



**BENEMÉRITA UNIVERSIDAD AUTÓNOMA DE
PUEBLA**

FACULTAD DE CIENCIAS FÍSICO MATEMÁTICAS

LICENCIATURA EN MATEMÁTICAS APLICADAS

ANÁLISIS DE LA RADIACIÓN SOLAR EN EL MUNICIPIO
DE PUEBLA USANDO LA METODOLOGÍA DE
BOX-JENKINS Y REDES NEURONALES ARTIFICIALES

T E S I S

QUE PARA OBTENER EL TÍTULO DE:

LICENCIADO EN MATEMÁTICAS APLICADAS

PRESENTA:

JOSÉ MIGUEL HERNÁNDEZ LÓPEZ

DIRECTOR DE TESIS:

DR. BULMARO JUÁREZ HERNÁNDEZ

Heroica Puebla de Zaragoza

Junio de 2023

Título: Análisis de la radiación solar en el municipio de Puebla
usando la metodología de Box-Jenkins y Redes Neuronales Artificiales
Estudiante: JOSÉ MIGUEL HERNÁNDEZ LÓPEZ

COMITÉ

Dra. Hortensia Josefina Reyes Cervantes
Presidente

Dra. Gladys Linares Fleites
Secretario

Dr. Fernando Velasco Luna
Vocal

Dr. Bulmaro Juárez Hernández
Asesor

*Dedicado a mis padres
Gudelia López Merino y Margarito Hernández Sánchez.*

Agradecimientos

A Dios, por permitirme culminar esta meta que alguna vez creí inalcanzable.

A mis padres, que siempre me han brindado su apoyo incondicional para poder cumplir todos mis objetivos personales y académicos. Ellos son los que con su cariño me han impulsado siempre a perseguir mis metas y nunca abandonarlas frente a las adversidades. También son los que me han brindado el soporte material y económico para poder concentrarme en los estudios y nunca abandonarlos.

A mi hermana Hada Itzel, gracias por su ayuda incondicional, por su comprensión y apoyo en este tiempo.

A Karina Ortega Navarro, por siempre creer en mí y sacar mi mejor versión, por apoyarme porque cuando yo creía que este sueño era imposible, ella siempre me dejó en claro que ve en mí a un excelente profesionalista.

A mi director de tesis, Dr. Bulmaro Juárez Hernández a quien admiro y respeto por su labor como docente e investigador. Gracias por su dedicación y paciencia, sin sus palabras y correcciones precisas no hubiese podido lograr llegar a esta meta tan anhelada.

A mis sinodales, Dra. Hortensia Josefina Reyes Cervantes, Dra. Gladys Linares Fleites y Dr. Fernando Velasco Luna, por los conocimientos transmitidos, por haber aceptado revisar este trabajo y por sus aportaciones al mismo.

Índice general

| | |
|---|------------|
| Introducción | XVI |
| 1. Conceptos Meteorológicos | 1 |
| 1.1. Clima | 1 |
| 1.2. Variables atmosféricas | 3 |
| 1.3. El sol | 5 |
| 1.4. Radiación solar | 6 |
| 1.4.1. Tipos de radiación solar | 7 |
| 1.4.2. Unidades de medida de la radiación solar | 8 |
| 1.5. Medición de la radiación solar | 8 |
| 1.6. Mapa mundial de la radiación solar | 9 |
| 1.7. Radiación solar en México | 9 |
| 2. Modelos de series de tiempo | 11 |
| 2.1. Procesos estocásticos | 11 |
| 2.2. Series de Tiempo | 12 |
| 2.3. Procesos estacionarios y autocorrelación | 13 |
| 2.3.1. Series de tiempo estrictamente y débilmente estacionaria | 14 |
| 2.3.2. Autocovarianza, autocorrelación y autocorrelación parcial | 15 |
| 2.3.3. Ejemplos de series de tiempo estacionarias | 18 |
| 2.4. Estimación de la media, autovarianzas y autocorrelaciones | 21 |
| 2.5. Modelo Autorregresivo de Medias Móvil (ARMA) | 22 |
| 2.5.1. Introducción | 22 |
| 2.5.2. Procesos autorregresivos AR(p) | 24 |
| 2.5.3. Proceso AR(1) | 25 |
| 2.5.4. Procesos de media móvil | 28 |
| 2.5.5. Proceso MA(1) | 28 |
| 2.5.6. Proceso Autorregresivo de Medias Móviles (ARMA) | 31 |
| 2.5.7. Proceso ARMA(1,1) | 32 |
| 2.6. Modelo Autorregresivo de Medias Móviles Integrado ARIMA | 35 |
| 2.6.1. Modelo autorregresivo integrado de medias móviles de orden p, d, q ARIMA(p, d, q) | 36 |
| 2.6.2. Modelo ARIMA(0,1,0) o modelo de caminata aleatoria | 36 |
| 2.6.3. Varianza y Autovarianza de los modelos ARIMA | 38 |
| 2.6.4. Análisis de estacionariedad | 39 |

| | |
|---|-----------|
| 2.7. Modelos multiplicativos | 42 |
| 2.7.1. Modelo SARIMA(0,1,1)(0,1,1) ₁₂ | 43 |
| 2.8. Metodología Box-Jenkins | 44 |
| 2.8.1. Estacionariedad | 44 |
| 2.8.2. Identificación | 45 |
| 2.8.3. Estimación | 46 |
| 2.8.4. Diagnóstico | 46 |
| 2.8.5. Pronóstico | 46 |
| 3. Redes Neuronales Artificiales | 48 |
| 3.1. Machine learning | 48 |
| 3.2. Conjunto de entrenamiento y conjunto de prueba (Training set y Test set) | 49 |
| 3.3. Sobreajuste u overfitting | 49 |
| 3.4. Normalización de los datos | 50 |
| 3.4.1. Técnicas de normalización | 51 |
| 3.5. Perceptrón | 54 |
| 3.6. Función de activación | 55 |
| 3.7. Red neuronal artificial | 59 |
| 3.8. La arquitectura de las redes neuronales | 59 |
| 3.9. Función de Costo | 62 |
| 3.9.1. Error cuadrático medio | 63 |
| 3.9.2. Error absoluto medio | 63 |
| 3.10. Aprendizaje con Descenso de Gradiente (Gradient descent) | 64 |
| 3.10.1. Descenso de gradiente estocástico (Stochastic Gradient Descent) | 68 |
| 3.11. Aprendizaje con Backpropagation | 69 |
| 3.11.1. Algoritmo | 72 |
| 3.12. Aprendizaje profundo | 73 |
| 3.13. Optimizadores alternativos | 73 |
| 3.13.1. Descenso de Gradiente con momentum | 73 |
| 3.13.2. RMSProp (Root Mean Square Propagation) | 74 |
| 3.13.3. ADAM (Adaptive Moment Estimation) | 75 |
| 3.14. Redes Neuronales Recurrentes (RNN) | 75 |
| 3.15. Arquitecturas variantes de RNN | 76 |
| 3.16. Red de memoria a largo plazo a corto plazo (LSTM) | 76 |
| 3.16.1. Arquitectura de red de LSTM | 77 |
| 4. Caso de estudio | 78 |
| 4.1. Zona de estudio | 78 |
| 4.2. Análisis del conjunto de datos de la radiación solar del municipio de Puebla | 81 |
| 4.2.1. Análisis usando la metodología Box-Jenkins | 81 |
| 4.2.2. Análisis usando Redes Neuronales Artificiales | 94 |
| 4.2.3. Preparación de los datos | 95 |
| 4.2.4. Análisis de los datos para la creación de la Red | 96 |
| 4.2.5. Creación de la Red Neuronal Artificial | 97 |
| 4.2.6. Descripción de la Red Neuronal Artificial | 99 |
| 4.2.7. Pronóstico con la Red Neuronal Artificial | 100 |

| | |
|--|------------|
| <i>ÍNDICE GENERAL</i> | XI |
| Conclusión | 102 |
| 5. Análisis y conclusiones | 102 |
| 5.1. Análisis y Resultados | 104 |
| 5.2. Conclusiones | 108 |
| A. Prueba de Dickey–Fuller | 109 |
| B. Prueba de Ljung-Box | 112 |
| C. Código en Python para el análisis de la Metodología Box-Jenkins | 113 |
| D. Código en Python para el análisis de Redes Neuronales Artificiales | 117 |
| E. Selección de datos | 120 |
| Bibliografía | 124 |

Índice de figuras

| | |
|---|----|
| 1.1. Estación de monitoreo atmosférico del municipio de Puebla (Figura extraída de https://calidaddelaire.puebla.gob.mx/views/principal_monitoreo.php). | 2 |
| 1.2. El sistema climático de la Tierra (Figura extraída de [1]). | 3 |
| 1.3. Estructura del Sol (Figura extraída de https://www.todamateria.com/el-sol/). | 6 |
| 1.4. Espectro de radiación electromagnética solar (Figura extraída de https://cienciaycosmetica.com/2020/09/23/luz-visible-azul-infrarrojos-cosmetica-ciencia/). | 7 |
| 1.5. Instrumentos para la medición de la radiación solar (Figuras extraídas de https://es.wikipedia.org/wiki/Panel_fotovoltaico y https://es.wikipedia.org/wiki/Piran%C3%B3metro). | 8 |
| 1.6. Mapa mundial de radiación solar (Figura extraída de [9]). | 10 |
| 1.7. Mapa de Irradiación Solar en México (Elaboro: Ing. Manuel Muñoz Herrera. Datos: Servicio Meteorológico Nacional). | 10 |
| 2.1. Ejemplos del comportamiento de la FAC (De autoría propia). | 17 |
| 2.2. Ejemplo del proceso de ruido blanco (De autoría propia). | 20 |
| 2.3. Ejemplo del proceso $AR(2)$ con $\phi_1 = 1.5$ y $\phi_2 = -0.75$ (De autoría propia). | 20 |
| 2.4. Ejemplo del proceso $AR(1)$ con $\phi_1 = 0.8$ (De autoría propia). | 27 |
| 2.5. Ejemplo del proceso $AR(1)$ con $\phi_1 = -0.8$ (De autoría propia). | 27 |
| 2.6. Ejemplo del proceso $MA(1)$ con $\theta_1 = 0.8$ (De autoría propia). | 30 |
| 2.7. Ejemplo del proceso $MA(1)$ con $\theta_1 = -0.8$ (De autoría propia). | 30 |
| 2.8. Ejemplo del proceso $ARMA(1,1)$ con $\theta_1 = 0.5$ y $\phi_1 = 0.4$ (De autoría propia). | 33 |
| 2.9. Ejemplo del proceso $ARMA(1,1)$ con $\theta_1 = 0.5$ y $\phi_1 = -0.4$ (De autoría propia). | 33 |
| 2.10. Serie histórica del tipo de cambio peso dólar desde 1954, fecha de consulta: 27/05/2023 (Datos extraídos del Banco de México). | 35 |
| 2.11. Serie histórica de Remesas Totales (Millones de dólares), fecha de consulta: 27/05/2023 (Datos extraídos del Banco de México). | 35 |
| 2.12. Trayectoria y FAC de un proceso $ARIMA(0,1,0)$ (De autoría propia). | 37 |
| 2.13. Trayectoria y FAC de un modelo de caminata aleatoria (De autoría propia). | 37 |
| 2.14. Ejemplo de la transformación Box-Cox (De autoría propia). | 41 |
| 2.15. Ejemplo de transformación por diferencias (De autoría propia). | 41 |
| 2.16. Datos de pasajeros aéreos (Datos extraídos de Kaggle). | 42 |
| 2.17. Diagrama de Flujo de la metodología Box-Jenkins (De autoría propia). | 47 |

| | |
|--|-----|
| 3.1. Técnicas de normalización (Figura extraída de [32]). | 51 |
| 3.2. Ejemplo de la técnica escalamiento logarítmico (Figura extraída de [32]). . . | 53 |
| 3.3. Diagrama de perceptrón (De autoría propia). | 54 |
| 3.4. Comportamiento de una neurona artificial (De autoría propia). | 55 |
| 3.5. Gráfica de la función lineal. | 56 |
| 3.6. Gráfica de la función escalón. | 57 |
| 3.7. Gráfica de la función sigmoide. | 57 |
| 3.8. Gráfica de la función ReLu. | 58 |
| 3.9. Gráfica de la función Tangente hiperbólica. | 59 |
| 3.10. Ejemplos de redes neuronales (De autoría propia). | 60 |
| 3.11. Tipos de redes neuronales (De autoría propia). | 61 |
| 3.12. Ejemplo del funcionamiento del algoritmo Descenso de Gradiente (Figura extraída de [34]). | 65 |
| 3.13. Segundo ejemplo del funcionamiento del algoritmo Descenso de Gradiente (Figura extraída de [34]). | 66 |
| 3.14. Ejemplo del uso de la notación para el algoritmo Backpropagation (Figura extraída de [34]). | 70 |
| 3.15. Arquitectura de un bloque LSTM (Figura extraída de [35]). | 77 |
| 3.16. Diagrama de una red LSTM (Figura extraída de [36]). | 77 |
| | |
| 4.1. Localización geográfica del municipio de Puebla, Puebla (Figura extraída del Instituto Nacional de Estadística y Geografía (INEGI)). | 79 |
| 4.2. Imagen de NOAA. | 81 |
| 4.3. Gráfica de la serie de tiempo de la radiación solar. | 81 |
| 4.4. Descomposición aditiva de la serie de tiempo de la radiación solar (De au- toría propia). | 82 |
| 4.5. Parte estacional de la descomposición aditiva (De autoría propia). | 82 |
| 4.6. Grafica de Cajas y bigotes (De autoría propia). | 83 |
| 4.7. <i>FAC</i> Y <i>FACP</i> de la serie de tiempo. | 86 |
| 4.8. Trayectoria, <i>FAC</i> y <i>FACP</i> de la serie de tiempo de la radiación solar con una diferencia regular (De autoría propia). | 87 |
| 4.9. Trayectoria, <i>FAC</i> y <i>FACP</i> de la serie de tiempo de la radiación solar dife- renciada regularmente y estacionalmente (De autoría propia). | 88 |
| 4.10. Serie original y ajustada (De autoría propia). | 91 |
| 4.11. Pronóstico de 5 semanas con la metodología Box-Jenkins (De autoría propia). 92 | |
| 4.12. Gráfico de residuales del modelo <i>SARIMA</i> (1, 1, 1)(0, 1, 1) ₅₂ (De autoría pro- pia). | 93 |
| 4.13. Gráfica del conjunto de datos de la radiación solar (De autoría propia). . . . | 94 |
| 4.14. Gráfica de la serie de tiempo de la radiación solar (De autoría propia). . . . | 96 |
| 4.15. Resultados de distintas redes neuronales (De autoría propia). | 98 |
| 4.16. Estructura de la red neuronal artificial implementada (De autoría propia). . . | 99 |
| 4.17. Función de costo (De autoría propia). | 100 |
| 4.18. Comparación entre los datos reales de la variable meteorológica radiación solar del municipio de Puebla y el pronóstico de la RNN (De autoría propia). 101 | |
| | |
| 5.1. Comparación gráfica entre los pronósticos obtenidos (De autoría propia). . . | 104 |

| | |
|---|-----|
| 5.2. Comparación gráfica entre los datos reales y los pronósticos obtenidos por las dos metodologías (De autoría propia). | 106 |
| 5.3. Diagramas de densidad generadas por el método Kernel (De autoría propia). | 107 |
| E.1. Datos del 17 de julio de 2020. | 122 |
| E.2. Datos faltantes 2022. | 122 |
| E.3. Datos faltantes en todos los años. | 123 |
| E.4. Conjunto de datos semanalmente. | 123 |

Índice de tablas

| | |
|--|-----|
| 2.1. Clasificación de los procesos estocásticos (Tabla extraída de [24]). | 12 |
| 2.2. Ejemplos de análisis de series de tiempo por campos científicos (De autoría propia). | 13 |
| 2.3. Propiedades de la función de autocovarianza y función de autocorrelación (Tabla extraída de [24]). | 16 |
| 2.4. Resumen de la función de autocorrelación y la función de autocorrelación parcial para los diferentes procesos (Tabla extraída de [22]). | 34 |
| 2.5. Transformaciones generales (Tabla extraída de [22]). | 40 |
| 4.1. Versión 2 de la prueba de Dickey–Fuller para la serie de tiempo (De autoría propia). | 86 |
| 4.2. Versión 1 de la prueba de Dickey–Fuller para la serie con una diferencia regular (De autoría propia). | 89 |
| 4.3. Versión 1 de la prueba de Dickey–Fuller para la serie con una diferencia regular y una diferencia estacional (De autoría propia). | 89 |
| 4.4. Diferentes modelos propuestos (De autoría propia). | 89 |
| 4.5. Valores obtenidos del modelo (De autoría propia). | 90 |
| 4.6. Resultados con la metodología Box-Jenkins (De autoría propia). | 91 |
| 4.7. Conjunto de datos de entrenamiento (De autoría propia). | 96 |
| 4.8. Conjunto de datos de prueba (De autoría propia). | 97 |
| 4.9. Resumen estadístico descriptivo (De autoría propia). | 97 |
| 4.10. Comparación entre los datos reales y los datos pronosticados de la RNN, y los errores (De autoría propia). | 100 |
| 5.1. Pronósticos de las primeras 5 semanas del año 2023 (De autoría propia). . . | 104 |
| 5.2. Datos reales obtenidos en la página web de SINAICA y pronósticos obtenidos por la metodología Box-Jenkins y RNN para las primeras 5 semanas del año 2023 (De autoría propia). | 105 |
| 5.3. Comparación de los pronósticos obtenidos para las primeras 5 semanas del año 2023 (De autoría propia). | 106 |

Introducción

La energía proveniente del sol, se denomina energía solar, aunque se conoce como radiación solar, esta energía es emitida en forma de radiación electromagnética que llega a la atmósfera. La medición de la radiación solar de cada estación meteorológica es presentada en unidades de potencia y está en vatios por metro cuadrado (W/m^2).

El objetivo principal de este trabajo es obtener predicciones a corto plazo de la variable meteorológica radiación solar del municipio de Puebla, para esto se hizo uso de un conjunto de datos los cuales fueron recolectados en una estación meteorológica de la zona norte del municipio de Puebla localizada en la Universidad Tecnológica de Puebla. El conjunto de datos está conformado con información a partir del 1 de enero de 2020 hasta el 5 de febrero de 2023, este conjunto de datos su frecuencia es de hora, es decir los datos fueron recolectados cada hora, después se agruparon los datos y fueron promediados semanalmente.

Se realiza el análisis de los datos usando series de tiempo, una serie de tiempo la podemos definir como una secuencia de observaciones sobre intervalos de tiempo separados de manera regular. Existen varios métodos para el estudio de las series de tiempo, en este trabajo se usarán dos metodologías distintas las cuales son:

- Metodología de Box – Jenkins.
- Redes Neuronales Artificiales.

La metodología Box - Jenkins se puede definir como un ciclo iterativo que busca encontrar el mejor modelo de ajuste para la realización de pronósticos. Esta metodología es nombrada así en honor a los estadísticos George Edward Pelham Box y Gwilym Meirion Jenkins. Mientras que las Redes neuronales artificiales es un método de la inteligencia artificial que enseña a las computadoras a procesar datos de una manera que está inspirada en la forma en que lo hace el cerebro humano, estas han sido utilizadas exitosamente en varios distintos problemas por ejemplo en la clasificación de imágenes, en el reconocimiento de dígitos, predicciones, etc. En el caso de la predicción se ha demostrado que las redes neuronales artificiales son capaces de modelar y predecir series de tiempo lineales y no líneas con un alto grado de precisión.

Mediante estas dos metodologías se hallan dos distintos modelos los cuales son seleccionados por el mejor que se ajusta a los datos para después obtener los pronósticos

para la radiación solar.

Esta tesis inicia presentando algunos conceptos meteorológicos, lo cual se presenta en el Capítulo 1, donde se describen algunos conceptos como el clima, las variables atmosféricas, la radiación solar, instrumentos de medición de la radiación solar, sus unidades, etc. En el Capítulo 2, se mencionan los modelos de series de tiempo, iniciando por el concepto de proceso estocástico posteriormente se introduce la definición de serie de tiempo, y se explican más definiciones como las series de tiempo estacionarias, la función de autocovarianza, autocorrelación y autocorrelación parcial, la metodología Box-Jenkins, entre otros. En el Capítulo 3, se explica la teoría y las definiciones para el entendimiento de las redes neuronales artificiales, donde se incluye como es el tratamiento de los datos, como aprenden las redes neuronales, los tipos de arquitecturas para una red neuronal. En el Capítulo 4, se aplican los conceptos vistos en los capítulos anteriores para el análisis de los datos de la radiación solar del municipio de Puebla. Y, por último, en el capítulo 5 se presentan los análisis y conclusiones de este trabajo.

Capítulo 1

Conceptos Meteorológicos

Nuestra atmósfera puede compararse a una campana vibrante. Responde presta y distintivamente a ciertos impulsos exteriores, pero lo hace a su manera. Las respuestas de la atmósfera a las actividades solares sean constantes, cíclicas o esporádicas, depende de muchos factores, tales como las dimensiones y la velocidad de rotación de la Tierra, su distancia al Sol, el campo magnético y la composición química de la atmósfera [11].

La atmósfera cuenta con cinco capas: la Tropósfera, la Estratósfera, la Mesósfera, la Termósfera y Exosfera, está compuesta de gases y de partículas sólidas y líquidas en suspensión atraídas por la gravedad terrestre.

En ella se producen todos los fenómenos climáticos y meteorológicos que afectan al planeta, regula la entrada y salida de energía de la tierra y es el principal medio de transferencia del calor.

1.1. Clima

El clima terrestre es producto de la interacción entre la atmósfera, los océanos, las capas de hielo y nieve, los continentes y, muy importante, la vida en el planeta.

El clima es el estado más frecuente de la atmósfera de un lugar de la superficie terrestre; es decir, una descripción estadística de las condiciones meteorológicas más frecuentes de una región en cierto periodo de tiempo. En un sentido amplio, el clima se refiere al estado del sistema climático como un todo, incluyendo sus variaciones y descripciones estadísticas [1].

El clima y el estado del tiempo

Mientras el tiempo también denominado tiempo meteorológico o simplemente tiempo se refiere a las variaciones diarias en las condiciones atmosféricas de nuestro planeta, el clima se refiere al estado más frecuente de la atmósfera de una localidad. Para conocer el clima de un país es necesario medir diariamente por al menos tres décadas las condiciones de temperatura, lluvia, humedad y viento, observar las condiciones de nubosidad, la

trayectoria de los huracanes, las masas de aire frío, etc. [1].

Para conocer el tiempo existen las estaciones meteorológicas, y para conocer el clima, las estaciones climatológicas, en la Figura 1.1 se muestra una fotografía de una estación meteorológica y climatológica ubicada en la zona norte del municipio de Puebla, específicamente en la Universidad Tecnológica de Puebla (UTP).



Figura 1.1: Estación de monitoreo atmosférico del municipio de Puebla (Figura extraída de https://calidaddelaire.puebla.gob.mx/views/principal_monitoreo.php).

¿Cómo conocemos el clima y el estado del tiempo?

Tradicionalmente, se ha conocido el clima y el tiempo atmosférico a través del estudio de las variables que los afectan de manera más directa, como son la temperatura atmosférica, el viento que se encuentra cerca de la superficie de la Tierra, las precipitaciones en sus distintas formas (lluvia, nieve, granizo), humedad, tipo y cantidad de nubes, y la radiación solar. Estas variables son observadas cada hora por una gran cantidad de estaciones climatológicas y meteorológicas alrededor del mundo y generalmente la información se expresa por medio de mapas que permiten mostrar la evolución temporal y la distribución espacial del estado atmosférico. Para comprender el clima de la Tierra, sus variaciones y tener la posibilidad de proyectar los cambios climáticos producidos por las actividades humanas, no podemos ignorar ninguno de los diversos factores y componentes que lo determinan, y que permiten describirlo. Esto incluye la dinámica y composición de la atmósfera, el océano, el hielo, la nieve, la superficie terrestre, y los procesos biológicos que acontecen dentro de ellos [2].

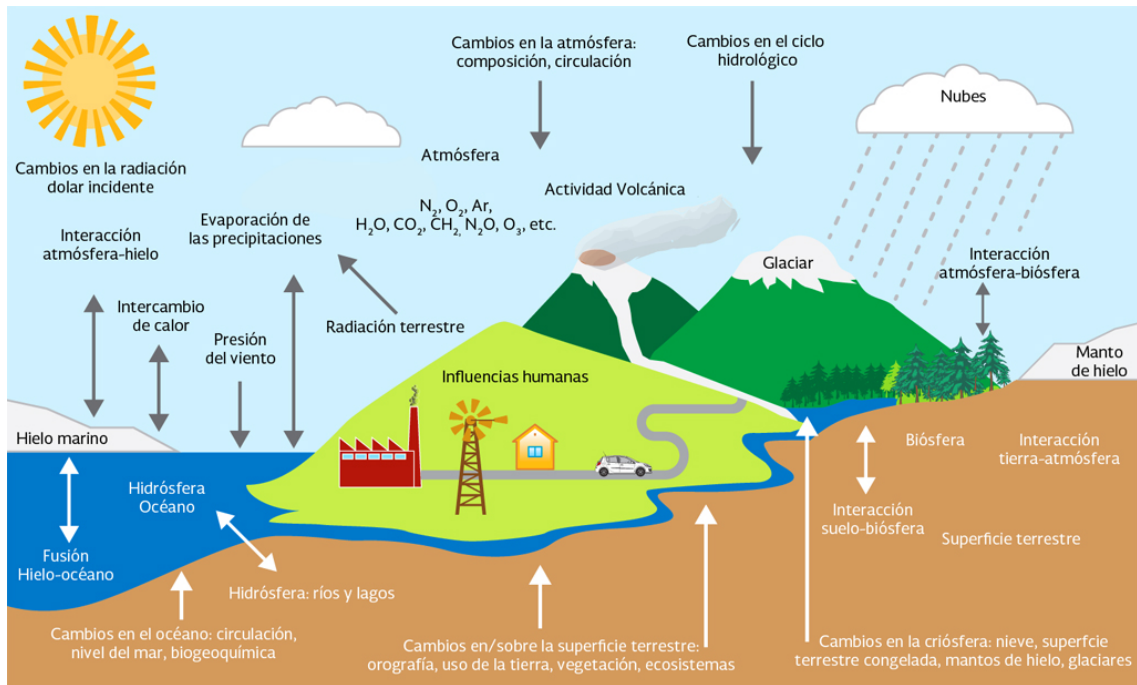


Figura 1.2: El sistema climático de la Tierra (Figura extraída de [1]).

1.2. Variables atmosféricas

Anteriormente ya mencionamos cuales son las variables atmosféricas ahora las explicaremos a detalle.

Temperatura

Es bien sabido que la temperatura es una de las magnitudes más utilizadas para describir el estado de la atmósfera. Sabemos que las temperaturas varían del día a la noche, de una estación a otra y de la ubicación geográfica. Puede estar por debajo de $0^{\circ}C$ en invierno y por encima de $40^{\circ}C$ en verano.

Formalmente, la temperatura es una magnitud relacionada con la rapidez del movimiento de las partículas que constituyen la materia. Cuanta mayor agitación presente éstas, mayor será la temperatura.

Para medir la temperatura, tenemos que basarnos en propiedades de la materia que se ven alteradas cuando ésta cambia: la resistencia eléctrica de algunos materiales, el volumen de un cuerpo, el color de un objeto, etc. El instrumento que se utiliza para medir la temperatura se llama termómetro y fue inventado por Galileo en 1593. Como toda magnitud física, la temperatura tiene una unidad de medida asociada, dependiendo de la escala que elijamos, estas unidades son Celsius ($^{\circ}C$), Fahrenheit ($^{\circ}F$) y Kelvin (K). [3].

Precipitación

Se denomina precipitación al agua de la atmósfera que cae en forma líquida, sólida, o líquida y sólida desde las nubes hasta la superficie de la tierra.

Las gotas de agua son de diámetro muy pequeño del orden de las 5 micras, el tamaño de las gotas de lluvia apenas tiene 0.1 mm de diámetro por lo que las gotas iniciales deben aumentar de volumen y uno de los procesos de crecimiento es por colisión. La precipitación se puede presentar en las formas siguientes: Lluvia, nieve o granizo.

Para que las gotas de agua que forman las nubes lleguen a precipitarse, se requiere que éstas aumenten de tamaño y que el aire situado debajo de las nubes no sea demasiado cálido, ni muy seco, ya que de ser así, las gotas de agua que inician la caída pueden volver a evaporarse.

La precipitación se puede dar también en forma sólida. El origen de la misma está en la formación de cristales de hielo en las nubes que tienen su tope a grandes alturas y bajísimas temperaturas (-40°C) [4].

Presión

La presión en un punto de la atmósfera es simplemente el peso de la columna de aire de base unidad sobre ese punto. La presión atmosférica en superficie será el peso de toda la columna que se alarga desde el suelo hasta el límite superior de la atmósfera.

La presión atmosférica depende de muchas variables, sobre todo de la altitud. Cuanto más arriba en la atmósfera nos encontremos, la cantidad de aire por encima nuestro será menor, lo que hará que también sea menor la presión que éste ejerza sobre un cuerpo ubicado allí.

La presión disminuye conforme aumenta la altura ya que los gases que componen la atmósfera pierden densidad; así cuanto más alto esté un punto de la superficie terrestre menos espesor de atmósfera soporta encima, entonces será menor la presión que se ejerce sobre este punto [4].

Humedad relativa

La humedad relativa es la relación entre la cantidad de vapor de agua que realmente hay en el aire y la cantidad máxima de vapor de agua requerida para la saturación a esa temperatura (y presión) particular. En otras palabras, es la relación entre el contenido de vapor de agua del aire y su capacidad.

La humedad relativa se da como un porcentaje. El aire con una humedad relativa del 50 por ciento en realidad contiene la mitad de la cantidad requerida para la saturación. Se dice que el aire con una humedad relativa del 100 por ciento está saturado porque está completamente lleno de vapor de agua. Se dice que el aire con una humedad relativa superior al 100 por ciento está sobresaturado [5].

Velocidad y dirección del viento

La velocidad del viento describe qué tan rápido se mueve el aire más allá de cierto punto. Esto puede ser un promedio de una unidad de tiempo determinada, como metros por hora, o una velocidad instantánea, que se informa como velocidad máxima del viento.

La dirección del viento describe la dirección en una brújula de la que emana el viento, por ejemplo, del norte o del oeste.

La velocidad del viento generalmente se mide en millas por hora, nudos o metros por segundo y la dirección del viento generalmente se mide en grados y describe la dirección desde la que emana el viento [6].

1.3. El sol

El Sol es una estrella que pertenece a nuestro Sistema Solar. Su origen, como el de la Tierra, fue una colección de gases incandescentes que al girar rápidamente provocó que los gases más pesados fluyan hacia el centro y los más ligeros hacia el exterior.

En comparación con los miles de millones de estrellas del universo, se ignora el paso del Sol. Sin embargo, para la Tierra y los demás planetas que la rodean, el sol es un poderoso punto focal: su luz da vida, calor y mantiene unido el sistema solar.

El Sol es una estrella enorme. Con un diámetro de 1.4 millones de kilómetros podría albergar a 109 planetas en su superficie. Si fuera hueco, más de un millón de Tierras podrían vivir en su interior, pero no lo es. Está relleno de gases calientes que representan más del 99.8 por ciento de la masa total del sistema solar. ¿A qué llamamos caliente? La temperatura alcanza los 5,500 grados centígrados en la superficie y más de 15.5 millones de grados centígrados en el núcleo [8].

En el núcleo del Sol, se producen reacciones de fusión en las que el hidrógeno se transforma en helio, que genera la energía. Unas pequeñas partículas de luz llamadas fotones transportan esta energía a través de la zona radiante hasta la capa superior del interior del Sol, la zona convectiva. Ahí, el movimiento de los gases hirviendo (como en una lámpara de lava) lleva la energía a la superficie.

La superficie del Sol, o atmósfera, está dividida en tres regiones: la fotosfera, la cromosfera y la corona solar. La fotosfera es la superficie visible del Sol y la capa más baja de la atmósfera. Por encima de la fotosfera se encuentran la cromosfera y la corona, que también emiten luz visible pero solamente se pueden ver durante eclipses solares, cuando la luna pasa entre la Tierra y el Sol [8].

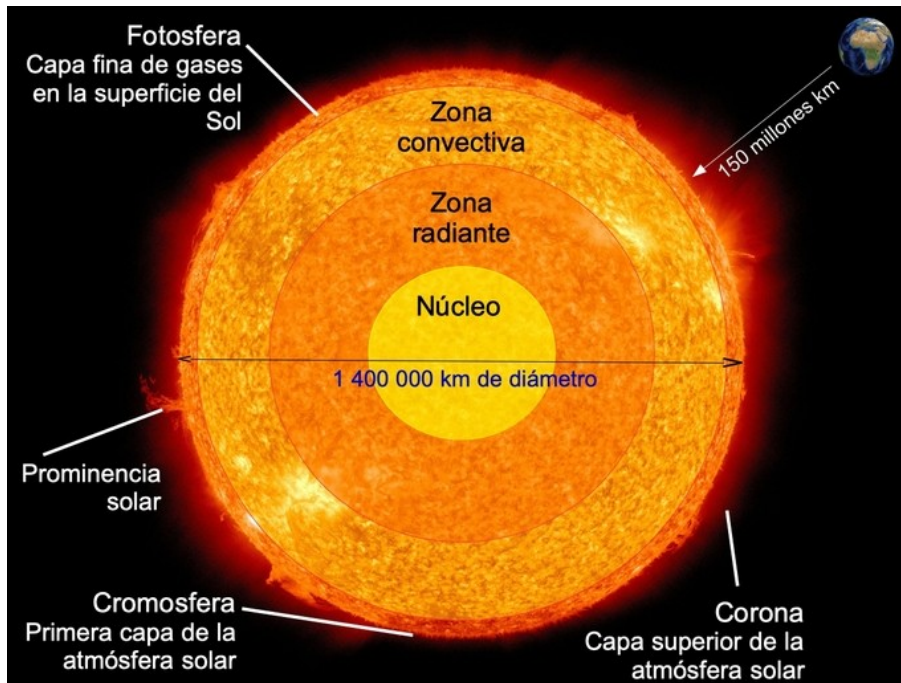


Figura 1.3: Estructura del Sol (Figura extraída de <https://www.todamateria.com/el-sol/>).

1.4. Radiación solar

La radiación solar la podemos definir como el flujo de energía que recibimos del Sol en forma de ondas electromagnéticas que permite la transferencia de energía solar a la superficie terrestre. Estas ondas electromagnéticas son de diferentes frecuencias y aproximadamente la mitad de las que recibimos están entre los rangos de longitud de onda de $0.4 \mu\text{m}$ y $0.7 \mu\text{m}$, y pueden ser detectadas por el ojo humano, constituyendo lo que conocemos como luz visible ver Figura 1.4. De la otra mitad, la mayoría se sitúa en la parte infrarroja del espectro y una pequeña parte en la ultravioleta [9].

La radiación solar es emitida por la superficie solar, esta energía determina la dinámica de los procesos atmosféricos y climatológicos. Además, es directa o indirectamente responsable de circunstancias de nuestro día a día, como la fotosíntesis de las plantas, el mantenimiento de una temperatura compatible con la vida y de la formación del viento.

El Sol emite energía en forma de radiación de onda corta y ésta sufre un proceso de debilitamiento en la atmósfera por la presencia de las nubes y su absorción por parte de moléculas de gases o de partículas en suspensión. Tras superar la atmósfera, la radiación solar alcanza la superficie terrestre oceánica y continental, reflejándose o absorbiéndose. Finalmente, la superficie la devuelve con dirección al espacio exterior en forma de radiación de onda larga [10].

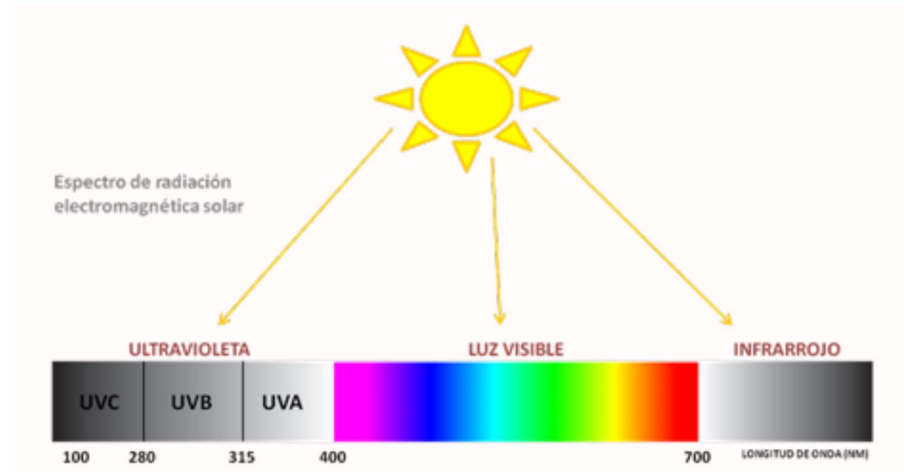


Figura 1.4: Espectro de radiación electromagnética solar (Figura extraída de <https://cienciaycosmetica.com/2020/09/23/luz-visible-azul-infrarrojos-cosmetica-ciencia/>).

1.4.1. Tipos de radiación solar

La radiación solar se puede clasificar de dos distintas maneras:

Según la forma en la que llega a la Tierra:

- Radiación solar directa. Este tipo de radiación es la que traspasa la atmósfera y alcanza la superficie de la Tierra sin haber sufrido dispersión alguna en su trayectoria.
- Radiación solar difusa. Es la radiación que alcanza la superficie de la Tierra tras haber sufrido múltiples desviaciones en su trayectoria, por ejemplo, por los gases presentes en la atmósfera.
- Radiación solar reflejada. Es aquella fracción de radiación solar que es reflejada por la propia superficie terrestre, en un fenómeno conocido como efecto albedo.

Según los tipos de rayos:

- Rayos infrarrojos (IR). De mayor longitud de onda que la luz visible, emiten calor y cualquier cuerpo cuya temperatura sea mayor a 0° Kelvin los desprende.
- Rayos visibles (VI). Emiten luz y son aquellos que el ojo humano percibe en forma de colores (rojo, naranja, amarillo, verde, cian, azul y violeta).
- Rayos ultravioleta (UV). Son invisibles al ojo humano y son los que inciden más gravemente sobre la piel (quemaduras, manchas, arrugas). Se dividen en tres subcategorías:
 - Ultravioleta (UVA). Son aquellos que atraviesan con facilidad la atmósfera, alcanzando mayoritariamente la superficie terrestre.



(a) Celdas fotovoltaicas.



(b) Piranómetros.

Figura 1.5: Instrumentos para la medición de la radiación solar (Figuras extraídas de https://es.wikipedia.org/wiki/Panel_fotovoltaico y <https://es.wikipedia.org/wiki/Piran%C3%B3metro>).

- Ultravioleta B (UVB). Tienen mayores dificultades para traspasar la atmósfera. Aun así, llegan a la superficie terrestre y son los que pueden producir daños más graves en la piel.
- Ultravioleta C (UVC). Este tipo de rayos no consiguen atravesar la atmósfera, ya que son absorbidos por la capa de ozono [10].

1.4.2. Unidades de medida de la radiación solar

Las cantidades de radiación son expresadas generalmente en términos de exposición radiante o irradiancia, siendo esta última una medida del flujo de energía recibida por unidad de área en forma instantánea como energía/área-tiempo y cuya unidad es el Watt por metro cuadrado (W/m^2). Un Watt es igual a un Joule por segundo.

Irradiancia Solar

La irradiancia solar es la potencia por unidad de área recibida del Sol en forma de radiación electromagnética medida en el rango de longitud de onda del instrumento de medición.

1.5. Medición de la radiación solar

La radiación solar se mide con distintos propósitos, uno de ellos es fundamental para una buena planificación, diseño e implementación de un proyecto de aplicación solar. Para la medición de la radiación solar existen diversos instrumentos, siendo los más comunes las celdas fotovoltaicas y los piranómetros (Figura 1.5) [7].

Piranómetros

Un piranómetro (también llamado solarímetro y actinómetro) es un instrumento meteorológico utilizado para medir de manera muy precisa la radiación solar incidente sobre la superficie de la Tierra. Los piranómetros absorben la radiación solar por medio de termocuplas que convierten el calor en voltaje eléctrico proporcional a la radiación.

Celda fotovoltaica

Una celda fotovoltaica es un dispositivo electrónico que convierte la energía luminosa, la luz, en energía eléctrica. Es decir, absorbe los fotones de la luz para liberar electrones que puede usar en una corriente eléctrica. La celda fotovoltaica en tanto es un pequeño panel de silicio conectado a una resistencia que genera una corriente proporcional a la radiación.

1.6. Mapa mundial de la radiación solar

El mapeado de la radiación solar media en el mundo, nos da una idea de qué zonas son las que reciben mayor radiación solar, (Véase Figura 1.6.). Éstas se encuentran principalmente en la zona del ecuador, difuminándose hacia los polos, aunque entran en juego distintos factores como la nubosidad o las características geológicas del suelo, refractando esta mayor o menor radiación, contribuyendo al efecto invernadero y al aumento de la temperatura [9].

En este mapa podemos ver que México, Estados Unidos y algunos países de Sudamérica son los que cuentan con mayor potencial solar en todo el continente.

Por otro lado, algunos países en donde menos radiación de energía solar hay es en donde actualmente más se está usando la energía solar. Por ejemplo, si vemos el mapa, Alemania está en color azul, lo que quiere decir que recibe menos de $150 W/m^2$ de energía solar. Ahora imaginémonos cuanta energía se puede aprovechar en lugares en donde la radiación supera los $350 W/m^2$ [9].

1.7. Radiación solar en México

El sol juega un papel cada vez más importante en nuestra economía, y puede llegar a ser una fuente de empleos para nuestro país. Actualmente en México se está construyendo la planta fotovoltaica más grande de Latinoamérica, ubicada en el desierto de Sonora. Esta planta contará con una capacidad de 1,000 megavatios y suministrará electricidad a 1.6 millones de hogares.

México es un país con alta incidencia de energía solar en la gran mayoría de su territorio; la zona norte es de las más soleadas del mundo. La mitad del territorio nacional presenta una insolación promedio de 5.3 KWh por metro cuadrado al día, suficiente para satisfacer la necesidad de un hogar mexicano promedio. Esto nos coloca en una situación muy favorable para el uso de la energía solar [9].

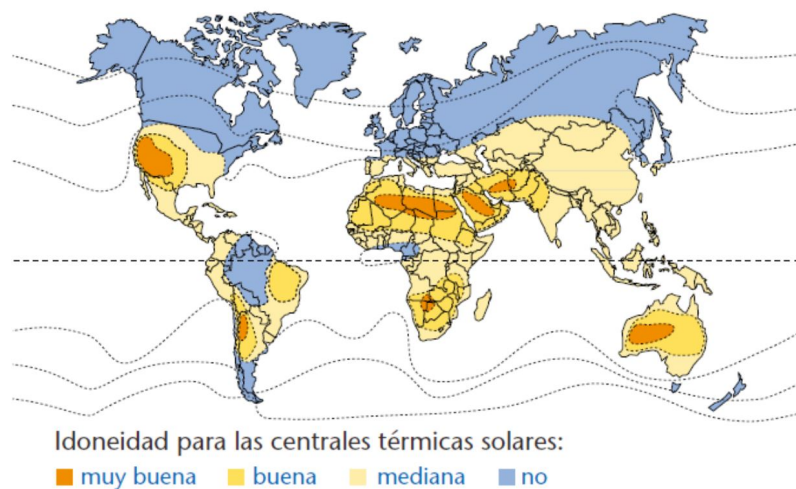


Figura 1.6: Mapa mundial de radiación solar (Figura extraída de [9]).

Hay muchos factores que influyen en la decisión de usar esta fuente de energía para resolver los requerimientos de energía eléctrica, pero las condiciones de muchas zonas del país permiten que las inversiones en el Noroeste de México, se antojen como las más rentables, la mayor cantidad de irradiación de que se dispone se encuentra en esta zona del país [9].

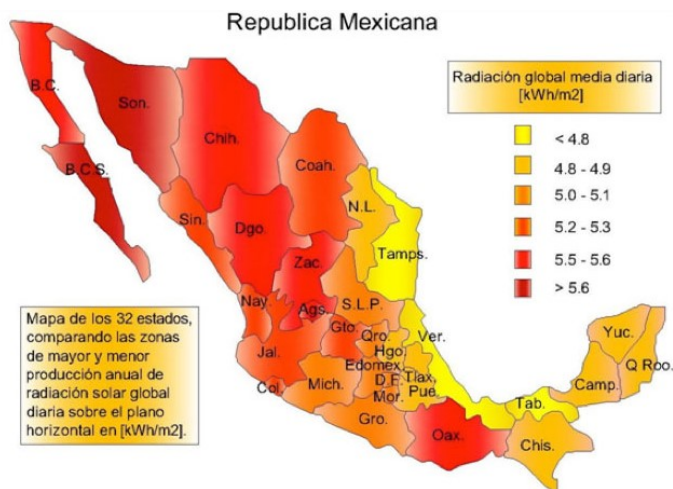


Figura 1.7: Mapa de Irradiación Solar en México (Elaboro: Ing. Manuel Muñoz Herrera. Datos: Servicio Meteorológico Nacional).

Capítulo 2

Modelos de series de tiempo

2.1. Procesos estocásticos

Un proceso estocástico es un concepto matemático que sirve para representar magnitudes aleatorias que varían con el tiempo o para caracterizar una sucesión de variables aleatorias que evolucionan en función de otra variable, generalmente el tiempo. Las series temporales se definen como un caso particular de los procesos estocásticos. En otras palabras, una serie de tiempo se considerará como una realización de un proceso estocástico, es por eso la importancia de darle una definición formal.

Definición 2.1.1 (Proceso estocástico). *Un proceso estocástico es una familia de variables aleatorias (X_t) , $t \in T$ indexadas por un conjunto T , definidas en algún espacio de probabilidad común (Ω, F, \mathbb{P}) , y con valores en un conjunto S llamado espacio de estados [20].*

Por tanto, para cada instante t tendremos una variable aleatoria distinta representada por X_t , con lo que un proceso estocástico puede interpretarse como una sucesión de variables aleatorias cuyas características pueden variar a lo largo del tiempo.

A los posibles valores que puede tomar la variable aleatoria se le denominaran estados, por lo que se puede tener un espacio de estados discreto y un espacio de estados continuo. Por otro lado, la variable tiempo puede ser de tipo discreto o de tipo continuo. Por tanto, dependiendo de cómo sea el conjunto de subíndices T y el tipo de variable aleatoria dado por X_t se puede establecer una clasificación de los procesos estocásticos, obsérvese la Tabla 2.1.

Una serie de tiempo es una realización de un proceso estocástico en tiempo discreto, donde los elementos de T están ordenados y corresponden a instantes equidistantes del tiempo.

| | | |
|------------------|---|---------------------|
| $T \backslash X$ | Discreta | Continua |
| Discreta | Cadenas de Markov, Procesos de ramificación | Serie de tiempo |
| Continua | Procesos de Poisson, Teoría de colas, Procesos de nacimiento y muerte | Procesos Brownianos |

Tabla 2.1: Clasificación de los procesos estocásticos (Tabla extraída de [24]).

2.2. Series de Tiempo

Definición 2.2.1 (Serie de tiempo). *Una serie de tiempo es una sucesión cronológica de observaciones X_t , en la que cada una de ellas corresponde a una variable aleatoria distinta, y la ordenación de la sucesión de observaciones es esencial para el análisis de esta. Mas aun una serie de tiempo X_t es un proceso estocástico donde el índice está relacionado con el tiempo [18].*

En general en una serie de tiempo, no existen limitaciones para el lapso a medir: puede medirse en minutos, días, meses, años, etcétera. Lo único que se requiere es un punto de inicio y un punto final. Entre estos dos puntos se toman las mediciones; el tiempo transcurrido entre cada medición se considera constante y se conoce como periodo, en tanto su valor es llamado frecuencia.

Existen muchos ejemplos de análisis de series de tiempo, algunos de los cuales se mencionan en la Tabla 2.2 por campos científicos.

A menudo, los datos de series temporales se examinan para descubrir algún patrón. Para identificar tal patrón, es conveniente pensar que una serie de tiempo consta de varios componentes que a continuación se definen.

Definición 2.2.2 (Tendencia). *La tendencia se refiere al desplazamiento hacia arriba o hacia abajo que caracteriza a las series de tiempo con relación a un periodo de tiempo. Esta refleja el crecimiento o declinación de extensa duración en las series de tiempo. La tendencia puede ser lineal y no lineal [17].*

Definición 2.2.3 (Ciclo). *El ciclo se refiere a los movimientos hacia arriba o hacia abajo alrededor de los niveles de tendencia. Estas fluctuaciones tienen una duración que varía dependiendo del problema a estudiar [17].*

Definición 2.2.4 (Variaciones estacionales). *Las variaciones estacionales son patrones periódicos en una serie de tiempo que se completa dentro de un periodo anual que se repiten cada año [17].*

Definición 2.2.5 (Fluctuaciones irregulares). *Las fluctuaciones irregulares son movimientos erráticos que siguen un patrón indeterminado o irregular [17].*

| Campo científico | Ejemplos |
|------------------|--|
| Meteorología | Velocidad del viento Mediciones de lluvia Lecturas de temperatura Precipitaciones mensuales |
| Medicina | Monitoreo de la frecuencia cardíaca Actividad eléctrica cerebral |
| Economía | Precios de las acciones Tasas de interés Tasa de inflación Precio de combustibles |
| Demografía | Crecimiento de una población Casamientos mensuales Tasa de mortalidad |
| Marketing | Suscriptores mensuales de una red social Clientes nuevos Gastos mensuales |

Tabla 2.2: Ejemplos de análisis de series de tiempo por campos científicos (De autoría propia.).

2.3. Procesos estacionarios y autocorrelación

Anteriormente se mencionó, que una serie de tiempo es una realización que tiene el objetivo de determinar el comportamiento de ciertos fenómenos y con estos tratar de predecir o pronosticar el futuro, para lograr una predicción segura no es posible utilizar cualquier proceso estocástico, es decir, el proceso estocástico debe tener cierta estabilidad, pero si a cada momento de tiempo se observa un comportamiento diferente e inseguro, el proceso no es de mucha utilidad para poder lograr lo que uno desea. Este tipo de condiciones se denominan estacionariedad. La estacionariedad implica tener un tipo de estabilidad estadística en los datos que permite aprender sobre las propiedades del fenómeno.

Un proceso estocástico se puede caracterizar cuando se puede determinar el conjunto de funciones de distribución para cada conjunto finito de variables aleatorias del proceso, pero caracterizar el proceso a través de su función de distribución suele ser complicado, por lo que suele caracterizarse por los dos primeros momentos (media y varianza) de cada X_t y la covarianza de X_t y X_s .

Definición 2.3.1 (Media). *Se define la esperanza matemática de un proceso estocástico $\{X_t\}$ al tiempo t , como $\mu_x(t)$ tal que*

$$\mu_x(t) = \mathbb{E}[X_t] = \int_{-\infty}^{\infty} x f_t(x) dx, \quad \text{si } \int |x| f_t(x) dx < \infty. \quad (2.1)$$

En general, $\mu_x(t)$ puede ser diferente en cada instante de tiempo t [22].

Definición 2.3.2 (Varianza). *Se denomina varianza de un proceso estocástico $\{X_t\}$ al tiempo t , como $\sigma^2(t)$, tal que*

$$\sigma^2 = \text{Var}[X_t] = \mathbb{E}[(X_t - \mu_x(t))^2] = \int_{-\infty}^{\infty} (x - \mu_x(t))^2 f_t(x) dx, \quad (2.2)$$

$$\text{si } \int (x - \mu_x(t))^2 f_t(x) dx < \infty \quad [22].$$

Se entiende por covarianza al costo que indica el nivel de alteración conjunta de dos variables aleatorias en interacción a sus medias. Este dato ayuda a decidir si hay una dependencia entre las dos variables. Una covarianza positiva sugiere que una vez que una variable crece la otra variable también lo hace, o sea, poseen una interacción directa. Por otra parte, una covarianza negativa sugiere que una vez que una variable crece la otra decrece, manteniendo una relación inversa. En otras palabras, la covarianza es el valor que refleja en que cuantía dos variables aleatorias varían de forma conjunta respecto a sus medias.

Definición 2.3.3 (Covarianza). *La covarianza esta definida como:*

$$\begin{aligned} \text{Cov}(X_t, X_{t+k}) &= \text{Cov}(X_{t-t}, X_{t+k-t}) \\ &= \text{Cov}(X_0, X_k), \quad k > 0 \\ &= \text{Cov}(X_0, X_{|k|}) \\ &= \gamma_k \quad [23]. \end{aligned} \quad (2.3)$$

2.3.1. Series de tiempo estrictamente y débilmente estacionaria

Definición 2.3.4. *El proceso estocástico $\{X_t : t \geq 0\}$ es estacionario en el sentido estricto si para cualesquiera $t_1, \dots, t_n, h \geq 0$,*

$$(X_{t_1}, \dots, X_{t_n}) \stackrel{d}{=} (X_{t_1+h}, \dots, X_{t_n+h}), \quad (2.4)$$

es decir, el vector aleatorio del lado izquierdo se puede trasladar en el tiempo como indica el vector del lado derecho sin que cambie su distribución de probabilidad.

Determinar que el proceso estocástico es estacionario en sentido estricto suele ser difícil, ya que se requeriría determinar todas las distribuciones de la serie. Afortunadamente existe una condición más débil la cual podemos usar, se define a continuación.

Definición 2.3.5. *Sea $\{X_t\}$ un proceso estocástico y $\forall k, t \in \mathbb{R}$, se dice que este proceso es débilmente estacionario si:*

1. $\mathbb{E}(X_t) = \mu, \quad \mu < \infty.$
2. $\text{Var}(X_t) = \sigma^2, \quad \sigma^2 < \infty.$
3. $\text{Cov}(X_t, X_{t+k}) = \gamma_k.$

Ahora, si esperamos que X_t sea estacionaria, la media, la varianza, y la covarianza de X_t y X_{t+k} debe ser la misma que la covarianza de X_{t+h} y X_{t+k+h} , para cualquier h . En resumen, si una serie de tiempo es estacionaria, su media, su varianza y su covarianza (en los diferentes rezagos) permanecen iguales sin importar el momento en el cual se midan; es decir, son invariantes respecto del tiempo [21].

2.3.2. Autocovarianza, autocorrelación y autocorrelación parcial

En general, el análisis estadístico asume la independencia entre las variables aleatorias. Sin embargo, en las series temporales existen ciertas correlaciones entre las observaciones, es decir, un cambio que sufre una de las variables afecta a otra variable.

De la Definición 2.3.3 notemos que la autocovarianza sólo depende del lapso k .

Definición 2.3.6 (Función de autocovarianzas (FACV)). *La función de autocovarianzas de un proceso estocástico es una función de k (número de periodos de separación entre las variables) que recoge el conjunto de las autocovarianzas del proceso y se denota por γ_k , $k = 0, 1, 2, 3, \dots$.*

$$\gamma_k = \text{Cov}(X_t, X_{t+k}) = \mathbb{E}[(X_t - \mu)(X_{t+k} - \mu)]. \quad (2.5)$$

La función de autocovarianzas de un proceso estocástico recoge toda la información sobre la estructura dinámica lineal del mismo. Pero depende de las unidades de medida de la variable, por lo que, en general, se suele utilizar la función de autocorrelación [18].

Definición 2.3.7 (Función de Autocorrelación (FAC)). *La función de autocorrelación de un proceso estocástico mide el grado de asociación lineal existente entre dos variables aleatorias del proceso separadas k periodos:*

$$\rho_k = \frac{\mathbb{E}[(X_t - \mu)(X_{t+k} - \mu)]}{\sqrt{\mathbb{E}[(X_t - \mu)^2]\mathbb{E}[(X_{t+k} - \mu)^2]}} = \frac{\text{Cov}(X_t, X_{t+k})}{\text{Var}(X_t)} = \frac{\gamma_k}{\gamma_0}. \quad (2.6)$$

Puede observarse que el coeficiente de correlación no depende de unidades y $|\rho_k| \leq 1, \forall k$.

La función de autocorrelación de un proceso estocástico es una función de k que recoge el conjunto de los coeficientes de autocorrelación del proceso y se denota por $\rho_k, k = 0, 1, 2, 3, \dots$ [18].

Con lo anterior, la función de autocovarianza γ_k y función de autocorrelación ρ_k tienen las propiedades mostradas en la Tabla 2.3.

En la Figura 2.1 se muestran ejemplos de distintos comportamientos de la función de autocorrelación.

Además de la autocorrelación entre X_t y X_{t+k} , podemos querer investigar la correlación entre X_t y X_{t+k} después de que su dependencia lineal mutua de las variables que intervienen $X_{t+1}, X_{t+2}, \dots, X_{t+k-1}$ haya sido eliminada. La correlación antes mencionada la definimos a continuación.

Modelos de series de tiempo
2.3 Procesos estacionarios y autocorrelación

| | Función de autocovarianza (γ_k) | Función de autocorrelación (ρ_k) |
|---|---|---|
| 1 | $\gamma_0 \geq 0$ | $\rho_0 = 1$ |
| 2 | $ \gamma_k \leq \gamma_0$ | $ \rho_k \leq 1$ |
| 3 | $\gamma_k = \gamma_{-k}$ | $\rho_k = \rho_{-k}$ |
| 4 | $\sum_{i=1}^n \sum_{j=1}^n \alpha_i \alpha_j \gamma_{ i-j } \geq 0$ | $\sum_{i=1}^n \sum_{j=1}^n \alpha_i \alpha_j \rho_{ i-j } \geq 0$ |

Tabla 2.3: Propiedades de la función de autocovarianza y función de autocorrelación (Tabla extraída de [24]).

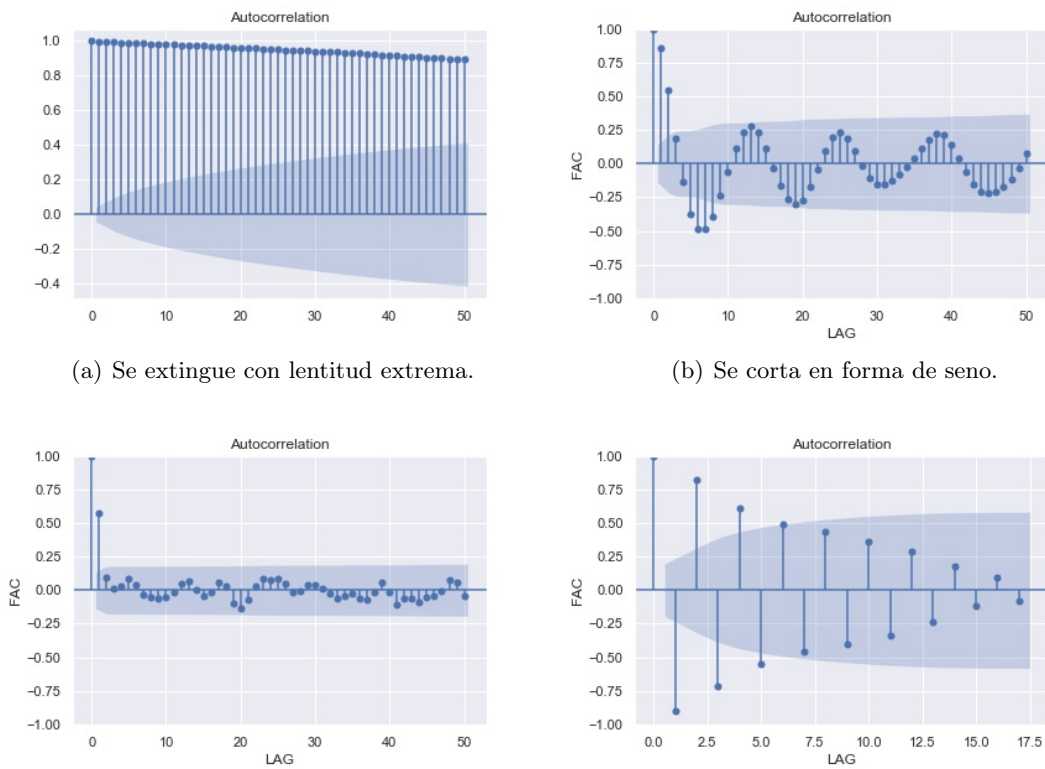
Definición 2.3.8 (Función de autocorrelación parcial (FACP)). *La autocorrelación parcial es denota por $\phi_{k,k}$ con $k = 1, 2, \dots$ es la correlación entre X_t y X_{t+k} después de que su dependencia lineal mutua en las que intervienen las variables $X_{t+1}, X_{t+2}, \dots, X_{t+k-1}$ haya sido eliminada. La correlación antes mencionada es la siguiente:*

$$\phi_{k,k} = \text{Corr}(X_t, X_{t+k} | X_{t+1}, X_{t+2}, \dots, X_{t+k-1}), \quad (2.7)$$

las variables X_t y X_{t+k} no están correlacionadas con $X_{t+1}, X_{t+2}, \dots, X_{t+k-1}$ es decir $\phi_{k,k}$ es la correlación de X_t y X_{t+k} en una dependencia lineal [22].

Nota 2.3.1. También la función de autocorrelación parcial (FACP) se puede representar en una división de determinantes la cual se muestra en la Ecuación (2.8).

$$\phi_{kk} = \frac{\begin{vmatrix} 1 & \rho_1 & \rho_2 & \cdots & \rho_{k-2} & \rho_1 \\ \rho_1 & 1 & \rho_1 & \cdots & \rho_{k-3} & \rho_2 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \\ \rho_{k-1} & \rho_{k-2} & \rho_{k-3} & \cdots & \rho_1 & \rho_k \end{vmatrix}}{\begin{vmatrix} 1 & \rho_1 & \rho_2 & \cdots & \rho_{k-2} & \rho_{k-1} \\ \rho_1 & 1 & \rho_1 & \cdots & \rho_{k-3} & \rho_{k-2} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \\ \rho_{k-1} & \rho_{k-2} & \rho_{k-3} & \cdots & \rho_1 & 1 \end{vmatrix}}. \quad (2.8)$$



(a) Se extingue con lentitud extrema.

(b) Se corta en forma de seno.

(c) Se corta o muere después del desfasamiento 1.

(d) Se corta en forma exponencial con oscilación.

Figura 2.1: Ejemplos del comportamiento de la *FAC* (De autoría propia).

2.3.3. Ejemplos de series de tiempo estacionarias

Ejemplo 2.3.1 (Proceso Ruido Blanco (RB)). Sea X_t un proceso estocástico, decimos que es un proceso estocástico puramente aleatorio o proceso estocástico de ruido blanco, si éste tiene las siguientes propiedades:

1. $\mathbb{E}(X_t) = 0, \quad \forall t.$
2. $Var(X_t) = \sigma^2, \quad \forall t.$
3. $Cov(X_t, X_s) = 0, \quad \forall t \neq s.$

En resumen, un proceso X_t es denominada ruido blanco si su valor promedio es cero, tiene varianza finita y constante, y no le importan sus valores pasados, de tal forma que su valor presente no se ve afectado por sus valores pasados. El proceso es débilmente estacionario y lo denotaremos como $X_t \sim RB(0, \sigma^2)$ [18].

Con función de autocovarianza:

$$\gamma_k = \begin{cases} \sigma^2, & \text{si } k = 0; \\ 0, & \text{si } k \neq 0. \end{cases}$$

Función de autocorrelación:

$$\rho_k = \begin{cases} 1, & \text{si } k = 0; \\ 0, & \text{si } k \neq 0. \end{cases}$$

Y función de autocorrelación parcial:

$$\phi_{k,k} = \begin{cases} 1, & \text{si } k = 0; \\ 0, & \text{si } k \neq 0. \end{cases}$$

En la Figura 2.2 se muestra la trayectoria que realiza un ruido blanco, su función de Autocorrelación y función de Autocorrelación parcial.

Ejemplo 2.3.2 (Modelo autorregresivo de segundo orden $AR(2)$). Sea X_t un proceso estocástico, supongamos que es estacionario y cumple la siguiente ecuación:

$$X_t = \phi_1 X_{t-1} + \phi_2 X_{t-2} + \epsilon_t \quad \text{para } t = \dots, -1, 0, 1, \dots, \quad (2.9)$$

en donde $\epsilon_t \sim RB(0, \sigma^2)$, $|\phi_t| < 1$ y ϵ_t no está correlacionada con X_s para cada $s < t$.

El modelo autorregresivo de segundo orden $AR(2)$ tiene función de autocovarianza:

$$\gamma_k = \phi_1 \gamma_{k-1} + \phi_2 \gamma_{k-2}, \quad k \geq 1.$$

Función de autocorrelación:

$$\rho_k = \phi_1 \rho_{k-1} + \phi_2 \rho_{k-2}, \quad k \geq 1. \quad (2.10)$$

Para la función de autocorrelación parcial:

Por la Ecuación (2.10) tenemos lo siguiente:

$$\phi_{11} = \rho_1 = \frac{\phi_1}{1 - \phi_2}, \quad \text{con } 1 - \phi_2 \neq 0$$

$$\begin{aligned} \phi_{22} &= \frac{\begin{vmatrix} 1 & \rho_1 \\ \rho_1 & \rho_2 \end{vmatrix}}{\begin{vmatrix} 1 & \rho_1 \\ \rho_1 & 1 \end{vmatrix}} \\ &= \frac{\rho_2 - \rho_1^2}{1 - \rho_1^2} \quad \text{con } 1 - \rho_1^2 \neq 0 \\ &= \frac{\phi_2[(1 - \phi_2)^2 - \phi_1^2]}{(1 - \phi_2)^2 - \phi_1^2} = \phi_2, \end{aligned}$$

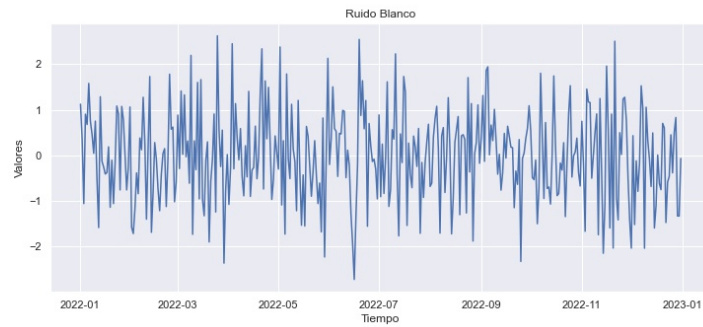
$$\begin{aligned} \phi_{33} &= \frac{\begin{vmatrix} 1 & \rho_1 & \rho_1 \\ \rho_1 & 1 & \rho_2 \\ \rho_2 & \rho_1 & \rho_3 \end{vmatrix}}{\begin{vmatrix} 1 & \rho_1 & \rho_2 \\ \rho_1 & 1 & \rho_1 \\ \rho_2 & \rho_1 & 1 \end{vmatrix}} \\ &= \frac{\begin{vmatrix} 1 & \rho_1 & \phi_1 + \phi_2 \rho_1 \\ \rho_1 & 1 & \phi_1 \rho_1 + \phi_2 \\ \rho_2 & \rho_1 & \phi_1 \rho_2 + \phi_2 \rho_1 \end{vmatrix}}{\begin{vmatrix} 1 & \rho_1 & \rho_2 \\ \rho_1 & 1 & \rho_1 \\ \rho_2 & \rho_1 & 1 \end{vmatrix}} \\ &= 0. \end{aligned}$$

Observemos que la última columna del numerador es una combinación lineal de las dos primeras columnas. Del mismo modo, se puede demostrar que para $\phi_{kk} = 0$ para $k \geq 3$. Por lo que la *FACP* de un proceso $AR(2)$ se corta después del retraso 2.

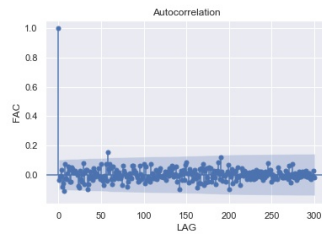
En la Figura 2.3 se muestra la trayectoria que realiza un modelo $AR(2)$, su función de Autocorrelación y función de Autocorrelación parcial.

Modelos de series de tiempo

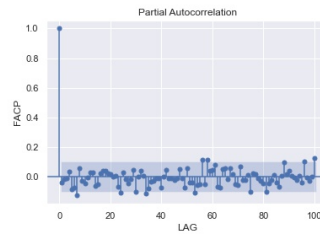
2.3 Procesos estacionarios y autocorrelación



(a) Realización del ruido blanco.

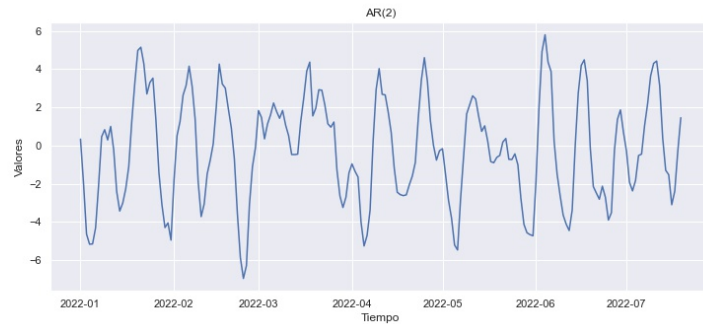


(b) Autocorrelación.

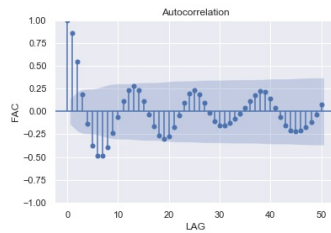


(c) Autocorrelación parcial.

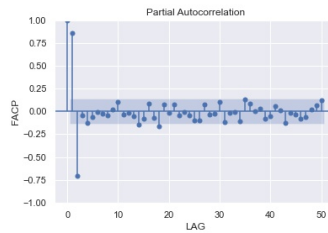
Figura 2.2: Ejemplo del proceso de ruido blanco (De autoría propia).



(a) Trayectoria $AR(2)$.



(b) Autocorrelación.



(c) Autocorrelación parcial.

Figura 2.3: Ejemplo del proceso $AR(2)$ con $\phi_1 = 1.5$ y $\phi_2 = -0.75$ (De autoría propia).

2.4. Estimación de la media, autovarianzas y autocorrelaciones

Una serie temporal estacionaria se caracteriza por su media μ , varianza σ^2 , autocorrelaciones ρ_k y autocorrelaciones parciales $\phi_{k,k}$. Los valores exactos de estos parámetros se pueden calcular si se conoce el conjunto de todas las realizaciones posibles. De lo contrario, se pueden estimar si se dispone de múltiples realizaciones independientes. Sin embargo, es difícil o imposible obtener múltiples realizaciones. Dado que los valores de una realización del proceso representan al mismo proceso, podemos usar la muestra del proceso para hacer inferencia de los parámetros poblacionales.

Definición 2.4.1 (Media muestral). Sean x_1, x_2, \dots, x_n observaciones de una serie de tiempo, la media muestral está definida como:

$$\bar{x} = \frac{1}{n} \sum_{t=1}^n x_t. \quad (2.11)$$

Definición 2.4.2 (Varianza muestral). Sean x_1, x_2, \dots, x_n observaciones de una serie de tiempo, la varianza muestral está definida como:

$$\hat{\sigma}_x^2 = \frac{1}{n} \sum_{t=1}^n (x_t - \bar{x})^2. \quad (2.12)$$

Definición 2.4.3 (Función de autocovarianza muestral). Sean x_1, x_2, \dots, x_n observaciones de una serie de tiempo, la función de autocovarianza muestral está definida como:

$$\hat{\gamma}_k = \frac{1}{n} \sum_{t=1}^{n-k} (x_t - \bar{x})(x_{t+k} - \bar{x}), \quad \text{donde } -n < k < n. \quad (2.13)$$

Definición 2.4.4 (Función de autocorrelación muestral). Sean x_1, x_2, \dots, x_n observaciones de una serie de tiempo, la función de autocorrelación muestral está definida como:

$$\hat{\rho}_k = \frac{\hat{\gamma}_k}{\hat{\gamma}_0} = \frac{\sum_{t=1}^{n-k} (x_t - \bar{x})(x_{t+k} - \bar{x})}{\sum_{t=1}^n (x_t - \bar{x})^2}, \quad k = 0, 1, 2, \dots \quad \text{y} \quad -n < k < n. \quad (2.14)$$

Definición 2.4.5 (Función de autocorrelación parcial muestral (FACPM)). La función de autocorrelación parcial muestral es obtenida por un método recursivo, iniciando con $\hat{\phi}_{1,1} = \hat{\rho}_1$ y así calcular $\hat{\phi}_{k,k}$.

$$\hat{\phi}_{k+1,k+1} = \frac{\hat{\rho}_{k+1} - \sum_{j=1}^k \hat{\phi}_{kj} \hat{\rho}_{k+1-j}}{1 - \sum_{j=1}^k \hat{\phi}_{kj} \hat{\rho}_j} \quad (2.15)$$

y

$$\hat{\phi}_{k+1,j} = \hat{\phi}_{k,j} - \hat{\phi}_{k+1,k+1} \hat{\phi}_{k,k+1-j}, \quad j = 1, \dots, k. \quad (2.16)$$

Este método también funciona para el calcular la FACP [22].

2.5. Modelo Autorregresivo de Medias Móvil (ARMA)

2.5.1. Introducción

La estructura de dependencia temporal de un proceso estocástico está recogida en la función de auto covarianzas y/o en la función de autocorrelación. En este contexto, se trata de utilizar la información de estas funciones para extraer un patrón sistemático y, a partir de este, un modelo que reproduzca el comportamiento de la serie y se pueda utilizar para predecir.

En un modelo de series temporales univariante se descompone la serie X_t en dos partes, una que recoge el patrón de regularidad, o parte sistemática, y otra parte puramente aleatoria denominada también innovación.

$$X_t = PS_t + \epsilon_t \quad t = 1, 2, \dots \quad (2.17)$$

donde:

- PS es la parte sistemática.
- ϵ_t es la innovación.

La parte sistemática es la parte predecible con el conjunto de información que se utiliza para construir el modelo, es decir, la serie temporal X_1, X_2, \dots, X_t . La innovación respecto al conjunto de información con el que se construye el modelo es una parte aleatoria en la que sus valores no tienen ninguna relación o dependencia entre sí. La innovación en el momento t no está relacionada, por lo tanto, ni con las innovaciones anteriores ni con las posteriores, ni con la parte sistemática del modelo.

Dentro de los procesos estocásticos estacionarios se considerará únicamente la clase de procesos lineales que se caracterizan porque se pueden representar como una combinación lineal de variables aleatorias. En el caso de los procesos estacionarios con distribución normal y media cero, la teoría de procesos estocásticos señala que, bajo condiciones muy generales, X_t se puede expresar como combinación lineal de los valores pasados infinitos de X más una innovación ruido blanco [18] :

$$X_t = \pi_1 X_{t-1} + \pi_2 X_{t-2} + \pi_3 X_{t-3} + \dots + \epsilon_t, \quad \forall t, \quad \text{donde, } t = 1, 2, \dots, \quad (2.18)$$

y los coeficientes π_i son números reales.

Para que un modelo sea admitido por un proceso estocástico estacionario, debe cumplir las siguientes dos condiciones:

1. El proceso sea no anticipante, es decir, que el presente no venga determinado por el futuro, luego el valor de X en el momento t no puede depender de valores futuros de X o de las innovaciones.

2. El proceso sea invertible, es decir, que la correlación entre una variable y su pasado debe ir disminuyendo conforme nos alejemos del pasado. Esta condición se cumple si los parámetros del modelo cumplen la siguiente restricción:

$$\sum_{i=1}^{\infty} \pi_i^2 < \infty. \quad (2.19)$$

Los modelos que estudiaremos pueden tener tres representaciones, todas igualmente validas bajo los supuestos señalados:

1. Medias Móviles de orden q , $MA(q)$.
2. Autorregresivos de orden p , $AR(p)$.
3. Medias Móviles de Autorregresivos, $ARMA(p, q)$.

Definición 2.5.1 (Proceso Lineal). *Sea X_t una serie de tiempo, se dice que es un proceso lineal si éste se puede representar como:*

$$X_t = \sum_{j=-\infty}^{\infty} \psi_j \epsilon_{t-j}, \quad \forall t \in T, \quad (2.20)$$

donde $\epsilon_t \sim RB(0, \sigma^2)$ y ψ_j es una sucesión de constantes tales que $\sum_{j=-\infty}^{\infty} |\psi_j| < \infty$.

La condición $\sum_{j=-\infty}^{\infty} |\psi_j| < \infty$ garantiza que la suma infinita de la Ecuación (2.20) converge.

Lo anterior se simplificará en términos de B que es el operador de retardo, esto último surge de algunos métodos diferentes, se define el operador Δ de diferencias de lapso $k = 1$, por:

$$\Delta X_t = X_t - X_{t-1} = (1 - B)X_t. \quad (2.21)$$

Así:

$$BX_t = X_{t-1}.$$

Se define a la potencia del operador B como $B^j(X_t)$ y $\Delta^j(X_t) = \Delta(\Delta^{j-1}(X_t))$, $j \geq 1$ con $\Delta^0(X_t) = X_t$ polinomios en B y en Δ son manipulados de la misma forma como una función polinómica de variables reales, por ejemplo:

$$\begin{aligned} \Delta^2 X_t &= \Delta(\Delta(X_t)) \\ &= (1 - B)(1 - B)X_t \\ &= (1 - 2B + B^2)X_t \\ &= X_t - 2X_{t-1} + X_{t-2}. \end{aligned} \quad (2.22)$$

Tomando en cuenta que $B\epsilon_t = \epsilon_{t-1}$, el proceso lineal se puede expresar como:

$$X_t = \psi(B)\epsilon_t,$$

donde $\psi(B) = \sum_{j=-\infty}^{\infty} \psi_j B^j$ [23].

De lo anterior se desprenden algunos casos particulares, que reciben nombres específicos, por ejemplo:

Decimos que un proceso lineal es un proceso de media móvil o $MA(\infty)$ si $\psi_j = 0$, $\forall j < 0$, es decir, si:

$$X_t = \sum_{j=0}^{\infty} \psi_j \epsilon_{t-j} \quad \forall t \in T,$$

donde $\psi_0 = 1$, ϵ_t y $\sum_{j=0}^{\infty} \psi_j^2 < \infty$.

2.5.2. Procesos autorregresivos AR(p)

El modelo autorregresivo finito de orden p , $AR(p)$ es una aproximación natural al modelo lineal general (2.18). Se obtiene al representar el valor actual del proceso X_t como una combinación lineal de los p últimos valores pasados de sí mismos más una innovación contemporánea (ruido blanco). El modelo es representado de la siguiente forma:

$$X_t = \phi_1 X_{t-1} + \phi_2 X_{t-2} + \dots + \phi_p X_{t-p} + \epsilon_t \quad t = 1, 2, \dots \quad (2.23)$$

donde $\epsilon_t \sim RB(0, \sigma^2)$, $\pi_1 = \phi_1, \pi_2 = \phi_2, \dots, \pi_p = \phi_p, \pi_k = 0$ y $k > p$ [18], [22].

Nota 2.5.1. Observemos, ya que $\sum_{i=1}^{\infty} |\pi_i| = \sum_{i=1}^{\infty} |\phi_i| < \infty$ los procesos autorregresivos de cualquier orden son invertibles [22].

Las características del proceso estacionario $AR(p)$ estacionario son:

El modelo autorregresivo $AR(p)$ tiene función de autocovarianzas:

$$\gamma_k = \phi_1 \gamma_{k-1} + \phi_2 \gamma_{k-2} + \dots + \phi_p \gamma_{k-p}, \quad (2.24)$$

con $k > 0$ [22].

Función de autocorrelación:

$$\rho_k = \frac{\gamma_k}{\gamma_0} = \phi_1 \rho_{k-1} + \phi_2 \rho_{k-2} + \dots + \phi_p \rho_{k-p}, \quad k \geq 1. \quad (2.25)$$

La función de autocorrelación, ρ_k , de un proceso $AR(p)$ decrece exponencialmente hacia cero sin truncarse. Ahora bien, para valores de $p > 1$ al ser el modelo más complejo, la estructura de la *FAC* es también más rica y puede presentar más variedad de formas [18].

Y para la función de autocorrelación parcial ϕ_{kk} , dado que $\rho_k = \phi_1\rho_{k-1} + \phi_2\rho_{k-2} + \dots + \phi_p\rho_{k-p}$, $k \geq 1$, podemos ver que cuando $k > p$ la última columna del determinante en el numerador de 2.8 puede ser escrita como una combinación lineal de las columnas previas del determinante, de esta manera el determinante es igual a cero y la *FACP* se corta después del lapso p [22].

De ejemplo veamos las características del modelo autorregresivo de orden 1, $AR(1)$.

2.5.3. Proceso $AR(1)$

El proceso $AR(1)$ la variable X_t viene determinada únicamente por su valor pasado, X_{t-1} , y la perturbación contemporánea, ϵ_t :

$$X_t = \phi_1 X_{t-1} + \epsilon_t \quad t = 1, 2, \dots \quad \epsilon_t \sim RB(0, \sigma^2). \quad (2.26)$$

Por lo mencionado en la Nota (2.5.1), el proceso es invertible.

Para verificar si el modelo $AR(1)$ es estacionario para cualquier valor del parámetro ϕ debemos comprobar las condiciones de estacionariedad.

a) Estacionario en media:

$$\mathbb{E}(X_t) = \mathbb{E}(\phi X_{t-1} + \epsilon_t) = \phi \mathbb{E}(X_{t-1}).$$

Para que el proceso sea estacionario, la media ha de ser constante y finita lo que implica:

$$\mathbb{E}(X_t) = \phi \mathbb{E}(X_t) \Rightarrow (1 - \phi)\mathbb{E}(X_t) = 0 \Rightarrow \mathbb{E}(X_t) = \frac{0}{1 - \phi} = 0.$$

Por lo tanto, para que el proceso sea estacionario es necesario que el parámetro $\phi \neq 1$.

b) Estacionario en covarianza:

Para que el proceso sea estacionario, la varianza ha de ser constante y finita:

$$\begin{aligned} \gamma_0 &= \mathbb{E}(X_t - \mathbb{E}(X_t))^2 \\ &= \mathbb{E}(\phi X_{t-1} + \epsilon_t - 0)^2 \\ &= \phi^2 \mathbb{E}(X_{t-1})^2 + \mathbb{E}(\epsilon_t)^2 + 2\phi \mathbb{E}(X_{t-1}\epsilon_t) \\ &= \phi^2 V(X_{t-1}) + \sigma^2 + 0. \end{aligned}$$

Dada la estructura de autocorrelación del proceso

$$\mathbb{E}(X_{t-1}\epsilon_t) = \mathbb{E}[(X_{t-1} - 0)(\epsilon_t - 0)] = cov(X_{t-1}\epsilon_t) = 0.$$

Bajo el supuesto de que el proceso es estacionario,

$$\mathbb{E}(X_{t-1})^2 = V(X_{t-1}) = V(X_t) = \gamma_0$$

por lo que:

$$\gamma_0 = \phi\gamma_0 + \sigma^2 \Rightarrow (1 - \phi^2)\gamma_0 = \sigma^2 \Rightarrow \gamma_0 = \frac{\sigma^2}{1 - \phi^2}.$$

Para que el proceso sea estacionario, la varianza debe ser constante y finita, por lo que es necesario que $|\phi| < 1$.

En lo que se refiere a la función de autocovarianzas, la autocovarianza de orden k es:

$$\begin{aligned} \gamma_k &= \mathbb{E}(X_t - \mathbb{E}(X_t))^2 \\ &= \mathbb{E}(\phi X_{t-1} + \epsilon_t - 0)^2 \\ &= \phi^2 \mathbb{E}(X_{t-1})^2 + \mathbb{E}(\epsilon_t)^2 + 2\phi \mathbb{E}(X_{t-1}\epsilon_t) \\ &= \phi^2 V(X_{t-1}) + \sigma^2 + 0. \end{aligned}$$

Por lo que:

$$\begin{aligned} \gamma_1 &= \phi\gamma_0 \\ \gamma_2 &= \phi\gamma_1 \\ \gamma_3 &= \phi\gamma_2 \\ &\vdots = \vdots \end{aligned}$$

Por lo tanto, si el parámetro autorregresivo es $|\phi| < 1$, la varianza es finita y el resto de las autocovarianzas también lo son, y además dependen únicamente de los periodos de separación entre las variables y no del tiempo, por lo que el proceso es estacionario.

Se puede concluir, por lo tanto, que el proceso $AR(1)$ es estacionario si y sólo si $|\phi| < 1$.

El modelo autorregresivo de primer orden $AR(1)$ tiene función de autocovarianza:

$$\gamma_k = \begin{cases} \frac{\sigma^2}{(1+\phi_1^2)}, & \text{si } k = 0; \\ \phi_1^{|k|}\gamma_0, & \text{si } k > 0. \end{cases}$$

Función de autocorrelación:

$$\rho_k = \begin{cases} 1, & \text{si } k = 0; \\ \phi_1^{|k|}, & \text{si } k \neq 0. \end{cases}$$

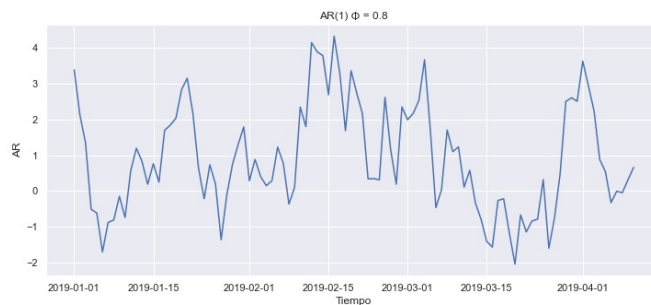
Y función de autocorrelación parcial:

$$\phi_{kk} = \begin{cases} \rho_1 = \phi_1, & \text{si } k = 1; \\ 0, & \text{si } k > 1. \end{cases}$$

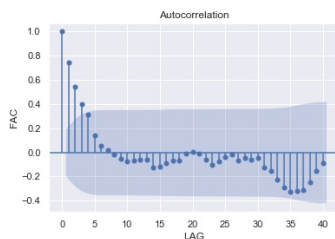
En las Figuras 2.4 y 2.5 se muestran dos trayectorias de un modelo $AR(1)$ diferentes, sus funciones de Autocorrelación y funciones de Autocorrelación parcial.

Modelos de series de tiempo

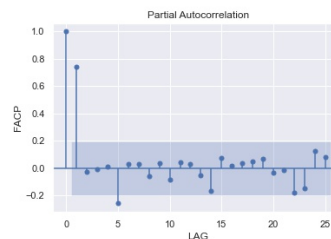
2.5 Modelo Autorregresivo de Medias Móvil (ARMA)



(a) Trayectoria $AR(1)$.

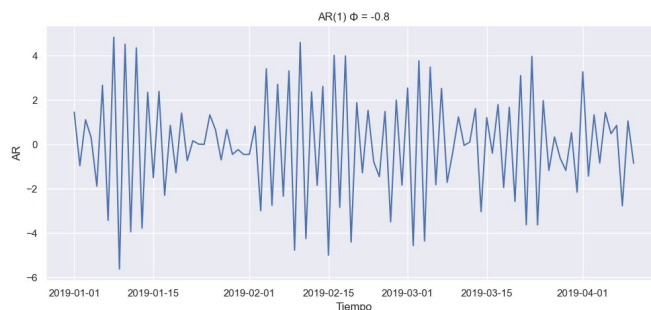


(b) Autocorrelación.

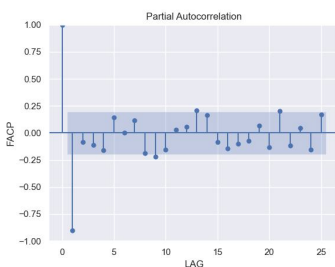


(c) Autocorrelación parcial.

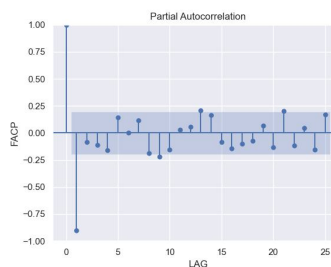
Figura 2.4: Ejemplo del proceso $AR(1)$ con $\phi_1 = 0.8$ (De autoría propia).



(a) Trayectoria $AR(1)$.



(b) Autocorrelación.



(c) Autocorrelación parcial.

Figura 2.5: Ejemplo del proceso $AR(1)$ con $\phi_1 = -0.8$ (De autoría propia).

2.5.4. Procesos de media móvil

Los modelos de media móvil de orden finito q , también llamados $MA(q)$, representan un proceso estocástico X_t como una suma finita ponderada de ruidos blancos, que pertenecen a periodos pasados, estas ponderaciones están representadas por θ_j , $j = 1, 2, \dots, q$, es decir, $\psi_1 = -\theta_1$, $\psi_2 = -\theta_2$, \dots , $\psi_q = -\theta_q$ y $\psi_k = 0$ para $k > q$. Entonces se dice que el proceso resultante es un proceso de media móvil de orden q , se denota como $MA(q)$ y es representado como:

$$X_t = \epsilon_t - \theta_1\epsilon_{t-1} - \theta_2\epsilon_{t-2} - \dots - \theta_q\epsilon_{t-q}, \quad (2.27)$$

donde $\epsilon_t \sim RB(0, \sigma^2)$ [22].

Nota 2.5.2. Como $1 + \theta_1^2 + \dots + \theta_q^2 < \infty$, entonces un proceso de media móvil es siempre estacionario [22].

Las características del proceso $MA(q)$ estacionario son:

El modelo de media móvil de orden finito q , $MA(q)$ tiene función de autocovarianza:

$$\gamma_k = \begin{cases} \sigma^2(1 + \theta_1^2 + \dots + \theta_q^2) & \text{si } k = 0; \\ \sigma^2(-\theta_k + \theta_1\theta_{k+1} + \dots + \theta_{q-k}\theta_q) & \text{si } k = 1, 2, \dots, q; \\ 0 & \text{si } |k| > q. \end{cases} \quad (2.28)$$

Función de autocorrelación:

$$\rho_k = \begin{cases} \frac{-\theta_k + \theta_1\theta_{k+1} + \dots + \theta_{q-k}\theta_q}{1 + \theta_1^2 + \dots + \theta_q^2}, & \text{si } k = 1, 2, \dots, q; \\ 0, & \text{si } k > q. \end{cases} \quad (2.29)$$

La función de autocorrelación de un proceso $MA(q)$ se corta después del retraso q . Esta importante propiedad nos permite identificar si una serie de tiempo es generada por un proceso de media móvil.

Y para la función de autocorrelación parcial, resulta ser una mezcla de decaimientos exponenciales y ondas sinusoidales amortiguadas dependiendo de las raíces de $(1 - \theta_1 B - \dots - \theta_q B^q) = 0$ el cual es el polinomio de retardo [22].

De ejemplo veamos las características del modelo de media móvil de orden 1, $MA(1)$.

2.5.5. Proceso MA(1)

El modelo $MA(1)$ determina el valor de X en el momento t en función de la innovación contemporánea y su primer retardo:

$$X_t = \epsilon_t - \theta_1\epsilon_{t-1} \quad (2.30)$$

donde $\epsilon_t \sim RB(0, \sigma^2)$, $t = 1, 2, \dots$

La función de autocovarianzas de un $MA(1)$ es:

$$\gamma_k = \begin{cases} \gamma_0 = (1 + \theta^2)\sigma^2, & k = 0, \\ \gamma_1 = -\theta\sigma^2, & k = 1, \\ \gamma_k = 0, & k > 1. \end{cases} \quad (2.31)$$

La función de autocovarianzas es finita y depende sólo de k y no del tiempo, para cualquier valor del parámetro θ .

La función de autocorrelación de un proceso $MA(1)$ es:

$$\rho_k = \begin{cases} 1, & k = 0, \\ \frac{-\theta}{1+\theta^2}, & k = 1, \\ 0, & k > 1. \end{cases} \quad (2.32)$$

La función de autocorrelación de un proceso $MA(1)$ se caracteriza por ser una función truncada que sufre un corte brusco tomando el valor cero a partir del retardo 1.

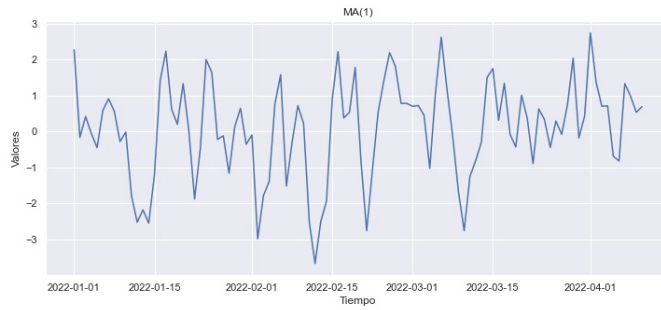
Y la función de autocorrelación parcial es la siguiente:

$$\phi_{kk} = \frac{-\theta_1^k(1 - \theta_1^2)}{1 - \theta_1^{2(k+1)}}, \quad \text{para } k \geq 1. \quad (2.33)$$

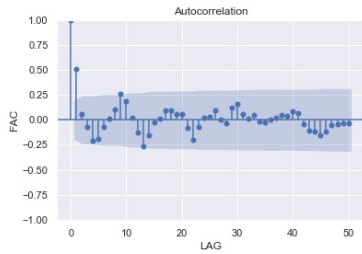
Al contrario de su función de autocorrelación, que se interrumpe después del retardo 1, la *FACP* de un modelo $MA(1)$ se reduce exponencialmente en una de dos formas dependiendo del signo de θ_1 (por lo tanto, del signo de ρ_1).

En las Figuras 2.6 y 2.7 se muestra una trayectoria que realiza un modelo $MA(1)$, su función de Autocorrelación y función de Autocorrelación parcial.

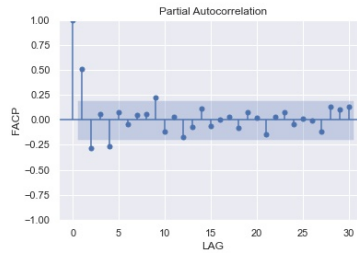
Modelos de series de tiempo
2.5 Modelo Autorregresivo de Medias Móvil (ARMA)



(a) Trayectoria $MA(1)$.

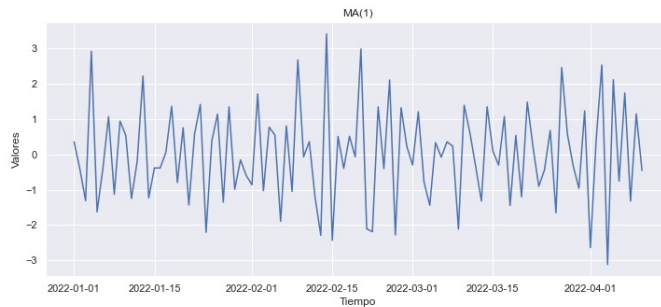


(b) Autocorrelación.

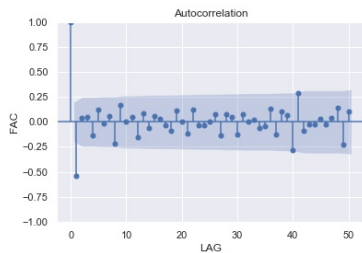


(c) Autocorrelación parcial.

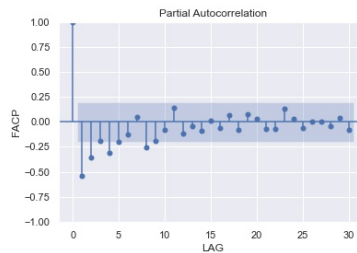
Figura 2.6: Ejemplo del proceso $MA(1)$ con $\theta_1 = 0.8$ (De autoría propia).



(a) Trayectoria $MA(1)$.



(b) Autocorrelación.



(c) Autocorrelación parcial.

Figura 2.7: Ejemplo del proceso $MA(1)$ con $\theta_1 = -0.8$ (De autoría propia).

2.5.6. Proceso Autorregresivo de Medias Móviles (ARMA)

La combinación de los procesos de $AR(q)$ y $MA(p)$ da como resultado a los procesos mixtos a los cuales se les denomina como $ARMA(p, q)$ y los cuales se definen como sigue:

Definición 2.5.2. Sea X_t un proceso estocástico, se dice que X_t es un proceso $ARMA(p, q)$ si es estacionario y si para cada t , se cumple:

$$X_t = \phi_1 X_{t-1} + \cdots + \phi_p X_{t-p} + \epsilon_t - \theta_1 \epsilon_{t-1} - \cdots - \theta_q \epsilon_{t-q}, \quad (2.34)$$

donde $\epsilon_t \sim RB(0, \sigma^2)$. Expresado de forma equivalente:

$$(1 - \phi_1 B - \cdots - \phi_p B^p) X_t = (1 - \theta_1 B - \cdots - \theta_q B^q) \epsilon_t \quad (2.35)$$

o bien,

$$\Phi_p(B) X_t = \Theta_q(B) \epsilon_t$$

donde $\Phi_p(B)$ es el polinomio autorregresivo y $\Theta_q(B)$ es el polinomio de medias móviles.

El modelo $ARMA(p, q)$ es estacionario si cumple que:

- La parte $MA(p)$ es estacionaria.
- La parte $AR(q)$ es estacionaria.

Para hallar la función de autocovarianza (γ_k) de un proceso $ARMA(p, q)$ multiplicamos X_{t-k} por ambos lados a la Ecuación (2.34) y obteniendo la esperanza, se obtiene:

$$\mathbb{E}[X_{t-k} X_t] = \phi_1 \mathbb{E}[X_{t-k} X_{t-1}] + \cdots + \phi_p \mathbb{E}[X_{t-k} X_{t-p}] + \mathbb{E}[X_{t-k} \epsilon_t] - \theta_1 \mathbb{E}[X_{t-k} \epsilon_{t-1}] - \cdots - \theta_q \mathbb{E}[X_{t-k} \epsilon_{t-q}],$$

como $\mathbb{E}[X_{t-k} \epsilon_{t-i}] = 0$, $k > i$, tenemos:

$$\gamma_k = \phi_1 \gamma_{k-1} + \cdots + \phi_p \gamma_{k-p}, \quad k \geq (q-1). \quad (2.36)$$

Así, la función de autocorrelación es:

$$\rho_k = \phi_1 \rho_{k-1} + \cdots + \phi_p \rho_{k-p}, \quad k \geq (q+1). \quad (2.37)$$

La función de autocorrelación del proceso $ARMA$ es análoga a la del proceso $AR(p)$. Por lo tanto, la función de autocorrelación de un modelo $ARMA(p, q)$ se corta después del lapso q así como ocurre en un proceso $AR(p)$.

Y para la función de autocorrelación parcial ya que el proceso $ARMA$ contiene al proceso MA como un caso especial, su FACP también será una mezcla de decaimientos exponenciales u ondas sinusoidales amortiguadas dependiendo de las raíces de $\phi_p(B) = 0$ y $\theta_q(B) = 0$ [22].

2.5.7. Proceso ARMA(1,1)

El modelo *ARMA* más sencillo es el modelo *ARMA*(1, 1), este modelo se determina en función de su pasado hasta el primer retardo, la innovación contemporánea y el pasado de la innovación hasta el retardo 1:

$$X_t = \phi_1 X_{t-1} + \epsilon_t - \theta_1 \epsilon_{t-1}, \quad \epsilon_t \sim RB(0, \sigma^2) \quad t = 1, 2, \dots \quad (2.38)$$

El proceso *ARMA*(1, 1), es estacionario cuando $|\phi| < 1$, e invertible cuando $|\theta| < 1$. Cuando $\phi_1 = 0$ el modelo se reduce al proceso *MA*(1) y cuando $\theta_1 = 0$ el modelo se reduce al proceso *AR*(1) [22].

Algunas características del proceso estacionario *ARMA*(1, 1) estacionario son:

El modelo autorregresivo *ARMA*(1, 1) tiene función de autocovarianzas:

$$\gamma_k = \begin{cases} \frac{(1+\theta^2-2\phi\theta)\sigma^2}{1-\phi^2}, & k = 0, \\ \phi\gamma_0 - \theta\sigma^2, & k = 1, \\ \phi\gamma_{k-1}, & k > 1. \end{cases} \quad (2.39)$$

La autocovarianza de orden 1, γ_1 , es la suma de la autocovarianza de orden 1 de la parte *AR*(1) y de la autocovarianza de orden 1 de la parte *MA*(1). Esta función depende de los parámetros *AR*, ϕ , y *MA*, θ , hasta $k = 1$ y luego decrece exponencialmente, siguiendo la estructura marcada por la parte autorregresiva de primer orden [18].

La función de autocorrelación para un modelo *ARMA*(1, 1) es el siguiente:

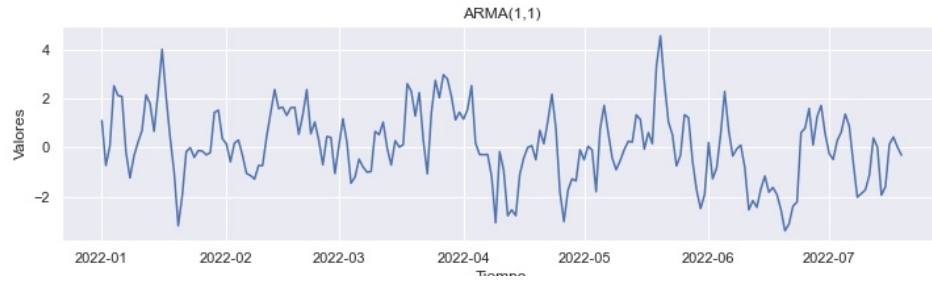
$$\rho_k = \begin{cases} 1, & k = 0, \\ \frac{(\phi_1 - \theta_1)(1 - \phi_1 \theta_1)}{1 + \theta_1^2 - 2\phi_1 \theta_1}, & k = 1, \\ \phi_1 \rho_{k-1}, & k > 1. \end{cases} \quad (2.40)$$

La *FAC* presenta la misma estructura que la función de autocovarianzas, es decir, es una función cuyo coeficiente, ρ_1 , depende de los parámetros autorregresivos y de medias móviles, pero a partir del retardo 2, decrece exponencialmente, siguiendo la estructura dada por la parte autorregresiva de orden 1 [18].

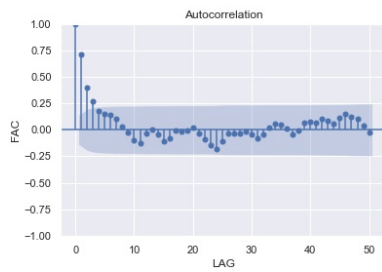
Para la función de autocorrelación parcial de un modelo mixto, la forma general es complicada y no es necesaria. Es suficiente notar que, debido a que el proceso *ARMA*(1, 1) contiene al proceso *MA*(1), la *FACP* del proceso *ARMA*(1, 1) también disminuye exponencialmente como la *FAC*, con su forma dependiendo en los signos y magnitudes de ϕ_1 y θ_1 [22].

En las Figuras 2.8 y 2.9 se muestra una trayectoria que realiza un modelo *ARMA*(1, 1), su función de autocorrelación y función de autocorrelación parcial.

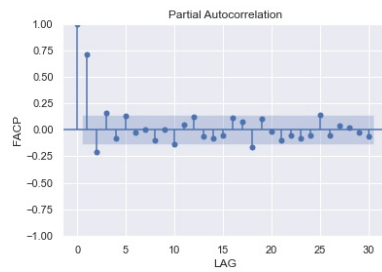
Modelos de series de tiempo
2.5 Modelo Autorregresivo de Medias Móvil (ARMA)



(a) Trayectoria ARMA(1,1).

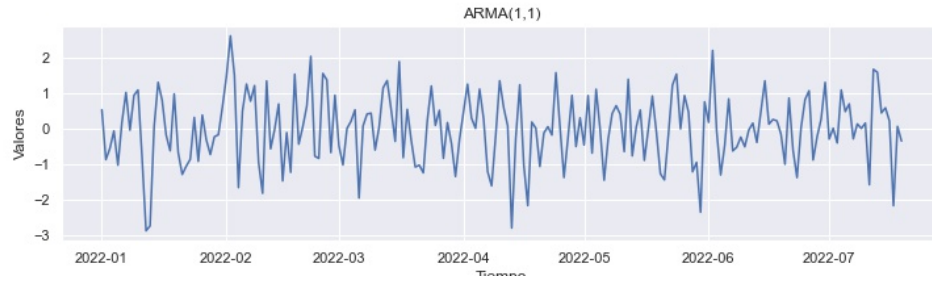


(b) Autocorrelación.

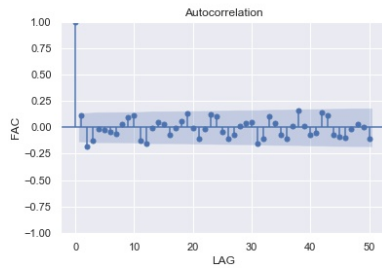


(c) Autocorrelación parcial.

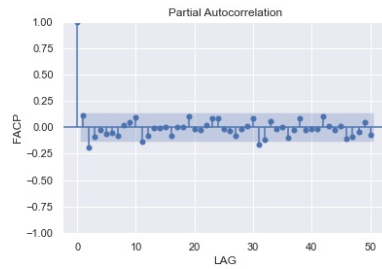
Figura 2.8: Ejemplo del proceso ARMA(1,1) con $\theta_1 = 0.5$ y $\phi_1 = 0.4$ (De autoría propia).



(a) Trayectoria ARMA(1,1).



(b) Autocorrelación.



(c) Autocorrelación parcial.

Figura 2.9: Ejemplo del proceso ARMA(1,1) con $\theta_1 = 0.5$ y $\phi_1 = -0.4$ (De autoría propia).

Modelos de series de tiempo
2.5 Modelo Autorregresivo de Medias Móvil (ARMA)

Finalmente, en la Tabla 2.5.7 se da un resumen de la *FAC* y *FACP* para los procesos generales vistos anteriormente.

| Proceso | <i>FAC</i> | <i>FACP</i> |
|--------------|--|--|
| $AR(p)$ | Resulta una mezcla de decaimientos exponenciales u ondas sinusoidales amortiguadas, dependiendo de los signos. | Se corta después del lapso p . |
| $MA(q)$ | Se corta después del lapso q . | Resulta una mezcla de decaimientos exponenciales u ondas sinusoidales amortiguadas, dependiendo de los signos. |
| $ARMA(p, q)$ | Disminuye después del retraso $(q - p)$. | Disminuye después del retraso $(p - q)$. |

Tabla 2.4: Resumen de la función de autocorrelación y la función de autocorrelación parcial para los diferentes procesos (Tabla extraída de [22]).

2.6. Modelo Autorregresivo de Medias Móviles Integrado ARIMA

Los modelos de series de tiempo analizados se basan en el supuesto de que las series de tiempo consideradas son estacionarias. En pocas palabras, la media y la varianza de una serie de tiempo estacionaria son constantes y su covarianza es invariante en el tiempo. Pero sabemos que muchas series de tiempo son no estacionarias, por ejemplo, véase las Figuras 2.10 y 2.11.

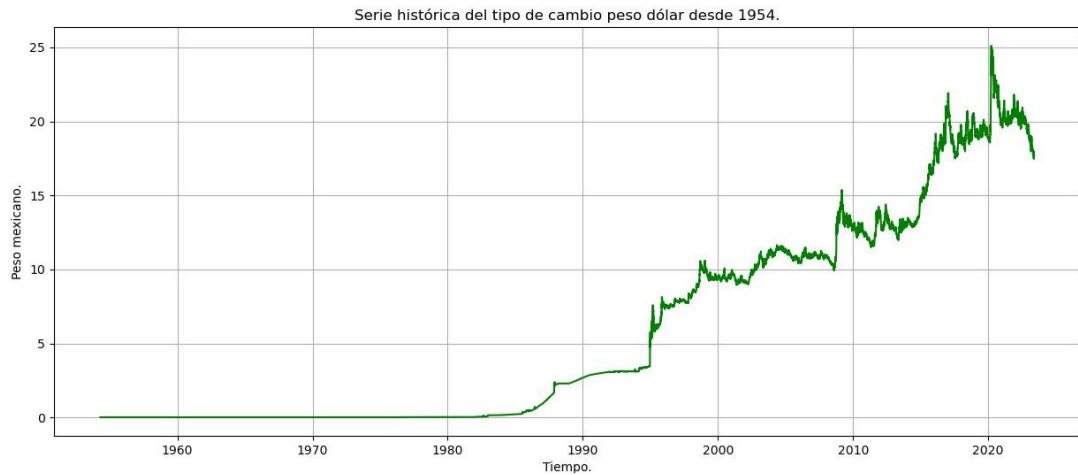


Figura 2.10: Serie histórica del tipo de cambio peso dólar desde 1954, fecha de consulta: 27/05/2023 (Datos extraídos del Banco de México).

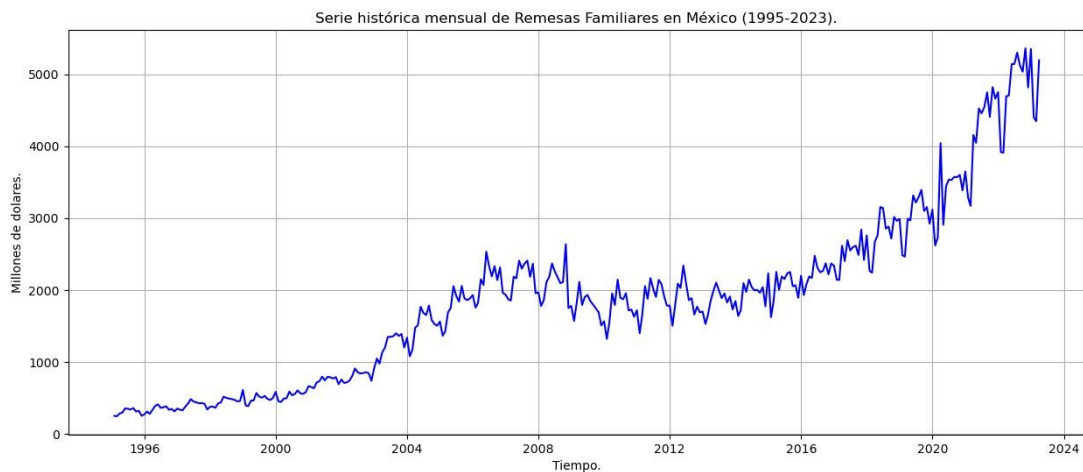


Figura 2.11: Serie histórica de Remesas Totales (Millones de dólares), fecha de consulta: 27/05/2023 (Datos extraídos del Banco de México).

2.6.1. Modelo autorregresivo integrado de medias móviles de orden p , d , q ARIMA(p , d , q)

Al igual que el modelo *ARMA* el modelo *ARIMA* es la combinación de los modelos de autorregresivos y el de medias móviles con la diferencia de incluir un proceso de restablecimiento de inestabilidad original presente en nuestra serie de tiempo al cual se le denomina integración d . La forma general de un modelo *ARIMA* se presenta a continuación:

$$X'_t = c + \phi X'_{t-1} + \dots + \phi_p X'_{t-p} + \epsilon_t + \theta_1 \epsilon_{t-1} + \dots + \theta_q \epsilon_{t-q}. \quad (2.41)$$

En donde X' , expresa que sobre la serie original X_t se han aplicado d diferencias y $\epsilon_t \sim RB(0, \sigma^2)$ [23].

Los parámetros (p, d, q) representan lo siguiente:

- p , es el orden del polinomio autorregresivo.
- d , es el exponente en el operador de diferencia.
- q , es el orden del polinomio de medias móviles.

Cuando $p = 0$ el modelo *ARIMA*(p, d, q) es también llamado modelo integrado de media móvil de orden (d, q) y es denotado *IMA*(d, q). Si $p = 0$, $d = 1$ y $q = 0$ entonces se tiene lo que se conoce como modelo de caminata aleatoria. Y si $q = 0$ éste es llamado autorregresivo integrado denotado por *ARI*(p, d) [22].

2.6.2. Modelo ARIMA(0,1,0) o modelo de caminata aleatoria

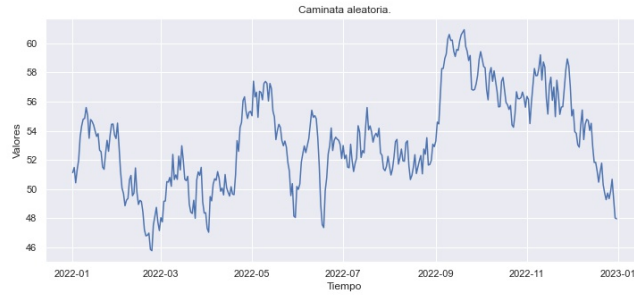
Como se mencionó anteriormente el modelo *ARIMA*(0, 1, 0) también se conoce como modelo de caminata aleatoria, el cual es el siguiente:

$$X_t = X_{t-1} + \epsilon_t \quad \text{con} \quad \epsilon_t \sim RB(0, \sigma^2). \quad (2.42)$$

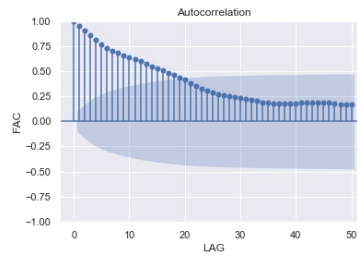
Este modelo ha sido ampliamente utilizado para describir el comportamiento de la serie del precio de una acción. En el modelo de caminata aleatoria, el valor de X en el tiempo t es igual a su valor en el tiempo $(t - 1)$ más un choque aleatorio. Este comportamiento es similar a seguir a un hombre borracho cuya posición en el tiempo t es su posición en el tiempo $(t - 1)$ más un paso en una dirección aleatoria en el tiempo t [22].

En lo que se refiere a la *FAC* del Modelo (2.42), se puede comprobar que se caracteriza porque sus coeficientes decrecen muy lentamente en las Figuras 2.12 y 2.13 se muestran dos trayectorias diferentes de un modelo de caminata aleatoria y sus *FAC*.

Modelos de series de tiempo
2.6 Modelo Autorregresivo de Medias Móviles Integrado ARIMA

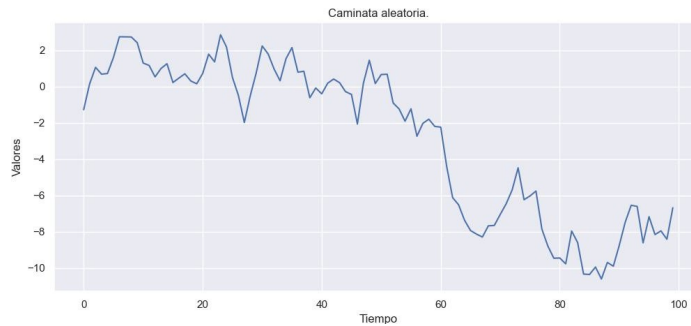


(a) Trayectoria caminata aleatoria.

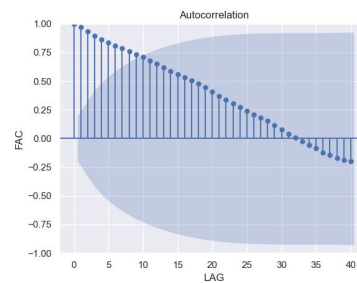


(b) Autocorrelación.

Figura 2.12: Trayectoria y *FAC* de un proceso $ARIMA(0,1,0)$ (De autoría propia).



(a) Trayectoria caminata aleatoria.



(b) Autocorrelación.

Figura 2.13: Trayectoria y *FAC* de un modelo de caminata aleatoria (De autoría propia).

2.6.3. Varianza y Autovarianza de los modelos ARIMA

Un proceso que es estacionario en la media no necesariamente es estacionario en la varianza y la autocovarianza. Sin embargo, un proceso que es no estacionario en la media también será no estacionario en la varianza y la autocovarianza.

Un fenómeno muy fundamental sobre el modelo *ARIMA* es que el modelo no es estacionario pero las características completas del proceso están determinadas para todo el tiempo solo por un número finito de parámetros.

Por ejemplo consideremos las variables X_1, X_2, \dots, X_n y supongamos que los ajustamos a un modelo *IMA(1, 1)*, esto es, se tiene el siguiente modelo:

$$X_t = X_{t-1} + \epsilon_t - \theta\epsilon_{t-1}, \quad (2.43)$$

consideremos la serie anterior en una serie de n_0 observaciones. Con referencia al origen de tiempo anterior para $t > n_0$ podemos escribir por sustituciones sucesivas como sigue:

$$\begin{aligned} X_t &= X_{t-1} + \epsilon_t - \theta\epsilon_{t-1} \\ &= X_{t-2} + \epsilon_t + (1 - \theta)\epsilon_{t-1} - \theta\epsilon_{t-2} \\ &\vdots \\ &= X_{n_0} + \epsilon_t + (1 - \theta)\epsilon_{t-1} + \dots + (1 - \theta)\epsilon_{n_0+1} - \theta\epsilon_{n_0}. \end{aligned}$$

Similarmente a lo anterior, pero ahora para $t - k > n_0$ tenemos:

$$X_{t-k} = X_{n_0} + \epsilon_{t-k} + (1 - \theta)\epsilon_{t-k-1} + \dots + (1 - \theta)\epsilon_{n_0+1} - \theta\epsilon_{n_0}. \quad (2.44)$$

Por lo tanto con respecto al origen de tiempo n_0 tenemos:

$$Var(X_t) = [1 + (t - n_0 - 1)(1 - \theta)^2]\sigma_a^2 \quad (2.45)$$

$$Var(X_{t-k}) = [1 + (t - k - n_0 - 1)(1 - \theta)^2]\sigma_a^2 \quad (2.46)$$

$$Cov(X_{t-k}, X_t) = [(1 - \theta) + (t - k - n_0 - 1)(1 - \theta)^2]\sigma_a^2 \quad (2.47)$$

observamos que X_{n_0} y ϵ_{n_0} son valores conocidos con respecto al origen de tiempo n_0 , y

$$Corr(X_{t-k}, X_t) = \frac{Cov(X_{t-k}, X_t)}{\sqrt{Var(X_{t-k})Var(X_t)}} = \frac{(1 - \theta) + (t - k - n_0 - 1)(1 - \theta)^2}{\sqrt{[1 + (t - k - n_0 - 1)(1 - \theta)^2][1 + (t - n_0 - 1)(1 - \theta)^2]}}. \quad (2.48)$$

Ahora, de (2.45) y (2.48), tenemos las siguientes observaciones importantes:

1. La varianza, $Var(X_t)$, de un modelo *ARIMA* depende del tiempo.
2. $Var(X_t) \neq Var(X_{t-k})$ para $k \neq 0$.

3. La varianza es no acotada cuando $t \rightarrow \infty$.
4. La covarianza y la autocorrelación también dependen del tiempo y por tanto, no son invariantes respecto a la traslación del tiempo.

En general, con sólo una realización es difícil o imposible hacer inferencias de un proceso no estacionario tanto de la media como de la autocovarianza o autocorrelación [22].

2.6.4. Análisis de estacionariedad

Estacionariedad en varianza

Una serie será estacionaria en varianza cuando pueda mantenerse el supuesto de que existe una única varianza para toda la serie temporal, es decir, cuando la variabilidad de la serie en torno a su media se mantenga constante a lo largo del tiempo. Si la serie no es estacionaria en varianza, se utilizan las transformaciones estabilizadoras de varianza, es decir, las transformaciones Box-Cox,

$$X_t^{(\lambda)} = \begin{cases} \frac{X_t^\lambda - 1}{\lambda}, & \lambda \neq 0, \\ \ln X_t, & \lambda = 0. \end{cases} \quad (2.49)$$

Las transformaciones Box-Cox incluye una familia infinita de funciones: raíz cuadrada, inversa, etc. Los instrumentos que se utilizan para analizar la estacionariedad en varianza de una serie son el gráfico de la serie original y de las transformaciones correspondientes [18]. En la Tabla 2.5, se puede ver los valores de uso común de λ .

En el Figura 2.14 se presenta una serie de tiempo del gasto público mensual (millones de dólares) como parte del Plan de Beneficios Farmacéuticos según lo registrado por la Comisión Australiana de Seguros de Salud, capturados en Julio 1991 hasta Junio 2008.

La primera gráfica de la Figura 2.14 se puede observar la serie original y la segunda gráfica es la serie transformada, tomando logaritmos a la serie (la transformación Box-Cox más utilizada para las series económicas).

Estacionariedad en media

Si una serie es estacionaria en media oscila en torno a un nivel constante. A continuación, se enuncian características que diferencian las series estacionarias de las no estacionarias.

Características de una serie estacionaria:

- Una serie es estacionaria en media cuando se puede mantener el supuesto de que existe una única media para toda la serie temporal, es decir, cuando fluctúa alrededor de una media constante.
- La función de autocorrelación teórica de un proceso estacionario en media decae rápidamente, exponencialmente.

| Valores de λ . | Transformación $(X_t^{(\lambda)})$. |
|------------------------|--------------------------------------|
| -1.0 | $\frac{1}{X_t}$ |
| -0.5 | $\frac{1}{\sqrt{X_t}}$ |
| 0.0 | $\ln X_t$ |
| 0.25 | $4(\sqrt[4]{X_t} - 1)$ |
| 0.5 | $\sqrt{X_t}$ |
| 1.0 | X_t (No hay transformación) |

Tabla 2.5: Transformaciones generales (Tabla extraída de [22]).

Características de una serie no estacionaria:

- La serie no es estacionaria en media cuando presenta tendencia o varios tramos con medias diferentes.
- En general, un proceso con alguna raíz unitaria presenta una función de autocorrelación muestral con un decaimiento muy lento.

Si la serie no es estacionaria en media se puede lograr la estacionariedad transformándola, tomando diferencias. Así, si la serie no es estacionaria en media, se tomarán d sucesivas diferencias de orden 1 sobre la serie hasta obtener una serie estacionaria.

Las primeras diferencias de los valores de la serie temporal x_1, x_2, \dots, x_n son:

$$z_t = x_t - x_{t-1} \quad \text{donde } t = 2, \dots, n.$$

Sucesivamente se pueden obtener las siguientes diferencias, por ejemplo, las segundas diferencias de los valores de la serie temporal x_1, x_2, \dots, x_n son:

$$\begin{aligned} z_t &= (x_t - x_{t-1}) - (x_{t-1} - x_{t-2}) \\ &= x_t - 2x_{t-1} + x_{t-2} \quad \text{para } t = 3, 4, \dots, n \quad [17]. \end{aligned}$$

En la Figura 2.15 se muestra una serie de tiempo del uso mensual de tarjetas de débito en Islandia (millones de ISK). Los datos fueron recolectados en enero 2000 hasta agosto 2013. En la primera gráfica se muestra la serie de tiempo original y la segunda gráfica se muestra la serie transformada por diferencias.

Modelos de series de tiempo

2.6 Modelo Autorregresivo de Medias Móviles Integrado ARIMA

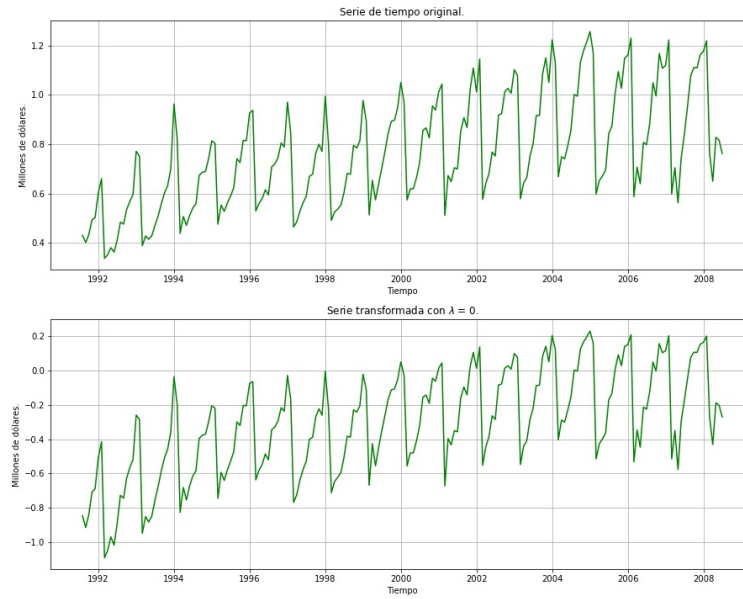


Figura 2.14: Ejemplo de la transformación Box-Cox (De autoría propia).

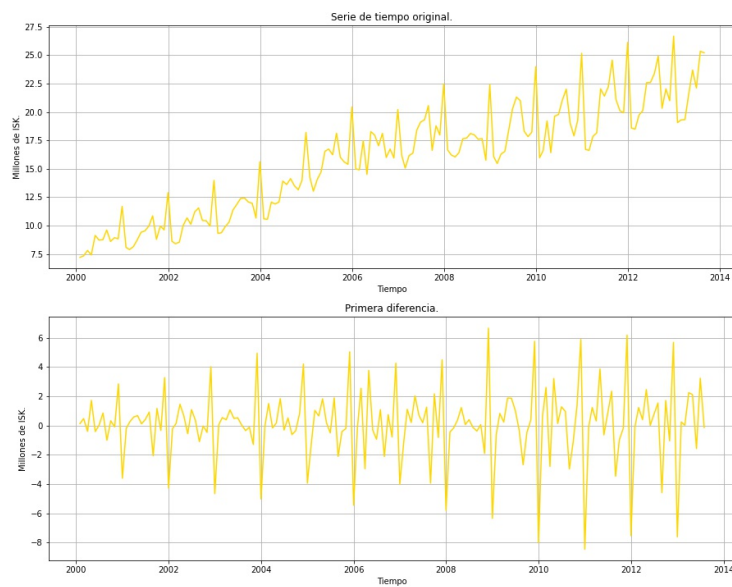


Figura 2.15: Ejemplo de transformación por diferencias (De autoría propia).

2.7. Modelos multiplicativos

Muchas series de tiempo económicas y de negocios contienen un fenómeno estacional que se repite después de un período de tiempo regular. El período de tiempo más pequeño para este fenómeno repetitivo se denomina período estacional. Por ejemplo, la serie trimestral de ventas de helados es alta cada verano y la serie repite este fenómeno cada año, dando un período estacional de 4. De manera similar, las ventas mensuales de automóviles y las ganancias tienden a disminuir durante agosto y septiembre de cada año debido al cambio a nuevos modelos, y las ventas mensuales de juguetes aumentan cada año en el mes de diciembre. El período estacional en estos últimos casos es de 12. Los fenómenos estacionales pueden deberse a factores como el clima, que afecta muchas actividades comerciales y económicas como el turismo y la construcción de viviendas; eventos culturales como la Navidad, que está íntimamente relacionado con ventas como joyas, juguetes, tarjetas de felicitación y estampillas; y ceremonias de graduación en los meses de verano, que están directamente relacionadas con la situación de la fuerza laboral en estos meses [22].

Por ejemplo, en la Figura 2.16 se muestra la gráfica de un conjunto de datos de pasajeros aéreos, el conjunto de datos se obtuvo de Kaggle. Este conjunto de datos brinda información sobre el número total de pasajeros mensuales de una aerolínea estadounidense desde 1949 hasta 1960. La estacionalidad de la serie es evidente ya que los picos están en el mes de agosto y los puntos más bajos están en noviembre de cada año. El fenómeno se repite cada 12 meses, por lo que el período estacional es 12.

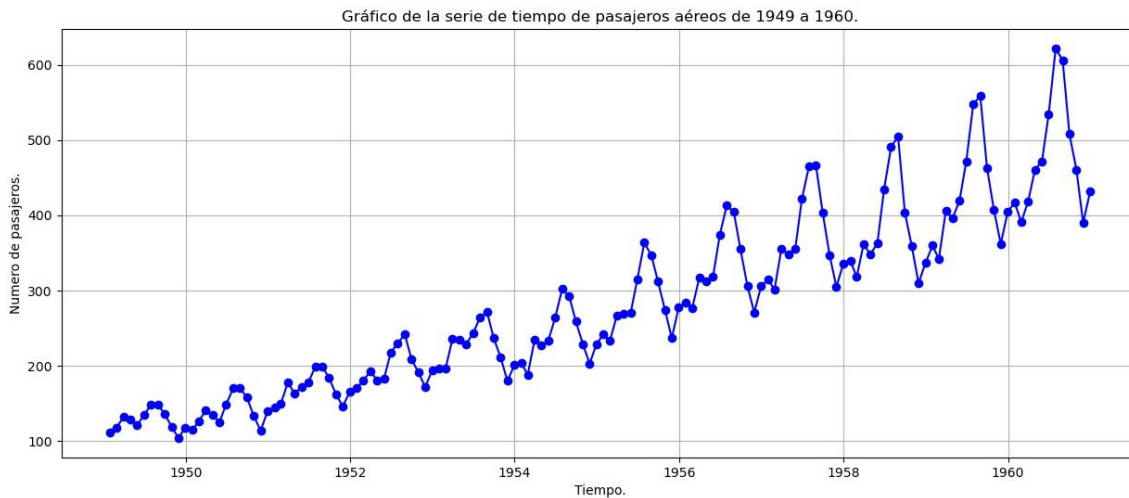


Figura 2.16: Datos de pasajeros aéreos (Datos extraídos de Kaggle).

Cuando una serie temporal estacionaria presenta estacionalidad, resulta útil describir la evolución del proceso estocástico con un modelo $ARMA$ multiplicativo denotado por $ARMA(p, q) \times ARMA(P, Q)_S$ del tipo:

$$\phi_p(B)\Phi_P(B^S)X_t = \delta + \theta_q(B)\Theta_Q(B^S)a_t,$$

donde S es la cantidad de observaciones anuales. En este modelo $\phi_p(B)$ es el operador autorregresivo estacionario de orden p y $\theta_q(B)$ es el operador del proceso de media móvil de orden q , en este contexto suelen denominarse **operadores regulares**.

Los polinomios $\Phi_P(B^S) = 1 - \Phi_1(B^S) - \Phi_2(B^{2S}) - \dots - \Phi_P(B^{PS})$ y $\Theta_Q(B^S) = 1 - \Theta_1(B^S) - \Theta_2(B^{2S}) - \dots - \Theta_Q(B^{QS})$ con raíces fuera del círculo unitario, suelen denominarse **operadores estacionales** y a_t variables independientes e idénticamente distribuidas con media cero y varianza constante. En este modelo, los operadores regulares modelan la correlación entre pares de observaciones de X_t separadas entre sí por periodos básicos $k = 1, 2, \dots$, mientras que los operadores estacionales describen la correlación entre pares de componentes del proceso separados por desfases estacionales $k = S, 2S, \dots$

Cuando la serie presenta variaciones cíclicas o estacionales, es decir, hay una tendencia periódica, es idóneo agregar la componente de la tendencia secular d y una componente para la parte cíclica D donde $d, D > 0$ esto es, un modelo multiplicativo $ARIMA(p, d, q) \times ARIMA(P, D, Q)_S$ lo cual se denota por $SARIMA(p, d, q)(P, D, Q)_S$ y se define como sigue:

Definición 2.7.1. *Un modelo autorregresivo estacional integrado de media móvil multiplicativo o modelo SARIMA es representado con la siguiente ecuación:*

$$\phi_p(B)\Phi_P(B^S)\Delta^d\Delta_S^D X_t = \delta + \theta_q(B)\Theta_Q(B^S)\epsilon_t,$$

donde:

- $\phi_p(B)$, es el operador del proceso autorregresivo de orden p .
- $\theta_q(B)$, es el operador del proceso de media móvil de orden q .
- $\Phi_P(B^S)$, es el operador estacional autorregresivo de orden P .
- $\Theta_Q(B^S)$, es el operador estacional de media móvil de orden Q .
- Δ^d , es el componente de diferencia de orden d .
- Δ_S^D , es el componente de diferencia estacional de orden D y de periodo s .
- Las diferencias ordinarias y estacionales son respectivamente $\Delta^d = (1 - B)^d$ y $\Delta_S^D = (1 - B^S)^D$
- ϵ_t , es un proceso de Ruido Blanco [22], [24].

2.7.1. Modelo SARIMA(0,1,1)(0,1,1)₁₂

Consideremos el modelo $SARIMA(0, 1, 1)(0, 1, 1)_{12}$ con la siguiente ecuación:

$$\Delta^1\Delta_{12}^1 X_t = \theta_1(B)\Theta_1(B^{12})\epsilon_t,$$

o bien:

$$(1 - B)(1 - B^{12})X_t = (1 - \theta B)(1 - \Theta B^{12})\epsilon_t. \quad (2.50)$$

Desarrollando la Ecuación (2.50) tenemos:

$$(1 - B^{12})(X_t - X_{t-1}) = (1 - \Theta B^{12})(\epsilon_t - \theta\epsilon_{t-1})$$

$$X_t - X_{t-12} - X_{t-1} + X_{t-13} = \epsilon_t - \Theta\epsilon_{t-12} - \theta\epsilon_{t-1} + \Theta\theta\epsilon_{t-13}.$$

Despejando X_t tenemos:

$$X_t = X_{t-1} + X_{t-12} - X_{t-13} - \theta\epsilon_{t-1} - \Theta\epsilon_{t-12} + \Theta\theta\epsilon_{t-13} + \epsilon_t.$$

Se ha encontrado que este modelo es muy útil para representar una variedad de series temporales estacionales, como datos de líneas aéreas y series comerciales. El modelo fue introducido por primera vez por Box y Jenkins para representar los datos de viajes aéreos internacionales. Por lo tanto, también se le conoce como el modelo de línea aérea en la literatura [22].

2.8. Metodología Box-Jenkins

Ahora nos preguntamos al tener una serie de tiempo, ¿cómo sabemos si sigue un proceso *AR* puro (de ser así, cuál es el valor de p), un proceso *MA* puro (de ser así, cuál es el valor de q), un proceso *ARMA* (de ser así, cuáles son los valores de p y q), un proceso *ARIMA* (de ser así se deben conocer los valores de p , d y q) o un proceso *SARIMA*, en cuyo caso se deben conocer los valores de p , d , q , P , D y Q ? La metodología Box-Jenkins resulta útil para responder la pregunta anterior, nombrada así en honor a los estadísticos George E. P. Box y Gwilym Jenkins [21].

El método considera cinco pasos consecutivos:

1. Estacionariedad.
2. Identificación.
3. Estimación.
4. Diagnóstico.
5. Pronóstico.

2.8.1. Estacionariedad

Previamente se explica que los procesos estocásticos se clasifican en estacionarios y no estacionarios. La estacionariedad está principalmente relacionada con la estabilidad de una serie. De la definición mencionada anteriormente se puede decir que una serie de tiempo es estacionaria si su media y su varianza son constantes a través del tiempo. Este concepto es de gran importancia para la metodología Box-Jenkins ya que en caso de que los datos no se comporten de manera estable a lo largo del tiempo, es necesario aplicar

una transformación a la serie para llevarla a ello. Existen algunas formas para conocer si una serie es estacionaria: por medio del gráfico de la serie, por medio de la función de autocorrelación simple y por la prueba de Dickey-Fuller.

1. Inspección gráfica.

Primero graficamos la serie de tiempo, si se observa que los valores de la serie fluctúan respecto a una media constante, entonces es razonable pensar que la serie es estacionaria, si no ocurre lo anterior se llega a la conclusión de que la serie no es estacionaria. De manera visual se puede detallar si la serie es no estacionaria si es que se detectan elevaciones o inclinaciones en las observaciones.

2. Funciones de autocorrelación.

Como se vio anteriormente la *FAC* puede mostrar una variedad de comportamientos diferentes. Dependiendo del comportamiento del correlograma correspondiente a la función de autocorrelación, éste nos puede ayudar a saber si nuestra serie es estacionaria:

- Si en el correlograma sus valores se cortan claramente con rapidez o se desvanecen (decrecen) muy rápidamente, entonces se debe considerar que los valores de la serie temporal son estacionarios.
- Si por el contrario, en el correlograma sus valores se desvanecen con lentitud extrema, entonces se debe considerar que los valores de la serie temporal son no estacionarios [17].

3. Prueba Dickey-Fuller.

2.8.2. Identificación

Identificación. Es decir, encontrar los valores apropiados de los parámetros p , d y q .

En la práctica, la Función de autocorrelación y la Función de autocorrelación parcial son desconocidas, por lo que deben ser estimadas por la Función de autocorrelación muestral y la Función de autocorrelación parcial muestral, de manera que éstas coincidan con los patrones de la Función de autocorrelación y la Función de autocorrelación parcial para los modelos *ARMA*.

Si se quiere identificar un modelo *ARIMA*, se requiere un número mínimo de 50 observaciones y el número de autocorrelaciones muestrales y parciales en un lapso k deben calcularse aproximadamente para $\frac{n}{4}$. Aunque a veces para datos de alta calidad, podemos usar tamaños de muestra más pequeños para identificar modelos adecuados. Para determinar el orden de p y q , los patrones de la Función de autocorrelación muestral y Función de autocorrelación parcial muestral deben coincidir con patrones teóricos del modelo [22].

2.8.3. Estimación

Tras identificar los valores apropiados de p y q , la siguiente etapa es estimar los parámetros de los términos autorregresivos y de medias móviles incluidos en el modelo. Algunas veces, este cálculo se efectúa mediante mínimos cuadrados simples, pero otras hay que recurrir a métodos de estimación no lineal (en parámetros). Como esta labor se lleva a cabo ahora a través de rutinas en diversos paquetes estadísticos, en la práctica no es preciso preocuparse por los desarrollos matemáticos de la estimación [21].

2.8.4. Diagnóstico

Después de seleccionar un modelo particular y de estimar sus parámetros, tratamos de ver si el modelo seleccionado se ajusta a los datos en forma razonablemente buena, pues es posible que exista otro modelo que también lo haga. Es por lo que el diseño de modelos de Box-Jenkins es un arte más que una ciencia; se requiere gran habilidad para seleccionar el modelo correcto. Una simple prueba del modelo seleccionado es ver si los residuales estimados a partir de este modelo son de ruido blanco; si lo son, aceptamos el ajuste particular; si no lo son, debemos empezar de nuevo. Por tanto, la metodología Box-Jenkins es un proceso iterativo [21].

El análisis de residuos consiste en una serie de contrastes de diagnóstico con el objetivo de determinar si los residuos replican el comportamiento de un ruido blanco, es decir, si su media es cero, su varianza constante y las autocorrelaciones nulas. Algunas maneras que se puede hacer este análisis se mencionan a continuación:

- Para comprobar si la media es cero, se realiza un análisis gráfico, representando los residuos a lo largo del tiempo y observando si los valores oscilan alrededor de cero.
- Varianza constante. Si en el gráfico de los residuos la dispersión de estos es constante, concluiremos que la varianza de los residuos permanece constante.
- Si los residuos se comportaran como un ruido blanco, los coeficientes de la *FAC* y *FACP* muestrales deben ser prácticamente nulos para todos los retardos.
- Otra alternativa es realizar pruebas de hipótesis por ejemplo la Prueba de Ljung-Box [18].

2.8.5. Pronóstico

Una razón de la popularidad del proceso de construcción de modelos con la Metodología de Box-Jenkins es su éxito en el pronóstico. En muchos casos, los pronósticos obtenidos por este método son más confiables que los obtenidos por otros modelos, en particular en el caso de pronósticos de corto plazo [21].

La Figura 2.17 muestra un diagrama de flujo el cual representa los pasos a seguir para la metodología Box-Jenkins.

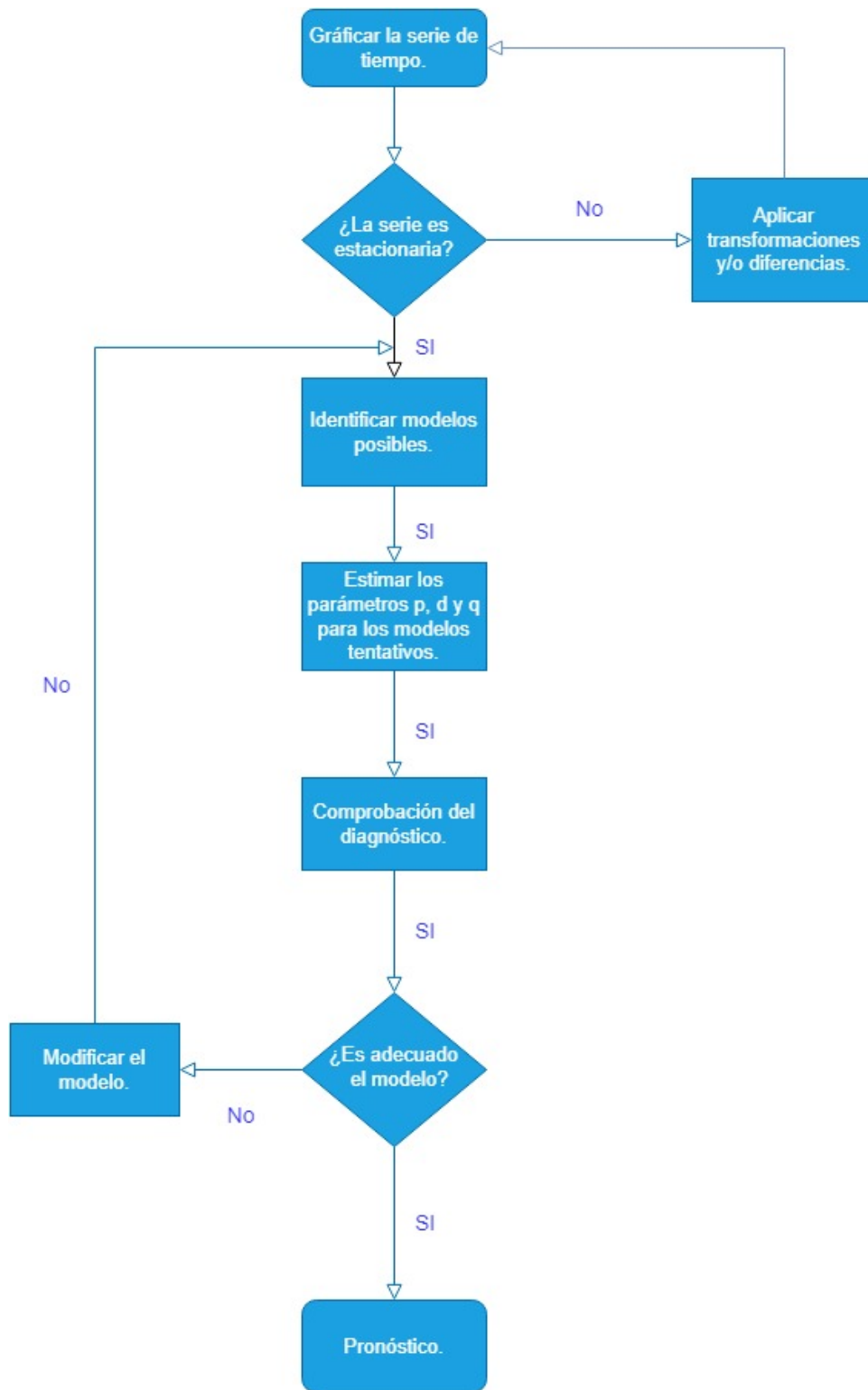


Figura 2.17: Diagrama de Flujo de la metodología Box-Jenkins (De autoría propia).

Capítulo 3

Redes Neuronales Artificiales

3.1. Machine learning

El machine learning es una rama de la inteligencia artificial que permite que las máquinas aprendan sin la necesidad de ser programadas para ello. Una habilidad indispensable para hacer sistemas capaces de identificar patrones entre los datos para hacer predicciones. Esta tecnología está presente en un sinnúmero de aplicaciones como las recomendaciones de Netflix o Spotify, las respuestas inteligentes de Gmail o el habla de Google Home y Alexa [26].

Los sistemas de computación utilizan algoritmos de machine learning para procesar grandes cantidades de datos históricos e identificar patrones de datos. Esto les permite generar resultados con mayor precisión a partir de un conjunto de datos de entrada.

El origen del machine learning se remonta al siglo pasado. “La estadística es sin duda la base fundamental del aprendizaje automático, que básicamente consiste en una serie de algoritmos capaces de analizar grandes cantidades de datos para deducir cuál es el resultado más óptimo para un determinado problema”, menciona José Luis Espinoza, científico de datos de BBVA México [26].

El machine learning ha avanzado de forma exponencial durante los últimos 20 años, y es responsable de crear programas capaces de derrotar a jugadores humanos en distintos juegos para lograr esto es, en concreto, el aprendizaje por refuerzo que es una de las cuatro clasificaciones de estilos de aprendizaje en función de la salida esperada y del tipo de entrada [26]:

1. Aprendizaje supervisado.
2. Aprendizaje no supervisado.
3. Aprendizaje semisupervisado.
4. Aprendizaje por refuerzo.

3.2. Conjunto de entrenamiento y conjunto de prueba (Training set y Test set)

Los algoritmos de Machine Learning aprenden de los datos con los que los entrenamos. A partir de ellos, intentan encontrar o inferir un patrón que les permita predecir el resultado para un nuevo caso. Pero, para poder calibrar si un modelo funciona, necesitaremos probarlo con un conjunto de datos diferente. Por ello, en todo proceso de Machine Learning, el conjunto de datos se divide en dos subconjuntos:

1. Conjunto de entrenamiento: Es un subconjunto para entrenar un modelo.
2. Conjunto de prueba: Es un subconjunto para probar el modelo entrenado.

Los datos de entrenamiento o training data son los datos que usamos para entrenar un modelo. La calidad de nuestro modelo de aprendizaje automático va a ser directamente proporcional a la calidad de los datos. Por ello las labores de limpieza o depuración es muy importante.

Por otro lado, los datos de prueba, o testing data, son los datos que nos “reservamos” para comprobar si el modelo que hemos generado a partir de los datos de entrenamiento “funciona”. Es decir, si las respuestas predichas por el modelo para un caso totalmente nuevo son acertadas o no.

Es importante que el conjunto de datos de prueba tenga un volumen suficiente como para generar resultados estadísticamente significativos, y a la vez, que sea representativo del conjunto de datos global.

Normalmente el conjunto de datos se suele repartir en un 70% de datos de entrenamiento y un 30% de datos de prueba, pero se puede variar la proporción según el caso. Lo importante es ser siempre conscientes de que hay que evitar el sobreajuste u “overfitting”.

3.3. Sobreajuste u overfitting

El sobreajuste es un concepto en machine learning, que ocurre cuando un modelo se ajusta exactamente a sus datos de entrenamiento. Cuando esto sucede, el algoritmo desafortunadamente no puede funcionar con precisión contra datos no conocidos, frustrando su propósito.

Cuando se construyen algoritmos de machine learning, se utiliza un subconjunto de datos para entrenar el modelo. Sin embargo, cuando el modelo se entrena durante demasiado tiempo con o cuando el modelo es demasiado complejo, puede comenzar a aprender el “ruido” o información irrelevante dentro del conjunto de datos. Cuando el modelo memoriza el ruido y se ajusta demasiado al conjunto de entrenamiento, el modelo se “sobreajusta” y no puede generalizar bien a los nuevos datos. Si un modelo no puede generalizarse bien a nuevos datos, entonces no podrá realizar las tareas de clasificación o

predicción para las que fue diseñado. Si los datos de entrenamiento tienen una tasa de error baja y los datos de prueba tienen una tasa de error alta, indica un sobreajuste [29].

¿Cómo podemos evitar el sobreajuste?

A continuación, se muestran una serie de técnicas que pueden ayudar a evitar el sobreajuste [29]:

- **Parada anticipada:** Este método busca pausar el entrenamiento antes de que el modelo comience a aprender el ruido dentro del modelo. Este enfoque corre el riesgo de detener el proceso de entrenamiento demasiado pronto, lo que lleva al problema opuesto de subajuste. Encontrar el punto óptimo entre el subajuste y el sobreajuste es el objetivo final aquí.
- **Entrenar con más datos:** Expandir el conjunto de entrenamiento para incluir más datos puede aumentar la precisión del modelo al brindar más oportunidades para analizar la relación dominante entre las variables de entrada y salida. Dicho esto, este es un método más eficaz cuando se inyectan datos limpios y relevantes en el modelo. De lo contrario, podría continuar agregando más complejidad al modelo, haciendo que se sobreajuste.
- **Aumento de datos:** Si bien es mejor inyectar datos limpios y relevantes en sus datos de entrenamiento, a veces se agregan datos ruidosos para hacer que un modelo sea más estable. Sin embargo, este método debe realizarse con prudencia.
- **Selección de características:** Cuando construye un modelo, tendrá una serie de parámetros o características que se utilizan para predecir un resultado dado, pero muchas veces, estas características pueden ser redundantes para otras. La selección de características es el proceso de identificar las más importantes dentro de los datos de entrenamiento y luego eliminar las irrelevantes o redundantes.
- **Regularización:** Si se produce un sobreajuste cuando un modelo es demasiado complejo, tiene sentido que reduzcamos el número de funciones. Pero ¿qué pasa si no sabemos qué entradas eliminar durante el proceso de selección de funciones? Si no sabemos qué características eliminar de nuestro modelo, los métodos de regularización pueden ser particularmente útiles. La regularización aplica una penalización a los parámetros de entrada con los coeficientes más grandes, lo que posteriormente limita la cantidad de varianza en el modelo. Si bien existen varios métodos de regularización, como la regularización L1, la regularización de Lasso y la deserción, todos buscan identificar y reducir el ruido dentro de los datos.

3.4. Normalización de los datos

Para que funcionen mejor muchos algoritmos de Machine Learning, hay que normalizar las variables de entrada al algoritmo. Normalizar significa, en este caso, comprimir o extender los valores de la variable para que estén en un rango definido. Lo primero de todo es preprocesar los datos para facilitarle la faena lo máximo posible a nuestra red. En caso contrario, puede pasarnos que, por ejemplo, al tener datos que van de 0 a 255, la red

nunca llegue a aprender nada [32].

El objetivo de la normalización es transformar los atributos para que estén en una escala similar. Esto mejora el rendimiento y la estabilidad del entrenamiento del modelo.

3.4.1. Técnicas de normalización

Cuatro técnicas comunes de normalización que pueden ser útiles son las siguientes:

- Escalamiento a un rango.
- Recorte.
- Escalamiento logarítmico.
- Puntuación z.

En la Figura 3.1 se muestra el efecto de cada técnica de normalización en la distribución de la característica sin procesar (precio). Los gráficos se basan en el conjunto de datos del Yearbook de Automotive de Ward de 1985 que forma parte del UCI Machine Learning Repository en Automobile Data Set [32].

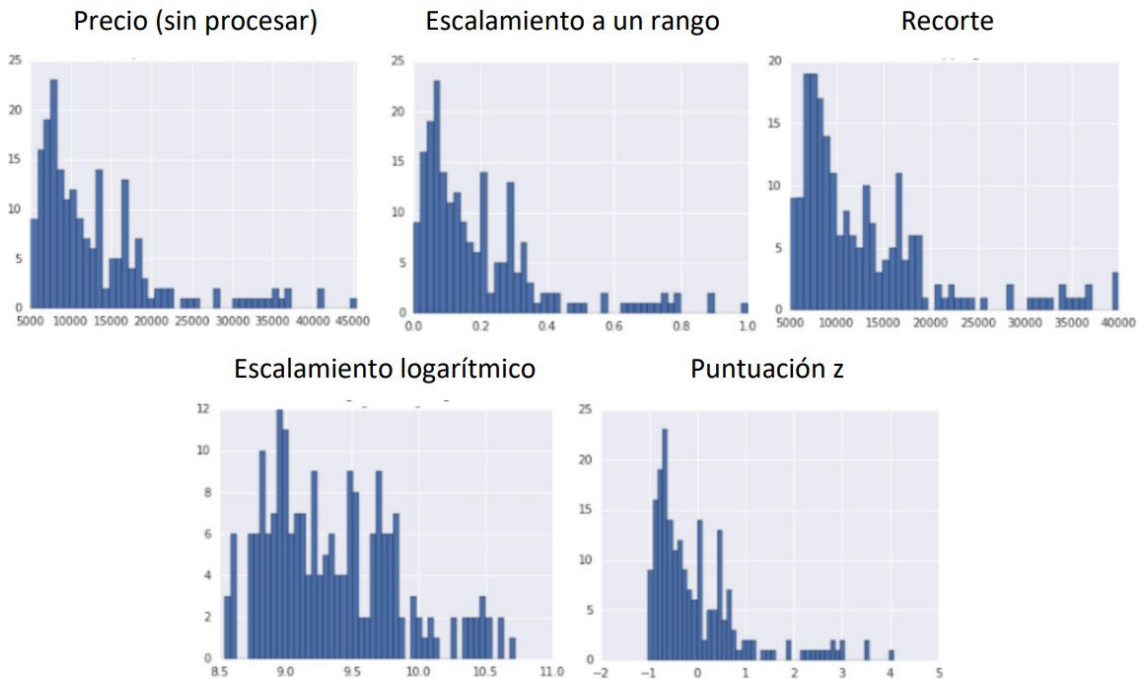


Figura 3.1: Técnicas de normalización (Figura extraída de [32]).

Escalamiento a un rango

El escalamiento significa convertir los valores de atributos de punto flotante de su rango natural (por ejemplo, de 100 a 900) en un rango estándar, por lo general, de 0 a 1 (o a veces de -1 a +1). La siguiente fórmula nos ayuda a escalar nuestros datos a un rango:

$$x' = \frac{x - x_{min}}{x_{max} - x_{min}}, \quad (3.1)$$

donde:

- x_{min} es el dato más pequeño del conjunto de datos.
- x_{max} es el dato más grande del conjunto de datos.

El escalamiento a un rango es una buena opción cuando se cumplen las siguientes condiciones:

- Se conoce los límites inferior y superior aproximados de los datos con pocos valores atípicos o sin valores atípicos.
- Los datos se distribuyen de manera aproximadamente uniforme en ese rango.

Recorte

Si el conjunto de datos contiene valores atípicos extremos, se puede probar el recorte de atributos, que limita todos los valores de atributos anteriores (o inferiores) a un valor determinado. Por ejemplo, se puede recortar todos los valores de temperatura por encima de 40 para que sean exactamente 40. Puedes aplicar el recorte de atributos antes o después de otras normalizaciones.

Escalamiento logarítmico

El escalamiento logarítmico es útil cuando varios de tus valores tienen muchos puntos, mientras que la mayoría de los demás valores tienen pocos puntos. Esta distribución de datos se conoce como distribución de ley de energía. Las calificaciones de películas son un buen ejemplo. En la Figura 3.2, la mayoría de las películas tienen muy pocas calificaciones (los datos en la cola), mientras que unas pocas tienen muchas calificaciones (los datos en la cabeza). El escalamiento logarítmico cambia la distribución, lo que ayuda a mejorar el rendimiento del modelo.

La fórmula para realizar esta técnica es la siguiente:

$$x' = \log(x).$$

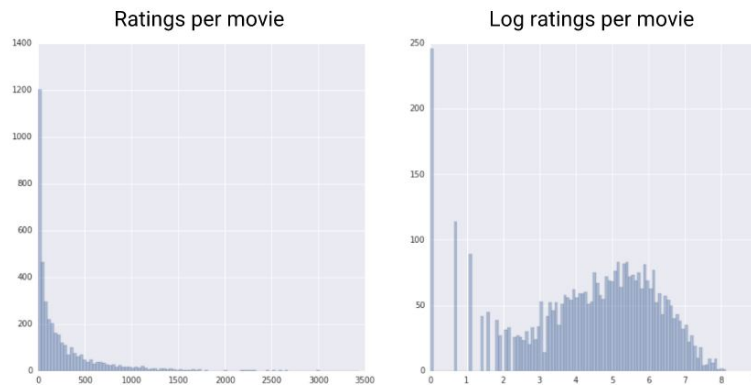


Figura 3.2: Ejemplo de la técnica escalamiento logarítmico (Figura extraída de [32]).

Puntuación Z

La puntuación Z es una variación del escalamiento que representa la cantidad de desviaciones estándar de la media. Usa la puntuación Z para garantizar que las distribuciones de atributos tengan una media = 0 y un valor de desviación estándar igual a 1. Es útil cuando hay algunos valores atípicos, pero no tan extremo que necesitas recortar.

La fórmula para calcular la puntuación z de un punto, x , es la siguiente:

$$x' = \frac{(x - \mu)}{\sigma},$$

donde:

- μ es la media del conjunto de datos.
- σ es la desviación estándar del conjunto de datos, recordando que $\sigma = \sqrt{Var(x_i)}$.

3.5. Perceptrón

Los perceptrones fueron desarrollados en las décadas de 1950 y 1960 por el científico Frank Rosenblatt, inspirado en trabajos anteriores de Warren McCulloch y Walter Pitts. Hoy en día, es más común utilizar otros modelos de neuronas artificiales. Pero para entender los otros tipos de neuronas, vale la pena comprender primero los perceptrones.

Entonces, ¿cómo funcionan los perceptrones? Un perceptrón toma varias entradas binarias, x_1, x_2, \dots , y produce una sola salida binaria:

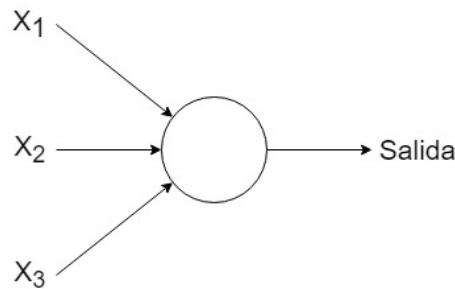


Figura 3.3: Diagrama de perceptrón (De autoría propia).

En la Figura 3.3 se muestra un perceptrón que tiene tres entradas, x_1, x_2, x_3 . En general podría tener más o menos entradas. Rosenblatt propuso una regla simple para calcular la salida. Introdujo pesos, w_1, w_2, \dots, w_j números reales que expresan la importancia de las entradas respectivas a la salida donde $w_j \in (0, 1)$. La salida de la neurona es 0 o 1, está determinada por si la suma ponderada $\sum_j w_j x_j$ es menor o mayor que algún valor de umbral. Al igual que los pesos, el umbral es un número real que es un parámetro de la neurona. En términos algebraicos es lo siguiente:

$$\text{Salida} = \begin{cases} 0 & \text{si } \sum_j w_j x_j \leq \text{umbral.} \\ 1 & \text{si } \sum_j w_j x_j > \text{umbral.} \end{cases} \quad (3.2)$$

Simplifiquemos la forma en que describimos los perceptrones. La condición $\sum_j w_j x_j > \text{umbral}$ es engorrosa y se puede hacer dos cambios de notación para simplificarla. El primer cambio es escribir $\sum_j w_j x_j$ como un producto escalar, $w \cdot x = \sum_j w_j x_j$, donde w y x son vectores con la misma dimensión cuyos componentes son los pesos y las entradas, respectivamente. El segundo cambio es mover el umbral al otro lado de la desigualdad y reemplazarlo por lo que se conoce como el sesgo del perceptrón, $b \equiv -\text{umbral}$. Usando el sesgo en lugar del umbral, la regla del perceptrón se puede reescribir:

$$\text{Salida} = \begin{cases} 0 & \text{si } w \cdot x + b \leq 0. \\ 1 & \text{si } w \cdot x + b > 0. \end{cases} \quad (3.3)$$

Introducir el sesgo es solo un pequeño cambio en cómo describimos los perceptrones, pero la ventaja es que conduce a más simplificaciones de notación. Debido a esto, no

usaremos el umbral, siempre usaremos el sesgo.

En general, dentro de cada neurona ocurre la siguiente ecuación:

$$XW^t = (x_1, x_2, \dots, x_n) \begin{pmatrix} w_1 \\ w_2 \\ \vdots \\ w_n \end{pmatrix} = \sum_{i=1} x_i w_i. \quad (3.4)$$

El valor de la ecuación anterior se le conoce como el valor neto de entrada a la neurona. Luego, este valor se pasa por una función, conocida como función de activación, ver Ecuación (3.5), que transforma el valor neto de entrada en un valor de salida.

$$\phi \left(\sum_{i=1} x_i w_i \right). \quad (3.5)$$

En la Figura (3.4) se observa el comportamiento de una red neuronal artificial.

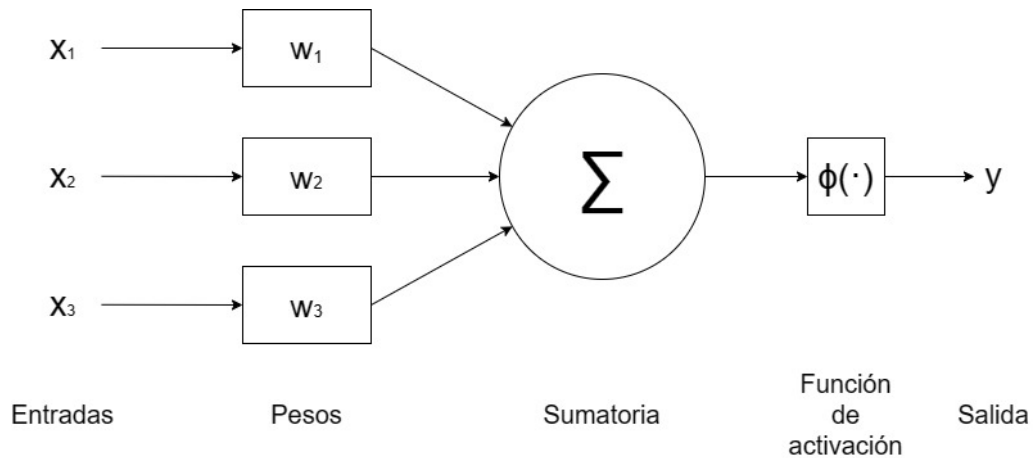


Figura 3.4: Comportamiento de una neurona artificial (De autoría propia).

3.6. Función de activación

En la salida de la neurona, existe un filtro o umbral, que modifica el valor del resultado o impone un límite que se debe sobrepasar para decidir si una neurona debe activarse o no. Esta función se denomina función de activación. Una función de activación es, por tanto, una función que transmite la información generada por la combinación lineal de los pesos y las entradas, es decir, son la manera de transmitir la información por las conexiones de salida. La información puede ser transmitida sin modificación, esto lo hace la función identidad. Dado que el objetivo de las redes neuronales es poder resolver problemas cada vez más complejos, las funciones de activación a menudo hacen que el modelo no sea

lineal.

Las funciones de activación más utilizadas o conocidas son:

- **Función Lineal.**

La función de activación lineal, también conocida como sin activación o función identidad es donde la activación es proporcional a la entrada. La función no hace nada con la suma ponderada de la entrada, simplemente devuelve el valor que se le dio. La función de activación lineal es la siguiente:

$$f(x) = x, \quad \text{para } x \in [-\infty, \infty]. \quad (3.6)$$

En la Figura (3.5) podemos ver su comportamiento gráfico.

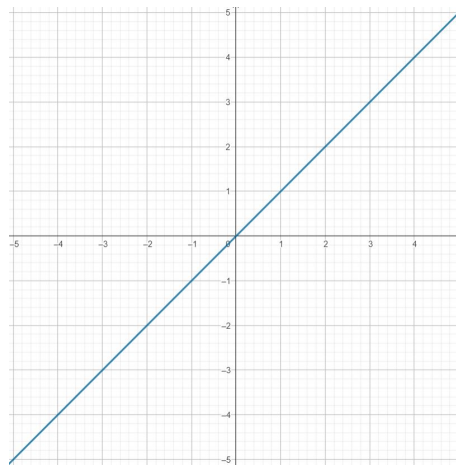


Figura 3.5: Gráfica de la función lineal.

- **Función Escalón.**

La función escalón es también conocida como función de paso binario o función umbral, esta función depende un valor de umbral que decide si una neurona debe activarse o no.

La entrada alimentada a la función de activación se compara con un cierto umbral; si la entrada es mayor que ella, entonces la neurona se activa; de lo contrario, se desactiva. Matemáticamente se puede representar como:

$$f(x) = \begin{cases} 0 & \text{para } x < 0, \\ 1 & \text{para } x \geq 0. \end{cases} \quad (3.7)$$

En la Figura (3.6) podemos ver su comportamiento gráfico.

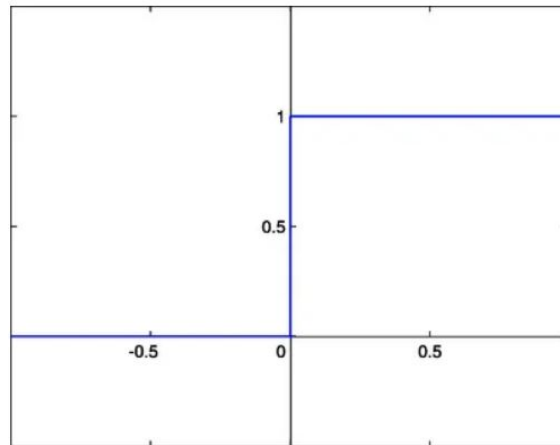


Figura 3.6: Gráfica de la función escalón.

- Función Sigmoide.

Esta función también conocida como función logística, está en un rango de valores de salida entre cero y uno por lo que la salida es interpretada como una probabilidad. Si se evalúa la función con valores de entrada muy negativos, la función será igual a cero, si se evalúa en cero la función dará 0.5 y en valores altos su valor es aproximadamente a 1.

Matemáticamente se puede representar como:

$$f(x) = \frac{1}{1 + e^{-x}}, \quad \text{para } x \in [0, 1]. \quad (3.8)$$

Actualmente la sigmoide no es una función muy utilizada debido a que no está centrada y esto afecta en el aprendizaje y entrenamiento de la neurona por lo que influye con el problema de desaparición de gradiente.

En la Figura (3.7) podemos ver su comportamiento gráfico.

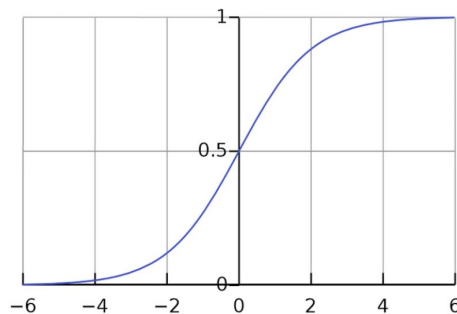


Figura 3.7: Gráfica de la función sigmoide.

- Función ReLu.

ReLu (Rectified Lineal Unit) significa Unidad lineal rectificada. Está función es la más utilizada en este momento debido a que permite el aprendizaje muy rápido en las redes neuronales. Si a esta función se le da valores de entrada muy negativos el resultado es cero, pero si se le da valores positivos queda igual y además el gradiente de esta función será cero en el segundo cuadrante y uno en el primer cuadrante. Cuando se tiene que la función es igual a cero y su derivada también lo es se genera lo que es la muerte de neuronas.

Matemáticamente se puede representar como:

$$f(x) = \max(0, x). \quad (3.9)$$

En la Figura (3.8) podemos ver su comportamiento gráfico.

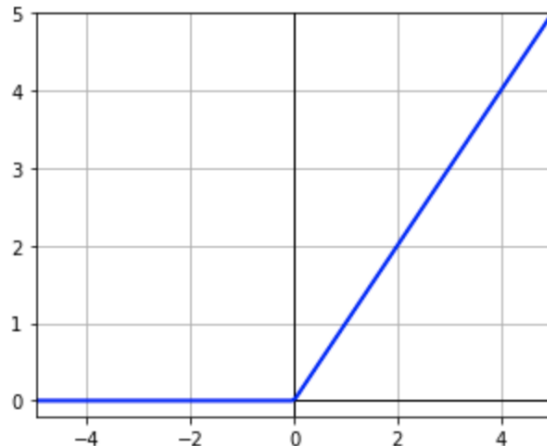


Figura 3.8: Gráfica de la función ReLu.

- Función Tangente Hiperbólica.

La función tangente hiperbólica es muy similar a la función de activación sigmoide, e incluso tiene la misma forma de S con una diferencia en el rango de salida de -1 a 1. En esta función, cuanto mayor sea la entrada (más positiva), más cercano será el valor de salida 1.0, mientras que cuanto menor sea la entrada (más negativa), más cerca estará la salida -1.0. Matemáticamente se puede representar como:

$$f(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}. \quad (3.10)$$

En la Figura (3.9) podemos ver su comportamiento gráfico.

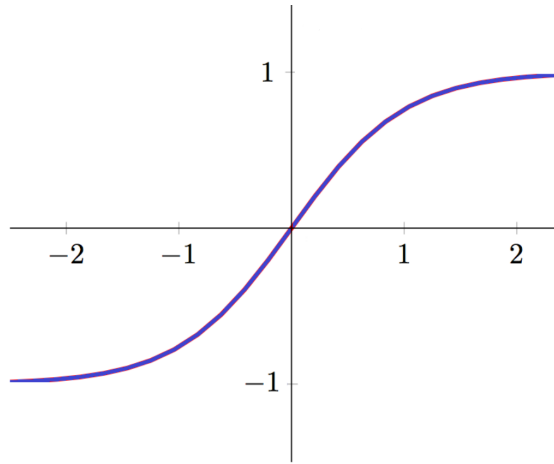


Figura 3.9: Gráfica de la función Tangente hiperbólica.

3.7. Red neuronal artificial

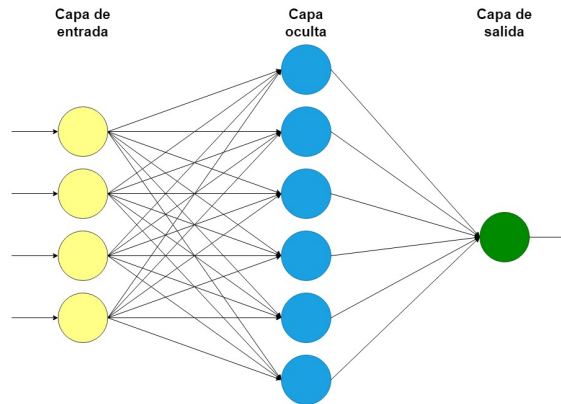
Las redes neuronales, también conocidas como redes neuronales artificiales, son un subconjunto del machine learning y están en el corazón de los algoritmos de aprendizaje profundo. Su nombre y estructura están inspirados en el cerebro humano, imitando la forma en que las neuronas biológicas se envían señales entre sí.

Las redes neuronales se componen de capas de neuronas artificiales, que contienen una capa de entrada, una o más capas ocultas y una capa de salida. Cada neurona, se conecta con otro y tiene asociado un peso y un umbral.

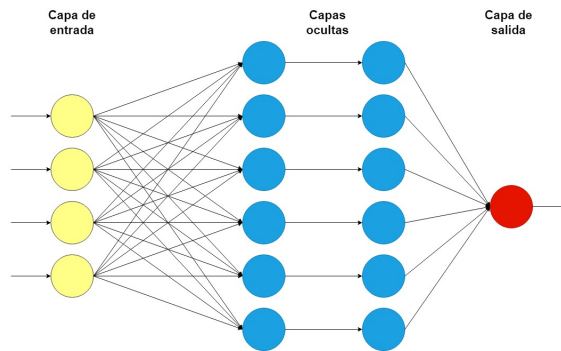
Las redes neuronales se basan en datos de entrenamiento para aprender y mejorar su precisión con el tiempo. Sin embargo, una vez que estos algoritmos de aprendizaje se ajustan con precisión, se convierten en herramientas poderosas en informática e inteligencia artificial, lo que nos permite clasificar y agrupar datos a alta velocidad [27].

3.8. La arquitectura de las redes neuronales

Como se mencionó anteriormente, una red neuronal está formada por capas por ejemplo en la Figura 3.10(a) se muestra una red neuronal donde la capa que está más a la izquierda en esta red se llama capa de entrada (input layer), y las neuronas dentro de la capa se llaman neuronas de entrada. La capa que está más a la derecha se le llama capa de salida (output layer) y contiene las neuronas de salida o, como en este caso, una única neurona de salida. La capa intermedia se denomina capa oculta (hidden layer), ya que las neuronas de esta capa no son ni entradas ni salidas, esta capa recibe los valores de la capa de entrada, ponderado por los pesos (flechas negras). La red anterior tiene una sola capa oculta, pero algunas redes tienen varias capas ocultas. Por ejem-



(a) Diagrama de una red neuronal con una capa oculta.



(b) Diagrama de una red neuronal con dos capas ocultas.

Figura 3.10: Ejemplos de redes neuronales (De autoría propia).

plo, en la Figura 3.10(b) se muestra una red de cuatro capas la cual tiene dos capas ocultas.

De manera un tanto confusa, y por razones históricas, estas redes de múltiples capas a veces se denominan perceptrones multicapa o MLP.

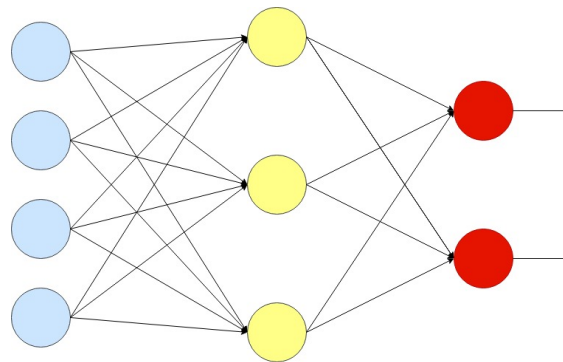
Si bien el diseño de las capas de entrada y salida de una red neuronal suele ser sencillo, el diseño de las capas ocultas puede ser todo un arte. En particular, no es posible resumir el proceso de diseño de las capas ocultas con algunas reglas generales simples. En cambio, los investigadores de redes neuronales han desarrollado muchas heurísticas de diseño para las capas ocultas, que ayudan a las personas a obtener el comportamiento que desean de sus redes. Por ejemplo, tales heurísticas se pueden usar para ayudar a determinar cómo compensar la cantidad de capas ocultas con el tiempo requerido para entrenar la red [34].

Hasta ahora, hemos estado discutiendo redes neuronales donde la salida de una capa se usa como entrada para la siguiente capa. Este tipo de redes se denominan redes neuronales feedforward. Esto significa que no hay bucles en la red. Si tuviéramos

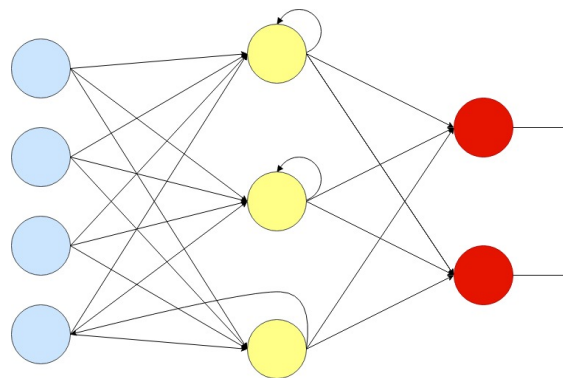
bucles, terminaríamos con situaciones en las que la entrada a la función de activación dependiera de la salida. Eso sería difícil de entender, por lo que no permitimos tales bucles.

Sin embargo, existen otros modelos de redes neuronales artificiales en los que son posibles los bucles de retroalimentación. Estos modelos se denominan redes neuronales recurrentes. La idea en estos modelos es tener neuronas que se disparen durante un período de tiempo limitado, antes de volverse inactivas. Ese disparo puede estimular otras neuronas, que pueden dispararse un poco más tarde, también por una duración limitada. Eso hace que se disparen aún más neuronas, y así, con el tiempo, obtenemos una cascada de disparos de neuronas. Los bucles no causan problemas en dicho modelo, ya que la salida de una neurona solo afecta su entrada en un momento posterior, no instantáneamente [34].

Las redes neuronales se pueden clasificar en términos generales en redes neuronales feedforward, redes neuronales recurrentes y sus híbridos, en la Figura 3.11 se muestra algunos ejemplos de estos [33].



(a) Red neuronal feedforward.



(b) Red neuronal recurrente.

Figura 3.11: Tipos de redes neuronales (De autoría propia).

3.9. Función de Costo

En un modelo de Machine Learning se busca tener un nivel alto de precisión para funcionar bien con aplicaciones del mundo real. Pero ¿cómo calcular la precisión del modelo?, es decir, ¿qué tan bien o mal funcionará nuestro modelo en el mundo real? En tal caso, surge la función de costo.

¿Qué es la función de costo?

Una función de costo es un parámetro importante que determina qué tan bien funciona un modelo de Machine Learning para un conjunto de datos determinado. Calcula la diferencia entre el valor esperado y el valor pronosticado.

Una vez que entrenamos nuestro modelo, queremos ver qué tan bien se está desempeñando. Aunque hay varias funciones de precisión que le indican el rendimiento de su modelo, no le darán información para mejorarlas. Por lo tanto, necesitamos una función que pueda encontrar cuándo el modelo es más preciso al encontrar el punto entre el modelo subentrenado y el sobreentrenado.

En otras palabras, "La función de costo mide qué tan equivocado está el modelo, al estimar la relación entre el parámetro X (entrada) y Y (salida)". Una función de costo a veces también se denomina función de pérdida y se puede estimar ejecutando iterativamente el modelo para comparar las predicciones estimadas con los valores conocidos.

Las funciones de costo pueden ser de varios tipos dependiendo del problema. No obstante, se pueden clasificar en tres tipos, que son los siguientes:

- Función de costo de regresión.
- Funciones de costo de clasificación binaria.
- Función de costo de clasificación multiclase.

Como en este trabajo se plantea un problema de regresión entonces describiremos las funciones de costo para este problema.

Función de costo de regresión

Los modelos de regresión se usan para hacer una predicción de las variables continuas, como el precio de productos, la predicción de la temperatura, las predicciones de acciones, etc. Las funciones de costo de regresión se calculan basado en la distancia, como:

$$\text{Costo} = \text{Dato real} - \text{Dato pronosticado}.$$

Las funciones de costo de regresión comúnmente utilizadas son las siguientes:

3.9.1. Error cuadrático medio

El Error cuadrático medio (MSE, por sus siglas en inglés, Mean Squared Error), es uno de los métodos de función de costo más utilizados. Mide el promedio de los errores al cuadrado, es decir, la diferencia entre el valor real y el valor pronosticado.

La fórmula para calcular el MSE se da a continuación:

$$C(w, b) = \frac{1}{2n} \sum_x \|y(x) - a\|^2, \quad (3.11)$$

donde w denota la colección de todos los pesos en la red, b todos los sesgos, n es el número total de entradas de entrenamiento, a es el vector de salidas de la red cuando se ingresa x y la suma es sobre todas las entradas de entrenamiento, x . Por supuesto, la salida a depende de x , w y b , pero para simplificar la notación, no se indicada explícitamente esta dependencia.

Al inspeccionar la forma de la función de costo cuadrática, vemos que $C(w, b)$ no es negativa, ya que todos los términos de la suma no son negativos. Además, la función de costo $C(w, b)$ se vuelve pequeño, es decir, $C(w, b) \approx 0$, precisamente cuando $y(x)$ es aproximadamente igual a la salida, a , para todas las entradas de entrenamiento, x [34].

En el MSE, cada error se eleva al cuadrado y ayuda a reducir una pequeña desviación en la predicción. Pero si el conjunto de datos tiene valores atípicos, entonces el cuadrado de este error aumentará aún más el error. Por lo tanto, podemos decir que el MSE es menos resistente a los valores atípicos.

3.9.2. Error absoluto medio

El error absoluto medio (MAE, por sus siglas en inglés, Mean Absolute Error), consiste en promediar el error absoluto de las predicciones. La fórmula para calcular el error absoluto medio se da a continuación:

$$C(w, b) = \frac{1}{n} \sum_x |y(x) - a|, \quad (3.12)$$

donde w denota la colección de todos los pesos en la red, b todos los sesgos, n es el número total de entradas de entrenamiento, a es el vector de salidas de la red cuando se ingresa x y la suma es sobre todas las entradas de entrenamiento, x .

El MAE no se ve afectado por el ruido o los valores atípicos, por lo que ofrece mejores resultados si el conjunto de datos tiene ruido o valores atípicos a comparación del MSE.

3.10. Aprendizaje con Descenso de Gradiente (Gradient descent)

Ahora que tenemos un diseño para nuestra red neuronal, ¿cómo puede aprender nuestra red?

Lo que nos gustaría es un algoritmo que nos permita encontrar pesos (w) y sesgos (b) para que la salida de la red se aproxime a $y(x)$ para todas las entradas de entrenamiento x . Para cuantificar qué tan bien estamos logrando este objetivo se usa alguna función de costo.

El objetivo de nuestro algoritmo de entrenamiento será minimizar la función de costo $C(w, b)$ en función de los pesos y sesgos. En otras palabras, queremos encontrar un conjunto de pesos y sesgos que hagan que el costo sea lo más pequeño posible. Lo haremos usando un algoritmo conocido como Descenso de Gradiente [34].

Descenso de Gradiente es un algoritmo de optimización que se usa comúnmente para entrenar modelos de machine learning y redes neuronales. Los datos de entrenamiento ayudan a estos modelos a aprender con el tiempo, y la función de costo dentro del Descenso de Gradiente actúa específicamente como un barómetro, midiendo su precisión con cada iteración de actualizaciones de parámetros. Hasta que la función sea cercana o igual a cero, el modelo continuará ajustando sus parámetros para producir el menor error posible. Una vez que los modelos de aprendizaje automático se optimizan para la precisión, pueden ser herramientas poderosas para aplicaciones de inteligencia artificial (IA) y ciencias de la computación [28].

¿Cómo funciona el algoritmo Descenso de Gradiente?

Para comprender fácilmente el algoritmo Descenso de Gradiente supongamos que estamos tratando de minimizar alguna función, $C(v)$. Esta podría ser cualquier función de valor real de muchas variables, $v = v_1, v_2, \dots$. Tengamos en cuenta que se reemplazó la notación w y b por v para enfatizar que esto podría ser cualquier función; ya no estamos pensando específicamente en el contexto de las redes neuronales. Para minimizar $C(v)$, ayuda imaginar a C como una función de solo dos variables, a las que llamaremos v_1 y v_2 (véase Figura 3.12):

Y también utilicemos la siguiente analogía, pensemos en nuestra función como una especie de valle e imaginamos una pelota rodando por la ladera del valle. Nuestra experiencia cotidiana nos dice que la pelota finalmente rodará hasta el fondo del valle. ¿Quizás podamos usar esta idea como una forma de encontrar un mínimo para la función? Elegiríamos al azar un punto de partida para una pelota (imaginaria) y luego simularíamos el movimiento de la pelota mientras rodaba hacia el fondo del valle. Podríamos hacer esta simulación simplemente calculando las derivadas (y quizás algunas segundas derivadas) de C ; esas derivadas nos dirían todo lo que necesitamos saber sobre la forma local del valle y

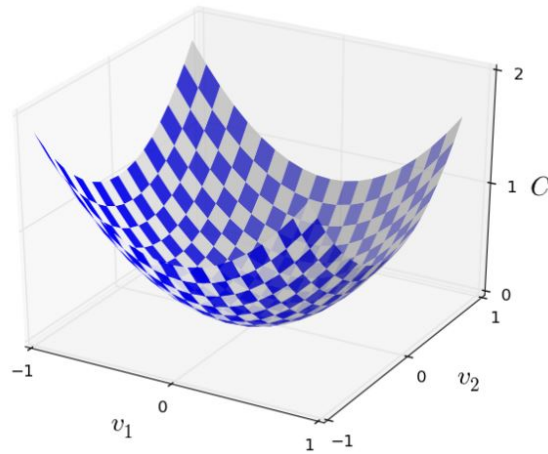


Figura 3.12: Ejemplo del funcionamiento del algoritmo Descenso de Gradiente (Figura extraída de [34]).

por lo tanto, cómo debería rodar nuestra bola.

En realidad, no vamos a tomarnos la analogía de la pelota rodando tan en serio, estamos diseñando un algoritmo para minimizar C , no desarrollando una simulación precisa de las leyes de la física.

Pensemos en lo que sucede cuando movemos la pelota una pequeña cantidad de Δv_1 en la dirección v_1 y una pequeña cantidad de Δv_2 en la dirección v_2 . El cálculo nos dice que C cambia de la siguiente manera:

$$\Delta C \approx \frac{\partial C}{\partial v_1} \Delta v_1 + \frac{\partial C}{\partial v_2} \Delta v_2. \tag{3.13}$$

Vamos a encontrar una manera de elegir Δv_1 y Δv_2 para que ΔC sea negativo; es decir, los elegiremos para que la pelota ruede hacia el valle. Para descubrir cómo hacer tal elección, es útil definir Δv como el vector de cambios en v , $\Delta v \equiv (\Delta v_1, \Delta v_2)^T$, convirtiendo los vectores de fila en vectores de columna. También definiremos el gradiente de C como el vector de derivadas parciales, $\left(\frac{\partial C}{\partial v_1}, \frac{\partial C}{\partial v_2}\right)^T$. Denotamos el vector gradiente por ∇C , es decir:

$$\nabla C \equiv \left(\frac{\partial C}{\partial v_1}, \frac{\partial C}{\partial v_2}\right)^T. \tag{3.14}$$

Con lo anterior, la Expresión (3.13) para ΔC se puede reescribir como:

$$\Delta C \approx \nabla C \cdot \Delta v. \tag{3.15}$$

Esta ecuación ayuda a explicar por qué ∇C se llama vector gradiente: ∇C relaciona

los cambios en v con los cambios en C , tal como esperaríamos que hiciera algo llamado gradiente. Pero lo realmente emocionante de la ecuación es que nos permite ver cómo elegir Δv para hacer que ΔC sea negativo. En particular, supongamos que elegimos:

$$\Delta v = -\eta \nabla C, \tag{3.16}$$

donde η es un parámetro positivo pequeño (conocido como tasa de aprendizaje). Entonces la Ecuación (3.15) nos dice que $\Delta C \approx -\eta \nabla C \cdot \nabla C = -\eta \|\nabla C\|^2$. Debido a que $\|\nabla C\|^2 \geq 0$, esto garantiza que $\Delta C \leq 0$, es decir, C siempre disminuirá, nunca aumentará, si cambiamos v de acuerdo con la Ecuación de (3.16). (Dentro, por supuesto, de los límites de la aproximación en la Ecuación (3.15)). ¡Esta es exactamente la propiedad que queríamos! Y entonces tomaremos la Ecuación (3.16) para definir la “ley de movimiento” para la pelota en nuestro algoritmo de descenso de gradiente. Es decir, usaremos la Ecuación (3.16) para calcular un valor para Δv , luego moveremos la posición v de la pelota en esa cantidad:

$$v \rightarrow v' = v - \eta \nabla C. \tag{3.17}$$

Entonces usaremos esta regla de actualización nuevamente, para hacer otro movimiento. Si seguimos haciendo esto, una y otra vez, seguiremos disminuyendo C hasta que lleguemos a un mínimo global.

En resumen, la forma en que funciona el algoritmo Descenso de Gradiente es calcular repetidamente el gradiente ∇C y luego moverse en la dirección opuesta, cayendo por la pendiente del valle. Podemos visualizarlo en la Figura (3.13):

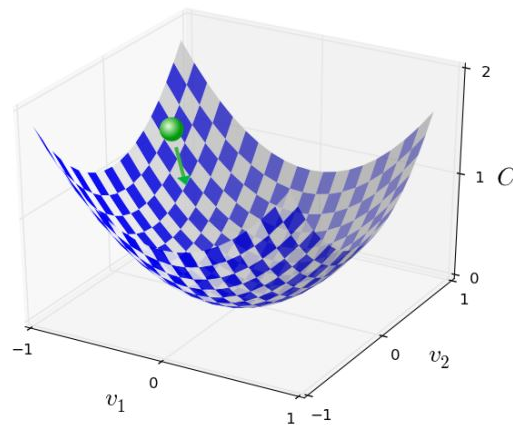


Figura 3.13: Segundo ejemplo del funcionamiento del algoritmo Descenso de Gradiente (Figura extraída de [34]).

Para lograr que el algoritmo Descenso de Gradiente funcione correctamente, debemos elegir que la tasa de aprendizaje η sea lo suficientemente pequeña como para que la Ecuación (3.15) sea una buena aproximación. Si no lo hacemos, podríamos terminar con $\Delta C > 0$. Al mismo tiempo, no queremos que η sea demasiado pequeño, ya que eso hará que los cambios de Δv sean pequeños y por lo tanto, el algoritmo Descenso de Gradiente funcionará muy lentamente. En las implementaciones prácticas, η suele variar, de modo que la Ecuación (3.15) sigue siendo una buena aproximación.

Se ha explicado el algoritmo Descenso de Gradiente cuando C es una función de solo dos variables. Pero, de hecho, todo funciona igual de bien incluso cuando C es una función de muchas más variables. Supongamos en particular que C es una función de m variables, v_1, \dots, v_m . Entonces el cambio ΔC en C es producido por un pequeño cambio $\Delta v = (\Delta v_1, \dots, \Delta v_m)^T$ es:

$$\Delta C \approx \nabla C \cdot \Delta v, \tag{3.18}$$

donde el gradiente ∇C es el vector:

$$\nabla C \equiv \left(\frac{\partial C}{\partial v_1}, \dots, \frac{\partial C}{\partial v_m} \right)^T. \tag{3.19}$$

Al igual que para el caso de dos variables, podemos elegir:

$$\Delta v = -\eta \nabla C, \tag{3.20}$$

y estamos garantizados que nuestra Expresión (3.18) (aproximada) para ΔC será negativa. Esto nos da una forma de seguir el gradiente al mínimo, incluso cuando C es una función de muchas variables, aplicando repetidamente la regla de actualización:

$$v \rightarrow v' = v - \eta \nabla C. \tag{3.21}$$

Ahora ¿Cómo podemos aplicar el algoritmo Descenso de Gradiente para aprender en una red neuronal? La idea es usar el algoritmo Descenso de Gradiente para encontrar los pesos w_k y los sesgos b_l que minimizan la función de costo. Para ver cómo funciona esto, reformulemos la regla de actualización del algoritmo, con los pesos y sesgos reemplazando las variables v_j . En otras palabras, nuestra “posición” ahora tiene componentes w_k y b_l , y el vector gradiente ∇C tiene componentes correspondientes $\partial C / \partial w_k$ y $\partial C / \partial b_l$.

Escribiendo la regla de actualización del algoritmo Descenso de Gradiente en términos de componentes, tenemos:

$$w_k \rightarrow w'_k = w_k - \eta \frac{\partial C}{\partial w_k}. \tag{3.22}$$

$$b_l \rightarrow b'_l = b_l - \eta \frac{\partial C}{\partial b_l}. \tag{3.23}$$

Al aplicar repetidamente esta regla de actualización, podemos bajar la colina y, encontrar un mínimo de la función de costo. En otras palabras, esta es una regla que se puede usar para que una red neuronal aprenda.

3.10.1. Descenso de gradiente estocástico (Stochastic Gradient Descent)

Desafortunadamente el algoritmo Descenso de Gradiente en la práctica, cuando la cantidad de entradas es muy grande, puede llevar mucho tiempo y por lo tanto, el aprendizaje ocurre lentamente. Para hacer más rápido el algoritmo podemos utilizar el algoritmo llamado Descenso de gradiente estocástico.

El cual consiste en tomar un subconjunto para estimar el gradiente ∇C calculando ∇C_x para una pequeña muestra de entradas de entrenamiento elegidas al azar. Al promediar esta pequeña muestra, resulta que podemos obtener rápidamente una buena estimación del verdadero gradiente ∇C , y esto ayuda a acelerar el algoritmo Descenso de Gradiente y por lo tanto el aprendizaje [34].

Para hacer estas ideas más precisas, el Descenso de gradiente estocástico funciona seleccionando al azar un pequeño número m de entradas de entrenamiento elegidas al azar. Etiquetaremos esas entradas de entrenamiento aleatorias como X_1, X_2, \dots, X_m , y se refieren a ellos como un mini lote (mini-batch). Siempre que el tamaño de la muestra m sea lo suficientemente grande, esperamos que el valor promedio de ∇C_{X_j} sea aproximadamente igual al promedio de todos los ∇C_x , es decir:

$$\frac{\sum_{j=1}^m \nabla C_{X_j}}{m} \approx \frac{\sum_x \nabla C_x}{n} = \nabla C, \quad (3.24)$$

donde la segunda suma es sobre todo el conjunto de datos de entrenamiento. Intercambiando lados obtenemos:

$$\nabla C \approx \frac{1}{m} \sum_{j=1}^m \nabla C_{X_j}, \quad (3.25)$$

lo cual nos dice que podemos estimar el gradiente general calculando gradientes solo para el mini-lote elegido al azar.

Para conectar esto explícitamente con el aprendizaje en las redes neuronales, supongamos que w_k y b_l denotan los pesos y sesgos en nuestra red neuronal. Luego, el Descenso de gradiente estocástico funciona seleccionando un mini-lote elegido al azar de entradas de entrenamiento y entrenando así tenemos:

$$w_k \rightarrow w'_k = w_k - \frac{\eta}{m} \sum_j \frac{\partial C_{X_j}}{\partial w_k} \quad (3.26)$$

$$b_l \rightarrow b'_l = b_l - \frac{\eta}{m} \sum_j \frac{\partial C_{X_j}}{\partial b_l}, \quad (3.27)$$

donde las sumas son sobre todos los datos de entrenamiento X_j en el mini-lote actual. Luego elegimos otro mini-lote elegido al azar y entrenamos con esos. Y así sucesivamente,

hasta que hayamos agotado las entradas de entrenamiento, a lo que se dice que completa una **época** de entrenamiento. En ese momento comenzamos de nuevo con una nueva época.

3.11. Aprendizaje con Backpropagation

Anteriormente vimos cómo las redes neuronales pueden aprender sus pesos y sesgos utilizando el algoritmo Descenso de gradiente. Sin embargo, hubo una brecha en la explicación: no discutimos cómo calcular el gradiente de la función de costo. Un algoritmo rápido para calcular tales gradientes es conocido por el nombre de Backpropagation.

El algoritmo de backpropagation se introdujo originalmente en la década de 1970, pero su importancia no se apreció por completo hasta un famoso artículo de 1986 de David Rumelhart, Geoffrey Hinton y Ronald Williams. Ese documento describe varias redes neuronales en las que el algoritmo backpropagation funciona mucho más rápido que los enfoques anteriores de aprendizaje, lo que hace posible el uso de redes neuronales para resolver problemas que antes eran insolubles [34].

El aprendizaje con el algoritmo Backpropagation es el más popular para realizar tareas de aprendizaje supervisado. No solo se usa para entrenar redes feedforward, sino que también se adapta a redes neuronales recurrentes. Gracias al algoritmo Backpropagation, una red neuronal puede estar formada por múltiples capas [33].

En el corazón del algoritmo backpropagation hay una expresión para la derivada parcial $\partial C/\partial w$ de la función de costo C con respecto a cualquier peso w (o sesgo b) en la red. La expresión nos dice qué tan rápido cambia el costo cuando cambiamos los pesos y sesgos. Y si bien la expresión es un tanto compleja, también tiene una belleza, ya que cada elemento tiene una interpretación natural e intuitiva. Y así, el backpropagation no es solo un algoritmo rápido para aprender. En realidad, nos brinda información detallada sobre cómo cambiar los pesos y sesgos para cambiar el comportamiento general de la red [34].

¿Como funciona el algoritmo Backpropagation?

El objetivo del algoritmo backpropagation es calcular las derivadas parciales $\partial C/\partial w$ y $\partial C/\partial b$ de la función de costo C con respecto a cualquier peso w o sesgo b en la red. Para explicar el algoritmo necesitamos la siguiente notación:

- w_{jk}^l : Peso del enlace entre la neurona k de la capa $l - 1$ y la neurona j de la capa l .
- b_j^l : Sesgo de la neurona j en la capa l .
- a_j^l : Activación (salida) de la neurona j en la capa l , donde $a_j^l = \sigma(\sum_k w_{jk}^l a_k^{l-1} + b_j^l)$.

En la Figura 3.14 se encuentra un diagrama donde se muestran ejemplos del uso de estas notaciones, por ejemplo, muestra el peso de una conexión desde la cuarta neurona en la segunda capa hasta la segunda neurona en la tercera capa de una red.

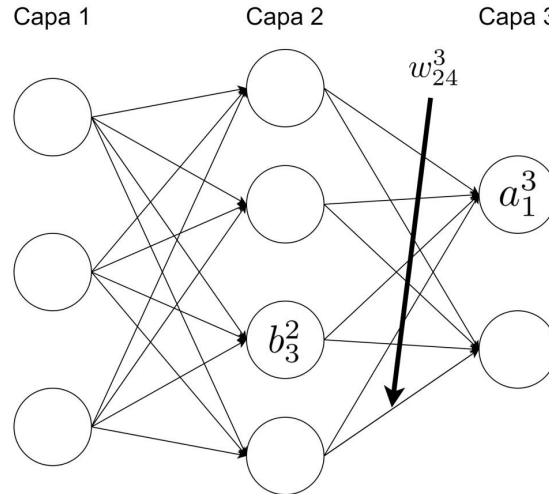


Figura 3.14: Ejemplo del uso de la notación para el algoritmo Backpropagation (Figura extraída de [34]).

Al argumento de la función σ le llamaremos z_j^l , en otras palabras $z_j^l = \sum_k w_{jk}^l a_k^{l-1} + b_j^l$.

Para emplear la notación compacta, es necesario vectorizar la función σ (función de activación), es decir $\sigma(\vec{z})_i = \sigma(z_i)$.

De modo que podemos escribir:

$$a^l = \sigma(w^l a^{l-1} + b^l), \quad \text{donde} \quad z^l \equiv w^l a^{l-1} + b^l \quad (3.28)$$

por lo que:

$$a^l = \sigma(z^l).$$

Ahora como:

$$C = \frac{1}{n} \sum_x C_x$$

entonces:

$$\frac{\partial C}{\partial w} = \frac{1}{n} \sum_x \frac{\partial C_x}{\partial w}. \quad (3.29)$$

Ahora nos enfocaremos en calcular:

$$\frac{\partial C_x}{\partial w} \quad \text{y} \quad \frac{\partial C_x}{\partial b}.$$

Definiremos una variable auxiliar llamada error:

$$\delta_j^l \equiv \frac{\partial C_x}{\partial z_j^l} \quad (3.30)$$

o bien

$$\delta^l = \begin{pmatrix} \delta_1^l \\ \delta_2^l \\ \delta_3^l \\ \vdots \\ \delta_n^l \end{pmatrix}. \quad (3.31)$$

Empezaremos calculando δ_j^L (el error de la última capa):

$$\delta_j^L = \frac{\partial C_x}{\partial a_j^L} \frac{\partial a_j^L}{\partial z_j^L}$$

pero como $C_x = \frac{1}{2} \|y(x) - a^L\|^2$:

$$-\frac{\partial C_x}{\partial a_j^L} = y(x) - a_j^L \quad \text{y} \quad \frac{\partial a_j^L}{\partial z_j^L} = \frac{\partial \sigma(z_j^L)}{\partial z_j^L} = \sigma'(z_j^L)$$

es decir:

$$\delta_j^L = \frac{\partial C_x}{\partial a_j^L} \sigma'(z_j^L).$$

Ahora calcularemos el error δ_j^l para las capas anteriores:

$$\delta_j^l = \frac{\partial C_x}{\partial z_j^l} = \sum_k \frac{\partial C_x}{\partial z_k^{l+1}} \frac{\partial z_k^{l+1}}{\partial z_j^l}, \quad \text{donde:} \quad \frac{\partial C_x}{\partial z_k^{l+1}} = \delta_k^{l+1}$$

entonces:

$$\delta_j^l = \sum_k \frac{\partial z_k^{l+1}}{\partial z_j^l} \delta_k^{l+1}.$$

Notemos también que:

$$z_k^{l+1} = \sum_j w_{kj}^{l+1} a_j^l + b_k^{l+1}$$

por lo que:

$$\frac{\partial z_k^{l+1}}{\partial z_j^l} = w_{kj}^{l+1} \sigma'(z_j^l)$$

al sustituir tenemos:

$$\delta_j^l = \sum_k w_{kj}^{l+1} \sigma'(z_j^l) \delta_k^{l+1}.$$

Finalmente,

$$\frac{\partial C_x}{\partial w_{jk}^l} = \frac{\partial C_x}{\partial z_j^l} \frac{\partial z_j^l}{\partial w_{jk}^l} = \delta_j^l \frac{\partial}{\partial w_{jk}^l} \left(\sum_k w_{jk}^l a_k^{l-1} + b_j^{l-1} \right)$$

$$\frac{\partial C_x}{\partial w_{jk}^l} = \delta_j^l a_k^{l-1}.$$

De forma similar se obtiene:

$$\frac{\partial C_x}{\partial b_j^l} = \delta_j^l.$$

Resumiendo:

1. $\delta^L = \vec{\nabla} C_x \odot \sigma'(z^L)$.
2. $\delta^l = ((w^{l+1})^T \delta^{l+1}) \odot \sigma'(z^l)$ o bien $\delta_j^l = \sum_k w_{kj}^{l+1} \sigma'(z_j^l) \delta_k^{l+1}$.
3. $\frac{\partial C_x}{\partial b_j^l} = \delta_j^l$.
4. $\frac{\partial C_x}{\partial w_{jk}^l} = a_k^{l-1} \delta_j^l$.

Nota: \odot representa el producto Hadamard (entrada por entrada).

3.11.1. Algoritmo

El algoritmo es como sigue:

1. **Input** x: Propagamos hacia adelante 1 vez y guardamos en memoria las a_s y z_s .
2. Calculamos el error δ^L :

$$\delta^L = \nabla_a C \odot \sigma'(z^L).$$

3. Propagamos hacia atras el error: Calcular para cada capa

$$\delta^l = ((w^{l+1})^T \delta^{l+1}) \odot \sigma'(z^l).$$

4. **Salida:** Finalmente calculamos las componentes del gradiente

$$\frac{\partial C_x}{\partial w_{jk}^l} = a_k^{l-1} \delta_j^l \quad \text{y} \quad \frac{\partial C_x}{\partial b_j^l} = \delta_j^l.$$

3.12. Aprendizaje profundo

Gracias al algoritmo Backpropagation se puede utilizar varias capas en una red neuronal, esto se conoce como Aprendizaje profundo.

El aprendizaje profundo es un tipo de técnica de machine learning que se basa en el cerebro humano. Los algoritmos de aprendizaje profundo analizan los datos con una estructura lógica similar a la que utilizan los humanos. El aprendizaje profundo utiliza sistemas inteligentes, denominados redes neuronales artificiales, para procesar información por capas. Los datos fluyen desde la capa de entrada a través de varias capas de redes neuronales “profundas” ocultas antes de llegar a la capa de salida. Las capas adicionales ocultas permiten un aprendizaje mucho más eficaz que el de los modelos estándar de machine learning [25].

3.13. Optimizadores alternativos

El algoritmo Descenso de gradiente estocástico con Backpropagation da muy buenos resultados y es muy popular, por ejemplo, en el problema de clasificación de dígitos. Sin embargo, hay muchos otros enfoques para optimizar la función de costo y, a veces, esos otros enfoques ofrecen un rendimiento superior al Descenso de gradiente estocástico.

3.13.1. Descenso de Gradiente con momentum

Para comprender la técnica de Descenso de Gradiente con momentum, pensemos en la Figura original del Descenso de Gradiente (3.13), en la que considerábamos una pelota rodando hacia un valle. Anteriormente, observamos que el Descenso de Gradiente es, solo vagamente similar a una pelota que cae al fondo de un valle. La técnica con momentum modifica al Descenso de Gradiente de dos maneras que lo hacen más similar a la analogía física. Primero, introduce una noción de “velocidad” para los parámetros que estamos tratando de optimizar. En segundo lugar, el método con momentum introduce una especie de término de fricción que tiende a reducir gradualmente la velocidad.

Veamos una descripción matemática más precisa. Introducimos las variables de velocidad $v = v_1, v_2, \dots$, uno para cada variable w_j correspondiente. Luego reemplazamos la regla de actualización del Descenso de Gradiente $w \rightarrow w' = w - \eta \nabla C$ por:

$$v \rightarrow v' = \mu v - \eta \nabla C \tag{3.32}$$

$$w \rightarrow w' = w + v'. \tag{3.33}$$

En estas ecuaciones, μ es un hiperparámetro que controla la cantidad de amortiguamiento o fricción en el sistema. Para comprender el significado de las ecuaciones, es útil considerar primero el caso donde $\mu = 1$, que corresponde a la ausencia de fricción. Cuando ese es el caso, la inspección de las ecuaciones muestra que la “fuerza” ∇C ahora modifica la velocidad, v , y la velocidad controla la tasa de cambio de w . Intuitivamente, construimos la velocidad agregando repetidamente términos de gradiente. Pensemos, por ejemplo, en

lo que sucede si nos movemos en línea recta cuesta abajo:

Con cada paso, la velocidad se hace mayor cuesta abajo, por lo que nos movemos más y más rápidamente hacia el fondo del valle. Esto puede permitir que la técnica de momentum funcione mucho más rápido que el Descenso de Gradiente estándar. Un problema es que una vez que llegemos al fondo del valle nos pasaremos. O, si el gradiente cambiara rápidamente, entonces podríamos estar moviéndonos en la dirección equivocada. Esa es la razón del hiperparámetro μ en (3.32). Antes se mencionó que μ controla la cantidad de fricción en el sistema; para ser un poco más preciso, debe pensar en $1 - \mu$ como la cantidad de fricción en el sistema. Cuando $\mu = 1$, como hemos visto, no hay fricción y la velocidad está completamente impulsada por el gradiente ∇C . Por el contrario, cuando $\mu = 0$ hay mucha fricción, la velocidad no puede acumularse, y las Ecuaciones (3.32) y (3.33) se reducen a la ecuación usual para el Descenso de Gradiente, $w \rightarrow w' = w - \eta \nabla C$. En la práctica, usar un valor de μ intermedio entre 0 y 1 puede darnos gran parte del beneficio de poder aumentar la velocidad, pero sin causar excesos.

Lo bueno de la técnica con momentum es que casi no requiere trabajo modificar una implementación de Descenso de Gradiente para incorporar momentum. En la práctica, la técnica con momentum se usa comúnmente y, a menudo, acelera el aprendizaje.

3.13.2. RMSProp (Root Mean Square Propagation)

El gradiente descendente estocástico con momentum usa una única tasa de aprendizaje para todos los parámetros. Otros algoritmos de optimización buscan mejorar el entrenamiento de red usando tasas de aprendizaje que difieren según el parámetro y pueden adaptarse automáticamente a la función de pérdida que se optimiza. RMSProp (propagación del valor cuadrático medio) es uno de estos algoritmos. Mantiene una media móvil de los cuadrados a nivel de elemento de los gradientes de parámetros,

$$g_l = \beta_2 g_{l-1} + (1 - \beta_2) [\nabla E(\theta_l)]^2,$$

β_2 es la tasa de decaimiento de la media móvil. Los valores comunes de la tasa de decaimiento son 0,9, 0,99 y 0,999. Las correspondientes longitudes medias de los gradientes cuadrados son iguales a $1/(1 - \beta_2)$, es decir, 10, 100 y 1000 actualizaciones de parámetros, respectivamente. Los lenguajes de programación nos permiten especificar β_2 . El algoritmo RMSProp utiliza esta media móvil para normalizar las actualizaciones de cada parámetro individualmente,

$$\theta_{l+1} = \theta_l - \frac{\eta \nabla E(\theta_l)}{\sqrt{g_l} + \epsilon},$$

donde la división se realiza a nivel de elemento. Utilizando RMSProp se reducen de forma efectiva las tasas de aprendizaje de los parámetros con gradientes grandes y se incrementan las tasas de aprendizaje de los parámetros con gradientes pequeños. ϵ es una constante pequeña añadida para evitar la división entre cero. También los lenguajes de programación nos permiten especificar ϵ , pero el valor predeterminado normalmente funciona bien [31].

3.13.3. ADAM (Adaptive Moment Estimation)

ADAM es un algoritmo para la técnica de optimización. El método es realmente eficiente cuando se trabaja con un gran problema que involucra una gran cantidad de datos o parámetros. Requiere menos memoria y es eficiente. Intuitivamente, es una combinación del algoritmo Descenso de gradiente con momentum y el algoritmo RMSProp. Mantiene una media móvil a nivel de elemento tanto de los gradientes de parámetros como de sus valores cuadrados,

$$m_l = \beta_1 m_{l-1} + (1 - \beta_1) \nabla E(\theta_l).$$

$$v_l = \beta_2 v_{l-1} + (1 - \beta_2) [\nabla E(\theta_l)]^2.$$

Los lenguajes de programación nos permiten especificar las tasas de decaimiento β_1 y β_2 . Adam usa las medias móviles para actualizar los parámetros de red como:

$$\theta_{l+1} = \theta_l - \frac{\eta m_l}{\sqrt{v_l} + \epsilon}.$$

Si los gradientes a lo largo de varias iteraciones son similares, utilizar una media móvil del gradiente permite a las actualizaciones de parámetros coger impulso en una determinada dimensión. Si los gradientes contienen principalmente ruido, la media móvil del gradiente se reduce, por lo que las actualizaciones de parámetros también se reducen. También los lenguajes de programación nos permiten especificar ϵ . El valor predeterminado normalmente funciona bien, pero para ciertos problemas, un valor tan grande como 1 funciona mejor [31].

3.14. Redes Neuronales Recurrentes (RNN)

Una red neuronal recurrente (recurrent neural network) es una red neuronal artificial que utiliza datos secuenciales o de series temporales. Estos algoritmos de aprendizaje profundo se utilizan a menudo para problemas secuenciales o temporales, como la traducción de un idioma a otro, reconocimiento de voz, análisis del sentimiento y pronósticos.

Al igual que las redes neuronales feedforward, las redes neuronales recurrentes usan datos de entrenamiento para lograr aprender. Se distinguen por su “memoria”, ya que toman información de entradas anteriores para utilizarse en los datos de entrada y en los resultados.

Mientras que las redes neuronales profundas tradicionales asumen que los datos de entrada y salida son independientes entre sí, la salida de las redes neuronales recurrentes depende de elementos previos en la secuencia. Aunque los eventos futuros también serían útiles para determinar el resultado de una secuencia determinada, las redes neuronales recurrentes unidireccionales no pueden tener en cuenta estos eventos en sus predicciones [30].

3.15. Arquitecturas variantes de RNN

Redes neuronales bidireccionales recurrentes (BRNN)

Se trata de una variante de la arquitectura de las RNN. Mientras que los RNN unidireccionales solo pueden extraerse de datos de entrada anteriores para hacer predicciones sobre el estado actual, los RNN bidireccionales extraen datos futuros para mejorar su precisión.

Memoria a largo plazo a corto plazo (LSTM)

Esta es una arquitectura de red neuronal recurrente muy popular, la cual fue introducida por Sepp Hochreiter y Jürgen Schmidhuber como una solución al problema de desvanecimiento o explosión de gradiente. En su artículo científico, ellos abordan el problema de las dependencias a largo plazo. Es decir, si el estado anterior que influye en la predicción actual no es del pasado reciente, es posible que el modelo Red neuronal recurrente no pueda predecir con precisión el estado actual.

Unidades recurrentes cerradas (GRU)

Esta variante de red neuronal recurrente es similar a la variante LSTM, ya que también puede resolver el problema de la memoria a corto plazo de los modelos de redes neuronales recurrentes. En lugar de utilizar una información de regulación de “estado de celda”, utiliza estados ocultos y, en lugar de tres puertas, tiene dos: una puerta de reinicio y una puerta de actualización. Al igual que las puertas dentro de los LSTM, las puertas de reinicio y actualización controlan cuánta y qué información retener [30].

3.16. Red de memoria a largo plazo a corto plazo (LSTM)

Como se mencionó anteriormente las redes LSTM son una forma especializada de arquitectura RNN. En la práctica, las RNN simples tienen una capacidad limitada para aprender dependencias a más largo plazo. Los RNN comúnmente se entrenan a través del algoritmo backpropagation, en la que pueden experimentar un problema de gradiente de desaparición o explosión. Estos problemas hacen que los pesos de la red se vuelvan muy pequeños o muy grandes, lo que limita la efectividad en aplicaciones que requieren que la red aprenda relaciones a largo plazo.

Para superar este problema, las redes LSTM usan puertas adicionales para controlar qué información en la celda oculta se exporta como salida y al siguiente estado oculto. Las puertas adicionales permiten que la red aprenda relaciones a largo plazo en los datos de manera más efectiva. La menor sensibilidad al intervalo de tiempo hace que las redes LSTM sean mejores para analizar datos secuenciales que las RNN simples [35].

Además del estado oculto en los RNN tradicionales, la arquitectura de un bloque LSTM generalmente tiene una celda de memoria, una puerta de entrada, una puerta de salida y una puerta de olvido, como se muestra en la Figura (3.15).

Los pesos y sesgos de la puerta de entrada (input gate) controlan la medida en que un nuevo valor fluye hacia la celda. De manera similar, los pesos y sesgos de la puerta

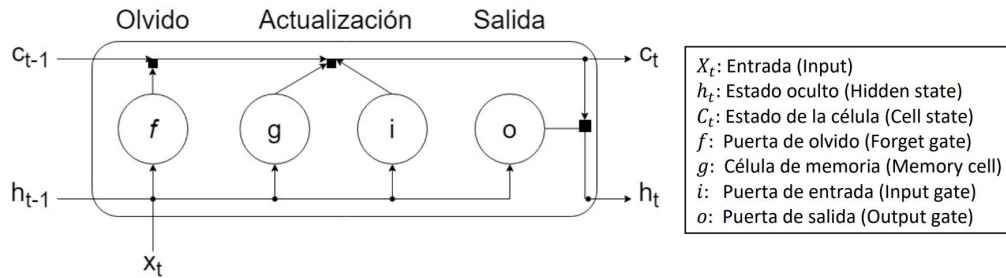


Figura 3.15: Arquitectura de un bloque LSTM (Figura extraída de [35]).

de olvido (forget gate) y la puerta de salida (output gate) controlan la medida en que un valor permanece en la celda y la medida en que el valor de la celda se usa para calcular la activación de salida del bloque LSTM, respectivamente [35].

3.16.1. Arquitectura de red de LSTM

Los componentes principales de una red de LSTM son una capa de entrada de secuencias y una capa de LSTM. Una capa de entrada de secuencias introduce datos secuenciales o de series de tiempo en la red. Una capa de LSTM aprende dependencias a largo plazo entre las unidades de tiempo de los datos secuenciales.

En la Figura 3.16 se muestra un diagrama que ilustra la arquitectura de una red de LSTM sencilla para regresión. La red comienza con una capa de entrada de secuencias seguida de una capa de LSTM. La red termina con una capa totalmente conectada y una capa de regresión de salida [36].

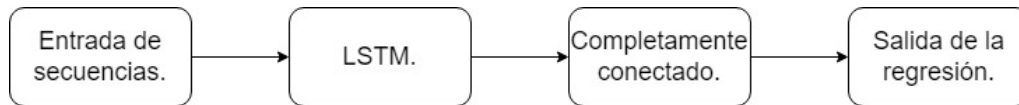


Figura 3.16: Diagrama de una red LSTM (Figura extraída de [36]).

Capítulo 4

Caso de estudio

En este capítulo se expone el principal aporte de este trabajo, que es la aplicación de la teoría previamente expuesta la cual consistente en un análisis del comportamiento de la variable meteorológica radiación solar en la zona norte del municipio de Puebla, considerando un conjunto de datos la cual fue recolectada en la estación meteorológica ubicada en la Universidad Tecnológica de Puebla.

La limpieza y el procesamiento de los datos de la variable radiación solar se puede ver en el Apéndice B. Antes de iniciar el análisis, se describirá brevemente algunas características de la zona de estudio (municipio de Puebla), incluyendo algunas influencias meteorológicas como “El Niño”, “La Niña” y la canícula.

4.1. Zona de estudio

El municipio de Puebla se encuentra localizado en el valle de Puebla–Tlaxcala en la zona central del estado e inmediatamente al sur del límite con el estado de Tlaxcala, sus coordenadas y altitud son entre los paralelos $18^{\circ} 50'$ y $19^{\circ} 14'$ de latitud norte; los meridianos $98^{\circ} 01'$ y $98^{\circ} 18'$ de longitud oeste; altitud entre 1 980 y 4 500 *m*. Su extensión territorial es de 534.33 *km*² que lo convierten en el quinto municipio más grande del estado de Puebla.

Limita al noreste con el municipio de Tepatlaxco de Hidalgo, al este con el municipio de Amozoc y con el municipio de Cuautinchán, al sureste con el municipio de Tzicatlacoyan, al sur con el municipio de Huehuetlán el Grande y con el municipio de Teopantlán, al suroeste con el municipio de Ocoyucan, al oeste con el municipio de San Andrés Cholula y con el municipio de San Pedro Cholula y al noroeste con el municipio de Cuautlancingo; al norte limita con el estado de Tlaxcala, en particular con el municipio de Papalotla de Xicohtécatl, el municipio de Tenancingo, el municipio de San Pablo del Monte y el municipio de Teolocholco.

El clima es templado subhúmedo con lluvias en verano, de mayor humedad (48.69%), templado subhúmedo con lluvias en verano, de humedad media (47.30%) y semifrío

subhúmedo con lluvias en verano (4.01 %). El rango de precipitación es de 400 – 900 milímetros [12].



Figura 4.1: Localización geográfica del municipio de Puebla, Puebla (Figura extraída del Instituto Nacional de Estadística y Geografía (INEGI)).

Canícula

La canícula es un evento climático que sucede a mediados del mes de julio y termina a finales de agosto, dura aproximadamente 40 días y se caracteriza por ser un periodo de aumento de la temperatura debido a un calentamiento excesivo del aire, cielos despejados y baja precipitación.

Las lluvias disminuyen porque los vientos alisios, al soplar con fuerza desde el Este, impiden la formación de nubes sobre el océano, con lo cual se reducen las lluvias en la parte del territorio continental. Se presenta en algunas regiones del país entre ellos en el estado de Puebla.

La canícula es generada por los llamados vientos "alisios", que soplan con fuerza en el Golfo de México, impidiendo el desarrollo de nubes convectivas en el océano y, como resultado, la formación de lluvia en el territorio continental de México. La palabra Canícula se deriva de la palabra "canes", que significa "perros", y su alusión al fenómeno de calor abrasivo tiene una base astronómica, ya que se refiere a la constelación del Can Mayor y su estrella Sirio "La Abrasadora", que apareció por primera vez en el horizonte coincidía con el fenómeno de calor excesivo [13], [14].

El niño

El Niño es un fenómeno climático que describe el calentamiento inusual de las aguas

superficiales en el Océano Pacífico oriental. El Niño tiene un impacto en las temperaturas del océano, la velocidad y la fuerza de las corrientes oceánicas, la salud de las pesquerías costeras y el clima local desde Australia hasta América del Sur y más allá. Los eventos de El Niño ocurren irregularmente en intervalos de dos a siete años. Sin embargo, El Niño no es un ciclo regular, ni predecible en el sentido en que lo son las mareas oceánicas. El Niño fue reconocido por los pescadores de la costa de Perú como la aparición de aguas inusualmente cálidas.

El Niño también produce cambios generalizados y a veces severos en el clima. La convección sobre las aguas superficiales más cálidas provoca un aumento de las precipitaciones. Por lo general, alcanza su punto máximo entre los meses de noviembre y enero, aunque las acumulaciones se pueden detectar con meses de anticipación y sus efectos pueden tardar meses en extenderse por todo el mundo [15].

La niña

La Niña es un fenómeno climático que describe el enfriamiento de las aguas superficiales del océano. Se caracteriza por una presión de aire más baja de lo normal sobre el Pacífico occidental. Estas zonas de baja presión contribuyen al aumento de las precipitaciones.

Se considera que La Niña es la contraparte de El Niño, que se caracteriza por temperaturas oceánicas inusualmente cálidas en la región ecuatorial del Océano Pacífico. Los eventos de La Niña a veces siguen a los eventos de El Niño, que ocurren en intervalos irregulares de alrededor de dos a siete años. Los efectos locales sobre el clima causados por La Niña son generalmente opuestos a los asociados con El Niño.

La Niña es causada por una acumulación de aguas más frías de lo normal en el Pacífico tropical, el área del Océano Pacífico entre el Trópico de Cáncer y el Trópico de Capricornio.

Tanto El Niño como La Niña afectan los patrones de lluvia, la presión atmosférica y la circulación atmosférica global. La circulación atmosférica es el movimiento de aire a gran escala que, junto con las corrientes oceánicas, distribuye energía térmica sobre la superficie de la Tierra. Estos cambios son las principales fuentes de variabilidad del clima en muchas áreas del mundo [16].

En la Figura 4.2 se muestra una imagen extraída del NOAA (Oficina Nacional de Administración Oceánica y Atmosférica) que muestra anomalías (desviaciones de lo normal) de temperaturas en la superficie del mar. Los colores azules representan aguas más frías de lo normal, mientras que las rojas representan aguas más calientes de lo normal. Cuando está más caliente de lo normal en esa región del Pacífico oriental, se le conoce como el fenómeno de El Niño, mientras que cuando está más frío de lo normal, se le conoce como La Niña. Este ciclo de temperaturas de más calientes a más frías es normal en las corrientes del océano del hemisferio sur, y por eso se le conoce como la Oscilación del Sur-El Niño.

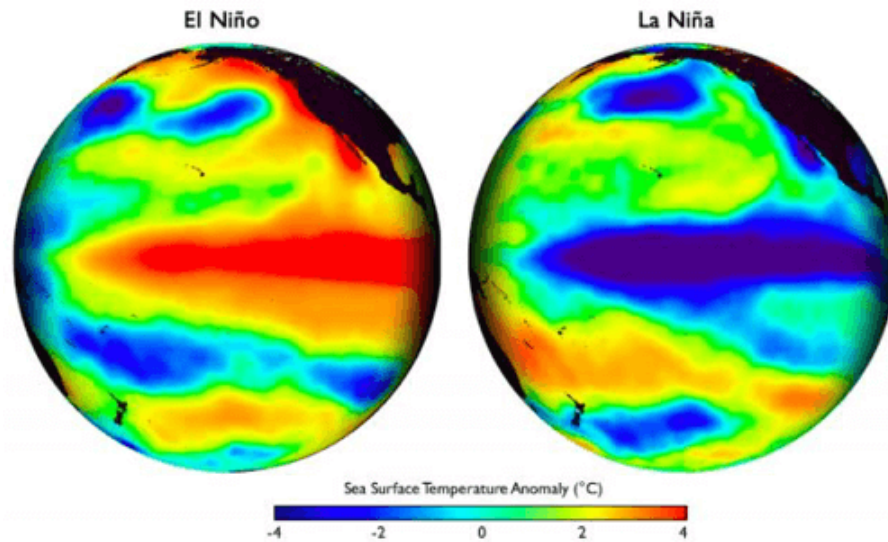


Figura 4.2: Imagen de NOAA.

4.2. Análisis del conjunto de datos de la radiación solar del municipio de Puebla

4.2.1. Análisis usando la metodología Box-Jenkins

En la Figura 4.3 se puede observar la gráfica de la serie de tiempo de la radiación del municipio de Puebla promediada semanalmente, podemos observar que la serie de tiempo presenta cierta estacionalidad y no presenta una tendencia muy visible, pero analizaremos esto para saber si la serie es estacionaria, claramente podemos observar que la serie de tiempo no oscila alrededor del cero.

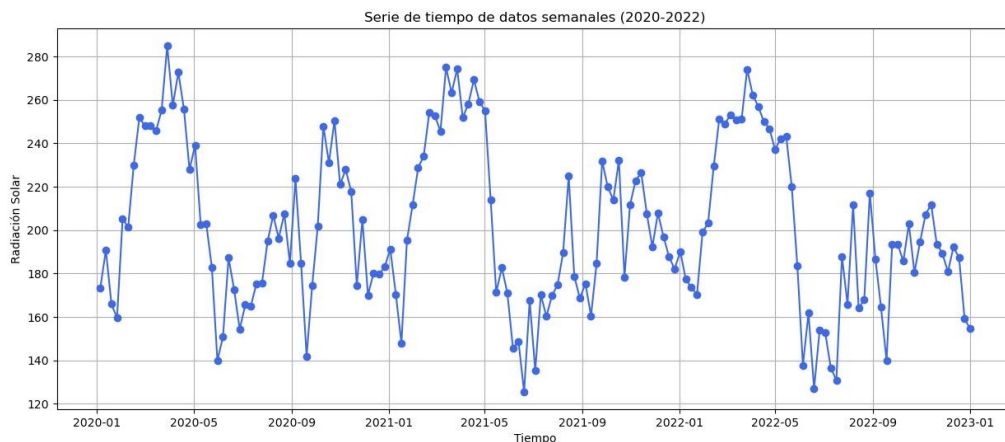


Figura 4.3: Gráfica de la serie de tiempo de la radiación solar.

4.2 Análisis del conjunto de datos de la radiación solar del municipio de Puebla

Primeramente, se analizará la estacionalidad. Con la ayuda de la función *seasonal_decompose* de la paquetería *statsmodels* del Software Python podemos hacer una descomposición de la serie de tiempo la cual se puede observar en la Figura 4.4, esta descomposición nos divide la serie de tiempo en tres partes:

- Tendencia.
- Estacional.
- Residual.

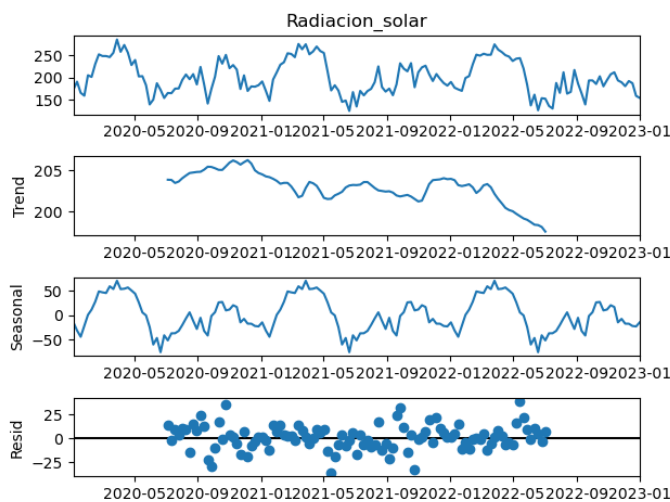


Figura 4.4: Descomposición aditiva de la serie de tiempo de la radiación solar (De autoría propia).

En la Figura 4.5 se observa con mayor claridad la parte estacional de la descomposición, esta Figura nos ayuda a observar que hay presencia de estacionalidad.

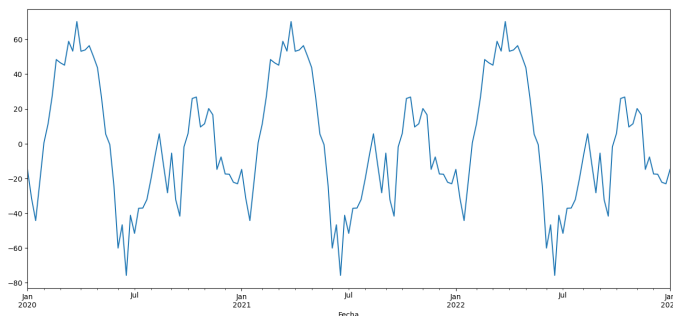


Figura 4.5: Parte estacional de la descomposición aditiva (De autoría propia).

Para reafirmar la presencia de estacionalidad se realizó un análisis de cajas y bigotes, ver Figura 4.6. Se graficó agrupando las mismas semanas año por año, de ahí que en el diagrama de cajas tenga 53 cajas ya que el año 2020 cuenta con 53 semanas mientras los años 2021 y 2022 cuentan con 52 semanas. Este análisis se realizó para verificar la estacionalidad, y concluimos que si es estacional.

Además, se analizaron las partes del diagrama basándonos en los reportes del Clima en México los cuales son resúmenes anuales relacionados con el clima en la escala nacional [37]. En estos reportes se mencionaron los fenómenos recurrentes tales como “La niña”, “el niño”, sequías, huracanes, ondas de calor, etcétera.

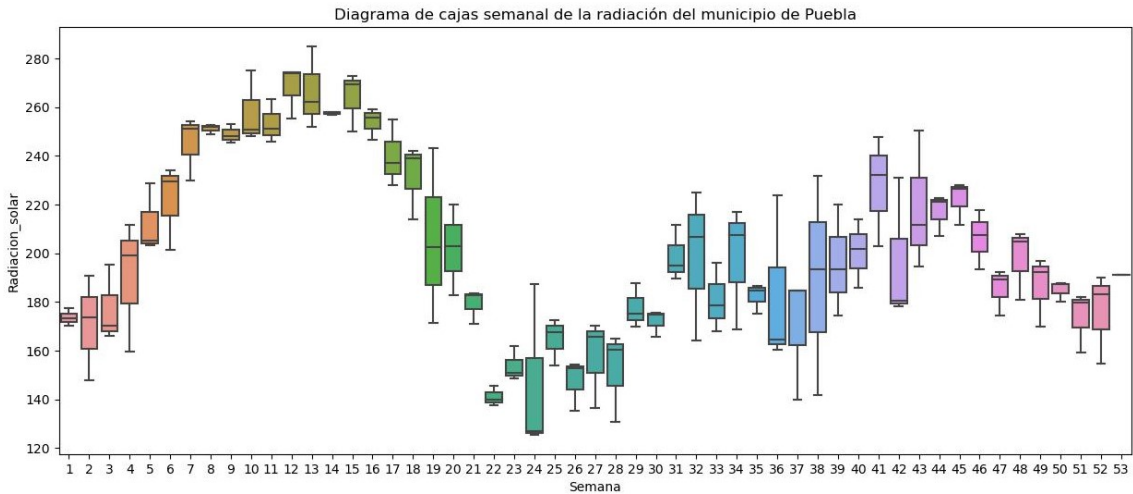


Figura 4.6: Grafica de Cajas y bigotes (De autoría propia).

Análisis

Mayor valor máximo

En la Figura 4.6 se puede observar que el mayor valor máximo se presenta en la semana 13 la cual corresponde aproximadamente a la última semana de marzo (2020-2022) donde este valor se encuentra por arriba de los 280 ($watts/m^2$)² esto se podría deber a la ausencia de lluvias durante esta temporada a excepción del año 2022, en el que se registraron lluvias en estas fechas. Además, que el año 2020 fue el año más cálido, mientras que para 2021 y 2022 fueron años más frescos con un umbral de 5 a 15°.

Ahora, enfocándose en la caja 10, la cual corresponde a la primera semana de marzo, aproximadamente, se observa que para estas semanas se alcanza un valor máximo mayor a 260 ($watts/m^2$), sin sobrepasar los valores de la caja 13, cabe recalcar que ambas cajas tienen valores similares pues tienen apenas unas semanas de diferencia.

Para la caja 12 y 15 se tienen valores similares, siendo mayores a 260 como en la caja 10 pero no llegan a rebasarlos. La caja 12 corresponde del 21 al 27 de marzo, aproximadamente; mientras que la caja 15 corresponde del 11 al 17 de abril, aproximadamente. A diferencia de las anteriores cajas el mes de abril cuenta con mayores precipitaciones,

especialmente en el año 2020, y ésta es una de las razones por las que llega a ser la que tenga el mayor valor máximo.

Por último, se tienen las cajas 11 y 16 las cuales pertenecen a la segunda semana de marzo y tercera semana de abril, donde hubo mayor precipitación en el año 2020.

Menor valor mínimo

El menor valor mínimo se presenta en la caja 24, la cual pertenece a la segunda semana de junio, aproximadamente, donde se registraron lluvias arriba de lo normal, especialmente en el año 2021, además que, junio de este año fue el que tuvo menor temperatura media, es decir, fue uno de los más fríos en el estado de Puebla.

A esta caja le sigue la caja número 28, en cuestión de valores mínimos, esta caja corresponde a la segunda semana de julio, aproximadamente. La diferencia de esta caja a la 24 podría deberse a que durante el mes de julio se registraron menos lluvias en el año 2021 y 2022, mientras que en el año 2020 se registraron casi la misma cantidad de lluvias. A continuación, le sigue la caja 26 y 27, las cuales representan la penúltima y última semana de junio. Cabe mencionar que como estas cajas se encuentran entre la caja 24 y 28, mencionadas anteriormente, las condiciones climáticas son similares.

La caja que le sigue sería la 37 y 38, las cuales corresponden a la segunda y tercera semana de septiembre, esto podría deberse a que en el año 2022 hubo un frente frío durante la tercera semana de septiembre la cual trajo consigo lluvias y bajas temperaturas.

Cajas largas

La caja más larga es la 38 la cual pertenece a la tercera semana de septiembre en la cual se tuvo un frente frío en el año 2022, además de ser uno de los meses que tiene más lluvias registradas, sin embargo, también mantuvo temperaturas máximas arriba de los 24°C.

Le caja que le sigue es la 19, la cual representa a la segunda semana del mes de mayo, el largo de esta caja podría deberse a que en el año 2021 hubo un frente frío el cual provocó lluvias intensas en Puebla además que hubo temperaturas máximas superiores a los 28°C y hasta los 30°C.

Caja pequeña

La caja más pequeña es la correspondiente a la semana 14 la cual abarca la primera semana de abril, lo cual podría deberse a que la temperatura promedio del mes de abril para los años 2020,2021,2022 es similar, siendo ésta 22.4°C, 20.9°C y 20.7°C, respectivamente. Además, se registró un frente frío durante esta semana en el año 2022, sin embargo, éste frente frío fue corto y no trajo consigo temperaturas mínimas extremas.

Cajas ligeramente pequeñas

Las cajas que le siguen a la más pequeña son: 1,8,9,21,22,23,26,27,30,35,44,45,50; es importante destacar que la mayoría de las cajas mencionadas anteriormente no tienen bigotes, o bien, sus bigotes son pequeños.

Caja 1: Representa a la primera semana del mes de enero.

Caja 8: Representa a la última semana del mes de febrero.

Caja 9: Representa a la primera semana del mes de marzo.

Caja 21: Representa a la última semana del mes de mayo.

Caja 22: Representa a la primera semana del mes de junio.

- Caja 23: Representa a la segunda semana del mes de junio.
Caja 26: Representa a los últimos días de junio y primeros de julio.
Caja 27: Representa a la primera semana del mes de julio.
Caja 30: Representa a la última semana del mes de julio.
Caja 35: Representa a los últimos días de agosto y primeros de septiembre.
Caja 44: Representa a la primera semana del mes de noviembre.
Caja 45: Representa a la segunda semana del mes de noviembre.
Caja 50: Representa a la segunda semana del mes de diciembre.

Mayor mediana

La mayor mediana está en la caja 15, la cual representa a la segunda semana del mes de abril. Es probable que aquí se encuentra la mayor mediana puesto que no ocurren frentes fríos durante esta semana, también es el mes que mayor temperatura máxima tiene, siendo la única excepción el año 2022 donde la mayor temperatura máxima fue durante mayo añadiendo que es el mes donde menos lluvias se registraron.

Seguida de esta caja se encuentra la correspondiente a la semana 13, la cual representa a los últimos días del mes de marzo y primeros del mes de abril diferencia entre ambas cajas probablemente sea que la temperatura registrada para el mes de marzo sea menor que la de abril.

Menor mediana

La menor mediana es la de la caja 22 y le sigue la caja 23, la cual representa a la primera y segunda semana del mes de junio esto podría deberse a que regularmente en estas temporadas salían los frentes fríos, además que junio se encuentra entre los 3 meses con mayor cantidad de lluvias.

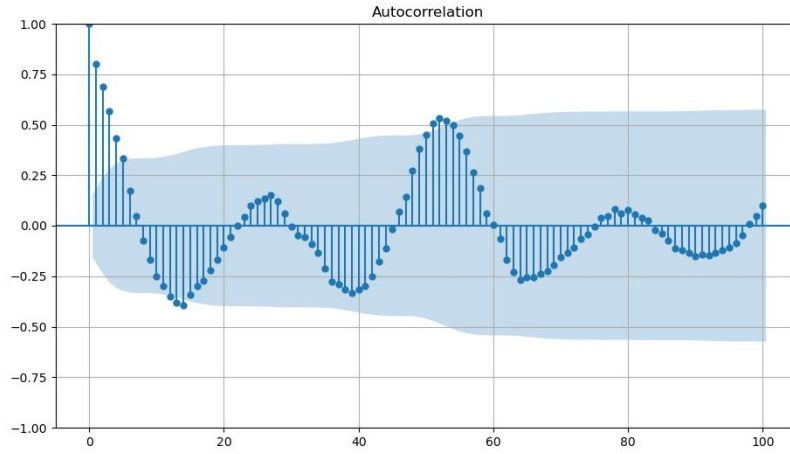
Valores atípicos

Es importante mencionar que no hay valores atípicos, es probable que esto sea porque no hubo mucha variación en las precipitaciones de cada año, y tampoco en las temperaturas máximas y temperaturas promedio de comparándolas año con año.

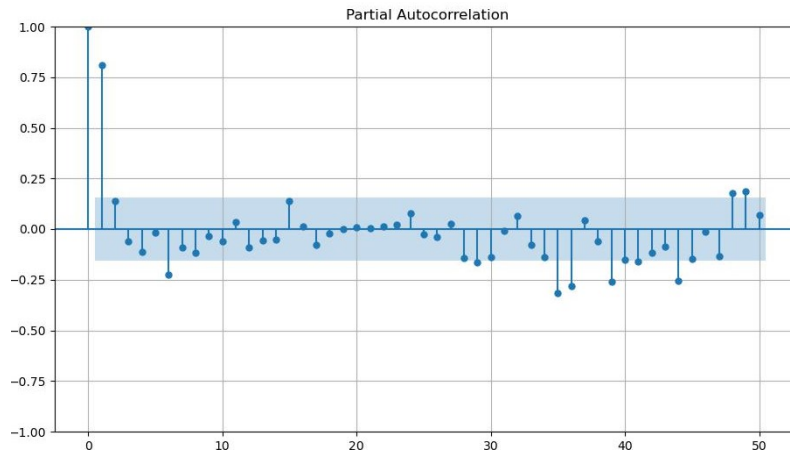
Por otro lado, para analizar la estacionariedad de una serie de tiempo, primero se verifica si la serie es al menos débilmente estacionaria. Se graficó la *FAC* y el *FACP* de la serie de tiempo, consulte las Figuras 4.7(a) y 4.7(b). Podemos observar que *FAC* se corta con rapidez y se extingue en forma de onda sinusoidal amortiguada, para la *FACP* se puede observar que se extingue desde el primer retraso y vuelve a ser significativo en los retrasos 6, 35, 36, 44, 48, 49.

Las autocorrelaciones de la serie de tiempo sugieren que es una serie de tiempo estacionaria, por lo que para estar seguros de esto se decidió ocupar la prueba de estacionariedad de Dickey–Fuller, nótese que la serie de tiempo en la Figura 4.3 esta fluctuando alrededor de una media distinta de cero entonces se utilizará la versión 2 de la prueba de Dickey–Fuller. Esta versión se aplica cuando la serie tiene término constante pero no tendencia.

La Tabla 4.1 muestra los resultados de la prueba de Dickey–Fuller.



(a) *FAC*



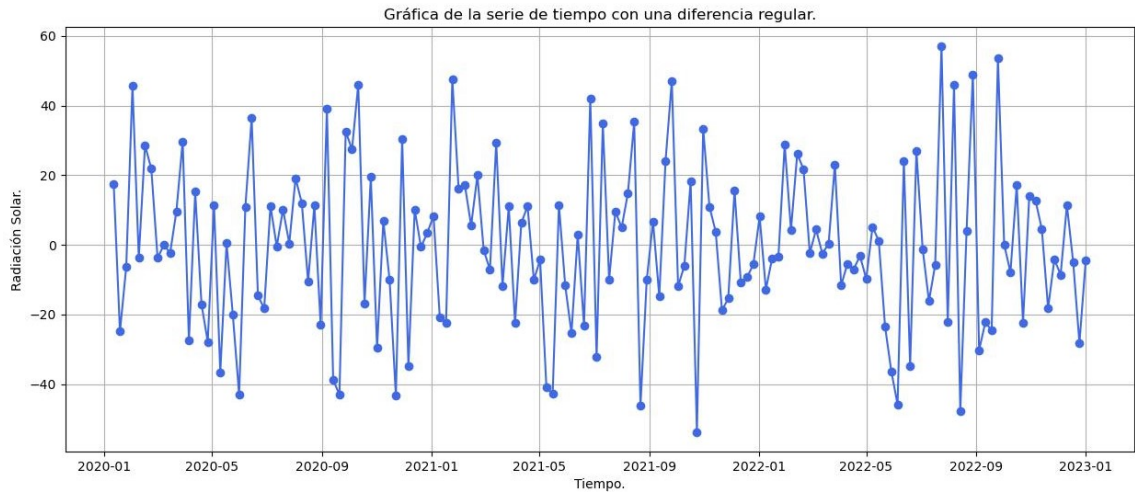
(b) *FACP*

Figura 4.7: *FAC* Y *FACP* de la serie de tiempo.

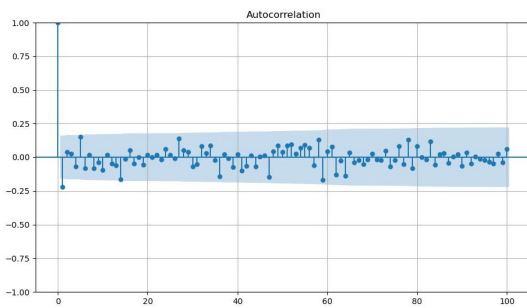
| H_0 : La serie es no estacionaria vs H_a : La serie es estacionaria | | | | |
|---|-----------------------|----------------------|----------------------|-----------------------|
| | Estadístico de prueba | Valor crítico al 1 % | Valor crítico al 5 % | Valor crítico al 10 % |
| Dickey-Fuller con intercepto | -4.11777 | -3.47441 | -2.88088 | -2.57708 |

Tabla 4.1: Versión 2 de la prueba de Dickey–Fuller para la serie de tiempo (De autoría propia).

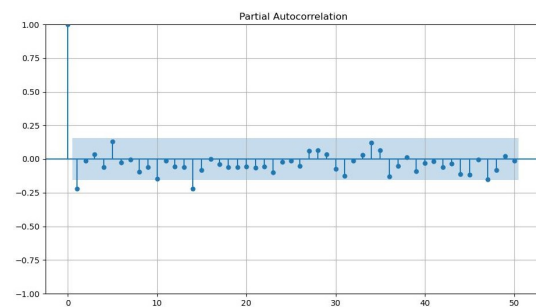
4.2 Análisis del conjunto de datos de la radiación solar del municipio de Puebla



(a) Trayectoria.



(b) *FAC*



(c) *FACP*

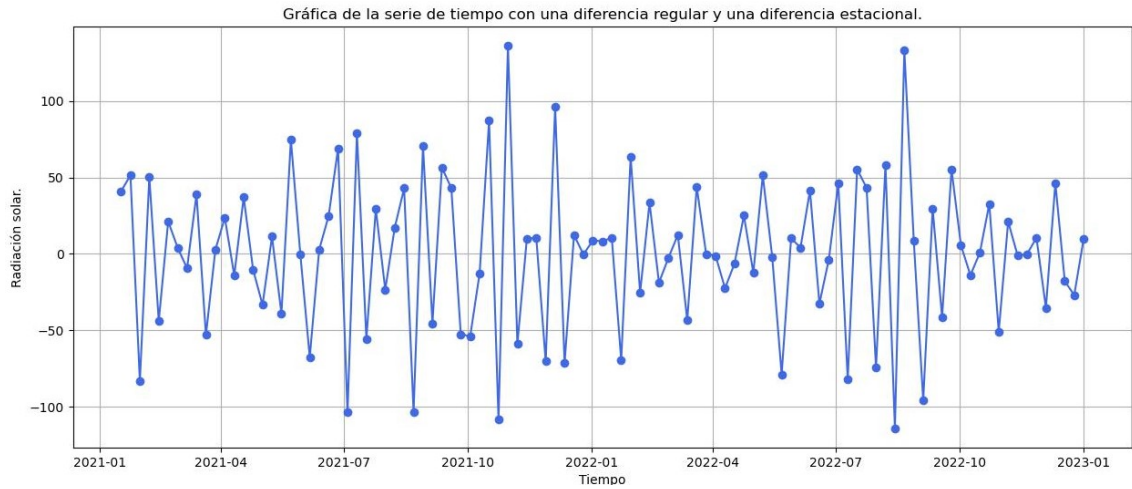
Figura 4.8: Trayectoria, *FAC* y *FACP* de la serie de tiempo de la radiación solar con una diferencia regular (De autoría propia).

Realizando el contraste de hipótesis, la hipótesis nula es rechazada en cualquier nivel de significancia. Por lo tanto, la serie de tiempo es estacionaria y presenta estacionalidad.

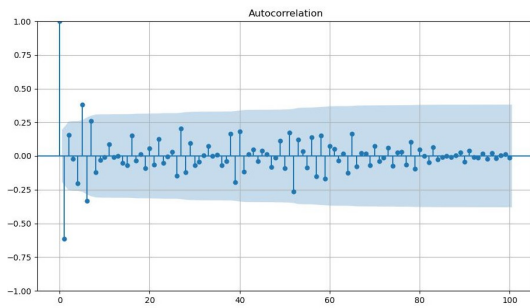
Sin embargo, en busca del mejor y más simple modelo, se decidió aplicar una diferencia regular a la serie de tiempo original y una diferencia estacional a la serie con la diferencia regular, obsérvese las Figuras 4.8(a) y 4.9(a) y podemos notar que ambas fluctúan alrededor de una media cero.

Ahora verifiquemos que la serie de tiempo con la diferencia regular y la serie de tiempo con la diferencia regular y estacional son estacionarias, con la ayuda de la prueba de Dickey–Fuller, pero ahora utilizaremos la versión 1 ya que las series de tiempo oscilan alrededor de una media igual a cero. En las Tablas 4.2 y 4.3 podemos observar los resultados de la versión 1 de la prueba Dickey–Fuller en ambos casos a hipótesis nula es rechazada en cualquier nivel de significancia, por lo tanto ambas series son estacionarias.

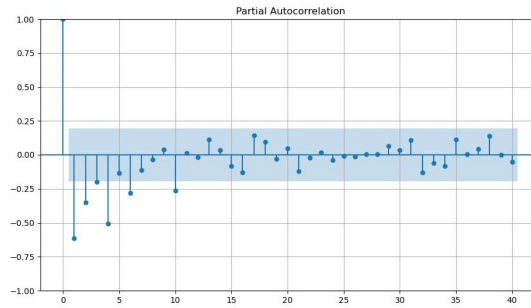
4.2 Análisis del conjunto de datos de la radiación solar del municipio de Puebla



(a) Trayectoria.



(b) *FAC*



(c) *FACP*

Figura 4.9: Trayectoria, *FAC* y *FACP* de la serie de tiempo de la radiación solar diferenciada regularmente y estacionalmente (De autoría propia).

La *FAC* y la *FACP* de la serie con una diferencia regular se pueden observar en las Figuras 4.8(b), 4.8(c) mientras que la *FAC* y la *FACP* de la serie de tiempo con una diferenciada regular y una estacional, se pueden observar en las Figuras 4.9(b) y 4.9(c).

La *FAC* y la *FACP* de las Figuras 4.8(b) y 4.8(c) ayudan a definir los órdenes de p y q para describir la parte no estacional de un modelo de series de tiempo. Observemos que la *FAC* de la Figura 4.8(b) sobresale la espiga del primer retraso y después se extingue. Entonces q puede tomar el valor de 0 o 1.

Mientras que en la *FACP*, hay 2 espigas prominentes, una en el primer retraso donde en esta se corta y la otra espiga que sobresale es en el 14 retraso. Entonces el valor de p puede ser 0 o 1. Por tanto, se propone el modelo *ARIMA*(1, 1, 1) para la parte no estacional del modelo.

Por otra parte, se analizan la *FAC* y la *FACP* del gráfico de la Figura 4.9. Podemos ver que la *FAC* tiene una espiga prominente en el primer retraso, y se corta, después

| H_0 : La serie es no estacionaria vs H_a : La serie es estacionaria | | | | |
|---|-----------------------|----------------------|----------------------|-----------------------|
| | Estadístico de prueba | Valor crítico al 1 % | Valor crítico al 5 % | Valor crítico al 10 % |
| Dickey-Fuller con intercepto | -15.53969 | -2.58031 | -1.94286 | -1.61521 |

Tabla 4.2: Versión 1 de la prueba de Dickey–Fuller para la serie con una diferencia regular (De autoría propia).

| H_0 : La serie es no estacionaria vs H_a : La serie es estacionaria | | | | |
|---|-----------------------|----------------------|----------------------|-----------------------|
| | Estadístico de prueba | Valor crítico al 1 % | Valor crítico al 5 % | Valor crítico al 10 % |
| Dickey-Fuller con intercepto | -6.41473 | -2.59075 | -1.94432 | -1.61419 |

Tabla 4.3: Versión 1 de la prueba de Dickey–Fuller para la serie con una diferencia regular y una diferencia estacional (De autoría propia).

otros 2 picos son prominentes, pero son muy pequeños. Entonces el valor de Q puede ser igual a 0 o 1.

Mientras la *FACP*, hay 4 espigas prominentes en los primeros seis retrasos y se corta después del retraso 4, luego las espigas 6 y 10 son prominentes, pero son muy pequeños, como la espiga 3. Por lo que el valor de P puede ser 0, 1, 2, 3 o 4. Por tanto para la parte estacional del modelo, se propone con $s = 52$ el modelo $ARIMA(1, 1, 0)_{52}$.

Por lo tanto, el modelo propuesto es $SARIMA(1, 1, 1)(1, 1, 0)_{52}$.

Se plantearon otros 3 modelos con la ayuda de la paquetería *pdmarmima* del software Python. Los resultados se pueden observar en la Tabla 4.4.

| Modelos propuestos. | | |
|---------------------------------|----------|----------|
| Modelo. | AIC | BIC |
| $ARIMA(1, 1, 1)$ | 1424.088 | 1433.238 |
| $ARIMA(2, 0, 2)$ | 1416.567 | 1434.905 |
| $SARIMA(0, 1, 1)(2, 0, 0)_{52}$ | 1415.580 | 1427.779 |
| $SARIMA(1, 0, 0)(1, 1, 0)_{52}$ | 956.414 | 964.376 |
| $SARIMA(1, 1, 1)(1, 1, 0)_{52}$ | 952.448 | 963.026 |

Tabla 4.4: Diferentes modelos propuestos (De autoría propia).

Tras observar los resultados obtenidos según el criterio de información bayesiano y el criterio Akaike, el modelo escogido es $SARIMA(1, 1, 1)(1, 1, 0)_{52}$, ya que toma el menor valor según los criterios.

Por lo tanto, el modelo propuesto queda:

$$\Phi_1(B^{52})\phi_1(B)\nabla_{52}^1\nabla^1 X_t = \theta_1(B)\epsilon_t. \quad (4.1)$$

donde $\nabla_{52}^1 = (1 - B^{52})$ y $\nabla^1 = (1 - B)$.

Desarrollando 4.1 resulta lo siguiente:

$$\begin{aligned} (1 - \theta_1 B)\epsilon_t &= (1 - \Phi_1 B^{52})(1 - \phi_1 B)(1 - B^{52})(1 - B)X_t \\ \epsilon_t - \theta_1 \epsilon_{t-1} &= (1 - \Phi_1 B^{52})(1 - \phi_1 B)(1 - B^{52})(X_t - X_{t-1}) \\ \epsilon_t - \theta_1 \epsilon_{t-1} &= (1 - \Phi_1 B^{52})(1 - \phi_1 B)(X_t - X_{t-52} - X_{t-1} + X_{t-53}) \\ \epsilon_t - \theta_1 \epsilon_{t-1} &= (1 - \Phi_1 B^{52})(X_t - \phi_1 X_{t-1} - X_{t-52} + \phi_1 X_{t-53} - X_{t-1} + \phi_1 X_{t-2} + X_{t-53} \\ &\quad - \phi_1 X_{t-54}) \\ \epsilon_t - \theta_1 \epsilon_{t-1} &= X_t - \phi_1 X_{t-1} - X_{t-52} + \phi_1 X_{t-53} - X_{t-1} + \phi_1 X_{t-2} + X_{t-53} - \phi_1 X_{t-54} \\ &\quad - \Phi_1 X_{t-52} + \Phi_1 \phi_1 X_{t-53} + \Phi_1 X_{t-104} - \Phi_1 \phi_1 X_{t-105} + \Phi_1 X_{t-53} - \Phi_1 \phi_1 X_{t-54} \\ &\quad - \Phi_1 X_{t-105} + \Phi_1 \phi_1 X_{t-106}. \end{aligned}$$

Despejando X_t tenemos:

$$\begin{aligned} X_t &= X_{t-1} + \phi_1 X_{t-1} - \phi_1 X_{t-2} + X_{t-52} + \Phi_1 X_{t-52} - X_{t-53} - \phi_1 X_{t-53} - \Phi_1 X_{t-53} \\ &\quad - \Phi_1 \phi_1 X_{t-53} + \phi_1 X_{t-54} + \Phi_1 \phi_1 X_{t-54} - \Phi_1 X_{t-104} + \Phi_1 X_{t-105} + \Phi_1 \phi_1 X_{t-105} \\ &\quad - \Phi_1 \phi_1 X_{t-106} - \theta_1 \epsilon_{t-1} + \epsilon_t. \end{aligned}$$

Con la ayuda del software Python, se utiliza el método de máxima verosimilitud para la estimación de los parámetros, y los resultados se muestran en la Tabla 4.5.

| Estimación de parámetros. | | |
|---------------------------|---------------------|-----------------|
| Parámetro. | Parámetro estimado. | Error estándar. |
| $AR(1)$ | 0.2471 | 0.102 |
| $MA(1)$ | -0.9572 | 0.053 |
| $SAR(1)$ | -0.5199 | 0.107 |

Tabla 4.5: Valores obtenidos del modelo (De autoría propia).

Una vez estimados los parámetros, el modelo se escribe de la siguiente manera:

$$X_t = (1.2471)X_{t-1} - (0.2471)X_{t-2} + (0.4801)X_{t-52} - (0.598733)X_{t-53} + (0.118633)X_{t-54} + (0.5199)X_{t-104} - (0.64836729)X_{t-105} + (0.12846729)X_{t-106} + (0.9572)\epsilon_{t-1} + \epsilon_t.$$

En la Figura 4.10 se muestra la serie de tiempo original sobre la ajustada.

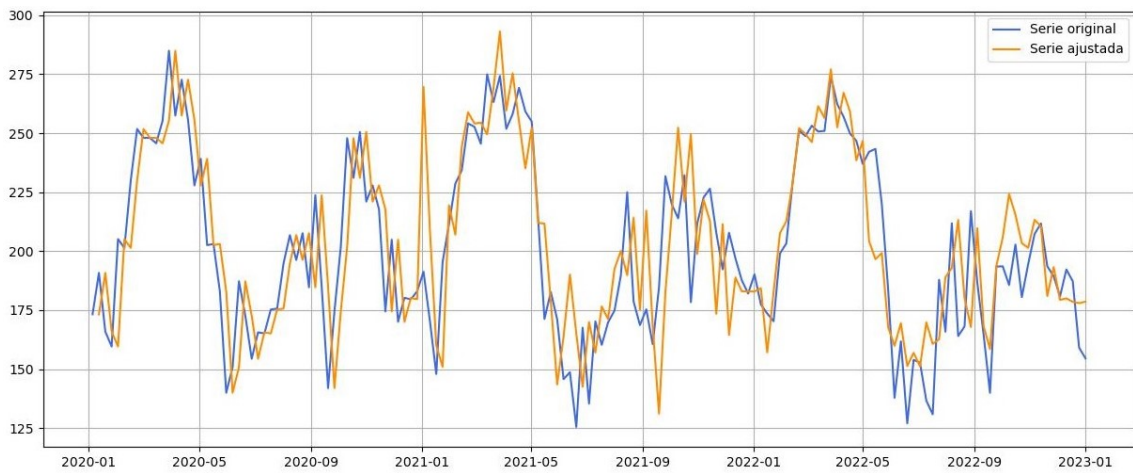


Figura 4.10: Serie original y ajustada (De autoría propia).

Para la realización del pronóstico se utilizaron las 5 primeras semanas del año 2023, esto con la finalidad de comprobar la eficiencia del modelo propuesto mediante el error, a continuación, se comparan los datos reales de las primeras 5 semanas del año 2023 contra los valores pronosticados, obsérvese Tabla 4.6. Y la Figura 4.11 se puede observar el pronóstico.

| Pronósticos obtenidos 2023 | | | | |
|----------------------------|-----------|-------------|---------------------------------------|---|
| Periodo | Dato real | Box-Jenkins | Error absoluto $e_a = V_r - V_p $ | Error relativo $e_r = \frac{e_a}{V_r}$ |
| 08-01-2023 | 195.08 | 157.41 | 37.67 | 0.1931 |
| 15-01-2023 | 187.60 | 148.78 | 38.82 | 0.2069 |
| 22-01-2023 | 187.72 | 173.11 | 14.61 | 0.0778 |
| 29-01-2023 | 197.75 | 195.58 | 2.17 | 0.0109 |
| 05-02-2023 | 211.30 | 206.63 | 4.67 | 0.0221 |

Tabla 4.6: Resultados con la metodología Box-Jenkins (De autoría propia).

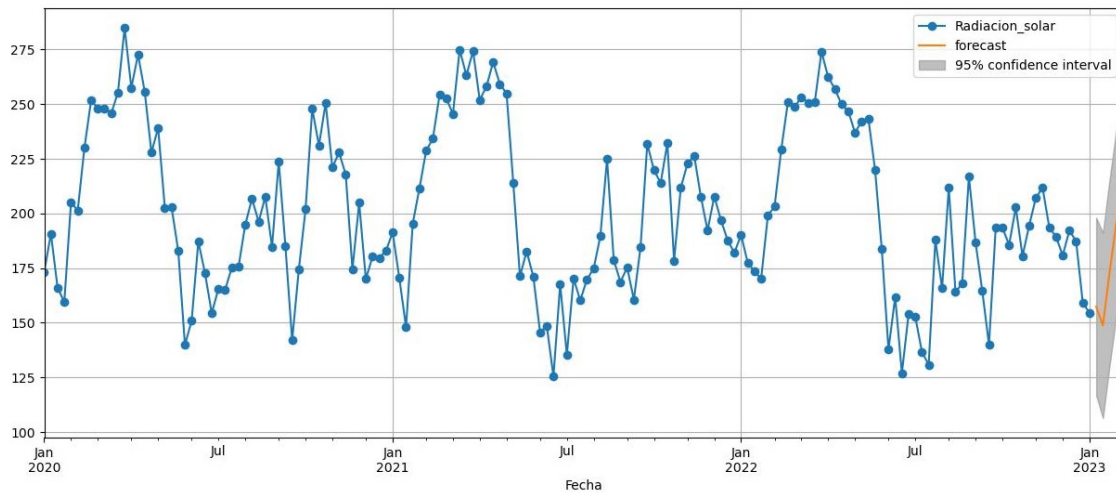


Figura 4.11: Pronóstico de 5 semanas con la metodología Box-Jenkins (De autoría propia).

Ahora validemos el modelo ya que es importante para demostrar la idoneidad de este. Analizar los residuales son una buena estrategia de diagnóstico para la validación del modelo. Para analizar los residuales, se utilizó de la función *plot_diagnostics* del software Python.

Observemos que en la primera grafica de la Figura 4.12 muestra los residuales estandarizados, de los cuales se puede observar que están distribuidos aleatoriamente alrededor de cero sin ninguna tendencia alguna, entonces estos los podemos considerar independientes.

En la segunda grafica se observa la densidad teórica de una distribución normal estándar en color naranja y en verde se observa la densidad de los residuales generado por el método Kernel, estas dos curvas son muy parecidas y es posible observar presencia de simetría por lo que es posible considerar el supuesto de normalidad. En la tercera grafica se observa un gráfico Q-Q gracias a este podemos observar que los residuales se distribuyen en línea recta, y la linealidad de los puntos indica que los residuales se distribuyen normalmente.

En la cuarta y última imagen, se muestra la *FAC* de los residuales la cual se asemeja a la *FAC* de un proceso de ruido blanco.

Gracias al análisis que se muestra en la Figura 4.12, se puede observar que los residuales muestran que el componente estocástico de estos es muy parecido a un ruido blanco entonces se puede concluir que se apoya la suposición de que los residuos tienden a distribuirse normalmente.

Por consiguiente, se puede concluir que el mejor modelo que representa adecuadamente los datos es el modelo $SARIMA(1, 1, 1)(1, 1, 0)_{52}$.

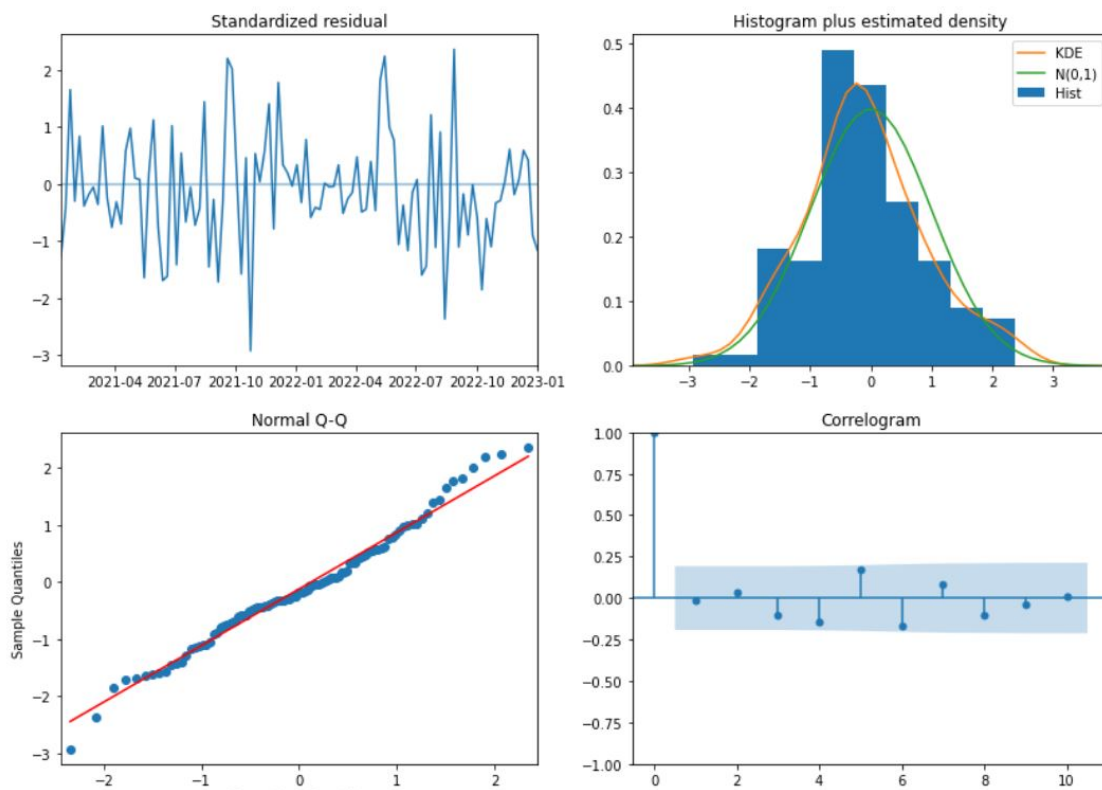


Figura 4.12: Gráfico de residuales del modelo $SARIMA(1, 1, 1)(0, 1, 1)_{52}$ (De autoría propia).

4.2.2. Análisis usando Redes Neuronales Artificiales

Ahora en esta sección se hará un análisis de los datos de la radiación solar, utilizando la metodología de Redes Neuronales Artificiales con el objetivo de crear una RNN que pronostique datos futuros de la radiación solar a base de entrenamiento.

Se usará el mismo conjunto de datos con los que se realizó el análisis con la metodología Box-Jenkins, recordando que estos datos están promediados semanalmente y comprenden desde la primera semana del año 2020 hasta la quinta semana del año 2023, el comportamiento de este conjunto de datos se muestra en la Figura 4.13. El conjunto de datos se dividirá en dos, en el subconjunto de datos de entrenamiento y en el subconjunto de datos de prueba.

Teniendo el conjunto de datos listo con ayuda del Software Python se procede a crear una red neuronal que sea capaz de pronosticar dicha variable a través del tiempo. Existen diversos métodos para hacer pronósticos. Anteriormente se utilizó el modelo *SARIMA*, el cual arrojó buenos resultados.

En esta sección se aplicará un modelo Machine Learning: una Red Neuronal LSTM para hacer el pronóstico. Crear esta red es completamente computacional con la ayuda de la paquetería *TensorFlow* de Python. Se usará una arquitectura sencilla de Red Neuronal LSTM (Long Short-Term Memory), con varias neuronas y con una función de activación ReLU.

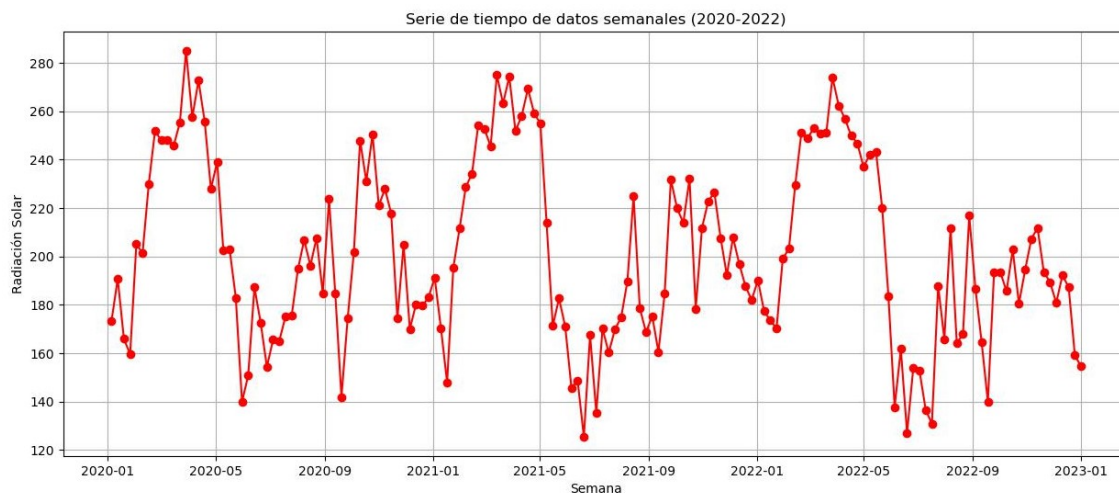


Figura 4.13: Gráfica del conjunto de datos de la radiación solar (De autoría propia).

4.2.3. Preparación de los datos

Primeramente, se convierte a la columna Fecha en formato Datetime especificando que nuestros datos están ordenados semanalmente después a esta misma columna la convertimos en nuestro índice, todo esto es porque vamos a trabajar con una serie de tiempo.

Ahora dividimos el conjunto de datos en dos, en el subconjunto de datos de entrenamiento y en el subconjunto de datos de prueba, los datos de entrenamiento abarcarán el 80% de los datos y el 20% restante será el conjunto de datos de prueba. El conjunto de datos de entrenamiento abarcará desde la primera semana del año 2020 hasta la última semana del mes de mayo del año 2022. Mientras el conjunto de datos de prueba empieza desde la primera semana del mes de junio del año 2022 y terminará hasta la última semana del año 2022.

Por otro lado, como se mencionó en la parte teórica para ayudar a que nuestra Red neuronal artificial aprenda se puede utilizar una técnica de normalización al conjunto de datos. Por lo que a los datos de la radiación solar se le realizó una normalización, utilizando la técnica de escalamiento a un rango. El rango fue entre 0 y 1, esto se hizo con la función *MinMaxScaler* de la paquetería *scikit-learn* del software Python.

Una vez escalados los datos, es importante que los datos de series temporales deben transformarse en una estructura de muestras con componentes de entrada y salida antes de que puedan usarse para ajustarse a un modelo de aprendizaje supervisado.

La biblioteca *Keras* del software Python proporciona la función *TimeseriesGenerator* para transformar automáticamente datos de series temporales tanto univariantes como multivariantes en muestras, listas para entrenar modelos de aprendizaje profundo.

Ahora se aplica la función mencionada al conjunto de datos escalados, proporcionando los siguientes argumentos:

- **Datos:** Datos escalados.
- **n_input:** 52 entradas.
- **n_output:** 1 salida.

4.2.4. Análisis de los datos para la creación de la Red

En la Figura 4.14 podemos observar el comportamiento de nuestros datos de entrenamiento y datos de prueba. Mientras que en las Tablas 4.7 y 4.8 se muestran cómo están formados estos conjuntos en forma de tabla, en donde podemos observar que del lado izquierdo se encuentran las fechas que funciona como índice a los valores de la variable a estudiar y del lado derecho los valores obtenidos de la radiación solar.

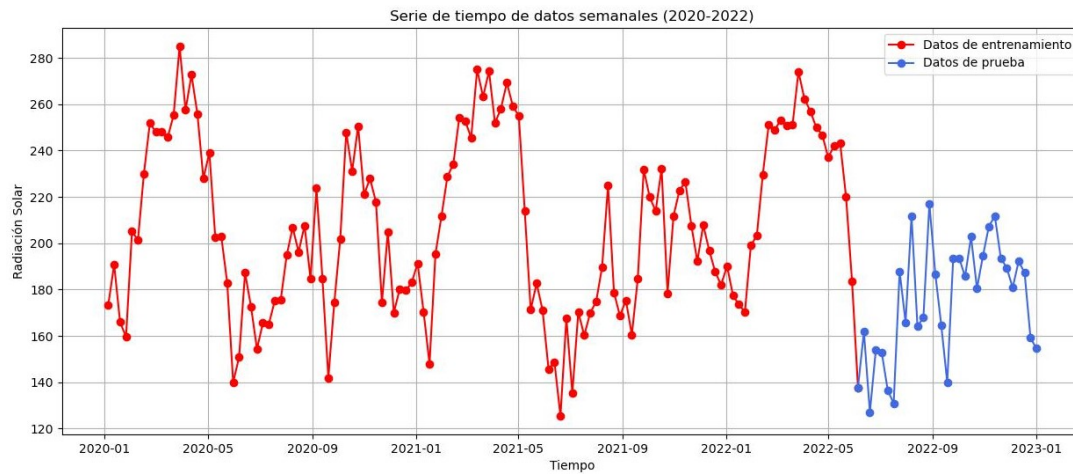


Figura 4.14: Gráfica de la serie de tiempo de la radiación solar (De autoría propia).

| Fecha | Radiación solar |
|------------|-----------------|
| 2020-01-05 | 173.237500 |
| 2020-01-12 | 190.791667 |
| 2020-01-19 | 165.910714 |
| 2020-01-26 | 159.541667 |
| 2020-02-02 | 205.110119 |
| ... | ... |
| 2022-05-01 | 237.005952 |
| 2022-05-08 | 242.136905 |
| 2022-05-15 | 243.336310 |
| 2022-05-22 | 219.937500 |
| 2022-05-29 | 183.648810 |

Tabla 4.7: Conjunto de datos de entrenamiento (De autoría propia).

En la Tabla 4.9 se puede observar un resumen estadístico de la información de los datos de la radiación solar, se observan 157 datos con una media total de 201.327778, una desviación estándar de 37.701579, como dato mínimo se tiene 125.458333 y como máximo dato se tiene 284.958333.

| Fecha | Radiación solar |
|------------|-----------------|
| 2022-06-05 | 137.794643 |
| 2022-06-12 | 161.761905 |
| 2022-06-19 | 126.997024 |
| 2022-06-26 | 153.904762 |
| 2022-07-03 | 152.732143 |
| ... | ... |
| 2022-12-04 | 180.809524 |
| 2022-12-11 | 192.220238 |
| 2022-12-18 | 187.279762 |
| 2022-12-25 | 159.083333 |
| 2023-01-01 | 154.550595 |

Tabla 4.8: Conjunto de datos de prueba (De autoría propia).

| | Radiación solar |
|-------|-----------------|
| count | 157 |
| mean | 201.327778 |
| std | 37.701579 |
| min | 125.458333 |
| 25 % | 173.237500 |
| 50 % | 194.511905 |
| 75 % | 231.083333 |
| max | 284.958333 |

Tabla 4.9: Resumen estadístico descriptivo (De autoría propia).

4.2.5. Creación de la Red Neuronal Artificial

Nuestro objetivo es crear una red neuronal artificial recurrente LSTM, ya que vamos a trabajar con una serie de tiempo.

Para conseguir el mejor modelo de red neuronal, se hicieron varios entrenamientos cambiando parámetros, número de neuronas, número capas, etc. Se decidió por el mejor modelo el cual tenga menor error relativo entre los datos de prueba con los datos pronosticados.

En la Figura 4.15 se muestra distintos resultados de cuatro redes neuronales, seleccionando la que tiene menor error relativo, es decir se ajusta mejor a los datos.

4.2 Análisis del conjunto de datos de la radiación solar del municipio de Puebla

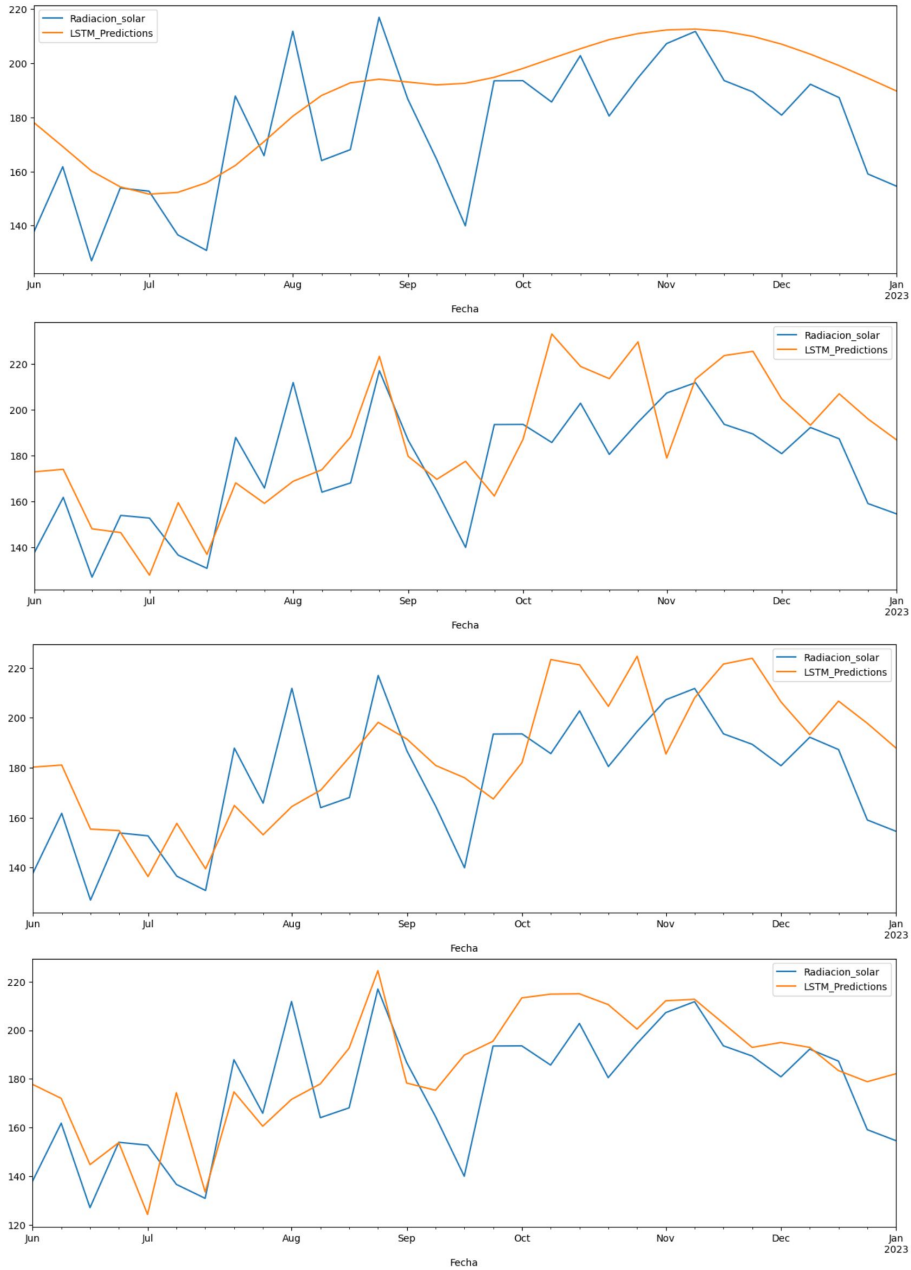


Figura 4.15: Resultados de distintas redes neuronales (De autoría propia).

4.2.6. Descripción de la Red Neuronal Artificial

En la RNN elegida se utilizó un modelo secuencial, con una capa de entrada, una capa intermedia y una capa de salida.

Por lo que la arquitectura de la red neuronal recurrente queda de la siguiente manera:

- Modelo secuencial.
- La capa de entrada es de tipo LSTM con 200 neuronas.
- 1 capa intermedia u oculta LSTM con 200 neuronas.
- La salida es una capa densa con una única neurona.
- Como función de activación se utiliza la función ReLU.
- Por optimizador se utiliza el Método Adam.
- Por función de costo se utilizó el Error cuadrático medio (MSE).

En la Figura 4.16 se muestra la estructura de la RNN implementada para el pronóstico de la variable meteorológica radiación solar del municipio de Puebla.

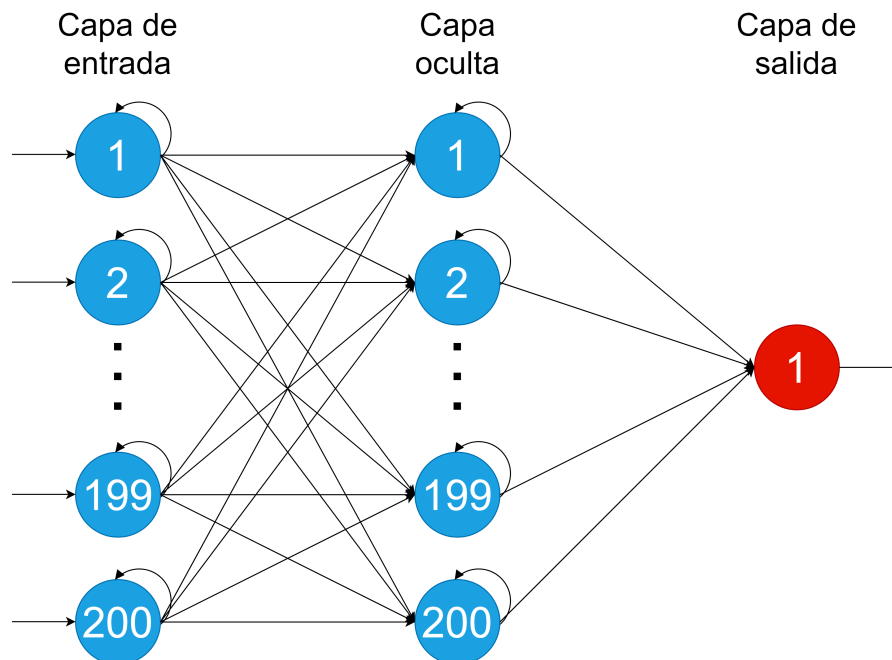


Figura 4.16: Estructura de la red neuronal artificial implementada (De autoría propia).

4.2.7. Pronóstico con la Red Neuronal Artificial

Para el entrenamiento de la red neuronal se ocuparon 100 épocas, en la Figura 4.17 se puede observar el comportamiento de la función de costo a lo largo de las 100 épocas, observemos que la función disminuye al paso de éstas.

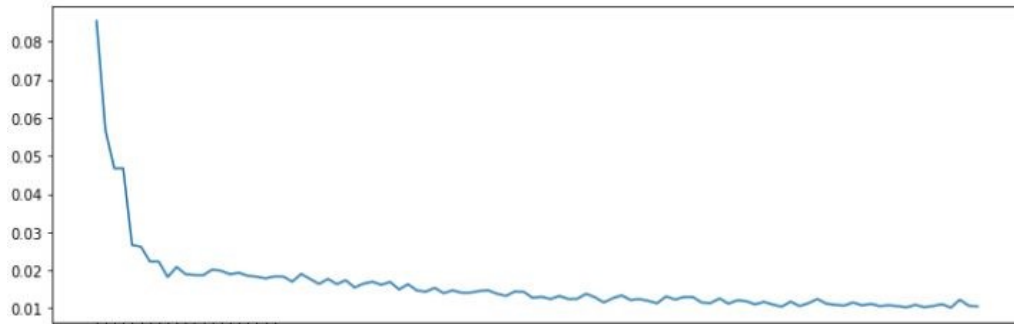


Figura 4.17: Función de costo (De autoría propia).

Finalmente se comparan los pronósticos arrojados por la Red Neuronal Artificial con los datos reales de la radiación solar del municipio de Puebla, para así poder observar la eficacia de nuestra Red Neuronal, mediante los errores totales y relativos. Recordando que estos datos son las primeras 5 semanas del año 2023.

La comparación y los errores se presentan en la Tabla 4.10.

| Pronósticos obtenidos 2023 | | | | |
|----------------------------|------------|-------------|---|---|
| Periodo. | Dato real. | Pronóstico. | Error absoluto $e_a = V_r - V_p $. | Error relativo $e_r = \frac{e_a}{V_r}$. |
| 08-01-2023 | 195.08 | 171.04 | 24.04 | 0.1232 |
| 15-01-2023 | 187.60 | 177.33 | 10.27 | 0.0547 |
| 22-01-2023 | 187.72 | 184.65 | 3.07 | 0.0164 |
| 29-01-2023 | 197.75 | 198.25 | 0.5 | 0.0025 |
| 05-02-2023 | 211.30 | 216.73 | 5.43 | 0.0257 |

Tabla 4.10: Comparación entre los datos reales y los datos pronosticados de la RNN, y los errores (De autoría propia).

Finalmente, en la Figura 4.18 se muestra la comparación de los datos reales con los datos pronosticados de la RNN en forma gráfica.

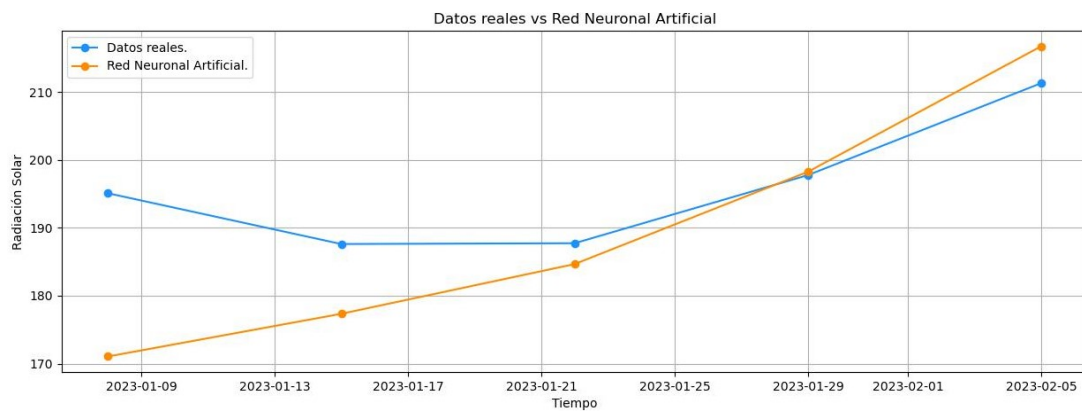


Figura 4.18: Comparación entre los datos reales de la variable meteorológica radiación solar del municipio de Puebla y el pronóstico de la RNN (De autoría propia).

Capítulo 5

Análisis y conclusiones

Este trabajo se inició dando una explicación de los conceptos meteorológicos donde se mencionó la definición de las variables atmosféricas, gracias a estas, es posible el estudio del tiempo meteorológico y del clima, después nos centramos en la variable a estudiar la cual es la radiación solar. Con esta variable se profundizó los conceptos para el entendimiento de esta, por ejemplo, de donde proviene esta variable, los instrumentos de medición, las unidades con la que se mide, etc.

Después se expuso la teoría necesaria para explicar y aplicar la teoría de series de tiempo, tratando de analizar la radiación solar del municipio de Puebla siguiendo la metodología de Box-Jenkins y Redes Neuronales Artificiales, para así compáralas y tratar de concluir que metodología es más conveniente para este tipo de análisis.

Al analizar los datos de la radiación solar nos damos cuenta de que presenta cierta estacionalidad, después con un análisis de cajas y bigotes confirmamos esto, así al aplicar la metodología Box-Jenkins sabemos que podemos ocupar un modelo $SARIMA(p, d, q)(P, D, Q)_S$.

Al aplicar la metodología Box-Jenkins a los datos de la radiación solar, en un principio se realiza una prediferenciación para disminuir la variabilidad, pero notamos que no hay gran diferencia con los datos originales, por lo cual se decide ocupar los datos sin prediferenciación.

Después se lleva a cabo un análisis de la serie original con lo cual se concluye que la serie es estacionaria pero como se busca obtener el mejor modelo y el más sencillo se decide realizar una diferencia regular y una diferencia estacional.

Se observa mediante el análisis de las FAC y $FACP$ de la serie diferenciada regularmente, que los valores que q puede tomar son 0 o 1, mientras que los valores que p puede tomar son también 0 o 1. En el análisis de la FAC y $FACP$ de la serie de tiempo con una diferencia regular y una diferencia estacional, se observa que los valores que Q puede tomar son 0 o 1 mientras que los valores que P puede tomar son 0, 1, 2, 3 o 4.

El modelo propuesto por el análisis que se hizo fue el modelo $SARIMA(1, 1, 1)(1, 1, 0)_{52}$.

Después con ayuda del criterio de Akaike y el criterio de información Bayesiano se comparan varios modelos y se elige al mejor modelo. Por lo que el modelo propuesto es:

$$SARIMA(1, 1, 1)(1, 1, 0)_{52}.$$

Este modelo es idóneo para la predicción de datos de la radiación solar de acuerdo con el análisis de los residuos y la prueba de Ljung-Box.

Una vez obtenido el mejor modelo para pronosticar la radiación solar del municipio de Puebla por la metodología Box-Jenkins se procede a la creación de un Red Neuronal Artificial para el pronóstico de la radiación solar del municipio de Puebla y una vez obtenidos los resultados se comparan los pronósticos de las dos metodologías utilizadas para tratar de determinar cuál modelo pronostica de la mejor manera.

Las redes neuronales artificiales se han aplicado con éxito a varios tipos de problemas, como detección de dígitos, clasificación de imágenes, predicción, etc. En los problemas de predicciones se ha mostrado que las redes neuronales son capaces de modelar y pronosticar series de tiempo como lineales y no lineales con alta precisión, captura cualquier tipo de relación entre los datos y no necesitan tener ninguna distribución en los datos de origen, estos pueden o no tener correlación. El objetivo de las redes neuronales es formar un sistema inteligente que pueda realizar con éxito tareas complejas. Para el análisis de este caso de estudio se decidió crear una red neuronal recurrente que fuera capaz de pronosticar los datos de la radiación del municipio de Puebla a través de su información histórica y el entrenamiento para poder tener el mejor pronóstico para las semanas futuras.

Para realizar la Red neurona artificial recurrente se empezó con un pequeño análisis del conjunto de datos para observar cómo está conformado, su comportamiento, etc. Después se prosiguió con la normalización de los datos con la técnica de escalamiento en un rango, esto para que nuestra Red neuronal aprenda fácilmente.

Una vez teniendo los datos normalizados se divide el conjunto de datos en dos sub-conjuntos:

- Conjunto de datos de entrenamiento.
- Conjunto de datos de prueba.

Después se sigue con la creación de la Red Neuronal Recurrente, para conseguir el mejor modelo, se hicieron varios entrenamientos cambiando parámetros, número de neuronas, número capas, etc. obteniendo que el modelo con mejor adaptación tiene la siguiente arquitectura:

- Modelo secuencial.
- La capa de entrada es de tipo LSTM con 200 neuronas.

- 1 capa oculta LSTM con 200 neuronas.
- La salida es una capa densa con una única neurona.
- Como función de activación se utiliza la función ReLU.
- Por optimizador se utiliza el Método Adam.
- Por función de costo se utilizó el error cuadrático medio (mse).

Este modelo se entrenó durante 100 épocas.

5.1. Análisis y Resultados

En este trabajo se diseñaron dos modelos con distintas metodologías para tratar de pronosticar valores futuros de la radiación solar del municipio de Puebla, en ambas metodologías se obtuvieron dos posibles pronósticos para las primeras 5 semanas del año 2023. A continuación, se presenta la información obtenida y su comparación con los datos reales.

En la Tabla 5.1 se muestran los pronósticos obtenidos mientras que en la Figura 5.1 se muestra el comportamiento de ambos pronósticos en forma gráfica.

| Pronósticos obtenidos 2023 | | | |
|----------------------------|---------------|------------------|-------------|
| Periodo. | Red Neuronal. | Metodología B-J. | Diferencia. |
| 08-01-2023 | 171.04 | 157.41 | 13.63 |
| 15-01-2023 | 177.33 | 148.78 | 28.55 |
| 22-01-2023 | 184.65 | 173.11 | 11.54 |
| 29-01-2023 | 198.25 | 195.58 | 2.67 |
| 05-02-2023 | 216.73 | 206.63 | 10.1 |

Tabla 5.1: Pronósticos de las primeras 5 semanas del año 2023 (De autoría propia).

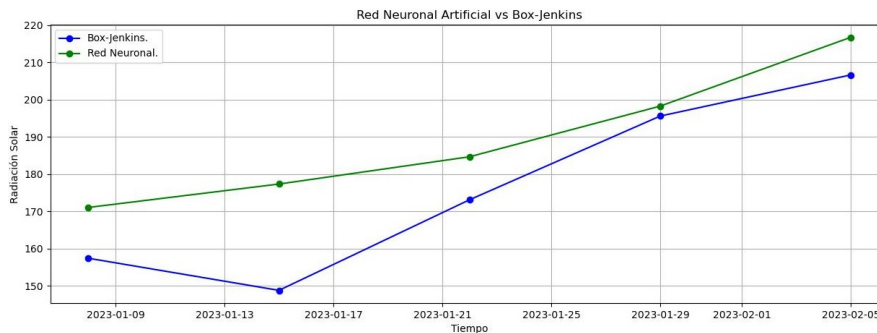


Figura 5.1: Comparación gráfica entre los pronósticos obtenidos (De autoría propia).

Se observa que ambas gráficas tienen una tendencia parecida con un punto máximo en la quinta semana del año 2023 mostrando un poco más definido la Red Neuronal, mientras el punto mínimo es diferente.

Al parecer ambos pronósticos cuentan con un comportamiento semejante con cierta variabilidad no tan alta en la mayoría de los puntos excepto por los datos de la segunda semana del año 2023 la cual hay mayor presencia de diferencia.

Ahora se procede a realizar la comparación de ambos pronósticos con los datos reales obtenidos en la página web de SINAICA. En la Tabla 5.2 se muestran los pronósticos obtenidos y los datos reales de la radiación solar del municipio de Puebla.

| Pronósticos obtenidos 2023 | | | |
|----------------------------|------------|---------------|--------------------------|
| Periodo. | Dato real. | Red Neuronal. | Metodología Box-Jenkins. |
| 08-01-2023 | 195.08 | 171.04 | 157.41 |
| 15-01-2023 | 187.60 | 177.33 | 148.78 |
| 22-01-2023 | 187.72 | 184.65 | 173.11 |
| 29-01-2023 | 197.75 | 198.25 | 195.58 |
| 05-02-2023 | 211.30 | 216.73 | 206.63 |

Tabla 5.2: Datos reales obtenidos en la página web de SINAICA y pronósticos obtenidos por la metodología Box-Jenkins y RNN para las primeras 5 semanas del año 2023 (De autoría propia).

En la Figura 5.2 se muestra el comportamiento gráfico de los resultados obtenidos en ambas metodologías comparados con el comportamiento de los datos reales obtenidos por SINAICA. Se puede apreciar que los comportamientos de las dos metodologías son parecidos con los datos reales, acercándose más a la realidad el pronóstico de la Red Neuronal Artificial.

En la Figura 5.3 se muestran diagramas de densidad generadas por el método Kernel y se puede observar normalidad en las tres figuras, sin embargo se puede decir que la metodología que se asemeja más a la densidad de los datos reales es la Red Neuronal Artificial, por lo que podemos tener como primera conclusión que para el caso estudio del pronóstico de la radiación solar del municipio de Puebla el mejor pronosticador es el modelo de Red Neuronal Artificial ya que presenta un comportamiento más semejante a la realidad.

Seguidamente en la Tabla 5.3 se comparan los pronósticos obtenidos por ambas metodologías y se muestran los errores relativos para cada pronóstico.

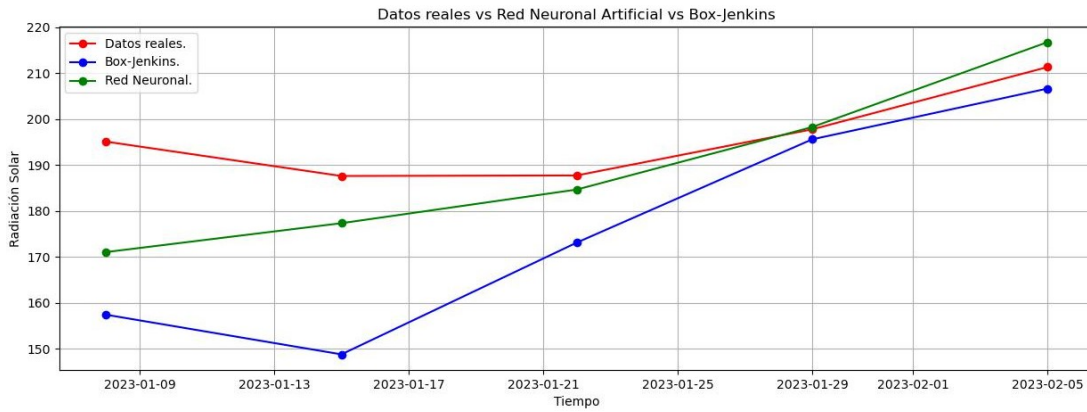


Figura 5.2: Comparación gráfica entre los datos reales y los pronósticos obtenidos por las dos metodologías (De autoría propia).

| Pronósticos obtenidos 2023 | | | | | |
|----------------------------|-----------|-------------|--------------|----------------------------|--------------------|
| Periodo | Dato real | Box-Jenkins | Red Neuronal | Error relativo Box-Jenkins | Error relativo RNA |
| 08-01-2023 | 195.08 | 157.41 | 171.04 | 0.193104 | 0.12321 |
| 15-01-2023 | 187.60 | 148.78 | 177.33 | 0.206932 | 0.05477 |
| 22-01-2023 | 187.72 | 173.11 | 184.65 | 0.077848 | 0.01635 |
| 29-01-2023 | 197.75 | 195.58 | 198.25 | 0.011009 | 0.0025 |
| 05-02-2023 | 211.30 | 206.63 | 216.73 | 0.022111 | 0.0257 |

Tabla 5.3: Comparación de los pronósticos obtenidos para las primeras 5 semanas del año 2023 (De autoría propia).

En la Tabla 5.3 se puede observar que la metodología de Redes neuronales artificiales tiene menor error relativo en casi todos los pronósticos excepto por la quinta semana del año 2023, aunque esta es muy poca diferencia al error relativo del método de Box-Jenkins. El comportamiento del modelo de Redes neuronales parece tener un comportamiento más estable, mientras que con la metodología Box-Jenkins los resultados obtenidos están un poco bajos, pero en la mayoría de los pronósticos conservan un comportamiento similar.

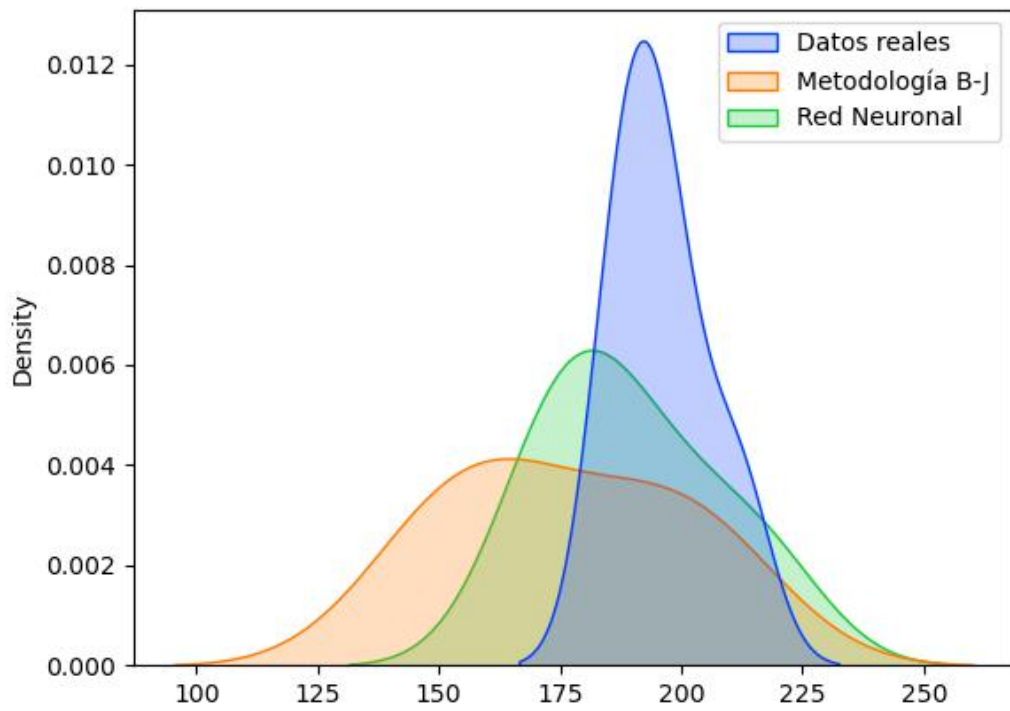


Figura 5.3: Diagramas de densidad generadas por el método Kernel (De autoría propia).

5.2. Conclusiones

En este trabajo se han presentado los resultados obtenidos con los datos de la radiación solar del municipio de Puebla con la metodología Box-Jenkins y Redes neuronales artificiales para la predicción a corto plazo de la radiación solar. El desempeño del modelo predictivo RNN desarrollado en este trabajo se comparó con el modelo Box-Jenkins *SARIMA*, que ha sido ampliamente utilizado para el pronóstico de series de tiempo. Nuestros hallazgos revelaron que tanto el modelo *SARIMA* como el modelo de Red Neuronal recurrente pueden lograr un buen pronóstico en la aplicación a problemas de la vida real y, por lo tanto, pueden emplearse de manera efectiva para la predicción de variables meteorológicas.

Se puede decir que ambas metodologías son buenas ya que generan un pronóstico bastante adecuado a la realidad. Pero existen diferencia entre la metodología de Box Jenkins y la utilización de redes neuronales para pronósticos de series de tiempo, entre ellas se destaca que para trabajar con redes neuronales no es necesario que los datos necesiten estacionariedad o estacionalidad. Lo anterior se cumple porque las Redes Neuronales Artificiales aprenden sobre sus mismos datos y su propio comportamiento dando una gran ventaja, sin embargo no se puede dejar de lado la metodología que hay detrás de cada una de estas, ya que la metodología Box-Jenkins cuenta con respaldo más sólido desde el punto de vista matemático y las Redes Neuronales Artificiales son aún un método un tanto descriptivo ya que para poder llegar a un pronóstico adecuado, se tuvieron que hacer varios modelos para así determinar el mejor.

Los resultados muestran que el modelo de RNN supero al modelo *SARIMA* en nuestro problema de pronóstico, pero fue muy poca diferencia. Esto hace que la elección final del método a utilizar dependa definitiva de otros factores, como pueden ser el conocimiento más profundo de algún de los dos métodos o bien la relativa sencillas del aprendizaje y costo de aplicación para cada usuario en cuestión, de un método respecto al otro.

La mejor solución posiblemente sea no dejar al lado ninguno de los dos métodos, y utilizarlos conjuntamente de forma que los resultados obtenidos finalmente puedan ser validados por el otro método.

Apéndice A

Prueba de Dickey–Fuller

La prueba de Dickey–Fuller está diseñada para determinar si existe una raíz unitaria en una serie de tiempo. La hipótesis nula de esta prueba es que existe una raíz unitaria en la serie. Hay tres versiones principales de la prueba, a continuación, las expondremos:

- Versión 1: sin término constante y sin tendencia.

Supongamos el siguiente proceso $AR(1)$:

$$X_t = \phi X_{t-1} + \epsilon_t \quad \text{donde} \quad \epsilon_t \sim RB(0, \sigma^2), \quad (\text{A.1})$$

que es estacionario si $|\phi| < 1$. Si en cambio, $\phi = 1$ el proceso X_t no es estacionario porque tiene una raíz unitaria, observemos que sería un modelo de caminata aleatoria:

$$X_t = X_{t-1} + \epsilon_t.$$

Por lo tanto, se podría pensar en contrastar la hipótesis de existencia de raíz unitaria en la serie X_t (no estacionariedad) contrastando la hipótesis nula $H_0 : \phi = 1$ vs $H_a : \phi < 1$ en el modelo (A.1), pero esta prueba de cola izquierda es posible expresarla de una manera más conveniente.

El modelo (A.1) se puede escribir como sigue restando X_{t-1} en ambos lados,

$$X_t - X_{t-1} = \phi X_{t-1} - X_{t-1} + \epsilon_t \quad \text{o bien:}$$

$$\Delta X_t = \beta X_{t-1} + \epsilon_t \quad \text{con} \quad \beta = \phi - 1. \quad (\text{A.2})$$

Ahora, contrastar la hipótesis nula de raíz unitaria equivale a plantear el siguiente contraste de hipótesis en el modelo (A.2):

$$H_0 : \beta = 0 \quad \text{vs} \quad H_a : \beta < 0.$$

El modelo (A.2) es un modelo de regresión y el estadístico habitual para realizar este tipo de contraste es:

$$t = \frac{\hat{\beta} - 0}{\sqrt{\hat{V}(\hat{\beta})}} \quad (\text{A.3})$$

donde:

$$\hat{\beta} = \frac{\sum_{t=2}^T x_{t-1} \Delta x_t}{\sum_{t=2}^T x_{t-1}^2} \quad \text{y} \quad \hat{V}(\hat{\beta}) = \frac{\hat{\sigma}^2}{\sum_{t=2}^T x_{t-1}^2}. \quad (\text{A.4})$$

Al estadístico t se le denomina estadístico de Dickey-Fuller. Este estadístico t no sigue ninguna distribución conocida bajo la hipótesis nula de no estacionariedad por lo que Dickey-Fuller calcularon los percentiles de este estadístico bajo la H_0 proporcionando las tablas con los niveles críticos correctos para el estadístico en función del tamaño muestral (T) y el nivel de significación (α).

Se rechaza H_0 , es decir, la existencia de raíz unitaria a un nivel de significación α si el valor muestral del estadístico t es menor que el valor crítico de las tablas de Dickey-Fuller (DF_α):

$$t < DF_\alpha.$$

- Versión 2: con término constante, pero sin tendencia.

Si el modelo $AR(1)$ incluye una constante:

$$X_t = c + \phi X_{t-1} + \epsilon_t, \quad \epsilon_t \sim RB(0, \sigma^2). \quad (\text{A.5})$$

El contraste de existencia de raíz unitaria se realiza de forma similar a la versión 1. Restando X_{t-1} en ambos lados del proceso se obtiene:

$$\begin{aligned} \Delta X_t &= c + \beta X_{t-1} + \epsilon_t, \\ \beta &= \phi - 1. \end{aligned} \quad (\text{A.6})$$

El contraste de la hipótesis nula de raíz unitaria se plantea en este modelo (A.6) como sigue:

$$H_0 : \beta = 0 \quad \text{vs} \quad H_a : \beta < 0.$$

y el estadístico de contraste es:

$$t_c = \frac{\hat{\beta}}{\sqrt{\hat{V}(\hat{\beta})}},$$

donde $\hat{\beta}$ es el estimador *MCO* de β y $\hat{V}(\hat{\beta})$ el de su varianza. Como anteriormente, Dickey-Fuller construyeron las tablas que proporcionan los niveles críticos correctos para el estadístico t_c en función del tamaño muestral, T , y el nivel de significación, α .

- Versión 3: con término constante y con tendencia.

Si el modelo *AR*(1) incluye el término constante y una tendencia lineal:

$$X_t = c + \gamma t + \phi X_{t-1} + \epsilon_t, \quad \epsilon_t \sim RB(0, \sigma^2). \quad (\text{A.7})$$

El modelo que se estima por MCO es:

$$\Delta X_t = c + \gamma t + \phi X_{t-1} + \epsilon_t \quad (\text{A.8})$$

y se utiliza el estadístico t que denotaremos por t_γ para contrastar la hipótesis nula de raíz unitaria $H_0 : \beta = 0$ frente $H_a : \beta < 0$ [18].

Se han visto tres versiones de la prueba Dickey-Fuller, ahora la pregunta es: ¿Cuál versión de la prueba usar?

En respuesta de la pregunta anterior tenemos lo siguiente:

- Si la serie parece estar fluctuando alrededor de una muestra de media cero, usar la versión 1 de la prueba Dickey-Fuller.
- Si la serie parece estar fluctuando alrededor de una muestra de media que no es cero, usar la versión 2 de la prueba Dickey-Fuller.
- Si la serie parece estar fluctuando alrededor de una tendencia lineal, usar la versión 3 de la prueba Dickey-Fuller [24].

Apéndice B

Prueba de Ljung-Box

El análisis de residuos consiste en una serie de contrastes de diagnóstico con el objetivo de determinar si los residuos replican el comportamiento de un ruido blanco, es decir, si su media es cero, su varianza constante y las autocorrelaciones nulas.

Un método para el análisis de residuos es realizar un contraste de significatividad sobre los coeficientes de la *FAC*, siendo la hipótesis nula,

$$H_0 : \rho_1 = \rho_2 = \dots = \rho_K = 0$$

y la hipótesis alternativa que algún coeficiente ρ_k sea no nulo, para $k = 1, \dots, K$. El estadístico más utilizado para contrastar esta hipótesis es el propuesto por Ljung-Box(1978):

$$Q = n(n+2) \sum_{k=1}^K \frac{\hat{\rho}_k^2}{n-k}$$

donde Q es también conocido como estadístico de Ljung - Box. Este estadístico sigue aproximadamente una distribución chi-cuadrada con $(K - m)$ grados de libertad, donde $m = p + q$ es el número total de parámetros estimados.

Se rechaza la hipótesis nula de ausencia de correlación serial, para un nivel de significación del 5%, para valores grandes del estadístico, es decir, si

$$Q > \chi_{0.05}^2(K - m).$$

En este caso, existiría correlación serial en los residuos del modelo por lo que se concluye que el modelo no ha sido capaz de reproducir el patrón de comportamiento sistemático de la serie y habría que reformularlo [18].

Apéndice C

Código en Python para el análisis de la Metodología Box-Jenkins

```
1 #Cargando paqueterias necesarias.
2
3 import numpy as np
4 import pandas as pd
5 import warnings
6 warnings.filterwarnings('ignore')
7 import matplotlib.pyplot as plt
8 import statsmodels.tsa.stattools as sts
9 import statsmodels.graphics.tsaplots as sgt
10 import statsmodels.tsa.statespace
11 import statsmodels.api as sm
12 import matplotlib.pyplot as plt
13 plt.rcParams["figure.figsize"] = (11,6)
14
15
16 #Cargando el conjunto de datos de la radiacion solar.
17
18 df = pd.read_excel('Df1.xlsx')
19
20 #Se selecciona la columna Fecha como indice.
21
22 df['Fecha']=pd.date_range(pd.datetime(2020,1,1), periods=157, freq="w")
23 df['Fecha']=pd.to_datetime(df['Fecha'])
24 df.set_index('Fecha',inplace=True)
25
26 #Grafica de la serie de tiempo.
27
28 fig, ax = plt.subplots(figsize=(15,6))
29 ax.plot(df.index, df['Radiacion_solar'], color='r', marker='o', linestyle='
    -')
30 ax.set_title("Datos promediados semanalmente (2020-2022)")
31 plt.xlabel('Semana')
32 plt.ylabel('Radiacion Solar')
33 plt.grid()
34 plt.show()
35
36
```

Código en Python para el análisis de la Metodología Box-Jenkins

```
37 #Descomposicion aditiva de la serie temporal.
38
39 descomposicion_aditiva_RS = seasonal_decompose(df['Radiacion_solar'], model
    ="additive")
40 descomposicion_aditiva_RS.plot()
41 plt.figure(figsize=(15,5))
42 plt.show()
43
44 #Grafica del diagrama de cajas para verificar la estacionalidad.
45
46 fig, ax = plt.subplots(figsize=(15,6))
47 sns.boxplot(data=df, x='Semana', y='Radiacion_solar').set(title='Diagrama
    de cajas semanal de la radiacion del municipio de Puebla.')
48 plt.show()
49
50
51
52 sgt.plot_acf(df['Radiacion_solar'], lags=100, zero=True)
53 plt.show()
54
55 sgt.plot_pacf(df['Radiacion_solar'], lags=50, zero=True, method=('ywm'))
56 plt.show()
57
58 # Test Dickey - Fuller para la serie de tiempo
59 # Version 2: con termino constante pero sin tendencia
60
61 sts.adfuller(df['Radiacion_solar'], regression='c')
62
63 # Primera diferencia regular
64
65 df1=statsmodels.tsa.statespace.tools.diff(df, k_diff=1, k_seasonal_diff=
    None, seasonal_periods=1)
66 fig, ax = plt.subplots(figsize=(15,6))
67 ax.plot(df1.index, df1['Radiacion_solar'], color='r', marker='o', linestyle
    ='-')
68
69 ax.set_title("Grafica de la serie de tiempo con una diferencia regular.")
70 plt.xlabel('Tiempo.')
71 plt.ylabel('Radiacion Solar.')
72 plt.grid()
73 plt.show()
74
75 # Test Dickey - Fuller para la serie de tiempo con una diferencia regular
76 # Version 1: sin termino constante y sin tendencia.
77
78 sts.adfuller(df1['Radiacion_solar'], regression='nc')
79
80
81
82 sgt.plot_acf(df1['Radiacion_solar'], lags=100, zero=True)
83 plt.grid()
84 plt.show()
85
86 sgt.plot_pacf(df1['Radiacion_solar'], lags=50, zero=True, method=('ywm'))
87 plt.grid()
88 plt.show()
```

Código en Python para el análisis de la Metodología Box-Jenkins

```
89 # Diferencia regular y despues una estacional
90
91 diff_esta2=statsmodels.tsa.statespace.tools.diff(df1, k_seasonal_diff=1,
          seasonal_periods=52)
92
93 fig, ax = plt.subplots(figsize=(15,6))
94 ax.plot(diff_esta2.index, diff_esta2['Radiacion_solar'], color='r', marker=
          'o', linestyle='-')
95
96 ax.set_title("Grafica de la serie de tiempo con una diferencia regular y
          una diferencia estacional.")
97 plt.xlabel('Semana')
98 plt.ylabel('Tiempo')
99 plt.grid()
100 plt.show()
101
102
103 # Test Dickey - Fuller para la serie de tiempo con una diferencia regular y
          una estacional.
104 # Version 1 del Test Dickey - Fuller.
105
106 sts.adfuller(diff_esta2['Radiacion_solar'], regression='nc')
107
108
109
110 sgt.plot_acf(diff_esta2['Radiacion_solar'], lags=100, zero=True)
111 plt.grid()
112 plt.show()
113
114 sgt.plot_pacf(diff_esta2['Radiacion_solar'], lags=40, zero=True, method=(
          'ywmm'))
115 plt.grid()
116 plt.show()
117
118 # Creando el modelo $SARIMA(1,1,1)(1,1,0)_{52}$
119
120 modelo = SARIMAX(df['Radiacion_solar'], order=(1,1,1), seasonal_order=(1,
          1, 0, 52))
121 resultados = modelo.fit()
122
123 #Resumen del modelo creado.
124
125 print(resultados.summary())
126
127
128 # Grafico de los datos originales y el modelo ajustado.
129
130 fig, ax = plt.subplots(figsize=(15,6))
131 plt.plot(df['Radiacion_solar'], color="red", label='Serie original')
132 plt.plot(resultados.fittedvalues.iloc[1:], color="royalblue", label='Serie
          ajustada')
133 ax.legend()
134 plt.grid()
135 plt.show()
136
137
```

Código en Python para el análisis de la Metodología Box-Jenkins

```
138 # Imprimimos 5 predicciones del modelo.
139
140 resultados.predict(0,165).tail(6)
141
142 # Grafico de las predicciones del modelo.
143
144 fig, ax = plt.subplots(figsize=(15,6))
145 ax = df['Radiacion_solar'].plot(ax=ax, marker='o')
146 plot_predict(resultados, '2023-01-01', '2023-02-05', ax=ax)
147 plt.grid()
148 plt.show()
149
150 # AutoArima
151
152 model_auto = pmdarima.auto_arima(df['Radiacion_solar'], test='adf',
153                                 start_p=0,
154                                 start_q=0,
155                                 max_p=3,
156                                 max_q=3,
157                                 d=1,
158                                 D=1,
159                                 m=52,
160                                 seasonal=True,
161                                 error_action='ignore',
162                                 trace=True,
163                                 suppress_warnings=True,
164                                 stepwise=True)
165
166 #Resumen del modelo generado por AutoArima.
167
168 model_auto.summary()
169
170 # Analisis de los residuales.
171
172 model_auto.plot_diagnostics(figsize=(14,10))
173 plt.show()
```

Listing C.1: Código en Python.

Apéndice D

Código en Python para el análisis de Redes Neuronales Artificiales

```
1 #Cargando paqueterias necesarias.
2
3 import numpy as np
4 import pandas as pd
5 import warnings
6 warnings.filterwarnings("ignore")
7 import matplotlib.pyplot as plt
8 %matplotlib inline
9
10 from sklearn.metrics import mean_squared_error
11 from sklearn.preprocessing import MinMaxScaler
12 from statsmodels.tools.eval_measures import rmse
13 from keras.preprocessing.sequence import TimeseriesGenerator
14 from keras.models import Sequential
15 from keras.layers import Dense
16 from keras.layers import LSTM
17
18
19 #Cargando el conjunto de datos de la radiacion solar.
20
21 df = pd.read_excel('Df_rn.xlsx')
22
23 #Se selecciona la columna Fecha como indice.
24
25 df['Fecha']=pd.date_range(pd.datetime(2020,1,1), periods=157, freq="w")
26 df['Fecha']=pd.to_datetime(df['Fecha'])
27 df.set_index('Fecha', inplace=True)
28
29 #Imprimimos un resumen del conjunto de datos.
30
31 df.info()
32
33 # Observamos una estadística descriptiva de nuestro conjunto de datos.
34
35 df.describe()
36
37
```

Código en Python para el análisis de Redes Neuronales Artificiales

```
38
39 #Graficamos el conjunto de datos
40
41 fig, ax = plt.subplots(figsize=(15 ,6))
42 ax.plot(df.index , df["Radiacion_solar"], color = 'r', marker = 'o',
43         linestyle = '-')
44 ax.set_title(" Datos promediados semanalmente (2020 -2022)")
45 plt.grid()
46 plt.show()
47
48 #Dividimos al conjunto de datos en dos subconjuntos
49
50 Train__data = df[:len(df)-31]
51 Test__data = df[len(df)-31:]
52
53
54 #Definimos nuestro m todo de normalizaci n de datos.
55
56 scaler = MinMaxScaler()
57 #scaler = MinMaxScaler(feature_range=(-1, 1))
58
59
60 #Normalizamos al conjunto de datos de entrenamiento y prueba.
61
62 scaler.fit(Train__data)
63 scaled_Train__data = scaler.transform(Train__data)
64 scaled_Test__data = scaler.transform(Test__data)
65
66 # Como vamos a trabajar con datos temporales estos deben transformarse en
67 # una estructura de muestras con componentes de entrada y salida antes de
68 # que puedan usarse para ajustarse a un modelo de aprendizaje supervisado.
69
70 from keras.preprocessing.sequence import TimeseriesGenerator
71
72 n_input = 52
73 n_features= 1
74 generator = TimeseriesGenerator(scaled_Train__data, scaled_Train__data,
75                                length=n_input, batch_size=1)
76
77 # Generamos nuestro modelo de Red Neuronal
78
79 Modelo_LSTM = Sequential()
80 Modelo_LSTM.add(LSTM(200, activation='relu', return_sequences=True,
81                    input_shape=(n_input, n_features)))
82 Modelo_LSTM.add(LSTM(200, activation='relu'))
83 Modelo_LSTM.add(Dense(1))
84
85 # Definimos el optimizador que utilizaremos y la funcion de costo
86
87 Modelo_LSTM.compile(optimizer='Adam', loss='mse')
88
89 #Observamos el resumen de nuestra red neuronal.
90
91 Modelo_LSTM.summary()
```

```
91
92 # Entrenamos a nuestra Red Neuronal durante 100 epocas.
93
94 Modelo_LSTM.fit_generator(generator, epochs=100)
95
96
97 #Graficamos el comportamiento de la funcion de costo.
98
99 losses_lstm = Modelo_LSTM.history.history['loss']
100 plt.figure(figsize=(11,5))
101 plt.xticks(np.arange(0,21,1))
102 plt.plot(range(len(losses_lstm)), losses_lstm);
103
104
105 Predicciones_LSTM_escalados = list()
106
107 batch = scaled_Train__data[-n_input:]
108 CurrentBatch = batch.reshape((1, n_input, n_features))
109
110 for i in range(len(Test__data)):
111     Pred_LSTM = Modelo_LSTM.predict(CurrentBatch)[0]
112     Predicciones_LSTM_escalados.append(Pred_LSTM)
113     CurrentBatch = np.append(CurrentBatch[:,1:,:], [[Pred_LSTM]], axis=1)
114
115
116 Predicciones_LSTM_escalados
117
118 Predicciones_LSTM = scaler.inverse_transform(Predicciones_LSTM_escalados)
119
120 # Guardamos los datos pronosticados en una columna de nuestro dataframe.
121
122 Test__data['Predicciones_LSTM'] = Predicciones_LSTM
123
124 # Imprimimos a los pronosticados.
125
126 Test__data
127
128 # Graficamos el conjunto de datos de prueba junto a los datos pronosticados
129 .
130 Test__data['Radiacion_solar'].plot(figsize = (15,6), legend=True)
131
132 Test__data['Predicciones_LSTM'].plot(legend = True)
```

Listing D.1: Código en Python.

Apéndice E

Selección de datos

Para la selección de datos se realizó lo siguiente.

1. En el conjunto de datos de meteorología de la estación UTP se encuentra información a partir del 1 de enero del 2001 al 31 de diciembre del 2021 de distintas variables por ejemplo velocidad del viento, intensidad del aire, temperatura, radiación solar, etc. pero en este trabajo se enfocó solamente en los datos de la radiación solar, los cuales están disponibles del 1 de enero del 2017 al 31 de diciembre del 2021 y se completó este conjunto de datos con datos del año 2022 los cuales fueron descargados de la página de SINAICA. El conjunto de datos se recoleta con una frecuencia de hora, por ejemplo, para la fecha 01/01/2019 se pude observar lo siguiente:

- En la Figura E.1, obsérvese que la recolección de información inicia a las 12:00 a.m. y a partir de ahí cada hora, hasta las 11:00 p.m.

Continuando sobre el mismo ejemplo, se decidió encontrar el promedio de la columna Radiación solar de la fecha 01/01/2019 tomando todos los valores e incluso los ceros que aparecen. Así, se obtuvo que el promedio de la fecha 01/01/2019 y fue 200.75.

De la misma manera se ejecutó lo mismo sobre la totalidad del conjunto de datos, siempre y cuando ya estuvieran completos los datos.

2. Por otro lado, se observó datos faltantes en todos los años 2017-2022. Para los años 2017 y 2018 tenemos demasiados datos faltantes, 3343 y 3458 correspondientemente y mientras que el año 2019 contenía inconsistencias en los datos, por lo cual estos años fueron descartados. Para los años 2020, 2021 y 2022 contaban con pocos datos faltantes 450, 381 y 2000 correspondientemente. Entonces, para completar los datos faltantes, se hizo lo siguiente.

- Primero, se inició completando los datos que faltan para 2022, lo cual se hizo tomando las mismas fechas y horas de los dos años anteriores para

posteriormente promediarlos (obsérvese la Figura E.2).

Por ejemplo, en la Figura E.2, hacía falta el dato de la fecha 01/01/2022 08:00 entonces se rellenó con la parte entera del promedio de las fechas 01/01/2020 08:00 y 01/01/2021 08:00.

Para los años 2020 y 2021 se hizo análogamente a lo anterior, para el año 2021 se realizó con las fechas de un año anterior (2020) y uno siguiente (2022), y para el año 2020 se realizó con las fechas de los dos años siguientes (2021 y 2022).

Ya realizado lo anterior aun existan algunos datos faltantes, no se pudo hacer un promedio ya que el dato faltaba en los cuatro años en la misma fecha y hora (ver Figura E.3). Por lo cual se prosiguió a juntar todas las fechas de todos los años en una sola columna, después con la paquetería *forecast* de R se utilizó la función *na.interp* la cual nos ayuda hacer una interpolación para rellenar estos datos faltantes.

3. Una vez completos todo el conjunto de datos, con funciones de la paquetería *pandas* de Python se agrupo por semana y por último se empezó a promediar semanalmente, hasta obtener un conjunto de datos como se observa en la Figura E.4.
 - El conjunto de datos cuenta con 157 filas ya que los años 2021 y 2022 cuentan con 52 semanas mientras el año 2020 cuenta con 53 semanas.

La primera semana empieza desde el 01/01/2020 00:00:00 hasta el 05/01/2020 23:00:00, así que, el promedio de la semana del 1 al 5 de enero de 2020 resulta ser 173.23750, la segunda semana empieza a partir del 06/01/2020 00:00:00 al 12/01/2020 00:00:00 y su respectivo promedio es 190.791667, este proceso continúa sucesivamente hasta obtener un conjunto de datos de promedios semanales el cual es el que se utiliza para realizar el análisis.

| Fecha | Hora | Radiación solar. |
|------------|-------------|------------------|
| 17/07/2020 | 12:00 a. m. | 0 |
| 17/07/2020 | 01:00 a. m. | 0 |
| 17/07/2020 | 02:00 a. m. | 0 |
| 17/07/2020 | 03:00 a. m. | 0 |
| 17/07/2020 | 04:00 a. m. | 0 |
| 17/07/2020 | 05:00 a. m. | 0 |
| 17/07/2020 | 06:00 a. m. | 0 |
| 17/07/2020 | 07:00 a. m. | 0 |
| 17/07/2020 | 08:00 a. m. | 36 |
| 17/07/2020 | 09:00 a. m. | 51 |
| 17/07/2020 | 10:00 a. m. | 72 |
| 17/07/2020 | 11:00 a. m. | 75 |
| 17/07/2020 | 12:00 p. m. | 84 |
| 17/07/2020 | 01:00 p. m. | 393 |
| 17/07/2020 | 02:00 p. m. | 502 |
| 17/07/2020 | 03:00 p. m. | 677 |
| 17/07/2020 | 04:00 p. m. | 586 |
| 17/07/2020 | 05:00 p. m. | 631 |
| 17/07/2020 | 06:00 p. m. | 425 |
| 17/07/2020 | 07:00 p. m. | 99 |
| 17/07/2020 | 08:00 p. m. | 30 |
| 17/07/2020 | 09:00 p. m. | 0 |
| 17/07/2020 | 10:00 p. m. | 0 |
| 17/07/2020 | 11:00 p. m. | 0 |

Figura E.1: Datos del 17 de julio de 2020.

| Fecha | Radiacion solar 2020 | Radiacion solar 2021 | Radiacion solar 2022 |
|------------------|----------------------|----------------------|----------------------|
| 01/01/2020 08:00 | 58 | 95 | 76 |
| 01/01/2020 09:00 | 142 | 217 | 179 |
| 01/01/2020 10:00 | 167 | 413 | 290 |
| 01/01/2020 11:00 | 498 | 562 | 530 |
| 01/01/2020 12:00 | 400 | 666 | 533 |
| 01/01/2020 13:00 | 331 | 709 | 520 |
| 01/01/2020 14:00 | 496 | 689 | 592 |
| 01/01/2020 15:00 | 495 | 603 | 549 |
| 01/01/2020 16:00 | 426 | 458 | 442 |
| 01/01/2020 17:00 | 304 | 268 | 286 |
| 01/01/2020 18:00 | 75 | 67 | 71 |
| 01/01/2020 19:00 | 1 | 1 | 1 |
| 01/01/2020 20:00 | 0 | 0 | 0 |
| 01/01/2020 21:00 | 0 | 0 | 0 |
| 01/01/2020 22:00 | 0 | 0 | 0 |
| 01/01/2020 23:00 | 0 | 0 | 0 |

Figura E.2: Datos faltantes 2022.

| Fecha | Radiacion solar 2020 | Radiacion solar 2021 | Radiacion solar 2022 |
|------------|----------------------|----------------------|----------------------|
| 01/01/2020 | | | |
| 02/01/2020 | | | |
| 03/01/2020 | | | |
| 04/01/2020 | | | |
| 05/01/2020 | | | |
| 06/01/2020 | | | |
| 07/01/2020 | | | |
| 08/01/2020 | | | |
| 09/01/2020 | | | |
| 10/01/2020 | 47 | | |
| 11/01/2020 | | | |
| 12/01/2020 | | | |
| 13/01/2020 | | | |
| 14/01/2020 | | | |

Figura E.3: Datos faltantes en todos los años.

| Fecha | Semana | Radiacion_solar |
|------------|--------|-----------------|
| 2020-01-05 | 1 | 173.237500 |
| 2020-01-12 | 2 | 190.791667 |
| 2020-01-19 | 3 | 165.910714 |
| 2020-01-26 | 4 | 159.541667 |
| 2020-02-02 | 5 | 205.110119 |
| ... | ... | ... |
| 2022-12-04 | 48 | 180.809524 |
| 2022-12-11 | 49 | 192.220238 |
| 2022-12-18 | 50 | 187.279762 |
| 2022-12-25 | 51 | 159.083333 |
| 2023-01-01 | 52 | 154.550595 |

Figura E.4: Conjunto de datos semanalmente.

Bibliografía

- [1] GOBIERNO DE MÉXICO. ¿QUÉ ES EL CLIMA?, <https://www.gob.mx/inecc/acciones-y-programas/que-es-el-clima>.
- [2] MÉXICO ANTE EL CAMBIO CLIMÁTICO. ¿QUÉ ES EL CLIMA?, <https://cambioclimatico.gob.mx/que-es-el-clima/>, 2021.
- [3] RODRÍGUEZ JIMÉNEZ ROSA MARÍA, BENITO CAPA ÁGUEDA y PORTELA LOZANO ADELAIDA, *Meteorología y Climatología*, primera edición, FECYT, 2004.
- [4] ACUÑA VALVERDE DAYANA y ROBLES SÁNCHEZ DANNY, *MANUAL DE METEOROLOGÍA Y DE GESTIÓN DE LA INFORMACIÓN CLIMÁTICA*, primera edición, Instituto de Montaña, 2015.
- [5] AHRENS C. DONALD, *Essentials of Meteorology: An Invitation to the Atmosphere*, tercera edición, Cengage Learning Custom Publishing, 2001.
- [6] FONDRIEST STAFF. <https://www.fondriest.com/news/wind-speed-and-direction.htm>, 2022.
- [7] MARTÍNEZ MARCELO, *RADIACIÓN SOLAR – CONCEPTOS Y APLICACIONES*, Informativo INAI-URURI, no. 109, 2016.
- [8] NATIONAL GEOGRAPHIC. EL SOL: QUÉ ES, CÓMO AFECTA A LA TIERRA Y CUÁNDO SE APAGARÁ, <https://www.nationalgeographic.es/espacio/el-sol-0>.
- [9] P. UNAM. CAPÍTULO 2. EL POTENCIAL DE LA RADIACIÓN SOLAR, <http://www.ptolomeo.unam.mx:8080/xmlui/bitstream/handle/132.248.52.100/277/A5.pdf>.
- [10] IBERDROLA. RADIACIÓN SOLAR, <https://www.iberdrola.com/compromiso-social/radiacion-solar>.
- [11] PETERSSEN SVERRE, *Introducción a la Meteorología*, quinta edición, Espasa-Calpe, S. A., Madrid, 1976.
- [12] MÉXICO EN CIFRAS, <https://www.inegi.org.mx/app/areasgeograficas/?ag=21114#collapse-Resumen>, 2020.
- [13] INSTITUTO MEXICANO DE TECNOLOGÍA DEL AGUA. ¿QUÉ ES LA CANÍCULA?, <https://www.gob.mx/imta/articulos/que-es-la-canicula?idiom=es>, 2019.

- [14] SERVICIO METEOROLÓGICO NACIONAL. *Inicia el periodo de la canícula en México: SMN.*, <https://acortar.link/In9QkI>, 2010.
- [15] NATIONAL GEOGRAPHIC SOCIETY, S. F. EL NIÑO, <https://education.nationalgeographic.org/resource/el-nino>, 2022.
- [16] NATIONAL GEOGRAPHIC SOCIETY, S. F. LA NIÑA, <https://education.nationalgeographic.org/resource/la-nina>, 2022.
- [17] BOWERMAN BRUCE L., O'CONNELL, R. T., Y KOEHLER, A. B., *Pronósticos, series de tiempo y regresión: Un enfoque aplicado*, cuarta edición, CENGAGE Learning, 2007.
- [18] GONZÁLEZ, C. M. P., *SARRIKO-ON Análisis de series temporales : Modelos ARIMA*, Universidad del País Vasco, 2009.
- [19] COMISIÓN NACIONAL DEL AGUA. REPORTE DEL CLIMA EN MÉXICO, <https://smn.conagua.gob.mx/es/climatologia/diagnostico-climatico/reporte-del-clima-en-mexico>
- [20] BARBOSA CORREA RODRIGO Y LLINAS SOLANO HUMBERTO, *PROCESOS ESTO-CÁSTICOS CON APLICACIONES*, primera edición, Universidad del Norte, 2016.
- [21] GUJARATI DAMODAR N. Y PORTER DAWN C., *Econometría*, quinta edición, MCGRAW-HILL, 2010.
- [22] WILLIAM W. S. WEI, *Time Series Analysis : Univariate and Multivariate Methods*, segunda edición, Pearson Addison Wesley, 2006.
- [23] BONILLA ALARCON JULIO ADRIAN, *Tesis sobre Análisis del crecimiento económico a través del indicador global de la actividad económica (IGAE) bajo la metodología de Box-Jenkins y redes neuronales artificiales*, Benemérita Universidad Autónoma de Puebla, 2022.
- [24] ARAGÓN MERINO GLORIA, *Tesis sobre Análisis de la radiación solar en el municipio de Tlaxco-Tlaxcala usando la metodología de Box-Jenkins*, Benemérita Universidad Autónoma de Puebla, 2018.
- [25] AWS, S. F. ¿QUÉ ES EL MACHINE LEARNING? - EXPLICACIÓN SOBRE EL MACHINE LEARNING EMPRESARIAL, <https://aws.amazon.com/es/what-is/machine-learning/>.
- [26] BBVA. MACHINE LEARNING: ¿QUÉ ES Y CÓMO FUNCIONA?, <https://www.bbva.com/es/machine-learning-que-es-y-como-funciona/>.
- [27] IBM, S. F. WHAT ARE NEURAL NETWORKS?, <https://www.ibm.com/topics/neural-networks>.
- [28] IBM, S. F. WHAT IS GRADIENT DESCENT?, <https://www.ibm.com/topics/gradient-descent>.

- [29] IBM, S. F. ¿QUÉ ES SOBREAJUSTE?, <https://www.ibm.com/mx-es/topics/overfitting#:~:text=de%20su%20modelo.-,%C2%BFQu%C3%A9%20es%20overfitting%3F,datos%20invisibles%2C%20frustrando%20su%20prop%C3%B3sito.>
- [30] IBM, S. F. ¿QUÉ SON LAS REDES NEURONALES RECURRENTE?, <https://www.ibm.com/mx-es/topics/recurrent-neural-networks>.
- [31] MATHWORKS AMÉRICA LATINA. (S. F.). OPCIONES PARA ENTRENAR UNA RED NEURONAL DE DEEP LEARNING., https://la.mathworks.com/help/deeplearning/ref/trainingoptions_es.html.
- [32] GOOGLE DEVELOPERS. NORMALIZACIÓN MACHINE LEARNING, <https://developers.google.com/machine-learning/data-prep/transform/normalization?hl=es-419>.
- [33] M. N. S. SWAMY Y KE-LIN DU, *Neural Networks and Statistical Learning*, segunda edición, Springer London, 2019.
- [34] NIELSEN MICHAEL, *Neural Networks and Deep Learning: Introduction to the core principles*, primera edición, 2019.
- [35] MATHWORKS. LONG SHORT-TERM MEMORY (LSTM) NETWORKS, <https://la.mathworks.com/discovery/lstm.html>.
- [36] MATHWORKS AMÉRICA LATINA. REDES DE MEMORIA DE CORTO-LARGO PLAZO) NETWORKS, <https://la.mathworks.com/help/deeplearning/ug/long-short-term-memory-networks.html>.
- [37] GOBIERNO DE MÉXICO. REPORTE DEL CLIMA EN MÉXICO, <https://smn.conagua.gob.mx/es/climatologia/diagnostico-climatico/reporte-del-clima-en-mexico>.