos picos.

De acuerdo con el estudio (Amber et al., 2015) el cual se condujo en un edificio de tipo administrativo en el "Technopark" ubicado en London South Bank University (LSBU), se indica que un modelo confiable de predicción para el consumo eléctrico es deseable para ayudar a los administradores de recursos energéticos a realizar diversas tareas entre las cuales destacan:

- Identificar las variables que afectan principalmente al consumo eléctrico
- Identificación de potenciales ahorros
- Desarrollo de políticas y mejora de las instalaciones de producción y distribución de electricidad

Para las empresas tanto de provisión como de consumo eléctrico, una herramienta de visualización de la línea base del consumo así como de las variables relacionadas y un modelo confiable de predicción ayuda a comprender y predecir el consumo eléctrico durante los diferentes períodos del año.

En (Oquendo, 2016), se muestra una aplicación de redes neuronales artificiales en el pronóstico de la demanda eléctrica a corto plazo en el Sistema Nacional Interconectado (SNI), los resultados indican que la demanda eléctrica sigue un perfil

de carga horaria el cual se ve influenciado por las características del usuario, así como de variables climáticas.

En diversos estudios como los realizados por (Del Carmen Ruiz-Abellón et al., 2018; San Miguel Salas, 2016; Vaghefi et al., 2015; Yuan et al., 2018), se menciona lo exitoso de la aplicación de modelos de redes neuronales y de Maquinas de soporte vectorial o SVM por sus siglas en ingles en la predicción de la demanda de Energía eléctrica con base a parámetros meteorológicos.

En (Rodriguez & Massa, 2014) los autores indican que mejorar el modelo de predicción de demanda implica una mejora en la gestión energética y por tanto ahorros en costos de operación, lo cual es importante para las empresas vinculadas a la generación y satisfacción de la demanda.

1.1 **Descripción del problema**

Se menciona anteriormente que uno de los factores que inciden en el consumo eléctrico es el clima, siendo la ciudad de Guayaquil donde se encuentra ubicado el Campus Gustavo Galindo de la Escuela Superior Politécnica del Litoral de clima cálido y húmedo, el uso de acondicionamiento de aire es intenso para el desarrollo de sus actividades académicas, de investigación y administrativas.

Con el presente trabajo se busca proponer una herramienta para la visualización de los datos provistos por la ESPOL, para las variables climáticas, para el consumo de energía eléctrica, además, esta herramienta permitirá la predicción usando un modelo de aprendizaje de máquina profundo aplicado al consumo de electricidad para el campus Politécnico Gustavo Galindo. La herramienta permitirá establecer una línea base de consumo de electricidad y de predicción, para mejorar la gestión energética del Campus y evaluar políticas de eficiencia energética.

1.2 Justificación

Actualmente la ESPOL y específicamente el Centro de Energías Renovables y Alternativas CERA por sus siglas, no cuenta con una herramienta de visualización de datos y predicción del consumo eléctrico basado en métodos de aprendizaje de máquina, que permita establecer una línea base de consumo y predicción, basado en la influencia que tiene el clima en el consumo eléctrico del Campus.

Es importante anotar que la implementación de una herramienta de predicción y de visualización del consumo permitirá una mejor planificación por parte de los administradores del campus.

1.3 Solución Propuesta

Uno de los principales aspectos que se debe tener en cuenta, por parte del personal que administra los recursos del campus, es el de conocer cuál es el histórico del consumo eléctrico, cómo varía a lo largo del tiempo, y como las variables climáticas afectan su comportamiento, a esto lo denominamos “Línea Base de Consumo Eléctrico”.

Una vez que se cuenta con la línea base de Consumo eléctrico y su relación con las variables climáticas, lo siguiente será el poder predecir el consumo eléctrico teniendo como variables predictoras a las variables climáticas, permitiendo a los administradores del Campus y de CERA saber cómo se comportará el campus bajo condiciones climáticas previstas, dichas predicciones las realizaremos usando redes aprendizaje de máquina.

Se plantea además una herramienta que permite graficar la línea base de consumo, las variables climáticas y las predicciones, además de poder interactuar con la gráfica para observar cómo varía su comportamiento en los diferentes intervalos de tiempo seleccionados.

A continuación, se establece los objetivos a alcanzar en el desarrollo de este proyecto, junto con la metodología general que se implementará para desarrollar la solución propuesta.

El capítulo de Estado del Arte se describe la teoría sobre los algoritmos de predicción que se analizarán en el presente estudio, en el capítulo Diseño e Implementación se describe los pasos que se siguieron para la elaboración de la solución, en el Capítulo de Análisis de Resultados analizaremos las métricas de la predicción y seleccionaremos el modelo que mejor realizó las predicciones, se describirá la puesta en marcha del prototipo y pruebas para finalmente revisar los costos y beneficios de la solución.

Finalmente se proponen las conclusiones y recomendaciones del presente trabajo.

1.4 Objetivos

1.4.1 Objetivos generales

- Desarrollar una herramienta de visualización y de predicción de la demanda eléctrica del campus Gustavo Galindo de la ESPOL usando técnicas de aprendizaje de máquinas.

1.4.2 Objetivos específicos

- Establecer una base de la demanda eléctrica del campus usando datos históricos del consumo eléctrico
- Evaluar modelos de aprendizaje de máquina para predecir el consumo eléctrico incorporando variables climatológicas estableciendo pronósticos a diferentes horizontes de tiempo.
- Desarrollar una herramienta de procesamiento y de visualización de los datos para facilitar el análisis del consumo de energía eléctrica del campus.

1.5 **Metodología**

Para el desarrollo del presente proyecto usaremos la metodología base de los proyectos de Ciencia de datos aprendida durante el desarrollo de la Maestría en Ciencias de Datos de la ESPOL, el aplicar esta metodología nos permitirá tratar el problema de forma sistemática hasta su resolución, se presenta a continuación los diferentes pasos que deben ser realizados a fin de cumplir con esta metología.

1.5.1 Problema o pregunta relevante a la investigación

La metodología de Ciencias de Datos nos indica que como primer paso se requiere conocer el problema y plantearnos una pregunta relevante o problema a resolver. Teniendo en cuenta esta primera etapa se planteó al departamento de Sostenibilidad, así como al CERA, la siguiente pregunta:

¿“Existe una herramienta de visualización y de predicción de consumo de energía del Campus Gustavo Galindo de la Escuela Superior Politécnica del Litoral basada en algoritmos de aprendizaje de máquina, que permita a los centros de Gestión y de Investigación, como el CERA, tomar decisiones basadas en datos” ?

La respuesta a esta interrogante fue que no existe actualmente dicha herramienta, el pronóstico se hace actualmente mediante soluciones en hojas de cálculo sin incluir en el análisis al clima como variable predictora.

1.5.2 Obtención de los datos

El siguiente paso es la obtención de los datos, estos se solicitaron al Programa Sostenibilidad de la ESPOL, firmando un acuerdo de confidencialidad (NDA), el cual se adjunta como anexo.

Para el desarrollo de las exposiciones se anonimizará los datos para honrar los acuerdos de confidencialidad con la ESPOL, por lo que los datos reales solo se mostraran a los beneficiarios finales del producto, en este caso el Programa de Sostenibilidad de la ESPOL.

1.5.3 Limpieza y verificación de novedades del conjunto de datos

Se iniciará con la limpieza y revisión del conjunto de datos, en este apartado se trata de revisar que la data esté completa, no exista duplicidad y en caso de ser necesario eliminar registros duplicados, imputar los datos de ser factible, eliminar registros que se encuentran con valores que no corresponden a la serie temporal.

1.5.4 Exploración de los datos

La exploración y visualización de los datos de forma temprana permite revisar las variables de clima o también llamadas predictoras y de consumo eléctrico o variable de análisis, en este paso se realiza la estadística descriptiva de los datos, su correlación y comportamiento de las observaciones.

Se define la resolución de trabajo para los datos, es decir si estos serán con observaciones cada 15 minutos o de forma horaria a fin de poder mantener una sincronía tanto en variables predictoras como para la variable de análisis.

1.5.5 Enriquecimiento de datos

El dataset es enriquecido por variables explicativas de tiempo, en el estudio conducido por (Al-Qahtani & Crone, 2013) se indica que las variables predictoras pueden enriquecer al modelo aportando data adicional, en este caso las variables de tiempo que se agregaron como data predictora son extraídas de la variable tiempo, estas variables son: año, mes, día, hora y día de la semana, respectivamente nombradas “año”, “mes”, “día”, “hora”, “Dia de semana”, también se incluyó una variable binaria denominada “Pandemia” para identificar los datos posteriores al 17 de marzo de 2020 a las 00:00:00 donde se puso en vigencia el periodo de pandemia en el territorio nacional.

La variable hora es además convertida mediante la transformada de Fourier a dos series compuestas por seno y coseno para poder recuperar la característica cíclica que tiene y es una técnica recomendada en (Peixeiro, 2022).

1.5.6 Normalización de las variables

El proceso de normalización de datos tiene por objetivo escalar los valores de cada variable a un rango [0, 1], esto se realiza con el objetivo de que los modelos de aprendizaje de maquina sean más eficientes al momento de realizar los cálculos matriciales.

1.5.7 Modelo Predictivo

Posterior a la normalización de datos se encuentra el desarrollo de un modelo determinístico de predicción de la demanda, usando aprendizaje de máquina, la revisión bibliográfica es importante para conocer las técnicas usadas para el análisis y predicción de series temporales, citamos las siguientes (Agarwal et al., 2009; Ahmad et al., 2017; Andrić et al., 2019; Azar & Menassa, 2012; Del Carmen Ruiz-Abellón et al., 2018).

Los estudios antes citados realizan predicciones de consumo eléctrico usando modelos de aprendizaje de máquina, dichos estudios son variados y utilizan tanto datos univariantes como multivariantes, y de la misma forma son amplios en el uso de algoritmos.

Dentro de este apartado se realizará las evaluaciones a los modelos ajustando los hiper parámetros correspondientes y comparando los modelos de aprendizaje de máquina para la selección del modelo que presente el mejor desempeño en la predicción a diferentes horizontes de tiempo.

1.5.8 Visualización

Posterior a la revisión y selección del modelo predictivo se realiza el prototipo para la visualización tanto de la línea base y las variables climáticas.

El prototipo de herramienta de visualización además muestra el resultado de la predicción que se realiza con los datos disponibles.

1.6 **Resultados esperados**

Al finalizar el proyecto se espera encontrar el mejor modelo de aprendizaje de máquina que se ajuste a la variable de consumo de energía eléctrica y condiciones climáticas, así como proveer al CERA de una herramienta de visualización y exploración de los datos que ayude a la toma de decisiones de gestión de energía del campus.

1.7 **Dataset**

Se realiza una primera revisión del conjunto de datos o dataset, a continuación, se describe cada uno de los estadísticos descriptivos:

- Número de Observaciones. _ Es el número total de observaciones totales en el conjunto de datos.
- Valor Promedio. _ Se define como la suma de los valores de la característica dividida para el número de observaciones.
- Desviación estándar. _ Es una medida que indica que tan dispersos se encuentran los datos en relación con su media como lo indica la ecuación (1.1)

$$\sigma = \sqrt{\frac{\sum_1^n (x_i - \bar{x})}{n}} \qquad (1.1)$$

- Valor mínimo. _ El valor menor de entre todas las observaciones de una característica.
- Valor máximo. _ El valor mayor de entre todas las observaciones de una característica.

- Primer Cuartil. _ 25% de los datos es menor o igual al valor indicado.
- Segundo Cuartil. _50% de los datos es menor que o igual a este valor, también representa la mediana.
- Tercer Cuartil. _75% de los datos es menor que o igual a este valor, o dicho en otras palabras 25% de las observaciones son mayores o iguales al valor indicado.

Los conjuntos de datos o “Dataset” necesarios para el presente estudio se obtuvieron por medio del Programa de Sostenibilidad de ESPOL(dataset Medidor Principal), además las variables meteorológicas (Dataset Clima) fueron obtenidas del Centro de Energías Renovables y Alternativas de ESPOL.

El conjunto de datos del Medidor Principal contiene la información referente a Fecha, hora y ENERGIA ACTIVA, el primer registro es de 2017-01-01 00-00-00 y el último registro el 2020-12-18 00-00-00, las mediciones se realizan cada 15 minutos, es decir que para cada hora de medición se generan 4 registros, el dataset cuenta con 136897 registros.

Se presenta en la **Tabla 1.1** los valores estadísticos que fueron definidos previamente:

Tabla 1.1 Estadísticos Consumo Eléctrico ESPOL 2017 – 2020

	ENERGIA ACTIVA
número de observaciones	136897
Valor promedio	360.47
desviación estándar	255.62
Valor mínimo	0.00
Primer Cuartil 25%	201.60
Segundo Cuartil 50%	226.80
Tercer Cuartil 75%	415.80

Valor máximo	1260.00

Fuente: El Autor

El conjunto de datos “Datos_meteorologicos_Edificio_Litoral” contiene las siguientes variables: Fecha, Tiempo GMT-05:00,Temp °C,Humedad relativa (HR)%, Lluvia mm (milímetros), Dirección del viento, Velocidad del viento m/s y Velocidad de Rafagas m/s, el primer registro es de 2016-01-21 02-00-00 y el último registro el 2020-07-27 10-01-40, las mediciones se realizan cada 10 minutos, por tanto por cada hora de medición se generan 6 registros, el conjunto de datos cuenta con 254937 observaciones.

Se presenta en la **Tabla 1.2** las principales estadísticas descriptivas del conjunto de datos de Clima de la Espol.

Tabla 1.2 Estadísticos Variables Clima ESPOL 2016 - 2020

	Temperatura °C	HR. %	Lluvia mm	Dirección del viento	Velocidad del viento m/s	Velocidad de Rafagas m/s
Número de observaciones	254937	254937	254937	254937	254937	254937
Valor promedio	25.95	81.59	0.02	192.56	-213.92	-212.17
desviación estándar	3.04	9.80	0.24	64.70	380.73	381.73
Valor mínimo	18.37	45.30	0.00	0.00	-888.88	-888.88
Primer Cuartil 25%	23.62	74.50	0.00	167.10	0.00	0.00
Segundo Cuartil 50%	25.72	83.20	0.00	195.10	0.50	2.52
Tercer Cuartil 75%	28.17	89.10	0.00	213.40	1.01	3.78
Valor máximo	35.96	100.00	17.60	355.20	4.03	10.32

Fuente: El Autor

En la **Tabla 1.2** se puede apreciar valores marcados como -888.88, este valor se registra como una inconsistencia en la data, estos valores se podrían imputar o reemplazar por ejemplo con valores tomados de centros meteorológicos de la Ciudad de Guayaquil, sin embargo y como veremos

luego debido a que la primera observación en el conjunto de datos de consumo es del 2017-01-01 00-00-00, estos valores marcados como -888.88 son eliminados.

CAPÍTULO 2

2 ESTADO DEL ARTE

A continuación, describiremos las nociones generales sobre series temporales, aprendizaje de máquina, haciendo énfasis en redes neuronales recurrentes, revisaremos además trabajos relacionados con la predicción usando redes recurrentes para la comparación del presente trabajo y finalmente las medidas para evaluar la precisión en la predicción de los modelos.

2.1 **Predicción de series temporales**

Una serie temporal se define como una secuencia de valores que están ordenados de acuerdo con el tiempo en el que se observaron, y las observaciones se realizan de forma periódica o intervalos regulares.

La predicción de series temporales se puede realizar de diversas formas entre ellas usando modelos probabilísticos, modelos predictivos con aprendizaje de máquina entre otras, con data univariante es decir la serie a predecir es la entrada al modelo, así como series multivariantes las cuales constan de dos o más variables y al menos una de las variables es la variable denominada dependiente. En cuanto a los modelos estos van desde el modelo ingenuo o también llamado “naive” donde el modelo repite el último valor de la serie hasta modelos complejos donde se analiza su estacionalidad, ciclicidad y la tendencia, es el caso de los modelos de ARIMA, estos modelos son del tipo probabilístico es decir asumen un intervalo de confianza en sus pronósticos (Martin, 1987).

Otra forma de predecir series temporales es usando aprendizaje de máquina, como lo menciona (Torres et al., 2021), donde menciona el uso extendido que se le ha dado al aprendizaje de máquina en el pronóstico de Series temporales y más aún en series con gran cantidad de observaciones o “Big data” por su término en inglés.

En el estudio conducido por (Fathi & Srinivasan, 2019), se indica que el 50% del consumo de la energía se asocia con los edificios, el estudio se realizó en el campus de la Universidad de Florida, Gainesville, FL, con los consumos por hora de energía eléctrica así como de contar con información termo - física y de espacio, para escenarios (cálido, medio y frío) de cambio climático a 40 años, estos escenarios de clima fueron desarrollados por el programa Norteamericano de cambio climático o por sus siglas en inglés “NARCCAP” de Gainesville, FL, dicho estudio utilizó técnicas de clustering, k-mean y regresión polinomial, se usó aprendizaje de máquina basado en redes Neuronales.

El aprendizaje de máquina también fue probado por (Kim et al., 2020) en el Campus Universitario de Pensilvania en Estados Unidos, donde se probó tanto la regresión lineal como el aprendizaje de máquina para investigar el impacto de las variables de ocupación, y climáticas como temperatura, humedad relativa, radiación solar y velocidad del viento sobre el consumo en días laborables y no laborables, destacando el aprendizaje de máquina por sus exactitud en los resultados, en comparación con el modelo de regresión.

En el estudio conducido por (Abdulrahman et al., 2019) en la predicción del consumo eléctrico en un barrio residencia de Rikkos Jos-City, Nigeria, los datos se obtuvieron mediante el uso de medidores inteligentes de energía eléctrica y además se usaron variables climáticas, usando modelos de aprendizaje de máquina se demostró una vez más valores muy bajos de error al revisar los resultados.

Los investigadores (Yuan et al., 2018), realizaron estudios sobre la predicción del consumo eléctrico usando aprendizaje de máquina en tres campus universitarios de Japón, se usaron las siguientes variables Día de la semana, hora del día, temperatura por hora, humedad relativa por hora, irradiación solar por hora, y el consumo de energía de la hora previa, logrando un nivel de presión inferior al 5% al comparar la predicción con los datos de prueba.

Como se puede apreciar en los diferentes estudios citados, el uso de aprendizaje automático para la predicción del consumo eléctrico está

ampliamente extendido en los campus universitarios, así como para predecir el consumo en zonas residenciales, el fin de todos ellos es el de mejorar la administración de recursos, una mejor planificación de infraestructura y estar preparados ante eventuales variaciones del clima,

2.2 **Modelos de aprendizaje de máquina**

El aprendizaje de máquina consiste en extraer información de un vasto número de datos, es el campo en el que interactúan las ciencias computacionales y la estadística para desarrollar modelos predictivos (Müller & Guido, 2015).

Existe dos tipos de modelos para el aprendizaje de máquina, los modelos Supervisados y no supervisados; en los modelos supervisados se conoce la salida del sistema dadas las variables de entrada, es así como dichos modelos aprenden las secuencias haciendo clasificación o predicciones, reconociendo patrones sobre salidas conocidas, entre los modelos generales en esta clasificación por citar algunos algoritmos tenemos:

- Modelos lineales
- K-vecindarios cercanos
- Arboles de decisión
- *Máquinas de soporte vectorial
- *Redes neuronales o Deep Learning.

Los modelos no supervisados tratan de agrupar la data descubriendo como agruparla, es decir que tan fuerte o débil son las características de los datos para poder ser clasificada en uno u otro grupo, dentro de esta clasificación de modelos tenemos por citar a algunos a:

- Algoritmos de agrupación o clustering en inglés como el K-means

- Reducción dimensional como Análisis de componente principal

Cabe mencionar que los modelos de soporte vectorial y Deep learning también pueden ser usados como modelos no supervisados.

Como se mencionó, dentro de la clasificación de modelos supervisados tenemos el aprendizaje profundo o como se conoce en inglés “deep learning” que es una técnica mediante la cual se trata de emular parcialmente el comportamiento biológico de cientos y hasta miles de neuronas biológicas agrupadas e interactuando en diferentes capas, donde cada unión entre ellas es un recuerdo o aprendizaje.

Como se mencionó en el párrafo anterior el aprendizaje profundo se trata de emular el comportamiento de las neuronas biológicas, es así como se presenta el modelo básico de una neurona en el contexto matemático y como la unión de cientos o de miles de estos modelos interactuando entre si forman redes neuronales artificiales.

El modelo más básico de neurona computacional se denomina perceptrón, cuya representación se muestra en la **Figura 2-1**:

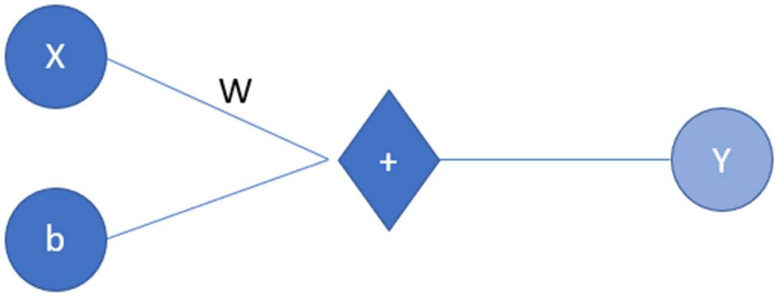

Figura 2-1 El perceptrón

Fuente: El Autor

El perceptrón está formado por entradas (X, b), enlaces o sinapsis o también llamados pesos (W) y una salida Y. El modelo que describe el perceptrón es:

$$Y = WX + b \qquad (2.1)$$

Es decir, un perceptrón en su forma más básica proporciona a la máquina la capacidad de representar una recta, de variable X, pendiente W y el punto donde intercepta al eje de las Y se denomina sesgo y se representa por b.

De la misma forma cuando el modelo recibe múltiples entradas es decir $x1, x2, \dots xn, es\ decir\ nuestra\ entrada\ es\ n-dimensional$ la forma general de la ecuación no cambia, esta pasa a representarse en la forma siguiente:

$$\hat{y} = w_1 x_1 + w_2 x_2 w_n x_n + b \qquad (2.2)$$

Donde el símbolo sombrero sobre la y denota "el valor esperado".

Si expresamos todas las características en un vector $\boldsymbol{x} \in \mathbb{R}^n$ donde $\mathbb{R}^n$ es el conjunto de n dimensiones de un vector de números reales, se puede expresar (2.2) de forma compacta usando el producto punto a continuación (2.3).

$$\hat{y} = w^T x + b \qquad (2.3)$$

Donde $\boldsymbol{x}$ corresponde a las características de una observación, debido a la gran cantidad de observaciones dentro de un conjunto de datos o dataset, con n observaciones en forma de matriz, quedaría expresada de la siguiente forma $\boldsymbol{X} \in \mathbb{R}^{n*m}$, donde $\boldsymbol{X}$ es una matriz de números reales de $n\ filas\ y\ m\ columnas$, con lo que el vector $\hat{\boldsymbol{y}}$ también puede expresarse de la forma (2.4):

$$\hat{y} = Xw + b \qquad (2.4)$$

Sin embargo, para poder encontrar el valor estimado $\hat{\boldsymbol{y}}$ es necesario conocer como las redes neuronales ajustan la matriz $\boldsymbol{w}$ y los valores de b (también conocidos como sesgo o Bias en inglés), por lo que definiremos la función de perdida l.

La función de pérdida o de error l cuantifica la distancia entre el valor real u objetivo y y el valor estimado $\hat{y}$, comúnmente el valor de l es un valor positivo, donde valores muy cercanos a 0 se prefieren indicando una aproximación casi perfecta al valor objetivo o y.

La función más usada para el cálculo del error o función de perdida en los problemas de regresión es el error cuadrático, donde nuestra predicción en la observación i es $\hat{y}^{(i)}$ y el valor que se espera es y^i i definida por la ecuación (2.5)

$$l^{(i)}(w, b) = \frac{1}{2}(\hat{y}^{(i)} - y^{(i)})^2 \qquad (2.5)$$

Para tener una idea de la ecuación ecuación (2.5) de forma gráfica para la observación i- esima, donde las líneas azules representan el error:

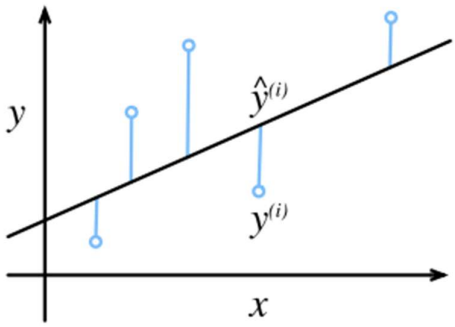

Figura 2-2 Valor estimado y real

Fuente: El autor

La expresión general que describe el error o función de pérdida para un conjunto de observaciones se describe en la ecuación (2.6).

$$L(w,b) = \frac{1}{2}\sum_{i=1}^{n}\frac{1}{2}(w^T x^{(i)} + b - y^{(i)})^2 \qquad (2.6)$$

Es así como al mencionar la expresión “Entrenar un modelo” realmente se hace referencia a encontrar los pesos o parámetros ($w*$, $b*$) que minimicen la pérdida o error total L.

Para poder minimizar los valores de ($w*$, $b*$) se usa una técnica llamada Gradiente descendente, la cual se puede describir como la reducción de forma iterativa del error actualizando los parámetros ($w*$, $b*$) en la dirección en la que la función de pérdida o error desciende de forma gradual, con lo cual se debe encontrar la primera derivada de $L(w,b)$ con respecto a w y b, es así como se muestran las derivadas parciales con respecto a w y b.

$\frac{\partial L}{\partial w}$, $\frac{\partial L}{\partial b}$ para la función $L(w,b)$ (2.7)

Sin embargo, un modelo lineal como el descrito en la ecuación (2.4) asumiría para todos los problemas un grado alto de monotonía, es decir, cuando un parámetro crece los otros también y viceversa, sin embargo, en el mundo real existen modelos de mayor complejidad que no siguen una relación lineal o monótona.

El problema descrito se puede solucionar incorporando capas adicionales h o de estado oculto o también llamadas en inglés como "hidden states", y realizando una conexión de todos los elementos de la capa de Entrada X contra todos los múltiples perceptrones hasta la capa de salida O, se la conoce como red totalmente conectada. Nótese que en la figura se encuentra representada la capa intermedia h.

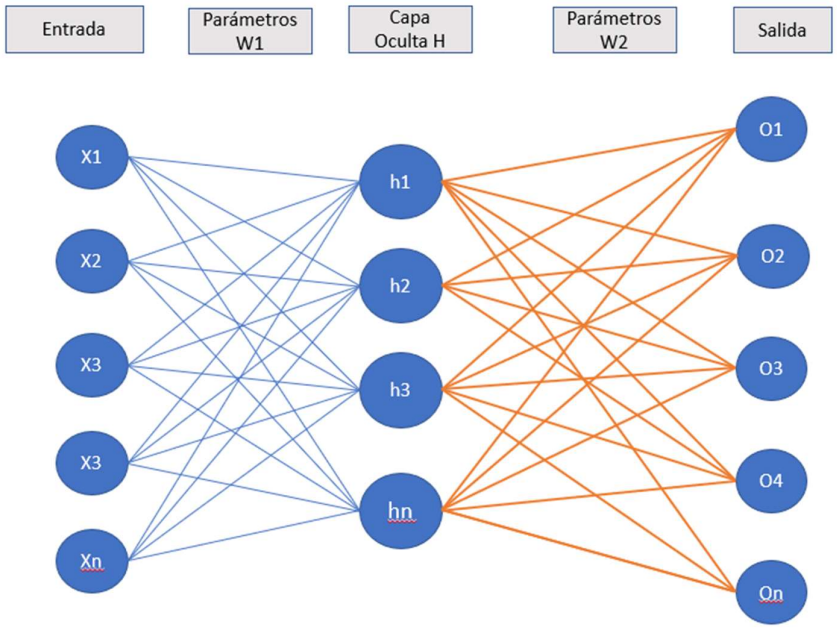

Figura 2-3 Perceptrón multicapa totalmente conectada o FNN
Fuente: El Autor

Al igual que $\boldsymbol{X} \in R^{n*d}$ para las capas ocultas podemos indicar que $\boldsymbol{H} \in \mathbb{R}^{\boldsymbol{n*h}}$, donde h es el número de las capas ocultas, también definimos los pesos $W1$ para las capas de entrada y $W2$ para las capas de salida, por tanto, las

ecuaciones de las capas ocultas y de las capas de salida se muestran a continuación en (2.8) y (2.9):

$$H = XW(1) + b(1) \quad (2.8)$$

$$O = HW(2) + b(n) \quad (2.9)$$

Si reemplazamos H en (2.9) y tomando en cuenta que $W\ y\ b$ se pueden expresar en términos de $W(1)$, $W(2)$, $b(1)$ y $b(2)$ entonces tenemos lo siguiente:

$$W = W(1)W(2)\ y\ b = b(1)W(2) + b(2) \quad (2.10)$$

$$O = \big(XW(1) + b(1)\big)W(2) + b(2) = XW(1)W(2) + b(1)W(2) + b(2) = XW + b \quad (2.11)$$

Sin embargo, como se puede apreciar en la ecuación (2.11) tenemos una ecuación lineal, por tanto, un ingrediente extra es necesario para capturar la no linealidad de los datos, es la introducción de una función de activación no lineal σ.

$$H = \sigma(XW(1) + b(1)) \quad (2.12)$$

$$O = HW(2) + b(2) \quad (2.13)$$

Con este cambio ya no es posible reducir las ecuaciones como en la ecuación (2.11) evitando su cambio a la forma lineal.

En resumen el objetivo de las redes neuronales es aprender las relaciones entre las entradas de la red y las salidas construyendo modelos lineales o no lineales (Shi et al., 2018), donde las entradas representan las características de los datos en un dataset para realizar la predicción de la salida, pudiendo convertirse en una tarea o acción al final del proceso, en el presente proyecto las entradas se definen con las variables climatológicas, año, mes, día, hora, día de la semana, mientras que la salida o predicción es el consumo eléctrico del campus.

Los modelos de aprendizaje de máquina que se propone en el estudio se definen a continuación.

2.2.1 Redes Neuronales recurrentes (RNN)

Como se mencionó previamente uno de los usos de las redes neuronales recurrentes o de Elman es la predicción de series temporales, en este caso la predicción del consumo Eléctrico del Campus ESPOL, a continuación, se introducirá los conceptos matemáticos detrás de las redes neuronales recurrentes.

Vamos a asumir que tenemos un conjunto de datos de entrada $X_t \in \mathbb{R}^{n*d}$ $al\ paso\ de\ tiempo\ t$, en otras palabras, X_t es una secuencia de n observaciones en el paso de tiempo t. Vamos además a definir $H_t \in \mathbb{R}^{n*h}$ $las\ variables\ de\ la\ capa\ oculta\ al\ paso\ de\ tiempo\ t$, donde a diferencia de las redes de Perceptrón Multicapas nos interesa guardar el estado H_{t-1} del paso de tiempo anterior, definiendo además h_t de la siguiente forma:

$$h_t = f(x_t, h_{t-1}) \qquad (2.14)$$

y además introducimos un nuevo parámetro de peso $W_{hh} \in \mathbb{R}^{h*h}$ para describir el uso de las variables ocultas del paso de tiempo anterior en el paso de tiempo actual. Específicamente, el cálculo de la variable oculta del paso de tiempo actual está dado por

la entrada del paso de tiempo actual junto con la variable oculta del paso de tiempo anterior como se describe en la ecuación (2.15).

$$H_t = \phi(\mathrm{X}_t\mathrm{W}_{xh} + \mathrm{H}_{t-1}W_{hh} + b_h) \qquad (2.15)$$

Donde ϕ es una función de activación, los parámetros de pesos representados por $\mathrm{W}_{xh} \in R^{d*h}$ a la entrada, $\mathrm{W}_{hh} \in R^{h*h}$, el sesgo o bias $b_h \in \mathbb{R}^{1*h}$ y h representa el número de unidades ocultas o "hidden", H_{t-1} estas son las variables de estado ocultas que almacenan información del histórico de la secuencia hasta el paso de tiempo presente. Las capas ocultas usan la información del paso de tiempo previo en el paso de tiempo presente, el computo se vuelve recurrente, de aquí el nombre de redes neuronales recurrentes.

Para la ecuación (2.15), la salida de esta red la definimos en la ecuación (2.16):

$$\mathbf{O}_t = \mathbf{H}_t\mathbf{W}_{hq} + \mathbf{b}_q \qquad (2.16)$$

Donde, $\mathrm{W}_{hq} \in \mathbb{R}^{h*q}$, el sesgo o bias $b_q \in \mathbb{R}^{1*q}$ son los parámetros y el bias de la capa de salida.

La **Figura 2-4** muestra la lógica de una RNN para tres pasos de tiempo, a cualquier paso de tiempo t , el computo del estado oculto puede ser tratado de la siguiente forma:

i. Concatena la entrada $\mathbf{X}_t$en el paso de tiempo presente t y el estado oculto $\mathbf{H}_{t-1}$ del paso de tiempo previo $t-1$.
ii. Alimentado esta concatenación resultante a una capa con una función de activación ϕ, la salida de esta capa resulta en la capa oculta $\mathbf{H}_t$ en el paso de tiempo presente t, a este los parámetros del modelo son la concatenación de W_{xh} y W_{hh} además del bias b_h, parámetros de la ecuación 2.15.

iii. La capa oculta al tiempo t, $\mathbf{H}_t$ participará en el cómputo del estado oculto $\mathbf{H}_{t+1}$ del siguiente paso de tiempo $t+1$, además $\mathbf{H}_t$también participará del computo de la capa totalmente conectada de salida dando como resultado $\mathbf{O}_t$ al paso de tiempo t.

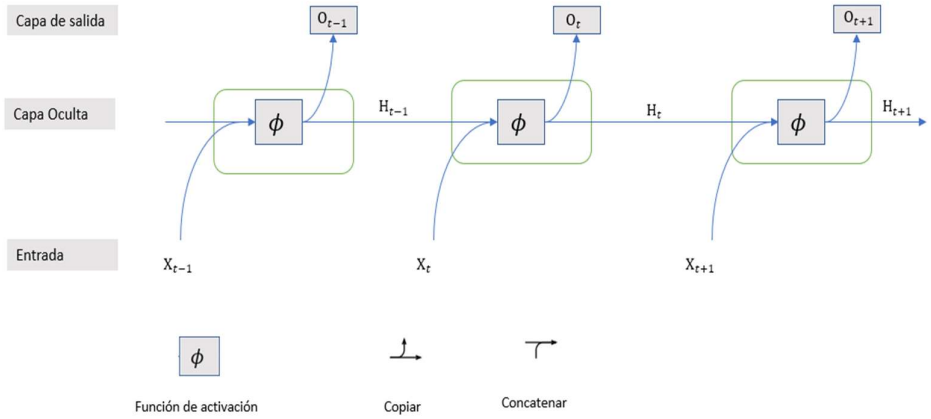

Figura 2-4 Red Neuronal RNN
Fuente: El Autor

2.2.2 Redes de Memoria de corto y largo plazo (LSTM)

El reto de mantener la información a largo plazo y la omisión de información en el corto plazo en modelos de variables latentes ha permanecido por mucho tiempo. Uno de los estudios para abordar este problema es la memoria de corto y largo plazo o LSTM (Hochreiter & Schmidhuber, 1997), como dato interesante las LSTM comparten propiedades de con las redes GRU que se describirán en la siguiente sección y son algo más complejas en su diseño.

Las redes LSTM introducen en su concepto una celda de memoria, que sigue una lógica similar al de los estados ocultos, en la cual se guarda información adicional, es así que para controlar esta celda de memoria se necesita algunas compuertas adicionales. Una de estas compuertas se requiere para leer las entradas, refiriéndonos a ella como "compuerta de salida", una segunda compuerta es necesaria para decidir cuándo leer

datos de la celda, refiriéndonos a esta como “compuerta de entrada”, y por último se necesita un mecanismo para reiniciar el contenido de la celda, al cual se denomina “compuerta de olvido”, es decir que el objetivo es proveer al modelo la habilidad para “recordar” y cuando “olvidar o descartar” información no útil.

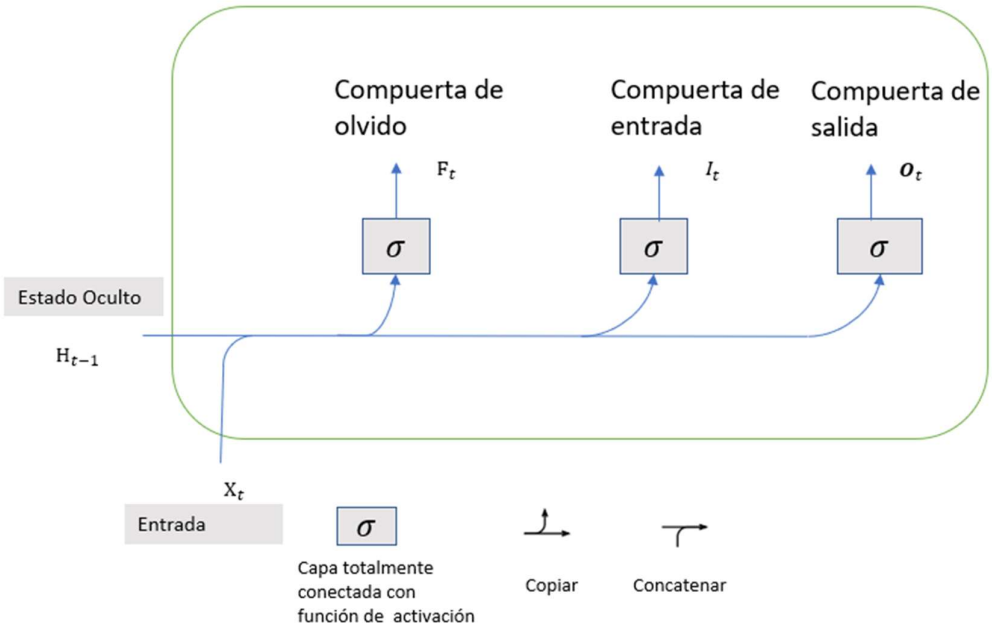

Figura 2-5 Compuertas de LSTM
Fuente: El Autor

La data ingresa a la red LSTM son los datos de entrada en el paso de tiempo actual y el estado oculto del paso de tiempo anterior como se muestra en la Figura 2-5, estas entradas son procesadas por capas totalmente conectadas con una función de activación sigmoide para procesar los valores en las compuertas de entrada, olvido y salida, lo que da como resultado valores entre (0,1).

Se define además la función sigmoide continuación:

Una función sigmoide tiene la particularidad de estar acotada entre 0 y 1 para el eje Y y para el eje de las X entre -∞ y ∞, la función se expresa de la siguiente forma en la ecuación 2.17:

$$y = \frac{1}{1+e^{-x}} \qquad (2.17)$$

La Figura 2-6 muestra la gráfica de la función sigmoide.

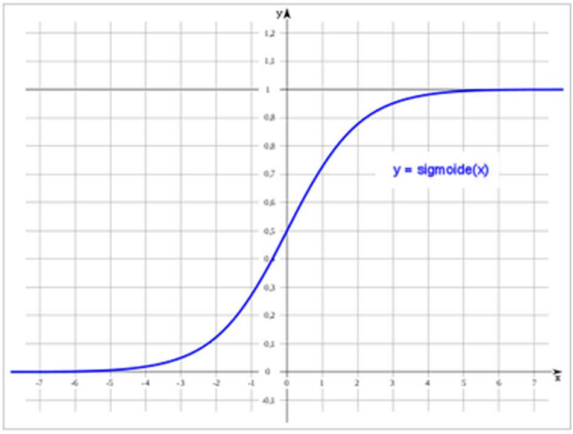

Figura 2-6 Función sigmoide
Fuente: El Autor

Definimos además las $\mathbf{X}_t \in R^{n*d}$ con número de observaciones n y número de entradas d , y el estado oculto en el paso de tiempo anterior $\mathrm{H}_{t-1} \in \mathbb{R}^{n*h}$. Las compuertas al paso de tiempo t se definen de la siguiente forma: la puerta de entrada $\mathbf{I}_t \in R^{n*h}$, la compuerta de olvido $\mathbf{F}_t \in R^{n*h}$ y la compuerta de salida $\mathbf{O}_t \in R^{n*h}$, expresaremos además sus ecuaciones (2.18), (2.19) y (2.20):

$$\mathbf{I_t} = \sigma(\mathbf{X_t W_{xi}} + \mathbf{H_{t-1}} \boldsymbol{W_{hi}} + \boldsymbol{b_i}) \qquad (2.18)$$

$$\mathbf{F_t} = \sigma(\mathbf{X_t W_{xf}} + \mathbf{H_{t-1} W_{hf}} + \boldsymbol{b_f}) \qquad (2.19)$$

$$\mathbf{O}_t = \sigma(\mathbf{X_t W_{xo}} + \mathbf{H_{t-1} W_{ho}} + \boldsymbol{b_o}) \qquad (2.20)$$

Donde $\mathbf{W_{xi}}, \mathbf{W_{xf}}, \mathbf{W_{xo}} \in R^{d*h}$ y $\boldsymbol{W_{hi}}, \boldsymbol{W_{hf}}$ **y** $\boldsymbol{W_{ho}} \in R^{h*h}$ son los parámetros de pesos y $\boldsymbol{b_i}, \boldsymbol{b_f}, \boldsymbol{b_o} \in R^{1*h}$ son los sesgos o bias.

Se define además la celda de memoria, o celda candidata de memoria $\widetilde{\mathbf{C}}_t \in R^{n*h}$, y su funcionamiento al de las otras tres compuertas $I, F\ y\ O$, con la salvedad de que su función de activación es una $\tanh()$ la cual tiene un rango de valores (-1,1), la ecuación 2.21 al paso de tiempo t.

$$\tilde{\mathbf{C}}_t = tanh(\mathbf{X_t W_{xc}} + \mathbf{H_{t-1} W_{hc}} + \boldsymbol{b_c}) \qquad (2.21)$$

Donde $\mathbf{W_{xc}} \in R^{d*h}$ y $\mathbf{W_{hc}} \in R^{h*h}$ son los parámetros de peso y $\boldsymbol{b_c} \in R^{1*h}$ es el sesgo, a continuación, la **Figura 2-7** ilustra la celda candidata.

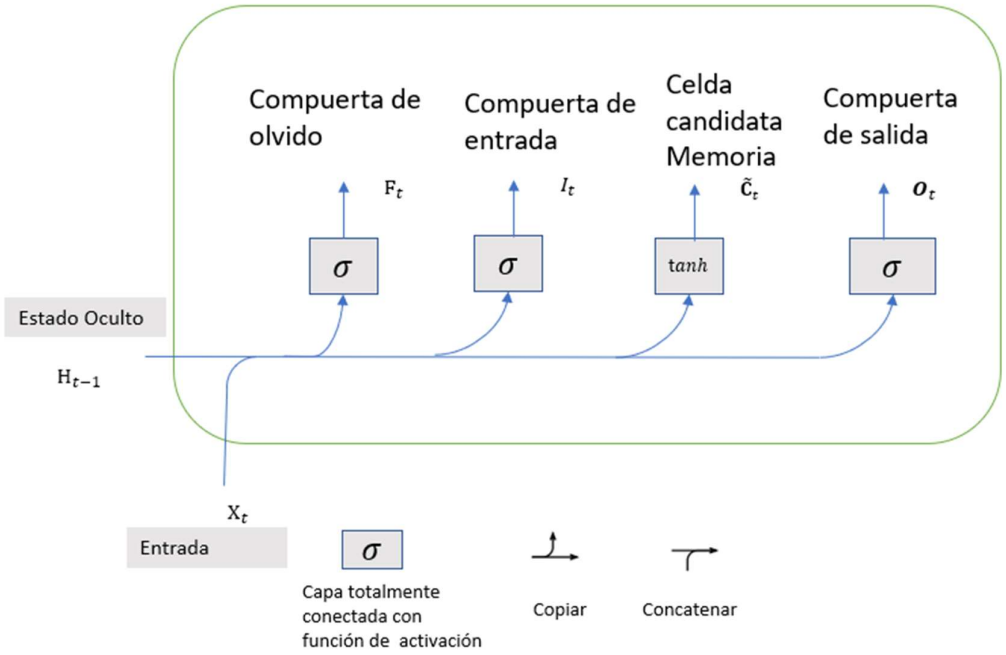

Figura 2-7 Celda candidata $\tilde{C}_t$
Fuente: El Autor

La puerta de entrada $\mathbf{I_t}$ controla cuanta información se tomará en cuenta sobre $\tilde{\mathbf{C}}_t$ y la compuerta de olvido $\mathbf{F_t}$ que cantidad de información del contenido de la celda anterior $\boldsymbol{C_{t-1}} \in R^{n*h}$ se retendrá. Para esto se usa el producto uno a uno ($\odot$) como se describe en ecuación 2.22.

$$\mathbf{C_t} = \mathbf{F}_t \odot \mathbf{W}_{t-1} + \mathbf{I}_t \odot \tilde{\mathbf{C}}_t \qquad (2.22)$$

Si la puerta de olvido es cercana a 1, y en la puerta de entrada tenemos valores cercanos a 0, el pasado de la celda de memoria $\boldsymbol{C_{t-1}}$ será guardado y traspasado al paso de tiempo actual, este diseño ayuda a evitar el desvanecimiento del gradiente y a mejorar la captura de información útil a lo largo de las secuencias, como se indica en **Figura 2-8** a continuación.

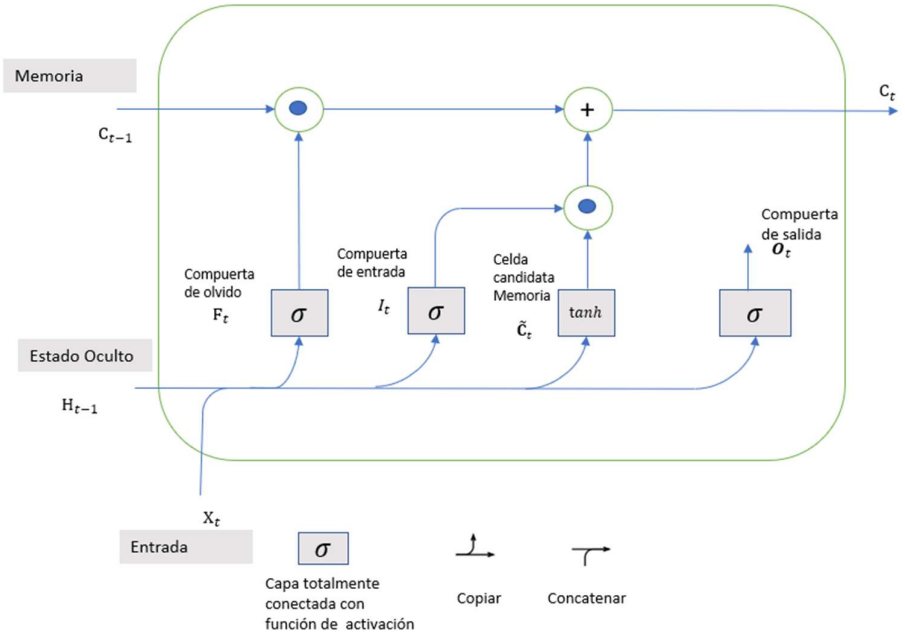

Figura 2-8 Celda de memoria en LSTM
Fuente: El Autor

Lo último es definir como se procesa el estado oculto $\mathbf{H_t} \in R^{n*h}$, aquí es donde la puerta de salida juega un rol, en las LSTM la $\mathbf{H_t}$ se encuentra entre valores de (-1, 1) esto se logra introduciendo una vez más la función $\tanh()$, la ecuación (2.23) representa lo indicado.

$$\mathbf{H_t} = \mathbf{O}_t \odot \tanh(\mathbf{C}_t) \qquad (2.23)$$

Lo mencionado anteriormente indica que cualquier entrada cercana a 1 en la salida pasará toda la información en memoria al predictor, por otra parte, si el valor en la puerta de salida es cercana a 0 se retendrá toda la información dentro de la celda de memoria y no se realizará otro proceso, este proceso se ilustra en la **Figura 2-9**.

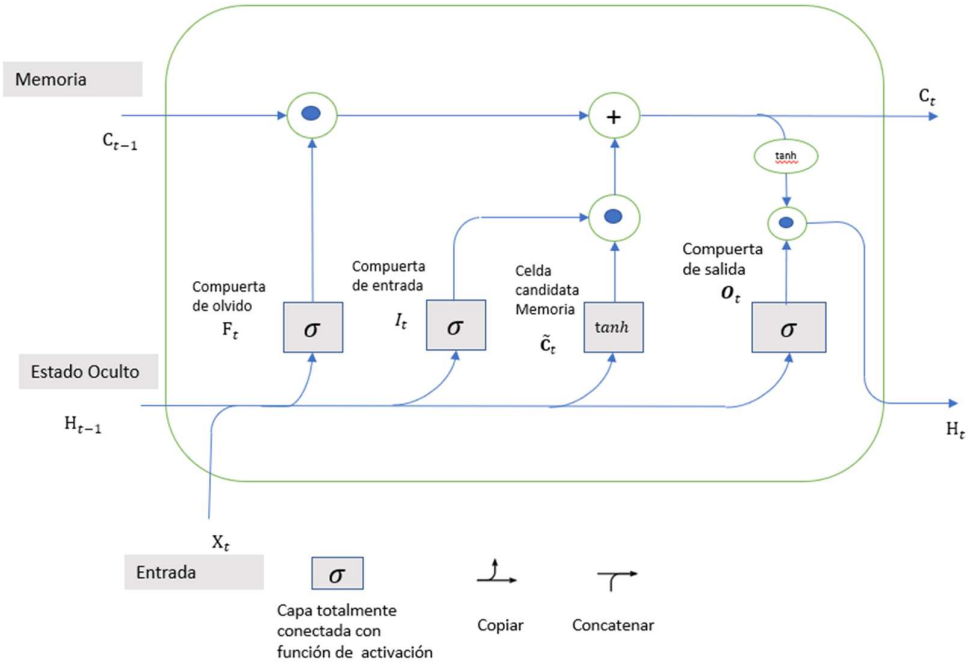

Figura 2-9 Red LSTM
Fuente: El Autor

2.2.3 Unidad de compuertas recurrentes (GRU)

Uno de los problemas principales de las redes recurrentes es que el gradiente se desvanezca o crezca desmesuradamente esto último se conoce como explosión del gradiente, esto debido a los largos cálculos matriciales que se realizan, en la práctica lo siguiente puede suceder:

i. Podemos encontrar la situación donde una observación temprana es altamente significativa para predecir las observaciones futuras, para el caso dicha observación temprana tiene información de verificación y este proceso de verificación se requiere para revisar la salida al final de la secuencia, para este escenario la primera observación es vital en el proceso, para lo cual desearíamos tener algún mecanismo para almacenar esta importante información temprana, es decir una celda de memoria. Sin este mecanismo, tendríamos que asignar un

gradiente muy largo a esta observación, donde esto afecta a todas las subsiguientes observaciones.

ii. Otra situación que podríamos encontrar es con información no relevante, por ejemplo, información de posición geo referencial en un análisis de sentimientos, con lo cual se requiere de algún mecanismo que evite tal información ya que no es relevante para el proceso en el estado latente.

iii. Además, podríamos encontrar la situación donde hay un punto de quiebre entre las secuencias, por tanto, una transición por ejemplo entre capítulos de una serie o un nuevo comportamiento en una serie temporal, dado este escenario sería bueno tener una forma de reiniciar nuestra representación de interna del estado.

La distinción principal entre las RNN y las GRU, es que las GRU permiten una activación del estado oculto, esto quiere decir que se cuenta con mecanismos dedicados para cuando la capa oculta necesita ser actualizada o cuando necesita ser reiniciada.

Si la primera observación es de gran importancia, aprenderemos a no actualizar el estado oculto después de la primera observación. Asimismo, aprenderemos a omitir observaciones temporales irrelevantes. Por último, aprenderemos a reiniciar el estado latente siempre que sea necesario, a continuación, las puertas de reinició y actualización:

Diseñamos un vector (0,1) tal que podamos realizar operaciones, para el caso una puerta de reinicio nos permitirá controlar que tanto de la información previa nos será de utilidad recordar, igualmente una compuerta de actualización nos permitirá controlar que tanto del nuevo estado es una copia del estado anterior.

En la **Figura 2-10** se ilustra las entradas de ambas compuertas, reinicio y actualización en una GRU, dado una entrada en el paso de tiempo actual y una capa oculta en el paso de tiempo previo. Las salidas de ambas compuertas son dadas por capas totalmente conectadas con una función de activación sigmoide.

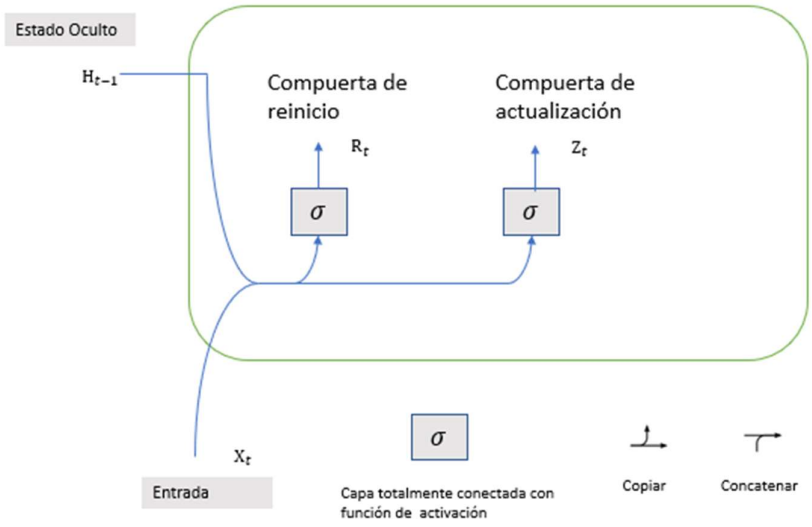

Figura 2-10 Compuertas de reinicio y de actualización en una GRU
Fuente: El Autor

Matemáticamente para un paso de tiempo t suponemos una entrada de datos $\mathbf{X}_t \in \mathbb{R}^{n*d}$, donde n es el número de observaciones y d *es el número de entradas o características,* y el estado oculto en el paso de tiempo previo es $\mathrm{H}_{t-1} \in \mathbb{R}^{n*h}$ donde h es el número de unidades ocultas, entonces la compuerta $\mathrm{R}_t \in \mathbb{R}^{n*h}$ y la compuerta $\boldsymbol{Z_t} \in \mathbb{R}^{n*h}$ se pueden describir de la siguiente forma matemática en (2.24) y (2.25):

$$\mathrm{R}_t = \sigma(\mathrm{X}_t \mathrm{W}_{xr} + \mathrm{H}_{t-1} W_{hr} + b_r) \qquad (2.24)$$

$$\mathrm{Z}_t = \sigma(\mathrm{X}_t \mathrm{W}_{xz} + \mathrm{H}_{t-1} W_{hz} + b_z) \qquad (2.25)$$

Donde $\mathrm{W}_{xr}, \mathrm{W}_{xz} \in \mathbb{R}^{d*h}$ y $\mathrm{W}_{hr}, \mathrm{W}_{hz} \in \mathbb{R}^{h*h}$ son parámetros de pesos y b_r, b_z son los sesgos. La función sigmoide es usada para trasformar los datos de entrada en el intervalo (0,1).

Lo siguiente es integrar la compuerta de reinicio $\mathbf{R}_t$ con el estado latente regular y mecanismo de actualización descrito por la ecuación (2.25), esto provoca el llamado

estado candidato oculto o $\widetilde{\mathrm{H}}_{\mathrm{t}} \in \mathbb{R}^{n*h}$ al paso de tiempo t como se muestra en la ecuación (2.26).

$$\widetilde{\mathbf{H}}_t = tanh(\mathbf{X}_t\mathbf{W}_{xh} + (\mathbf{R}_t \odot \mathbf{H}_{t-1})\mathbf{W}_{hh} + \boldsymbol{b}_h) \quad (2.26)$$

Donde $W_{xh} \in \mathbb{R}^{d*h}$ y $W_{hh} \in \mathbb{R}^{h*h}$ son parámetros de pesos, $\boldsymbol{b}_h \in \mathbb{R}^{1*h}$ es el bias y el símbolo $\odot$ es el producto uno a uno de dos matrices, aquí representamos la no linealidad en la forma de la función de activación Tangente Hiperbólica, $tanh()$ que asegura que los valores de salida del estado candidato $\widetilde{\mathbf{H}}_t$ se encuentren en el intervalo (-1,1).

El resultado como su nombre lo indica es un candidato (o posible como sinónimo) debido a que aún se necesita incorporar la puerta de actualización, comparado con la ecuación (2.14) la influencia de los estados previos puede ser reducida con el producto uno a uno de $\mathbf{R}_t \; y \; \mathbf{H}_{t-1}$ como se indica en la ecuación (2.26). Cualquiera entrada en la compuerta de reinicio $\mathbf{R}_t$ cercana a 1 hará que actúe como una red RNN, mientras que si la entrada $\mathbf{R}_t$el valor de entrada es cercano a 0 el candidato de estado oculto es el resultado de un Perceptrón Multicapa con $\mathbf{X}_t$como entrada. Cualquier estado oculto preexistente será por lo tanto reiniciado a sus valores por defecto.

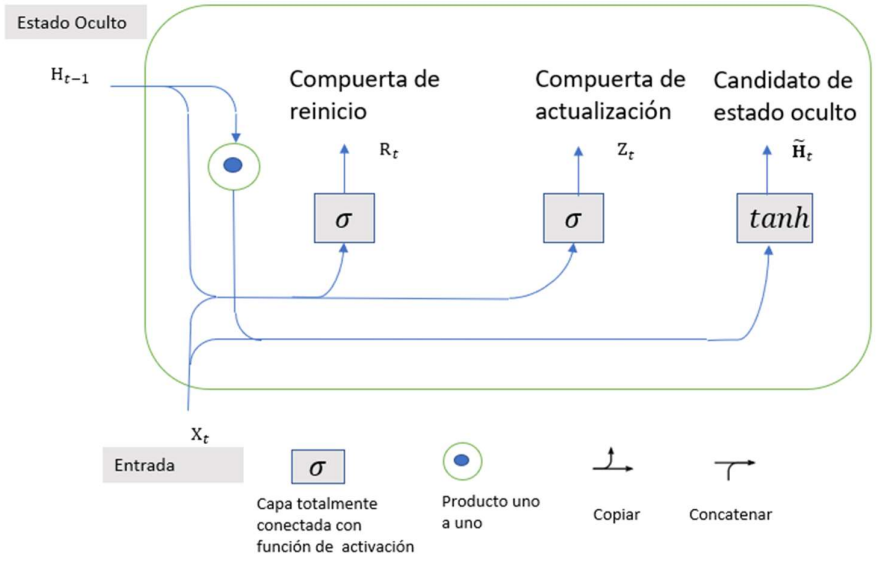

Figura 2-11 Compuerta Candidato en una GRU

A continuación, se describe el efecto de la compuerta de actualización de la compuerta $\boldsymbol{Z_t}$, con esto se determina la extensión en la cual el nuevo estado oculto $\mathbf{H}_t \in \mathbb{R}^{n*h}$ forma parte del estado antiguo $\mathbf{H}_{t-1}$ y en qué medida el estado candidato $\tilde{\mathbf{H}}_t$ es usado. La compuerta de actualización $\boldsymbol{Z_t}$ puede ser usada para ese propósito, tomando el producto uno a uno entre $\mathbf{H}_{t-1}$ y $\tilde{\mathbf{H}}_t$, esto contiene la ecuación de actualización de una red GRU y se muestra en (2.27).

$$\mathbf{H}_t = \mathbf{Z}_t \odot \mathbf{H}_{t-1} + (1 - \mathbf{Z}_t) \odot \tilde{\mathbf{H}}_t \qquad (2.27)$$

Siempre que la compuerta de actualización $\mathbf{Z}_t$ sea cercana a 1, se retendrá el estado antiguo, en este caso la información de $\mathbf{X}_t$ es ignorada, efectivamente evitando el paso de tiempo t en la cadena de dependencia. En contraste si $\mathbf{Z}_t$ sea cercana a 0, el nuevo estado latente $\mathbf{H}_t$se aproxima al estado candidato latente $\tilde{\mathbf{H}}_t$. Este diseño nos permite hacer frente al desvanecimiento del gradiente en las redes RNN y mejorar la captura de secuencias dependientes con distancias largas de pasos de tiempo. Por otra parte, si $\mathbf{Z}_t$ es cercano a 1 para todos los pasos de tiempo de una secuencia completa, el estado

oculto al inicio se mantendrá a lo largo de la secuencia hasta el final, independientemente de la longitud de la secuencia, la figura 7 muestra el proceso descrito.

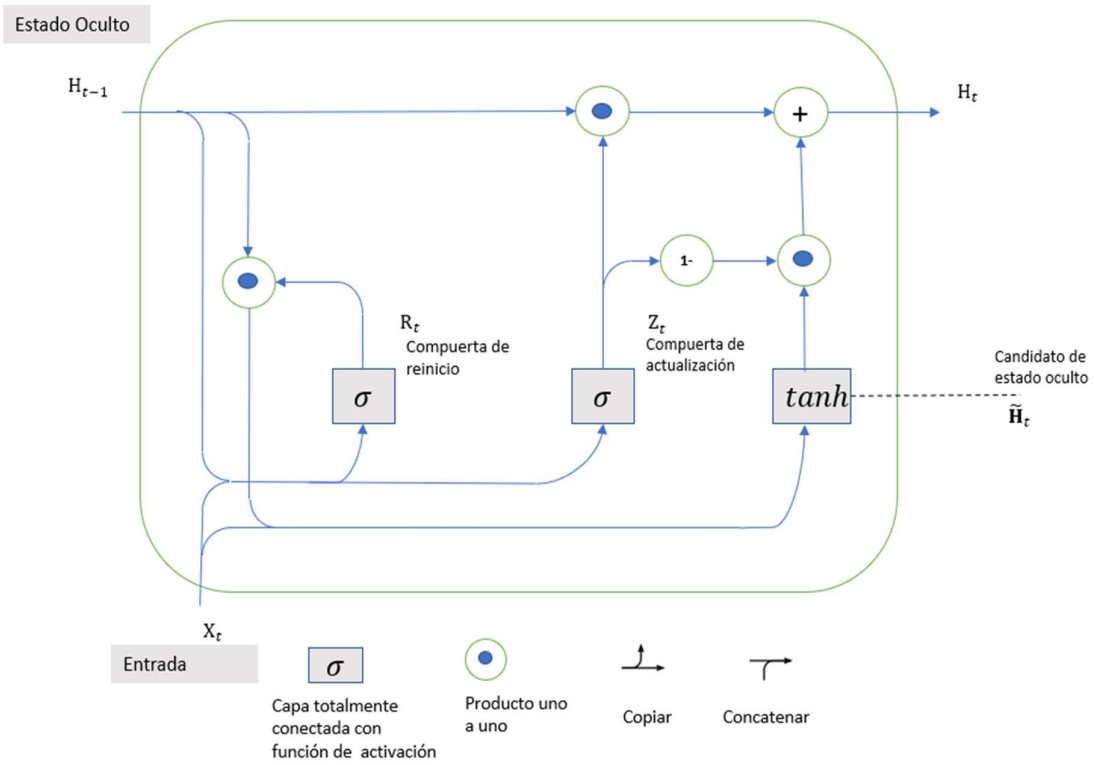

Figura 2-12 Red GRU
Fuente: El Autor

2.3 **Estudios sobre predicción de la demanda Eléctrica**

Existen varios estudios dedicados sobre la predicción de la demanda eléctrica como se mencionó en el apartado 2.1, a continuación citaremos los estudios realizados en campus universitarios usando variables climáticas, en el estudio realizado por (Taylor, 2013), analiza el uso de redes neuronales con un conjunto de datos con información de 1 año desde Abril 2011 hasta Marzo 31 de 2012, las variables usadas como datos de entrada fueron Irradiación solar (W/m2), temperatura (ºC), humedad relativa (%), promedio de la velocidad del viento (m/s), Presión atmosférica (mBar), precipitación promedio (mm)

además de los datos del consumo eléctrico y hora y en forma de rezagos, cuando mencionamos rezagos quiere decir que la data es ingresada al modelo de forma desfasada, el presente trabajo se desarrolla con el uso de variables climáticas como de la información de consumo eléctrico, ambas en forma de rezagos desfasados a 24 horas brindando al modelo mayor información al momento de encontrar el comportamiento de la serie temporal, además del uso de variables climáticas como son las de temperatura, humedad relativa, lluvia(mm) y de temporalidad como día, mes y día de la semana.

Similarmente el estudio (Amber et al., 2015), utiliza un dataset compuesto por el consumo (W h/m2) por área como variable a predecir y las variables independientes temperatura, radiación solar, humedad relativa y velocidad del viento, además también codifican una variable Wdi (Dia semana de trabajo), este modelo se diferencia del resto de ejemplos ya que utiliza modelos de programación genética y de regresión lineal en lugar de redes neuronales.

El estudio realizado al campus de la Universidad Técnica de Cartagena, España en (Del Carmen Ruiz-Abellón et al., 2018) trabajó con árboles de regresión. Además de la variable temperatura los autores introdujeron variables relacionadas a diferenciar días de la semana y feriados. También, similar a otros estudios discutidos anteriormente se introduce data retrasada del consumo eléctrico, hasta 7 días para asistir la tarea de predicción. Una diferencia que destacar con nuestra propuesta es que utilizamos data retrasada de la carga hasta en 24 horas como se mencionó previamente.

Un estudio que llama la atención por el horizonte de predicción que se muestra es el de (Fathi & Srinivasan, 2019), ya que proponen pronósticos para 2041, 2057 y 2063, basan sus modelos evaluando los algoritmos K means, Regresión Polinomial y una red LSTM similar a la de nuestro estudio, su dataset incluye variables como tipo y uso de los edificios, variables climáticas de temperatura, humedad relativa, irradiación solar. Se introdujo como variable la hora del día, también se consideró como variable la desviación absoluta de la temperatura promedio y la línea base de los días de enfriamiento, por sus siglas en inglés (Tcdd, 65ºF), la propuesta del estudio

fue comparar estos tres años de predicción (2041, 2057 y 2063) respecto a 2018, el resultado más cercano indica que es comparable al año 2041, el trabajo que nos ocupa propone un horizonte más corto, es decir 3 meses con la data de prueba.

El estudio realizado por (Kong et al., 2019), donde al igual que en nuestro estudio se realiza el uso de redes neuronales RNN y LSTM, sin embargo, se usa para edificaciones familiares, donde se evaluaron 69 clientes con data completa de tres meses de Junio 2013 a Agosto 2013, se utiliza una serie univariante, es decir el consumo de energía únicamente. De modo diferente nuestra propuesta y otros trabajos discutidos anteriormente utiliza variables externas para la predicción, sin embargo, hace valido el uso de las redes neuronales propuestas en el presente estudio, así como el uso de la variable a predecir como una entrada del modelo.

Cabe mencionar que en los estudios antes citados se usa la resolución de 1 hora para los datos de Consumo eléctrico, la resolución en la que la ESPOL entregó los datos para este trabajo fue de 15 min para la variable de Consumo eléctrico y de 10 min para las variables climáticas, acorde a lo descrito en el apartado 1.7 Dataset, dado que el consumo Eléctrico se mide y se paga por hora de consumo eléctrico, se lleva ambos datasets a esta resolución, realizando la suma de las 4 observaciones que se tiene por hora de consumo eléctrico y tomando el valor promedio para las 6 observaciones para cada hora de las variables climáticas.

2.4 **Herramientas de desarrollo y visualización**

Existen diversas herramientas en el mercado para poder implementar modelos de aprendizaje profundo, por citar las más importantes Pytorch, Tensoflow/Keras, las cuales se desarrollaron pensando en facilitar el modelado del aprendizaje de máquina, mientras que las herramientas como Matlab, R se enfocan más en funciones matemáticas y estadísticas, en el

desarrollo se propone el uso de Pytorch como plataforma de implementación de los modelos de redes neuronales.

En la literatura se encuentran diversas opciones y requerimientos de hardware y software para el desarrollo de soluciones de predicción y visualización de consumo de energía eléctrica. Por ejemplo en el estudio (Kong et al., 2019), los autores utilizan una PC con procesador Intel de 3.4Ghz, 8GB de memoria, junto con Keras como plataforma de desarrollo de Deep Learning.

Por otro lado en (Taylor, 2013), el autor utilizan una PC con procesador Intel de 2.3GHz y Matlab como plataforma de desarrollo para la red Neuronal. Estos ejemplos muestran que el desarrollo de trabajos, así como el presente proyecto es posible realizarlo en computadores personales.

Existe además la posibilidad de desarrollar soluciones basadas en Aprendizaje profundo en la nube, por ejemplo, en Google Colab y AWS Amazon y con acceso a mayor poder de cómputo y almacenamiento en forma de subscripción pagada.

En el proyecto utilizaremos las siguientes herramientas:

- Python
- Pandas
- R y R Estudio
- Pythorch
- CUDA
- Conda
- Numpy
- Plotly
- Dash
- Laptop

Dichas herramientas cumplen los siguientes roles en el desarrollo del proyecto:

- Pyhton, es por excelencia el lenguaje para ciencias de datos, el mismo provee la flexibilidad para manejo de librerías que trabajan con Grandes volúmenes de datos como Pandas, permite el análisis estadístico, manejo de series temporales, trabajo con matrices n dimensionales por medio de Numpy.
- Pandas Es una Librería de Python que se orienta al manejo y análisis de estructuras de datos.
- R y R Estudio, se usa para realizar el análisis de correlación y de envejecimiento de data.
- Pythorch es un framework de clase mundial para la configuración de redes neuronales y mayormente usado en las investigaciones científicas por su flexibilidad al momento de configurar parámetros y la desmitificación de que los modelos de aprendizaje son cajas negras.
- CUDA, se usa en el presente trabajo para aprovechar el poder de cómputo paralelo que provee la tarjeta de video de la laptop sobre la cual se desarrolló el estudio.
- Miniconda es un entorno de desarrollo completo donde se puede albergar Python en su versión 3.8 como lenguaje de programación, la versión usada es libre de pagos, es flexible, y permite correr Jupyter Notebook para la programación con Python, esto hace que se pueda ejecutar programación por lotes.
- Pandas, se realiza el contacto con el dataset, limpieza, estadística descriptiva.

- Numpy, Librería de análisis matemático y manejo de grandes volúmenes de datos.

Para el desarrollo de la visualización del proyecto se usó lo siguiente:

- Plotly, es una herramienta de visualización interplataforma, que de forma temprana en el proyecto se usa para realizar un análisis exploratorio de los datos y posteriormente para la visualización de los datos en la herramienta de visualización.
- Dash, permite generar páginas web usando código Python

2.5 **Métricas para modelos aprendizaje de máquina**

Las principales métricas usadas en los estudios discutidos en apartado 2.3 para determinar el rendimiento de un modelo de predicción de consumo eléctrico son las siguientes:

i. MAE o media de error absoluto es un indicador que indica que tan bien se realiza una predicción al comparar los valores reales y_i con la predicción obtenida $\hat{y}_i$, mayor el error cuanto más grande es el valor del MAE, su forma matemática en (2.28):

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|\hat{y}_i - y_i| \qquad (2.28)$$

Donde y_i es la medición real, $\hat{y}_i$ es la predicción y n es el número de observaciones.

ii. R2 o coeficiente de determinación, mide la capacidad del modelo para predecir futuros resultados, los resultados posibles se encuentran entre

0 y 1, siendo 1 una predicción perfecta, la fórmula utilizada para el cálculo es la siguiente (2.29).

$$R^2 = 1 - \sum_{i=1}^{n} \frac{(y_i - \hat{y}_i)^2}{(y_i - \bar{y})^2} \qquad (2.29)$$

Donde y_i es el valor real, $\hat{y}_i$ es el valor pronosticado, $\bar{y}$ es el valor promedio de los valores reales.

iii. MSE o error cuadrático medio, mide la diferencia que existe entre dos mediciones por Ejemplo una predicción $\hat{y}$ y el valor real y_i , la fórmula utilizada para el cálculo es la siguiente (2.30).

$$MSE = \frac{1}{n} \sum_{i=1}^{n} (\hat{y}_i - y_i)^2 \qquad (2.30)$$

CAPÍTULO 3

3 DISEÑO E IMPLEMENTACIÓN

En el Capítulo 1 se realizó la descripción de la metodología a seguir, mientras que en el presente capitulo revisaremos las acciones que se siguieron usando la metodología descrita.

En el proyecto se definieron dos procesos denominados Proceso Interno y Proceso Cliente, los cuales definimos a continuación:

- Proceso Interno. _ Orientado al Preprocesamiento de Datos y la Aplicación del Modelo de Machine Learning, tiene por objeto proporcionar la línea base de consumo, calcular la correlación entre variables y el modelo a ser usado por el Proceso cliente. El Proceso Interno no se muestra al usuario final y se destina para el mantenimiento y ajuste de la red neuronal.

- Proceso Cliente, orientado a la visualización de los datos de la línea base de consumo eléctrico, la correlación de variables, así como la visualización de los resultados de la predicción.

  Se muestra a continuación en la *Figura 3-1* los procesos descritos.

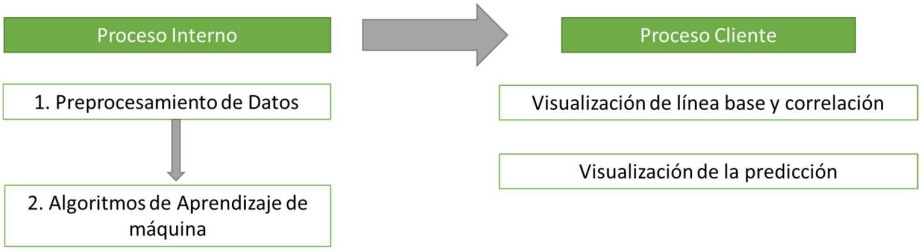

***Figura 3-1* Flujo del sistema**
Fuente: El Autor

En el subproceso Preprocesamiento de datos se enmarca la limpieza, la agrupación y la normalización además del análisis exploratorio de datos, el objetivo de este proceso es preparar la data para el entrenamiento del algoritmo de aprendizaje de máquina.

Luego el subproceso Algoritmo de aprendizaje de máquina utiliza la data preprocesada para generar el modelo que se usará en la predicción del consumo eléctrico.

El proceso Cliente recibe tanto la data de línea base de consumo, como el modelo de aprendizaje de máquina, con el fin de visualizar la data histórica de consumo eléctrico, así como su predicción destacando que el poder visualizar la información con un medio de cómputo favorece al razonamiento cognitivo del usuario (Card, 1999).

En los siguientes apartados se explica en detalle los subprocesos descritos.

3.1 Limpieza, normalización y agrupación de los datos

Como se describió en el Capítulo 1 la limpieza, normalización y agrupación de los datos es necesaria para poder ingresar datos al modelo de aprendizaje de máquina y posterior visualización.

Dentro de este Subproceso la data provista por la ESPOL se carga mediante la librería Pandas, dicha librería nos permite realizar la estadística básica, nos permitirá el preprocesamiento de los datos, así como la identificación de las variables con un nombre abreviado para una mejor identificación durante los procesos subsiguientes, el nombre abreviado de las variables se muestra en las **¡Error! No se encuentra el origen de la referencia.** y **¡Error! No se encuentra el origen de la referencia.** junto con sus descripciones.

Tabla 3.1 Características del dataset Clima

Nombre de la Variable	Nombre abreviado	Descripción
N.'	Index	Índice de la observación

'Fecha Tiempo. GMT-05:00'	Fecha Tiempo	Fecha y hora de la observación en formato mm/dd/año hh:mm:ss
'Temp. °C (LGR S/N: 10450278. SEN S/N: 10431781. LBL: AmbTemp'	TempCelcius	Temperatura en Grados Celcius cada 10 min
HR. % (LGR S/N: 10450278. SEN S/N: 10431781. LBL: AmbHR)'	HR	Humedad relativa
Lluvia. mm (LGR S/N: 10450278. SEN S/N: 10440752. LBL: Lluvia)'	Lluviamm	Precipitaciones en milímetros
'Dirección del viento. ø (LGR S/N: 10450278. SEN S/N: 10468828. LBL: Direccion del vient'	Dirdelviento	Dirección del viento
'Velocidad del viento. m/s (LGR S/N: 10450278. SEN S/N: 10470750. LBL: Velocidad del viento)'	Veldelviento_m/s	Velocidad del viento
'Velocidad de Ráfagas. m/s (LGR S/N: 10450278. SEN S/N: 10470750. LBL: Velocidad de rafagas'	Velderafagas_m/s	Velocidad de ráfagas de viento

Fuente: El Autor

Tabla 3.2 Características de Consumo Eléctrico

Nombre de la Variable	Nombre abreviado	Definición
'Unnamed	'tiempo',	Fecha y hora de la observación en formato dd/mm/aaaa, hh:mm:ss
ENERGIA ACTIVA'	'energia_activa',	Se mide en Kwh y corresponde al consumo agregado de todo el campus de la ESPOL

Fuente: El Autor

Una vez identificadas las variables con un nombre abreviado se prosigue con los siguientes pasos:

1. Se realiza la búsqueda de datos Nulos o faltantes
2. Se elimina información duplicada presente en los datasets

3. Posterior a esto se realiza el cambio de resolución de los datos, este cambio como se mencionó en la sección 2.3 es necesario debido a que los consumos se miden y facturan por hora. A continuación, se describe el proceso para dejar la resolución de los datos en forma horaria.

    a. Se suma las 4 observaciones de 15 min de consumo de energía Eléctrica en una Hora para ser compatibles con la resolución en que se factura los servicios de Energía Eléctrica.

    b. Para variables climáticas se obtiene un promedio de las mediciones por cada hora, esto con el fin de tener una representación de las 6 mediciones que se realizan durante una hora.

4. La rutina de normalización es aplicada al ingreso de los datos al modelo de Aprendizaje de Máquina.

    Esta rutina tiene por objeto normalizar los datos a un intervalo específico, para que las operaciones matriciales que se realizan en el algoritmo de aprendizaje de máquina sean ágil y no se penalice las variables con datos de magnitud menos significativa

    a. La rutina de normalización toma el valor mínimo y el valor máximo de cada variable.

    b. Mediante el método MinMax de la librería Scikit-learn se normaliza cada una de las variables o características x tomando su $i-ésimo\ valor$ denotado por x_i el valor mínimo x_{min} y x_{max} , y relacionándolos en la formula (3.1), lo cual dará como resultado valores entre (0, 1).

$$x = \frac{x_i - x_{min}}{x_{max} - x_{min}} \qquad (3.1)$$

3.2 Análisis Exploratorio y validación de datos

Dentro del proceso análisis exploratorio y validación de datos es recomendable visualizar la data con el fin de darnos una idea de sus características, y si presentan alguna estacionalidad visible.

Las gráficas que se muestran a continuación reflejan el comportamiento de las series temporales con las que trabajamos, Energía Activa y las variables climáticas de Temperatura y Humedad Relativa, dirección del viento, Velocidad del viento, velocidad de ráfaga del viento y Lluvia en mm.

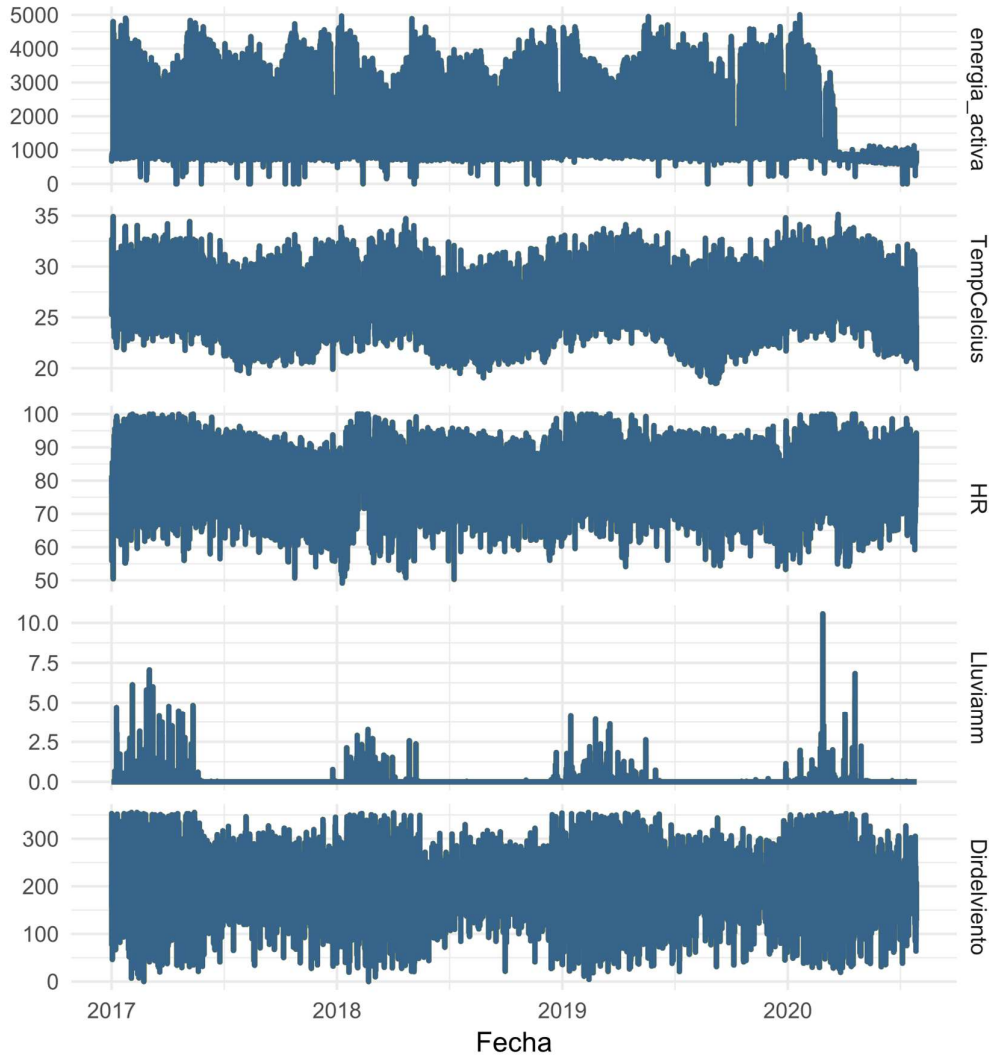

Figura 3-2 Visualización de series temporales
Fuente: El Autor

En la **¡Error! No se encuentra el origen de la referencia.** se muestra una visualización exploratoria de las series temporales, en la cual se puede apreciar lo siguiente:

- energía_activa. _ Se aprecia cierta estacionalidad y una baja en el consumo hacia mediados del 2020 producto de la pandemia, se requiere cambiar la resolución de los datos para apreciar de mejor forma los datos.
- TempCelcius. _ comportamiento sinusoidal con baja de temperatura a mediados de año y picos de temperatura hacia finales de año.
- HR. _ Humedad relativa con comportamiento sinusoidal con alta incidencia a inicio de año coincidente con la época de lluvias, las cuales descienden hacia mediados de año.
- Lluviamm. _ la gráfica presenta la Lluvia en milímetros un comportamiento estacional propio de la época de lluvias durante los primeros 3 o 4 meses del año.
- Dirdelviento. _ Se presenta la dirección del viento en variación de grados desde 0 a 360 con mayor actividad durante los primeros tres meses de cada año.

Sin embargo, en la gráfica no es sencillo determinar si existe correlación entre las variables de clima y de Energía Activa.

Debido a que no se puede establecer visualmente una correlación en la Figura 3-2, realizamos un cambio en la visualización de los datos por gráficos de cajas por hora para cada variable, la idea de realizar la visualización por cajas es el de apreciar su comportamiento y agrupación por horas, recordando que los gráficos de cajas nos brindan información resumida de los cuartiles estadísticos, es decir cómo se agrupan los datos y de la mediana, es posible resumir los datos a un periodo de 24 horas, esto con el fin de determinar si mediante este cambio se puede apreciar alguna relación en el

comportamiento de las variables, esto da como resultado las visualizaciones de la **Figura 3-3**, donde se puede apreciar una ligera correlación entre el consumo por hora con la temperatura y la Humedad relativa HR:

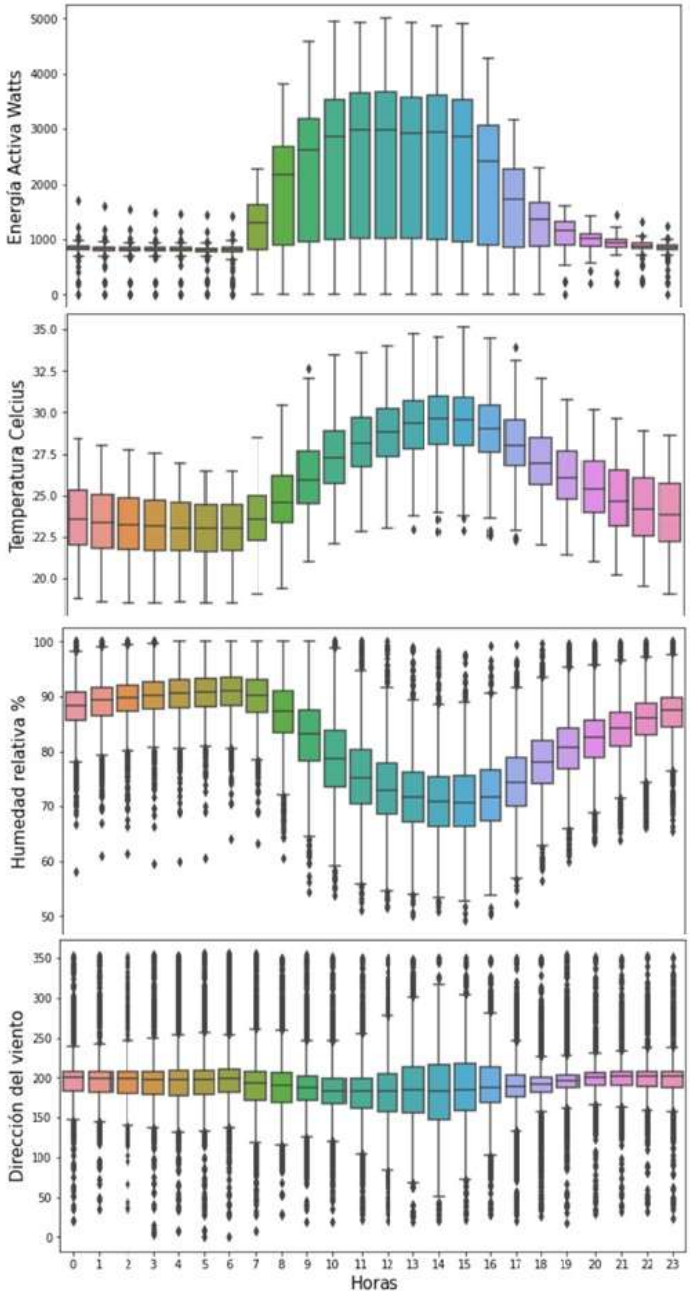

Figura 3-3 Visualización por hora de las variables del Dataset
Fuente: El Autor

La matriz de correlación de la **Figura 3-4** muestra que las variables Temperatura y Humedad relativa (HR) tienen un mayor índice de correlación con la variable Energía Activa, donde el coeficiente de correlación $r > |0.4|$, recordando que mientras más cercano a 1 se encuentre r tanto mayor será la correlación.

Con lo cual, podemos descartar la lluvia, dirección del viento y velocidad de ráfagas para los propósitos de predicción, ya que su coeficiente de correlación es menor $r < |0.4|$.

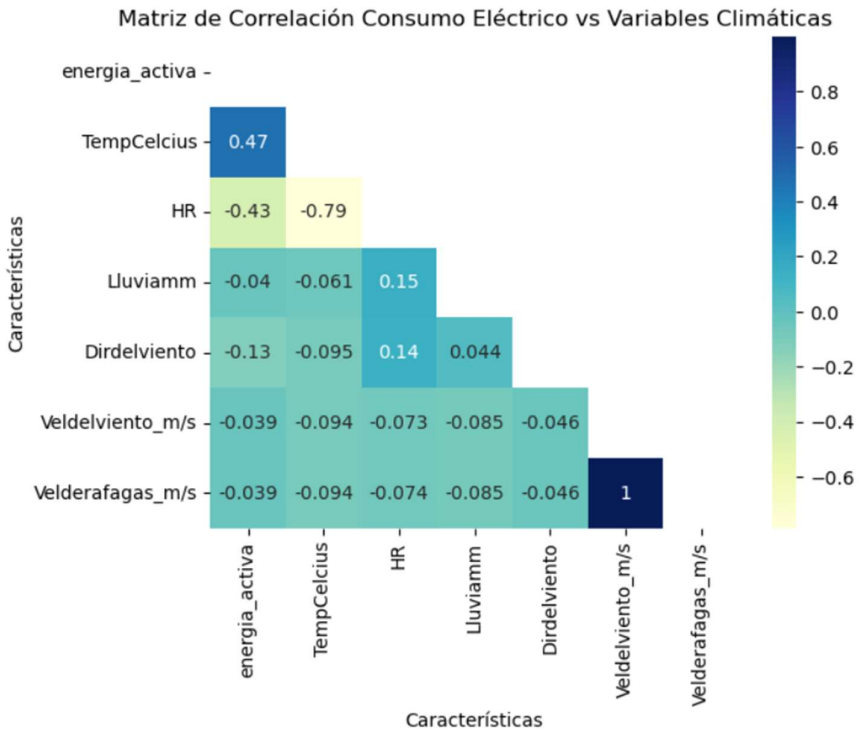

Figura 3-4 Matriz de correlación (diagonal inferior)
Fuente: El Autor

En cuanto al perfil de consumo de Energía Eléctrica por día de la semana, si se compara los días de la semana de cada año representado como columnas y los años representados en las filas como se muestra en la

Figura 3-5 tomando el promedio de cada hora por día de la semana, podemos determinar de forma visual que son similares, y que una anormalidad en el perfil de consumo eléctrico ocurre en 2020, esto debido a la pandemia declarada en el mes de marzo de 2020:

- Dia de semana 0 = Lunes
- Dia de semana 1 = Martes
- Dia de semana 2 = Miércoles
- Dia de semana 3 = Jueves
- Dia de semana 4 = Viernes
- Dia de semana 5 = Sabado
- Dia de semana 6 = Domingo

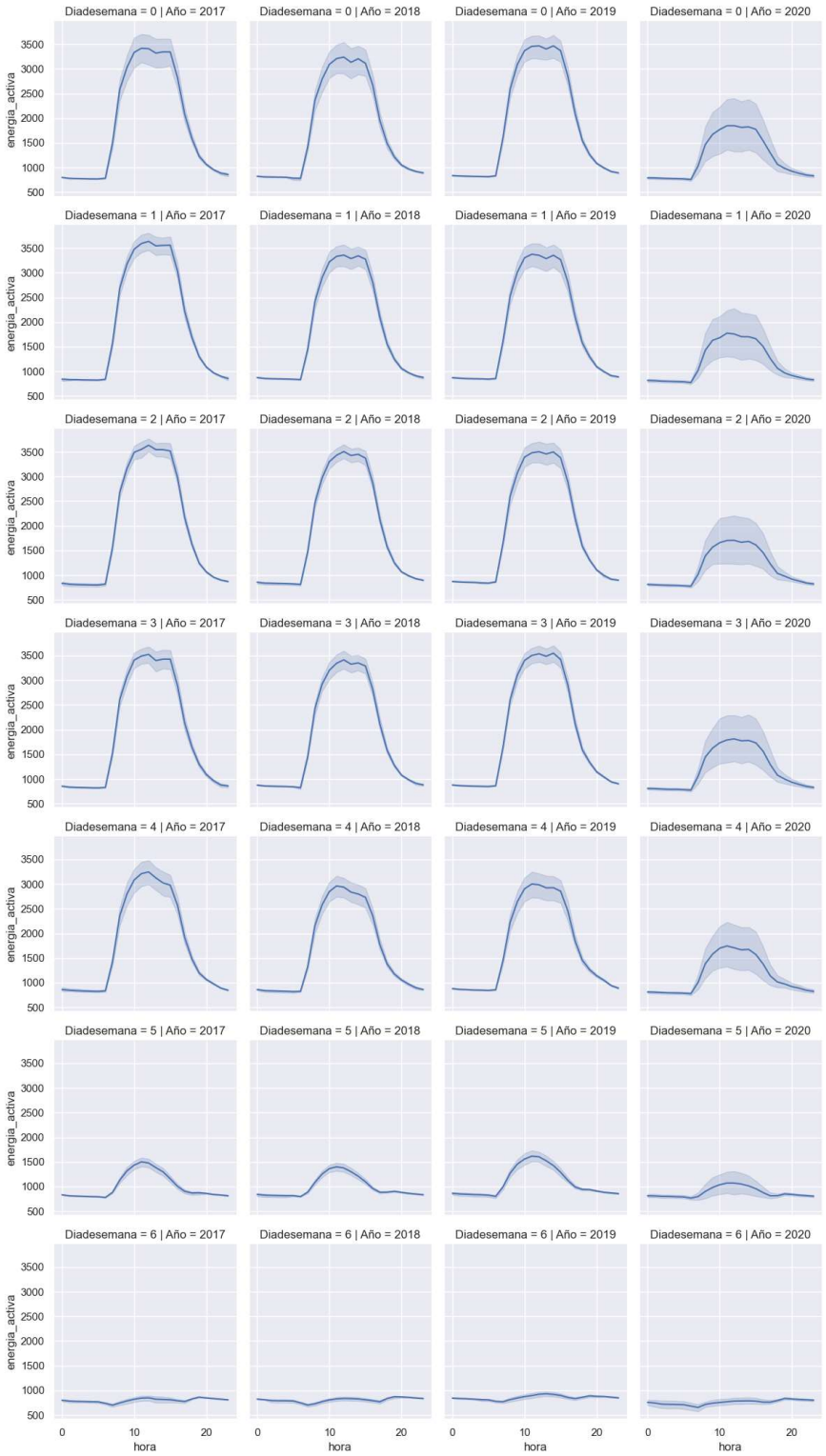

Figura 3-5 Perfil de consumo de Energía por hora por día
Fuente: El Autor

Se puede apreciar en las gráficas además una sombra sobre y debajo de cada línea azul sólida, esto representa los valores mínimos y máximos, y la línea en azul sólido representa el promedio.

También se puede apreciar que por día de la semana el mayor consumo de Energía eléctrica ocurre entre las 8 AM y las 4 PM, y es coincidente con el aumento de temperatura de la Figura 3-3.

3.3 **Prototipos de algoritmos y modelos de aprendizaje de máquina**

A continuación, se presenta los tres modelos de predicción basados en las redes neuronales recurrentes descritas en el Capítulo 2, a continuación, se realizará la descripción de la implementación y pruebas con los siguientes algoritmos para determinar los mejores resultados en la predicción del consumo eléctrico:

- Red neuronal recurrente de Elman o RNN,
- Red neuronal de memoria de corto y largo plazo o LSTM
- Unidad de compuertas recurrentes o GRU.

Estas redes fueron implementadas usando Pytorch, la arquitectura de red que se platea en el proyecto se describe en la Figura 3-6, la arquitectura muestra los datos de entrada hacia una red del tipo LSTM, RNN o GRU, la cual analizará y extraerá información de las variables temporales, a continuación su salida se alterna entre redes Totalmente conectadas FNN1 y FNN2, la implementación sigue el modelo planteado por (Kong et al., 2019) para la predicción de la demanda usando redes LSTM, se alterna con funciones de activación ReLu en las capas intermedias para evitar el desvanecimiento del gradiente como se menciona en el estudio conducido por (Ryu et al., 2017), la función ReLu se define en (3.2)

$$f(x) = \max(0, x) = \begin{cases} 0, & x < 0 \\ x, & x \geq 0 \end{cases} \qquad (3.3)$$

La función de estas redes Totalmente conectadas es detectar características adicionales en la salida de las redes recurrentes, tomando la cantidad de salidas del estado oculto y reduciéndolo hasta entregar una salida única.

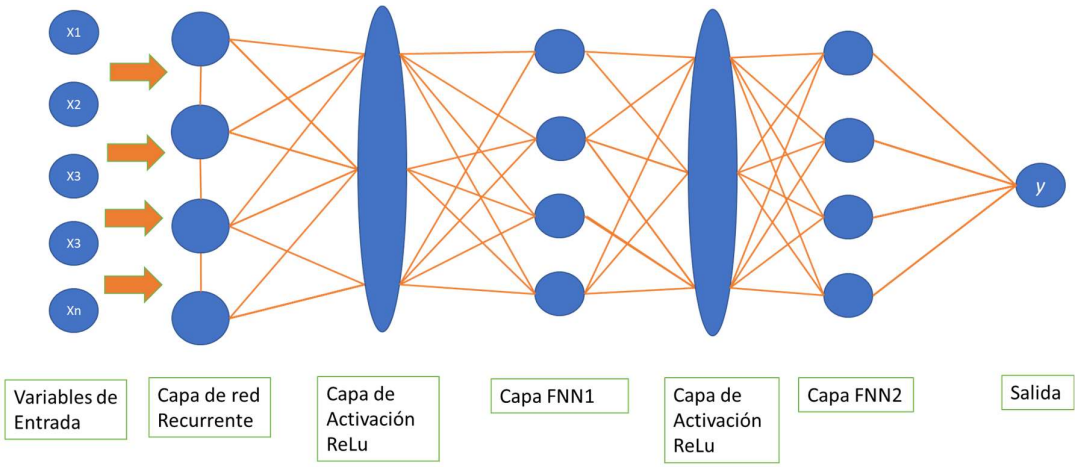

Figura 3-6 Arquitectura de Red

Los parámetros con los cuales se trabaja dentro de los algoritmos de Aprendizaje de máquina se denominan hiper parámetros, en el caso de las redes neuronales recurrentes se puede mencionar entre una gran variedad los siguientes:

- Dimensión de Entrada, es el número de variables que ingresan a la red neuronal, en este caso Temperatura, Humedad relativa y la data rezagada en 1 hora de la energía Activa y además se incluye variables para el año, el mes, día, día de la semana, pandemia, se aplica técnicas de codificación binaria para el mes, día y día de la semana.
- Dimensión de salida, es el número de variables de salida, en este caso, la predicción del consumo en Wh.

- Entrada Capa oculta, es el número neuronas por cada capa recurrente y está ubicada entre las entradas y las salidas.
- Capas, es el número de niveles que tiene la capa oculta.
- Tamaño de Lote, se refiere al número de muestras que se enviarán a la red neuronal.
- Longitud de secuencia, se refiere al número de observaciones que ingresan al modelo.
- Número de épocas, veces en las cuales la red realiza el entrenamiento, es decir que los valores realizan una vuelta completa para resultado y ajuste hacia atrás para determinar los pesos mínimos W.
- Tasa de aprendizaje, valor que se envía a la función de error poder moverse en esa proporción hacia el error mínimo.

De la revisión bibliográfica realizada en (Goodfellow et al., 2016) Afinamiento de hiper parámetros, indican los autores que uno de los más importantes elementos a tomar en cuenta en el afinamiento de hiper parámetros de los modelos de redes neuronales y que tiene un mayor impacto es la tasa de aprendizaje, se indica que cuanto mayor es la tasa de aprendizaje mayor es el error, y si la tasa de aprendizaje es muy pequeña no se podrá optimizar el error quedando el modelo detenido y cuyo resultado además es un elevado error. Durante el proceso de pruebas se seleccionaron 7 hiper parámetros y cuyo proceso de búsqueda se realizó usando el método de búsqueda en grilla, lo cual significa que se barren todas las combinaciones posibles para las variables a continuación:

- Tasa de Aprendizaje, 0.001, 0.0001, 0.00001
- Entradas Red, número de neuronas 2, 4, 8
- Número de capas, 2, 4, 8
- Redes neuronales, 'rnn', 'gru', 'lstm'

- Longitud del Batch 24, 48
- Capa Conectada totalmente, 4, 8
- Longitud de secuencia 24, 48

En total se probaron 648 combinaciones posibles, probándose en 20 épocas, definiendo épocas como el número de veces que el modelo es entrenado usando una combinación especifica de hiper parámetros y que alcanza un error menor, con el fin de probar el mejor modelo entre 3 se deja fijo la cantidad de épocas a 20.

En la **Tabla 3.3**, se muestra un ejemplo con 10 modelos de los 648 resultados.

Tabla 3.3 Hiper parámetros de arquitecturas Redes neuronales

Epocas	Tasa de aprendisaje	Red Neuronal	Entradas Red Recurrente	Número de capas	Entradas red Totalmente conectada	Longitud de Batch	Longitud de secuencia
20	0.001	lstm	8	2	8	24	24
20	0.001	rnn	8	8	8	48	48
20	0.001	gru	8	4	8	24	24
20	0.001	gru	8	4	8	48	24
20	0.001	rnn	8	4	8	24	24
20	0.001	gru	8	4	4	48	48
20	0.001	gru	8	8	4	48	48
20	0.001	gru	8	2	4	24	48
20	0.001	gru	8	8	8	48	24
20	0.001	lstm	8	2	4	48	24

Fuente: El Autor

3.4 Infraestructura para procesamiento y almacenamiento de datos

La arquitectura que se propone para la predicción y visualización de la Demanda se muestra a continuación en la Figura 3-7.

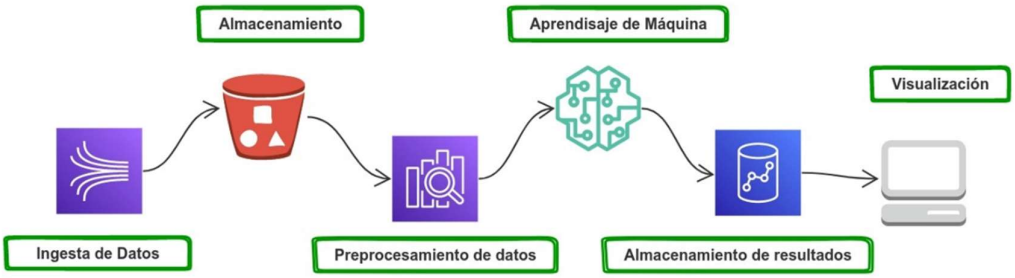

Figura 3-7 Solución de Infraestructura

En los siguientes apartados se describe la infraestructura propuesta de forma detallada.

3.4.1 Ingesta de Datos y Almacenamiento

En la ingesta de datos el usuario cuenta con dos repositorios; el primero, almacena los datos de Energía activa desde 2017 hasta 2020.

El segundo repositorio, almacena la información del clima organizado por años y por meses.

El tamaño total que ocupan ambos repositorios con la data disponible de 2017 a 2020 es de aproximadamente 20 MB, con lo cual se recomienda mantener al menos 40 MB, se estima que con este espacio se puede almacenar datos para los siguientes 4 años de data.

A continuación, se definirá los módulos de programa que actúan sobre el proceso de Ingesta y Almacenamiento.

Se implemento el módulo de programa “Loader”, el cual toma la información de las carpetas de Energía activa y de clima, realizando una operación de “Concatenación” de

todos los archivos de clima en un único dataset para variables climáticas, y lo propio con los distintos archivos de consumo eléctrico. La salida de la ingesta de data son dos archivos en formato .csv, uno para clima y otro para el consumo eléctrico, y se almacenan en el directorio “Clean”.

A continuación, el módulo de programa denominado “LIMPIEZA_DATASET” se encarga de realizar los pasos descritos en el apartado 3.1 Limpieza, Normalización y agrupación de los datos. La salida son dos archivos conteniendo data legible, sin datos nulos, eliminando duplicidad de registros, estos archivos contienen los registros de energía activa y su índice de tiempo en formato día-mes-año hora: minutos: segundos y los registros meteorológicos con índice día-mes-año hora: minutos: segundos, este proceso se resume en la Figura 3-8 Proceso de Ingesta de Datos.

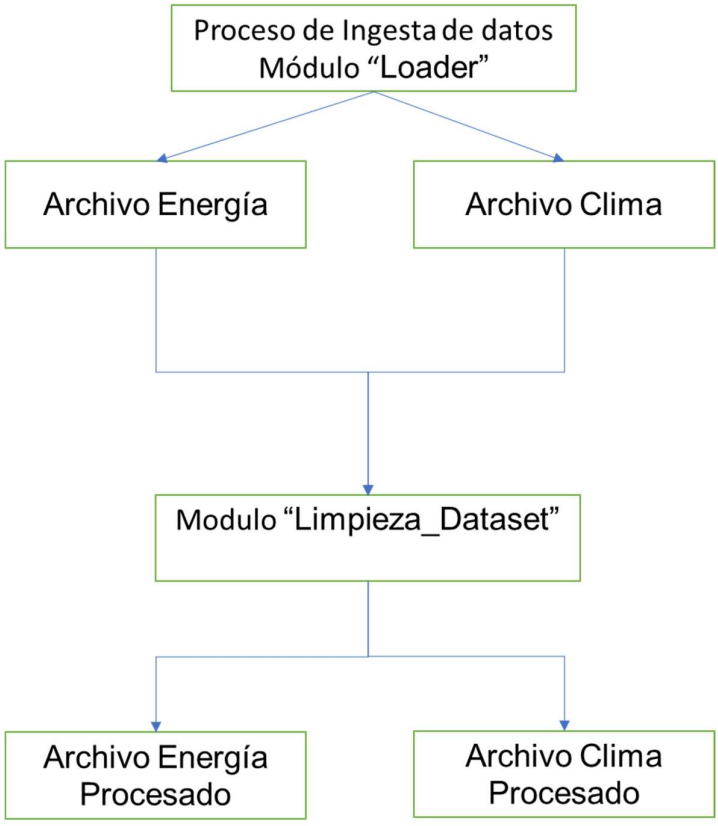

Figura 3-8 Proceso de Ingesta de Datos

Fuente: El Autor

3.4.2 Procesamiento de los datos y Machine Learning

El módulo denominado Procesamiento de los datos y Machine Learning toma los archivos que resultan del Proceso Ingesta de Datos y Almacenamiento, los datasets son procesados por el módulo de programa “Estat_graf”, este módulo unifica los datasets de Energía Activa y de Clima procesados como se muestra la **Figura 3-9,** esto con el fin de que el usuario pueda determinar la correlación existente entre las variables del dataset, una de sus salidas es directamente al proceso Cliente donde se visualizará la Línea Base de Consumo y la correlación de los datos, así como gráficas de Perfil de Consumo.

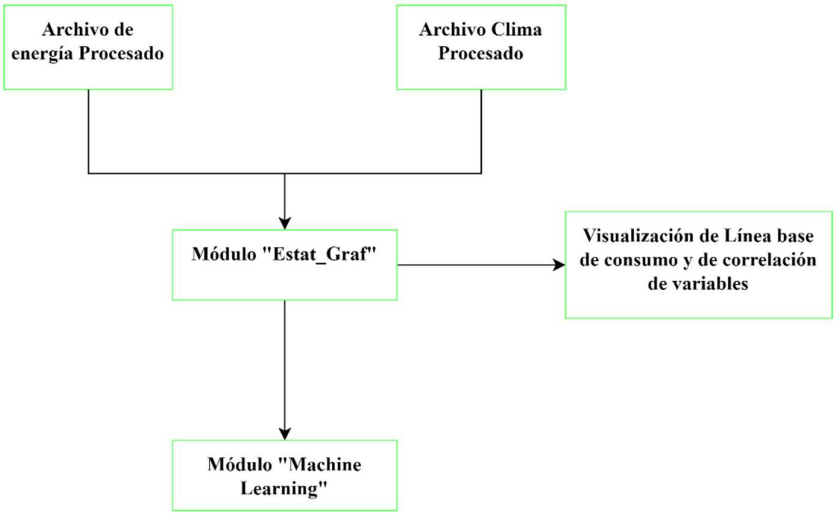

Figura 3-9 Procesamiento de los datos y Machine Learning

Fuente: El Auto

Con la información de correlación se define cuales variables se van a enviar al Módulo de Machine Learning, el cual se describe a continuación.

El módulo “Machine Learning” que se encarga de realizar el pronóstico y se implementa bajo la arquitectura descrita en el apartado 3.3, una vez analizadas las variables y su correlación.

El subproceso de Normalización es aplicado como se describió en el apartado 3.1, la normalización se realiza con el fin de que las operaciones matemáticas sean óptimas y que no se penalice a los coeficientes de las variables de entrada (Ajantha Devi & Naved, 2021).

El subproceso de división de data realiza cortes del dataset al 60% para datos de entrenamiento, 20% para datos de validación y 20% para datos de prueba, tomando en cuenta que los datos de prueba se usarán como resultado y visualización de la herramienta, estos datos no serán usados durante el entrenamiento o la validación.

El subproceso de entrenamiento envía los datos de “Entrenamiento” por la arquitectura propuesta en la Figura 3-6 tantas veces como épocas se hayan definido, en caso de que se haya realizado un entrenamiento previo, el subproceso recupera el modelo guardado, en caso de no existir, una vez terminado el entrenamiento el modelo se guarda.

La fase de validación ayuda con el ajuste de los hiper parámetros usando para ello las 648 posibles combinaciones, este proceso se lo lleva a cabo usando las librerías Ray Tune, la cual nos proporciona un ambiente tanto para pruebas como para reportar el error de validación de cada una de las combinaciones citadas.

Con esta última fase, es decir la fase de validación, el modelo queda listo para recibir la información por parte del usuario, esta data debe contener al menos 24 datos completos, ya que el modelo fue entrenado con dicha secuencia como entrada con energía activa, tiempo, humedad relativa y temperatura, además de los pronósticos de las variables climáticas descritas en el apartado 2.3 y que tienen una alta correlación con la variable a predecir.

El procedimiento indicado anteriormente se resume en la

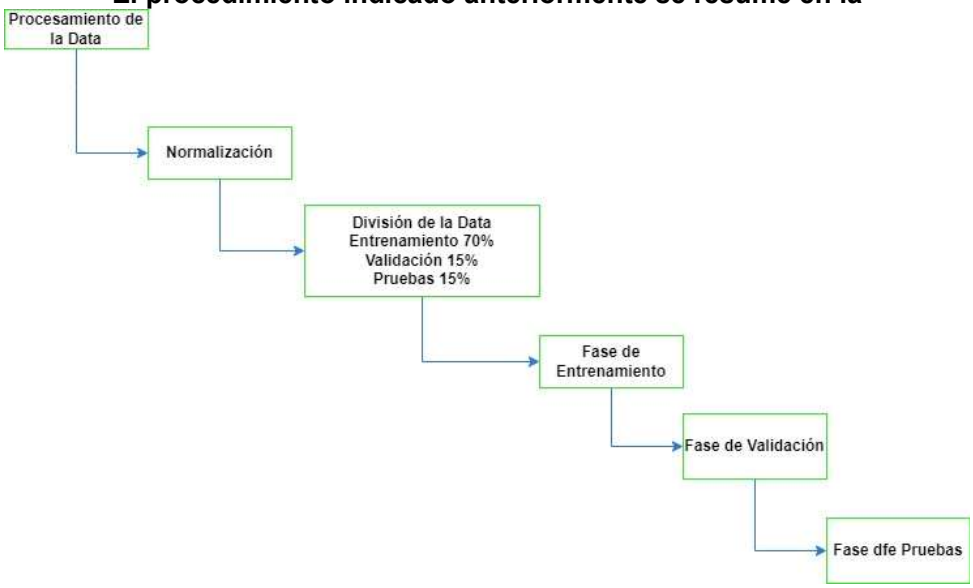

Figura 3-10.

Procesamiento de la Data
Normalización
División de la Data
Entrenamiento 70%
Validación 15%
Pruebas 15%
Fase de Entrenamiento
Fase de Validación
Fase dfe Pruebas

Figura 3-10 Subprocesos del Módulo de Machine Learning

Fuente: El Autor

3.5 Prototipo de Visualización

Uno de los objetivos de este trabajo es proporcionar una herramienta para que los usuarios que realizan gestión de energía y sostenibilidad del campus de la ESPOL, para este objetivo proponemos un modelo de visualización sencillo que busca comunicar las salidas tanto del consumo de la línea base como del modelo de predicción, el prototipo de visualización está dividido en tres secciones, las cuales se indican a continuación:

Consumo. _ Se puede visualizar los datos históricos y de esta forma contar con una herramienta que muestre la línea base tanto de Energía como de las variables climáticas correlacionadas, los valores se agrupan como una media por hora, mes, día y año.

A continuación, en la **Figura 3-11** se muestra la herramienta de visualización, donde al seleccionar Consumo se visualiza la serie temporal de consumo eléctrico del Campus, el usuario tiene la libertad de seleccionar el año, variable clima, así como el mes donde se realiza el consumo.

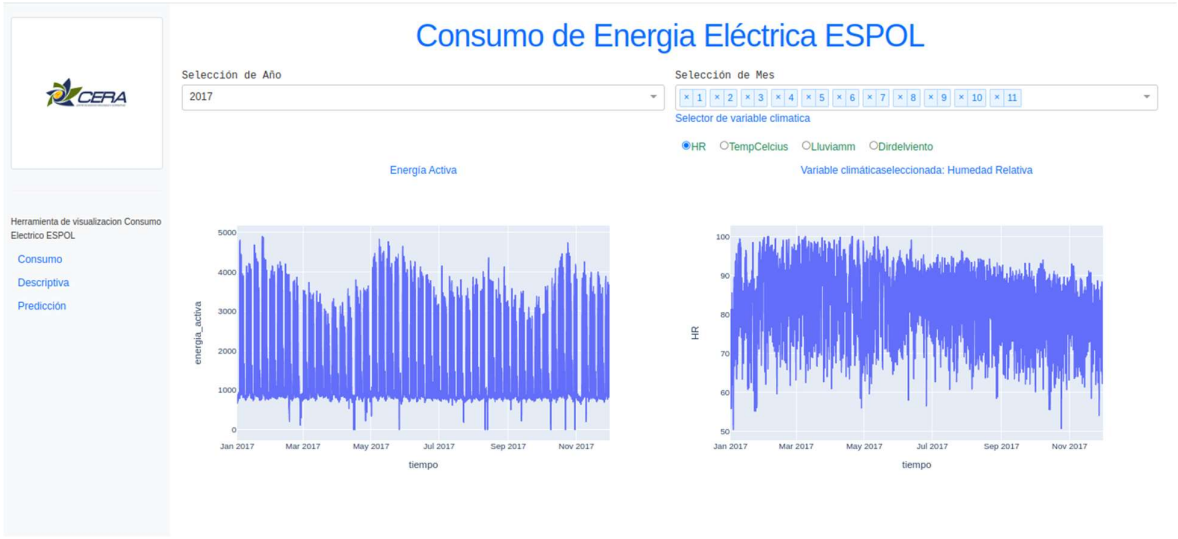

Figura 3-11 Prototipo de visualización de línea base de consumo

Fuente: El Autor

Descriptiva. _ Se muestra dos gráficos uno de cajas para cada variable, recordando que el gráfico de cajas es importante para mostrar la información de máximos, mínimos, cuartiles y consumo medio del periodo seleccionado, y como están agrupados los datos.

El gráfico de correlación muestra la correlación entre variables, el usuario puede en este punto seleccionar las variables de interés y la frecuencia de agrupamiento.

En la selección descriptiva como se muestra en la Figura 3-12 se muestra tanto las gráficas de caja como de correlación en una visualización del tipo Mapa de Calor, donde los colores más oscuros muestran una mayor correlación.

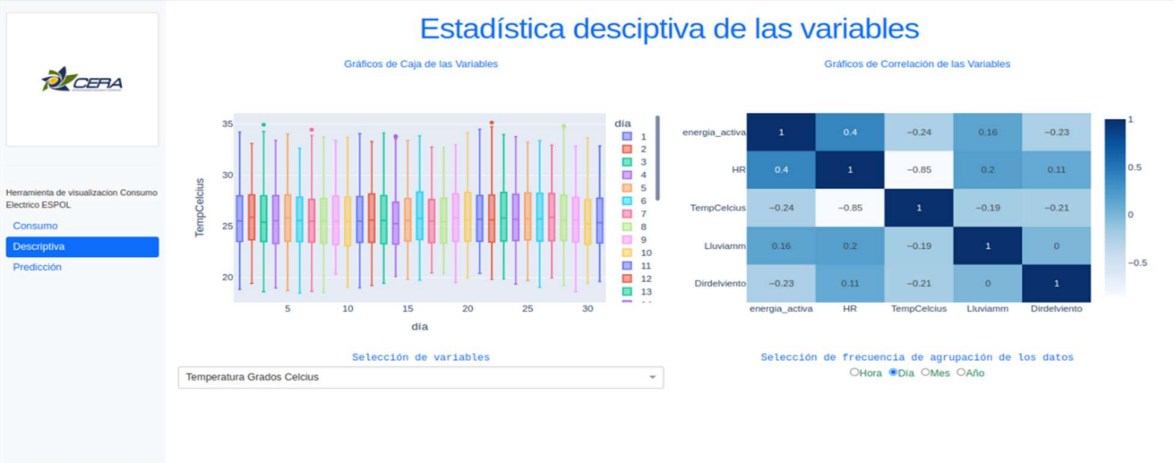

Figura 3-12 Prototipo de visualización Estadística de las Variables

Fuente: El Autor

Predicción._ El usuario debe subir un archivo .csv o Excel con al menos 24 horas de datos de, Energía Activa, Humedad Relativa (HR), temperatura, Lluvia en milímetros, y Dirección del viento, además del pronóstico de las variables climáticas descritas, luego al realizar click sobre el botón de "Predicción", se muestra la predicción sobre la data que el usuario está

proporcionando, como se indica se requiere como mínimo las 24 primeras observaciones con los campos solicitado, en la herramienta se muestran estas primeras observaciones en azul, esto es debido a que el modelo fue entrenado con esta longitud de secuencia para que el modelo realice el pronóstico de los siguientes valores de Energía Activa, esto lo realiza prediciendo un paso u hora al futuro y retroalimentando este valor a la variable energía activa, el algoritmo repite esta acción las n observaciones del dataset del usuario.

Figura 3-13 Visualización de la Predicción del consumo Eléctrico

Fuente: El Autor

3.6 **Métricas**

Las métricas que se definieron en el Capítulo 2 para evaluar el rendimiento de los modelos de pronóstico donde se compara el resultado de las predicciones con el valor real de consumo. Estas evaluaciones se realizan usando métricas como MSE, MAE. La evaluación final del comportamiento del modelo se la realizó en los datos de prueba, para medir el comportamiento de las predicciones al retroalimentar el modelo con datos de consumo predicho.

CAPÍTULO 4

4 ANÁLISIS DE RESULTADOS

En esta sección evaluaremos los algoritmos descritos en el Capítulo 3, utilizando los hiper parámetros mostrados en la Tabla 3.3 Hiper parámetros de arquitecturas Redes neuronales. Uno de los objetivos de este trabajo es realizar predicciones de consumo eléctrico a diferentes horizontes de tiempo en la siguiente sección se muestran los resultados obtenidos.

4.1 Selección del modelo acorde a los resultados de las métricas

Una vez que se cumplió el proceso Interno denominado 4. Algoritmos de Aprendizaje de Máquina y cuyo módulo de programa se denomina 'Main' para cada una de las redes recurrentes que se revisaron durante los capítulos previos se obtuvo los resultados en la **Tabla 4.1** con el dataset de entrenamiento y validación, recordando que el dataset de entrenamiento y de validación fueron descritos en el apartado 3.4.2, como mejor resultado en los datos de entrenamiento se obtuvo una configuración de red RNN, sin embargo al evaluar las métricas de validación los mejores resultados con los datos de validación se dieron con redes LSTM, RNN y GRU, en los tres primeros lugares, la diferencia es mínima como se puede apreciar en la Tabla 4.1.

Tabla 4.1 Resultados entrenamiento

Epocas	Error de Entrenamiento (MSE)	Error de Validación (MSE)	Longitud de Batch	Entradas a la red recurrente	Entradas en la red Recurrente	Tasa de Aprendizaje	Número de capas Ocultas Red Recurrente	Red Neuronal	Longitud de secuencia
20	6.28E-04	4.53E-04	24	8	8	1.0E-03	2	lstm	24
20	6.99E-04	4.63E-04	48	8	8	1.0E-03	8	rnn	48
20	5.90E-04	4.65E-04	24	8	8	1.0E-03	4	gru	24
20	6.28E-04	4.68E-04	48	8	8	1.0E-03	4	gru	24
20	6.22E-04	4.71E-04	24	8	8	1.0E-03	4	rnn	24
20	6.89E-04	4.79E-04	48	8	4	1.0E-03	4	gru	48
20	7.39E-04	4.94E-04	48	8	4	1.0E-03	8	gru	48
20	6.17E-04	4.95E-04	24	8	4	1.0E-03	2	gru	48
20	7.50E-04	5.03E-04	48	8	8	1.0E-03	8	gru	24
20	6.45E-04	5.24E-04	48	8	4	1.0E-03	2	lstm	24

Fuente: El Autor

En la figura Figura 4-1 se muestra la variación del error de validación con respecto a las Épocas de los 10 mejores resultados.

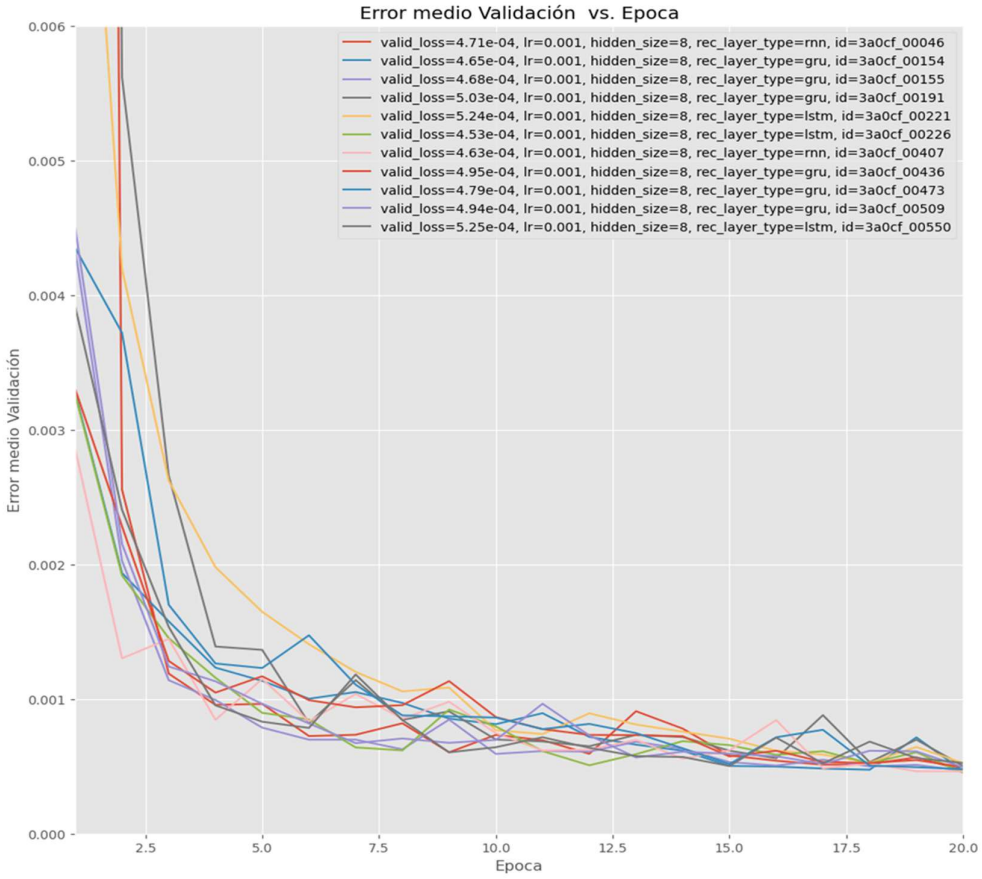

Figura 4-1 Gráficas de Error de Validación

Fuente: El autor

Para una mejor visualización en la Figura 4-2 se muestra el mejor resultado con su error de Validación,

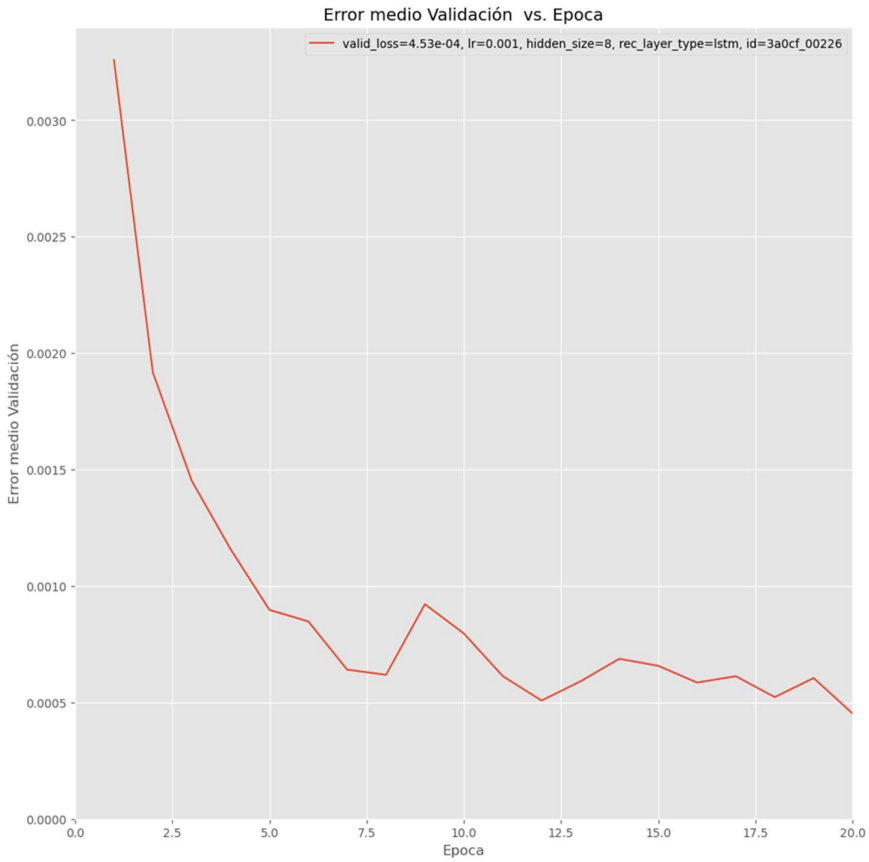

Figura 4-2 Error de Validación mejor resultado

Fuente: El autor

4.2 Puesta en marcha y funcionamiento

Para poner en marcha el sistema se debe correr el proceso Interno y el proceso Cliente descritos en el Capítulo 3, iniciando por el proceso Interno donde se realiza la ingesta de los datos tanto de consumo eléctrico, así como de variables climáticas, esto se realiza ejecutando el módulo “Loader”, una vez generados los archivos de energía y de clima se procede con el módulo “Limpieza_Dataset” para realizar la limpieza y procesamiento de los datos acorde a la Figura 3-8 Proceso de Ingesta de Datos.

A continuación, se ejecuta los módulos de Figura 3-9 Procesamiento de los datos y Machine Learning, se ejecuta el módulo “Estat_graf” para realizar el análisis de correlación y descubrir que variables usaremos para predecir el Consumo Eléctrico.

Se ejecuta luego el módulo “Machine Learning” el cual realiza entre otras tareas la ingesta de datos a la red neuronal para realizar el entrenamiento, como salida tenemos el modelo de la red LSTM como mejor modelo de predicción acorde a los resultados obtenidos en el apartado 4.1.

Finalmente, el proceso Cliente vía la herramienta de visualización podrá acceder a las gráficas de la Línea base, matriz de correlación, la data correspondiente al perfil de consumo eléctrico, además de la predicción donde también puede ingresar la data correspondiente para realizar subsiguientes pronósticos sin requerir del entrenamiento, esto debido a que la herramienta toma los datos del modelo previamente entrenado.

4.3 **Pruebas de funcionalidad**

Con el fin de probar el funcionamiento de la herramienta se realizaron varias pruebas, tomando en cuenta que, durante el año 2020, los datos desde el 17 de marzo hasta Julio 2020 son datos con pandemia con lo cual se realizó lo siguiente:

1. Se dividió el dataset en dos partes denominadas prepandemia con 28104 datos y Pandemia con 3179 datos.
2. Para los datos prepandemia se dividió el dataset en tres partes, 70% para datos de entrenamiento, 15% para datos de validación y 15% para datos de prueba, una vez más se indica que los datos de prueba son únicamente usados para probar los modelos finales y su desempeño, no son presentados previamente a los modelos en entrenamiento/validación y tampoco para ajuste de hiper parámetros.

3. El 15% de datos de prueba corresponden a un máximo de 3 meses, Enero a Marzo 2020, no se incluye los datos de pandemia.
4. La red neuronal que mejor desempeño mostró con los datos de prueba fue la GRU con número de iteración 30 para los datos prepandemia alcanzando un R2 del 92%, los datos en pandemia no se analizaron debido a la situación extraordinaria que se vivió posterior al 17 de Marzo de 2020 donde se suspendieron actividades en el Campus.

4.4 **Análisis costo/beneficio**

El contar con una herramienta de visualización de grandes volúmenes de datos permite a la ESPOL conocer su línea base de consumo de energía eléctrica, y al realizar el pronóstico con base a variables climatológicas permite al operador del campus conocer pronósticos de hasta 6 meses del consumo de energía del campus.

Para la implementación de la herramienta a continuación se plantea el siguiente presupuesto:

Tabla 4.2 Costos de la implementación

Importe	Cloud 12 meses	Equipos físicos
Nombre de instancia (ml.t3.xlarge), Número de científico(s) de datos (1), Número de instancias del bloc de notas de Studio por científicos de datos (1), Horas de bloc de notas de Studio por día (8), Días del bloc de notas de Studio por mes (22)	$ 35,20	$ 250,00
Almacenamiento (SSD de uso general (gp2)), Nombre de instancia (ml.c4.2xlarge), Número de trabajos de procesamiento al mes (10), Número de instancias por trabajo (1), Horas por instancia y trabajo (128)	$ 1.225,68	
Almacenamiento (SSD de uso general (gp2)), Nombre de instancia (ml.c4.2xlarge), Nombre de instancia (ml.c4.2xlarge), Número de modelos implementados (1), Número de modelos por punto de enlace (1), Número de instancias por punto de enlace (1), Horas de punto de enlace al día (8), Días de punto de enlace al mes (20), Datos de entrada procesados (10 GB), Datos de salida procesados (10 GB)	$ 938,40	$ 1.250,00

Desarrollo Inical	$ 660,00	$ 660,00
Ajuste Inicial	$ 150,00	$ 150,00
Mantenimiento	$ 240,00	$ 240,00
Total	$ 3.249,28	$ 2.550,00

Fuente: El Autor

Los costos presentados en la **Tabla 4.2** muestran dos opciones para el desarrollo de la herramienta y puesta en producción, la implementación en la nube o implementación física.

Durante la implementación del proyecto se determinó que el espacio físico utilizado para almacenar tanto la data proveniente de consumos eléctricos, de variables climáticas, así como los datos post procesados es de 139 MB para un total de 136897 datos de Energía Activa y 254937 datos de variables climáticas para un periodo comprendido entre 2017 y 2020, es decir 4 años, en total la solución almacenando los scripts, el espacio requerido es de 240MB, con lo cual contar con al menos 1GB de espacio resultaría suficiente para poder almacenar la solución, para los siguientes 12 años de información.

En el caso de la implementación en la nube, el almacenamiento cotizado es por 10 GB mientras que para la solución local se usó una laptop con espacio disponible de 500GB, esto debido a que estas soluciones tienen como mínimo estos formatos.

Para establecer cuál de las dos soluciones tiene una mejor rentabilidad evaluaremos con el método de la TIR, este método se plantea en el estudio realizado por (López, 2012) para evaluación económica de proyectos de software, la TIR es la medida de que tan viable resulta una inversión comparada con otra, la fórmula que se usa para este cálculo es la siguiente:

$$Van = -I_0 + \sum_{t=1}^{n} \frac{F_t}{(1+TIR)^t} \qquad (4.1)$$

Donde:

- Van es el valor actual neto
- $-I_0$ es la inversión inicial
- TIR es la tasa interna de retorno que se va a comparar entre los dos proyectos
- F_t son los flujos de caja o ingresos mensuales

Supondremos que los ingresos por alquiler de la herramienta de visualización de forma mensual son de 300 USD, lo cual incluye capacitación en el uso de la herramienta, ajuste fino cada tres meses, con lo cual debemos igualar el Van = 0 y procedemos a despejar la TIR de la ecuación 4.1.

El resultado se muestra en la **Tabla 4.3** Tasa Interna de Retorno donde se puede apreciar que el proyecto local es más rentable.

Tabla 4.3 Tasa Interna de Retorno

	Cloud	Local
Inversión Inicial	-3249	-2550
Ingresos Mensuales T1	300	300
Ingresos Mensuales T2	300	300
Ingresos Mensuales T3	300	300
Ingresos Mensuales T4	300	300
Ingresos Mensuales T5	300	300
Ingresos Mensuales T6	300	300
Ingresos Mensuales T7	300	300
Ingresos Mensuales T8	300	300
Ingresos Mensuales T9	300	300
Ingresos Mensuales T10	300	300
Ingresos Mensuales T11	300	300

Ingresos Mensuales T12	300	300
TIR	2%	6%

Fuente: El Autor

Al momento la ESPOL no cuenta con una herramienta de predicción en el formato que se presenta el proyecto, La ESPOL realiza los pronósticos mediante el uso de una Base de Datos de Edificios así como el uso de hojas de cálculo, no se cuenta con información sobre los valores que invierte la Universidad por licencias de hojas de cálculo, tiempo de una persona para procesar la información y realizar el pronóstico con la metodología descrita.

La herramienta de predicción y visualización propuesta no requiere del pago de licencias, ya que se utilizó herramientas de desarrollo libre, se requiere de soporte cada 6 meses para revisar el modelo de pronósticos con una persona en sitio para afinar el modelo con la nueva data que se genere durante este periodo, el costo se encuentra incluido dentro del pago mensual propuesto en la Tabla 4.3

CAPITULO 5

5 CONCLUSIONES Y RECOMENDACIONES

Una solución basada en datos permite visualizar data a gran escala para el monitoreo del base line de consumo, permite interactuar con la data, es decir evaluar el comportamiento de las partes o el todo, es Adaptable, con esfuerzos adicionales en desarrollo, se puede lograr visualizaciones acordes a los requerimientos del cliente y ofrece la posibilidad de implementarse en la nube o en equipos del cliente

5.1 Conclusiones

A partir de los resultados obtenidos en el presente trabajo se puede concluir lo siguiente:

- El uso de modelos de Aprendizaje de Máquina y específicamente de Aprendizaje Profundo con redes recurrentes se puede usar para predecir series temporales multivariadas.
- El uso exclusivo de variables climáticas como temperatura y humedad relativa no son suficiente para lograr un buen ajuste, por lo que el investigador debe recurrir a extraer data por ejemplo de la fecha y hora además de enviar al modelo la serie temporal de “Energía_Activa” con retraso de 1 hora.
- Las predicciones basadas en desarrollo interno y usando tablas de cálculo son fáciles de usar en comparación con un modelo basado en Datos y Aprendizaje de Máquina, sin embargo los modelos y herramientas basadas en Datos son más flexibles, soportan grandes volúmenes de datos, permite visualizaciones de forma interactiva e implementación en la nube, lo que las hacen accesibles.

- Las herramientas de visualización interactivas aportan al pensamiento cognitivo dando la oportunidad al usuario de explorar la data.

5.2 Recomendaciones

El presente trabajo se limitó a realizar tanto pruebas como verificaciones sobre data existente en los datasets de Consumo y Clima desde el 1 de Enero de 2017 hasta el 27 de Julio de 2020 dividiendo en datos de entrenamiento (60%) para evaluación y test el restante 40%, (20% para test y 20% para evaluación).

Cabe destacar que desde el 17 de marzo de 2020 el campus Gustavo Galindo ha permanecido cerrado por condiciones de pandemia y para precautelar la salud de los estudiantes, profesores y personal administrativo que labora en el campus, una vez que se reabra el campus, se recomienda lo siguiente:

1. Ajustar el modelo con la nueva carga, teniendo en cuenta condiciones de restricción de aforo de la universidad.
2. El uso de una serie temporal de ocupación del campus es deseable, se puede usar por ejemplo el uso de internet como medida de ocupación.
3. Usar datos de variables climáticas predichas para conocer como se comportará el campus en los siguientes periodos.

El presente trabajo también muestra una solución basada en datos la cuál encontró lo siguiente al momento de procesar la data:

1. Los datos de consumo eléctrico fueron entregados en archivos con extensión Excel, en algunos casos sin formato estándar, por lo que fue necesario preprocesarlos de forma independiente.
2. Los datos de clima se recibieron en un formato .csv y campos bien delimitados.

Por lo que para futuros trabajos uno de los requerimientos de la herramienta sería definir los formatos en los cuales la información se va a receptar.

BIBLIOGRAFÍA

Abdulrahman, M. L., Nerat, J. J., & Abdulsalam, Y. G. (2019). *LSTM Network for Predicting Medium to Long Term Electricity Usage in Residential Buildings*. *9*(2), 21–30. https://doi.org/10.5923/j.computer.20190902.01

Agarwal, Y., Weng, T., & Gupta, R. K. (2009). The energy dashboard: Improving the visibility of energy consumption at a campus-wide scale. In *BUILDSYS 2009 - Proceedings of the 1st ACM Workshop on Embedded Sensing Systems for Energy-Efficiency in Buildings, Held in Conjunction with ACM SenSys 2009* (pp. 55–60). https://doi.org/10.1145/1810279.1810292

Ahmad, M. W., Mourshed, M., & Rezgui, Y. (2017). Trees vs Neurons: Comparison between random forest and ANN for high-resolution prediction of building energy consumption. *Energy and Buildings*, *147*, 77–89. https://doi.org/10.1016/j.enbuild.2017.04.038

Ajantha Devi, V., & Naved, M. (2021). Dive in Deep Learning. *Machine Learning in Signal Processing*, 97–126. https://doi.org/10.1201/9781003107026-5

Al-Qahtani, F. H., & Crone, S. F. (2013). Multivariate k-nearest neighbour regression for time series data - A novel algorithm for forecasting UK electricity demand. *Proceedings of the International Joint Conference on Neural Networks*. https://doi.org/10.1109/IJCNN.2013.6706742

Amber, K. P., Aslam, M. W., & Hussain, S. K. (2015). Electricity consumption forecasting models for administration buildings of the UK higher education sector. *Energy and Buildings*, *90*, 127–136. https://doi.org/10.1016/j.enbuild.2015.01.008

Andrić, I., Koc, M., & Al-Ghamdi, S. G. (2019). A review of climate change implications for built environment: Impacts, mitigation measures and associated challenges in developed and developing countries. In *Journal of Cleaner Production* (Vol. 211, pp. 83–102). Elsevier Ltd. https://doi.org/10.1016/j.jclepro.2018.11.128

Azar, E., & Menassa, C. C. (2012). A comprehensive analysis of the impact of occupancy parameters in energy simulation of office buildings. *Energy and Buildings*, *55*, 841–853. https://doi.org/10.1016/j.enbuild.2012.10.002

Card, S. K. (1999). Reviewing Readings in information Visualization: Using Vision to Think. *IEEE Multimedia*, *6*(4), 93. https://doi.org/10.1109/MMUL.1999.809241

Del Carmen Ruiz-Abellón, M., Gabaldón, A., & Guillamón, A. (2018). Load forecasting for a campus university using ensemble methods based on regression trees. *Energies*, *11*(8). https://doi.org/10.3390/en11082038

Fathi, S., & Srinivasan, R. (2019). *Climate Change Impacts on Campus Buildings Energy Use An AI-based Scenario Analysis*. https://doi.org/10.1145/3363459.3363540

Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep learning*. *29*(7553), 1–802.

Hochreiter, S., & Schmidhuber, J. (1997). Long Short-Term Memory. *Neural Computation*, *9*(8), 1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735

Kim, M. K., Kim, Y. S., & Srebric, J. (2020). Predictions of electricity consumption in a campus building using occupant rates and weather elements with sensitivity analysis: Artificial neural network vs. linear regression. *Sustainable Cities and Society*, *62*, 102385. https://doi.org/10.1016/j.scs.2020.102385

Kong, W., Dong, Z. Y., Jia, Y., Hill, D. J., Xu, Y., & Zhang, Y. (2019). Short-Term Residential Load Forecasting Based on LSTM Recurrent Neural Network. *IEEE Transactions on Smart Grid*, *10*(1), 841–851. https://doi.org/10.1109/TSG.2017.2753802

López, Ing. J. R. (2012). Trabajo Final presentado en opción al título de Máster en Gestión de Proyectos Informáticos. *Tesis*.

Martin, T. (1987). The Time Series Approach to Short-Term Load Forecasting. *IEEE Power Engineering Review*, *PER-7*(8), 56–57. https://doi.org/10.1109/MPER.1987.5527072

Mastronardi, L. J., Sfeir, A., & Sánchez, S. (2016). La temperatura y su influencia en la demanda de energía eléctrica : Un análisis regional para Argentina usando modelos

econométricos. *Secretaría de Planeamiento Energético Estratégico*, *November*, 0–21.

Müller, A. C., & Guido, S. (2015). Introduction to Machine Learning with Python and Scikit-Learn. In *O'Reilly Media, Inc.*

Oquendo, S. (2016). Universidad Politécnica Salesiana Sede Quito. *Tesis*, 1–63.

Peixeiro, M. (2022). *Time Series Forecasting in Python*.

Rodriguez, S. (Instituto de E.-F. de C. E. y de A. / U. D. L. R., & Massa, F. (Instituto de E.-F. de C. E. y de A. / U. D. L. R. (2014). *Predicción y estacionalidad intra diaria de la demanda de energía eléctrica en Uruguay*.

Ryu, S., Noh, J., & Kim, H. (2017). Deep neural network based demand side short term load forecasting. *Energies*, *10*(1), 1–20. https://doi.org/10.3390/en10010003

San Miguel Salas, J. (2016). *Desarrollo con matlab de una red neuronal para estimar la demanda de energía eléctrica*. 1–109.

Shi, H., Xu, M., & Li, R. (2018). Deep Learning for Household Load Forecasting-A Novel Pooling Deep RNN. *IEEE Transactions on Smart Grid*, *9*(5), 5271–5280. https://doi.org/10.1109/TSG.2017.2686012

Taylor, E. L. (2013). *Trace: Tennessee Research and Creative Exchange Short-term Electrical Load Forecasting for an Institutional/Industrial Power System Using an Artificial Neural Network*.

Torres, J. F., Hadjout, D., Sebaa, A., Martínez-Álvarez, F., & Troncoso, A. (2021). Deep Learning for Time Series Forecasting: A Survey. In *Big Data* (Vol. 9, Issue 1, pp. 3–21). https://doi.org/10.1089/big.2020.0159

Vaghefi, A., Jafari, M. A., Bisse, E., Lu, Y., & Brouwer, J. (2015). Modeling and forecasting of cooling and electricity load demand. *Applied Energy*, *136*, 186–196. https://doi.org/10.1016/j.apenergy.2014.09.004

Yuan, J., Farnham, C., Azuma, C., & Emura, K. (2018). Predictive artificial neural network models to forecast the seasonal hourly electricity consumption for a University

Campus. *Sustainable Cities and Society*, *42*(January), 82–92. https://doi.org/10.1016/j.scs.2018.06.019