



**PREDICCIÓN DEL PRECIO DE LA ENERGÍA ELÉCTRICA EN COLOMBIA CON
HORIZONTE DE MUY CORTO PLAZO UTILIZANDO MODELOS DE DEEP
LEARNING**

Estudiantes:

SEBASTIÁN GIRALDO ZULUAGA
EDWARD ALEJANDRO RAMÍREZ LONDOÑO

Monografía presentada para optar al título de Especialista en Analítica y Ciencia de Datos

Asesores:

PhD. WALTER MAURICIO VILLA
PhD. ÁLVARO JARAMILLO DUQUE

Universidad de Antioquia
Facultad de Ingeniería
Especialización en Analítica y Ciencia de Datos
Medellín, Antioquia, Colombia
2023

Cita	(Giraldo Zuluaga & Ramírez Londoño, 2023)
Referencia	Giraldo Zuluaga, S., & Ramírez Londoño, E. A. (2023). <i>Predicción del precio de la energía eléctrica en Colombia con horizonte de 24 horas utilizando modelos de Deep Learning</i> , Trabajo de grado especialización]. Universidad de Antioquia, Medellín, Colombia.
Estilo APA 7 (2020)	



Especialización en Analítica y Ciencia de Datos, Cohorte V.
 Centro de Investigación Ambientales y de Ingeniería (CIA).



Centro de Documentación Ingeniería (CENDOI)

Repositorio Institucional: <http://bibliotecadigital.udea.edu.co>

Universidad de Antioquia - www.udea.edu.co

Rector: John Jairo Arboleda Céspedes.

Decano: Julio Cesar Saldarriaga Molina

Jefe departamento: Diego José Luis Botia Valderrama

El contenido de esta obra corresponde al derecho de expresión de los autores y no compromete el pensamiento institucional de la Universidad de Antioquia ni desata su responsabilidad frente a terceros. Los autores asumen la responsabilidad por los derechos de autor y conexos.

Tabla de contenido

Resumen	9
Abstract	10
1. Descripción del problema	11
1.1. Problema de negocio	12
1.2. Aproximación desde la analítica de datos	13
1.3. Origen de los datos	13
1.4. Métricas de desempeño	13
2. Objetivos	15
2.1. Objetivo general	15
2.2. Objetivos específicos.....	15
3. Datos	16
3.1. Datos originales.....	16
3.2. Datsets	17
3.3. Analítica descriptiva.....	18
4. Proceso de analítica.....	31
4.1. Pipeline.....	31
4.2. Preprocesamiento	32
4.3. Modelos de Deep Learning implementados	34
DNN.....	34
LSTM.....	35
GRU	36
CNN-LSTM.....	37
4.4. Métricas.....	38
5. Metodología.....	40

- 5.1. Baseline40
- 5.2. Validación e Iteraciones42
 - Tipo de ventanado y cantidad de retardos.....42
 - Sintonización de hiperparámetros.....44
- 5.3 Herramientas45
- 6. Resultados47
 - 6.1. Métricas y gráficas47
 - 6.2. Consideraciones de producción.....55
- 7. Conclusiones56
- Referencias58

Lista de Tablas

<i>Tabla 3-1. Datos descargados y procesados</i>	<i>17</i>
<i>Tabla 4-1. Estructura modelo DNN.....</i>	<i>34</i>
<i>Tabla 4-2. Estructura modelo LSTM</i>	<i>35</i>
<i>Tabla 4-3. Estructura modelo GRU.....</i>	<i>36</i>
<i>Tabla 4-4. Estructura modelo CNN-LSTM.....</i>	<i>37</i>
<i>Tabla 5-1. Resultados con el precio de oferta de los generadores 72_24.....</i>	<i>40</i>
<i>Tabla 5-2. Resultados sin el precio de oferta de los generadores 72_24.....</i>	<i>40</i>
<i>Tabla 6-1. Métricas con 24 retardos</i>	<i>47</i>
<i>Tabla 6-2. Métricas con 48 retardos</i>	<i>48</i>
<i>Tabla 6-3. Métricas con 72 retardos</i>	<i>49</i>
<i>Tabla 6-4. Métricas con 96 retardos</i>	<i>50</i>
<i>Tabla 6-5. Hiperparámetros modelo DNN.....</i>	<i>52</i>
<i>Tabla 6-6. Hiperparámetros modelo LSTM.....</i>	<i>52</i>
<i>Tabla 6-7. Hiperparámetros modelo GRU</i>	<i>52</i>
<i>Tabla 6-8. Hiperparámetros modelo CNN-LSTM</i>	<i>53</i>
<i>Tabla 6-9. Métricas de desempeño con teniendo en cuenta los hiper parámetros optimizados ...</i>	<i>53</i>

Lista de Figuras

<i>Figura 3-1. Precio de Energía Eléctrica en bolsa entre 2013 y 2018.....</i>	<i>19</i>
<i>Figura 3-2. Demanda de Energía Eléctrica entre 2013 y 2018.....</i>	<i>20</i>
<i>Figura 3-3. Volumen útil de Energía Eléctrica entre 2013 y 2018.....</i>	<i>21</i>
<i>Figura 3-4. Aportes de Energía Térmica entre 2013 y 2018.....</i>	<i>22</i>
<i>Figura 3-5. Precio Oferta de Energía Eléctrica entre 2013 y 2018.....</i>	<i>23</i>
<i>Figura 3-6. Precio de oferta y bolsa de Energía Eléctrica entre 2013 y 2018.....</i>	<i>23</i>
<i>Figura 3-7. Disponibilidad de Energía Térmica entre 2013 y 2018.....</i>	<i>24</i>
<i>Figura 3-8. Disponibilidad de Energía No Térmica entre 2013 y 2018.....</i>	<i>25</i>
<i>Figura 3-9. Boxplot de las variables.....</i>	<i>26</i>
<i>Figura 3-10. Matriz de dispersión de las variables.....</i>	<i>27</i>
<i>Figura 3-11. Matriz de correlación de las variables.....</i>	<i>28</i>
<i>Figura 3-12. Matriz de covarianza de las variables.....</i>	<i>29</i>
<i>Figura 3-13. Autocorrelación del precio en bolsa.....</i>	<i>30</i>
<i>Figura 4-1. Flujograma de trabajo.....</i>	<i>32</i>
<i>Figura 4-2. Etiqueta de Demanda de energía eléctrica en Colombia.....</i>	<i>33</i>
<i>Figura 4-3. Demanda de energía eléctrica en Colombia sin tendencia.....</i>	<i>33</i>
<i>Figura 5-1. Curvas original y pronóstico del precio de bolsa considerando el precio de oferta.....</i>	<i>41</i>
<i>Figura 5-2. Curvas original y pronóstico del precio de bolsa sin considerar el precio de oferta.....</i>	<i>42</i>
<i>Figura 5-3. Ventanado con Traslape de 3 registros.....</i>	<i>43</i>
<i>Figura 5-4. Métrica MAE en el entrenamiento haciendo traslape de todos los registros -1.....</i>	<i>43</i>
<i>Figura 5-5. Ventanado sin traslape.....</i>	<i>43</i>
<i>Figura 5-6. Métrica MAE en el entrenamiento sin traslape.....</i>	<i>44</i>
<i>Figura 6-1. Precio de bolsa Original y Predicción con 24 retardos.....</i>	<i>48</i>
<i>Figura 6-2. Precio de bolsa Original y Predicción con 48 retardos.....</i>	<i>49</i>

Figura 6-3. Precio de bolsa Original y Predicción con 72 retardos50

Figura 6-4. Precio de bolsa Original y Predicción con 96 retardos51

Figura 6-5. Valores de pérdida durante el entrenamiento modelos con parámetros optimizados54

Figura 6-6. Valores de MAE durante el entrenamiento de los modelos con parámetros optimizados.....54

Figura 6-7. Precio de bolsa original y pronosticado de los modelos con los parámetros optimizados.....55

Siglas, acrónimos y abreviaturas

DNN	Dense Neural Network
LSTM	Long Short-Term Memory
GRU	Gated Recurrent Unit
CNN	Convolutional Neural Network
MAE	Mean Absolute Error
MAPE	Mean Absolute Percentage Error
SMAPE	Symmetric Mean Absolute Percentage Error
RMSE	Root Mean Square Error
XM	Administrador y Operador del Mercado Eléctrico Colombiano
VAR	Vector AutoRegressive
ARIMA	AutoRegressive Integrated Moving Average
SARIMA	Seasonal AutoRegressive Integrated Moving Average
ARX	AutoRegressive with Exogenous Inputs
GARCH	Generalized AutoRegressive Conditional Heteroskedasticity
LEAR	Lasso Estimated AutoRegressive
SVM	Support Vector Machine
RF	Random Forest

Resumen

Este trabajo tiene como objetivo la implementación de diferentes modelos de Deep Learning aplicados a la predicción del precio de bolsa en el mercado eléctrico colombiano con un horizonte de 24 horas (corto plazo). Se entrenaron y se ajustaron modelos como: DNN, LSTM, GRU, CNN-LSTM, permitiendo hacer comparaciones entre los pronósticos usando métricas de desempeño como MAE, MAPE, SMAPE, RMSE. Los pronósticos de precio son importantes para los actores participantes en un mercado desregulado dada la alta volatilidad que presenta. Un buen pronóstico del precio de la electricidad en la bolsa de energía eléctrica da la posibilidad a los agentes participantes del mercado de ajustar sus ofertas para mitigar los riesgos y maximizar beneficios. Los modelos fueron entrenados a partir de la información recopilada en la base de datos de XM donde se obtuvieron las variables de: precio de bolsa, demanda de energía, volumen útil de embalses, Aportes hidrológicos, disponibilidad plantas térmicas, disponibilidad de las plantas no térmicas y precio de oferta de cada planta.

En la primera iteración se ejecutaron los modelos con un horizonte de predicción de 24 periodos, se consideraron diferente cantidad de retardos (24, 48, 72, 96) para realizar la predicción, obteniendo el menor error con 72 retardos de tiempo, con una resolución temporal horaria para la variable que se pronóstica. Luego se realizó la sintonización de los hiperparámetros de cada modelo de aprendizaje de máquina utilizado, realizando una predicción de 24 horas a futuro usando 72 retardos de tiempo, es decir, 72 horas con la información de variable a predecir y variables exógenas, con los mejores hiperparámetros encontrados en cada modelo. Los mejores resultados en las métricas de desempeño se obtuvieron en el modelo LSTM donde para la métrica MAPE y RMSE en los datos de pruebas fue de 6.93% y 11.87 \$COP/kWh respectivamente, siendo 7.13% menor que lo obtenido con DNN, 8.84% menor que lo obtenido con GRU y 54.24% menor que lo obtenido con la CNN-LSTM para la métrica MAPE.

Palabras clave: Precio de bolsa, Deep learning, mercado colombiano, pronóstico de corto plazo

<https://github.com/EdwarARL/MonografiaEAD>

<https://github.com/Sebasgiraldozuluaga/MonografiaEAD>

Abstract

This work aims to implement different Deep Learning models applied to the prediction of stock prices in the Colombian electrical market with a 24-hour horizon (short term). Models such as DNN, LSTM, GRU, CNN-LSTM were trained and adjusted, enabling comparisons between forecasts using performance metrics such as MAE, MAPE, SMAPE, RMSE. Electricity Price forecasts are crucial for participants in a deregulated market given its high volatility. Accurate electricity price forecasting in the electrical energy market allows market participants to adjust their offers to mitigate risks and maximize profits.

The models were trained using information collected from the XM database, including variables such as stock price, energy demand, reservoir capacity, hydrological contributions, availability of thermal plants, availability of non-thermal plants, and price offered by each plant.

In the first iteration, the models were executed with a 24-period prediction horizon, considering different delays (24, 48, 72, 96) to make predictions. The lowest error was obtained with a 72-hour time delay, with an hourly time resolution for the predicted variable. Hyperparameter tuning was then performed for each machine learning model used, making a 24-hour future prediction using a 72-hour time delay, meaning 72 hours of information for the variable to be predicted and exogenous variables, with the best hyperparameters found for each model. The best results in performance metrics were obtained in the LSTM model, where for the MAPE and RMSE metrics on test data, they were 6.93% and 11.87 \$COP/kWh respectively. These values were 7.13% lower than those obtained with DNN, 8.84% lower than with GRU, and 54.24% lower than with CNN-LSTM for the MAPE metric.

Keywords: Stock Prices, Deep Learning, Colombian Market, Short Term forecasting

1. Descripción del problema

El pronóstico del precio de la energía eléctrica en el mercado mayorista es un desafío importante para la planificación y la gestión de los recursos energéticos de un país. El sistema eléctrico colombiano se caracteriza por una alta participación de fuentes hidráulicas, lo que lo hace vulnerable a las variaciones climáticas y a los fenómenos de El Niño y La Niña [1]–[3]. Adicionalmente, esta variabilidad del precio también se puede ver afectada por aspectos estratégicos de los agentes los cuales buscan minimizar los riesgos y maximizar los beneficios. Lo anterior toma mayor relevancia teniendo en cuenta que el bien de la energía en Colombia tiene pocos generadores (Oferta) y muchos consumidores (Demanda), los cuales, su mayoría no tienen poder de mercado implicando que el mercado sea poco competitivo y que los determinantes en la formación del precio de la energía no reflejen exactamente la disposición a comprar o vender energía [4]. Este aspecto y a la alta volatilidad del precio en periodos secos por la alta dependencia de las fuentes hídricas (68% de la matriz energética), hacen que el pronóstico del precio en el mercado eléctrico colombiano sea un desafío [5].

El pronóstico del precio de bolsa en Colombia ha sido objeto de varias investigaciones desde hace varios años, donde se han utilizado diferentes técnicas basadas en métodos tanto estadísticos como de redes neuronales. En [1] utilizan modelos VAR para la estimación del precio de bolsa, mientras que en [5] utilizan modelos ARIMA, SARIMA y ARX. En [6] los autores usaron redes neuronales difusas cuyos resultados fueron comparados con un modelo GARCH donde incluyeron como entrada series de precios diarios y el nivel medio de los embalses. En los últimos años han surgido investigaciones donde incluyen técnicas de Machine Learning y Deep Learning que comparan los resultados entre ellas y con métodos estadísticos. En [7] proponen un marco de referencia para el pronóstico de precios usando 4 técnicas de Deep Learning como: DNN, híbrida LSTM-DNN, híbrida GRU-DNN y un modelo CNN, estas son a su vez comparadas con métodos estadísticos. Los análisis realizados encuentran que el Deep Learning tiene mayor exactitud, es decir, menor error en los pronósticos realizados. Se resaltan que los modelos LSTM y GRU son una buena alternativa para el problema planteado, donde se tiene modelos complejos no lineales con series de tiempo. También que las redes neuronales profundas, donde se tiene varias capas, pueden compensar redes de pocas capas, pero con muchas neuronas; por lo tanto, las redes profundas pueden llegar a ser menos complejas. Los mismos autores en [8] realizaron una revisión de las mejores prácticas y algoritmos, brindando un marco de referencia desde el cual las diferentes

investigaciones que se realicen pueden partir y comparar resultados usando datos de mercados europeos como el de Bélgica, Francia y Alemania, PJM y Noor Pool, con una ventana de tiempo de 6 años utilizando 4 años para entrenamiento y 2 años para prueba. Se propone principalmente un modelo estadístico LEAR, modelo de Deep Learning DNN con 1, 2, 3, 4 y 5 capas y modelos híbridos. Se utilizan varias métricas de desempeño de las cuales 4 son usadas para el presente trabajo. Los modelos de Machine Learning como SVM y RF también están siendo utilizados para pronóstico de precio en [9] los cuales son comparados usando los datos y modelos presentados en [8]. Finalmente, en [10] y [11] muestran casos de aplicación de pronóstico de precio para casos del mercado de Alemania y modelos para obtener la mejor estrategia de inversión en el mercado ibérico respectivamente.

En este trabajo se propone una metodología basada en técnicas de Deep Learning para estimar el precio de la electricidad con un horizonte de 24 horas, utilizando variables de entrada como: precio en bolsa, volumen útil, aportes de energía, precio oferta, demanda de energía, disponibilidad de energía térmica, disponibilidad de energía no térmicas y se agrega una etiqueta creada con la técnica de machine learning de clasificación KMEANS para determinar el tipo de día según el pronóstico de demanda, donde se clasificó principalmente con 3 etiquetas que denotan el tipo de día Ordinario (Lunes, Martes, Miércoles, Jueves y Viernes), Sábado. Y Festivo (incluye Domingos y días Sábado u ordinario que sean festivos). Se emplean diferentes arquitecturas de redes neuronales profundas como: redes recurrentes LSTM y GRU, redes densas DNN y un híbrido con CNN-LSTM. Los experimentos se realizaron con datos históricos del mercado eléctrico colombiano descargados con la API de XM para un horizonte de 6 años (2013-2018), se muestra que la metodología propuesta permite encontrar modelos que mejoran al hacer combinación de los retardos utilizados, diferentes arquitecturas y tipos de redes neuronales y que la hiperparametrización también es un factor importante a la hora de mejorar las predicciones.

1.1. Problema de negocio

Crear un modelo que pueda predecir el precio de la energía eléctrica con un horizonte de 24 horas, utilizando datos históricos y variables exógenas. La predicción del precio de la energía eléctrica puede beneficiar a los participantes del mercado eléctrico, tanto a los generadores como a comercializadores o grandes consumidores ya que les permite optimizar sus estrategias de producción, suministro y fijación de precios.

1.2. Aproximación desde la analítica de datos

Los modelos predictivos son herramientas que permiten estimar el comportamiento futuro de una variable de interés, basándose en datos pasados y presentes. En este caso, se busca predecir el precio de bolsa de la energía eléctrica, la cual es una variable compleja y dinámica que depende de múltiples factores. Para ello, se utilizarán técnicas de análisis exploratorio de datos, selección de variables, correlación de variables, escalado de valores, identificación de valores atípicos y faltantes. Se compararán diferentes modelos de Deep Learning, para elegir el que mejor se ajuste a los datos y al problema planteado.

1.3. Origen de los datos

Los datos utilizados en este proyecto fueron descargados de la base de datos de XM donde está disponible la información del mercado eléctrico de energía mayorista colombiano. Se trabajó con una ventana de tiempo que va desde el año 2013 hasta el 2018. Estos datos se obtienen a partir de una API proporcionada por la empresa XM llamada SINERGOX, que ofrece la información de consulta para extraer la información a partir de Python. Los datos descargados fueron precio de bolsa de energía, demanda de energía, volumen útil de los embalses, aportes hidrológicos, precio oferta de las plantas, disponibilidad de energía de las plantas térmicas y disponibilidad de energía de las plantas no térmicas. Estos datos son relevantes para analizar la dinámica del mercado eléctrico, los cuales se consideran que son los factores que influyen en la formación de precio de bolsa.

1.4. Métricas de desempeño

Las métricas de desempeño son una herramienta fundamental para evaluar la calidad y la eficacia de un modelo de machine learning. Estas métricas permiten medir el grado de ajuste entre las predicciones del modelo y los datos reales, así como identificar posibles fuentes de error o sesgo. En el caso de la predicción del precio de la energía eléctrica en el mercado colombiano, las métricas de desempeño son especialmente relevantes, ya que se trata de un problema complejo y dinámico, con múltiples factores que influyen en la oferta y la demanda. Algunas de las métricas de desempeño más utilizadas en este tipo de problemas de acuerdo con la revisión bibliográfica realizada son MAE, MAPE, SMAPE y RMSE.

- **MAE:** Mean Absolute Error, que mide la diferencia promedio entre el valor real y el valor predicho, sin tener en cuenta el signo. Es una medida de precisión que penaliza por igual los errores positivos y negativos [8].
- **MAPE:** Mean Absolute Percentage Error, que mide el error relativo entre el valor real y el valor predicho, expresado como un porcentaje. Es una medida de precisión que se utiliza ampliamente en la literatura, pero tiene algunas limitaciones, como el hecho de que no está definida cuando el valor real es cero o muy pequeño, y que puede dar valores muy altos cuando el error es pequeño [8].
- **SMAPE:** Symmetric Mean Absolute Percentage Error, que mide el error relativo entre el valor real y el valor predicho, expresado como un porcentaje. Es una medida de precisión que tiene en cuenta la escala de los datos y evita el problema del MAPE cuando el valor real es cero o muy pequeño [8].
- **RMSE:** Root Mean Square Error, es la raíz cuadrada del promedio de errores cuadrados. El efecto de cada error en el RMSE es proporcional al tamaño del error cuadrado; por lo tanto, los errores mayores tienen un efecto desproporcionadamente grande en la RMSE siendo es sensible a los valores atípicos [8].

2. Objetivos

2.1.Objetivo general

Implementar algoritmos de Deep Learning para pronosticar el precio de la electricidad en el mercado eléctrico colombiano con un horizonte de 24 horas, esto permitirá a los agentes de mercado disminuir riesgos y maximizar beneficios en el momento de hacer la oferta de precios.

2.2.Objetivos específicos

- Realizar una búsqueda bibliográfica que permite tener mayor contexto y conocimiento de lo desarrollado hasta el momento respecto al pronóstico de precios, identificando aspectos como variables importantes para la predicción del precio, arquitecturas y forma de ingresar la información para el entrenamiento de las redes neuronales.
- Desarrollar un módulo que permite preparar la base de datos para el entrenamiento de las redes neuronales, se incluirán funciones para descargar, limpiar y procesar los datos con las variables utilizadas para el desarrollo de los modelos de pronóstico.
- Implementar diferentes algoritmos DL como DNN, LSTM, GRU y CNN-LSTM, para la predicción de los precios.
- Evaluar el desempeño de los modelos de DL usando métricas de desempeño consideradas en la literatura técnica para determinar la calidad del pronóstico de cada modelo como MAE, MAPE, SMAPE y RMSE.

3. Datos

3.1. Datos originales

Los datos originales se componen de una tabla con 52.584 filas y 8 columnas. Cada fila representa un registro único horario de las diferentes variables influyentes en la predicción del precio de energía en bolsa en el mercado eléctrico colombiano en el periodo comprendido entre 2013 y 2018. La base de datos (Dataset) original no presenta datos faltantes, sin embargo, presenta datos atípicos explicados por el fenómeno del Niño y la Niña. Estos datos no fueron eliminados dado que son valores que se van a presentar en la realidad debido a las condiciones climatológicas que se dan en el país y se requiere que los diferentes modelos desarrollados aprendan y puedan hacer una predicción cuando se presenten este tipo de escenarios.

Formato de los Datos: Los datos están en formato de archivo CSV, y se obtuvieron por medio de SINERGOX una API proporcionada por XM la cual tiene conexión a través de Python y de la cual se pueden encontrar ejemplos de uso [12].

Distribución de los Datos: Los datos están disponibles en un solo archivo CSV. Cada fila del archivo representa una día y hora específico, y las columnas representan las diferentes variables de manera horario. Los aportes, el volumen útil y la oferta de precio de las planas (única para los 24 periodos del día) originalmente se descargaron de manera diaria, sin embargo, estos datos se replicaron de manera horaria para completar el Dataset.

Date: Representa la fecha y hora de cada entrada de datos.

PrecioB: Precio de energía eléctrica en bolsa dado en \$COP/kWh.

Demanda: Demanda real del sistema eléctrico colombiano dado en kWh.

DispNoTer: Disponibilidad de las plantas de generación no térmicas dado en kWh.

VolÚtil: Volumen Útil diario Energía por Sistema dado en kWh.

Aportes: Aportes en energía de los ríos que aportan agua a algún embalse del sistema interconectado nacional dado en kWh.

DispTer: Disponibilidad de las plantas de generación térmicas dado en kWh.

PrecioO: Precio de oferta de las plantas dade en \$COP/kWh.

Modo de Acceso a los Datos: La API SINERGOX de XM es una herramienta de acceso público que permite consultar algunas de las variables del Mercado de Energía Mayorista (MEM). Para interactuar con la API se usó un Jupyter Notebook haciendo uso de Python para transformar los resultados de las peticiones en un archivo CSV.

Los datos están contenidos en un archivo al cual se puede acceder con cualquier programa que pueda leer archivos CSV, y a partir de este archivo CSV se hizo un análisis exploratorio y se implementaron los diferentes modelos de Deep Learning.

A continuación, en la Tabla 3-1 se observa una muestra de los datos descargados y procesados.

Tabla 3-1. Datos descargados y procesados

Date	PrecioB	Demanda	DispTer	DispNoTer	VolUtil	Aportes	PrecioO
1/01/2013 0:00	168,39165	5635708,86	4312666	8609000	1,1155E+10	64981500	349,839712
1/01/2013 1:00	168,39165	5398892,95	4245000	8609000	1,1155E+10	64981500	349,839712
1/01/2013 2:00	170,55065	5132168,81	4245000	8609000	1,1155E+10	64981500	349,839712
1/01/2013 3:00	150,55165	4886320,69	4241500	8609000	1,1155E+10	64981500	349,839712
1/01/2013 4:00	147,01865	4735282,07	4222000	8609000	1,1155E+10	64981500	349,839712
1/01/2013 5:00	145,05165	4600366,1	4220166	8609000	1,1155E+10	64981500	349,839712
1/01/2013 6:00	145,05165	4191281,5	4096000	8614533	1,1155E+10	64981500	349,839712

3.2. Datasets

Para predecir con un horizonte de 24 periodos, se dividen los datos en conjuntos de entrenamiento, validación y test, donde cada matriz debe ser organizada como un tensor (3 dimensiones) para luego usar modelos de Deep Learning que pueda capturar las tendencias y patrones de las variables con el objetivo de predecir el precio de la energía eléctrica en bolsa del mercado colombiano.

Para realizar el procesamiento de los datos se realizan varios pasos:

Preprocesamiento de los datos: Para la predicción del precio de energía en bolsa se está trabajando con datos de series temporales, donde se requiere convertir los datos en una secuencia de ventanas de tiempo organizados en forma de tensor teniendo en cuenta que los datos de entrada deben tener la estructura (n_registros, n_steps, n_var_in) y los datos de salida deben tener la estructura (n_registros, n_horizon, n_var_out), donde:

n_registros: Número de registros con los cuales se va a entrenar, validar o probar el modelo

n_steps	Número de retardos a considerar
n_var_in	Número de variables a considerar en la entrada
n_horizon	Número periodos a predecir
n_var_out	Número de variables a considerar en la salida

División de los datos: Para el trabajo de series temporales se realiza una división estándar teniendo en cuenta que el último año de datos se utiliza para pruebas (8760 registros), de los datos restante, el 80% se utiliza para entrenamiento y un 20% para validación. Es importante que esta división se haga de manera cronológica, es decir, el primer 80% de los datos para el entrenamiento y el último 20% para validación.

Además de la división estándar, se aplica un concepto conocido como "ventaneo". En el contexto de los datos de series temporales, el ventaneo se refiere a la creación de "ventanas" de datos consecutivos. Cada ventana se utiliza como una muestra individual para el entrenamiento del modelo. En este caso, cada ventana de entrada tiene un tamaño de n_steps que se varía entre 96, 72, 48 y 24 horas para validar inicialmente cual es el que mejor resultados arroja. Además, cada ventana de salida tiene un tamaño n_horizon de 24 horas que es el número de pasos de tiempo en el futuro que el modelo debe predecir.

El particionamiento de los datos que se hace utilizando técnicas de división y ventaneo es de vital importancia cuando se tiene un problema de series temporales para el entrenamiento de los modelos. A partir de este particionamiento se pueden implementar las redes neuronales recurrentes para la predicción del precio de energía en bolsa en un horizonte de 24 horas.

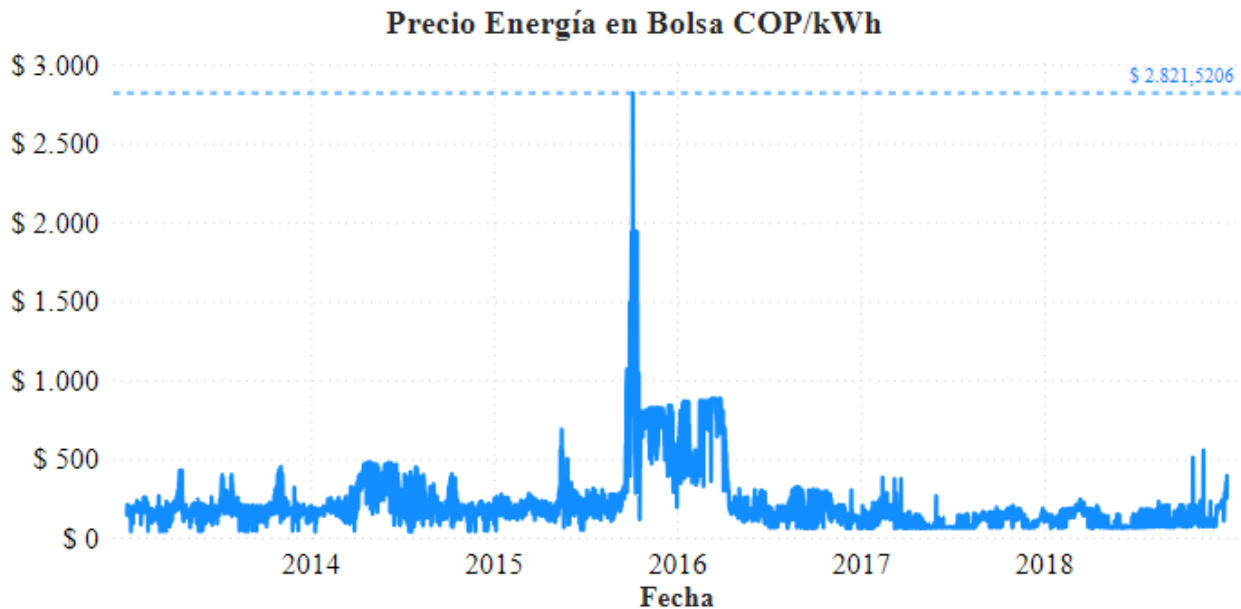
3.3. Analítica descriptiva

A continuación, se presentan una serie de figuras que muestran el comportamiento de las variables entre los años 2013 y 2018.

En la Figura 3-1 se presenta la variación de precios de energía eléctrica entre el 2013 y 2018, presentando sus valores máximos alrededor del año 2016. Se observa que el precio de energía en bolsa ha alcanzado un máximo histórico de 2821,5206 \$COP/kWh. Este aumento se debió principalmente a la escasez de lluvias y la menor disponibilidad de generación hidroeléctrica que obligó a recurrir a fuentes térmicas más costosas. A partir de 2016, el precio de energía eléctrica

empezó a disminuir gradualmente, debido a la recuperación de los embalses y la entrada en operación de nuevas plantas de generación.

Figura 3-1. Precio de Energía Eléctrica en bolsa entre 2013 y 2018

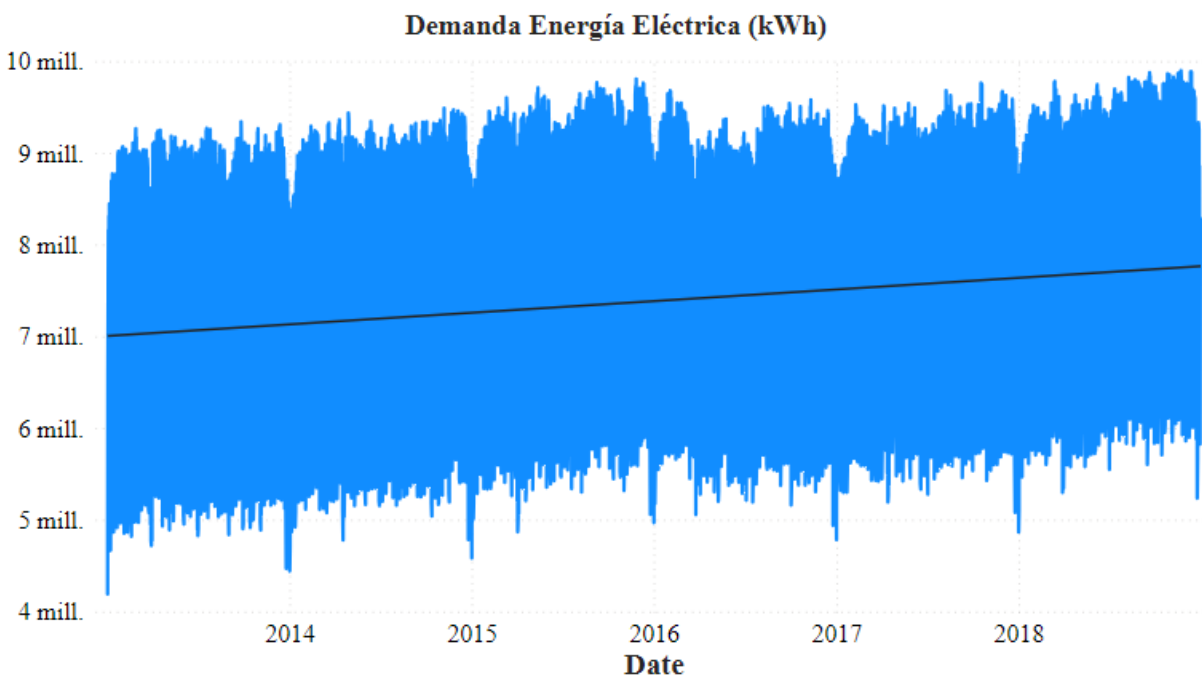


La demanda de energía eléctrica en Colombia entre 2013 y 2018 se presenta en la Figura 3-2, se observa una tendencia estacionaria con un incremento gradual. Esto significa que, aunque la demanda total de electricidad ha permanecido relativamente constante, ha habido un ligero aumento a lo largo de los años.

Este comportamiento puede ser el resultado de varios factores, incluyendo el crecimiento económico, los cambios en la población y las políticas de eficiencia energética. Por ejemplo, a medida que la economía de Colombia crece, es probable que aumente la demanda de electricidad para alimentar las industrias y los hogares. Además, el crecimiento de la población también puede llevar a un mayor uso de la electricidad.

Por otro lado, las políticas de eficiencia energética pueden haber ayudado a mantener la demanda de electricidad relativamente constante, a pesar del crecimiento económico y demográfico. Estas políticas pueden incluir incentivos para el uso de electrodomésticos eficientes, normas de construcción que promueven la eficiencia energética y programas de educación pública para fomentar el ahorro de energía.

Figura 3-2. Demanda de Energía Eléctrica entre 2013 y 2018

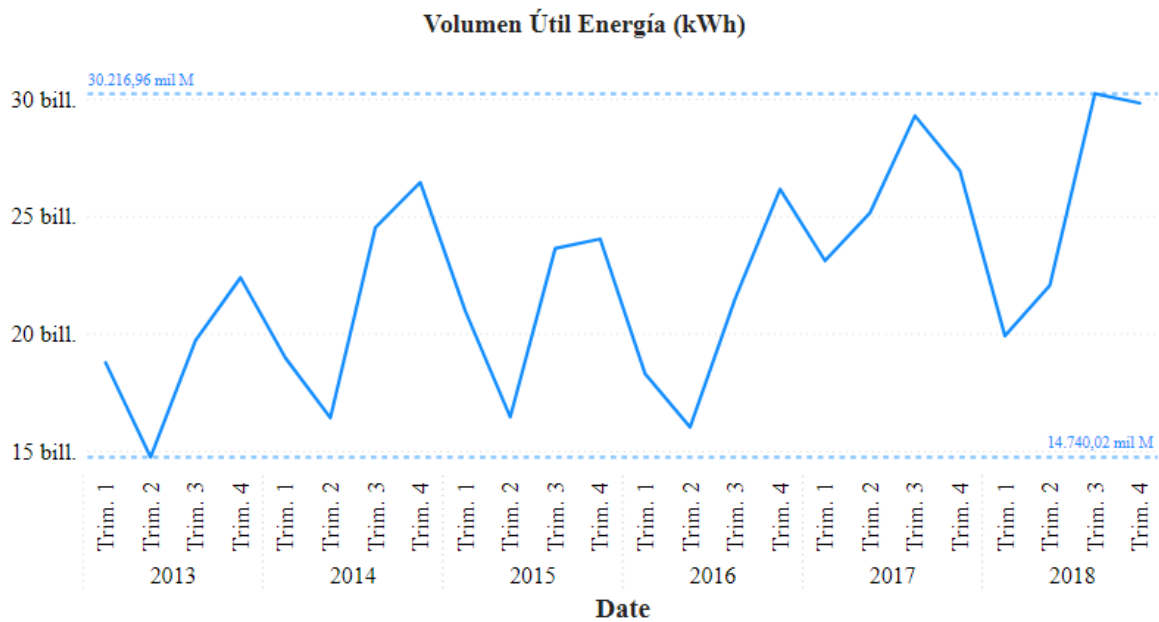


La Figura 3-3 muestra el volumen útil de los embalses en Colombia desde 2013 hasta 2018. Se puede observar que hay un aumento general en el volumen útil a lo largo del tiempo.

Puntos más altos: Se encuentran en 2018 con 30.1169 mil millones de kWh. Estos puntos máximos coinciden con los trimestres del año donde se presenta mayor cantidad de lluvias.

Puntos más bajos: Se encuentran en 2014 con 14.7407 mil millones de kWh. Estas variaciones están relacionadas con factores como el fenómeno de El Niño, que puede afectar la generación de energía hidroeléctrica, una fuente importante de energía en Colombia. Durante los años de El Niño, se presenta disminución de los niveles de los embalses, lo que a su vez podría haber afectado la cantidad de energía eléctrica que se pudo generar.

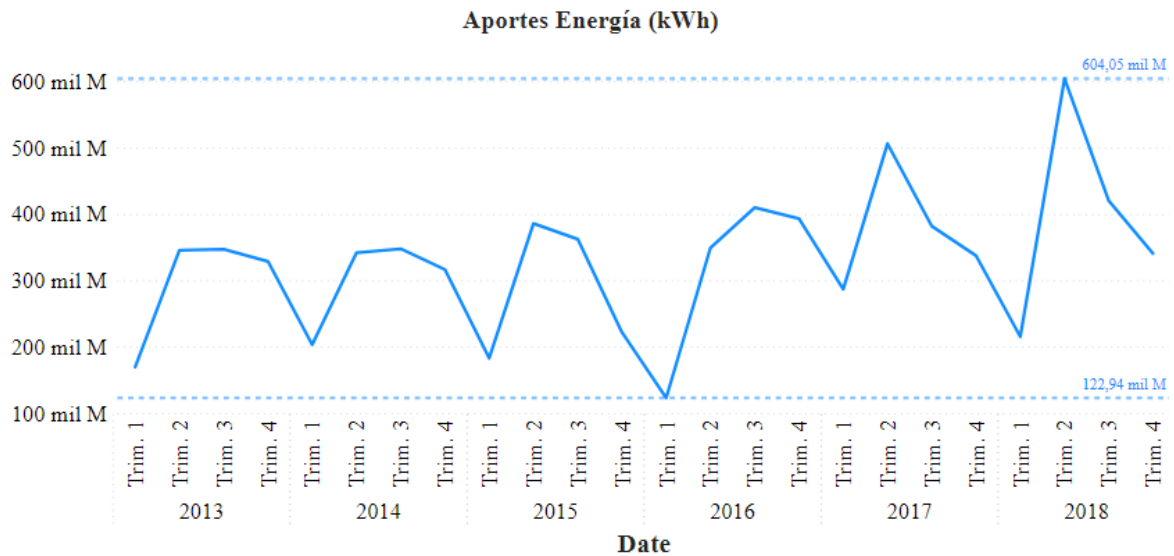
Figura 3-3. Volumen útil de Energía Eléctrica entre 2013 y 2018



Los ríos juegan un papel crucial en la generación de energía eléctrica en Colombia, especialmente a través de la energía hidroeléctrica. De hecho, la energía hidroeléctrica representa el 68% de la generación eléctrica de Colombia [13].

Sin embargo, es importante tener en cuenta que los fenómenos climáticos como El Niño pueden afectar la disponibilidad de agua en los ríos y, por lo tanto, la generación de energía hidroeléctrica. Por ejemplo, durante los períodos de sequía, el flujo de agua en los ríos puede disminuir, lo que puede reducir la capacidad de las plantas hidroeléctricas para generar electricidad. Esto podría explicar el descenso en los aportes de energía eléctrica por los ríos en 2016 que se muestra en la Figura 3-4.

Figura 3-4. Aportes de Energía Térmica entre 2013 y 2018



En la Figura 3-5 se muestra el precio de oferta de energía en la bolsa de energía en Colombia entre 2013 y 2018. Se presenta picos y valles en los precios ofertados.

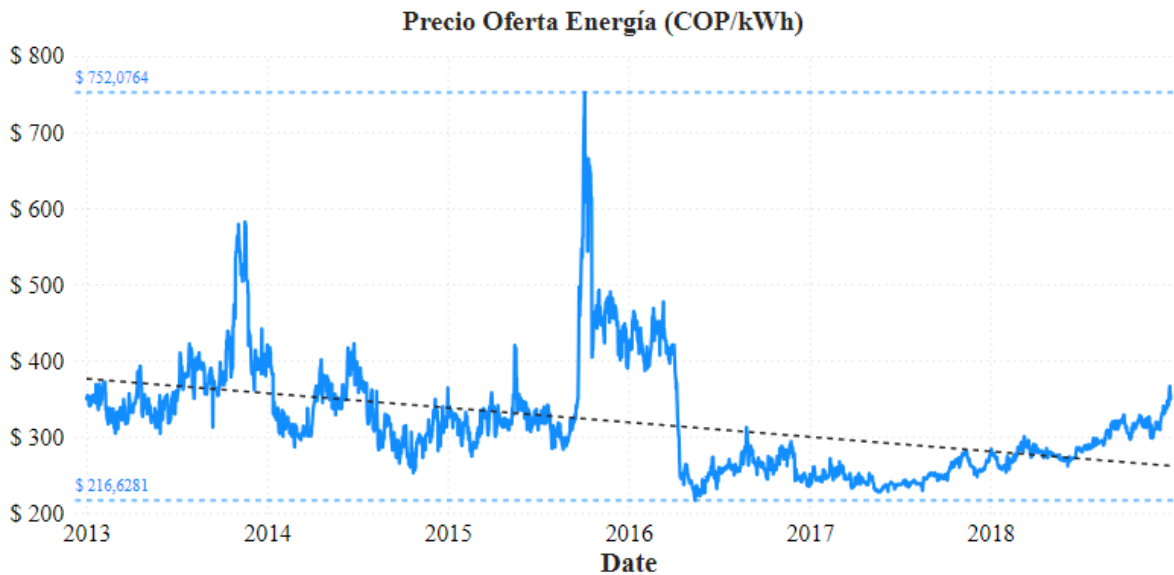
Picos: Se presentan en períodos de baja oferta de energía o la disminución del recurso hídrico, lo que puede aumentar el precio de oferta.

Valles: Se pueden presentar en períodos de alta oferta de energía o el aumento del recurso hídrico, lo que puede disminuir el precio de oferta.

Estas variaciones pueden estar influenciadas por varios factores, incluyendo las condiciones climáticas (como el fenómeno de El Niño), las políticas energéticas, los cambios en la economía y la población.

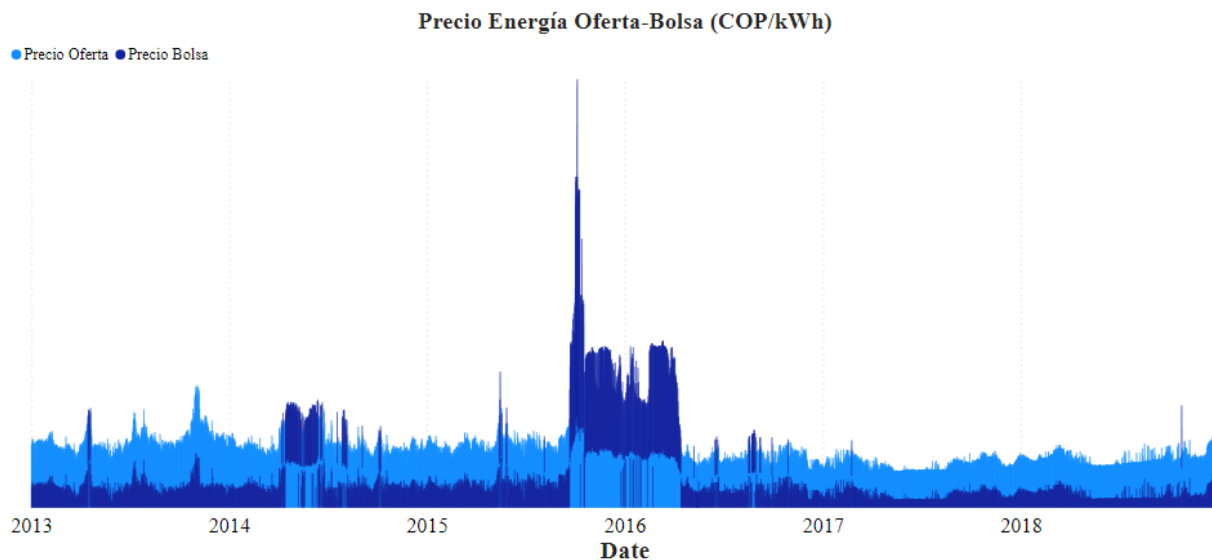
Es importante señalar que el promedio se obtuvo como el promedio horario del precio ofertado por cada uno de los agentes, lo que implica que se consideraron las ofertas de todos los agentes en cada hora del día.

Figura 3-5. Precio Oferta de Energía Eléctrica entre 2013 y 2018



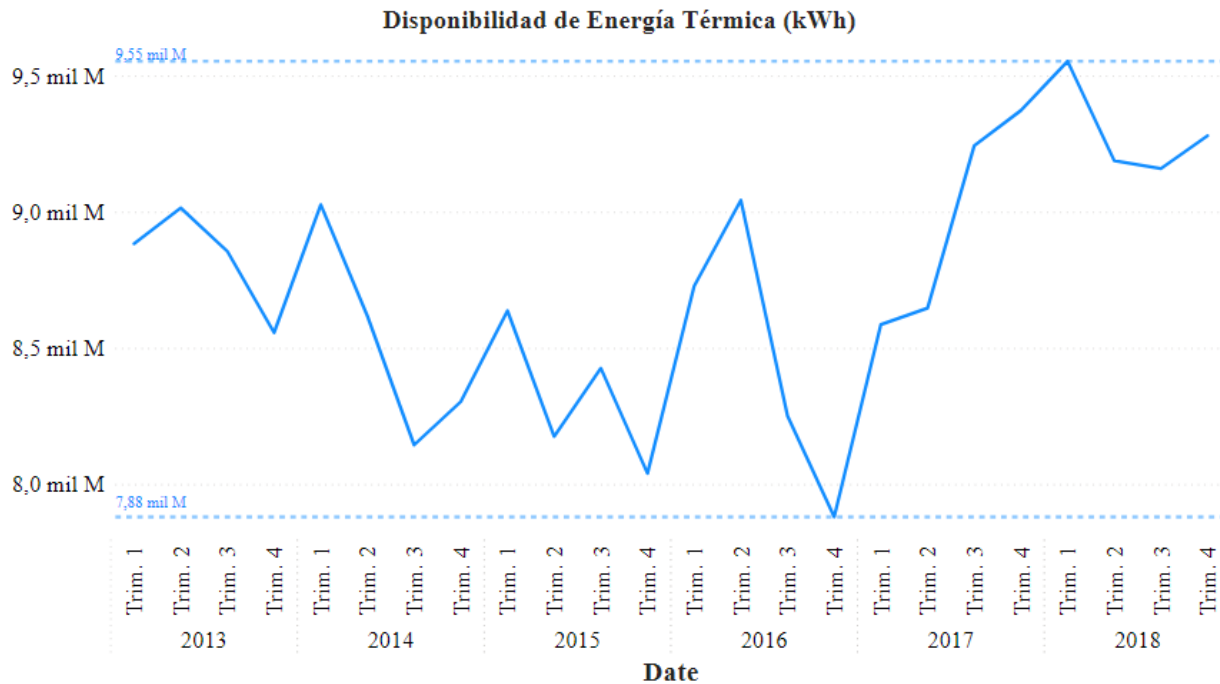
En la Figura 3-6 se presenta la variación de precios de energía eléctrica entre el 2013-2018, presentando sus valores máximos alrededor del año 2016, se observa que el precio de energía en bolsa y el precio ofertado por los agentes presenta una gran correlación. Se plantea un análisis exploratorio para analizar la correlación de ambas variables.

Figura 3-6. Precio de oferta y bolsa de Energía Eléctrica entre 2013 y 2018



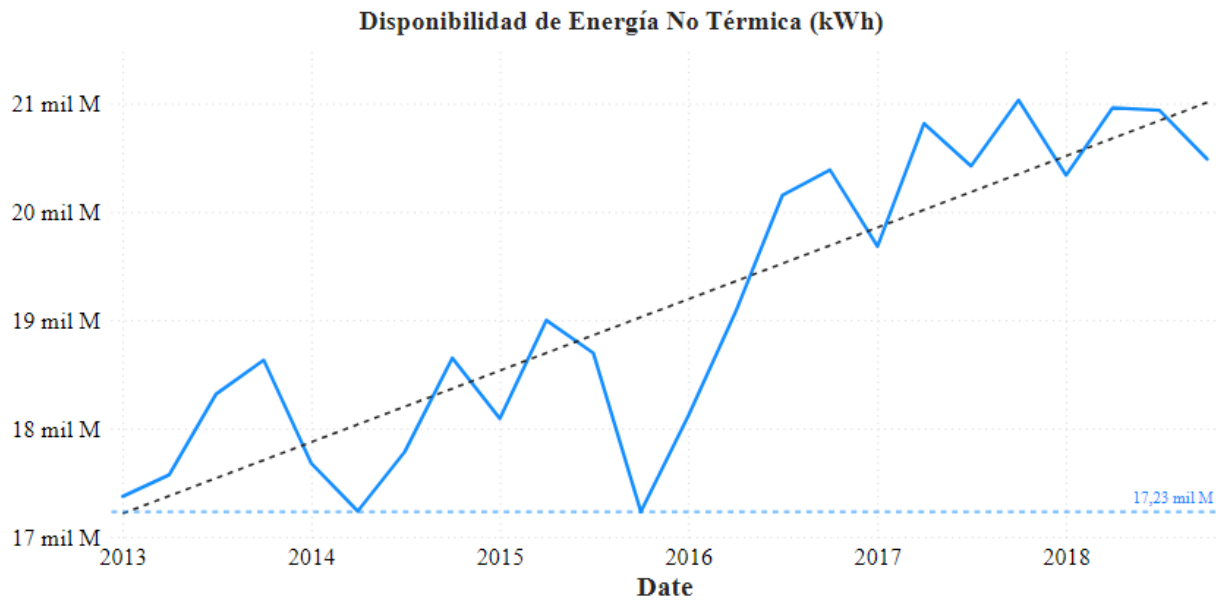
La Figura 3-7 muestra la disponibilidad de energía térmica (kWh) desde 2013 hasta 2018. La disponibilidad de energía térmica fluctúa entre 8 y 9.5 mil millones de kWh durante este período. Presentando su pico en las cercanías del año 2016 donde se presentó el costo más alto de energía eléctrica en Colombia como se evidenció en la Figura 3-1, indicando que en periodos secos este tipo de plantas entran a respaldar la generación en el país.

Figura 3-7. Disponibilidad de Energía Térmica entre 2013 y 2018



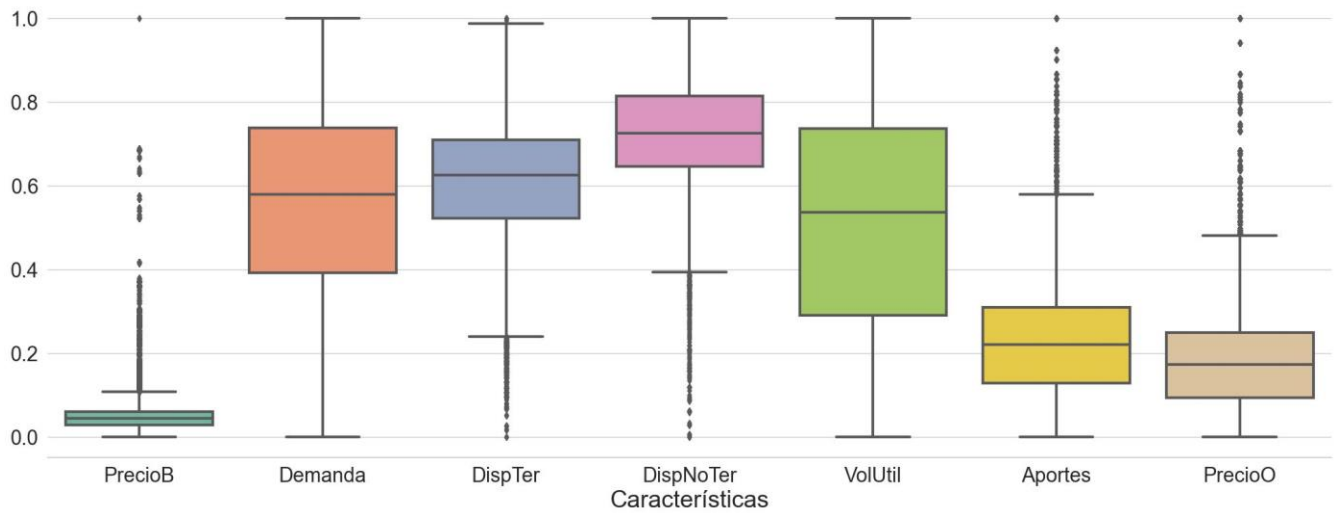
La Figura 3-8 muestra la disponibilidad de energía no térmica (kWh) en el sistema eléctrico colombiano desde 2013 hasta 2018. La disponibilidad de energía no térmica ha aumentado sostenidamente en el tiempo dada la entrada de los proyectos de generación por el plan de expansión. Se tiene un descenso en 2016 explicado por el fenómeno de El Niño presentado en este año, que también afectó fuertemente el pico del precio de energía eléctrica en bolsa. Es importante destacar que la principal fuente de generación de electricidad en Colombia es la energía hidroeléctrica. Esto significa que la mayor parte de la energía representada en este gráfico probablemente proviene de fuentes hidroeléctricas. La energía hidroeléctrica es una forma de energía renovable que utiliza el flujo de agua para generar electricidad. La sequía y la falta de lluvias pueden reducir el flujo de agua en las represas hidroeléctricas, lo que a su vez puede disminuir la capacidad de estas plantas para generar electricidad.

Figura 3-8. Disponibilidad de Energía No Térmica entre 2013 y 2018



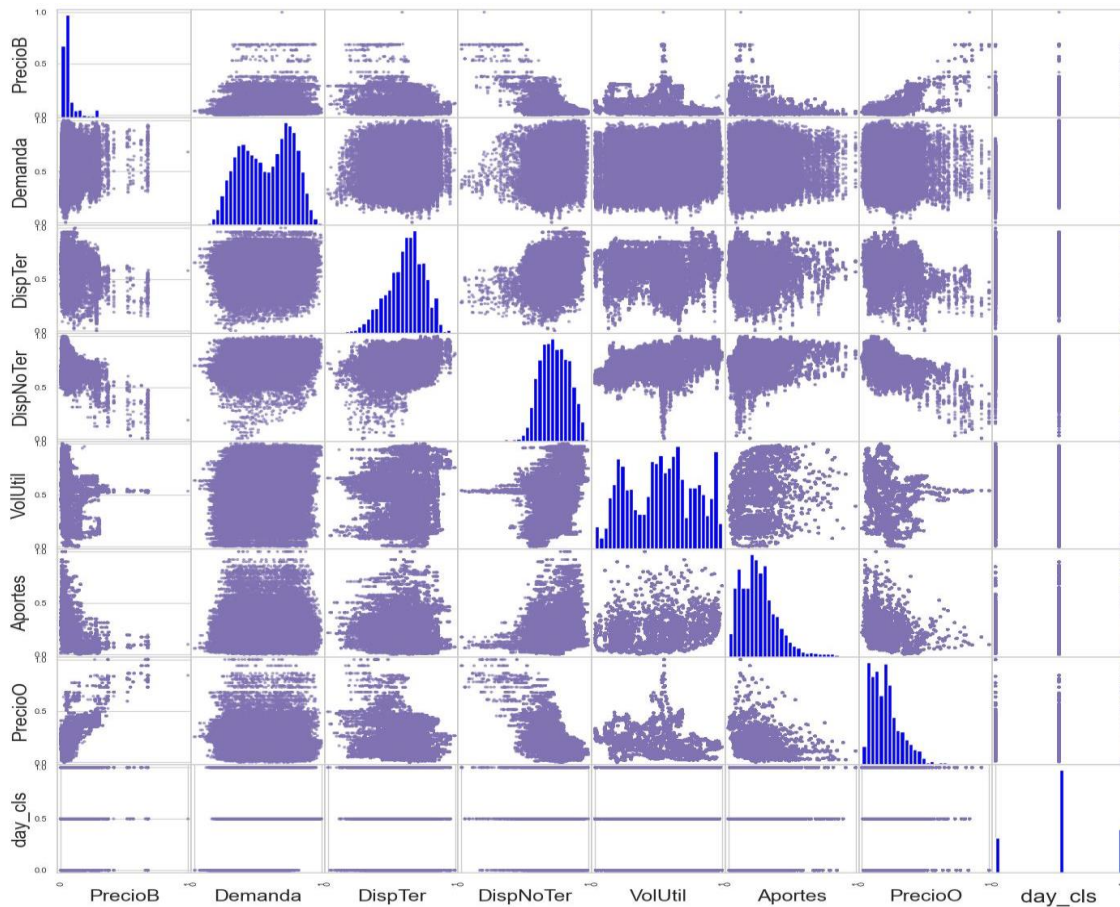
El boxplot de la Figura 3-9, proporciona una visión general de las tendencias y patrones en los datos, pero es importante tener en cuenta que puede haber otros factores y variables no representados en este gráfico que también pueden influir en estos resultados. En estos datos podemos evidenciar la cantidad de datos atípicos en el Dataset escalado entre 0 y 1. Se observa que las variables de demanda y volumen útil no presentan datos atípicos, sin embargo, las demás variables presentan datos atípicos, siendo el precio de bolsa la variable con valores más alejados respecto a la media. Los datos atípicos no se eliminaron dado que se requiere que el modelo aprenda este tipo de comportamientos por el fenómeno de El Niño y de La Niña, pero sin duda, agregan mayor complejidad en el momento de hacer predicciones.

Figura 3-9. *Boxplot de las variables*



En la Figura 3-10 se presenta una matriz de dispersión que muestra las relaciones entre las diferentes variables usadas para la predicción del precio de energía eléctrica. Del gráfico se observa que el precio de bolsa y el precio de oferta tiene una relación lineal creciente y que el precio de oferta y la disponibilidad no térmica tiene una relación lineal decreciente, indicando que cuando los generadores ofertan un precio alto, probablemente es debido a que no tienen mucha disponibilidad en sus embalses. Adicionalmente, se observa que las etiquetas del tipo de día son 3 como se había mencionado anteriormente.

Figura 3-10. Matriz de dispersión de las variables



En la Figura 3-11 se observa la matriz de correlación entre las diferentes variables. Aquí hay algunas conclusiones que podemos sacar:

- ✓ **PrecioB vs. PrecioO:** Existe una fuerte correlación positiva entre el precio de bolsa y el precio de oferta, lo que indica que tienden a moverse juntos. Dada una correlación de 0.73 que también puede verse identificado con la Figura 3-10 y un valor mayor a 0.7. Con esto se puede tomar la decisión de hacer algunas sensibilidades para verificar si se elimina la variable y poder ejecutar modelos con menor carga computacional.

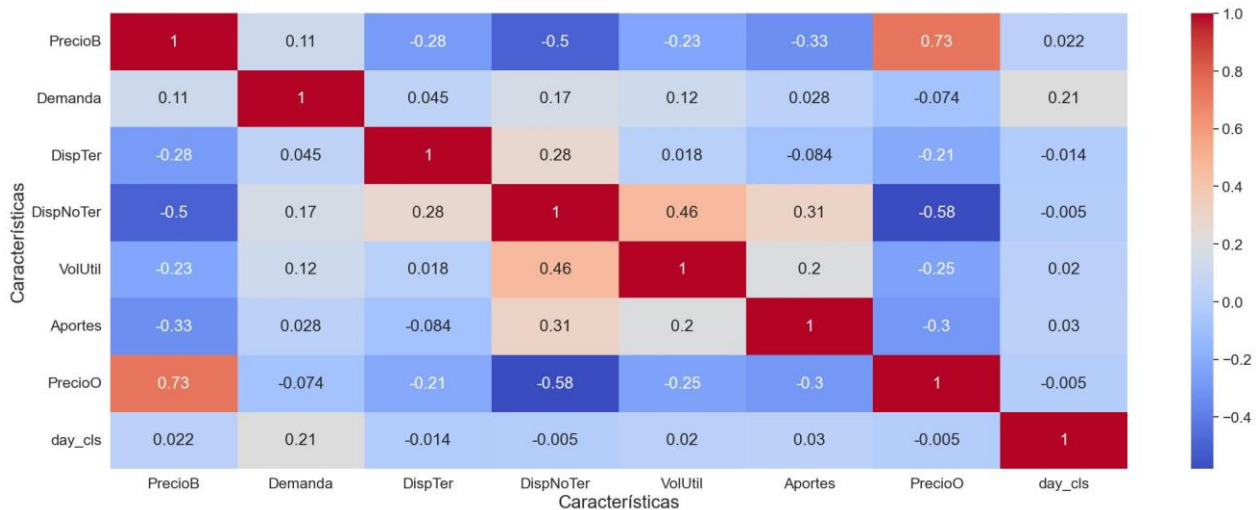
- ✓ **PrecioB vs. DispNoTer:** Existe una correlación negativa entre la disponibilidad no térmica y el precio en bolsa de energía. Esto sugiere que a medida que aumenta la disponibilidad

no térmica de energía, el precio de bolsa de energía disminuye. Con una correlación de -0.5 no suficiente para optar por eliminar la variable.

- ✓ **DispNoTer vs. VolUtil:** Existe una correlación positiva entre la disponibilidad de energía no térmica y el volumen útil. Con una correlación de 0.46 no suficiente para optar por eliminar alguna de las variables.

Estas correlaciones pueden ayudar a entender cómo estas variables se mueven y pueden ser útiles para predecir futuros comportamientos basados en cambios en una o más de estas variables.

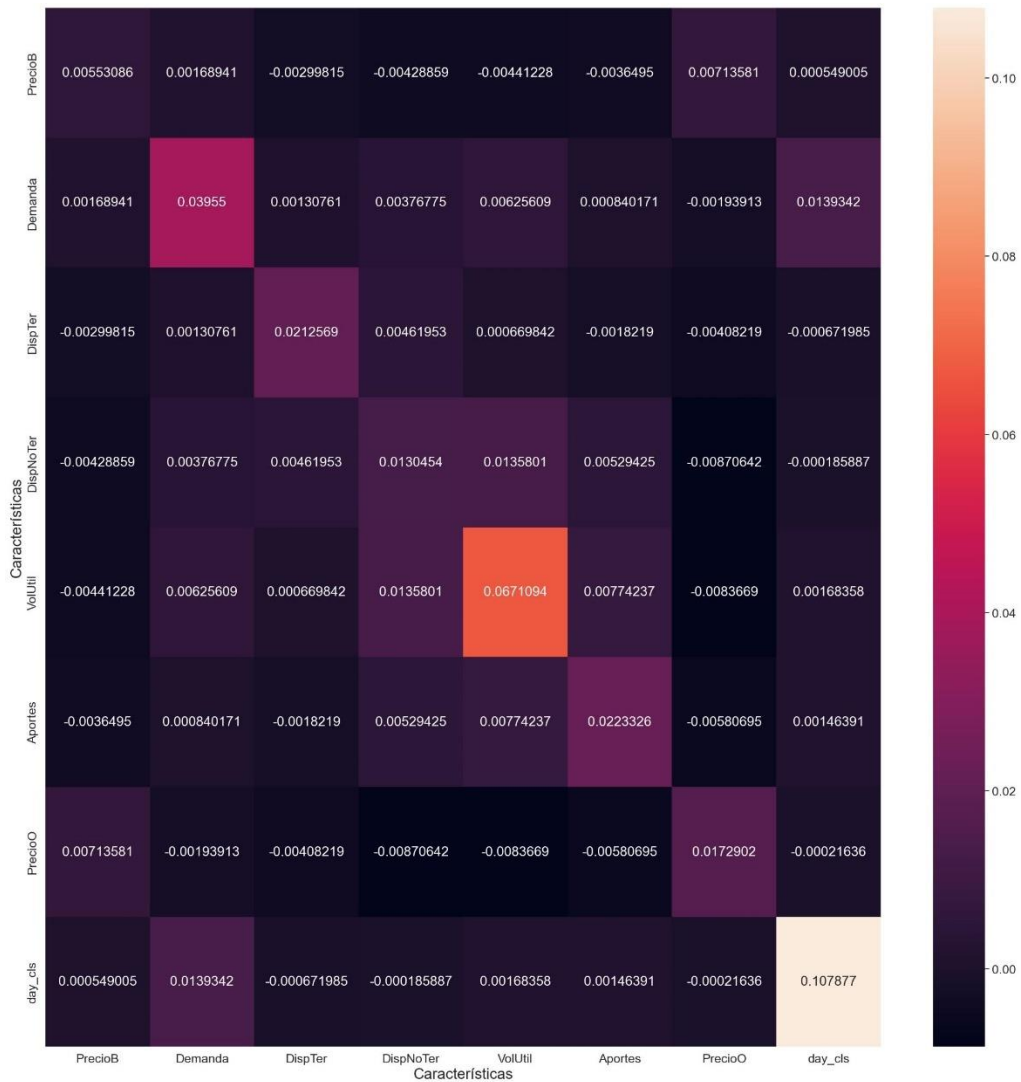
Figura 3-11. Matriz de correlación de las variables



En la Figura 3-12 se muestra un mapa de calor de una matriz de covarianza. La covarianza es una medida de cómo dos variables cambian juntas. Una covarianza alta indica que las dos variables tienden a cambiar en la misma dirección. Es decir, cuando una variable aumenta, la otra también tiende a aumentar, y cuando una variable disminuye, la otra también tiende a disminuir. Cuando se presenta una covarianza baja indica que las dos variables no están fuertemente relacionadas. Es decir, los cambios en una variable no están necesariamente asociados con cambios en la otra variable. Una covarianza negativa indica que las dos variables tienden a cambiar en direcciones opuestas. Es decir, cuando una variable aumenta, la otra tiende a disminuir, y viceversa.

Es importante tener en cuenta que la covarianza por sí sola puede ser difícil de interpretar, ya que su valor depende de las unidades de medida de las variables.

Figura 3-12. Matriz de covarianza de las variables



La Figura 3-13 muestra un gráfico de autocorrelación para la variable precio de bolsa. En el análisis de series de tiempo, la función de autocorrelación parcial juega un papel importante en los análisis de datos dirigido a la identificación de la medida del desfase en un modelo autorregresivo.

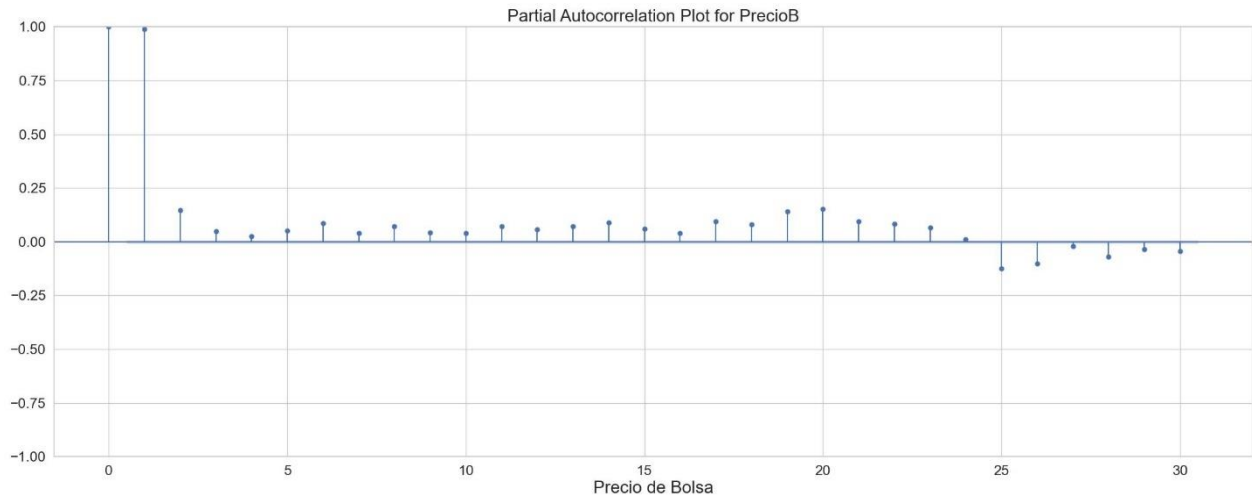
Eje X: Representa el desfase o retardo en las observaciones de la serie de tiempo. En este caso, se evalúa hasta el retardo 30.

Eje Y: Representa el valor de la autocorrelación, que varía de -1.00 a 1.00.

Líneas verticales: Cada línea vertical representa la autocorrelación parcial en un desfase específico. Por ejemplo, la línea más alta en el desfase 0 representa la autocorrelación parcial más alta.

La interpretación de este gráfico puede ayudar a identificar el orden de un modelo autorregresivo en el análisis de series de tiempo. Por ejemplo, si observas correlaciones significativas en el primer o segundo desfase, seguidas de correlaciones que no son significativas, esto podría indicar un término autorregresivo en los datos.

Figura 3-13. Autocorrelación del precio en bolsa



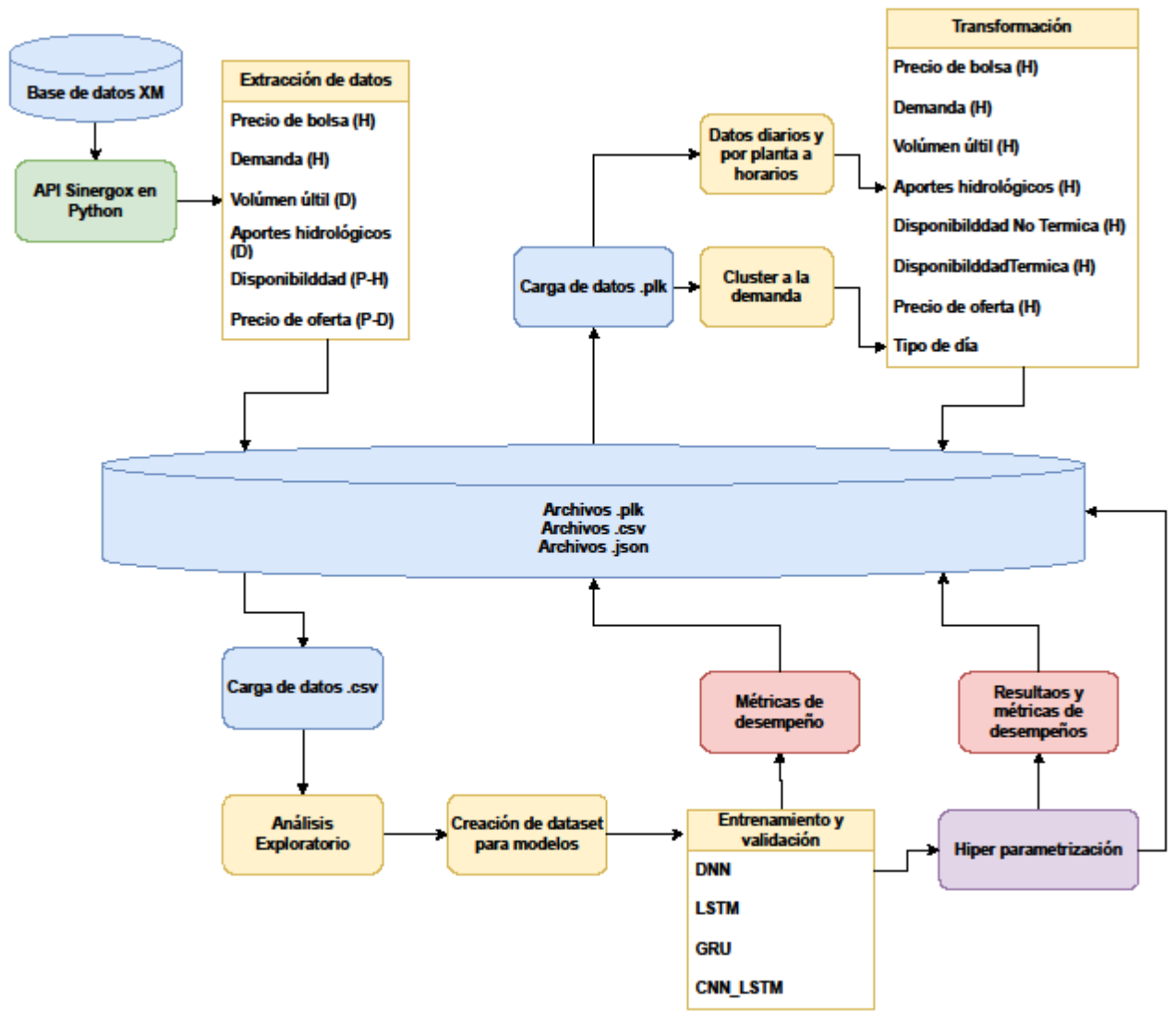
4. Proceso de analítica

4.1. Pipeline

A continuación, se muestra el esquema de flujo de trabajo que se realizó. Como primer paso se tiene la descarga de la información de las bases de datos de XM usando la API de Sinergox con código en Python. Se extrae la información en las resoluciones de la fuente original donde cada letra denota lo siguiente. H-Horario, D-Diario, P-Planta. Esta información es descargada de manera local en la computadora personal en archivo de formato plk. Luego, se carga de nuevo la información para ser procesada según los requerimientos del modelo a entrenar, en este caso se pasan las variables diarias a resolución horaria replicando los datos de un día para los 24 periodos, adicionalmente se realiza el clúster con la variable demanda para obtener el tipo de día de cada registro, esta información se almacena en un archivo .csv en el computador.

Posteriormente se carga el archivo .csv para hacer el análisis exploratorio de las variables para validar datos faltantes, outliers, estadísticas, correlaciones y obtener las gráficas. También este dataset es usado para obtener los tensores de entrada a los modelos de Deep Learning utilizados los cuales son entrenados y validados. Finalmente se realiza hiperparametrización para mejorar los modelos obteniendo las diferentes métricas de desempeño.

Figura 4-1. Flujo de trabajo



4.2. Preprocesamiento

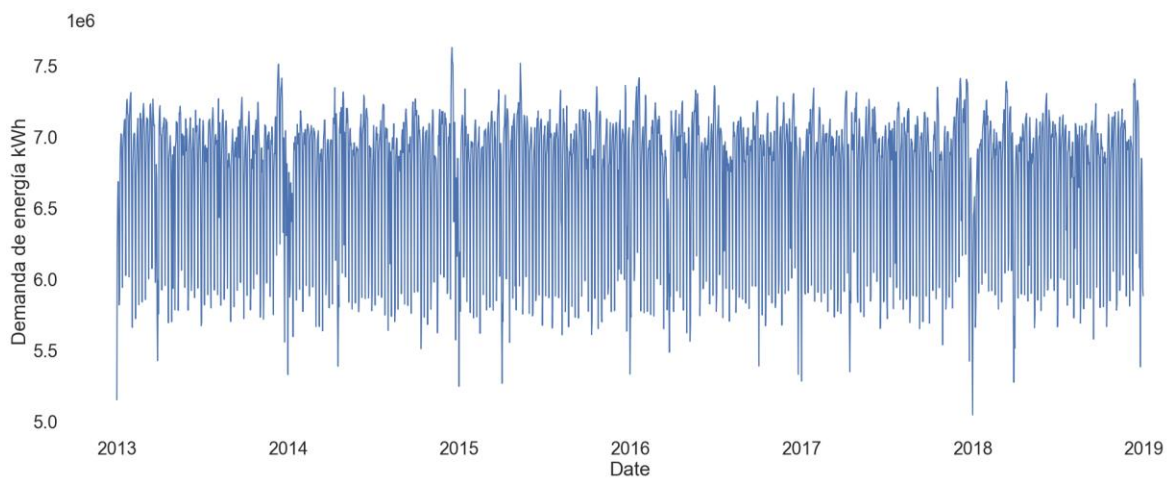
Para el modelo de predicción de energía en bolsa se utilizaron diferentes técnicas de preprocesamiento de datos, entre ellas se tiene: la normalización de las variables, la selección de características relevantes, la creación de nuevas características a partir de otras y la aumentación de datos. La aumentación de datos consistió en replicar los valores diarios de algunas variables (aportes, volumen útil y precio oferta) para cada hora del día, con el fin de tener todas las características en resolución horaria y mejorar el rendimiento de los modelos de deep learning. Además, se aplicó un algoritmo de agrupamiento, clustering, a la variable demanda de energía para crear una nueva columna llamada *day_cls*, que asigna una etiqueta horaria asociada al nivel de

demanda por tipo de día (ordinario, sábado y festivo) cuyo resultado se muestra en la Figura 4-2. Con puntos verdes se identifican los tipos de día ordinario, puntos amarillos identifican los sábados y los morados los días festivos, iniciando con el lunes identificado con el número cero. Se observa en la gráfica que algunos días de semana son identificados como festivos o sábados debido a nivel de demanda presentado. Antes de aplicar la técnica K-means la demanda, fue necesario eliminar la tendencia de los datos de tal manera que los tipos de día pueda ser comparable en cualquier año. Para lograr esto, se tomó como referencia el primer mes del año 2013. Después de quitar la tendencia los datos de demanda quedaron como se muestra en la Figura 4-3.

Figura 4-2. Etiqueta de Demanda de energía eléctrica en Colombia



Figura 4-3. Demanda de energía eléctrica en Colombia sin tendencia



4.3. Modelos de Deep Learning implementados

Se implementaron 4 tipos de modelos de DL tomado inicialmente de [14], los más comunes en el uso de series temporales (DNN, LSTM, GRU, CNN-LSTM). Estos modelos son un buen punto de partida para un problema de series temporales. Sin embargo, dependiendo del problema y del conjunto de datos específicos, es posible que se necesiten ajustar el número de capas, el número de neuronas en las capas, el dropout u otros parámetros que se realizaran en la sintonización de hiperparámetros. En este trabajo la DNN se refiere a una red neuronal artificial del tipo de propagación hacia adelante, conocida como red neuronal artificial “Feedforward”.

DNN

Tabla 4-1. Estructura modelo DNN

Capas	Tipo de Capas	Parámetros
1	Flatten	Forma de entrada: (n_steps, n_features)
2	Dense	128 neuronas, activación: relu
3	Dropout	Dropout = 0,3
4	Dense	128 neuronas, activación: relu
5	Dropout	Dropout = 0.3
6	Dense	n_horizon=24
Optimizer =Adam		

El modelo DNN es un modelo de Deep Neuronal Network construido utilizando la API Keras de TensorFlow. A continuación, se describe en términos generales que elementos tiene la arquitectura del modelo DNN utilizado:

- Es un modelo Secuencial, lo que significa que está compuesto por un apilado de capas.
- Utiliza la función Flatten que transforma el formato de las entradas de un array tridimensional a un array unidimensional.
- Utiliza capas Dense con 128 neuronas y función de activación ReLU (Rectified Linear Unit).
- ReLU es una función de activación popular que devuelve la entrada directamente si es positiva; de lo contrario, devuelve cero.
- Tiene función Dropout, que establece aleatoriamente una fracción (0.3 en este caso) de las unidades de entrada a 0 en cada actualización durante el tiempo de entrenamiento. Esto ayuda a prevenir el sobreajuste.

- La capa final es una capa Dense con un número de neuronas igual al número de salidas.
- Se utiliza la función de pérdida llamada Huber, que es menos sensible a los valores atípicos en los datos que la función de pérdida de error cuadrático medio.
- Se utiliza el optimizador Adam que es un algoritmo de optimización que se puede utilizar en lugar del procedimiento clásico de descenso de gradiente estocástico para actualizar los pesos de la red de forma iterativa basada en los datos de entrenamiento.

LSTM

Tabla 4-2. Estructura modelo LSTM

Capas	Tipo de Capas	Parámetros
1	LSTM	72 neuronas, activación: relu, Forma de entrada: (n_steps, n_features), return_sequences: True
2	LSTM	48 neuronas, activación: relu, return_sequences: False
3	(Flatten)	-
4	Dropout	Dropout: 0.3
5	Densa (Dense)	128 neuronas, activación: relu
6	Dropout	Dropout: 0.3
7	Densa (Dense)	n_horizon=24 neuronas

Optimizador =Adam

El modelo LSTM es un modelo de Long Short-Term Memory, que es un tipo de Red Neuronal Recurrente (RNN) construido utilizando la API Keras de TensorFlow. Las LSTM son particularmente adecuadas para procesar secuencias de datos con dependencias temporales, lo que las hace ampliamente utilizadas para el análisis de series temporales. A continuación, se describe en términos generales que elementos tiene la arquitectura del modelo LSTM utilizado:

- Es un modelo Secuencial, lo que significa que está compuesto por un apilado de capas.
- Utiliza capas LSTM con 72 y 48 unidades y la función de activación ReLU (Rectified Linear Unit). La LSTM devuelve una salida para cada paso de tiempo, no solo para el último. Esto es necesario cuando se apilan capas LSTM.
- Utiliza la función Flatten, que transforma el formato de las entradas de un array tridimensional a un array unidimensional.

- Utiliza la función Dropout, que establece aleatoriamente una fracción (0.3 en este caso) de las unidades de entrada a 0 en cada actualización durante el tiempo de entrenamiento. Esto ayuda a prevenir el sobreajuste.
- La capa final es una capa Dense con número de neuronas igual al número de salidas.
- Se utiliza la función de pérdida llamada Huber, que es menos sensible a los valores atípicos en los datos que la función de pérdida de error cuadrático medio.
- Se utiliza el optimizador Adam que es un algoritmo de optimización que se puede utilizar en lugar del procedimiento clásico de descenso de gradiente estocástico para actualizar los pesos de la red de forma iterativa basada en los datos de entrenamiento.

GRU

Tabla 4-3. Estructura modelo GRU

Capas	Tipo de Capas	Parámetros
1	GRU	72 neuronas, activación: relu, Forma de entrada: (n_steps, n_features), return_sequences: True
2	GRU	48 neuronas, activación: relu, return_sequences: False
3	(Flatten)	-
4	Dropout	Dropout: 0.3
5	Densa (Dense)	128 neuronas, activación: relu
6	Dropout	Dropout: 0.3
7	Densa (Dense)	n_horizon=24 neuronas
Optimizador =Adam		

Es un modelo GRU que es un modelo de Gated Recurrent Unit, que es un tipo de Red Neuronal Recurrente (RNN) construido utilizando la API Keras de TensorFlow. Las GRU son una variante de las LSTM que combina las puertas de entrada y olvido en una sola puerta de actualización. A continuación, se describe en términos generales que elementos tiene la arquitectura del modelo GRU utilizado:

- El modelo es un modelo Secuencial, lo que significa que está compuesto por un apilado de capas.

- Utiliza dos capas GRU con 72 y 48 unidades y la función de activación ReLU (Rectified Linear Unit). La GRU devolverá una salida para cada paso de tiempo, no solo para el último. Esto es necesario cuando se apilan capas GRU.
- Utiliza la función Flatten, que transforma el formato de las entradas de un array tridimensional a un array unidimensional.
- Utiliza la función Dropout, que establece aleatoriamente una fracción (0.3 en este caso) de las unidades de entrada a 0 en cada actualización durante el tiempo de entrenamiento. Esto ayuda a prevenir el sobreajuste.
- La capa final es una capa Dense cual número de neuronas igual al número de salidas.
- Se utiliza la función de pérdida Huber, que es menos sensible a los valores atípicos en los datos que la función de pérdida de error cuadrático medio.
- Se utiliza el optimizador Adam que es un algoritmo de optimización que se puede utilizar en lugar del procedimiento clásico de descenso de gradiente estocástico para actualizar los pesos de la red de forma iterativa basada en los datos de entrenamiento.

CNN-LSTM

Tabla 4-4. Estructura modelo CNN-LSTM

Capas	Tipo de Capas	Parámetros
1	Conv1D	64 filtros, tamaño del kernel: 6, activación: relu, Forma de entrada: (n_steps, n_features)
2	MaxPooling1D	MaxPooling1D =2
3	Conv1D	64 filtros, tamaño del kernel: 3, activación: relu
4	MaxPooling1D	MaxPooling1D =2
5	LSTM	72 neuronas, activación: relu, return_sequences: True
6	LSTM	48 neuronas, activación: relu, return_sequences: False
7	(Flatten)	-
8	Dropout	Dropout: 0.3
9	Densa (Dense)	128 neuronas
10	Dropout	Dropout: 0.3
11	Densa (Dense)	n_horizon=24 neuronas

Optimizador =Adam

El modelo CNN-LSTM es una combinación de una Red Neuronal Convolutiva (CNN) y una Red Neuronal Recurrente de Long Short-Term Memory (LSTM). A continuación, se describe en términos generales que elementos tiene la arquitectura del modelo CNN-LSTM utilizado:

- El modelo es un modelo Secuencial, lo que significa que está compuesto por un apilado de capas.
- Utiliza dos capas Conv1D, ambas con 64 filtros, una con tamaño de kernel de 6 y la otra de 3. La función de activación es ReLU (Rectified Linear Unit). Las capas convolucionales son efectivas para extraer características locales y espaciales de los datos.
- Utiliza capas MaxPooling1D con un tamaño de pool de 2. Las capas de agrupación se utilizan para reducir la dimensionalidad de los datos.
- Utiliza capas LSTM con 72 y 48 unidades y la función de activación ReLU (Rectified Linear Unit). La LSTM devuelve una salida para cada paso de tiempo, no solo para el último. Esto es necesario cuando se apilan capas LSTM.
- Utiliza la función Flatten, que transforma el formato de las entradas de un array bidimensional a un array unidimensional.
- Utiliza la función Dropout, que establece aleatoriamente una fracción (0.3 en este caso) de las unidades de entrada a 0 en cada actualización durante el tiempo de entrenamiento. Esto ayuda a prevenir el sobreajuste.
- La capa final es una capa Dense con el número de neuronas igual al número de salidas.
- Se utiliza la función de pérdida Huber, que es menos sensible a los valores atípicos en los datos que la función de pérdida de error cuadrático medio.
- Se utiliza el optimizador Adam que es un algoritmo de optimización que se puede utilizar en lugar del procedimiento clásico de descenso de gradiente estocástico para actualizar los pesos de la red de forma iterativa basada en los datos de entrenamiento.

4.4.Métricas

Para evaluar el desempeño de los modelos de predicción del precio de la energía en bolsa, basado en redes neuronales para series temporales, se usaron 4 métricas de negocio que miden la exactitud de las predicciones en relación con los valores reales del precio de la energía en bolsa. Estas métricas son MAE, MAPE, SMAPE y RMSE. A continuación, se muestran las ecuaciones de cada una respectivamente:

$$\text{MAE} = \frac{1}{n} \sum |y_{\text{real}} - y_{\text{pred}}|$$

$$\text{MAPE} = \frac{1}{n} \sum \left(\frac{|y_{\text{real}} - y_{\text{pred}}|}{y_{\text{real}}} \right) \times 100$$

$$\text{SMAPE} = \frac{1}{n} \sum \left(\frac{2 \times |y_{\text{real}} - y_{\text{pred}}|}{y_{\text{real}} + y_{\text{pred}}} \right) \times 100$$

$$\text{RMSE} = \sqrt{\frac{1}{n} \sum |y_{\text{real}} - y_{\text{pred}}|^2}$$

Donde:

n Es el número total de observaciones

y_{real} Es el valor real

y_{pred} Es el valor pronosticado

5. Metodología

5.1. Baseline

Como punto de partida se trabajó con 4 modelos de Deep Learning que son: DNN, LSTM, GRU y CNN-LSTM que se describieron en el capítulo anterior. Teniendo en cuenta los resultados obtenidos en la matriz de correlación donde se identificó una correlación mayor al 0.7 entre el precio de bolsa y el precio de oferta de los generadores, se realizó entrenamiento de los modelos teniendo en cuenta el precio de oferta (8 variables) y retirándolo (7 variables). Para esta primera validación se utilizaron 72 retardos para predecir los 24 periodos. Posteriormente, se calcularon las métricas de desempeños de cada modelo tanto para los conjuntos de entrenamiento, validación y prueba, donde para todos los modelos se calcula un promedio entre todos para cada tipo de métrica y así validar el comportamiento general. Los datos que se muestran en la Tabla 5-1 y Tabla 5-2:

Tabla 5-1. Resultados con el precio de oferta de los generadores 72_24

Model	Tipo	MAE	MAPE	SMAPE	RMSE
DNN	Train	15.78	8.74	8.50	17.93
	Val	116.35	18.64	20.92	131.65
	Test	45.87	32.88	27.95	48.00
LSTM	Train	15.98	8.91	9.55	20.24
	Val	78.59	13.15	13.87	93.85
	Test	25.40	18.15	16.31	28.72
GRU	Train	17.98	10.00	10.75	22.46
	Val	57.97	11.02	10.28	75.19
	Test	15.81	11.24	10.41	19.43
CNN-LSTM	Train	14.99	8.29	8.22	16.99
	Val	67.84	11.64	11.97	82.09
	Test	19.42	13.79	12.68	22.16
Promedio		41.00	13.87	13.45	48.23

Tabla 5-2. Resultados sin el precio de oferta de los generadores 72_24

Model	Tipo	MAE	MAPE	SMAPE	RMSE
DNN	Train	17.45	9.62	9.19	20.69
	Val	115.24	18.54	21.06	139.44
	Test	39.43	28.20	24.37	42.12
LSTM	Train	12.58	7.04	7.42	16.08
	Val	82.15	13.67	14.51	98.16
	Test	25.43	18.21	16.43	27.96

GRU	Train	18.78	10.47	11.49	24.18
	Val	74.10	13.46	12.97	87.81
	Test	17.85	12.75	11.73	21.27
CNN-LSTM	Train	20.80	11.69	12.80	24.98
	Val	67.50	11.58	11.88	80.44
	Test	8.73	6.40	6.68	10.98
Promedio		41.67	13.47	13.38	49.51

Con los resultados obtenidos se puede concluir que las métricas son muy similares, inclusive siendo un poco menor en el MAPE y SMAPE, pero mayor en el MAE y RMSE. Adicionalmente, en la Figura 5-1 y la Figura 5-2 se observa la gráfica del precio de bolsa en ambos casos donde el modelo CNN-LSTM parece más ajustado para los datos de prueba. Por lo tanto, se toma la decisión de eliminar la variable precio de oferta para los análisis posteriores, esto disminuye carga computacional a las simulaciones que se realizaran.

Figura 5-1. Curvas original y pronóstico del precio de bolsa considerando el precio de oferta

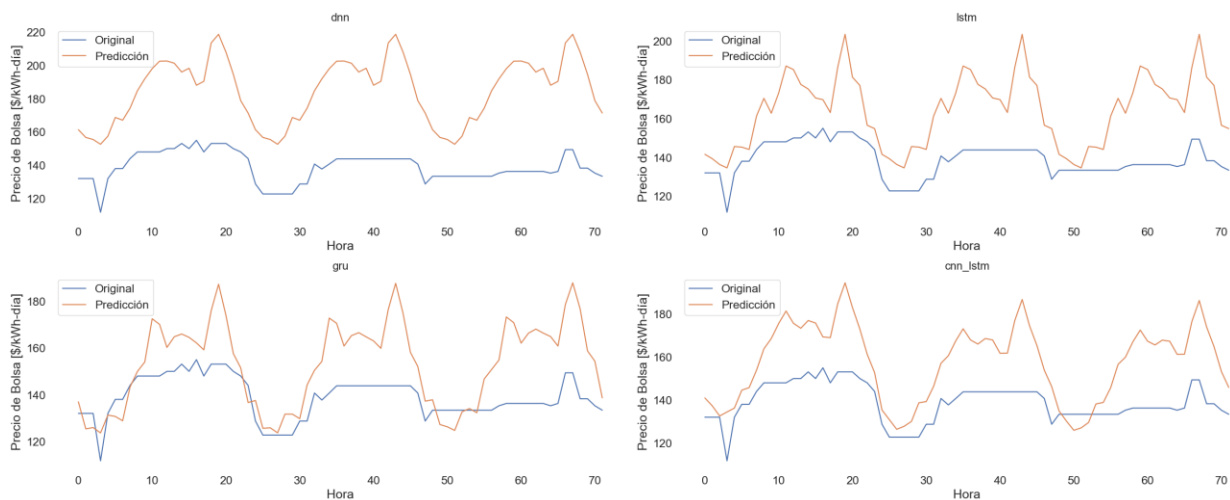
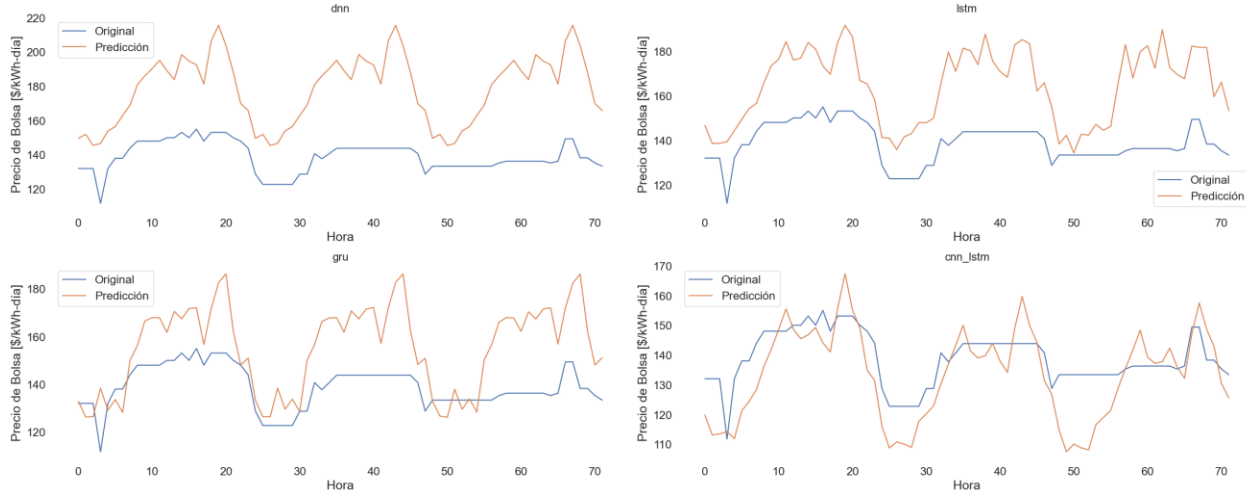


Figura 5-2. Curvas original y pronóstico del precio de bolsa sin considerar el precio de oferta



5.2. Validación e Iteraciones

El dataset considerado tiene un total de 52584 registros correspondiente a los datos horarios de cada una de las variables desde el año 2013 al 2018. Las variables que tenía resolución diaria fueron replicadas en cada hora para poder completar las columnas con las variables de volumen útil, aportes y tipo de día. La variable de disponibilidad la cual estaba por planta de generación y hora fue dividida en la disponibilidad de plantas térmicas y disponibilidad de plantas no térmicas, donde cada periodo fue calculado como el promedio de las disponibilidades de las plantas. Para el precio de bolsa y la demanda no fue necesario hacer tratamientos adicionales.

Tipo de ventanado y cantidad de retardos

Un aspecto importante es el tipo de ventanado a utilizar donde fue necesario definir aspectos como el traslape entre cada registro que se genera para ingresar a los modelos, en la Figura 5-3 se observa el traslape de 3 registros para 4 retardos utilizados implicando que en cada ventana se deja de usar un registro de la ventana anterior para hacer el pronóstico, esto puede generar inconvenientes ya que para pronosticar el datos 7 se están usando tres ventanas diferentes presentando problemas en las épocas de entrenamiento como se observa en la Figura 5-4, donde el comportamiento del MAE en cada época se vuelve oscilatorio.

Figura 5-3. Ventanado con Traslape de 3 registros

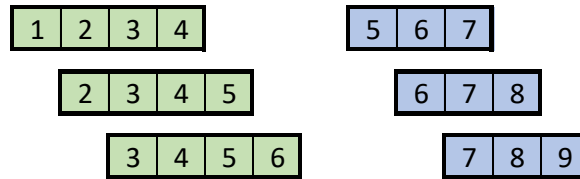
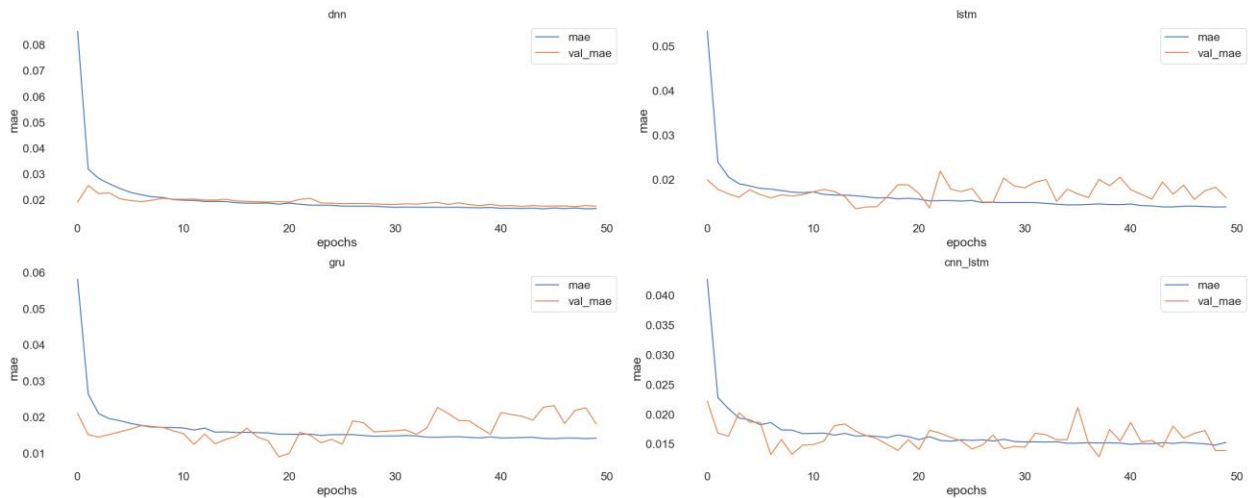


Figura 5-4. Métrica MAE en el entrenamiento haciendo traslape de todos los registros -1



El otro tipo de ventanado es sin realizar traslape como se muestra en la Figura 5-5, este ventanado fue el utilizado en este trabajo ya que se evitó el comportamiento oscilatorio del MAE en los modelos durante la etapa de entrenamiento. Aunque es probable que en algunos modelos se presente (Figura 5-6), pero en menor medida, en cuyo caso una de las recomendaciones es modificar parámetros como el batch de entrenamiento.

Figura 5-5. Ventanado sin traslape

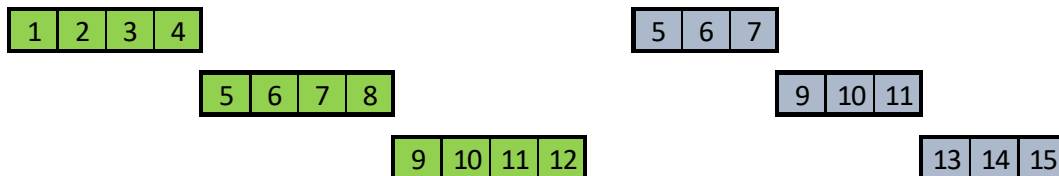
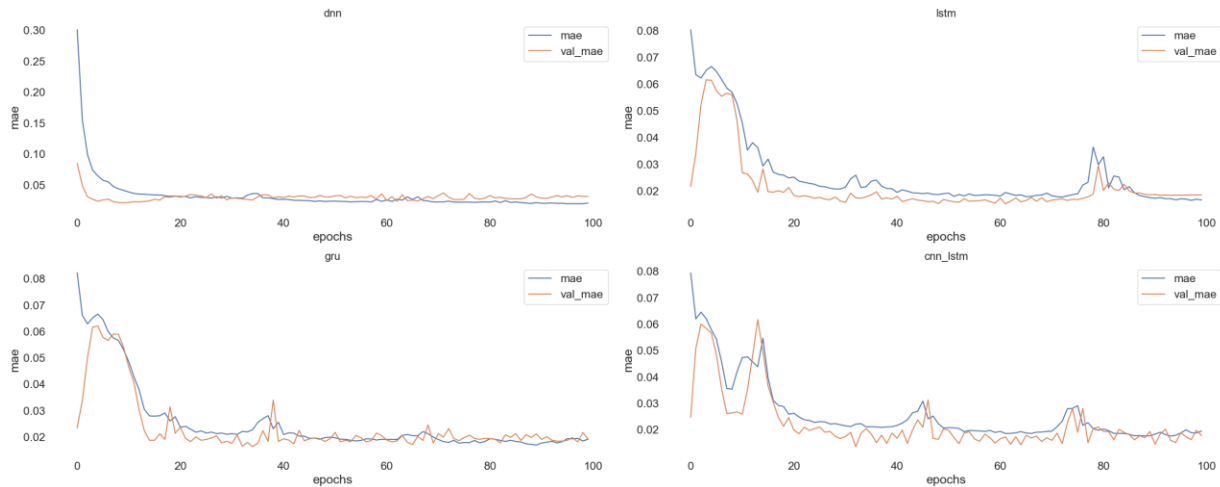


Figura 5-6. Métrica MAE en el entrenamiento sin traslape



Otro factor importante para determinar es la cantidad de retardos a utilizar para el modelo final, donde pueden ser múltiples las opciones a considerar. Teniendo en cuenta que el pronóstico a realizar es con un horizonte de 24 periodos, se parte de que los retardos a utilizar son con una ventana de igual tamaño y se prueban ventanas que son múltiplos de 24 periodos para así considerar datos que tienen información de los días hacia atrás. En este caso se usaron retardos de 24, 48, 72 y 96 horas correspondiente a 1, 2, 3 y 4 días respectivamente. Cada modelo fue probado con todos los retardos y con base en el análisis de los errores se decidió cual es la mejor cantidad de retardos a considerar para realizar la sintonización de hiperparámetros para las técnicas de DL. En este caso los resultados arrojaron que la mejor cantidad de retardos es de 72.

Sintonización de hiperparámetros

Con el objetivo de mejorar los modelos, a cada uno se le realizó sintonización de hiperparámetros utilizando la librería Keras Tuner disponible en Python. La ventaja de esta librería es que realiza una búsqueda aleatoria permitiendo mejorar los tiempos de procesamiento y además almacena de manera automática los resultados en carpetas donde queda cada una de las ejecuciones y los mejores hiperparámetros encontrados. Este proceso fue realizado utilizando 72 retardos y una ventana de predicción de 24 periodos. Las posibilidades de cambiar hiperparámetros son múltiples inclusive con la posibilidad de cambiar la arquitectura de la red neural. Para este trabajo la

arquitectura se dejó como se mostró en capítulos anterior y se hizo variación en los hiperparámetros de cada modelo como se muestra a continuación:

- DNN, LSTM y GRU:
 - Número de neuronas en cada capa
 - Semilla
 - Dropout en cada capa
 - Función de activación (relu, tanh, leaky_relu)
 - Optimizador (Adam, Adadelata)
 - Función de pérdidas (huber, MAE)
- CNN-LSTM:
 - Número de neuronas en cada capa
 - Semilla
 - Número de filtros considerados en las capas convolucionales
 - Dropout en cada capa
 - Función de activación (relu, tanh, leaky_relu)
 - Optimizador (Adam, adadelata)
 - Función de pérdidas (huber, MAE)

Adicionalmente, como la función objetivo a minimizar durante la sintonización de hiperparámetros se utilizó la función de pérdida Hubber.

5.3 Herramientas

Las herramientas usadas para el desarrollo de este trabajo fueron:

- Google Colab Pro para ejecutar scripts de sintonización de hiperparámetros
- Visual Studio Code como editor de desarrollo.
- Python como herramienta principal de desarrollo
- PowerBI para gráficas
- Excel para tablas y resultados
- Copilot para asistencia en la escritura del código
- One Drive para edición colaborativa del documento
- GitHub como repositorio

6. Resultados

Para presentar los resultados, se utilizaron tablas donde se muestran las métricas de desempeño de cada modelo con los conjuntos de datos de entrenamiento, validación, y pruebas para las métricas definidas que son MAE (Unidades en \$COP/kWh), MAPE (Unidades en %), SMAPE, MAPE (Unidades en %), y RMSE (Unidades en \$COP/kWh). Las métricas fueron calculadas de acuerdo con los valores reales, es decir, después de obtener la predicción se aplicó la inversa al escalamiento realizado. Adicionalmente se tomaron 72 periodos. A continuación, se muestran los resultados para los diferentes retardos utilizados y con base en el que presentó mejor comportamiento se realizó hiper parametrización.

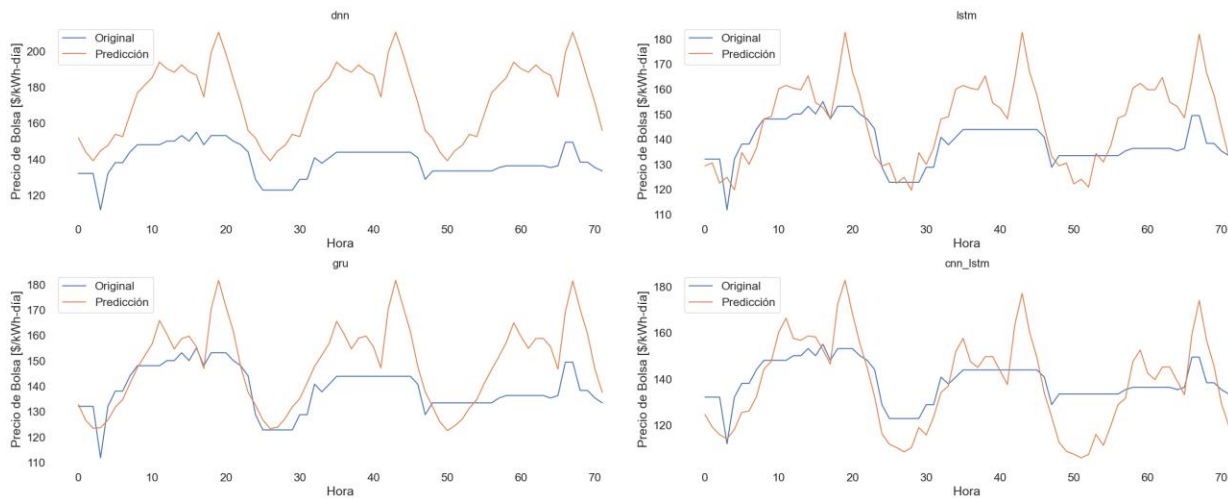
6.1. Métricas y gráficas

La Tabla 6-1 muestra los resultados obtenidos considerando 24 retardos, en este caso el modelo que mejor desempeño tiene en la métrica MAPE y RMSE para datos de prueba es la LSTM con 13.38 y 20.58 \$COP/kWh respectivamente, para datos de entrenamiento fue el modelo CNN-LSTM con 7.03% y 15.71 \$COP/kWh. En la Figura 6-1 se observa la gráfica del precio de bolsa original y pronosticado para 3 días consecutivos donde se evidencia que el modelo LSTM tiene una mejor aproximación, para el caso de la CNN-LSTM se tiene una buena aproximación para los dos primeros días.

Tabla 6-1. Métricas con 24 retardos

Model	Tipo	MAE	MAPE	SMAPE	RMSE
DNN	Train	25.40	13.45	12.25	31.92
	Val	177.14	218.29	97.60	196.30
	Test	32.41	24.33	20.89	37.03
LSTM	Train	15.81	8.51	9.04	19.52
	Val	73.12	90.62	60.57	75.93
	Test	17.42	13.38	12.19	20.58
GRU	Train	15.26	8.20	8.69	19.12
	Val	71.35	88.97	59.41	74.74
	Test	19.40	14.65	13.31	22.24
CNN-LSTM	Train	13.23	7.03	7.15	15.71
	Val	85.89	106.70	67.43	89.03
	Test	24.71	18.69	16.64	27.64
Promedio		47.59	51.07	32.10	52.48

Figura 6-1. Precio de bolsa Original y Predicción con 24 retardos



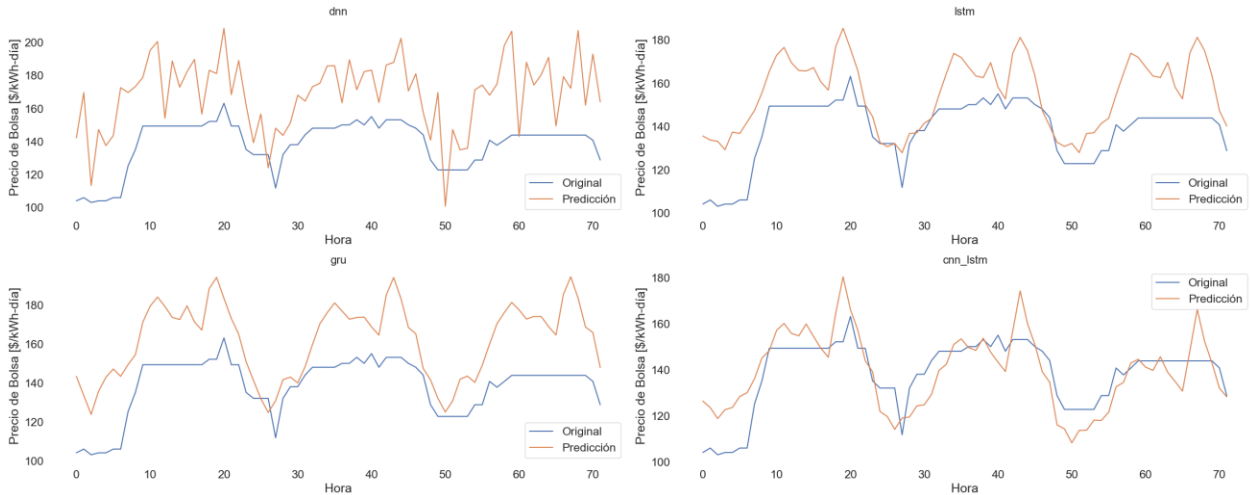
La Tabla 6-2 muestra los resultados obtenidos considerando 48 retardos, en este caso el modelo que mejor desempeño tiene en las métricas MAPE y RMSE para datos prueba es la CNN-LSTM con 6.87% y 11.07 \$COP/kWh, en entrenamiento fue de 7.82% y 17.83 \$COP/kWh respectivamente, aunque en los datos de validación las métricas fueron malas indicando un posible overfitting. En la

Figura 6-2 se observa la gráfica del precio de bolsa original y pronosticado para 72 periodos consecutivos donde se evidencia que el modelo CNN-LSTM tiene una mejor aproximación.

Tabla 6-2. Métricas con 48 retardos

Model	Tipo	MAE	MAPE	SMAPE	RMSE
DNN	Train	16.07	8.78	8.73	18.46
	Val	99.79	123.24	70.41	110.42
	Test	31.28	23.04	20.18	34.80
LSTM	Train	22.99	12.68	13.85	26.34
	Val	66.55	82.62	55.38	70.89
	Test	16.61	12.37	11.36	19.46
GRU	Train	18.32	9.97	10.40	22.66
	Val	72.39	89.33	58.52	76.78
	Test	23.41	17.13	15.49	25.93
CNN-LSTM	Train	14.05	7.82	8.31	17.83
	Val	59.20	73.40	50.94	63.55
	Test	8.97	6.87	6.74	11.07
Promedio		37.47	38.94	27.52	41.52

Figura 6-2. Precio de bolsa Original y Predicción con 48 retardos

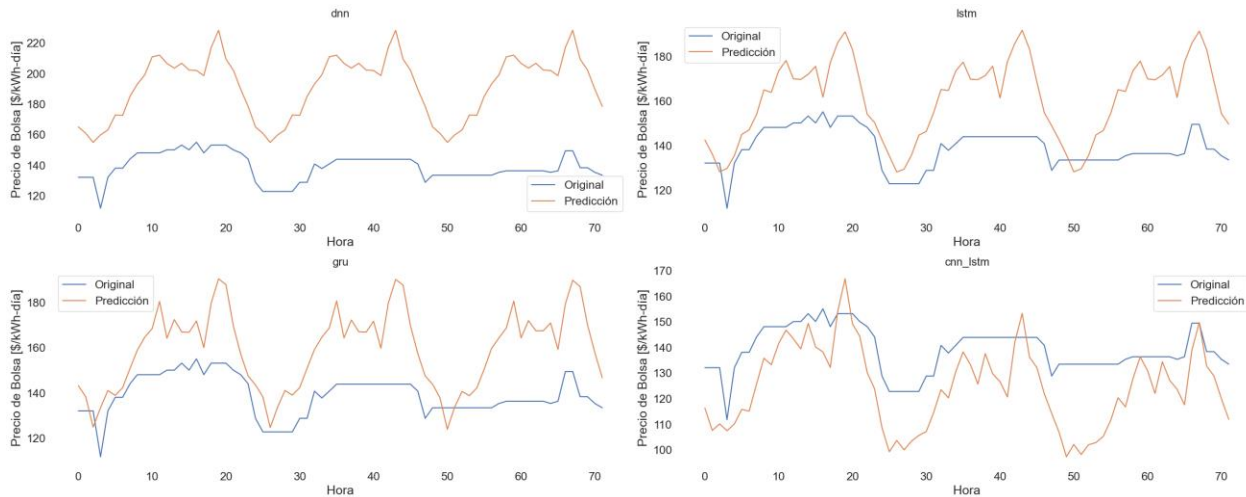


La Tabla 6-3 muestra los resultados obtenidos considerando 72 retardos, en este caso el modelo que mejor desempeño tiene en las métricas MAPE y RMSE para datos de prueba es el CNN-LSTM con 8.02% y 6.68 \$COP/kWh respectivamente. En entrenamiento también es el modelo CNN-LSTM con 6.7% y 14.34 \$COP/kWh. En la Figura 6-3 se observa la gráfica del precio de bolsa original y pronosticado para 72 periodos consecutivos donde se evidencia que el modelo CNN-LSTM es la que más se aproxima a la curva original.

Tabla 6-3. Métricas con 72 retardos

Model	Tipo	MAE	MAPE	SMAPE	RMSE
DNN	Train	15.59	8.70	9.09	18.91
	Val	95.59	16.29	17.03	122.67
	Test	35.20	25.14	21.95	38.29
LSTM	Train	14.52	8.09	8.52	17.86
	Val	56.03	10.05	10.14	80.78
	Test	11.48	8.20	7.78	14.42
GRU	Train	14.85	8.25	8.21	17.46
	Val	82.54	15.01	14.04	98.64
	Test	11.85	8.46	7.99	14.82
CNN-LSTM	Train	12.01	6.70	6.86	14.34
	Val	91.45	14.82	16.36	111.13
	Test	11.08	8.02	8.13	13.46
Promedio		37.68	11.48	11.34	46.90

Figura 6-3. Precio de bolsa Original y Predicción con 72 retardos

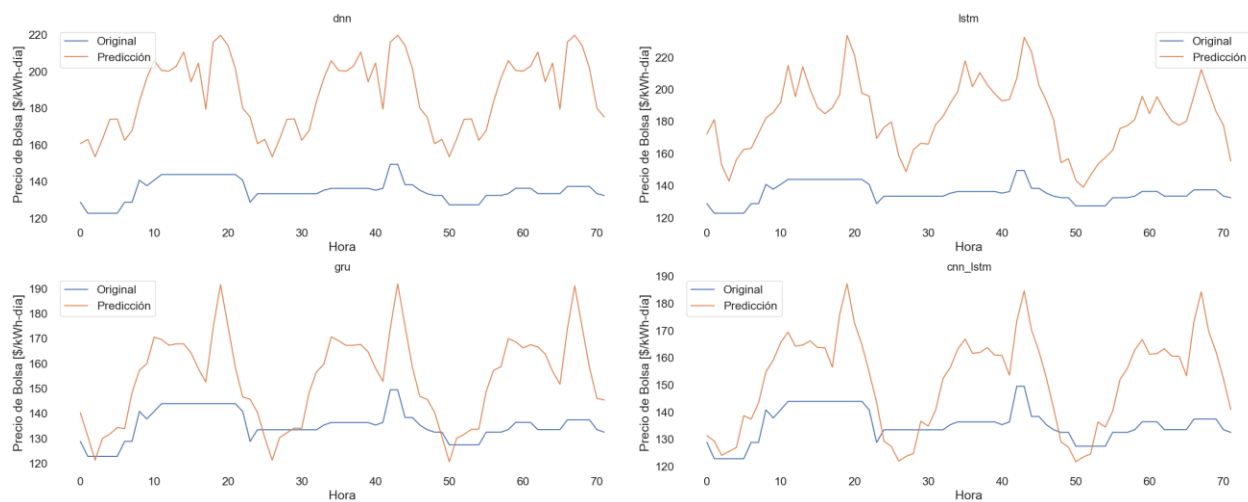


La Tabla 6-4 muestra los resultados obtenidos considerando 96 retardos, en este caso el modelo que mejor desempeño tiene en las métricas MAPE y RMSE para los datos prueba es el CNN-LSTM con 13.3% y 21.13 \$COP/kWh respectivamente. En datos de entrenamiento tiene valores de 8.07% y 18.31 \$COP/kWh. En la Figura 6-4 se observa la gráfica del precio de bolsa original y pronosticado para 72 periodos consecutivos donde el modelo CNN-LSTM el cual presenta el mejor ajuste.

Tabla 6-4. Métricas con 96 retardos

Model	Tipo	MAE	MAPE	SMAPE	RMSE
DNN	Train	23.12	12.99	11.95	27.94
	Val	106.39	107.78	63.08	118.03
	Test	52.46	38.60	31.99	54.83
LSTM	Train	15.22	8.62	8.53	17.37
	Val	68.16	69.17	47.45	74.47
	Test	48.21	35.36	29.60	51.27
GRU	Train	18.02	10.35	11.27	22.50
	Val	52.21	53.72	38.87	59.22
	Test	19.52	14.24	13.01	23.05
CNN-LSTM	Train	14.11	8.07	8.41	18.31
	Val	40.91	42.78	32.39	47.87
	Test	18.25	13.30	12.27	21.13
Promedio		39.72	34.58	25.74	44.66

Figura 6-4. Precio de bolsa Original y Predicción con 96 retardos



Teniendo en cuenta que el menor valor en el MAPE y RMSE para la DNN y la CNN-LSTM se presentó con 48 retardos y para la LSTM y GRU con 72 retardos y que adicionalmente la CNN-LSTM con 72 retardos tiene valores de error más próximos a la de 48 retardos, se calcularán los hiperparámetros de las técnicas de DL considerando 72 retardos para una ventana de predicción de 24 periodos. En las siguientes tablas se muestran los parámetros de cada modelo después de ejecutar la sintonización de hiperparámetros de los cuales se resaltan los siguientes aspectos:

- Para el modelo DNN se aumentó el número de neuronas de cada capa, se cambia la función de activación por una Relu y el dropout de la primera capa es cero.
- Para la LSTM se baja el número neuronas de la capa 1 pero se aumenta el de la capa 2 quedando ambas con 64, la función de activación es Leaky_Relu, aumenta el dropout de la primera capa a 0.4 y el número de neuronas en la capa Densa disminuye a 96.
- Para la GRU se disminuye el número de neuronas de la primera capa a 64, se aumenta las neuronas de la segunda capa a 96, se cambia la función de activación a Leaky_Relu y el dropout de la capa Densa queda en 0.
- Para le CNN-LSTM aumentan los filtros para las capas convolucionales a 80 y 256 respectivamente, aumenta el número de neuronas en la capa LSTM a 96 y 256, se disminuye el dropout de la capa LSTM a 0.2, se disminuye el número de neuronas en la capa densa a 96 y aumenta el dropout a 0.4.

- Todos los modelos mantienen la función de optimización Adam y la función de pérdidas Huber.

Tabla 6-5. Hiperparámetros modelo DNN

Capa	Tipo de Capa	Parámetros
1	Flatten	Forma de entrada: (n_steps=72, n_features=7)
2	Dense	160 neuronas, activación: Relu
3	Dropout	Dropout = 0
4	Dense	256 neuronas, activación: Relu
5	Dropout	Dropout = 0,4
6	Dense	n_horizon=24
Optimizador =Adam, loss= Huber, seed =35		

Tabla 6-6. Hiperparámetros modelo LSTM

Capa	Tipo de Capa	Parámetros
1	LSTM	64 neuronas, activación: leaky_relu, Forma de entrada: (n_steps=72, n_features=7), return_sequences: True
2	LSTM	64 neuronas, activación: leaky_relu, return_sequences: False
3	Flatten	-
4	Dropout	Dropout: 0.4
5	Densa (Dense)	96 neuronas, activación: leaky_relu
6	Dropout	Dropout: 0.3
7	Densa (Dense)	n_horizon=24 neuronas
Optimizador =Adam, Huber, seed =969		

Tabla 6-7. Hiperparámetros modelo GRU

Capa	Tipo de Capa	Parámetros
1	GRU	64 neuronas, activación: leaky_relu, Forma de entrada: (n_steps=72, n_features=7), return_sequences: True
2	GRU	96 neuronas, activación: leaky_relu, return_sequences: False
3	(Flatten)	-
4	Dropout	Dropout: 0.3
5	Densa (Dense)	128 neuronas, activación: leaky_relu
6	Dropout	Dropout: 0
7	Densa (Dense)	n_horizon=24 neuronas

Optimizador =Adam, loss= Huber, seed =389

Tabla 6-8. Hiperparámetros modelo CNN-LSTM

Capa	Tipo de Capa	Parámetros
1	Conv1D	80 filtros, tamaño del kernel: 6, activación: Tanh, Forma de entrada: (n_steps=72, n_features=7)
2	MaxPooling1D	MaxPooling1D =2
3	Conv1D	256 filtros, tamaño del kernel: 3, activación: Tanh
4	MaxPooling1D	MaxPooling1D =2
5	LSTM	96 neuronas, activación: Tanh, return_sequences: True
6	LSTM	256 neuronas, activación: Tanh, return_sequences: False
7	(Flatten)	-
8	Dropout	Dropout: 0.2
9	Densa (Dense)	96 neuronas
10	Dropout	Dropout: 0.4
11	Densa (Dense)	n_horizon=24 neuronas

Optimizador =Adam, loss= Huber, seed =160

Se realiza el entrenamiento de todos los modelos con los mejores hiperparámetros encontrados para validar si se tiene mejora con respecto a los resultados obtenidos anteriormente, en la Tabla 6-9 se observa que el modelo que mejora desempeño muestra en el LSTM con valor en le MAPE de 6.93% y el RMSE de 11.87 \$COP/kWh en prueba, para entrenamiento los errores fueron de 9.12% y el RMSE de 20.33 \$COP/kWh. Sin embargo, el modelo CNN-LSTM que había presentado mejores métricas teniendo en cuenta 48 retardos tiene un bajo desempeño en pruebas, pero el mejor en validación.

Tabla 6-9. Métricas de desempeño con teniendo en cuenta los hiper parámetros optimizados

Model	Tipo	MAE	MAPE	sMAPE	RMSE
DNN	Train	34.94	19.35	22.23	41.68
	Val	119.34	19.45	22.15	144.69
	Test	19.78	14.06	15.06	25.45
LSTM	Train	16.43	9.12	8.73	20.33
	Val	72.90	12.50	12.84	86.19
	Test	9.67	6.93	6.80	11.87
GRU	Train	14.05	7.82	8.19	17.44
	Val	78.50	14.50	13.53	91.67
	Test	21.77	15.77	17.62	24.85
	Train	24.30	13.65	14.95	27.25

CNN-LSTM	Val	59.79	10.50	10.62	75.27
	Test	84.44	61.18	89.90	85.50
Promedio		46.33	17.07	20.22	54.35

En la Figura 6-5 y Figura 6-6 se muestran los valores de pérdida y MAE durante las épocas de entrenamiento, en general presentan buen comportamiento a excepción del modelo CNN-LSTM que presenta valores oscilatorios en le mae durante el entrenamiento.

Figura 6-5. Valores de pérdida durante el entrenamiento modelos con parámetros optimizados

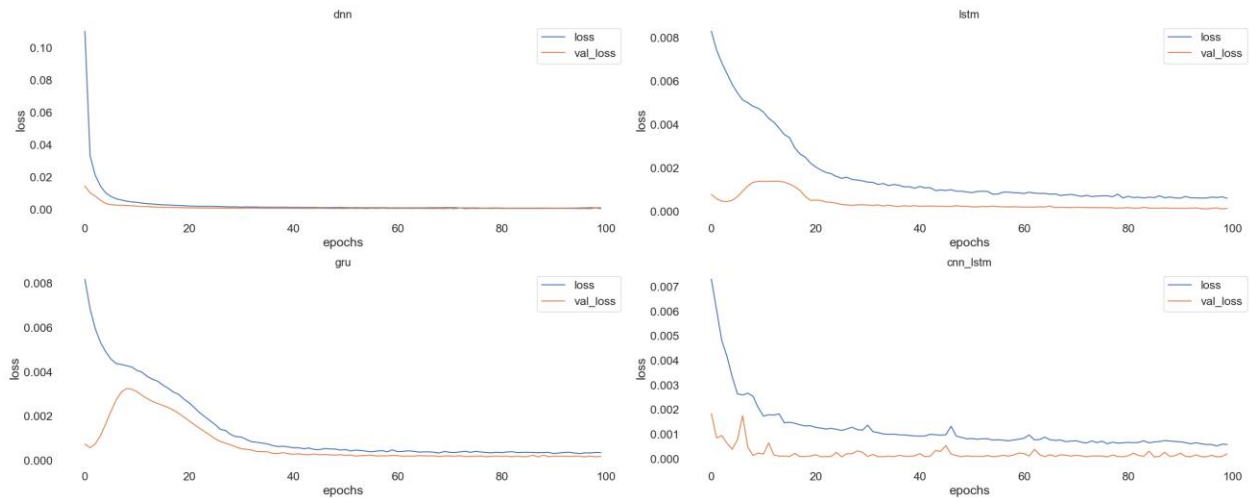
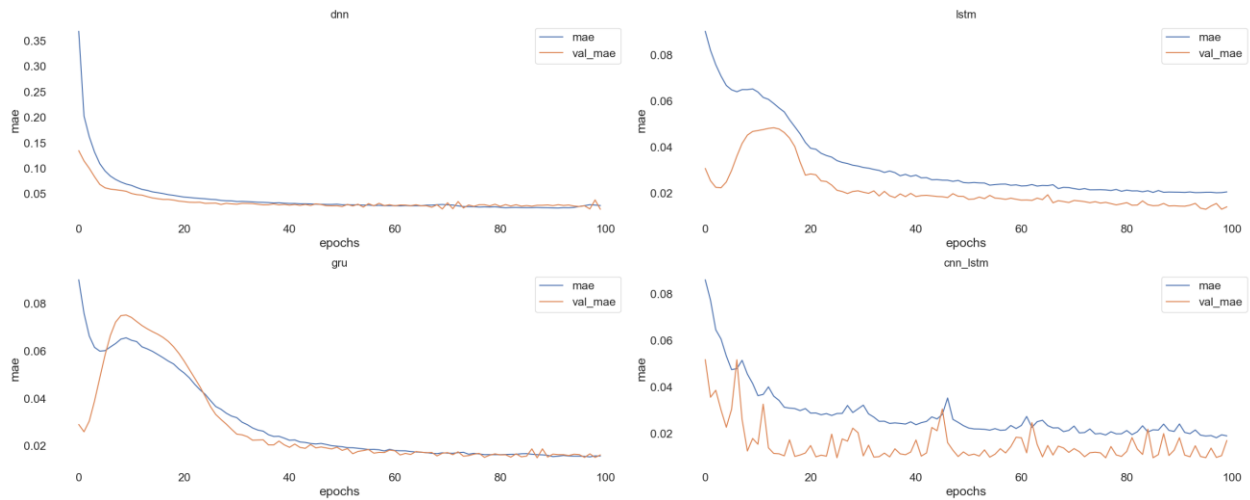
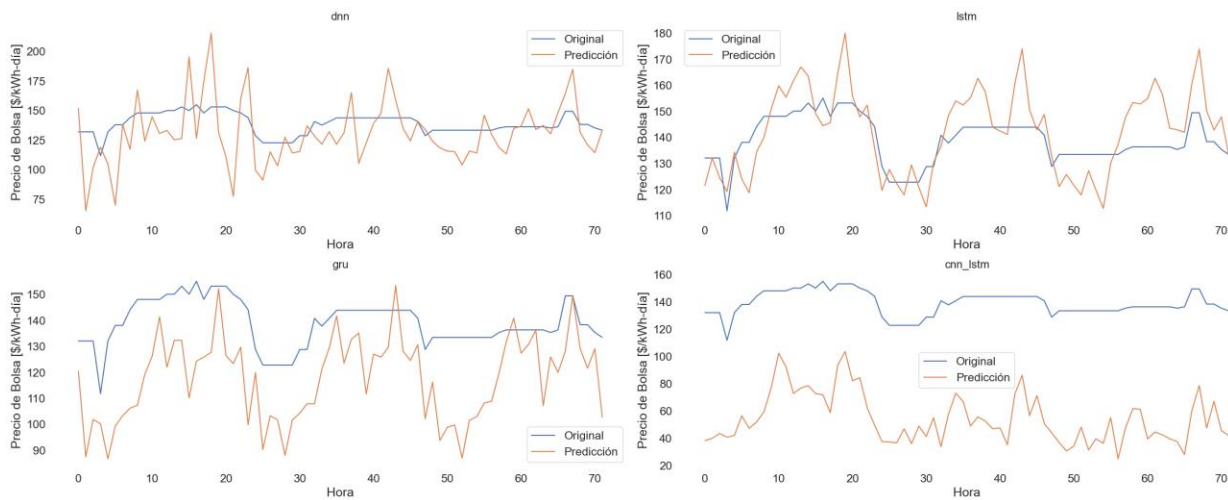


Figura 6-6. Valores de MAE durante el entrenamiento de los modelos con parámetros optimizados



En la Figura 6-7 se muestran las gráficas del precio de bolsa original y pronosticado donde se observa que el modelo LSTM presenta un buen ajuste, lo cual corresponde con las métricas de desempeño. El modelo DNN presenta un ajuste con varios picos que donde la media de mueve alrededor del valor original. El modelo GRU y CNN-LSTM presentan valores subestimados con respecto al pronóstico, siendo el modelo CNN-LSTM con el peor ajuste. Para este caso sería recomendable por ejemplo limitar el número de filtros y cambiar la función a minimiza ya que este modelo fue el que mejor se comportó en la etapa inicial.

Figura 6-7. Precio de bolsa original y pronosticado de los modelos con los parámetros optimizados



6.2. Consideraciones de producción

En caso de que se desee poner un modelo de pronóstico de precio de bolsa en producción, para este caso se recomienda primero hacer otras simulaciones teniendo en cuenta arquitecturas adicionales ya que a pesar de tener un pronóstico ajustado para el modelo LSTM, aún presente variaciones importantes. Adicionalmente, después de tener un modelo con mejor ajuste, se recomienda estar actualizando el modelo con los últimos datos que entrega el mercado debido a la alta volatilidad de la variable que se está prediciendo.

7. Conclusiones

El pronóstico de precio de bolsa en el mercado eléctrico colombiano es una variable con una volatilidad importante que esta influenciada por varios aspectos como los cambios climáticos donde las temporadas secas generan picos en el precio dada la alta dependencia de la generación hidráulica. Adicionalmente al ser un mercado con poca oferta (generadores) y mucha demanda (comercializadore), la estrategia de los agentes que determinan el precio de bolsa a través de la oferta de precio y disponibilidad pueden impactar la manera en la que se forma el precio en el mercado. También las políticas energéticas impulsadas por el gobierno de turno afectan el comportamiento de esta variable.

El tipo de ventando que se utiliza en los modelos de serie de tiempo tanto en el número de retados como en el traslape que se utilice es un factor de mucha relevancia dado que puede influir en la estabilidad del modelo y en los tiempos de ejecución cuando se está entrenando y haciendo las validaciones de los parámetros.

El mejor modelo encontrado fue el LSTM con 72 retardos, en el cual se obtuvo valores en el MAPE y RMSE en pruebas de 6.93% y 11.87 \$COP/kWh. Sin embargo, el modelo CNN-LSTM que había obtenido buenas métricas antes de la sintonización, arrojó hiperparámetros que dieron los valores de errores más altos en pruebas. Esto indica que en el proceso de sintonización de hiperparámetros, el rango de variabilidad de cada hiperparámetro debe ser más estrecho y próximo a los valores usados inicialmente.

Se tuvieron en cuenta inicialmente 8 variables para el pronóstico del precio de bolsa, el análisis de correlación permitió determinar que el precio de oferta estaba altamente correlacionado (> 0.7) y que era una variable que podía ser eliminada. Esto pudo ser corroborado haciendo algunos pronósticos teniendo en cuenta el precio de oferta en las variables de entrada y sin incluirlo, mostrando que las métricas de error no cambiaban considerablemente. Sin embargo, es importante mencionar que dada la complejidad de la variable a pronosticar es posible que otras variables pueda ser eliminadas por eventualmente presentar colinealidad, lo cual debe ser validado con otras técnicas de estadística para determinar si los pronósticos mejoran.

Para variables que presenten comportamientos cíclicos, por tipo de día o temporada como es el caso de la demanda de energía, es recomendable agregar variables dummy que permitan al modelo tener en cuenta este comportamiento estacional. Se podría hacer un análisis adicional con el caso de los aportes para determinar las temporadas secas y de lluvias y así validar si el modelo mejora.

Referencias

- [1] J. Barrientos and M. Toro, “Análisis de los fundamentales del precio de la energía eléctrica: evidencia empírica para Colombia,” *Rev. Econ. del Caribe*, vol. 2106, no. 19, pp. 34–63, 2017.
- [2] E. Castaño and J. Sierra, “Sobre la existencia de una raíz unitaria en la serie de tiempo mensual del precio de la electricidad en Colombia,” *Lect. Econ.*, no. 76, pp. 259–291, 2012, doi: 10.17533/udea.le.n76a12817.
- [3] D. Toro Restrepo, “Descripción e influencia de las variables hidrológicas en la determinación del precio spot de la energía eléctrica en Colombia,” pp. 1–32, 2015.
- [4] J. Barrientos, E. Rodas, E. Velilla, M. Lopera, and F. Villada, “Modelo para el pronóstico del precio de la energía eléctrica en Colombia TT - A model for forecasting electricity prices in Colombia TT - Modèle de prévision des prix d'électricité en Colombie,” *Lect. Econ.*, no. 77, pp. 91–127, 2012, [Online]. Available: http://www.scielo.org.co/scielo.php?script=sci_arttext&pid=S0120-25962012000200003&lang=pt%0Ahttp://www.scielo.org.co/pdf/le/n77/n77a3.pdf.
- [5] J. C. Gaviria, “Pronóstico de precios diarios de electricidad: Regresiones dinámicas con variables explicativas de mercados hidrotérmicos,” pp. 1–99, 2018.
- [6] F. Villada, E. García, and J. D. Molina, “Pronóstico del precio de la energía eléctrica usando redes neuro-difusas,” *Inf. Tecnol.*, vol. 22, no. 6, pp. 111–120, 2011, doi: 10.4067/S0718-07642011000600012.
- [7] J. Lago, F. De Ridder, and B. De Schutter, “Forecasting spot electricity prices: Deep learning approaches and empirical comparison of traditional algorithms,” *Appl. Energy*, vol. 221, no. February, pp. 386–405, 2018, doi: 10.1016/j.apenergy.2018.02.069.
- [8] J. Lago, M. Grzegorz, and B. De Schutter, “Forecasting day-ahead electricity prices: A review of state-of-the-art algorithms, best practices and an open-access benchmark,” *Appl. Energy*, vol. 293, no. April, p. 116983, 2021.
- [9] L. Tschora, E. Pierre, M. Plantevit, and C. Robardet, “Electricity price forecasting on the day-ahead market using machine learning,” *Appl. Energy*, vol. 313, no. December 2021, 2022, doi: 10.1016/j.apenergy.2022.118752.
- [10] A. Poggi, L. Di Persio, and M. Ehrhardt, “Electricity Price Forecasting via Statistical and Deep Learning Approaches: The German Case,” *AppliedMath*, vol. 3, no. 2, pp. 316–342,

2023, doi: 10.3390/appliedmath3020018.

- [11] B. G. Magalhães, P. M. R. Bento, J. A. N. Pombo, M. R. A. Calado, and S. J. P. S. Mariano, “Spot price forecasting for best trading strategy decision support in the Iberian electricity market,” *Expert Syst. Appl.*, vol. 224, no. April, 2023, doi: 10.1016/j.eswa.2023.120059.
- [12] XM Compania de Expertos en Mercados, “Página de Sinergox.” https://github.com/EquipoAnaliticaXM/API_XM.
- [13] XM Compania de Expertos en Mercados, “Página de XM.” <https://www.xm.com.co/>.
- [14] Nicholas Jhana, “Multi-Variate Time Series Forecasting Tensorflow,” *Kaggle*, 2020. <https://www.kaggle.com/code/nicholasjhana/multi-variate-time-series-forecasting-tensorflow>.