



ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO
FACULTAD DE CIENCIAS
CARRERA ESTADÍSTICA

ANÁLISIS ESTADÍSTICO PARA PRONÓSTICOS
METEREOLÓGICOS EN UN APLICATIVO WEB EN LA
PROVINCIA DE CHIMBORAZO

Trabajo de Titulación

Tipo: Proyecto de Investigación

Presentado para optar al grado académico de:

INGENIERA EN ESTADÍSTICA INFORMÁTICA

AUTORA: BELÉN CAROLINA PAUCAR CHAVARREA

DIRECTORA: ING. NANCY ELIZABETH CHARIGUAMÁN MAURISACA

Riobamba – Ecuador

2023

© 2023, **Belén Carolina Paucar Chavarrea**

Se autoriza la reproducción total o parcial, con fines académicos, por cualquier medio o procedimiento, incluyendo la cita bibliográfica del documento, siempre y cuando se reconozca el derecho de Autor.

Yo, BELÉN CAROLINA PAUCAR CHAVARREA, declaro que el presente trabajo de titulación es de mi autoría y los resultados del mismo son auténticos. Los textos en el documento que provienen de otras fuentes están debidamente citados y referenciados.

Como autora asumo la responsabilidad legal y académica de los contenidos de este trabajo de titulación; el patrimonio intelectual pertenece a la Escuela Superior Politécnica de Chimborazo.

Riobamba, 25 de abril del 2023


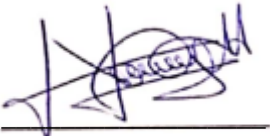

A handwritten signature in blue ink, consisting of several loops and a long horizontal stroke at the bottom.

Belén Carolina Paucar Chavarrea

060485796-1

ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO
FACULTAD DE CIENCIAS
CARRERA ESTADÍSTICA

El Tribunal del Trabajo de Titulación, certifica que: El Trabajo de Titulación; Proyecto de Investigación: **ANÁLISIS ESTADÍSTICO PARA PRONÓSTICOS METEREOLÓGICOS EN UN APLICATIVO WEB EN LA PROVINCIA DE CHIMBORAZO**, realizado por la señorita: **BELÉN CAROLINA PAUCAR CHAVARREA**, ha sido minuciosamente revisado por los Miembros del Tribunal del Trabajo de Titulación. El mismo que cumple con los requisitos científicos, técnicos, legales, en tal virtud el Tribunal autoriza su presentación.

	FIRMA	FECHA
Dra. Jaqueline Elizabeth Balseca Castro. PRESIDENTE DEL TRIBUNAL	 _____	2023-04-25
Ing. Nancy Elizabeth Chariguamán Maurisaca DIRECTOR DEL TRABAJO DE TITULACION	 _____	2023-04-25
Ing. Johanna Enith Aguilar Reyes ASESOR DEL TRABAJO DE TITULACIÓN	 _____	2023-04-25

DEDICATORIA

El presente trabajo es dedicado a Dios quien es mi guía y a mi hija Doménica por ser mi fuente de motivación e inspiración para poder superarme cada día más.

A mis padres Mauro Paucar y Inés Chavarrea, por el apoyo moral y económico, por cada consejo y la confianza que me brindaron quienes han inculcado en mi valores y principios, la mentalidad de siempre luchar por mis sueños y de nunca rendirme.

Belén

AGRADECIMIENTO

A Dios por guiarme y bendecirme para poder culminar una meta más, permitiendo que personas de buen corazón aporten con su ayuda incondicional, a mis padres por su constante apoyo y preocupación.

Agradecer a toda la planta docente de la Escuela que semestre a semestre fueron impartiendo conocimientos que me han servido para crecer como persona y profesional. Además, agradecer a la Ing. Nancy Chariguamán tutora de mi tesis y Docente de varias materias a lo largo de mi trayectoria en la institución, y a la Ing. Johanna Aguilar miembro del trabajo de titulación que ha guiado de la mejor manera por varios años mi camino como estudiante y en este trabajo.

Belén

ÍNDICE DE CONTENIDO

ÍNDICE DE TABLAS.....	ix
ÍNDICE DE ILUSTRACIONES.....	x
ÍNDICE DE ECUACIONES	xii
ÍNDICE DE ANEXOS	xiii
RESUMEN.....	xiv
ABSTRACT.....	xiv
INTRODUCCIÓN.....	1

CAPÍTULO I

1. MARCO TEÓRICO REFERENCIAL.....	6
1.1. Bases conceptuales.....	6
1.1.1. Meteorología	6
1.1.2. Pronósticos	7
1.1.3. Historias de los pronósticos	7
1.1.4. Box Jenkis	8
1.1.5. Identificación del modelo de Box Jenkins	9
1.1.6. Estimación del modelo Box Jenkins.....	10
1.1.7. Diagnóstico del modelo Box Jenkins.....	10
1.1.8. Modelo matemático ARIMA	11
1.1.9. Redes Neuronales	11
1.1.10. Funcionamiento de las redes neuronales.....	13
1.1.11. Propiedades de las redes neuronales	13
1.1.12. Tipos de redes neuronales.....	14
1.1.13. Modelo matemático de la red neuronal.....	15
1.1.14. Aplicaciones	15
1.1.15. Teoría del Caos	16
1.1.16. Diseño de aplicaciones web.....	18
1.2. Bases teóricas	19
1.2.1. Meteorología y climatología	19
1.2.2. Clima	19
1.2.3. Tiempo.....	20

1.2.4. Factores del clima.....	20
1.2.5. Elementos del clima.....	20
1.2.6. Estación meteorológica	20
1.2.7. Variable meteorológica	20
1.2.8. Temperatura	21
1.2.9. Velocidad del viento	21
1.2.10. Humedad	21
1.2.11. Pronóstico meteorológico.....	21
1.2.12. Métodos Estadísticos.....	22
1.2.13. Análisis exploratorio de Datos	22
1.2.14. Medidas de Tendencia Central	22
1.2.15. Medidas de Dispersión	23
1.2.16. Serie de tiempo	24
1.2.17. Coeficiente de Determinación R^2 Ajustado.....	24
1.2.18. Neurona.....	25
1.2.19. Inteligencia artificial	25
1.2.20. Usuario	25
1.2.21. Dispositivo.....	25
1.2.22. Aplicativo Web	26

CAPÍTULO II

2. MARCO METODOLÓGICO.....	27
2.1. Tipo de la Investigación	27
2.2. Diseño de la investigación	27
2.2.1. Localización de estudio	27
2.2.2. Población de estudio.....	27
2.2.3. Método de muestreo	28
2.2.4. Tamaño de la muestra.....	28
2.2.5. Técnica de recolección de datos	28
2.2.6. Identificación de variables.....	28
2.2.7. Modelo estadístico	28
2.3. Variables en estudio	29
2.3.1. Operacionalización de variables	29

CAPÍTULO III

3. MARCO DE RESULTADOS Y DISCUSIÓN DE RESULTADOS	30
3.1. Análisis exploratorio de información del GEAA	30
3.1.1. Depuración de datos meteorólogos	30
3.1.2. Análisis de datos	31
3.2. Modelos Box Jenkins, Redes Neuronales y Teoría del caos.....	39
3.2.1. Modelos Box Jenkins.....	39
3.2.2. Redes Neuronales	46
3.2.3. Teoría del caos	51
3.2.4. Comparación de métodos de pronósticos.....	53
3.2.5. Aplicativo web	56
DISCUSIÓN.....	58
CONCLUSIONES	59
RECOMENDACIONES.....	60
BIBLIOGRAFÍA	
ANEXOS	

ÍNDICE DE TABLAS

Tabla 1-2:	Operacionalización de variables.....	29
Tabla 1-3:	Valores faltantes según estación meteorológica y periodo	31
Tabla 2-3:	Resumen de estadísticas de estación meteorológica Alao con información imputada.	37
Tabla 3-3:	Resumen de estadísticas de estación meteorológica ESPOCH con información imputada.	38
Tabla 4-3:	Resumen de estadísticas de estación meteorológica Matus con información imputada.	38
Tabla 5-3:	Resumen de estadísticas de estación meteorológica Matus con información imputada.	38
Tabla 6-3:	Resumen de estadísticas de estación meteorológica Tunshi con información imputada.	39
Tabla 7-3:	Valores de p de la Prueba de Dickey Fuller.	44
Tabla 8-3:	Mejores modelos ARIMA, según estación y variable.	45
Tabla 9-3:	Valores p de pruebas de supuestos en modelos ARIMA según estación y variable.	46
Tabla 10-3:	Medidas de adecuación del modelo de Elma y Jordan según estación y variable.	47
Tabla 11-3:	Medidas de adecuación Teoría del Caos según estación y variable.....	52
Tabla 12-3:	Coeficiente U Theil según estación y variable.....	54
Tabla 12-3:	Test de Diabolt Mariano según estación y variable.	55

ÍNDICE DE ILUSTRACIONES

Ilustración 1-3:	Temperatura ambiente promedio según estaciones meteorológicas, periodo 2014-2021, con información imputada.....	33
Ilustración 2-3:	Temperatura ambiente promedio según estaciones meteorológicas, periodo 2014-2021, sin información imputada.....	33
Ilustración 3-3:	Humedad relativa promedio según estaciones meteorológicas, periodo 2014-2021, con información imputada.....	34
Ilustración 4-3:	Humedad relativa promedio según estaciones meteorológicas, periodo 2014-2021, sin información imputada.....	34
Ilustración 5-3:	Presión atmosférica promedio según estaciones meteorológicas, periodo 2014-2021, con información imputada.....	35
Ilustración 6-3:	Presión atmosférica promedio según estaciones meteorológicas, periodo 2014-2021, sin información imputada.....	35
Ilustración 7-3:	Radiación solar difusa promedio según estaciones meteorológicas, periodo 2014-2021, con información imputada.....	36
Ilustración 8-3:	Radiación solar difusa promedio según estaciones meteorológicas, periodo 2014-2021, sin información imputada.....	36
Ilustración 9-3:	Velocidad de viento promedio según estaciones meteorológicas, periodo 2014-2021, con información imputada.....	37
Ilustración 10-3:	Velocidad de viento promedio según estaciones meteorológicas, periodo 2014-2021, sin información imputada.....	37
Ilustración 11-3:	Gráfico de series de tiempo, temperatura ambiente promedio según estaciones meteorológicas, periodo 2014-2021.....	40
Ilustración 12-3:	Gráfico de series de tiempo, humedad relativa promedio según estaciones meteorológicas, periodo 2014-2021.....	41
Ilustración 13-3:	Gráfico de series de tiempo, presión atmosférica promedio según estaciones meteorológicas, periodo 2014-2021.....	42
Ilustración 14-3:	Gráfico de series de tiempo, radiación solar difusa promedio según estaciones meteorológicas, periodo 2014-2021.....	43
Ilustración 15-3:	Gráfico de series de tiempo, velocidad de viento promedio según estaciones meteorológicas, periodo 2014-2021.....	44

Ilustración 16-3:	Mejores modelo de Red Neuronal de Elman: Evolución del error según el número de iteraciones	49
Ilustración 18-3:	Mejores modelo de Red Neuronal de Elman: valores reales de la serie de tiempo (color negro) y valores pronosticados (color rojo) vs tiempo.....	50
Ilustración 19-3:	Mejores modelo de Red Neuronal de Jordan: Valores reales de la serie de tiempo (color negro) y valores pronosticados (color rojo) vs tiempo.....	50
Ilustración 20-3:	Teoría del Caos: valores reales de la serie de tiempo (color negro) y valores pronosticados (color rojo) vs tiempo.....	53
Ilustración 21-3:	Pantalla del aplicativo web.....	56
Ilustración 22-3:	Resultados de ARIMA en el aplicativo web	57

ÍNDICE DE ECUACIONES

Ecuación 1-1:	Modelo Matematico ARIMA	11
Ecuación 2-1:	Modelo de la red neurona.....	15
Ecuación 3-1:	Media Aritmética.....	23
Ecuación 4-1:	Mediana	23
Ecuación 5-1:	Desviación Estándar	24
Ecuación 6-1:	Rango.....	24
Ecuación 7-1:	Coficiente de determinacion R^2 ajustado.....	25

ÍNDICE DE ANEXOS

ANEXO A: LINK DE LA MATRIZ DE DATOS FACILITADO POR EL GEAA

ANEXO B: CÓDIGO EN R

RESUMEN

Las variables meteorológicas de las estaciones emitidas por Grupo de energía alternativa y ambiente de la Escuela Superior Politécnica de Chimborazo son difícil de comprender ya que presenta una constante variación climática lo cual nos impide tratar información con un patrón definido, por tanto tiene como objetivo determinar pronósticos meteorológicos mediante la utilización de tres metodologías de modelamiento definidas por: Box-Jenkins, Redes Neuronales y Teoría de Caos con la finalidad de ayudar en la toma de decisiones futuras. La metodología implementada tuvo un enfoque cuantitativo, se utilizó un diseño no experimental de tipo transversal ya que no existió ninguna participación directa en la manipulación de las variables en su ambiente natural, con un nivel de profundización exploratoria, un método de estudio inductivo por que se va a modelar variables meteorológicas y se desarrolló en tiempo determinado. Se realizó un análisis exploratorio de los datos en el cual se obtuvo, como porcentaje general que el 8.66% de la información son valores faltantes en el cual se procedió a realizar la imputación de los datos mediante regresión lineal. En términos generales se aplicó técnicas estadísticas para el cálculo de los pronósticos. Se concluye que el mejor método de pronóstico resulto ser las Redes Neuronales, seguido por los modelos ARIMA y luego la Teoría del Caos según el coeficiente U Theil y el Test de Diabolt Mariano y los resultados de los pronósticos meteorológicos según estaciones serán reflejados en un aplicativo web.

Palabras clave: <TEORIA DEL CAOS>, <ARIMA>, <REDES NEURONALES>, <APLICATIVO WEB>, <SERIES DE TIEMPO>, <METERELOGIA> <PRONOSTICOS>.


0898-DBRA-UPT-2023



ABSTRACT

The meteorological variables of the stations issued by the Alternative Energy and Environment Group of the Escuela Superior Politécnica de Chimborazo are difficult to understand since it presents a constant climatic variation which prevents us from treating information with a defined pattern, therefore its objective is to determine forecasts through the use of three modeling methodologies defined by: Box-Jenkins, Neural Networks and Chaos Theory in order to help in making future decisions. The implemented methodology had a quantitative approach, a non-experimental design of a cross-sectional type was used since there was no direct participation in the manipulation of the variables in their natural environment, with a level of exploratory depth, an inductive study method because it was going to model meteorological variables and it was developed in a determined time. An exploratory analysis of the data was done in which it was obtained, as a general percentage, that 8.66% of the information are missing values in which the imputation of the data was performed through linear regression. In general terms, statistical techniques were applied to calculate forecasts. It is concluded that the best forecast method proves to be the Neural Networks, followed by the ARIMA models and then the Chaos Theory according to the Theil's U coefficient and the Diebold- Mariano Test and the results of the meteorological forecasts according to stations will be reflected in a web application.

Keywords: <CHAOS THEORY>, <ARIMA>, <NEURAL NETWORKS>, <WEB APPLICATION>, <TIME SERIES>, <METEOROLOGY> <FORECASTS>.



Edgar Mesias Jaramillo Moyano.
0603497397

INTRODUCCIÓN

La meteorología es una representación simplificada del tiempo atmosférico en un futuro basados de relaciones climatológicas, que se construyen para ser utilizados de forma explícita en la investigación de los fenómenos naturales que tiende a tener un potencial alto y que sirven en la elaboración para los pronósticos de tiempos futuros.

Entre las instituciones que colaboran en la recolección de la información se cuenta con un incremento significativo de Estaciones Meteorológicas, tal es el caso que la Organización Mundial Meteorológica define a las estaciones como "sitios de observación con transmisión automática" y se han constituido con el objetivo de monitorear las condiciones atmosféricas presentes en un momento y lugar específico (Gattinoni, et al., 2011, pp. 75-85).

La disponibilidad de información meteorológica confiable y representativa de una zona contribuye a una buena planificación en urbanismo, ingeniería o en ordenación del territorio, también es utilizada para agronomía, hidrología, análisis del tiempo, pronósticos del tiempo, advertencias de clima extremo, operaciones locales que dependen del clima entre otras. En la agricultura es importante para: la estimación de evapotranspiración de referencia y para estimar el consumo hídrico (Arteaga, et al., 2017, p. 4).

Entre las variables utilizadas para estudios climáticos o aportes a otras áreas se destacan la temperatura, velocidad de viento, precipitación, presión atmosférica, humedad y radiación solar, cada una de ellas genera un aporte al momento de identificar el estado del tiempo pues la alteración de alguna variable genera grandes variaciones en los promedios del clima. Según el Grupo Intergubernamental sobre el Cambio Climático, las condiciones de variación se producen a escala mundial, sus efectos a menudo se modifican de una región a otra, en el caso de Ecuador las condiciones climáticas son influenciadas por dos variables principales la temperatura y la precipitación, que dan lugar a marcados cambios temporales y espaciales en las diferentes regiones del país. En este colectivo se observan dos épocas bien definidas por la distribución temporal de las precipitaciones, una época lluviosa y otra seca, excepto en la Amazonía, ya que las lluvias son consideradas durante todo el año (Alexandra Marcatoma, 2020, p. 3)

En esta investigación se plantea un análisis estadístico general de la información recolectada por el Grupo de Energía Alternativa y Ambiente "GEAA" de la Escuela Superior Politécnica de Chimborazo con la finalidad de calcular pronósticos de variables meteorológicas mediante la selección de una técnica de pronóstico idónea para lo cual se propone el uso de los modelos Box

Jenkins, Teoría del caos y Redes Neuronales a fin de evaluar el ajuste de los datos reales con los valores pronosticados.

Tras la selección de la mejor metodología de pronósticos se construye un aplicativo web donde la información está disponible a cualquier persona interesada en conocer el estado de clima de la provincia de Chimborazo y que la misma sea de beneficio no solo al área académica sino también económica, agrícola y social.

ANTECEDENTES

Desde el principio la meteorología como ciencia y técnica se remonta a los comienzos de la humanidad realizando intentos para comprender. En su inicio los estudios que describen la meteorología con un enfoque científico tuvieron lugar en la civilización griega con: Tales Mileto en los (624-545 a.C.), Anaximandro en los (611-547 a.C.), Hipócrates en los años (460-375 a.C.) y otros científicos griegos. El estudio más detallado de esta época se refiere a Aristóteles que fue en (384-322 a.C.) quien acuñó el término "Meteorología", que proviene de la palabra griega Meteoros, "alto en el cielo" y lógicamente, "hormiga". fórmula, tratado" (Palomares, 2011).

Se denomina metrología al estudio de la atmósfera, cuyas bases son: física, matemáticas, la mecánica y química todas estas aplicadas en cuanto al estudio de la atmósfera, por lo tanto, su desarrollo depende del progreso de estas ciencias. En el cual en base a estudios de (Ayllon, 1996), (Pelkowski, 2000) y (Seoanez, 2002), mencionan que desde un principio el ser humano quiso conocer las causas de los fenómenos meteorológicos y el estado del tiempo, para comprender y adaptarse en rotación (Aragón, 2014).

Actualmente, existen muchos estudios científicos de pronóstico para predecir las condiciones futuras de los fenómenos en diferentes campos entre los más destacados. Mediante el uso de la base de datos SCIELO en el cual en el estudio de Biología en la publicación titulada "Predicción de la temperatura superficial del mar frente a las costas del Perú mediante un modelo automático integrado de migración aguas arriba" (ARIMA), se refiere al evento entre el niño y su conexión global con: clima, ecosistema y aspectos socioeconómicos. Las predicciones realizadas entre 1980 y 2007 a través de modelos estadísticos y dinámicos no son suficientes, por lo que proponen explorar ARIMA a través de la identificación, estimación, verificación diagnóstica, predicción y validación del modelo real. Utilizaron funciones de autocorrelación simple y parcial (FAC y FACP) para definir y reformatear el orden de los parámetros (Pilco. V, Acuario.W 2019).

Así también, se han realizado diversos estudios para detectar posibles tendencias climáticas en todo el mundo y estos han reportado diferentes eventos que han provocado enormes consecuencias para la sociedad humana y para el medio ambiente natural, un ejemplo de ello es la temperatura extrema debido al cambio climático; como, la ola de calor de verano de 2003 sobre Europa y la ola de calor de Rusia (Mahmood, 2014, pp. 56-66).

Para, el caso ecuatoriano, el Instituto Nacional de Meteorología e Hidrología (INAMHI) evidenció que entre los años 1969 y 2006 la temperatura media anual aumentó en 0,8 °C, la temperatura máxima en 1,4 °C y la temperatura mínima 1°C. También entre 1969 y 2006 la precipitación anual promedio se incrementó un 33% en la costa y 8% en la sierra. Ocasionando un aumento de riesgos parasitarios y enfermedades transmitidas por mosquitos (Alexandra Marcatoma, 2020, p. 3).

PLANTEAMIENTO DEL PROBLEMA

La dinámica de las variables meteorológicas es difícil de comprender ya que hay una constante variación climática que impide tratar la información como un patrón definido y constante, el estudio propone el uso de tres metodologías de modelamiento como Box-Jenkins, Redes Neuronales y Teoría del Caos con el fin de incluir a la física dentro de la investigación y poder ajustar su comportamiento a sistemas dinámicos. El antecedente permite identificar el modelo más idóneo y con mejor ajuste para el cálculo de pronósticos que cooperen en la toma de decisiones del Grupo de Energía Alternativa y Ambiente “GEAA” de la Escuela Superior Politécnica de Chimborazo, al grupo de investigadores estudiantes del tema y a todas aquellas personas que busquen conectar los resultados de investigaciones académicas con el mundo real.

Formulación (Incógnita)

¿Cuál es la metodología que presenta un mejor ajuste en la predicción de variables meteorológicas inducidas por el Grupo de Energía Alternativa y Ambiente “GEAA” de la Escuela Superior Politécnica de Chimborazo?

JUSTIFICACIÓN

La predicción de series temporales meteorológicas es un problema complejo y desafiante que afecta a casi todos los sectores en la sociedad. Actualmente las variables meteorológicas son difíciles de comprender ya que presenta una constante variación climática lo cual nos impide tratar información con un patrón definido.

Existen varios métodos de modelación para fenómenos naturales basados en la climatología, pero muchos de estos presiden de supuestos teóricos que difícilmente se ajustan a la realidad, por ello gracias al avance tecnológico surgen nuevas metodologías como: Box-Jenkins, Redes Neuronales y Teoría del Caos lo cual nos va a permitir solucionar aquellos inconvenientes que van a existir en un futuro.

Uno de los acontecimientos más relevantes en la historia es la elaboración pronósticos meteorológicos, los cuales es la representación simplificada del tiempo atmosférico lo cual nos va a permitir contribuir de una manera explícita en la investigación de los fenómenos naturales basadas en las relaciones climatológicas.

El propósito de la investigación es que nos va a permitir identificar cuál de las tres metodologías antes mencionadas tiende a un mejor ajuste para el análisis de pronósticos de las variables meteorológicas y a la vez que cooperen en la toma de decisiones futuras del Grupo de Energías Alternativas y Ambiente (GEAA) de la Escuela Superior Politécnica de Chimborazo.

OBJETIVOS

Objetivo general

- Presentar un análisis estadístico para pronósticos meteorológicos en el aplicativo web con la finalidad de ayudar en la toma de decisión futura.

Objetivos específicos

- Depurar las variables meteorológicas provenientes de la base de datos de las estaciones Alao, Matus, ESPOCH, Tunshi y Multitud, en la Provincia de Chimborazo.
- Realizar un análisis estadístico para las variables meteorológicas y presentar la información.
- Determinar pronósticos para las variables meteorológicas provenientes de la base de datos de las estaciones Alao, Matus, ESPOCH, Tunshi y Multitud, en la Provincia de Chimborazo para determinar los patrones futuros atmosféricos.
- Presentar los resultados obtenidos de los pronósticos meteorológicas provenientes de la base de datos de las estaciones Alao, Matus, ESPOCH, Tunshi y Multitud, en la Provincia de Chimborazo a través del aplicativo web.

CAPÍTULO I

1. MARCO TEÓRICO REFERENCIAL

1.1. Bases conceptuales

1.1.1. *Meteorología*

La meteorología es un término que se remonta al filósofo griego Aristóteles quien, alrededor del 340 a.C., escribió un libro sobre filosofía natural titulado *Meteorológica*. Este trabajo representó la suma de conocimientos sobre el tiempo, el clima, astronomía, geografía y química de esa época. Algunos de los temas cubiertos incluyeron nubes, lluvia, nieve, viento, granizo, truenos y huracanes. En aquellos días, todas las sustancias que caían del cielo, y todo lo que se veía en el aire, recibían la denominación de meteoros, de ahí el término meteorología, que en realidad proviene de la palabra griega meteoros, que significa "alto en el aire", en otras palabras meteorología es el estudio de la atmósfera y sus fenómenos (Donald Ahrens, 2016, pp. 24-156).

Aristóteles intentó explicar los fenómenos atmosféricos de una manera filosófica y especulativa y a pesar de que se descubrió que muchas de sus especulaciones eran erróneas, las ideas de Aristóteles fueron aceptadas sin reservas durante casi dos mil años.

De hecho, el nacimiento de la meteorología como ciencia natural genuina no tuvo lugar hasta que se inventaron varios instrumentos meteorológicos, como el higrómetro a mediados del siglo XV, el termómetro a finales del siglo XVI y el barómetro a mediados del siglo XVII; cada uno de estos instrumentos y los que se desarrollaron en el siglo XIX dieron pie al progreso de la ciencia meteorológica (Donald Ahrens, 2016, pp. 27-112), (Shonk, 2020, pp. 1-67).

En 1843, la invención del telégrafo permitió la transmisión rutinaria de las observaciones del tiempo lo que causó una mayor comprensión del flujo del viento y el movimiento de las tormentas. Alrededor de 1920, los conceptos de masas de aire y frentes meteorológicos se formularon en Noruega. Además, en la década de 1950 la meteorología dio otro paso adelante cuando los científicos convirtieron las ecuaciones matemáticas que describen el comportamiento de la atmósfera en un software llamado modelos numéricos el cual podía ejecutarse en nuevas computadoras de alta velocidad. Estos cálculos fueron el comienzo de la predicción numérica del tiempo (Donald Ahrens, 2016, pp. 28-110), (Shonk, 2020, pp. 1-67).

En 1960, se lanzó el primer satélite meteorológico, Tiros I, que marcó el comienzo de la meteorología de la era espacial. Los satélites ofrecen una amplia gama de información, que

contempla datos desde el día hasta la noche con imágenes que van desde nubes y tormentas hasta imágenes que representan cintas arremolinadas de vapor de agua que fluyen alrededor del mundo (Donald Ahrens, 2016, pp. 31-100).

A lo largo de la década de 1990 y en el siglo XXI, se han desarrollado satélites aún más sofisticados que suministran a las computadoras una red de datos mucho mayor que permiten realizar pronósticos más precisos, hoy en día las computadoras trazan las observaciones, dibujan las líneas en el mapa y pronostican el estado de la atmósfera durante un tiempo deseado en el futuro (Donald Ahrens, 2016, pp. 35-101).

1.1.2. Pronósticos

Mucha gente depende de las previsiones meteorológicas pues una de las cosas más importantes que pueden hacer los pronósticos meteorológicos es anticipar si el tiempo va a ser peligroso o no para una correcta toma de decisiones, por ejemplo (Lyman, 2019, pp.1-20):

- Los agricultores necesitan saber si lloverá o si la temperatura estará por debajo del punto de congelación para el desarrollo de sus cultivos.
- Las personas encargadas de administrar aerolíneas deben tener informes meteorológicos muy completos que certifiquen la seguridad para realizar vuelos.
- Los dirigentes de empresas eléctricas deben prepararse para las tormentas y las olas de calor, etc. (Lyman, 2019, pp. 1-20).

1.1.3. Historias de los pronósticos

La gente ha intentado predecir el clima durante miles de años. Ya en 650 a. C., los babilonios utilizaron patrones de nubes y astronomía para hacer predicciones sobre el clima. En el año 300 a. C., los chinos crearon un calendario meteorológico que dividía el año en 24 festividades, y cada festividad estaba correlacionada con un tipo de clima diferente (Lyman, 2019, pp.1-20).

Pero la meteorología comenzó a cobrar confianza cuando se inventaron varias herramientas que ayudaron a aprender sobre la atmósfera; la gente creó el termómetro e inventó el barómetro y comenzaron a observar y registrar el clima de forma regular y fueron estas observaciones las que cimentaron ideas confiables sobre cómo se formaba el clima y cómo predecirlo (Lyman, 2019, pp.1-20).

A medida que más personas comenzaran a usar estas herramientas, y compartir sus observaciones y lograron predecir lo que sucedería en un lugar observando lo que sucedía en otro (Lyman, 2019, pp. 1-20).

El pronóstico del tiempo fue introducido por Robert Fitzroy en 1859, quien ideó un sistema de alerta de tormenta, este sistema ha mejorado enormemente desde los primeros intentos de Fitzroy y actualmente los pronósticos están basados en teorías físicas de la atmósfera, los océanos, la tierra y la energía del sol, y respaldadas por muchos millones de observaciones del tiempo actual e inmediato pasado (Jennifer Castle, David F. Hendry, 2019, pp.1-15).

A pesar de tener toda esta base científica y teórica la ciencia de pronosticar no es exacta pues existe mucha incertidumbre ya que con el clima nadie conoce el conjunto completo de posibilidades lo que hace que el investigador pronostique con un enfoque inteligente eventos inciertos de antemano (Jennifer Castle, David F. Hendry, 2019, pp.1-15).

1.1.4. Box Jenkins

En el análisis de series de tiempo, el método Box-Jenkins, lleva el nombre de los estadísticos George Box y Gwilym Jenkins, quienes publicaron un libro histórico sobre análisis y predicción de series de tiempo no estructurales. La contribución más importante de Box y Jenkins fue su amplia visión, articulación e ilustración de un marco operativo para la predicción no estructural aplicada, que consiste en ciclos iterativos de formulación de modelos, estimación, verificación de diagnóstico y predicción. Los modelos de media móvil autorregresiva (ARMA) o promedio móvil integrado autorregresivo (ARIMA) para encontrar el mejor ajuste de un modelo de series de tiempo a valores pasados de una serie de tiempo son la pieza central del marco de Box-Jenkins (Francis X. Diebold, 1999, pp. 1-100), (Terrell, 2019, pp. 7-200) .

Enfoque de Modelado

El enfoque de Box-Jenkins para el modelado de series de tiempo consiste en extraer movimientos (o patrones) predecibles de los datos observados a través de una serie de iteraciones de tres etapas (Aljandali, 2017, pp.1-175), (Terrell, 2019, pp. 9-115):

- **Identificación del modelo:** mediante el uso de varias estadísticas calculadas a partir de un análisis de los datos históricos se asegura que las variables sean estacionarias, identificando la estacionalidad en la serie dependiente y utilizando gráficos de la autocorrelación y funciones de autocorrelación parcial de la serie de tiempo dependiente para decidir cuál

componente de media móvil o autorregresiva se debe utilizar en el modelo (Terrell, 2019, pp. 9-115), (Aljandali, 2017, pp.1-160).

- **Estimación y verificación del modelo:** una vez identificado, el "mejor modelo" se estiman los parámetros mediante algoritmos de cálculo para llegar a los coeficientes que mejor se ajustan al modelo ARIMA seleccionado de manera que los valores ajustados se acerquen lo más posible a la captura del patrón exhibido por los datos reales. Los métodos más comunes utilizan la estimación de máxima verosimilitud o la estimación de mínimos cuadrados no lineales (Terrell, 2019, pp. 9-115), (Aljandali, 2017, pp.1-160).
- **Pronóstico y Verificación del modelo:** verificar si el modelo estimado se ajusta a las especificaciones de un proceso univariado estacionario. En particular, los residuos deben ser independientes entre sí y constantes en media y varianza a lo largo del tiempo. Si la estimación es inadecuada, se vuelve a la fase uno y se intenta construir un mejor modelo. El modelo final se utiliza para pronosticar la serie de tiempo y para desarrollar intervalos de confianza que miden la incertidumbre asociada con el pronóstico (Aljandali, 2017, pp.1-175), (Terrell, 2019, pp.1-200).

1.1.5. Identificación del modelo de Box Jenkins

Estacionariedad y estacionalidad: El primer paso para desarrollar un modelo de Box-Jenkins es determinar si la serie de tiempo es estacionaria y existe alguna estacionalidad significativa que necesita ser modelada (Terrell, 2019, pp.53-60).

Detección de estacionariedad: La estacionariedad se puede evaluar a partir de una gráfica de secuencia de ejecución. La gráfica de secuencia de ejecución debe mostrar la ubicación y escala constantes. También se puede detectar a partir de una gráfica de autocorrelación. En particular, la no estacionariedad es a menudo detectada a través de una gráfica de autocorrelación con un decaimiento muy lento (Terrell, 2019, pp.53-60).

Detección de la estacionalidad: La estacionalidad o periodicidad generalmente se puede evaluar a partir de una gráfica de autocorrelación, una gráfica de subserie estacional o un diagrama espectral (Terrell, 2019, pp.53-60).

Diferenciación para lograr la estacionariedad: Box y Jenkins recomiendan el enfoque de diferenciación para lograr la estacionariedad y en el mismo contexto, también se puede utilizar ajustar una curva y restar los valores ajustados de los datos originales (Terrell, 2019, pp. 53-60).

Diferenciación estacional: Para los modelos de Box-Jenkins, no se elimina explícitamente la estacionalidad antes de ajustar el modelo. En su lugar, se incluye el orden de los términos estacionales en la especificación del modelo para el software de estimación ARIMA. Sin embargo, puede ser útil aplicar una diferencia estacional a los datos y regenerar las gráficas de autocorrelación y autocorrelación parcial. Esto puede ayudar en la identificación del modelo del componente no estacional del modelo. En algunos casos, la diferenciación estacional puede eliminar la mayor parte o la totalidad del efecto de estacionalidad (Terrell, 2019, pp.53-60).

Identificar p y q: Una vez que se han abordado la estacionalidad y la estacionalidad, el siguiente paso es identificar el orden, es decir, la p y q de los términos autorregresivos y de media móvil. Hay varias maneras de abordar esta incógnita algunos autores usan el AICC o criterio de información de Akaike con corrección y otros autores utilizan la gráfica de autocorrelación y la gráfica de autocorrelación parcial (Terrell, 2019, pp.53-60), (Gilroy, 2018, pp. 16-111).

1.1.6. Estimación del modelo Box Jenkins

La estimación de los parámetros para los modelos de Box-Jenkins implica aproximar numéricamente las soluciones de ecuaciones no lineales. Los enfoques principales para ajustar los modelos de Box-Jenkins son los mínimos cuadrados no lineales y la estimación de máxima verosimilitud. La estimación de máxima verosimilitud es generalmente la técnica preferida. Hoy en día existen softwares estadísticos diseñados para realizar estos cálculos (Terrell, 2019, pp. 53-60).

1.1.7. Diagnóstico del modelo Box Jenkins

El diagnóstico del modelo para los modelos Box-Jenkins es similar a la validación del modelo para el ajuste por mínimos cuadrados no lineales. Es decir, se supone que el término de error sigue los supuestos para un proceso univariado estacionario. Los residuos deben ser independientes cuando sus distribuciones son normales con una media y varianza constantes. Una forma de evaluar si los residuos del modelo de Box-Jenkins siguen los supuestos es generar gráficos estadísticos de los residuos.

Si el modelo de Box-Jenkins es un buen modelo para los datos, los residuos deben satisfacer estos supuestos. Si no se cumplen estos supuestos, es necesario ajustar un modelo más apropiado. Es decir, volver a la fase de identificación del modelo e intentar desarrollar un modelo mejor (Terrell, 2019, pp. 53-60), (F., David, 2018, pp. 27-140).

1.1.8. Modelo matemático ARIMA

Un modelo con términos autorregresivos también puede combinarse con un modelo que tenga términos de promedio móvil para obtener un modelo autorregresivo de promedio móvil “mezclado”. Para representar estos modelos es conveniente usar la notación ARMA (p, q) donde p equivale al orden de la parte autorregresiva y q es el orden de la parte del promedio móvil. Un modelo ARMA (p, q) tiene la forma general

$$Y_t = \Phi_0 + \Phi_1 Y_{t-1} + \Phi_2 Y_{t-2} + \dots + \Phi_p Y_{t-p} + \varepsilon_t - \omega_1 \varepsilon_{t-1} - \omega_2 \varepsilon_{t-2} - \dots - \omega_q \varepsilon_{t-q} \quad (1-1)$$

Los modelos ARMA (p, q) pueden describir una amplia variedad de comportamientos para las series de tiempo estacionarias. Los pronósticos generados por un modelo ARMA (p, q) dependerán de valores actuales y anteriores de la respuesta Y , así como de los valores pasados y actuales de los errores (residuales) ε .

1.1.9. Redes Neuronales

Una red neuronal se caracteriza por la arquitectura de la red, las características de los nodos y las reglas de aprendizaje. Las redes neuronales pueden clasificarse en redes neuronales de retroalimentación, redes neuronales recurrentes y sus híbridos (Ke-Lin Du, 2019, pp. 7-20).

Las topologías de red más populares son:

- a) las redes de feedforward en capas completamente conectadas,
- b) las redes recurrentes,
- c) las redes de celosía,
- d) las redes de feedforward en capas con conexiones laterales y
- e) las redes celulares (Ke-Lin Du, 2019, pp. 7-20).

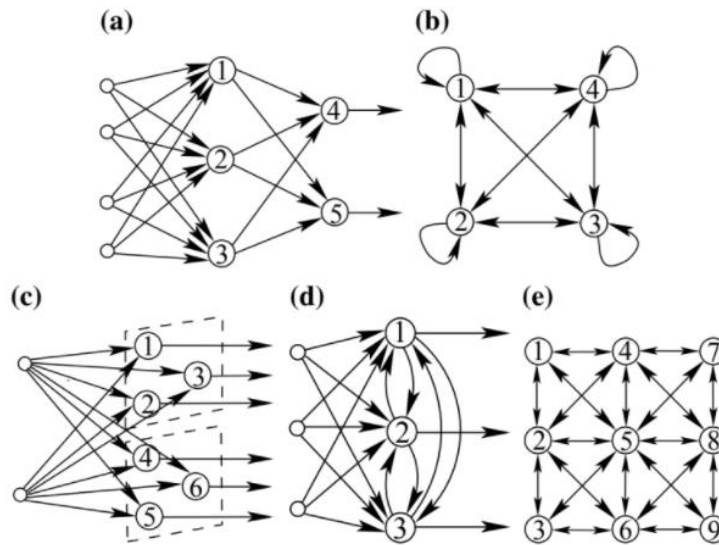


Figura 1-1: Redes Neuronales.

Fuente: Centro de Investigación CIMA.

Red feedforward: las conexiones entre neuronas van en una sola dirección, en una red de retroalimentación en capas de este tipo, no hay conexión entre las neuronas de la misma capa y no hay retroalimentación entre las capas (Ke-Lin Du, 2019, pp. 8-20).

Red recurrente: existe al menos una conexión de retroalimentación (Ke-Lin Du, 2019, pp. 8-20).

Red de celosía: consta de una matriz de neuronas de una, dos o más dimensiones. Cada matriz tiene un conjunto correspondiente de nodos de entrada (Ke-Lin Du, 2019, pp. 8-20).

Red de feedforward en capas con conexiones laterales: tiene conexiones laterales entre las unidades en la misma capa de su arquitectura de red de feedforward en capas. Las conexiones feedforward son excitadoras, mientras que las conexiones laterales en la misma capa son inhibitoras (Ke-Lin Du, 2019, pp. 7-20).

Red celular: consta de neuronas regularmente espaciadas, llamadas células, que se comunican solo con las neuronas en su vecindad inmediata, las celdas adyacentes están conectadas por interconexiones mutuas. Cada celda es excitada por sus propias señales y por señales que fluyen desde sus celdas adyacentes (Ke-Lin Du, 2019, pp. 7-20).

1.1.10. Funcionamiento de las redes neuronales

El funcionamiento de las redes neuronales se divide en dos etapas: aprendizaje y generalización. El entrenamiento de la red se logra típicamente usando ejemplos, y los parámetros de la red se adaptan usando un algoritmo de aprendizaje, en línea o fuera de línea. Una vez que la red está capacitada para lograr el desempeño deseado, el proceso de aprendizaje finaliza y luego se puede usar directamente para reemplazar la compleja dinámica del sistema (Ke-Lin Du, 2019, pp. 9-20), (Gurney, 1999, pp. 1-120).

Adicional cuentan con capas o niveles en el cual se distribuyen las neuronas y tiene un número específico de neuronas, distinguimos 3 tipos de clases de capas:

- Capa de entrada: es aquella que recibe información directamente de fuentes externas de una red
- Capa oculta: son capaces de tomar valores desde 0 hasta un número alto, estas capas se pueden conectar de diferentes formas al valor establecido, dando lugar a diferentes topologías de redes neuronales.
- Capa de salida: esta es la capa final, es aquella que nos arroja los resultados del procesamiento de la información (Ruiz et al., 2001, p. 16).

1.1.11. Propiedades de las redes neuronales

Una de las principales funciones del cerebro es la predicción. Las redes neuronales están motivadas biológicamente. Cada neurona es un nodo computacional, que representa una función no lineal. Las redes neuronales poseen las siguientes ventajas (Ke-Lin Du, 2019, pp. 9-20), (Ian Goodfellow, Yoshua Bengio, 2016, pp. 120-237):

- Aprendizaje adaptativo: pueden adaptarse cambiando los parámetros de la red en un entorno circundante.
- Generalización: una red neuronal entrenada tiene la capacidad de generalización superior.
- Naturaleza no lineal de uso general: funcionan como una caja negra.
- Autoorganización: algunas redes neuronales tienen una propiedad de autoorganización.
- Paralelismo masivo e implementaciones de VLSI simples: cada unidad de procesamiento básica generalmente tiene una propiedad uniforme. Esta estructura paralela permite implementaciones de software y hardware altamente paralelas (Ke-Lin Du, 2019, pp. 9-20).
- Robustez y tolerancia a fallas: una red neuronal puede manejar fácilmente información imprecisa, difusa, ruidosa y probabilística. Así, el desempeño general no se degrada significativamente cuando se pierde la información en algunos nodos o algunas conexiones

en la red están dañados. La red se repara sola y, por lo tanto, posee una capacidad de tolerancia a fallas (Ke-Lin Du, 2019, pp. 9-20).

1.1.12. Tipos de redes neuronales

Red Neuronal Acíclica

- Este tipo de red neuronal tiene dos tipos de conexiones predictivas y de retroalimentación entre neuronas, que son Esta conexión hace que la información sea transmitida y devuelta durante el funcionamiento de la red (Sepúlveda, 2018).
- Esta red permite implementar una determinada memoria, por lo que tienen sentido temporalmente. Para lograrlo, es necesario implementar neuronas que tomen el resultado de una de las capas de entrada y alimente su salida a una de las capas del nivel anterior (Díaz Kusztrich, 2018).

Redes recurrentes parciales

Se caracterizan porque en la capa de entrada existe un conjunto de neuronas conocidas como neuronas de contexto, neuronas de estado o neuronas accesorias, que actúan como neuronas receptoras de asociaciones repetitivas (Sepúlveda, 2011, p. 24), mediante el uso de estas redes se puede modelar las series temporales y obtener resultados adecuados (Díaz Kusztrich, 2018).

Neuronas de entrada a la red:

- Estas neuronas actúan como entradas al recibir señales del exterior.
- Las neuronas de contexto actúan como la memoria de la red donde se recopilan las tareas de las neuronas de una determinada capa de la red en un momento anterior o inmediato.

Las cuales se crean utilizando una topología de avance, en la que se tiene en cuenta la retroalimentación de las capas de salida u ocultas a las capas de entrada. En este tipo de redes se tienen en cuenta las redes de Elman y Jordan y variaciones de estas capas (Sepúlveda, 2018).

Red de Elman

También conocida como red iterativa simple, considero que los retornos de clases ocultas pasan a la capa de contexto y no considero retornos locales (Sepúlveda, 2018, pág 26).

Se dice que la entrada de estas neuronas se toma de la salida de las neuronas de una de las capas ocultas, y su salida está inversamente relacionada con las entradas de esta misma capa, lo que proporciona una especie de memoria encima de la capa oculta y el estado anterior de dicha clase (DÍAZ KUSZTRICH, 2018) p. 1.

Red de jordán

En esta red la entrada de neuronas de la capa de contexto se toma de la salida de la red (Diaz, 2016, p. primero). Su diseño en la capa de entrada se divide en dos partes: el conjunto de entradas externas y la capa de salida que permite la retroalimentación a través de conexiones de valor fijo (Sepúlveda, 2018, pp. 25-26).

1.1.13. Modelo matemático de la red neuronal

Las entradas x e y son el estímulo que la neurona artificial recibe del entorno que la rodea, y la salida z es la respuesta a tal estímulo. La neurona se adapta al medio circundante y aprende de él modificando el valor de sus pesos sinápticos w_1 y w_2 y su término aditivo b . Estos son conocidos como los parámetros libres del modelo, pues los mismos pueden ser modificados y adaptados para realizar una tarea determinada. En este modelo, la salida neuronal z está dada por

$$z = f(w_1x + w_2y + b) \quad (2-1)$$

La función de activación f es seleccionada de acuerdo a la tarea realizada por la neurona.

1.1.14. Aplicaciones

Las redes neuronales pueden tratarse como una herramienta estadística general para casi todas las disciplinas de la ciencia y la ingeniería. Las aplicaciones pueden ser en modelado e identificación de sistemas, clasificación, reconocimiento de patrones, optimización, control, aplicaciones industriales, comunicaciones, procesamiento de señales, análisis de imágenes, bioinformática y minería de datos (Ke-Lin Du, 2019, pp. 9-20).

Redes neuronales y computadoras digitales.

Para entender el potencial de la computación neuronal, sería necesario hacer una breve distinción entre sistemas de computación neuronales y digitales: los sistemas neurológicos no aplican

principios de circuitos lógicos o digitales. Un sistema de computación digital debe ser síncrono o asíncrono. Si fuera asíncrono, la duración de los impulsos neuronales debería ser variable para mantener uno de los valores binarios por periodos de tiempo indefinido, lo cual no es el caso. Si el principio fuera síncrono, se necesitaría un reloj global o maestro con el cual los pulsos estén sincronizados.

1.1.15. Teoría del Caos

Las matemáticas contienen un campo de estudio importante bajo el nombre de teoría del caos. La teoría del caos estudia el concepto y el comportamiento de sistemas dinámicos altamente insensibles y también estudia el comportamiento de los sistemas dinámicos en condiciones iniciales súper sensibles y altas. En la teoría del Caos, este concepto se conoce como efecto mariposa, el cual es el principal campo de estudio de esta teoría. Edward Lorenz trató de conceptualizar la teoría del caos mencionado que "El presente puede determinar el futuro, pero el presente aproximado no puede determinar el futuro aproximado" (IntroBooks, 2016, pp. 1-50), (Cambel, 1993, pp. 1-93).

Muchos sistemas naturales, como el tiempo, el clima, etc., poseen el mismo comportamiento caótico que se le atribuye a la teoría del caos es así como a través de un modelo matemático se pretende analizar y comprender la conducta de los sistemas dinámicos naturales y artificiales. Dicho proceso de análisis cuenta con la implementación de técnicas de análisis como gráficos de recurrencia y mapas de Poincaré (IntroBooks, 2016, pp. 1-50), (Zeraoulia, 2012, pp. 34-56).

Generalmente, caos significa estado de desorden, pero dejando de lado el concepto general, la teoría del Caos define el término "Caos" de manera más descriptiva utilizando la definición dada por Robert L. Devaney quien menciona que para que un sistema sea considerado como caótico debe cumplir con las siguientes condiciones (IntroBooks, 2016, pp. 1-50):

- Debe responder con sensibilidad en varias condiciones iniciales
- Debe actuar como sistema mixto de acuerdo con los conceptos de topología
- Debe poseer órbitas periódicas con densidad notable

Dentro de la teoría del caos los sistemas deterministas son uno de los temas principales a tratar debido a que su comportamiento está dentro del alcance de los principios y conceptos relacionados a la teoría. En un inicio, estos sistemas caóticos tienden a ser predecibles, pero luego de un tiempo resultan ser aleatorios lo que significa que los sistemas caóticos son predecibles

hasta un determinado período de tiempo desde su generación y esta cantidad de tiempo se puede derivar de tres aspectos (IntroBooks, 2016, pp. 1-50):

1. Nivel de incertidumbre esperado en el pronóstico o en la predicción
2. Precisión en la medición del tiempo de o en la última etapa disponible
3. Tiempo de Lyapunov, que es la escala totalmente dependiente de la dinámica del sistema. Algunos ejemplos del tiempo de Lyapunov son los circuitos eléctricos caóticos de casi 1 milisegundo, el sistema meteorológico y el sistema solar aproximadamente 50 millones de años (IntroBooks, 2016, pp. 1-50).

En sistemas caóticos, la incertidumbre del pronóstico aumenta exponencialmente y cuanto más pasa el tiempo más se incrementa la incertidumbre. Es así como, de acuerdo con la perspectiva matemática, si el tiempo de pronóstico se duplica, la incertidumbre aumentará más que el valor cuadrado de su nivel actual. Con esto en mente, la teoría del caos dice que es difícil lograr una predicción precisa si se hace después de un intervalo de tiempo (IntroBooks, 2016, pp. 1-50), (Straub, 1997, pp. 1-97).

Dentro de los campos y disciplinas en los que es aplicable la teoría del caos se encuentran:

- Meteorología
- Sociología
- Física
- Ciencias ambientales
- Ciencias de la computación
- Ingeniería
- Economía
- Biología
- Ecología
- Filosofía

La teoría del caos ha sido aplicada con éxito en todas las áreas antes mencionadas y actualmente hay muchos otros campos en los que aún se están investigando su aplicación (IntroBooks, 2016, pp. 1-50).

1.1.16. Diseño de aplicaciones web

El diseño de la interfaz de usuario (UID) para páginas web especifica lo que los usuarios necesitan en términos de apariencia de un sistema de software y lo que los ingenieros de software entienden según los requisitos del usuario. Este es un proceso abrumador que busca planificar un sistema interactivo, que sea atractivo, accesible y fácil de usar (Shamat et al., 2017, pp. 57-61), (Vemula, 2017, pp. 32-179).

UID es el diseño de interfaces que los usuarios pueden ver desde cualquier dispositivo, como computadoras y tablets funcionando como intermediario para que el usuario interactúe con el sistema. Un buen UID cuenta con una combinación de mecanismos de entrada y salida bien diseñados con buenas capacidades y condiciones que liberen de dudas y satisfagan las necesidades de los usuarios (Shamat et al., 2017, pp- 57-61).

Esto requiere una aplicación que permita una interacción instintiva y normal. Según Ginige y Murugesan las aplicaciones basadas en Web se pueden clasificar en siete clases (Shamat et al., 2017, pp. 57-61):

- 1. Informativo** como periódicos en línea, catálogos de productos, boletines informativos, manuales de servicio, clasificados en línea, libros electrónicos en línea, etc (Shamat et al., 2017, pp. 57-61).
- 2. Interactivo** como los formularios de registro, presentación de información personalizada, juegos en línea, etc (Shamat et al., 2017, pp. 57-61).
- 3. Transaccional** como las compras electrónicas, pedidos de bienes y servicios y banca en línea (Shamat et al., 2017, pp. 57-61).
- 4. Flujo de trabajo** presente en los sistemas de planificación y programación en línea, gestión de inventario y monitoreo de estado (Shamat et al., 2017, pp. 57-61).
- 5. Entorno de trabajo colaborativo** como los sistemas de autor distribuidos, herramientas de diseño colaborativo (Shamat et al., 2017, pp. 57-61).
- 6. Comunidades online, mercados** los cuales se encuentran en los grupos de chat, sistema de recomendación que recomienda productos o servicios, mercados en línea, subastas en línea (Shamat et al., 2017, pp. 57-61).

7. Portales web como los centros comerciales electrónicos, intermediarios online (Shamat et al., 2017, pp. 57-61).

A pesar de que una aplicación determinada puede pertenecer a más de una clase el UID debe enfocarse en realizar una aplicación web independientemente de las categorías centrándose en su funcionalidad de tal forma que brinden al usuario la oportunidad de completar las tareas con menos esfuerzo y tiempo (Shamat et al., 2017, pp. 57-61), (Rojas, 2020, pp. 1-200)

1.2. Bases teóricas

1.2.1. Meteorología y climatología

Son las dos ramas científicas esenciales en el estudio del clima: la Física de la Atmósfera o Meteorología estudia las leyes que gobiernan el funcionamiento del Sistema Climático y la Climatología estudia la localización y distribución espacial de las variables del clima y su relación con la vegetación, los cultivos o el hombre: fitoclimatología, agroclimatología y bioclimatología, son las tres principales ramas (Fernández, 2012).

Ambas están íntimamente correlacionadas y sus límites son difíciles de establecer, aunque de forma simplificada podemos decir que el pronóstico del tiempo es el objeto fundamental de la meteorología: el meteorólogo es el hombre del tiempo; el climatólogo, por su parte, define y caracteriza los diferentes tipos de climas que se observan en la superficie terrestre. (GARCÍA, 2012).

1.2.2. Clima

El clima se define a partir de las características que una serie de variables físicas como la temperatura, la humedad, la presión o el viento presentan sobre un lugar determinado (Fernández, 2012).

El conocimiento y comprensión del clima exige tener muy claras, cuatro ideas fundamentales: la diferenciación entre tiempo y clima, los factores condicionantes, los elementos y la escala (LÓPEZ, 2018).

1.2.3. *Tiempo*

El tiempo es la situación meteorológica concreta que se observa sobre un lugar en un momento determinado: frío o cálido, lluvioso o soleado, son adjetivos normalmente utilizados para definir el tiempo (Fernández, 2012).

1.2.4. *Factores del clima*

Los factores del clima son agentes como altitud, latitud, vientos tormentosos, corriente marina, distancia del mar y relieve los cuales modifican, acentúan o limitan los elementos del clima y dan lugar a los distintos tipos de climas.

Se agrupa bajo esta denominación al conjunto de rasgos o características geofísicas y geográficas de las que dependen los mecanismos fundamentales del clima de una zona concreta. Son de dos tipos: cósmicos o astronómicos y geográficos (GARCÍA, 2012).

1.2.5. *Elementos del clima*

En el que interfiere en cuanto a la medición de la temperatura, viento, presión atmosférica, etc. (Navarra, 2019). La principal característica es su gran variabilidad temporal y espacial (GARCÍA, 2012). A escala temporal, las temperaturas presentan un ciclo a lo largo de 24 horas; los regímenes térmicos y pluviométricos indican el ritmo mensual de las precipitaciones y temperaturas y la sucesión de años secos o lluviosos, fríos o cálidos (Fernández, 2012).

Espacialmente, los elementos del clima varían considerablemente de unos lugares a otros debido a la acción de los factores geográficos y cuya consecuencia es la gran variedad de tipos de climas que se observan en la superficie terrestre. (GARCÍA, 2012)

1.2.6. *Estación meteorológica*

Una Estación Meteorológica es una instalación destinada a medir y registrar regularmente variables meteorológicas mediante los instrumentos meteorológicos (Gonzalez, 2013).

1.2.7. *Variable meteorológica*

Es toda propiedad con condición de la atmósfera, cuyo conjunto define el estado del tiempo (a corto plazo) o del clima (a largo plazo). Se conoce también como parámetro meteorológico (Gonzalez, 2013).

1.2.8. *Temperatura*

Es la cantidad de energía calorífica que posee el aire en un momento determinado. Se mide mediante termómetros, habitualmente en grados Celsius (°C) y determina las sensaciones de calor y frío (Navarra, 2019).

1.2.9. *Velocidad del viento*

Corresponde a la relación que existe entre una cierta distancia y el tiempo que se tarda en recorrerla. Es el movimiento del aire en la atmósfera, que se desplaza desde las zonas de altas presiones a las de bajas presiones. Aunque este movimiento tiene lugar en las tres dimensiones del espacio, en meteorología se mide sólo la velocidad y dirección de su componente en el plano horizontal (Navarra, 2019).

1.2.10. *Humedad*

Cantidad de agua, vapor de agua o cualquier otro líquido que está presente en la superficie o el interior de un cuerpo o en el aire. (Navarra, 2019)

1.2.11. *Pronóstico meteorológico*

Consiste en predecir en un futuro el estado de la atmósfera usando modelos meteorológicos.

Tipos de pronóstico

Horizontal

- **Corto Plazo:** Se verifica mensualmente o menos, el período de planificación es de un año, se usa comúnmente para el suministro, la producción, la asignación de mano de obra y la planificación de pronósticos del departamento de producción.
- **Mediano Plazo:** Tiene un período de validez de seis meses a tres años, se utiliza para estimar planes de ventas, producción, flujo de caja y presupuesto.
- **Largo Plazo:** Se utiliza para planificar el lanzamiento de nuevos productos, nuevas inversiones, permite el desarrollo de proyectos, su duración estimada es de tres años o más.
- **Duración del período del pronóstico**
 - a. Apartar un grupo de datos según la serie antes de la modelación.

- b. Se grafica los datos reales de la serie de tiempo vs datos pronosticados.
- c. Para verificar el análisis de errores se utiliza un método de comparación.
- d. Encontrar un modelo que se ajuste a la serie de tiempo original que contenga el mínimo error en las estimaciones.
- e. Delimitar el número de datos a predecir indicando el error.
- f. Entregar las principales indicaciones en cuanto la factibilidad del modelo al momento de hacer pronósticos.

Datos: Los datos consisten en información proveniente de observaciones, recuentos, mediciones, o respuestas(Larson & Farber, 2015, pp. 1-100).

1.2.12. Métodos Estadísticos

Son aquellos procedimientos que nos permiten analizar datos cuantitativos y cualitativos mediante técnicas de recolección. Los procedimientos estadísticos permiten verificar las hipótesis y determinar las relaciones en un determinado fenómeno.

1.2.13. Análisis exploratorio de Datos

El cual ayuda en obtener de manera sencilla las principales características de un análisis en el que se puede obtener algunos fallos que pueden presentar en el diseño y recolección de datos, tratamiento y evaluación de datos ausentes, identificación de casos atípicos y comprobación de los supuestos subyacentes en la mayor parte de las técnicas multivariantes; la finalidad del AED es examinar los datos previamente a la aplicación de cualquier técnica estadística.

1.2.14. Medidas de Tendencia Central

Son indicadores estadísticos que muestran hacia qué valor (o valores) se agrupan los datos.

Media Aritmética

Es el valor resultante que se obtiene al dividir la sumatoria de un conjunto de datos sobre el número total de datos. Solo es aplicable para el tratamiento de datos cuantitativos, su fórmula es:

$$\bar{x} = \frac{x_1 + x_2 + x_3 + \dots + x_n}{N} \quad (3-1)$$

Mediana

Valor que divide una serie de datos en dos partes iguales. La cantidad de datos que queda por debajo y por arriba de la mediana son iguales, a pesar de existir fórmulas

$$Me = L_i + \frac{\frac{N}{2} - F_{i-1}}{f_i} \cdot a_i \quad (4-1)$$

Moda

Indica el valor que más se repite, o la clase que posee mayor frecuencia.

1.2.15. Medidas de Dispersión

También llamadas medidas de variabilidad muestran la variabilidad de una distribución, indicando por medio de un número si las diferentes puntuaciones de una variable están muy alejadas de la media. Cuanto mayor sea ese valor, mayor será la variabilidad, y cuanto menor sea, más homogénea será a la media. Así se sabe si todos los casos son parecidos o varían mucho entre ellos. Las medidas de dispersión nos informan sobre cuánto se alejan del centro los valores de la distribución.

Desviación estándar

La desviación estándar mide la dispersión de una distribución de datos. Entre más dispersa está una distribución de datos, más grande es su desviación estándar; la desviación estándar no puede ser negativa. Una desviación estándar cercana a 000 indica que los datos tienden a estar más cerca a la media (se muestra por la línea punteada). Entre más lejos estén los datos de la media, más grande es la desviación estándar, su fórmula es:

$$s = \sqrt{\frac{(X_1 - \bar{X})^2 + (X_2 - \bar{X})^2 + \dots + (X_n - \bar{X})^2}{n - 1}} \quad (5-1)$$

Rango

El rango es la diferencia entre el valor máximo y valor mínimo de la variable, la fórmula de cálculo es:

$$R = X_{m\acute{a}x} - X_{m\acute{i}n} \quad (6-1)$$

1.2.16. Serie de tiempo

Es una serie de observaciones, medidas en momentos determinados, dispuestas en un orden cronológico uniforme, en el cual los datos usualmente son dependientes entre sí. El principal objetivo de una serie de tiempo es su análisis para hacer pronóstico. (Villavicencio, 2016).

Componentes de una serie temporal

- **Tendencia:** Es una trayectoria con un movimiento suave a largo plazo de los datos en estudio es decir indica el decrecimiento o crecimiento de la serie.
- **Estacionalidad:** Es un patrón que se asemeja a ciclos, son aquellos cambios que se repiten año tras año, consisten en una característica de la duración de picos sucesivos: cíclicamente es mayor a un año y estacionalmente menor a un año, por lo que son de corta duración (Hanke & WICHERM, 2010).
- **Aleatoriedad:** Esta componente no responde a ningún patrón de comportamiento, sino que es el resultado de factores fortuitos o aleatorios que inciden de forma aislada en una serie de tiempo (Villavicencio, 2016).

Clasificación descriptiva de las series temporales

- **Estacionarias:** Presenta la estabilidad de una serie de tiempo a largo plazo en cuanto a su media y varianza permanecen constantes. La media y la variabilidad es constante y gráficamente los valores tienden a oscilar en la serie permaneciendo constante en el tiempo.
- **No estacionarias:** Estas son secuencias que tienden y es probable que cambien con el tiempo. Las variaciones en la media determinan una tendencia alta o baja a largo plazo, por lo que la serie no oscila alrededor de un valor constante (Villavicencio, 2016)

1.2.17. Coeficiente de Determinación R^2 Ajustado

Se define como un estadístico usado en el contexto de un modelo cuyo principal propósito es predecir futuros resultados o probar una hipótesis. Donde la proporción de la varianza de la variable explicada por las variables independientes en estadística descriptiva presenta dos propiedades iniciales de R^2 (Vélez, 2016).

$$R^2 = \frac{SCE}{SCT} = 1 - \frac{SCR}{SCT} \quad (7-1)$$

Dónde:

SCE es la suma de cuadrados explicados,

SCR es la suma de cuadrados de los residuos

SCT es la suma de cuadrados total.

Las desventajas son:

- Evalúa bien la bondad dentro del modelo, más no da seguridad que lo haga fuera.
- La variable endógena debe ser la misma en cuanto si se comparan dos o más variables.
- El coeficiente de determinación en cuanto al aumentar variables no se reduce (Vélez, 2016).

1.2.18. Neurona

Al igual que una neurona forma el elemento básico en el cerebro, una neurona forma la estructura básica de una red neuronal, una neurona recibe una entrada, la procesa y genera una salida que se envía a otras neuronas para su procesamiento posterior o la salida final (Ryszard Tadeusiewicz, Rituparna Chaki, 2015, pp. 20-115).

1.2.19. Inteligencia artificial

Comprende la teoría y el desarrollo de sistemas informáticos capaces de realizar tareas que normalmente requieren inteligencia humana, como la percepción visual, el reconocimiento de voz, la toma de decisiones y la traducción entre idiomas (Basilio de Braganca Pereira, Calyampudi Radhakrishna Rao, 2020, pp. 1-156).

1.2.20. Usuario

Un individuo que usa u opera algo, especialmente una computadora u otra máquina (Basilio de Braganca Pereira, Calyampudi Radhakrishna Rao, 2020, pp. 1-156).

1.2.21. Dispositivo

Una cosa hecha o adaptada para un propósito particular, especialmente una pieza de equipo mecánico o electrónico (Shamat et al., 2017, pp. 57-61) .

1.2.22. Aplicativo Web

Es un software de aplicación que se ejecuta en un servidor web, a diferencia de los programas de software basados en computadora que se ejecutan localmente en el sistema operativo (SO) del dispositivo. El usuario accede a las aplicaciones web a través de un navegador web con una conexión de red activa. Estas aplicaciones se programan utilizando una estructura modelada cliente-servidor: el usuario ("cliente") recibe servicios a través de un servidor externo alojado por un tercero (Shamat et al., 2017, pp. 57-61).

CAPÍTULO II

2. MARCO METODOLÓGICO

2.1. Tipo de la investigación

El estudio según el propósito fue aplicada científica ya que se midió variables meteorológicas específicas para pronosticar su comportamiento; según la profundidad del objeto de estudio la investigación fue exploratoria ya que el objetivo fue investigar y analizar información específica que no ha sido profundamente estudiada; de acuerdo con los datos empleados fue cuantitativa pues se utilizó un enfoque estadístico y matemático para el análisis de una gran cantidad de información recolectada con instrumentos de precisión; en relación al periodo de tiempo transversal ya que se analizó en un periodo determinado de tres años en cuanto al método de estudio inductivo ya que se modeló variables como la temperatura, velocidad de viento, presión atmosférica, humedad, y radiación solar con el fin de formular predicciones de mejor ajuste.

2.2. Diseño de la investigación.

La investigación fue diseñada de forma no experimental debido a la manipulación de variables en su ambiente natural, es decir, que no hay participación directa del investigador en la recolección de datos, las situaciones se muestran naturales y no son provocadas de forma intensional. (Hernández et al., 2010, pp. 151-160).

2.2.1. *Localización de estudio*

La investigación analizó información de las estaciones meteorológicas ubicadas en Alao, Matus, Espoch, Tunshi y Multitud, en la Provincia de Chimborazo durante el periodo 2012 – 2021.

2.2.2. *Población de estudio*

Los datos recolectados se captaron de los distintos ordenadores ubicados en cada una de las estaciones meteorológicas monitoreadas por el GEAA.

2.2.3. Método de muestreo

El método de muestreo utilizado por el GEAA fue un muestreo no probabilístico considerando puntos estratégicos para la toma de información ambiental lo que responde a una técnica de muestreo por conveniencia.

2.2.4. Tamaño de la muestra

Al no contar con un colectivo finito de información se utilizó el panel de captados de las estaciones Alao, Matus, ESPOCH, Tunshi y Multitud, en la Provincia de Chimborazo durante el periodo 2012 – 2021.

2.2.5. Técnica de recolección de datos.

El procedimiento manejado por el GEAA contó con la lectura de datos mediante ordenadores ubicados en las estaciones meteorológicas de Alao, Matus, ESPOCH, Tunshi y Multitud, en la Provincia de Chimborazo durante el periodo 2012 – 2021.

2.2.6. Identificación de variables

La investigación analizó variables meteorológicas como temperatura, velocidad de viento, presión atmosférica, humedad, y radiación solar en torno a la importancia de conocer y predecir el comportamiento de las variables que contribuyen a proyectos de investigación del GEAA.

2.2.7. Modelo estadístico

La investigación analizó 358588 datos cuantitativos, a los cuales se aplicó un análisis exploratorio de datos que incluía medidas de tendencia central, dispersión y porcentaje de datos faltantes, seguido se realizó un relleno de información con la aplicación de regresión lineal, seguido se aplicaron técnicas estadísticas para el cálculo de pronósticos como: Modelos ARIMA, Redes Neuronales y Teoría del Caos, la aplicación de cada técnica se desarrolló en R-Studio y los criterios de evaluación como U de Theil y el test de Diebold-Mariano permitieron escoger el mejor modelo de pronóstico; por último se diseñó un aplicativo web para socializar los resultados.

2.3. Variables en estudio.

2.3.1. Operacionalización de variables.

Tabla 1-2: Operacionalización de variables

Nombre de la variable	Descripción	Tipo de variable	Escala de medición
Temperatura	Es una magnitud escalar que sistematiza la cantidad de calor relacionando la energía interna de un sistema, de un cuerpo, de un objeto o del ambiente.	Cuantitativa Continua	Intervalo
Velocidad de viento	Magnitud que mide el componente horizontal del desplazamiento del aire en un punto y en un instante determinados	Cuantitativa Continua	Razón
Presión atmosférica	Mide el peso de la masa de aire que está actuando sobre la tierra.	Cuantitativa Continua	Razón
Humedad	Factor climatológico que se define como vapor de agua contenido en la atmósfera.	Cuantitativa Continua	Razón
Radiación solar	Es la energía emitida por el Sol, que se propaga en todas las direcciones a través del espacio mediante ondas electromagnéticas.	Cuantitativa Continua	Razón

Fuente: Paucar Belén, 2022

Realizado por: Paucar Belén, 2022.

CAPÍTULO III

3. MARCO DE RESULTADOS Y DISCUSIÓN DE RESULTADOS

3.1. Análisis exploratorio de información del GEAA

En este trabajo se estudió información meteorológica tomada por el GEAA de la ESPOCH en el periodo 2013 – 2021, con la finalidad de realizar pronósticos meteorológicos y presentar estos resultados en un aplicativo web, para ayudar a la toma de decisiones.

Para cumplir con este objetivo se presenta a continuación, en primer lugar, la depuración de los datos de variables meteorológicas provenientes de la base de datos de las estaciones Alao, Matus, Espoch, Tunshi y Multitud, en la Provincia de Chimborazo. En segundo lugar, se muestra un análisis exploratorio de datos, para estas variables meteorológicas. En tercer lugar, con la finalidad de obtener pronósticos adecuados de variables meteorológicas se muestran los resultados de la aplicación de los modelos Box Jenkins, Teoría del caos y Redes Neuronales, además, se presenta una comparación entre los ajustes de los diferentes métodos de pronóstico. En cuarto lugar, se presenta el aplicativo web, el cual muestra los pronósticos meteorológicos provenientes de la base de datos de las estaciones Alao, Matus, Espoch, Tunshi y Multitud, en la Provincia de Chimborazo. Es importante indicar que todo el análisis, se desarrolló con la ayuda del software estadístico R.

3.1.1. *Depuración de datos meteorológicos*

En el proceso de depuración de los datos meteorológicos se identificó las variables de interés de la información recogida por el GEA que son:

- fh: Fecha y hora.
- tap: Temperatura Ambiente Promedio.
- hrp: Humedad Relativa Promedio.
- pap: Presión Atmosférica Promedio.
- rsdp: Radiación Solar Difusa Promedio.
- vvp: Velocidad de Viento Promedio.

Luego, se organizó la información de todas las estaciones para que tengan la siguiente apariencia (este es un ejemplo para la estación de Alao) cuando se ejecuta la función *str* en R.:

```

> str(data.alao)
'data.frame': 68861 obs. of 6 variables:
 $ fh : POSIXct, format: "2014-01-28 19:00:00" "2014-01-28 20:00:00" "2014-01-28 21:00:00"
...
 $ ta : num 20.2 20.7 17.6 14.7 14 ...
 $ h : num 35.9 33.6 53 64.2 67 ...
 $ pa : num 705 705 705 706 706 ...
 $ rsd: num 244 251.4 137.2 78.1 24 ...
 $ vv : num 0 4.25 3.711 2.703 0.719 ...

```

En seguida, se identificó los datos faltantes en la información de las estaciones meteorológicas un resumen se presenta a continuación en la Tabla 3-1, donde se muestra que en general los porcentajes de datos faltantes son inferiores o iguales a 16.80%, por lo tanto, es factible realizar una imputación de la información mediante el método de regresión lineal, considerando como variable regresora el orden de toma de información.

Tabla 1-3: Valores faltantes según estación meteorológica y periodo

Estación Meteorológica	Periodo	Total de datos	Cantidad de Valores Faltantes	Porcentaje de Valores Faltantes
Alao	2014 - 2021	68861	4479	6.50%
ESPOCH	2015 - 2021	60960	2398	3.93%
Matus	2014 - 2021	61633	3028	4.91%
Multitud	2014 - 2021	69321	7028	10.14%
Tunshi	2013 - 2021	69244	11636	16.80%
TOTAL		330019	28569	8.66%

Fuente: Paucar Belén, 2022

Realizado por: Paucar Belén, 2022.

Posteriormente se realizó la imputación de la información, mediante el método de regresión lineal, donde los valores de R-cuadrado en general superaron el valor de 50%, por lo tanto, las imputaciones de los valores faltantes son aceptables.

3.1.2. Análisis de datos

A continuación, se muestra los resultados del análisis exploratorio de datos meteorológicos según estaciones con información imputada y sin información imputada, para verificar si la imputación de la información es adecuada (ver Gráfico 3.1 a 3.10). Además, se estudia varios estadísticos importantes de las variables de interés Fecha y hora, Temperatura Ambiente Promedio, Humedad Relativa Promedio, Presión Atmosférica Promedio, Radiación Solar Difusa Promedio y Velocidad de Viento Promedio (ver Tabla 3.2 a 3.6).

En la comparación de información imputada y sin imputar, se observa que las distribuciones de las variables de interés son semejantes, esto indica que la información imputada no agrede sesgos a los datos originales, por lo tanto, la imputación es satisfactoria y permite seguir con el análisis. Sin embargo, es importante indicar que existen posibles datos atípicos, estos se deben estudiar más a fondo, para decidir el tratamiento adecuado para ellos, esto, está fuera del alcance de esta investigación.

Por otro lado, el resumen estadístico de la información imputada muestra que:

- La estación Alao cuenta con información desde las 19:00:00 del 2014-01-28 hasta las 23:00:00 del 2021-12-06. En términos generales, las variables Temperatura Ambiente Promedio y Presión Atmosférica Promedio tienen una distribución con tendencia a la simetría, con forma aproximadamente mesocúrtica y una desviación estándar moderada. Las variables Humedad Relativa Promedio, Radiación Solar Difusa Promedio y Velocidad de Viento Promedio tienen una distribución asimétrica con forma leptocúrtica y con una desviación estándar moderada.
- La estación ESPOCH cuenta con información desde las 00:00:00 del 2015-01-01 hasta las 23:00:00 del 2021-12-14. En términos generales, las variables Temperatura Ambiente Promedio, Humedad Relativa Promedio y Presión Atmosférica Promedio tienen una distribución con tendencia a la simetría, con forma aproximadamente mesocúrtica y una desviación estándar moderada. Las variables, Radiación Solar Difusa Promedio y Velocidad de Viento Promedio tienen una distribución asimétrica con forma leptocúrtica y con una desviación estándar moderada.
- La estación Matus cuenta con información desde las 23:00:00 del 2014-11-27 hasta las 23:00:00 del 2021-12-08. En términos generales, la variable Presión Atmosférica Promedio tiene una distribución con tendencia a la simetría, con forma aproximadamente mesocúrtica y una desviación estándar moderada. Las variables Temperatura Ambiente Promedio, Humedad Relativa Promedio, Radiación Solar Difusa Promedio y Velocidad de Viento Promedio tienen una distribución asimétrica con forma leptocúrtica y con una desviación estándar moderada. Este mismo panorama se observa en la estación de Multitud, donde se cuenta con información desde las 15:00:00 del 2014-01-24 hasta las 23:00:00 del 2021-12-21.
- La estación Tunshi cuenta con información desde las 20:00:00 del 2013-12-18 hasta las 23:00:00 del 2021-11-11. En términos generales, las variables Temperatura Ambiente

Promedio y Humedad Relativa Promedio tiene una distribución con tendencia a la simetría, con forma aproximadamente mesocúrtica y una desviación estándar moderada. Las variables, Presión Atmosférica Promedio, Radiación Solar Difusa Promedio y Velocidad de Viento Promedio tienen una distribución asimétrica con forma leptocúrtica y con una desviación estándar moderada.

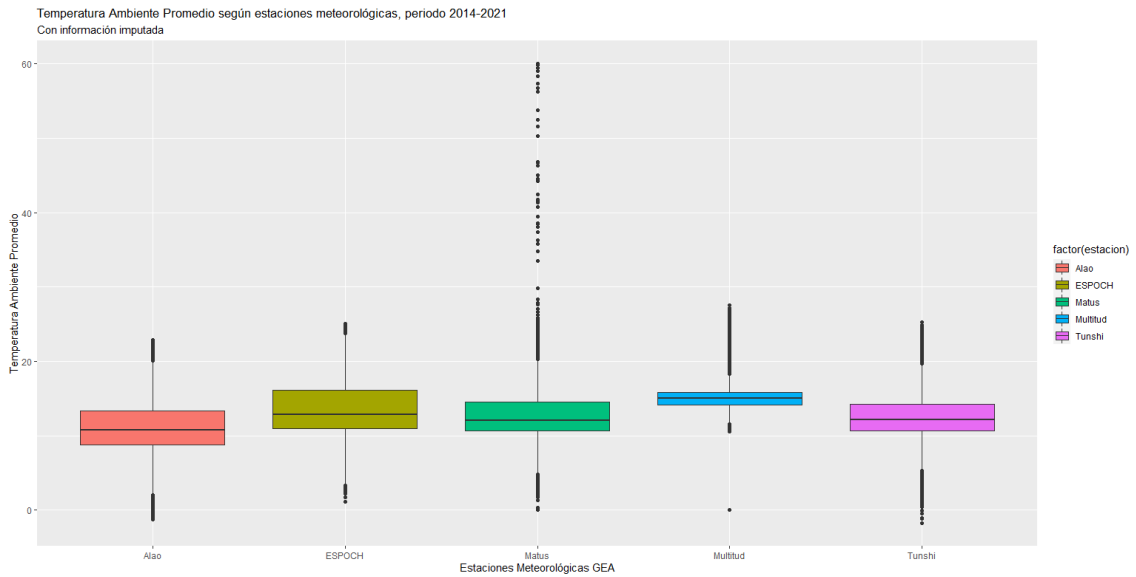


Ilustración 1-3: Temperatura ambiente promedio según estaciones meteorológicas, periodo 2014-2021, con información imputada.

Realizado por: Paucar, Belén, 2022.

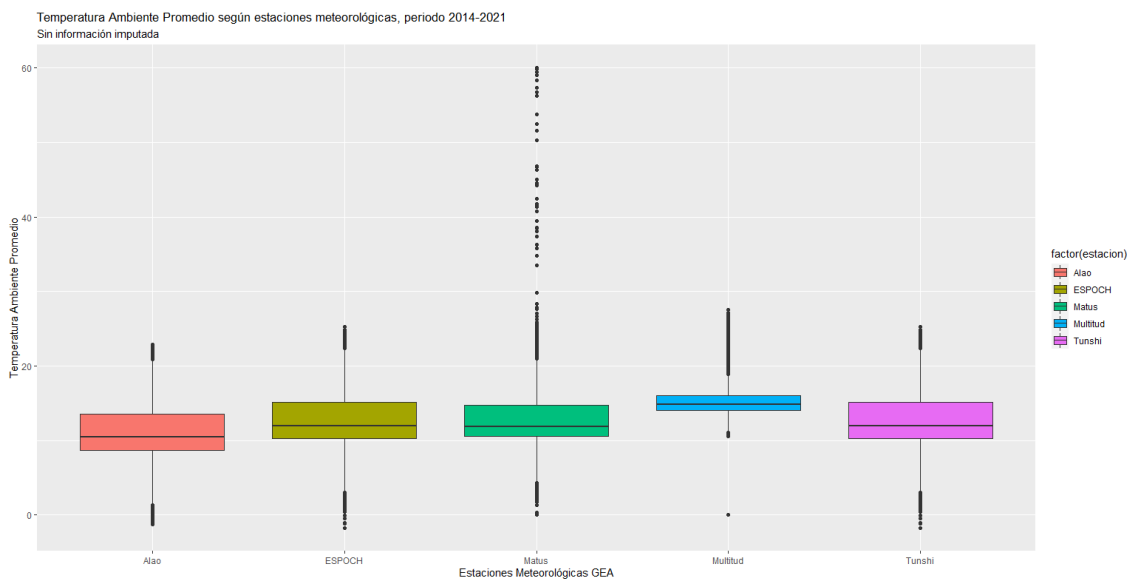


Ilustración 2-3: Temperatura ambiente promedio según estaciones meteorológicas, periodo 2014-2021, sin información imputada.

Realizado por: Paucar, Belén, 2022.

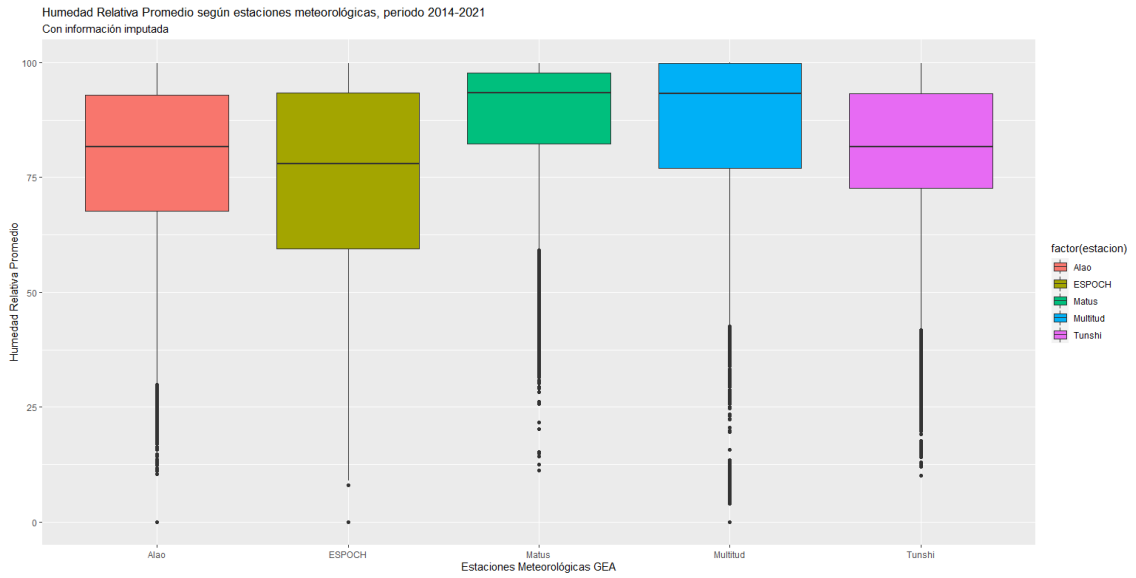


Ilustración 3-3: Humedad relativa promedio según estaciones meteorológicas, periodo 2014 2021, con información imputada.

Realizado por: Paucar Belén, 2022.

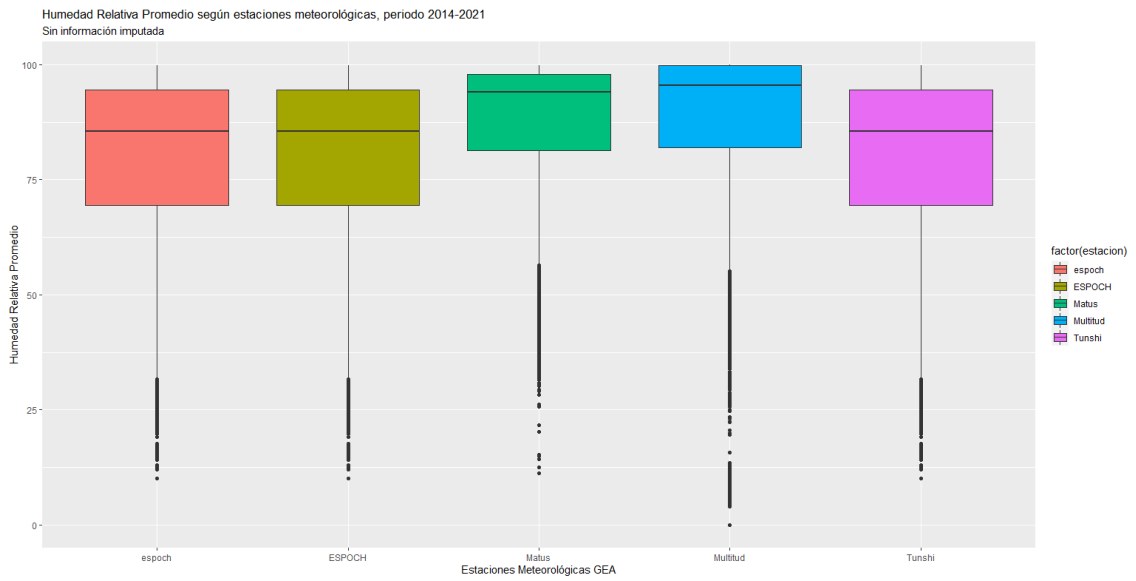


Ilustración 4-3: Humedad relativa promedio según estaciones meteorológicas, periodo 2014- 2021, sin información imputada.

Realizado por: Paucar Belén, 2022.

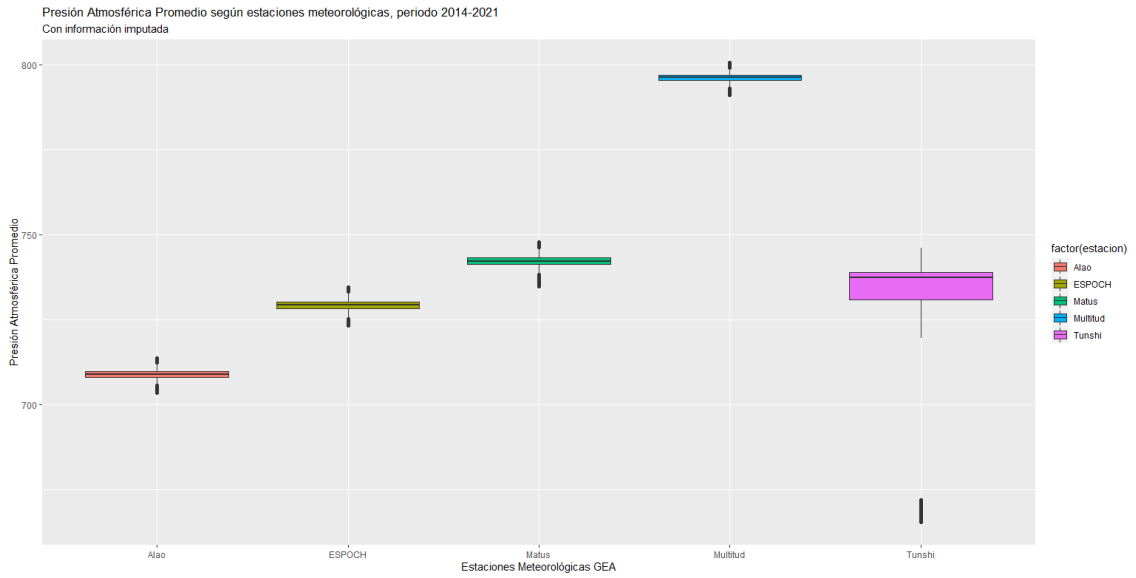


Ilustración 5-3: Presión atmosférica promedio según estaciones meteorológicas, periodo 2014-2021, con información imputada.

Realizado por: Belén E, Paucar C, 2022

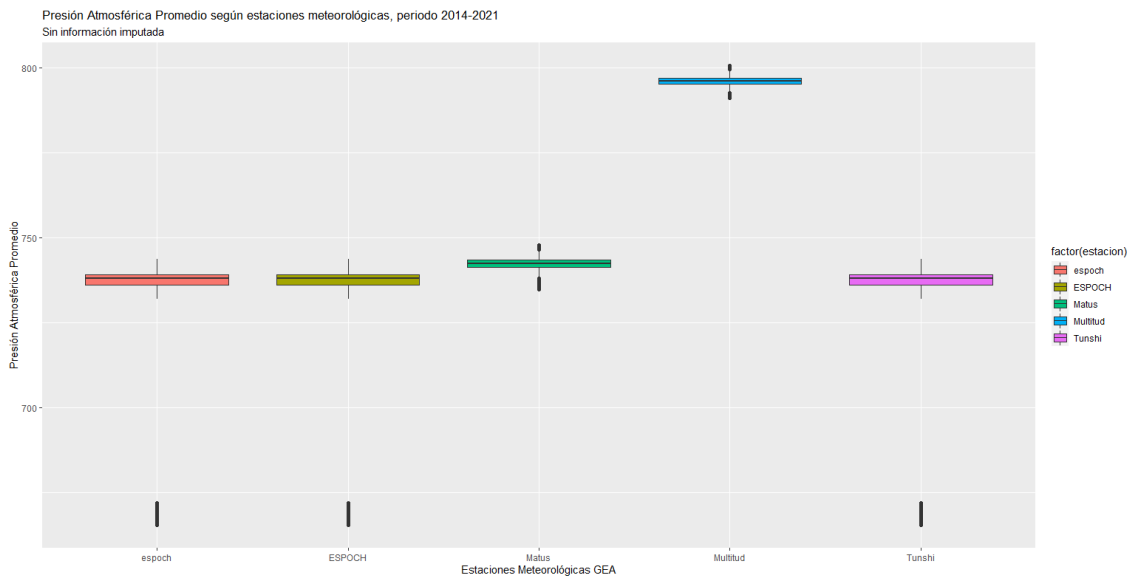


Ilustración 1-3: Presión atmosférica promedio según estaciones meteorológicas, periodo 2014-2021, sin información imputada.

Realizado por: Paucar Belén, 2022.

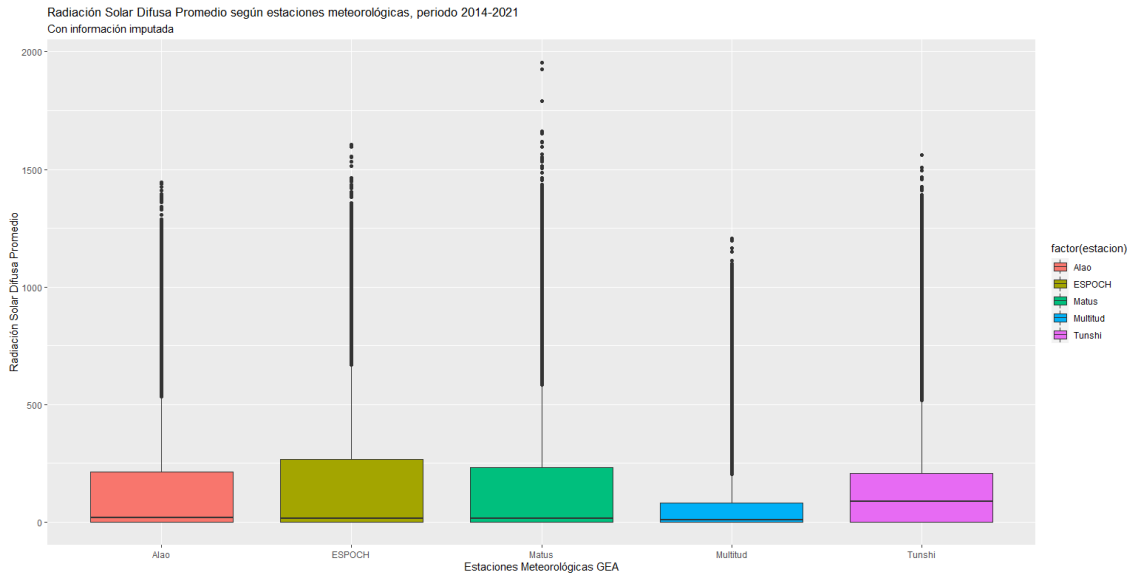


Ilustración 2-3: Radiación solar difusa promedio según estaciones meteorológicas, periodo 2014-2021, con información imputada.

Realizado por: Paucar Belén, 2022.

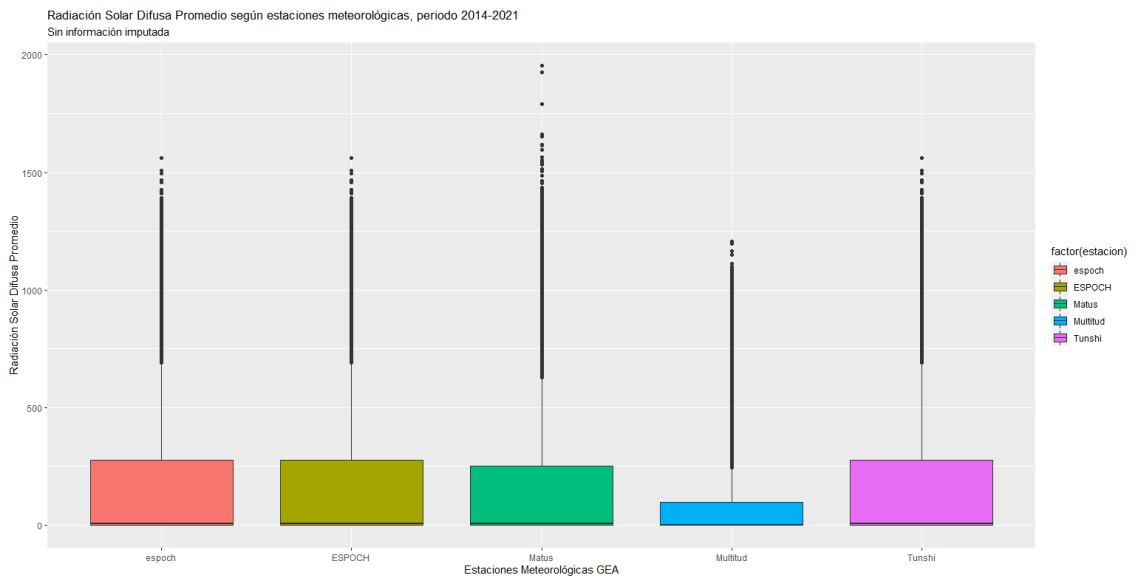


Ilustración 3-3: Radiación solar difusa promedio según estaciones meteorológicas, periodo 2014-2021, sin información imputada.

Realizado por: Paucar Belén, 2022.

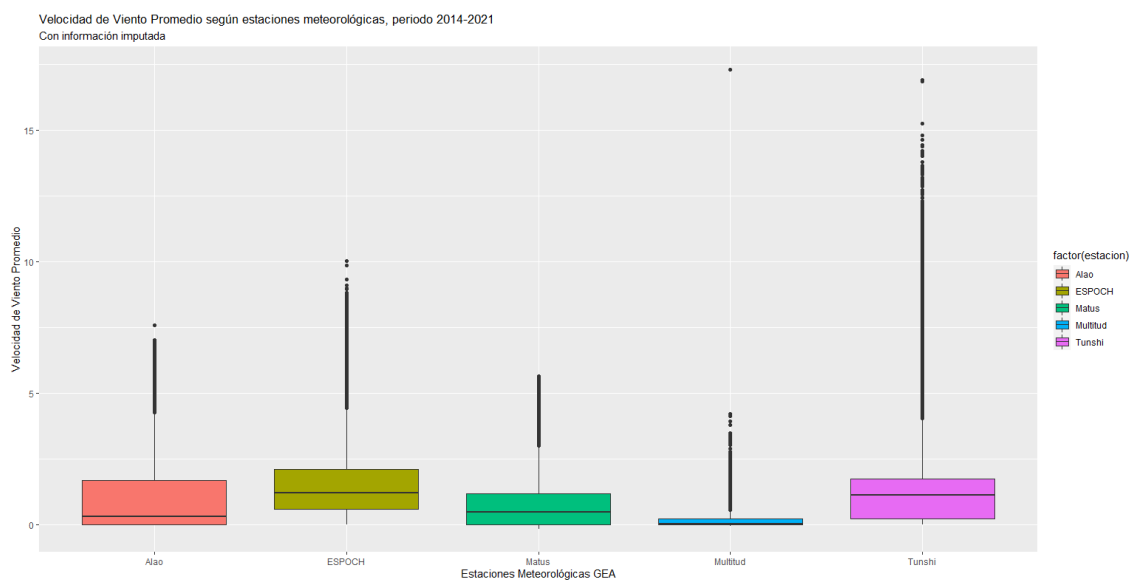


Ilustración 4-3: Velocidad de viento promedio según estaciones meteorológicas, periodo 2014-2021, con información imputada.

Realizado por: Paucar Belén, 2022.

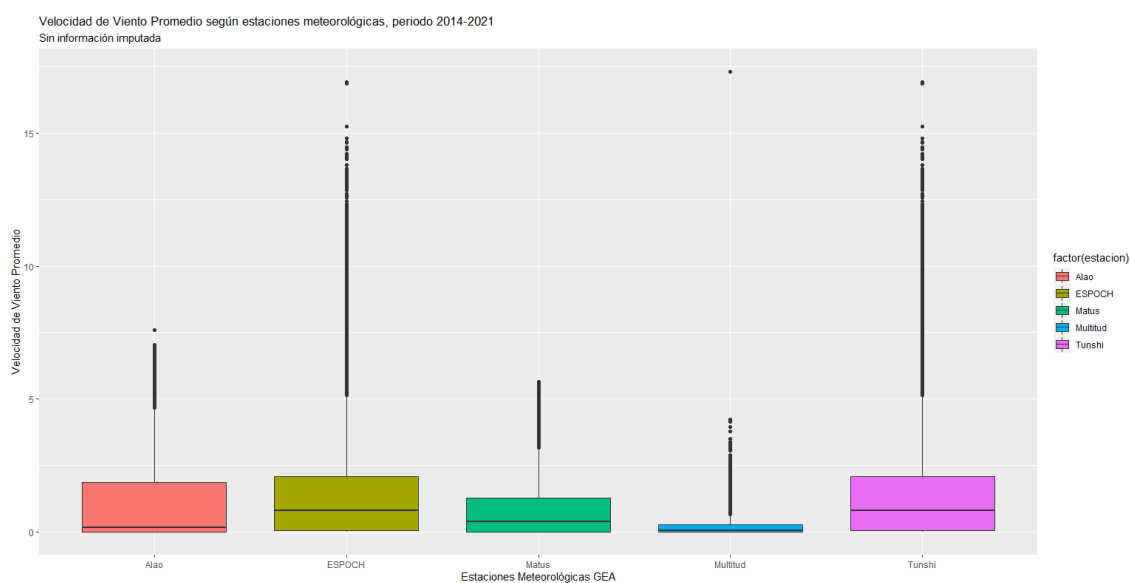


Ilustración 5-3: Velocidad de viento promedio según estaciones meteorológicas, periodo 2014-2021, sin información imputada.

Realizado por: Paucar Belén, 2022

Tabla 2-3: Resumen de estadísticas de estación meteorológica Alao con información imputada.

Estadístico	Fh	tap	hrp	Pap	rsdp	Vvp
N		68861	68861	68861	68861	68861
Mínimo	2014-01-28 19:00:00	-1.26	0.00	703.48	0.00	0.00
Q1	2016-01-16 02:00:00	8.80	67.73	708.10	0.00	0.00
Mediana	2018-01-02 09:00:00	10.74	81.73	708.93	17.53	0.32
Media	2018-01-02 09:00:00	11.07	79.14	708.90	137.16	1.01
Q3	2019-12-20 16:00:00	13.32	92.96	709.82	213.44	1.71

Máximo	2021-12-06 23:00:00	22.88	99.84	713.57	1443.95	7.59
Sd		3.38	16.48	1.33	205.86	1.36
Asimetría		0.30	-1.17	-0.38	2.06	1.39
Kurtosis		2.97	5.30	3.14	7.92	4.10

Fuente: Paucar Belén, 2022

Realizado por: Paucar Belén, 2022.

Tabla 3-3: Resumen de estadísticas de estación meteorológica ESPOCH con información imputada.

Estadístico	Fh	tap	hrp	Pap	rsdp	Vvp
N		60960	60960	60960	60960	60960
Mínimo	2015-01-01 00:00:00	1.15	0.00	723.14	0.00	0.00
Q1	2016-09-26 23:45:00	10.98	59.47	728.24	0.00	0.59
Mediana	2018-06-23 23:30:00	12.82	77.99	729.23	15.89	1.23
Media	2018-06-23 23:30:00	13.55	75.27	729.16	170.48	1.68
Q3	2020-03-19 23:15:00	16.09	93.44	730.22	267.44	2.13
Máximo	2021-12-14 23:00:00	25.10	99.86	734.41	1604.62	10.03
Sd		3.50	20.73	1.52	257.13	1.61
Asimetría		0.45	-0.80	-0.43	1.79	1.34
Kurtosis		2.68	3.51	3.11	5.85	4.26

Fuente: Paucar Belén, 2022

Realizado por: Paucar Belén, 2022.

Tabla 4-3: Resumen de estadísticas de estación meteorológica Matus con información imputada.

Estadístico	Fh	Tap	Hrp	Pap	rsdp	Vvp
N		61633	61633	61633	61633	61633
Mínimo	2014-11-27 23:00:00	0.00	11.17	734.73	0.00	-0.15
Q1	2016-08-30 23:00:00	10.62	82.33	741.28	0.00	0.00
Mediana	2018-06-03 23:00:00	12.03	93.36	742.23	14.40	0.48
Media	2018-06-03 23:00:00	12.70	88.53	742.19	153.77	0.86
Q3	2020-03-06 23:00:00	14.49	97.78	743.26	233.06	1.20
Máximo	2021-12-08 23:00:00	60.00	99.90	747.69	1952.60	5.65
Sd		3.34	12.10	1.56	235.71	1.12
Asimetría		1.36	-1.34	-0.48	1.94	1.50
Kurtosis		14.13	4.34	3.50	6.87	4.41

Fuente: Paucar Belén, 2022

Realizado por: Paucar Belén, 2022.

Tabla 5-3: Resumen de estadísticas de estación meteorológica Multitud con información imputada.

Estadístico	Fh	Tap	Hrp	Pap	rsdp	Vvp
N		69321	69321	69321	69321	69321
Mínimo	2014-01-24 15:00:00	0.00	0.00	791.04	0.00	-0.03
Q1	2016-01-16 17:00:00	14.12	76.93	795.31	0.00	0.00
Mediana	2018-01-07 19:00:00	15.05	93.25	796.17	10.59	0.05

Media	2018-01-07 19:00:00	15.09	83.58	796.06	73.99	0.15
Q3	2019-12-30 21:00:00	15.81	99.83	796.83	81.72	0.23
Máximo	2021-12-21 23:00:00	27.56	99.92	800.48	1207.22	17.30
Sd		2.13	25.34	1.19	129.85	0.25
Asimetría		-2.07	-2.37	-0.14	3.22	7.29
Kurtosis		22.44	8.06	3.16	16.76	323.09

Fuente: Paucar Belén, 2022

Realizado por: Paucar Belén, 2022.

Tabla 6-3: Resumen de estadísticas de estación meteorológica Tunshi con información imputada.

Estadístico	Fh	Tap	Hrp	Pap	rsdp	Vvp
N		69244	69244	69244	69244	69244
Mínimo	2013-12-18 20:00:00	-1.75	10.18	665.43	0.00	0.00
Q1	2015-12-10 02:45:00	10.67	72.65	730.77	0.00	0.23
Mediana	2017-11-30 09:30:00	12.17	81.61	737.44	86.83	1.12
Media	2017-11-30 09:30:00	12.52	81.16	727.35	164.04	1.52
Q3	2019-11-21 16:15:00	14.27	93.23	738.93	206.77	1.76
Máximo	2021-11-11 23:00:00	25.25	99.86	746.07	1562.25	16.93
Sd		3.39	14.14	22.90	228.89	1.76
Asimetría		0.27	-0.84	-2.02	2.00	2.28
Kurtosis		3.37	3.32	5.42	7.36	10.11

Fuente: Paucar Belén, 2022

Realizado por: Paucar Belén, 2022.

3.2. Modelos Box Jenkins, Redes Neuronales y Teoría del caos

3.2.1. Modelos Box Jenkins

Con la información detallada en los apartados anteriores se ajustan modelos Box Jenkins, estos modelos también son conocidos como modelos ARIMA y se ocupan en series temporales como las variables de interés de este estudio, donde generalmente existen dos clases de dependencia inter temporal que son: i) la relación lineal existente entre observaciones sucesivas (patrón de tendencial) y relación lineal existente entre observaciones del mismo mes en años sucesivos (patrón estacional).

Para estudiar los posibles patrones en las series de tiempo de interés se realiza gráficas de series de tiempo (ver Gráfico 3.11 a 3.15), en estos se observa que en general no existe un patrón de tendencia y no está claro el patrón estacional, sin embargo, para la variable Presión Atmosférica Promedio de la estación Tunshi la serie no parece ser estacionaria. Para corroborar esto, se realiza pruebas de hipótesis de Dickey Fuller para las 25 series de tiempo (ver Tabla 3.7). Luego, solo fue necesario realizar una diferenciación para la serie de tiempo de Presión Atmosférica Promedio de la estación Tunshi y se logró la estacionariedad.

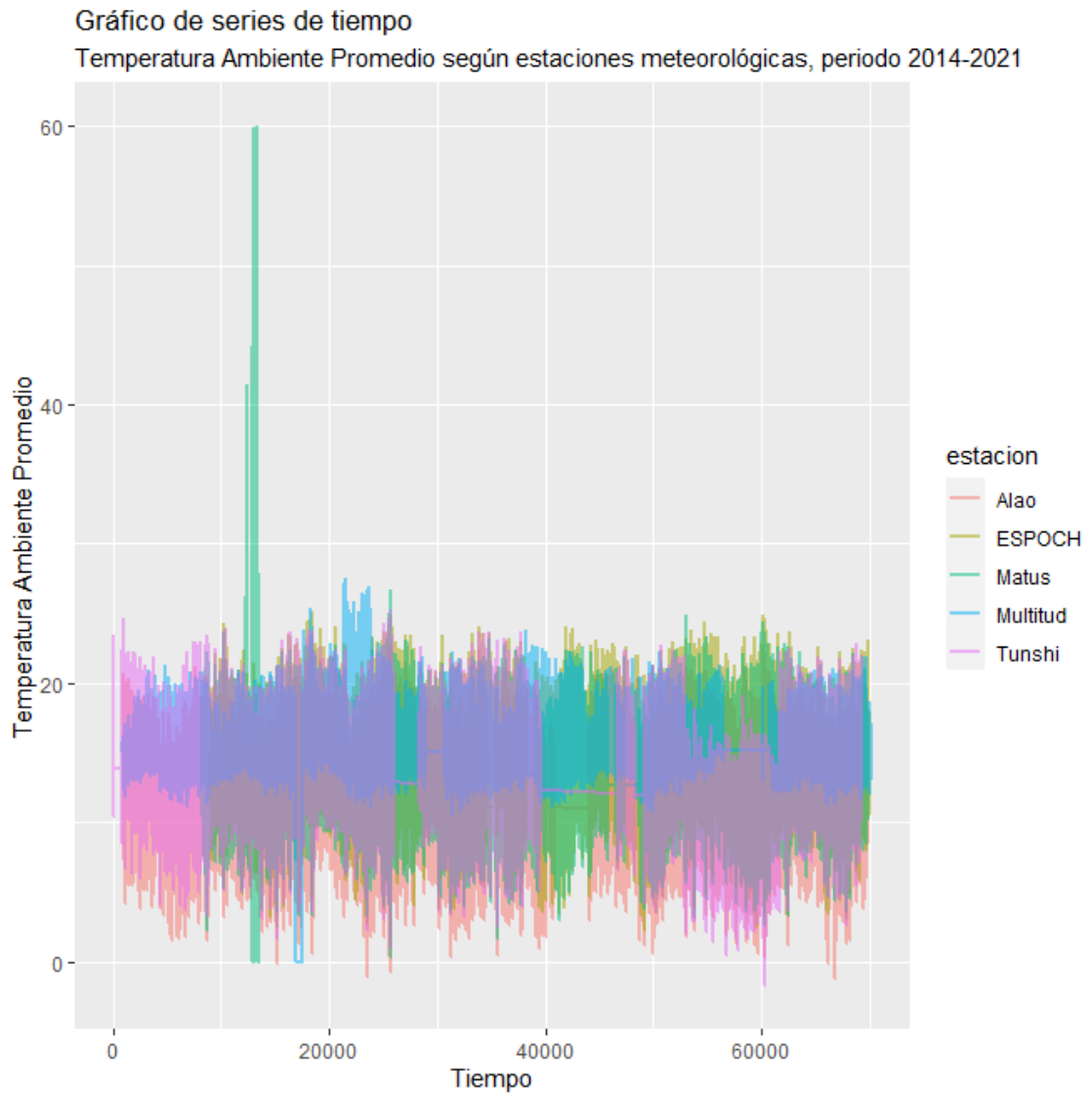


Ilustración 6-3: Gráfico de series de tiempo, temperatura ambiente promedio según estaciones meteorológicas, periodo 2014-2021.

Realizado por: Paucar Belén, 2022.

Gráfico de series de tiempo

Humedad Relativa Promedio según estaciones meteorológicas, periodo 2014-2021

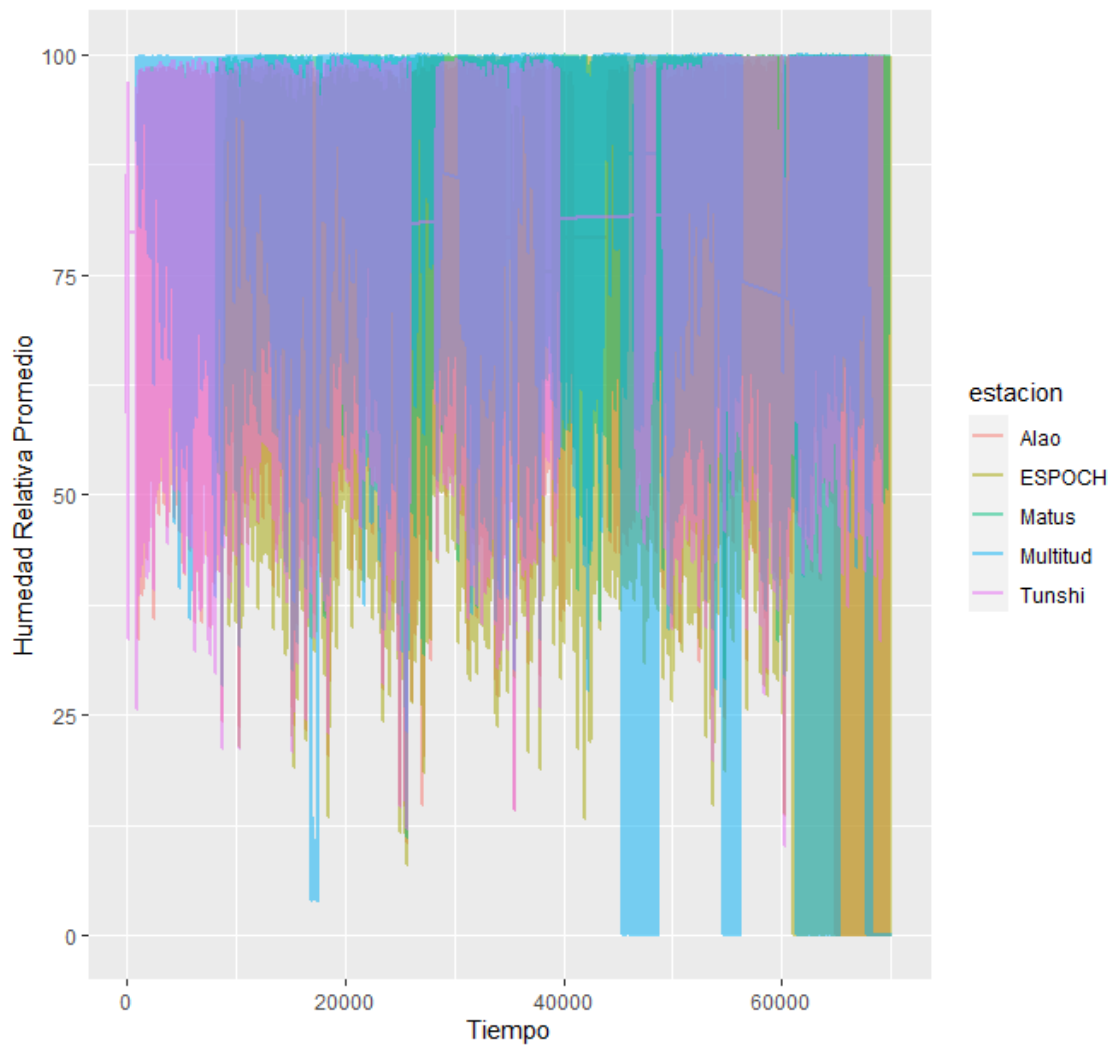


Ilustración 7-3: Gráfico de series de tiempo, humedad relativa promedio según estaciones meteorológicas, periodo 2014-2021.

Realizado por: Paucar Belén, 2022.

Gráfico de series de tiempo

Presión Atmosférica Promedio según estaciones meteorológicas, periodo 2014-2021

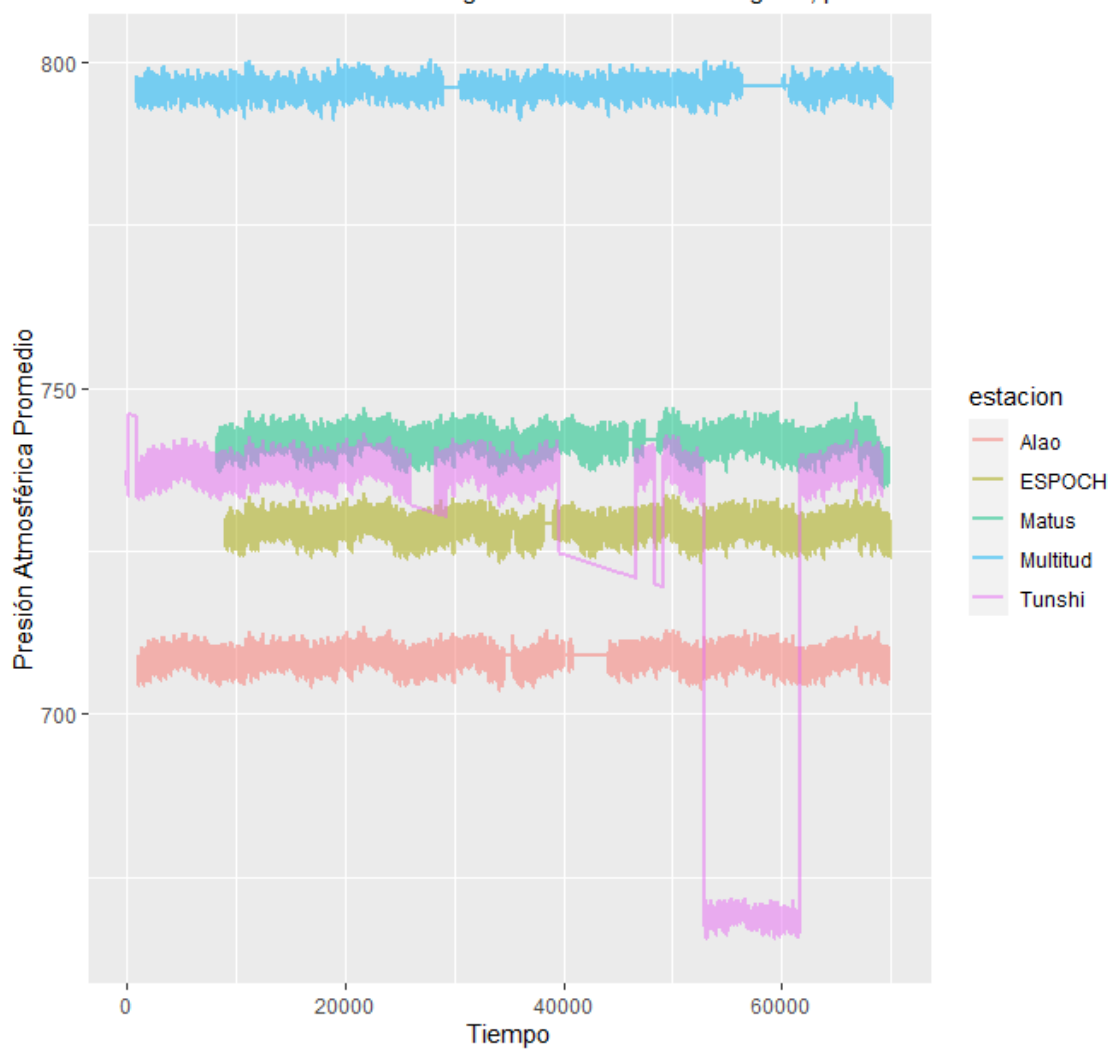


Ilustración 8-3: Gráfico de series de tiempo, presión atmosférica promedio según estaciones meteorológicas, periodo 2014-2021.

Realizado por: Paucar Belén, 2022.

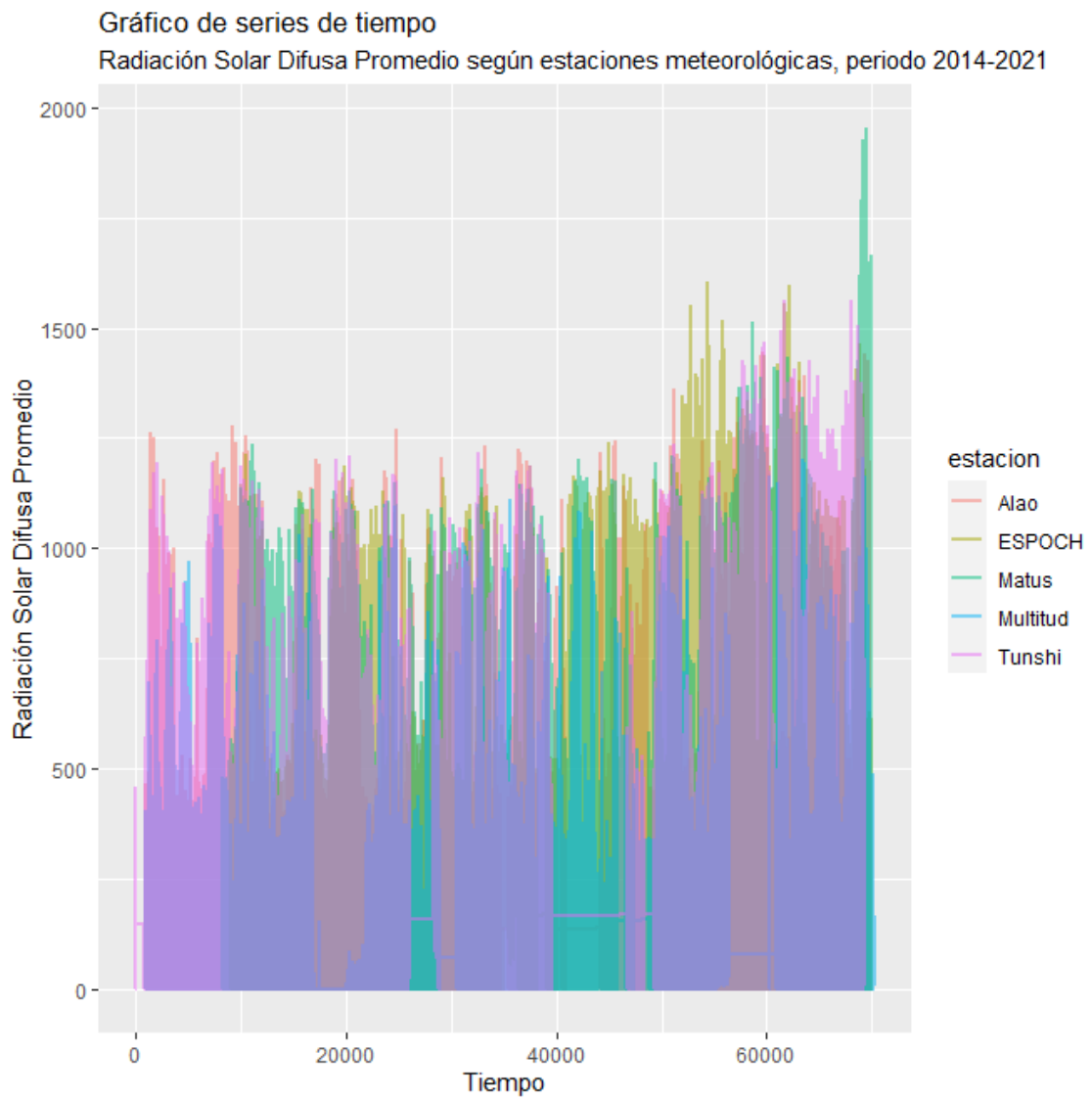


Ilustración 9-3: Gráfico de series de tiempo, radiación solar difusa promedio según estaciones meteorológicas, periodo 2014-2021.

Realizado por: Paucar Belén, 2022.

Gráfico de series de tiempo

Velocidad de Viento Promedio según estaciones meteorológicas, periodo 2014-2021

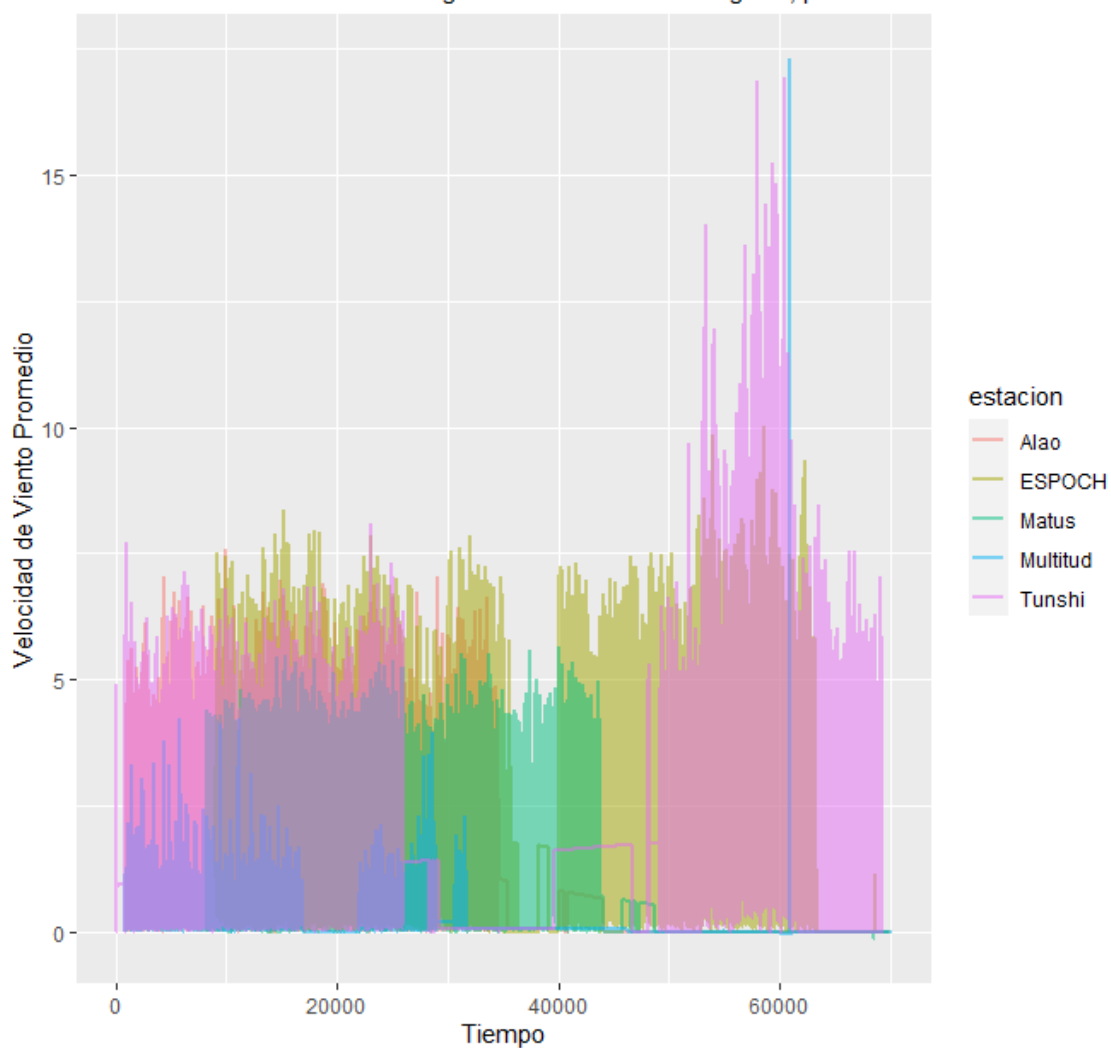


Ilustración 10-3: Gráfico de series de tiempo, velocidad de viento *promedio* según estaciones meteorológicas, periodo 2014-2021.

Realizado por: Paucar Belén, 2022.

Tabla 7-3: Valores de *p* de la Prueba de Dickey Fuller.

Estación	Tap	Hrp	pap	rsdp	vvp
Alao	0.01	0.01	0.01	0.01	0.01
ESPOCH	0.01	0.01	0.01	0.01	0.01
Matus	0.01	0.01	0.01	0.01	0.01
Multitud	0.01	0.01	0.01	0.01	0.01
Tunshi	0.01	0.01	0.46	0.01	0.01

Fuente: Paucar Belén, 2022

Realizado por: Paucar Belén, 2022.

Ahora que todas las series de tiempo son estacionarias, se determinan los mejores modelos ARIMA, siguiendo la metodología Box Jenkins. Los resultados de este proceso se muestran en la

Tabla 8-3. La medida de adecuación MAD muestra que el error promedio de pronóstico para las variables Temperatura Ambiente Promedio, Humedad Relativa Promedio, Presión Atmosférica Promedio, Radiación Solar Difusa Promedio y Velocidad de Viento Promedio son pequeñas en comparación del rango de cada variable. En el mismo sentido, los valores de MSE son los más pequeños de entre todos los modelos considerados, estos valores permitirán realizar una comparación con los modelos de redes neuronales y la teoría del caos.

Tabla 8-3: Mejores modelos ARIMA, según estación y variable.

Estación	Variable	Modelo	MAD	MSE
Alao	Tap	ARIMA(4,0,2)(2,1,0)[24]	0.57	0.66
	Hrp	ARIMA(5,0,0)(2,1,0)[24]	4.02	66.70
	Pap	ARIMA(3,0,1)(2,1,0)[24] with drift	0.14	0.04
	Rsdp	ARIMA(1,0,1)(2,1,0)[24]	40.94	8686.94
	Vvp	ARIMA(3,0,2)(2,1,0)[24]	0.36	0.45
ESPOCH	Tap	ARIMA(5,0,0)(2,1,0)[24]	0.53	0.60
	Hrp	ARIMA(3,0,1)(2,1,0)[24] with drift	4.83	97.40
	Pap	ARIMA(5,0,0)(2,1,0)[24]	0.15	0.05
	Rsdp	ARIMA(2,0,1)(2,1,0)[24]	52.96	12775.92
	Vvp	ARIMA(5,0,0)(2,1,0)[24]	0.49	0.57
Matus	Tap	ARIMA(3,0,5)(2,1,0)[24]	0.61	1.23
	Hrp	ARIMA(5,0,0)(2,1,0)[24]	2.41	14.52
	Pap	ARIMA(5,0,1)(2,1,0)[24]	0.14	0.04
	Rsdp	ARIMA(5,0,0)(2,1,0)[24]	48.98	11136.39
	Vvp	ARIMA(1,0,0)(2,1,0)[24] with drift	0.25	0.19
Multitud	Tap	ARIMA(5,0,0)(2,1,0)[24]	0.38	0.36
	Hrp	ARIMA(5,1,2)(0,0,2)[24]	3.22	76.03
	Pap	ARIMA(1,0,0)(2,1,0)[24] with drift	0.12	0.03
	Rsdp	ARIMA(0,0,1)(0,1,1)[24]	25.17	3702.86
	Vvp	ARIMA(2,1,4)(0,0,2)[24]	0.06	0.03
Tunshi	Tap	ARIMA(1,0,1)(2,1,0)[24]	0.46	0.52
	Hrp	ARIMA(1,0,0)(2,1,0)[24] with drift	2.57	17.50
	Pap	ARIMA(1,0,2)(2,1,0)[24]	0.14	0.26
	Rsdp	ARIMA(4,0,0)(2,1,0)[24]	44.89	11016.91
	Vvp	ARIMA(5,1,2)(0,0,2)[24]	0.46	0.70

Fuente: Paucar Belén, 2022.

Realizado por: Paucar Belén, 2022.

Con el propósito de validar los modelos presentados anteriormente, se comprobó si estos cumplen con los supuestos de independencia, estacionariedad, normalidad y homocedasticidad. Los resultados de este análisis se presentan en la Tabla 9-3. Para todos los modelos ARIMA ajustados no se cumplen los supuestos de Independencia, Normalidad y Homocedasticidad, sin

embargo, como se indicó anteriormente las series son estacionarias. Por lo tanto, se debe tener cuidado al utilizar estos modelos para la predicción. Una alternativa es utilizar la metodología Bootstrap para series de tiempo ya que esta no asume supuestos sobre las series de tiempo.

Tabla 9-3: Valores p de pruebas de supuestos en modelos ARIMA según estación y variable.

Estación	Variable	Modelo	Estacionariedad (Dickey-Fuller)	Independencia (Box-Ljung)	Normalidad (Jarque Bera)	Homocedasticidad (Goldfeld-Quandt)
Alao	tap	ARIMA(4,0,2)(2,1,0)[24]	0.01	0.00	0.00	0.00
	hrp	ARIMA(5,0,0)(2,1,0)[24]	0.01	0.00	0.00	0.00
	pap	ARIMA(3,0,1)(2,1,0)[24] with drift	0.01	0.00	0.00	0.00
	rsdp	ARIMA(1,0,1)(2,1,0)[24]	0.01	0.00	0.00	0.00
	vvp	ARIMA(3,0,2)(2,1,0)[24]	0.01	0.00	0.00	0.00
ESPOCH	tap	ARIMA(5,0,0)(2,1,0)[24]	0.01	0.00	0.00	0.00
	hrp	ARIMA(3,0,1)(2,1,0)[24] with drift	0.01	0.00	0.00	0.00
	pap	ARIMA(5,0,0)(2,1,0)[24]	0.01	0.00	0.00	0.00
	rsdp	ARIMA(2,0,1)(2,1,0)[24]	0.01	0.00	0.00	0.00
	vvp	ARIMA(5,0,0)(2,1,0)[24]	0.01	0.00	0.00	0.00
Matus	tap	ARIMA(3,0,5)(2,1,0)[24]	0.01	0.00	0.00	0.00
	hrp	ARIMA(5,0,0)(2,1,0)[24]	0.01	0.00	0.00	0.00
	pap	ARIMA(5,0,1)(2,1,0)[24]	0.01	0.00	0.00	0.00
	rsdp	ARIMA(5,0,0)(2,1,0)[24]	0.01	0.00	0.00	0.00
	vvp	ARIMA(1,0,0)(2,1,0)[24] with drift	0.01	0.00	0.00	0.00
Multitud	tap	ARIMA(5,0,0)(2,1,0)[24]	0.01	0.00	0.00	0.00
	hrp	ARIMA(5,1,2)(0,0,2)[24]	0.01	0.00	0.00	0.00
	pap	ARIMA(1,0,0)(2,1,0)[24] with drift	0.01	0.00	0.00	0.00
	rsdp	ARIMA(0,0,1)(0,1,1)[24]	0.01	0.00	0.00	0.00
	vvp	ARIMA(2,1,4)(0,0,2)[24]	0.01	0.00	0.00	0.00
Tunshi	tap	ARIMA(1,0,1)(2,1,0)[24]	0.01	0.00	0.00	0.00
	hrp	ARIMA(1,0,0)(2,1,0)[24] with drift	0.01	0.00	0.00	0.00
	pap	ARIMA(1,0,2)(2,1,0)[24]	0.01	0.00	0.00	0.00
	rsdp	ARIMA(4,0,0)(2,1,0)[24]	0.01	0.00	0.00	0.00
	vvp	ARIMA(5,1,2)(0,0,2)[24]	0.01	0.00	0.00	0.00

Fuente: Paucar Belén, 2022

Realizado por: Paucar Belén, 2022.

3.2.2. Redes Neuronales

Como respuesta al no cumplimiento de los supuestos en los modelos ARIMA, se plantea el ajuste de modelos de redes neuronales que son métodos no paramétricos que no necesitan de supuestos para su utilización.

Para el ajuste de modelos de redes neuronales recurrentes (RNN) que son un tipo especial de red neuronal artificial adaptada para trabajar con datos de series temporales o datos que involucran

secuencias se realizó primero una transformación de los datos mediante la ecuación logística, $y = (x - x_{min}) / (x_{max} - x_{min})$, luego se entrenó una red de Elman y, también de Jordan con el 70% de la información. La periodicidad de los datos fue de 24, se trabajó con dos capas ocultas, una de dos neuronas y otra de tres, para la red de Elman y para Jordan fue de 4 capas ocultas. El ritmo de aprendizaje fue de 0.07 con un número máximo de 5000 iteraciones. Lo anterior se realizó para todas las series de tiempo de las estaciones y las variables de interés,

A continuación, se presenta los resultados obtenidos del análisis de redes neuronales para este efecto se muestra una tabla (Tabla 10-3) con las medidas de adecuación de los pronósticos de los modelos de redes neuronales, luego por cada variable de interés se muestran resultados gráficos de los mejores modelos, es decir, de aquellos modelos con valores menores de MAD y MSE. Las gráficas que se muestran son: i) gráfica de evolución del error de la red según el número de iteraciones y ii) una gráfica de los valores reales y predichos para observar el ajuste de los pronósticos.

Tabla 10-3: Medidas de adecuación del modelo de Elma y Jordan según estación y variable.

Estación	Variable	Elman		Jordan	
		MAD	MSE	MAD	MSE
Alao	tap	0.94	1.37	1.26	2.21
	hrp	5.67	147.28	11.83	264.29
	pap	0.22	0.07	1.32	2.6
	rsdp	49.51	10905.75	99.05	36316.12
	vvp	0.02	0.01	0.03	0.01
ESPOCH	tap	0.77	1.01	1.31	2.44
	hrp	7.23	230.8	11.94	349.24
	pap	0.29	0.13	0.69	0.77
	rsdp	67.64	19494	116.27	26386.24
	vvp	0.65	0.75	0.48	0.72
Matus	tap	0.64	0.83	1.47	3.2
	hrp	2.89	17.26	41.96	3495
	pap	0.24	0.09	0.75	0.95
	rsdp	60.95	15002.72	158.99	146352.25
	vvp	11.94	142.6	0.05	0.02
Multitud	tap	0.35	0.27	0.54	0.52
	hrp	8.06	210.43	10.61	327.76
	pap	0.13	0.04	0.18	0.06
	rsdp	31.83	4806.26	41.23	5697.48
	vvp	0.05	0.02	0.04	0.02
Tunshi	tap	0.79	1.13	0.84	1.3
	hrp	4.29	34.38	8.64	133.17
	pap	2.9	9.63	20.44	744.65

rsdp	67.55	18882.74	506.26	764818.32
vvp	1.4	5.16	1.63	5.91

Fuente: Paucar Belén, 2022

Realizado por: Paucar Belén, 2022

La información de la Tabla 10-3 para el modelo de Elman, muestra que los mejores ajustes de las variables de interés según estación son:

- Temperatura Ambiente Promedio (tap): Multitud.
- Humedad Relativa Promedio (hrp): Matus.
- Presión Atmosférica Promedio (pap): Multitud.
- Radiación Solar Difusa Promedio (rsdp): Matus.
- Velocidad de Viento Promedio (vvp): Alao.

Por otro lado, se puede ver en la Tabla 3-10 para el modelo de Jordan los mejores ajustes de las variables de interés según estación son:

- Temperatura Ambiente Promedio (tap): Multitud.
- Humedad Relativa Promedio (hrp): Tunshi.
- Presión Atmosférica Promedio (pap): Multitud.
- Radiación Solar Difusa Promedio (rsdp): Multitud.
- Velocidad de Viento Promedio (vvp): Alao.

Los mejores modelos de Elman y Jordan fueron para las series de tiempo de Temperatura Ambiente Promedio (Multitud), Presión Atmosférica Promedio (Multitud) y Velocidad de Viento Promedio (Alao), mientras que para la serie de tiempo de Humedad Relativa Promedio el mejor modelo se obtuvo en la estación de Matus y Tunshi para la red neuronal de Elman y Jordan respectivamente. La serie de tiempo de Radiación Solar Difusa Promedio tuvo sus mejores modelos en las estaciones de Matus y Multitud, para la red neuronal de Elman y Jordan respectivamente.

Por lo tanto, para las variables y estaciones descritas anteriormente, se presentan a continuación las gráficas de evolución del error de la red según el número de iteraciones (ver Gráfico 3-16 y Gráfico 3-17) y las gráficas de los valores reales y predichos para observar el ajuste de los pronósticos con el 30% de la información reservadas para el proceso de prueba de los modelos (ver Gráfico 18-3 y Gráfico 19-3). En general las gráficas de evolución del error según el número de iteraciones tanto para Elman como para Jordan, muestran que el error converge a cero

rápidamente, esto es bueno ya que de esta manera se asegura que los pronósticos sean adecuados. En el mismo sentido, las gráficas de valores reales de la serie de tiempo y valores pronosticados vs tiempo representan el buen ajuste de los modelos de Elman y de Jordan. Sin embargo, de la información de la Tabla 10-3 se puede decir que el modelo de Elman es mejor que el modelo de Jordan en todas las variables de interés y estaciones meteorológicas, con la excepción de la variable Velocidad de Viento Promedio de las estaciones de Matus y ESPOCH.

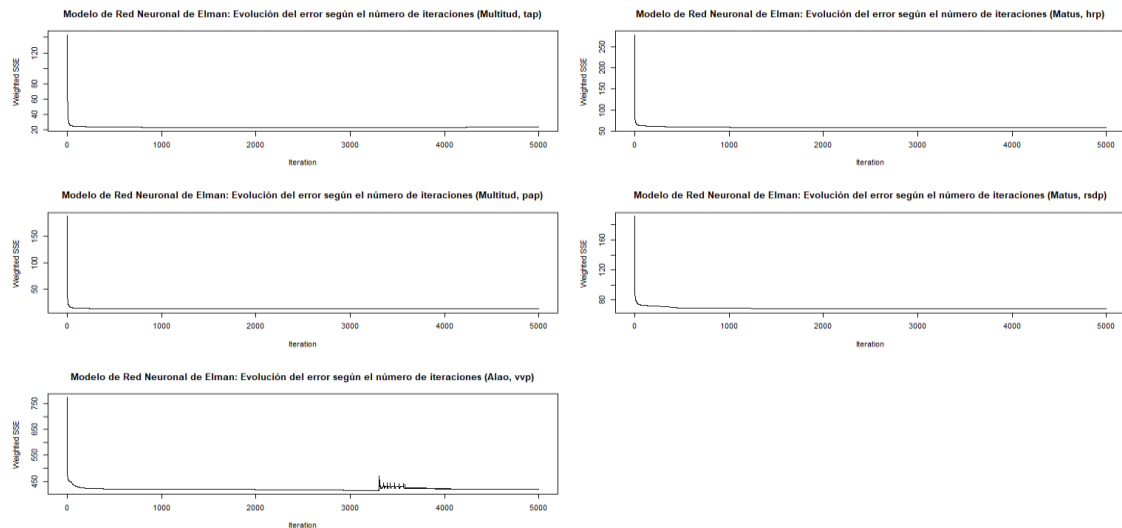


Ilustración 11-3: Mejores modelo de Red Neuronal de Elman: Evolución del error según el número de iteraciones

Realizado por: Paucar Belén, 2022

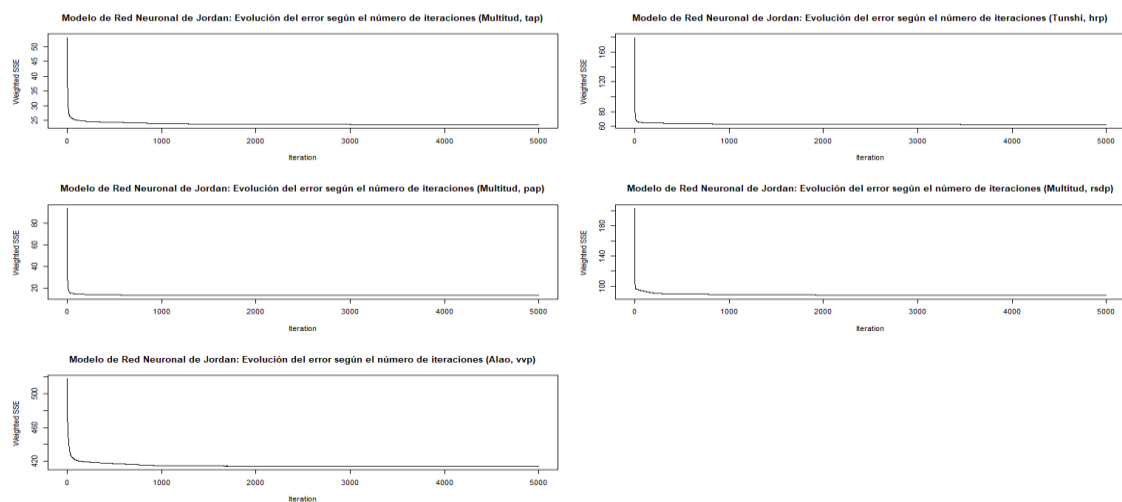


Ilustración 12-3: Mejores Modelo de Red Neuronal de Jordan: Evolución del error según el número de iteraciones.

Realizado por: Paucar Belén, 2022

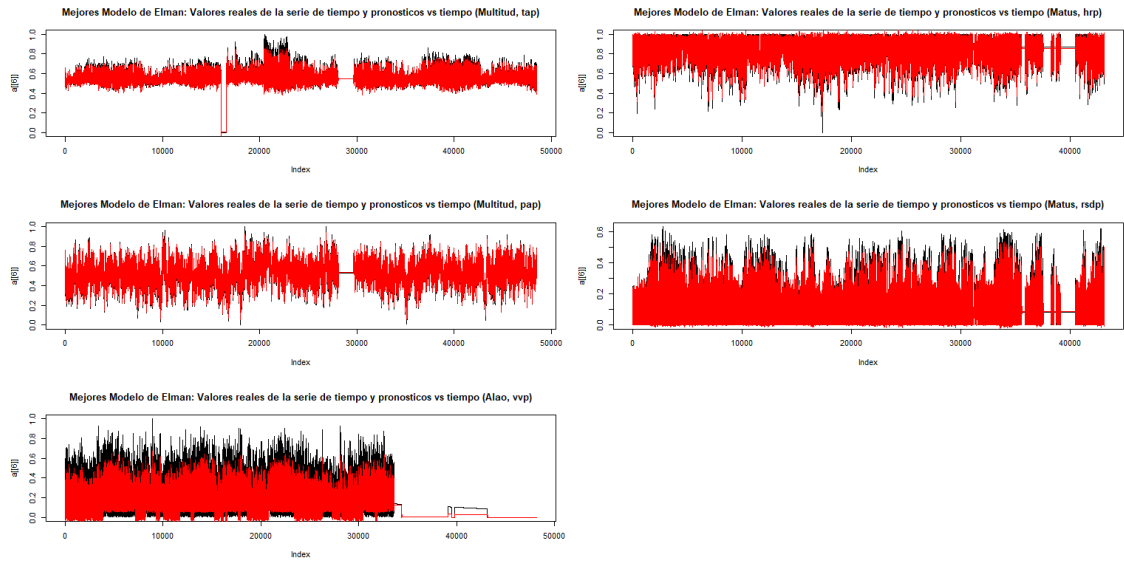


Ilustración 13-3: Mejores modelo de Red Neuronal de Elman: valores reales de la serie de tiempo (color negro) y valores pronosticados (color rojo) vs tiempo.

Realizado por: Paucar Belén, 2022

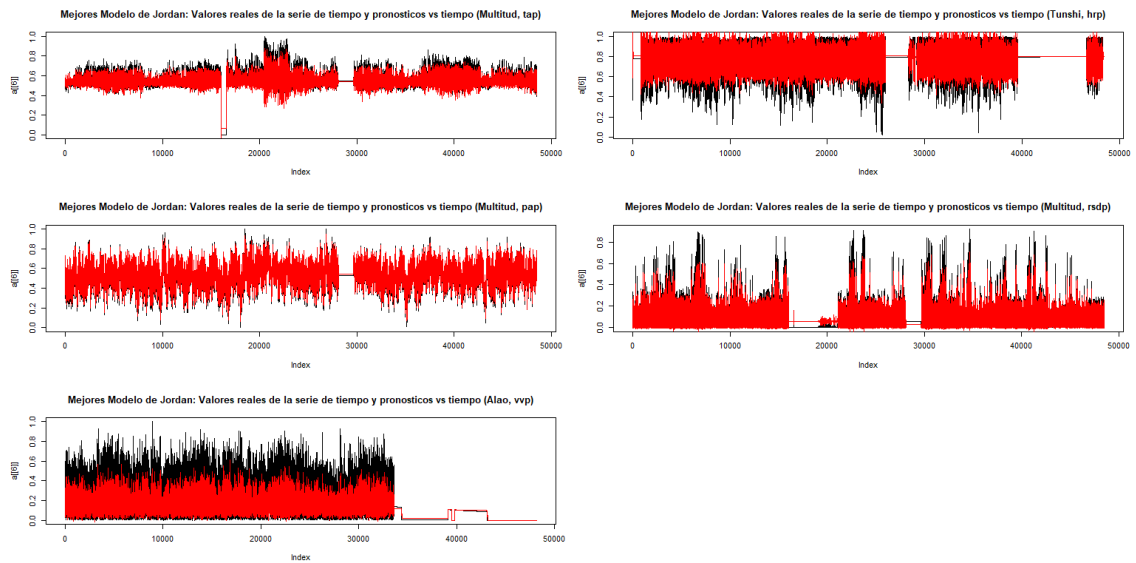


Ilustración 14-3: Mejores modelo de Red Neuronal de Jordan: Valores reales de la serie de tiempo (color negro) y valores pronosticados (color rojo) vs tiempo.

Realizado por: Paucar Belén, 2022

3.2.3. *Teoría del caos*

La incertidumbre y la no linealidad en una serie de tiempo meteorológica no tienen por qué estar siempre causadas únicamente por la aleatoriedad. Puede deberse a fenómenos deterministas, etiquetados como caos, que es muy sensible a las condiciones iniciales. La teoría del caos ofrece una nueva forma de modelar el comportamiento dinámico no lineal subyacente de un sistema complejo determinista al incorporar una serie de tiempo meteorológica escalar dada en su espacio de fase correspondiente utilizando parámetros como el retraso y la dimensión de incorporación, donde lag es el retraso de tiempo y la dimensión de incrustación significa el número de variables necesarias para representar la dinámica no lineal del sistema caótico. De esta manera a continuación se muestra los resultados del estudio de las variables meteorológicas, mediante la Teoría del Caos.

En la Tabla 11-3 se muestran las medidas de adecuación MAD y MSE de los modelos ajustados mediante la Teoría del Caos, donde los mejores modelos según Estación y Variables son:

- Temperatura Ambiente Promedio (tap): Multitud.
- Humedad Relativa Promedio (hrp): Multitud.
- Presión Atmosférica Promedio (pap): Multitud.
- Radiación Solar Difusa Promedio (rsdp): **Multitud.**
- Velocidad de Viento Promedio (vvp): **ESPOCH.**

La Gráfica 20-3 indica los valores pronosticados y reales para un tiempo de semana. En general, los pronósticos no son buenos ya que no captura toda la variabilidad, excepto para la variable Humedad Relativa Promedio.

Tabla 11-3: Medidas de adecuación Teoría del Caos según estación y variable.

Estación	Variable	Tiempo Retardo	MAD	MSE
Alao	Tap	6	2.76	12.52
	Hrp	7	29.40	2590.50
	Pap	7	1.17	2.65
	Rsdp	5	239.96	201648.27
	Vvp	8	1.16	3.01
ESPOCH	Tap	6	1.64	4.19
	Hrp	7	6034.44	6056073407.43
	Pap	6	1.27	2.67
	Rsdp	6	102.76	14342.97
	Vvp	6	1.16	3.01
Matus	Tap	6	1.84	4.92
	Hrp	6	4.33	19.77
	Pap	7	3.27	12.99
	Rsdp	5	284.44	203732.34
	Vvp	6	1.16	3.01
Multitud	Tap	8	1.03	1.68
	Hrp	13	8.53	115.67
	Pap	7	0.72	0.90
	Rsdp	6	52.69	4765.42
	Vvp	8	1.16	3.01
Tunshi	Tap	7	2.42	10.08
	Hrp	7	13.71	260.01
	Pap	7	7.58	80.64
	Rsdp	5	246.22	116318.26
	Vvp	10	1.16	3.01

Fuente: Paucar Belén, 2022

Realizado por: Paucar Belén, 2022

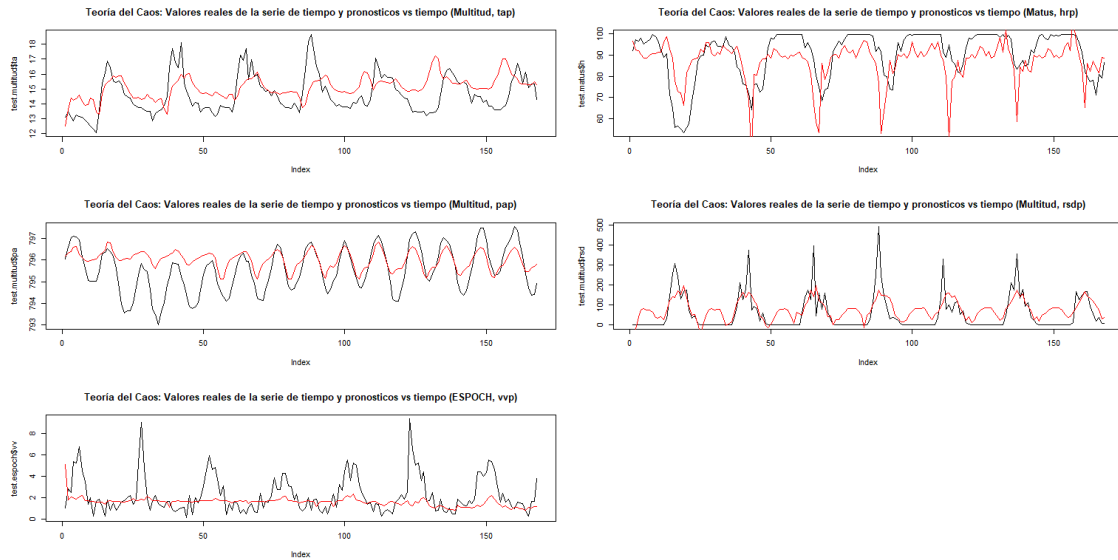


Ilustración 15-3: Teoría del Caos: valores reales de la serie de tiempo (color negro) y valores pronosticados (color rojo) vs tiempo.

Realizado por: Paucar Belén, 2022

3.2.4. Comparación de métodos de pronósticos

Se comparó los resultados de la última semana de la información, es decir, 168 datos de los pronósticos desarrollados por los modelos ARIMA, Redes Neuronales y Teoría del Caos mediante el coeficiente U Theil y el Test de Diabolt Mariano. El coeficiente U Theil permite determinar la precisión del pronóstico, mientras el valor del coeficiente se acerca a cero, la presión aumenta. En este sentido (ver Tabla 12-3), el método de Red Neuronal tuvo los valores más cercanos a cero en casi todas las variables y estaciones, con 7 excepciones de los 25 casos. Estas excepciones fueron para Alao en las variables Presión Atmosférica Promedio y Velocidad de Viento Promedio, ESPOCH y Matus en la variable Presión Atmosférica Promedio, Multitud en las variables Temperatura Ambiente Promedio y Presión Atmosférica Promedio, y Tunshi en la variable de Presión Atmosférica Promedio. De estas 7 excepciones, en 4 de ellas el coeficiente U Theil es igual entre los modelos ARIMA y la Teoría del Caos, mientras que en los 3 restantes el valor inferior del coeficiente resulto para el modelo ARIMA. Por lo tanto, en términos generales el mejor método de pronóstico resulto ser las Redes Neuronales, seguido por los modelos ARIMA y luego la Teoría del Caos. Siguiendo los resultados del apartado anterior, se realizó el Test de Diabolt Mariano para comparar los modelos de Red Neuronal y de la Teoría del Caos, probando la hipótesis alternativa que el Método de Red Neuronal es más preciso que el método ARIMA. En este sentido, observando la Tabla 3-13 se tiene que en todos los casos los modelos de Redes Neuronales son mejores, excepto para la estación meteorológica Alao en la variable Velocidad de Viento Promedio. Este resultado, apoya lo indicado en el apartado anterior y, por lo tanto, el

modelo de red neuronal genera los modelos más consistentes y confiables en comparación de los modelos ARIMA y Teoría del Caos.

Tabla 12-3: Coeficiente U Theil según estación y variable.

Estación	Variable	ARIMA	Red Neuronal Elman	Teoría Caos
Alao	tap	0.10	0.06	0.29
	hrp	0.43	0.08	0.69
	pap	0.01	0.02	0.01
	rsdp	0.63	0.19	1.45
	vvp	0.01	0.10	0.63
ESPOCH	tap	0.07	0.04	0.14
	hrp	0.53	0.05	1
	pap	0.01	0.03	0.01
	rsdp	0.36	0.28	0.60
	vvp	0.42	0.27	0.63
Matus	tap	0.06	0.05	0.17
	hrp	0.05	0.04	0.12
	pap	0.01	0.02	0.01
	rsdp	0.44	0.32	0.96
	vvp	0.29	0.01	0.63
Multitud	tap	0.03	0.04	0.09
	hrp	0.03	0.02	1
	pap	0.01	0.03	0.01
	rsdp	0.59	0.25	0.65
	vvp	0.10	0.01	0.63
Tunshi	tap	0.06	0.01	0.22
	hrp	0.10	0.02	0.21
	pap	0.01	0.05	0.01
	rsdp	0.50	0.17	0.84
	vvp	0.66	0.13	0.63

Fuente: Paucar Belén, 2022

Realizado por: Paucar Belén, 2022.

Tabla 13-3: Test de Diabolt Mariano según estación y variable.

Estación	Variable	ARIMA vs Red Neuronal
Alao	tap	0.00
	hrp	0.00
	pap	0.00
	rsdp	0.00
	vvp	1.00
ESPOCH	tap	0.00
	hrp	0.00
	pap	0.00
	rsdp	0.00
	vvp	0.00
Matus	tap	0.00
	hrp	0.00
	pap	0.00
	rsdp	0.00
	vvp	0.00
Multitud	tap	0.00
	hrp	0.00
	pap	0.00
	rsdp	0.00
	vvp	0.00
Tunshi	tap	0.00
	hrp	0.00
	pap	0.00
	rsdp	0.00
	vvp	0.00

Fuente: Paucar Belén, 2022

Realizado por: Paucar Belén, 2022.

3.2.5. *Aplicativo web*

El desarrollo de aplicativos webs con código R es importante básicamente al utilizar datos científicos, y sirven principalmente para la difusión de resultados de investigación. Este es el caso del presente estudio, donde se realizó un aplicativo web que muestra los resultados descritos en los apartados anteriores, cuenta con las siguientes pestañas:

- Gráficos de Series de Tiempo.
- ARIMA: Valores reales (negro) y pronosticados (rojo).
- RN: Valores reales (negro) y pronosticados (rojo).
- TC: Valores reales (negro) y pronosticados (rojo).

En la primera pestaña se muestran los gráficos de series de tiempo por estación de las variables de interés Temperatura Ambiente Promedio, Humedad Relativa Promedio, Presión Atmosférica Promedio, Radiación Solar Difusa Promedio y Velocidad de Viento Promedio. En las pestañas de la uno a la tres se presentan los valores reales y pronosticados por estación meteorológica según los modelos ARIMA, Redes Neuronales y Teoría del Caos. En la parte de anexos de muestra el código necesario para el desarrollo de este aplicativo junto con el código de los modelos ARIMA, Redes Neuronales y Teoría del Caos.

Finalmente, se muestra a continuación el aspecto del aplicativo web.



Ilustración 16-3: Pantalla del aplicativo web

Realizado por: Paucar Belén, 2022.

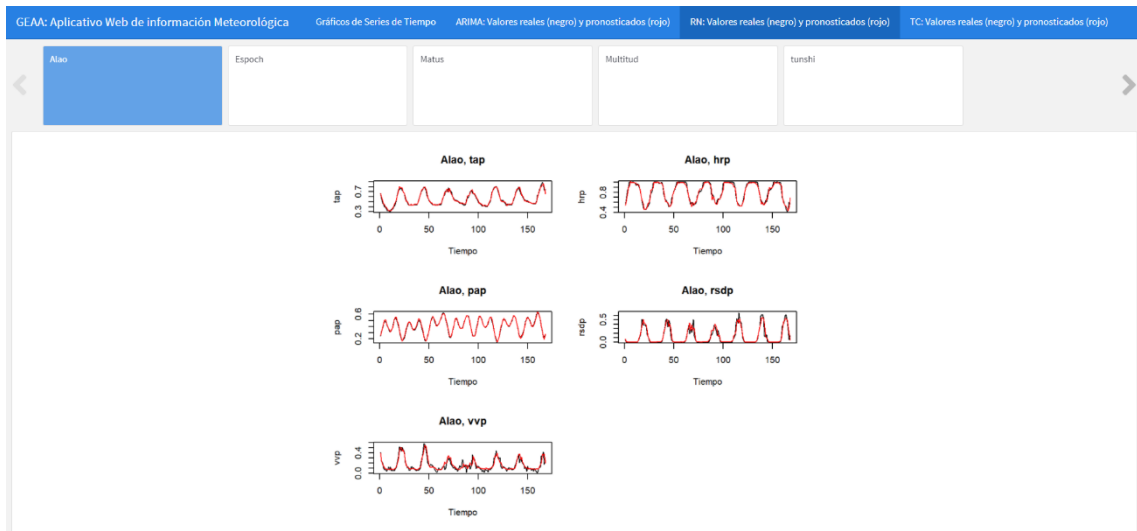


Ilustración 17-3: Resultados de ARIMA en el aplicativo web

Realizado por: Paucar Belén, 2022.

DISCUSIÓN

Los resultados presentados en el apartado anterior son consistentes con estudios contemporáneos. En el análisis de información faltante se tiene que el 8.66% de la información son valores faltantes, resultados similares se tiene en el estudio reportado por (Pilco & Acurio, 2019)

El resultado importante de que las estaciones meteorológicas según las variables de interés presentan distribuciones con tendencia a la simetría y con forma mesocúrtica para las variables Temperatura Ambiente Promedio y Presión Atmosférica Promedio es consistente con los resultados de (Serrano Vincenti, et al., 2012). En el mismo sentido el hecho que las variables Humedad Relativa Promedio, Radiación Solar Difusa Promedio y Velocidad de Viento Promedio tienen una distribución asimétrica con forma leptocúrtica concuerda con lo indicado por (Wright, 2014)

Por otro lado, se tiene que en términos generales el mejor método de pronóstico resulto ser las Redes Neuronales, seguido por los modelos ARIMA y luego la Teoría del Caos según el coeficiente U Theil y el Test de Diabolt Mariano. Este resultado va en el mismo sentido que los encontrados por (Pilco & Acurio, 2019), Paucar (2021) y Capito (2021). Por lo tanto, esto es evidencia a favor, para preferir realizar pronósticos meteorológicos mediante las Redes Neuronales de las variables Temperatura Ambiente Promedio, Humedad Relativa Promedio, Presión Atmosférica Promedio, Radiación Solar Difusa Promedio y Velocidad de Viento Promedio en el Grupo de Energía Alternativa y Ambiente.

El desarrollo de un aplicativo web para presentar los resultados de este trabajo es un aporte significativo para la difusión de estudios profundos de aplicación de métodos como ARIMA, Redes Neuronales y Teoría del Caos.

Una limitación fundamental del estudio es la imposibilidad de estudiar la generación de valores faltantes, es decir, tener una idea clara del surgimiento de los NA's. Entonces, es necesario un estudio profundo que debe la generación de valores faltantes en los datos meteorológicos por estación del Grupo de Energía Alternativa y Ambiente.

CONCLUSIONES

En este trabajo se presenta el estudio estadístico para pronosticar información meteorológica del Grupo de Energía Alternativa y Ambiente.

- Se tuvo que menos de un décimo de la información meteorológica fueron datos faltantes esto permitió la imputación mediante el método de regresión lineal.
- El estudio descriptivo de las series de tiempo por estación meteorológica de las variables Temperatura Ambiente Promedio, Humedad Relativa Promedio, Presión Atmosférica Promedio, Radiación Solar Difusa Promedio y Velocidad de Viento Promedio muestra que en general se presentan distribuciones con tendencia a la simetría y con forma mesocúrtica para las variables Temperatura Ambiente Promedio y Presión Atmosférica Promedio. En el mismo sentido, las variables Humedad Relativa Promedio, Radiación Solar Difusa Promedio y Velocidad de Viento Promedio tienen una distribución asimétrica con forma leptocúrtica.
- De los métodos utilizado el mejor resultó ser el de Redes Neuronales en comparación con el método ARIMA y Teoría del Caos. El método de Redes Neuronales genera modelos consistentes y confiables para la predicción de información meteorológica del Grupo de Energía Alternativa y Ambiente.
- Lo anterior, se puede ver en un aplicativo web desarrollado para la difusión de los resultados más importantes de este trabajo de titulación.

RECOMENDACIONES

- Realizar un estudio profundo sobre el motivo al que se debe la generación de valores faltantes en los datos meteorológicos por estación del Grupo de Energía Alternativa y Ambiente, es decir, cuáles son los determinantes de la aparición de valores faltante como, por ejemplo, fallas en el equipo al momento de la toma de información, información verdaderamente extrema, etc.
- Considerar el estudio de variables ambientales que no formaron parte del trabajo como por ejemplo aquellas variables tomadas con una periodicidad por hora.
- Reproducir en tiempo real este estudio, para generar pronósticos confiables y válidos.
- Difundir los resultados con la sociedad chimboracense mediante el aplicativo web.

BIBLIOGRAFÍA

- ACOSTA, A., 2006.** *Breve Historia Económica del Ecuador*. Primera ed. Quito, Ecuador: Corporación.
- AJ, G., KA, D. & LE, M. & J. S., 2014.** Injury prediction in a side impact crash using human body model simulation. *Accid Anal Prev*, Volumen 64, pp. 1-8.
- ALEXANDRA MARCATOMA, N. P. H. M., 2020.** *Análisis del cambio climático en un ecosistema Alto andino, Riobamba - Ecuador*. [Arte] (Perfiles).
- APARICIO, J., 2013.** *Modelos lineales Aplicados en R*. [Arte] (Universidad Miguel Hernández).
- APARICIO, J., 2018.** Modelos lineales aplicados en R. *Centro de investigación Operativa*, p. 229.
- ARAGÓN, 2014.** s.l.: s.n.
- ARTEAGA, P., BATANERO, C., DÍAZ, C. & CONTRERAS, J. M., 2009.** El lenguaje de los gráficos estadísticos. *Revista Iberoamericana de Educación Matemática*, Issue 18, pp. 93 -104.
- ARTEAGA, R., CERVANTES, R., VÁZQUEZ, M. A. & OJEDA, W., 2017.** *Estación meteorológica convencional versus automática sus diferencias evaluadas con índices estadísticos Chapingo, México*. [Arte].
- BAILEY, T. C. & GATRELL, A. C., 1995.** *Interactive Spatial Data Analysis*, s.l.: s.n.
- BEERS, W. V. & KLEIJNEN, J. P., 2003.** Kriging for interpolation in random simulation. *Journal of the Operational Research Society*, 54(3), pp. 113-120.
- BLACK, W. R. & THOMAS, I., 1998.** Accidents on Belgium's motorways: a network autocorrelation analysis. *J. Transp. Geogr*, 6(1), pp. 23-31.
- BROSA, J. V., 2002.** *El diagnóstico de la sobredispersión de la sobredispersión en modelos de análisis de datos de recuento*. [Arte] (Universitat Autònoma de Barcelona).
- CENTER FOR DISEASE CONTROL AND PREVENTION, 2020.** *Division of unintentional injury prevention*, s.l.: s.
- CESTERO, E. V. & CABALLERO, A. M., 2018.** *Data science y redes complejas*. primera ed. s.l.:s.n.
- CHÁVEZ, V. & USECHE, M., 2021.** Perfil descriptivo de los siniestros de tránsito en Ecuador durante el periodo 2015-2020. *Ciencias técnicas y aplicadas*, 7(3), pp. 5-6.
- CHEN, S., KUHN, M. & PRETTNER, K. & B. D. E., 2019.** The global macroeconomic burden of road injuries: estimates and projections for 166 countries.. *The Lancet Planetary Health*, Volumen 3, p. 9.
- CHRISTENSEN, A. P., GARRIDO, L. E. & GOLINO, H., 2020.** Unique Variable Analysis:A Novel Approach for Detecting Redundant Variables in Multivariate Data. pp. 1-36.
- CONGACHA, J., 2016.** *Estadística Aplicada a la educación con actividades de aprendizaje*. Segunda ed. Riobamba, Ecuador: editorial académica española.
- COUCKUYT, I. Y OTROS, 2012.** Blind Kriging: Implementation and performance analysis. *Advances in Engineering Software*, Volumen 49, pp. 1-13.
- CUBILLOS, A. & ESTENSSORO, F., 2011.** Los desafíos del crecimiento y desarrollo en el contexto del cambio climático. *idea*, Volumen 2, p. 103.
- DELMELLE, E. C. & THILL, J.-C., 2008.** *Urban bicyclists: spatial analysis of adult and youth traffic hazard intensity*, s.l.: s.n.
- DÍAZ MONROY, L. G. & MORALES RIVERA, M. A., 2009.** *Análisis estadístico de datos categóricos*. Primera ed. Bogotá, Colombia: Universidad Nacional de Colombia.
- ERDOGAN, S., YILMAZ, I., BAYBURA, T. & GULLU, M., 2008.** Geographical information systems aided traffic accident analysis system case study: city of Afyonkarahisar.. *Anal. Prev*, 40(1), pp. 174-181.
- ESCAMILLA, M., 2020.** Aplicación básica de los métodos científicos. *Universidad Estatal del estado de Hidalgo*, pp. 2-3.

- FERNÁNDEZ, F., 2012.** *Meteorología y climatología.* s.l.:s.n.
- GARCÍA, F., 2012.** Revista Índice. Enero.
- GATTINONI, N. N., BOCA, R. T., REBELLA, C. M. & DI BELLA, C. M., 2011.** *Comparación entre observaciones meteorológicas obtenidas de estaciones convencionales y automáticas a partir de la estimación de parámetros estadísticos.* [Arte] (INTA).
- GOLMAN AJ, D. K. M. L. S. J., 2014.** Injury prediction in a side impact crash using human body model simulation. Volumen 64, pp. 1-8.
- GONZALEZ, J., 2013.** *Introducción a la Meteorología y a Climatología.* Ecuador: s.n.
- GUNST, R. F. & MASON, R. L., 1980.** *Regression Analysis and Its Application: a Data-Oriented Approach.* 1st Edition ed. Boca Raton: CRC Press.
- HANKE, J. & WICHERM, D., 2010.** *Pronósticos en los Negocios.* Novena edición.. Mexico: s.n.
- LÓPEZ, J. M. M., 2007.** *Estadística Descriptiva.* Madrid: Paraninfo.
- MONTESINOS, A., 2011.** *Estudio del AIC y BIC en la selección de modelos de vida con datos censurados.* [Arte] (Centro de Investigación en Matemáticas A.C.)
- NAVARRA, M., 2019.** *Meteorología y climatología de Navarra.* Navarra: s.n.
- NAVARRO, J. D. L. H., 2003.** *Estadística Aplicada.* Tercera ed. Madrid: Diaz de Santos.
- PEÑA, D., 2002.** *Análisis de datos multivariantes.* s.l.:s.n.
- PEREZ, R., 2010.** *Nociones Básicas de Estadística.* 1986 ed. s.l.:Dpto de Economía Aplicada. Universidad de Oviedo.
- PULIDO, H. & SALAZAR, R., 2008.** *Análisis y Diseño de experimentos.* Segunda ed. México: McGraw-Hill/Ineramericana editores, S.A. de C.V.
- RENDÓN MACÍAS, M. E., VILLASÍS-KEEVER, M. Á. & MIRANDA-NOVALES, M. G., 2016.** Estadística descriptiva. *Revista Alergia México*, 63(4), pp. 397-407.
- RODRÍGUEZ, M. L. M. & RODRÍGUEZ, J. C. H., 2009.** *Sistemas de Información Geográfica,* s.l.: s.n.
- S.PULUGURTHA, S., K.KRISHNAKUMAR, V. & S.NAMBISAN, S., 2007.** New methods to identify and rank high pedestrian crash zones: an illustration.. *Anal. Prev*, 39(4), pp. 800-811.
- SALDÍVAR, S. M., 2008.** *Estadística Descriptiva.* México: Instituto Politécnico Nacional.
- SAMPIERI, R. H., COLLADO, C. F. & LUCIO, M. D. P. B., 2014.** *Metodología de la Investigación.* Sexta ed. México: McGRAW-HILL.
- SILVERMAN, B., 1998.** *Density Estimation for Statistics and Data Analysis* Chapman Hall, London, UK, s.l.: s.n.
- SPIEGEL, M. R. & STEPHENS, L. J., 2009.** *Estadística Schaum.* Cuarta ed. México: MCGraw-Hill/Interamericana Editores, S.A. de C.V..
- Supurrier, W., 1996. Impacto sobre el gasto público. *ILDIS*, Volumen 1, p. 155.
- T.C. BAILEY, A., 1995.** *Gatrell Interactive Spatial Data Analysis* Longman Scientific, Harlow, UK , s.l.: s.n.
- VÉLEZ, E., 2016.** *Introducción series temporales.* Mexico: s.n.
- VIDMA, C. D. L. P., 2018.** *Estadística descriptiva e inferencial.* Madrid, España: Ediciones IDT.
- VILLAVICENCIO, J., 2016.** *Introducción a series de tiempo.* Puerto Rico: s.n.
- WHO, W., 2013.** Global status report on road safety 2013: supporting a decade of action. p. 5.
- XIE, Z. & YAN, J., 2008.** Kernel density estimation of traffic accidents in a network space *Comput. Environ Urban Syst*, 35(5), pp. 396-406.
- YAHAYA, A. S., AHMAD, F. & TIING, L. Y., 2014.** Prediction of Sand Thickness Using Ordinary Kriging. *Journal of Civil Engineering Research*, 4(3), pp. 36-40.



ANEXOS

ANEXO A: LINK DE REPOSITORIO DE LA MATRIZ DE DATOS

<https://liveespochedu->

my.sharepoint.com/:w:/g/personal/estaciones_espoch_espoch_edu_ec/EfyOLP7fmu9Pjip0_BsYc0wBr8xAj3QkPO8PD8iB9_Qh1Q?e=3EAIBR

ANEXO B: CÓDIGO EN R

```
library(tidyverse)
library(lubridate)
library(moments)
```

```
load("D:/Documents/OneDrive - ESCUELA SUPERIOR POLITECNICA DE CHIMBORAZO/7. R - copia/CODIGO R/science planners/Data/Belen P/data_GEAA.RData")
```

```
#####
##### 1. Alao #####
#####
```

```
#### Preparacion de los datos ----
```

```
#data.13 <- read.csv('Data/Belen P/ALAO/ALAO-2013-L2-ANUAL/L2131219 - L2131231.csv',sep="," ,header=F) # Pocos datos
data.14 <- read.csv('Data/Belen P/ALAO/ALAO-2014-L2-ANUAL/L2140128 - L2141231.csv',sep="," ,header=F)
data.15 <- read.csv('Data/Belen P/ALAO/ALAO-2015-L2-ANUAL/L2150101 - L2151231.csv',sep="," ,header=F)
data.16 <- read.csv('Data/Belen P/ALAO/ALAO-2016-L2-ANUAL/L2160101 - L2161231.csv',sep="," ,header=F)
data.17 <- read.csv('Data/Belen P/ALAO/ALAO-2017-L2-ANUAL/L2170101 - L2171128.csv',sep="," ,header=F)
data.18 <- read.csv('Data/Belen P/ALAO/ALAO-2018-L2-ANUAL/L2180101 - L2180814.csv',sep="," ,header=F)
data.19 <- read.csv('Data/Belen P/ALAO/ALAO-2019-L2-ANUAL/L2190101 - L2191231.csv',sep="," ,header=F)
data.20 <- read.csv('Data/Belen P/ALAO/ALAO-2020-L2-ANUAL/L2200101 - L2201231.csv',sep="," ,header=F)
data.21 <- read.csv('Data/Belen P/ALAO/ALAO-2021-L2-ANUAL/L2210101 - L2211206.csv',sep="," ,header=F)
```

```
organizacion <- function(data){
```

```
##### Carga de informacion ----
```

```
u.d <- which(data[2,] == "date"); u.d
u.t <- which(data[2,] == "time"); u.t
u.1 <- which(data[1,] == "Stat_TA_1h" & data[2,] == "Avg"); u.1
u.2 <- which(data[1,] == "Stat_RH_1h" & data[2,] == "Avg"); u.2
u.3 <- which(data[1,] == "Stat_PA_1h" & data[2,] == "Avg"); u.3
u.4 <- which(data[1,] == "Stat_SR_Dif_1h" & data[2,] == "Avg"); u.4
if ( sum(data[1,] == "GenWind_1h_SDI12", na.rm = T) >= 1 ) {
  u.5 <- which(data[1,] == "GenWind_1h_SDI12" & data[2,] == "SpdAvg"); u.5
} else {
  u.5 <- which(data[1,] == "GenWind_1h" & data[2,] == "SpdAvg"); u.5
}
```

```
data.1 <- data[-c(1,2),c(u.d,u.t,u.1,u.2,u.3,u.4,u.5)]
row.names(data.1) <- NULL
colnames(data.1) <- c("date","time","ta","h","pa","rsd","vv")
#View(data.1)
str(data.1)
##### FIN Carga de informacion ----
```

```
#### Modificando el formato de fechas ----
```

```
#anio <- paste("20",substr(data.1$date,7,8),sep = "")
anio <- substr(data.1$date,nchar(data.1$date)-1,nchar(data.1$date))
mes <- substr(data.1$date,1,2)
mes <- str_remove_all(mes, "/")
dia <- substr(data.1$date,nchar(data.1$date)-4,nchar(data.1$date)-3)
dia <- str_remove_all(dia, "/")
```

```
h <- substr(data.1$time,1,nchar(data.1$time)-3)
am.pm <- substr(data.1$time,nchar(data.1$time)-1,nchar(data.1$time))
hora <- vector(); minuto <- vector(); segundo <- vector()
for (i in 1:length(h)) {
  if (nchar(h[i]) == 7) {
    if (am.pm[i] == "PM" ) {
      hora[i] <- as.numeric(substr(h[i],1,1)) + 12
    } else {
      hora[i] <- as.numeric(substr(h[i],1,1))
    }
    minuto[i] <- substr(h[i],3,4)
```

```

segundo[i] <- "00"
#segundo[i] <- substr(h[i],6,7) # Valor exacto
} else {

if (am.pm[i] == "PM" ) {
  if (as.numeric(substr(h[i],1,2)) == 12) {
    hora[i] <- as.numeric(substr(h[i],1,2))
  } else {
    hora[i] <- as.numeric(substr(h[i],1,2)) + 12
  }
} else {
  if (as.numeric(substr(h[i],1,2)) == 12) {
    hora[i] <- as.numeric(substr(h[i],1,2)) - 12
  } else {
    hora[i] <- as.numeric(substr(h[i],1,2))
  }
}
minuto[i] <- substr(h[i],4,5)
segundo[i] <- "00"
#segundo[i] <- substr(h[i],7,8) # valor exacto
}
}
data.1$date <- paste(dia, "-", mes, "-", anio, sep = "")
data.1$time <- paste(hora, ":", minuto, ":", segundo, sep = "")
library(tidyverse)
library(lubridate)
fh <- dmy_hms(paste(data.1$date, data.1$time)) # Transforma data and time
data.1 <- data.frame(ta = as.numeric(data.1$ta),
                    h = as.numeric(data.1$h),
                    pa = as.numeric(data.1$pa),
                    rsd = as.numeric(data.1$rsd),
                    vv = as.numeric(data.1$vv))
data.1 <- as.tibble(data.1)
data.1 <- data.1 %>% mutate(fh = fh)
data.1
} # Organiza la informacion para unir las matrices de informacion
#data.13 <- organizacion(data.13)
data.14 <- organizacion(data.14)
data.15 <- organizacion(data.15)
data.16 <- organizacion(data.16)
data.17 <- organizacion(data.17)
data.18 <- organizacion(data.18)
data.19 <- organizacion(data.19)
data.20 <- organizacion(data.20)
data.21 <- organizacion(data.21)
#data <- rbind(data.13, data.14, data.15, data.16, data.17, data.18, data.19, data.20, data.21)
data <- rbind(data.14, data.15, data.16, data.17, data.18, data.19, data.20, data.21)
data <- data[-which(duplicated(data$fh)),] # Quita los registros duplicados
data.alao.sin <- data
# rellenando los ceros, cuando existen NAs
horas_completas <- data.frame(
  fh = seq(
    floor_date(min(data$fh), unit = 'hour'),
    floor_date(max(data$fh), unit = 'hour'),
    by = 'hour'
  ))
dim(horas_completas)
# left join
data <- left_join(x = horas_completas, y = data, by = "fh", copy = TRUE)
dim(data)
dim(data)[1] == dim(horas_completas)[1] # Prueba de dimensiones
# Imputacion
summary(data) # Cantidad de NAs
dim(data)[1]; sum(is.na(data$ta)); round(sum(is.na(data$ta))/dim(data)[1] * 100, 2)
imputacion.NA <- function(y){
  if(length(na.omit(y)) >= 2 & length(na.omit(y)) <= (length(y) - 1) ) {
    data <- data.frame(x=1:length(y), y=y)
    lr <- lm(y ~ x , data = data)
    data$y[is.na(data$y)] <- lr$coefficients[1] + data$x[is.na(data$y)] * lr$coefficients[2]
    y <- data$y
  } else {
    return(y)
  }
}
return(y)
} # Imputacion mediante regresion lineal simple para NAs
data[,-1] <- apply(data[,-1], 2, imputacion.NA)

```



```

summary(data)
data.alao <- data
objects()
rm(data, data.14, data.15, data.16, data.17, data.18, data.19, data.20, data.21, horas_completas, imputacion.NA, organizacion)
objects()
#### FIN Preparacion de los datos ----

#####
##### 2. Epoch #####
#####

#### Preparacion de los datos ----
#data.13 <- read.csv('Data/Belen P/ESPOCH/ESPOCH-2013-L2-ANUAL/L2131218 - L2131231.csv',sep=";",header=F) # Pocos
datos
#data.14 <- read.csv('Data/Belen P/ESPOCH/ESPOCH-2014-L2-ANUAL/L2140101 - L2140122.csv',sep=";",header=F) # Pocos
datos
data.15 <- read.csv('Data/Belen P/ESPOCH/ESPOCH-2015-L2-ANUAL/L2150101 - L2151231.csv',sep=";",header=F)
data.16 <- read.csv('Data/Belen P/ESPOCH/ESPOCH-2016-L2-ANUAL/L2160101 - L2161231.csv',sep=";",header=F)
data.17 <- read.csv('Data/Belen P/ESPOCH/ESPOCH-2017-L2-ANUAL/L2170101 - L2171231.csv',sep=";",header=F)
data.18 <- read.csv('Data/Belen P/ESPOCH/ESPOCH-2018-L2-ANUAL/L2180101 - L2181231.csv',sep=";",header=F)
data.19 <- read.csv('Data/Belen P/ESPOCH/ESPOCH-2019-L2-ANUAL/L2190101 - L2191231.csv',sep=";",header=F)
data.20 <- read.csv('Data/Belen P/ESPOCH/ESPOCH-2020-L2-ANUAL/L2200101 - L2201231.csv',sep=";",header=F)
data.21 <- read.csv('Data/Belen P/ESPOCH/ESPOCH-2021-L2-ANUAL/L2210101 - L2211214.csv',sep=";",header=F)
organizacion <- function(data){
  #### Carga de informacion ----
  u.d <- which(data[2,] == "date"); u.d
  u.t <- which(data[2,] == "time"); u.t
  u.1 <- which(data[1,] == "Stat_TA_1h" & data[2,] == "Avg"); u.1
  u.2 <- which(data[1,] == "Stat_RH_1h" & data[2,] == "Avg"); u.2
  u.3 <- which(data[1,] == "Stat_PA_1h" & data[2,] == "Avg"); u.3
  u.4 <- which(data[1,] == "Stat_SR_Dif_1h" & data[2,] == "Avg"); u.4
  if ( sum(data[1,] == "GenWind_1h_SDI12", na.rm = T) >= 1 ) {
    u.5 <- which(data[1,] == "GenWind_1h_SDI12" & data[2,] == "SpdAvg"); u.5
  } else {
    u.5 <- which(data[1,] == "GenWind_1h" & data[2,] == "SpdAvg"); u.5
  }
  data.1 <- data[-c(1,2),c(u.d,u.t,u.1,u.2,u.3,u.4,u.5)]
  row.names(data.1) <- NULL
  colnames(data.1) <- c("date","time","ta","h","pa","rsd","vv")
  #View(data.1)
  str(data.1)
  #### FIN Carga de informacion ----

  #### Modificando el formato de fechas ----
  #anio <- paste("20",substr(data.1$date,7,8),sep = "")
  anio <- substr(data.1$date,nchar(data.1$date)-1,nchar(data.1$date))
  mes <- substr(data.1$date,1,2)
  mes <- str_remove_all(mes, "/")
  dia <- substr(data.1$date,nchar(data.1$date)-4,nchar(data.1$date)-3)
  dia <- str_remove_all(dia, "/")

  h <- substr(data.1$time,1,nchar(data.1$time)-3)
  am.pm <- substr(data.1$time,nchar(data.1$time)-1,nchar(data.1$time))
  hora <- vector(); minuto <- vector(); segundo <- vector()
  for (i in 1:length(h)) {
    if (nchar(h[i]) == 7) {
      if (am.pm[i] == "PM" ) {
        hora[i] <- as.numeric(substr(h[i],1,1)) + 12
      } else {
        hora[i] <- as.numeric(substr(h[i],1,1))
      }
      minuto[i] <- substr(h[i],3,4)
      segundo[i] <- "00"
      #segundo[i] <- substr(h[i],6,7) # Valor exacto
    } else {

      if (am.pm[i] == "PM" ) {
        if (as.numeric(substr(h[i],1,2)) == 12) {
          hora[i] <- as.numeric(substr(h[i],1,2))
        } else {
          hora[i] <- as.numeric(substr(h[i],1,2)) + 12
        }
      } else {
        if (as.numeric(substr(h[i],1,2)) == 12) {
          hora[i] <- as.numeric(substr(h[i],1,2)) - 12
        }
      }
    }
  }
}

```

```

    } else {
      hora[i] <- as.numeric(substr(h[i],1,2))
    }
  }
  minuto[i] <- substr(h[i],4,5)
  segundo[i] <- "00"
  #segundo[i] <- substr(h[i],7,8) # valor exacto
}
}
data.1$date <- paste(dia,"-",mes,"-",anio,sep = "")
data.1$time <- paste(hora,":",minuto,":",segundo,sep = "")
library(tidyverse)
library(lubridate)
fh <- dmy_hms(paste(data.1$date, data.1$time)) # Transforma data and time
data.1 <- data.frame(ta = as.numeric(data.1$ta),
  h = as.numeric(data.1$h),
  pa = as.numeric(data.1$pa),
  rsd = as.numeric(data.1$rsd),
  vv = as.numeric(data.1$vv))
data.1 <- as.tibble(data.1)
data.1 <- data.1 %>% mutate(fh = fh)
data.1
} # Organiza la informacion para unir las matrices de informacion
organizacion.previa <- function(data){
  ##### Carga de informacion ----
  u.d <- which(data[2,] == "date"); u.d
  u.t <- which(data[2,] == "time"); u.t
  u.1 <- which(data[1,] == "Stat_TA_1h" & data[2,] == "Avg"); u.1
  u.2 <- which(data[1,] == "Stat_RH_1h" & data[2,] == "Avg"); u.2
  u.3 <- which(data[1,] == "Stat_PA_1h" & data[2,] == "Avg"); u.3
  u.4 <- which(data[1,] == "Stat_SR_Dif_1h" & data[2,] == "Avg"); u.4
  if ( sum(data[1,] == "GenWind_1h_SDI12", na.rm = T) >= 1 ) {
    u.5 <- which(data[1,] == "GenWind_1h_SDI12" & data[2,] == "SpdAvg"); u.5
  } else {
    u.5 <- which(data[1,] == "GenWind_1h" & data[2,] == "SpdAvg"); u.5
  }
  data.1 <- data[-c(1,2),c(u.d,u.t,u.1,u.2,u.3,u.4,u.5)]
  row.names(data.1) <- NULL
  colnames(data.1) <- c("date","time","ta","h","pa","rsd","vv")
  data.1
}
organizacion.fin <- function(data){
  data.1 <- data

  ##### Modificando el formato de fechas ----
  #anio <- paste("20",substr(data.1$date,7,8),sep = "")
  anio <- substr(data.1$date,nchar(data.1$date)-1,nchar(data.1$date))
  mes <- substr(data.1$date,1,2)
  mes <- str_remove_all(mes, "/")
  dia <- substr(data.1$date,nchar(data.1$date)-4,nchar(data.1$date)-3)
  dia <- str_remove_all(dia, "/")

  h <- substr(data.1$time,1,nchar(data.1$time)-3)
  am.pm <- substr(data.1$time,nchar(data.1$time)-1,nchar(data.1$time))
  hora <- vector(); minuto <- vector(); segundo <- vector()
  for (i in 1:length(h)) {
    if (nchar(h[i]) == 7) {
      if (am.pm[i] == "PM" ) {
        hora[i] <- as.numeric(substr(h[i],1,1)) + 12
      } else {
        hora[i] <- as.numeric(substr(h[i],1,1))
      }
    }
    minuto[i] <- substr(h[i],3,4)
    segundo[i] <- "00"
    #segundo[i] <- substr(h[i],6,7) # Valor exacto
  } else {

    if (am.pm[i] == "PM" ) {
      if (as.numeric(substr(h[i],1,2)) == 12) {
        hora[i] <- as.numeric(substr(h[i],1,2))
      } else {
        hora[i] <- as.numeric(substr(h[i],1,2)) + 12
      }
    } else {
      if (as.numeric(substr(h[i],1,2)) == 12) {
        hora[i] <- as.numeric(substr(h[i],1,2)) - 12
      }
    }
  }
}

```

```

    } else {
      hora[i] <- as.numeric(substr(h[i],1,2))
    }
  }
  minuto[i] <- substr(h[i],4,5)
  segundo[i] <- "00"
  #segundo[i] <- substr(h[i],7,8) # valor exacto
}
}
data.1$date <- paste(dia,"-",mes,"-",anio,sep = "")
data.1$time <- paste(hora,":",minuto,":",segundo,sep = "")
library(tidyverse)
library(lubridate)
fh <- dmy_hms(paste(data.1$date, data.1$time)) # Transforma data and time
data.1 <- data.frame(ta = as.numeric(data.1$ta),
                    h = as.numeric(data.1$h),
                    pa = as.numeric(data.1$pa),
                    rsd = as.numeric(data.1$rsd),
                    vv = as.numeric(data.1$vv))
data.1 <- as.tibble(data.1)
data.1 <- data.1 %>% mutate(fh = fh)
data.1
}
data.15 <- organizacion(data.15)
data.16 <- organizacion(data.16)
#data.17 <- organizacion.previa(data.17)
#openxlsx::write.xlsx(data.17,'Data/Belen P/ESPOCH/ESPOCH-2017-L2-ANUAL/data.17.xlsx')
data.17 <- openxlsx::read.xlsx('Data/Belen P/ESPOCH/ESPOCH-2017-L2-ANUAL/data.17.xlsx')
data.17 <- organizacion.fin(data.17)
data.18 <- organizacion(data.18)
data.19 <- organizacion(data.19)
data.20 <- organizacion(data.20)
data.21 <- organizacion(data.21)
data <- rbind(data.15, data.16, data.17, data.18, data.19, data.20, data.21)
if ( sum(duplicated(data$fh)) != 0 ) {
  data <- data[-which(duplicated(data$fh)),] # Quita los registros duplicados
}
data$vv[which.max(data$vv)] <- median(data$vv) # Cambia el valor extremo 729.697 por la mediana
data.esepoch.sin <- data
# rellenando cuando existen NAs
horas_completas <- data.frame(
  fh = seq(
    floor_date(min(data$fh), unit = 'hour'),
    floor_date(max(data$fh), unit = 'hour'),
    by = 'hour'
  ))
dim(horas_completas)
# left join
data <- left_join(x = horas_completas, y = data, by = "fh", copy = TRUE)
dim(data)
dim(data)[1] == dim(horas_completas)[1] # Prueba de dimensiones
# Imputacion
summary(data) # Cantidad de NAs
dim(data)[1]; sum(is.na(data$ta)); round(sum(is.na(data$ta))/dim(data)[1] * 100, 2)
imputacion.NA <- function(y){
  if(length(na.omit(y)) >= 2 & length(na.omit(y)) <= (length(y) - 1) ) {
    data <- data.frame(x=1:length(y), y=y)
    lr <- lm(y ~ x , data = data)
    data$y[is.na(data$y)] <- lr$coefficients[1] + data$x[is.na(data$y)] * lr$coefficients[2]
    y <- data$y
  } else {
    return(y)
  }
}
return(y)
} # Imputacion mediante regresion lineal simple para NAs
data[,-1] <- apply(data[,-1], 2, imputacion.NA)
summary(data)
data.esepoch <- data
objects()
rm(data, data.15, data.16, data.17, data.18, data.19, data.20, data.21, horas_completas, imputacion.NA, organizacion,
  organizacion.fin, organizacion.previa)
objects()
#### FIN Preparacion de los datos ----

```

```
#####
```

```

##### 3. Matus #####
#####

#### Preparacion de los datos ----
data.14 <- read.csv('Data/Belen P/MATUS/MATUS-2014-L2-ANUAL/L2141126 - L2141231.csv',sep="," ,header=F)
data.15 <- read.csv('Data/Belen P/MATUS/MATUS-2015-L2-ANUAL/L2150101 - L2151231.csv',sep="," ,header=F)
data.16 <- read.csv('Data/Belen P/MATUS/MATUS-2016-L2-ANUAL/L2160101 - L2161231.csv',sep="," ,header=F)
data.17 <- read.csv('Data/Belen P/MATUS/MATUS-2017-L2-ANUAL/L2170101 - L2171231.csv',sep="," ,header=F)
data.18 <- read.csv('Data/Belen P/MATUS/MATUS-2018-L2-ANUAL/L2180101 - L2181219.csv',sep="," ,header=F)
data.19 <- read.csv('Data/Belen P/MATUS/MATUS-2019-L2-ANUAL/L2190101 - L2191231.csv',sep="," ,header=F)
data.20 <- read.csv('Data/Belen P/MATUS/MATUS-2020-L2-ANUAL/L2200101 - L2201231.csv',sep="," ,header=F)
data.21 <- read.csv('Data/Belen P/MATUS/MATUS-2021-L2-ANUAL/L2210101 - L2211208.csv',sep="," ,header=F)
organizacion <- function(data){
  #### Carga de informacion ----
  u.d <- which(data[2,] == "date"); u.d
  u.t <- which(data[2,] == "time"); u.t
  u.1 <- which(data[1,] == "Stat_TA_1h" & data[2,] == "Avg"); u.1
  u.2 <- which(data[1,] == "Stat_RH_1h" & data[2,] == "Avg"); u.2
  u.3 <- which(data[1,] == "Stat_PA_1h" & data[2,] == "Avg"); u.3
  u.4 <- which(data[1,] == "Stat_SR_Dif_1h" & data[2,] == "Avg"); u.4
  if ( sum(data[1,] == "GenWind_1h_SDI12", na.rm = T) >= 1 ) {
    u.5 <- which(data[1,] == "GenWind_1h_SDI12" & data[2,] == "SpdAvg"); u.5
  } else {
    u.5 <- which(data[1,] == "GenWind_1h" & data[2,] == "SpdAvg"); u.5
  }
  data.1 <- data[-c(1,2),c(u.d,u.t,u.1,u.2,u.3,u.4,u.5)]
  row.names(data.1) <- NULL
  colnames(data.1) <- c("date","time","ta","h","pa","rsd","vv")
  #View(data.1)
  str(data.1)
  #### FIN Carga de informacion ----

  #### Modificando el formato de fechas ----
  #anio <- paste("20",substr(data.1$date,7,8),sep = "")
  anio <- substr(data.1$date,nchar(data.1$date)-1,nchar(data.1$date))
  mes <- substr(data.1$date,1,2)
  mes <- str_remove_all(mes, "/")
  dia <- substr(data.1$date,nchar(data.1$date)-4,nchar(data.1$date)-3)
  dia <- str_remove_all(dia, "/")

  h <- substr(data.1$time,1,nchar(data.1$time)-3)
  am.pm <- substr(data.1$time,nchar(data.1$time)-1,nchar(data.1$time))
  hora <- vector(); minuto <- vector(); segundo <- vector()
  for (i in 1:length(h)) {
    if (nchar(h[i]) == 7) {
      if (am.pm[i] == "PM" ) {
        hora[i] <- as.numeric(substr(h[i],1,1)) + 12
      } else {
        hora[i] <- as.numeric(substr(h[i],1,1))
      }
      minuto[i] <- substr(h[i],3,4)
      segundo[i] <- "00"
      #segundo[i] <- substr(h[i],6,7) # Valor exacto
    } else {
      if (am.pm[i] == "PM" ) {
        if (as.numeric(substr(h[i],1,2)) == 12) {
          hora[i] <- as.numeric(substr(h[i],1,2))
        } else {
          hora[i] <- as.numeric(substr(h[i],1,2)) + 12
        }
      } else {
        if (as.numeric(substr(h[i],1,2)) == 12) {
          hora[i] <- as.numeric(substr(h[i],1,2)) - 12
        } else {
          hora[i] <- as.numeric(substr(h[i],1,2))
        }
      }
      minuto[i] <- substr(h[i],4,5)
      segundo[i] <- "00"
      #segundo[i] <- substr(h[i],7,8) # valor exacto
    }
  }
  data.1$date <- paste(dia,"-",mes,"-",anio,sep = "")
  data.1$time <- paste(hora,":",minuto,":",segundo,sep = "")
  library(tidyverse)

```

```

library(lubridate)
fh <- dmy_hms(paste(data.1$date, data.1$time)) # Transforma data and time
data.1 <- data.frame(ta = as.numeric(data.1$ta),
                    h = as.numeric(data.1$h),
                    pa = as.numeric(data.1$pa),
                    rsd = as.numeric(data.1$rsd),
                    vv = as.numeric(data.1$vv))
data.1 <- as.tibble(data.1)
data.1 <- data.1 %>% mutate(fh = fh)
data.1
} # Organiza la informacion para unir las matrices de informacion
organizacion.previa <- function(data){
#### Carga de informacion ----
u.d <- which(data[2,] == "date"); u.d
u.t <- which(data[2,] == "time"); u.t
u.1 <- which(data[1,] == "Stat_TA_1h" & data[2,] == "Avg"); u.1
u.2 <- which(data[1,] == "Stat_RH_1h" & data[2,] == "Avg"); u.2
u.3 <- which(data[1,] == "Stat_PA_1h" & data[2,] == "Avg"); u.3
u.4 <- which(data[1,] == "Stat_SR_Dif_1h" & data[2,] == "Avg"); u.4
if (sum(data[1,] == "GenWind_1h_SDI12", na.rm = T) >= 1) {
  u.5 <- which(data[1,] == "GenWind_1h_SDI12" & data[2,] == "SpdAvg"); u.5
} else {
  u.5 <- which(data[1,] == "GenWind_1h" & data[2,] == "SpdAvg"); u.5
}
data.1 <- data[-c(1,2),c(u.d,u.t,u.1,u.2,u.3,u.4,u.5)]
row.names(data.1) <- NULL
colnames(data.1) <- c("date", "time", "ta", "h", "pa", "rsd", "vv")
data.1
}
organizacion.fin <- function(data){
data.1 <- data

#### Modificando el formato de fechas ----
#anio <- paste("20",substr(data.1$date,7,8),sep = "")
anio <- substr(data.1$date,nchar(data.1$date)-1,nchar(data.1$date))
mes <- substr(data.1$date,1,2)
mes <- str_remove_all(mes, "/")
dia <- substr(data.1$date,nchar(data.1$date)-4,nchar(data.1$date)-3)
dia <- str_remove_all(dia, "/")

h <- substr(data.1$time,1,nchar(data.1$time)-3)
am.pm <- substr(data.1$time,nchar(data.1$time)-1,nchar(data.1$time))
hora <- vector(); minuto <- vector(); segundo <- vector()
for (i in 1:length(h)) {
  if (nchar(h[i]) == 7) {
    if (am.pm[i] == "PM" ) {
      hora[i] <- as.numeric(substr(h[i],1,1)) + 12
    } else {
      hora[i] <- as.numeric(substr(h[i],1,1))
    }
    minuto[i] <- substr(h[i],3,4)
    segundo[i] <- "00"
    #segundo[i] <- substr(h[i],6,7) # Valor exacto
  } else {
    if (am.pm[i] == "PM" ) {
      if (as.numeric(substr(h[i],1,2)) == 12) {
        hora[i] <- as.numeric(substr(h[i],1,2))
      } else {
        hora[i] <- as.numeric(substr(h[i],1,2)) + 12
      }
    } else {
      if (as.numeric(substr(h[i],1,2)) == 12) {
        hora[i] <- as.numeric(substr(h[i],1,2)) - 12
      } else {
        hora[i] <- as.numeric(substr(h[i],1,2))
      }
    }
    minuto[i] <- substr(h[i],4,5)
    segundo[i] <- "00"
    #segundo[i] <- substr(h[i],7,8) # valor exacto
  }
}
data.1$date <- paste(dia, "-", mes, "-", anio, sep = "")
data.1$time <- paste(hora, ":", minuto, ":", segundo, sep = "")
library(tidyverse)

```

```

library(lubridate)
fh <- dmy_hms(paste(data.1$date, data.1$time)) # Transforma data and time
data.1 <- data.frame(ta = as.numeric(data.1$ta),
                    h = as.numeric(data.1$h),
                    pa = as.numeric(data.1$pa),
                    rsd = as.numeric(data.1$rsd),
                    vv = as.numeric(data.1$vv))
data.1 <- as.tibble(data.1)
data.1 <- data.1 %>% mutate(fh = fh)
data.1
}
data.14 <- organizacion(data.14)
data.15 <- organizacion(data.15)
data.16 <- organizacion(data.16)
data.17 <- organizacion(data.17)
data.18 <- organizacion(data.18)
data.19 <- organizacion(data.19)
data.20 <- organizacion(data.20)
data.21 <- organizacion(data.21)
data <- rbind(data.14, data.15, data.16, data.17, data.18, data.19, data.20, data.21)
if ( sum(duplicated(data$fh)) != 0 ) {
  data <- data[-which(duplicated(data$fh)),] # Quita los registros duplicados
}
data.esepoch.sin <- data
# rellenando cuando existen NAs
horas_completas <- data.frame(
  fh = seq(
    floor_date(min(data$fh), unit = 'hour'),
    floor_date(max(data$fh), unit = 'hour'),
    by = 'hour'
  ))
dim(horas_completas)
# left join
data <- left_join(x = horas_completas, y = data, by = "fh", copy = TRUE)
dim(data)
dim(data)[1] == dim(horas_completas)[1] # Prueba de dimensiones
data.matus.sin <- data
# Imputacion
summary(data) # Cantidad de NAs
dim(data)[1]; sum(is.na(data$ta)); round(sum(is.na(data$ta))/dim(data)[1] * 100, 2)
imputacion.NA <- function(y){
  if(length(na.omit(y)) >= 2 & length(na.omit(y)) <= (length(y) - 1) ) {
    data <- data.frame(x=1:length(y), y=y)
    lr <- lm(y ~ x, data = data)
    data$y[is.na(data$y)] <- lr$coefficients[1] + data$x[is.na(data$y)] * lr$coefficients[2]
    y <- data$y
  } else {
    return(y)
  }
  return(y)
} # Imputacion mediante regresion lineal simple para NAs
data[,-1] <- apply(data[,-1], 2, imputacion.NA)
summary(data)
data.matus <- data
objects()
rm(data, data.14, data.15, data.16, data.17, data.18, data.19, data.20, data.21, horas_completas, imputacion.NA,
    organizacion, organizacion.fin, organizacion.previa)
objects()
#### FIN Preparacion de los datos ----

```

```

#####
##### 4. Multitud #####
#####

```

```

#### Preparacion de los datos ----

```

```

data.14 <- read.csv('Data/Belen P/MULTITUD/MULTITUD-2014-L2-ANUAL/L2140124 - L2141231.csv',sep=";",header=F)
data.15 <- read.csv('Data/Belen P/MULTITUD/MULTITUD-2015-L2-ANUAL/L2150101 - L2151231.csv',sep=";",header=F)
data.16 <- read.csv('Data/Belen P/MULTITUD/MULTITUD-2016-L2-ANUAL/L2160101 - L2161231.csv',sep=";",header=F)
data.17.1 <- read.csv('Data/Belen P/MULTITUD/MULTITUD-2017-L2-ANUAL/L2170101 - L2170801.csv',sep=";",header=F)
data.17.2 <- read.csv('Data/Belen P/MULTITUD/MULTITUD-2017-L2-ANUAL/L2170802 - L2171231.csv',sep=";",header=F)
data.18 <- read.csv('Data/Belen P/MULTITUD/MULTITUD-2018-L2-ANUAL/L2180101 - L2181231.csv',sep=";",header=F)
data.19.1 <- read.csv('Data/Belen P/MULTITUD/MULTITUD-2019-L2-ANUAL/L2190101 - L2190328.csv',sep=";",header=F)
data.19.2 <- read.csv('Data/Belen P/MULTITUD/MULTITUD-2019-L2-ANUAL/L2190329 - L2191231.csv',sep=";",header=F)
data.20 <- read.csv('Data/Belen P/MULTITUD/MULTITUD-2020-L2-ANUAL/L2200101 - L2201231.csv',sep=";",header=F)
data.21 <- read.csv('Data/Belen P/MULTITUD/MULTITUD-2021-L2-ANUAL/L2210101 - L2211221.csv',sep=";",header=F)

```

```

organizacion <- function(data){
#### Carga de informacion ----
u.d <- which(data[2,] == "date"); u.d
u.t <- which(data[2,] == "time"); u.t
u.1 <- which(data[1,] == "Stat_TA_1h" & data[2,] == "Avg"); u.1
u.2 <- which(data[1,] == "Stat_RH_1h" & data[2,] == "Avg"); u.2
u.3 <- which(data[1,] == "Stat_PA_1h" & data[2,] == "Avg"); u.3
u.4 <- which(data[1,] == "Stat_SR_Dif_1h" & data[2,] == "Avg"); u.4
if ( sum(data[1,] == "GenWind_1h_SDI12", na.rm = T) >= 1 ) {
  u.5 <- which(data[1,] == "GenWind_1h_SDI12" & data[2,] == "SpdAvg"); u.5
} else {
  u.5 <- which(data[1,] == "GenWind_1h" & data[2,] == "SpdAvg"); u.5
}
data.1 <- data[-c(1,2),c(u.d,u.t,u.1,u.2,u.3,u.4,u.5)]
row.names(data.1) <- NULL
colnames(data.1) <- c("date","time","ta","h","pa","rsd","vv")
#View(data.1)
str(data.1)
#### FIN Carga de informacion ----

#### Modificando el formato de fechas ----
#anio <- paste("20",substr(data.1$date,7,8),sep = "")
anio <- substr(data.1$date,nchar(data.1$date)-1,nchar(data.1$date))
mes <- substr(data.1$date,1,2)
mes <- str_remove_all(mes, "/")
dia <- substr(data.1$date,nchar(data.1$date)-4,nchar(data.1$date)-3)
dia <- str_remove_all(dia, "/")

h <- substr(data.1$time,1,nchar(data.1$time)-3)
am.pm <- substr(data.1$time,nchar(data.1$time)-1,nchar(data.1$time))
hora <- vector(); minuto <- vector(); segundo <- vector()
for (i in 1:length(h)) {
  if (nchar(h[i]) == 7) {
    if (am.pm[i] == "PM" ) {
      hora[i] <- as.numeric(substr(h[i],1,1)) + 12
    } else {
      hora[i] <- as.numeric(substr(h[i],1,1))
    }
    minuto[i] <- substr(h[i],3,4)
    segundo[i] <- "00"
    #segundo[i] <- substr(h[i],6,7) # Valor exacto
  } else {

    if (am.pm[i] == "PM" ) {
      if (as.numeric(substr(h[i],1,2)) == 12) {
        hora[i] <- as.numeric(substr(h[i],1,2))
      } else {
        hora[i] <- as.numeric(substr(h[i],1,2)) + 12
      }
    } else {
      if (as.numeric(substr(h[i],1,2)) == 12) {
        hora[i] <- as.numeric(substr(h[i],1,2)) - 12
      } else {
        hora[i] <- as.numeric(substr(h[i],1,2))
      }
    }
    minuto[i] <- substr(h[i],4,5)
    segundo[i] <- "00"
    #segundo[i] <- substr(h[i],7,8) # valor exacto
  }
}
data.1$date <- paste(dia,"-",mes,"-",anio,sep = "")
data.1$time <- paste(hora,":",minuto,":",segundo,sep = "")
library(tidyverse)
library(lubridate)
fh <- dmy_hms(paste(data.1$date, data.1$time)) # Transforma data and time
data.1 <- data.frame(ta = as.numeric(data.1$ta),
  h = as.numeric(data.1$h),
  pa = as.numeric(data.1$pa),
  rsd = as.numeric(data.1$rsd),
  vv = as.numeric(data.1$vv))
data.1 <- as.tibble(data.1)
data.1 <- data.1 %>% mutate(fh = fh)
data.1
} # Organiza la informacion para unir las matrices de informacion
organizacion.previa <- function(data){

```

```

#### Carga de informacion ----
u.d <- which(data[2,] == "date"); u.d
u.t <- which(data[2,] == "time"); u.t
u.1 <- which(data[1,] == "Stat_TA_1h" & data[2,] == "Avg"); u.1
u.2 <- which(data[1,] == "Stat_RH_1h" & data[2,] == "Avg"); u.2
u.3 <- which(data[1,] == "Stat_PA_1h" & data[2,] == "Avg"); u.3
u.4 <- which(data[1,] == "Stat_SR_Dif_1h" & data[2,] == "Avg"); u.4
if ( sum(data[1,] == "GenWind_1h_SDI12", na.rm = T) >= 1 ) {
  u.5 <- which(data[1,] == "GenWind_1h_SDI12" & data[2,] == "SpdAvg"); u.5
} else {
  u.5 <- which(data[1,] == "GenWind_1h" & data[2,] == "SpdAvg"); u.5
}
data.1 <- data[-c(1,2),c(u.d,u.t,u.1,u.2,u.3,u.4,u.5)]
row.names(data.1) <- NULL
colnames(data.1) <- c("date","time","ta","h","pa","rsd","vv")
data.1
}
organizacion.fin <- function(data){
  data.1 <- data

#### Modificando el formato de fechas ----
#anio <- paste("20",substr(data.1$date,7,8),sep = "")
anio <- substr(data.1$date,nchar(data.1$date)-1,nchar(data.1$date))
mes <- substr(data.1$date,1,2)
mes <- str_remove_all(mes, "/")
dia <- substr(data.1$date,nchar(data.1$date)-4,nchar(data.1$date)-3)
dia <- str_remove_all(dia, "/")

h <- substr(data.1$time,1,nchar(data.1$time)-3)
am.pm <- substr(data.1$time,nchar(data.1$time)-1,nchar(data.1$time))
hora <- vector(); minuto <- vector(); segundo <- vector()
for (i in 1:length(h)) {
  if (nchar(h[i]) == 7) {
    if (am.pm[i] == "PM" ) {
      hora[i] <- as.numeric(substr(h[i],1,1)) + 12
    } else {
      hora[i] <- as.numeric(substr(h[i],1,1))
    }
    minuto[i] <- substr(h[i],3,4)
    segundo[i] <- "00"
    #segundo[i] <- substr(h[i],6,7) # Valor exacto
  } else {

    if (am.pm[i] == "PM" ) {
      if (as.numeric(substr(h[i],1,2)) == 12) {
        hora[i] <- as.numeric(substr(h[i],1,2))
      } else {
        hora[i] <- as.numeric(substr(h[i],1,2)) + 12
      }
    } else {
      if (as.numeric(substr(h[i],1,2)) == 12) {
        hora[i] <- as.numeric(substr(h[i],1,2)) - 12
      } else {
        hora[i] <- as.numeric(substr(h[i],1,2))
      }
    }
    minuto[i] <- substr(h[i],4,5)
    segundo[i] <- "00"
    #segundo[i] <- substr(h[i],7,8) # valor exacto
  }
}
data.1$date <- paste(dia,"-",mes,"-",anio,sep = "")
data.1$time <- paste(hora,":",minuto,":",segundo,sep = "")
library(tidyverse)
library(lubridate)
fh <- dmy_hms(paste(data.1$date, data.1$time)) # Transforma data and time
data.1 <- data.frame(ta = as.numeric(data.1$ta),
  h = as.numeric(data.1$h),
  pa = as.numeric(data.1$pa),
  rsd = as.numeric(data.1$rsd),
  vv = as.numeric(data.1$vv))
data.1 <- as.tibble(data.1)
data.1 <- data.1 %>% mutate(fh = fh)
data.1
}
data.14 <- organizacion(data.14)

```



```

data.15 <- organizacion(data.15)
data.16 <- organizacion(data.16)
data.17.1 <- organizacion(data.17.1)
data.17.2 <- organizacion(data.17.2)
data.18 <- organizacion(data.18)
data.19.1 <- organizacion(data.19.1)
data.19.2 <- organizacion(data.19.2)
#data.20 <- organizacion.previa(data.20)
#openxlsx::write.xlsx(data.20,'Data/Belen P/MULTITUD/MULTITUD-2020-L2-ANUAL/data.20.xlsx')
data.20 <- openxlsx::read.xlsx('Data/Belen P/MULTITUD/MULTITUD-2020-L2-ANUAL/data.20.xlsx')
data.20 <- organizacion.fin(data.20)
data.21 <- organizacion(data.21)
data <- rbind(data.14, data.15, data.16, data.17.1, data.17.2, data.18, data.19.1, data.19.2, data.20, data.21)
if ( sum(duplicated(data$fh)) != 0 ) {
  data <- data[-which(duplicated(data$fh)),] # Quita los registros duplicados
}
data.esepoch.sin <- data
# rellenando cuando existen NAs
horas_completas <- data.frame(
  fh = seq(
    floor_date(min(data$fh), unit = 'hour'),
    floor_date(max(data$fh), unit = 'hour'),
    by = 'hour'
  ))
dim(horas_completas)
# left join
data <- left_join(x = horas_completas, y = data, by = "fh", copy = TRUE)
dim(data)
dim(data)[1] == dim(horas_completas)[1] # Prueba de dimensiones
data.multitud.sin <- data
# Imputacion
summary(data) # Cantidad de NAs
dim(data)[1]; sum(is.na(data$ta)); round(sum(is.na(data$ta))/dim(data)[1] * 100, 2)
imputacion.NA <- function(y){
  if(length(na.omit(y)) >= 2 & length(na.omit(y)) <= (length(y) - 1) ) {
    data <- data.frame(x=1:length(y), y=y)
    lr <- lm(y ~ x , data = data)
    data$y[is.na(data$y)] <- lr$coefficients[1] + data$x[is.na(data$y)] * lr$coefficients[2]
    y <- data$y
  } else {
    return(y)
  }
  return(y)
} # Imputacion mediante regresion lineal simple para NAs
data[,-1] <- apply(data[,-1], 2, imputacion.NA)
summary(data)
data.multitud <- data
objects()
rm(data, data.14, data.15, data.16, data.17.1, data.17.2, data.18, data.19.1, data.19.2, data.20, data.21, horas_completas,
  imputacion.NA, organizacion, organizacion.fin, organizacion.previa)
objects()
#### FIN Preparacion de los datos ----

#####
##### 5. Tunshi #####
#####

#### Preparacion de los datos ----
data.13 <- read.csv('Data/Belen P/TUNSHI/TUNSHI-2013-L2-ANUAL/L2131218 - L2131220.csv',sep="," ,header=F)
data.14 <- read.csv('Data/Belen P/TUNSHI/TUNSHI-2014-L2-ANUAL/L2140123 - L2141231.csv',sep="," ,header=F)
data.15 <- read.csv('Data/Belen P/TUNSHI/TUNSHI-2015-L2-ANUAL/L2150101 - L2151231.csv',sep="," ,header=F)
data.16 <- read.csv('Data/Belen P/TUNSHI/TUNSHI-2016-L2-ANUAL/L2160101 - L2161205.csv',sep="," ,header=F)
data.17 <- read.csv('Data/Belen P/TUNSHI/TUNSHI-2017-L2-ANUAL/L2170314 - L2171231.csv',sep="," ,header=F)
data.18 <- read.csv('Data/Belen P/TUNSHI/TUNSHI-2018-L2-ANUAL/L2180101 - L2180624.csv',sep="," ,header=F)
data.19 <- read.csv('Data/Belen P/TUNSHI/TUNSHI-2019-L2-ANUAL/L2190416 - L2191231.csv',sep="," ,header=F)
data.20 <- read.csv('Data/Belen P/TUNSHI/TUNSHI-2020-L2-ANUAL/L2200101 - L2201231.csv',sep="," ,header=F)
data.21 <- read.csv('Data/Belen P/TUNSHI/TUNSHI-2021-L2-ANUAL/L2210101 - L2211111.csv',sep="," ,header=F)
organizacion <- function(data){
  #### Carga de informacion ----
  u.d <- which(data[2,] == "date"); u.d
  u.t <- which(data[2,] == "time"); u.t
  u.1 <- which(data[1,] == "Stat_TA_1h" & data[2,] == "Avg"); u.1
  u.2 <- which(data[1,] == "Stat_RH_1h" & data[2,] == "Avg"); u.2
  u.3 <- which(data[1,] == "Stat_PA_1h" & data[2,] == "Avg"); u.3
  u.4 <- which(data[1,] == "Stat_SR_Dif_1h" & data[2,] == "Avg"); u.4
}

```

```

if ( sum(data[1,] == "GenWind_1h_SDI12", na.rm = T) >= 1 ) {
  u.5 <- which(data[1,] == "GenWind_1h_SDI12" & data[2,] == "SpdAvg"); u.5
} else {
  u.5 <- which(data[1,] == "GenWind_1h" & data[2,] == "SpdAvg"); u.5
}
data.1 <- data[-c(1,2),c(u.d,u.t,u.1,u.2,u.3,u.4,u.5)]
row.names(data.1) <- NULL
colnames(data.1) <- c("date","time","ta","h","pa","rsd","vv")
#View(data.1)
str(data.1)
#### FIN Carga de informacion ----

#### Modificando el formato de fechas ----
#anio <- paste("20",substr(data.1$date,7,8),sep = "")
anio <- substr(data.1$date,nchar(data.1$date)-1,nchar(data.1$date))
mes <- substr(data.1$date,1,2)
mes <- str_remove_all(mes, "/")
dia <- substr(data.1$date,nchar(data.1$date)-4,nchar(data.1$date)-3)
dia <- str_remove_all(dia, "/")

h <- substr(data.1$time,1,nchar(data.1$time)-3)
am.pm <- substr(data.1$time,nchar(data.1$time)-1,nchar(data.1$time))
hora <- vector(); minuto <- vector(); segundo <- vector()
for (i in 1:length(h)) {
  if (nchar(h[i]) == 7) {
    if (am.pm[i] == "PM" ) {
      hora[i] <- as.numeric(substr(h[i],1,1)) + 12
    } else {
      hora[i] <- as.numeric(substr(h[i],1,1))
    }
    minuto[i] <- substr(h[i],3,4)
    segundo[i] <- "00"
    #segundo[i] <- substr(h[i],6,7) # Valor exacto
  } else {

    if (am.pm[i] == "PM" ) {
      if (as.numeric(substr(h[i],1,2)) == 12) {
        hora[i] <- as.numeric(substr(h[i],1,2))
      } else {
        hora[i] <- as.numeric(substr(h[i],1,2)) + 12
      }
    } else {
      if (as.numeric(substr(h[i],1,2)) == 12) {
        hora[i] <- as.numeric(substr(h[i],1,2)) - 12
      } else {
        hora[i] <- as.numeric(substr(h[i],1,2))
      }
    }
    minuto[i] <- substr(h[i],4,5)
    segundo[i] <- "00"
    #segundo[i] <- substr(h[i],7,8) # valor exacto
  }
}
data.1$date <- paste(dia,"-",mes,"-",anio,sep = "")
data.1$time <- paste(hora,":",minuto,":",segundo,sep = "")
library(tidyverse)
library(lubridate)
fh <- dmy_hms(paste(data.1$date, data.1$time)) # Transforma data and time
data.1 <- data.frame(ta = as.numeric(data.1$ta),
  h = as.numeric(data.1$h),
  pa = as.numeric(data.1$pa),
  rsd = as.numeric(data.1$rsd),
  vv = as.numeric(data.1$vv))
data.1 <- as.tibble(data.1)
data.1 <- data.1 %>% mutate(fh = fh)
data.1
} # Organiza la informacion para unir las matrices de informacion
organizacion.previa <- function(data){
#### Carga de informacion ----
u.d <- which(data[2,] == "date"); u.d
u.t <- which(data[2,] == "time"); u.t
u.1 <- which(data[1,] == "Stat_TA_1h" & data[2,] == "Avg"); u.1
u.2 <- which(data[1,] == "Stat_RH_1h" & data[2,] == "Avg"); u.2
u.3 <- which(data[1,] == "Stat_PA_1h" & data[2,] == "Avg"); u.3
u.4 <- which(data[1,] == "Stat_SR_Dif_1h" & data[2,] == "Avg"); u.4
if ( sum(data[1,] == "GenWind_1h_SDI12", na.rm = T) >= 1 ) {

```

```

    u.5 <- which(data[1,] == "GenWind_1h_SDI12" & data[2,] == "SpdAvg"); u.5
  } else {
    u.5 <- which(data[1,] == "GenWind_1h" & data[2,] == "SpdAvg"); u.5
  }
data.1 <- data[-c(1,2),c(u.d,u.t,u.1,u.2,u.3,u.4,u.5)]
row.names(data.1) <- NULL
colnames(data.1) <- c("date","time","ta","h","pa","rsd","vv")
data.1
}
organizacion.fin <- function(data){
  data.1 <- data

  ##### Modificando el formato de fechas ----
  #anio <- paste("20",substr(data.1$date,7,8),sep = "")
  anio <- substr(data.1$date,nchar(data.1$date)-1,nchar(data.1$date))
  mes <- substr(data.1$date,1,2)
  mes <- str_remove_all(mes, "/")
  dia <- substr(data.1$date,nchar(data.1$date)-4,nchar(data.1$date)-3)
  dia <- str_remove_all(dia, "/")

  h <- substr(data.1$time,1,nchar(data.1$time)-3)
  am.pm <- substr(data.1$time,nchar(data.1$time)-1,nchar(data.1$time))
  hora <- vector(); minuto <- vector(); segundo <- vector()
  for (i in 1:length(h)) {
    if (nchar(h[i]) == 7) {
      if (am.pm[i] == "PM" ) {
        hora[i] <- as.numeric(substr(h[i],1,1)) + 12
      } else {
        hora[i] <- as.numeric(substr(h[i],1,1))
      }
      minuto[i] <- substr(h[i],3,4)
      segundo[i] <- "00"
      #segundo[i] <- substr(h[i],6,7) # Valor exacto
    } else {

      if (am.pm[i] == "PM" ) {
        if (as.numeric(substr(h[i],1,2)) == 12) {
          hora[i] <- as.numeric(substr(h[i],1,2))
        } else {
          hora[i] <- as.numeric(substr(h[i],1,2)) + 12
        }
      } else {
        if (as.numeric(substr(h[i],1,2)) == 12) {
          hora[i] <- as.numeric(substr(h[i],1,2)) - 12
        } else {
          hora[i] <- as.numeric(substr(h[i],1,2))
        }
      }
      minuto[i] <- substr(h[i],4,5)
      segundo[i] <- "00"
      #segundo[i] <- substr(h[i],7,8) # valor exacto
    }
  }
  data.1$date <- paste(dia,"-",mes,"-",anio,sep = "")
  data.1$time <- paste(hora,":",minuto,":",segundo,sep = "")
  library(tidyverse)
  library(lubridate)
  fh <- dmy_hms(paste(data.1$date, data.1$time)) # Transforma data and time
  data.1 <- data.frame(ta = as.numeric(data.1$ta),
                      h = as.numeric(data.1$h),
                      pa = as.numeric(data.1$pa),
                      rsd = as.numeric(data.1$rsd),
                      vv = as.numeric(data.1$vv))
  data.1 <- as.tibble(data.1)
  data.1 <- data.1 %>% mutate(fh = fh)
  data.1
}
data.13 <- organizacion(data.13)
data.14 <- organizacion(data.14)
data.15 <- organizacion(data.15)
data.16 <- organizacion(data.16)
data.17 <- organizacion(data.17)
data.18 <- organizacion(data.18)
data.19 <- organizacion(data.19)
data.20 <- organizacion(data.20)
data.21 <- organizacion(data.21)

```

```

data <- rbind(data.13, data.14, data.15, data.16, data.17, data.18, data.19, data.20, data.21)
if ( sum(duplicated(data$fh)) != 0 ) {
  data <- data[-which(duplicated(data$fh)),] # Quita los registros duplicados
}
data.esepoch.sin <- data
# rellenando cuando existen NAs
horas_completas <- data.frame(
  fh = seq(
    floor_date(min(data$fh), unit = 'hour'),
    floor_date(max(data$fh), unit = 'hour'),
    by = 'hour'
  )
)
dim(horas_completas)
# left join
data <- left_join(x = horas_completas, y = data, by = "fh", copy = TRUE)
dim(data)
dim(data)[1] == dim(horas_completas)[1] # Prueba de dimensiones
data.tunshi.sin <- data
# Imputacion
summary(data) # Cantidad de NAs
dim(data)[1]; sum(is.na(data$ta)); round(sum(is.na(data$ta))/dim(data)[1] * 100, 2)
imputacion.NA <- function(y){
  if(length(na.omit(y)) >= 2 & length(na.omit(y)) <= (length(y) -1) ) {
    data <- data.frame(x=1:length(y), y=y)
    lr <- lm(y ~ x , data = data)
    data$y[is.na(data$y)] <- lr$coefficients[1] + data$x[is.na(data$y)] * lr$coefficients[2]
    y <- data$y
  } else {
    return(y)
  }
}
return(y)
} # Imputacion mediante regresion lineal simple para NAs
data[,-1] <- apply(data[,-1], 2, imputacion.NA)
summary(data)
data.tunshi <- data
objects()
rm(data, data.13, data.14, data.15, data.16, data.17, data.18, data.19, data.20, data.21, horas_completas,
  imputacion.NA, organizacion, organizacion.fin, organizacion.previa)
objects()
#### FIN Preparacion de los datos ----

#####
##### Analisis #####
#####

#### Analisis exploratorio de datos ----

summary(data.alao)
summary(data.esepoch)
summary(data.matus)
summary(data.multitud)
summary(data.tunshi)

aed <- function(data){
  resumen <- rbind(summarise(data, n=length(ta), min=min(ta, na.rm = T),
    q1=quantile(ta, 0.25, na.rm = T), q2=quantile(ta, 0.50, na.rm = T),
    media=mean(ta, na.rm = T), q3=quantile(ta, 0.75, na.rm = T),
    max=max(ta, na.rm = T), sd=sd(ta, na.rm = T),
    Asimetria=skewness(ta, na.rm = T), kurtosis=kurtosis(ta, na.rm = T), NAs=sum(is.na(ta)) ),
  summarise(data, n=length(h), min=min(h, na.rm = T),
    q1=quantile(h, 0.25, na.rm = T), q2=quantile(h, 0.50, na.rm = T),
    media=mean(h, na.rm = T), q3=quantile(h, 0.75, na.rm = T),
    max=max(h, na.rm = T), sd=sd(h, na.rm = T),
    Asimetria=skewness(h, na.rm = T), kurtosis=kurtosis(h, na.rm = T), NAs=sum(is.na(h)) ),
  summarise(data, n=length(pa), min=min(pa, na.rm = T),
    q1=quantile(pa, 0.25, na.rm = T), q2=quantile(pa, 0.50, na.rm = T),
    media=mean(pa, na.rm = T), q3=quantile(pa, 0.75, na.rm = T),
    max=max(pa, na.rm = T), sd=sd(pa, na.rm = T),
    Asimetria=skewness(pa, na.rm = T), kurtosis=kurtosis(pa, na.rm = T), NAs=sum(is.na(pa)) ),
  summarise(data, n=length(rsd), min=min(rsd, na.rm = T),
    q1=quantile(rsd, 0.25, na.rm = T), q2=quantile(rsd, 0.50, na.rm = T),
    media=mean(rsd, na.rm = T), q3=quantile(rsd, 0.75, na.rm = T),

```

```

max=max(rsd, na.rm = T), sd=sd(rsd, na.rm = T),
Asimetria=skewness(rsd, na.rm = T), kurtosis=kurtosis(rsd, na.rm = T), NAs=sum(is.na(rsd)) ),
summarise(data, n=length(vv), min=min(vv, na.rm = T),
q1=quantile(vv, 0.25, na.rm = T), q2=quantile(vv, 0.50, na.rm = T),
media=mean(vv, na.rm = T), q3=quantile(vv, 0.75, na.rm = T),
max=max(vv, na.rm = T), sd=sd(vv, na.rm = T),
Asimetria=skewness(vv, na.rm = T), kurtosis=kurtosis(vv, na.rm = T), NAs=sum(is.na(vv)) ) )
resumen <- data.frame(variable=c("tap","hrp","pap","rsdp","vvp"), round(resumen,2))
resumen
} # Analisis Exploratorio de Datos
resumen.alao <- aed(data.alao)
resumen.esepoch <- aed(data.esepoch)
resumen.matus <- aed(data.matus)
resumen.multitud <- aed(data.multitud)
resumen.tunshi <- aed(data.tunshi)
resumen <- list(resumen.alao, resumen.esepoch, resumen.matus, resumen.multitud,resumen.tunshi)
#openxlsx::write.xlsx(resumen,"Exports/Belen P/resumen.aed.xlsx")

# Box Plot
estacion <- c(rep("Alao",dim(data.alao)[1]), rep("ESPOCH",dim(data.esepoch)[1]), rep("Matus",dim(data.matus)[1]),
rep("Multitud",dim(data.multitud)[1]), rep("Tunshi",dim(data.tunshi)[1]))
data <- data.frame(estacion = estacion, rbind(data.alao, data.esepoch, data.matus, data.multitud, data.tunshi))
estacion.sin <- c(rep("Alao",dim(data.alao.sin)[1]), rep("ESPOCH",dim(data.esepoch.sin)[1]), rep("Matus",dim(data.matus.sin)[1]),
rep("Multitud",dim(data.multitud.sin)[1]), rep("Tunshi",dim(data.tunshi.sin)[1]))
data.sin <- data.frame(estacion = estacion.sin, rbind(data.alao.sin, data.esepoch.sin, data.matus.sin, data.tunshi.sin))

# Temperatuta Ambiente Promedio
ggplot(data, aes(x=factor(estacion), y = ta, fill=factor(estacion))) +
geom_boxplot() +
labs(title="Temperatura Ambiente Promedio según estaciones meteorológicas, periodo 2014-2021",
subtitle="Con información imputada",
y="Temperatura Ambiente Promedio", x="Estaciones Meteorológicas GEA", color=NULL)
#
ggplot(data.sin, aes(x=factor(estacion), y = ta, fill=factor(estacion))) +
geom_boxplot() +
labs(title="Temperatura Ambiente Promedio según estaciones meteorológicas, periodo 2014-2021",
subtitle="Sin información imputada",
y="Temperatura Ambiente Promedio", x="Estaciones Meteorológicas GEA", color=NULL)

# Humedad Relativa Promedio
ggplot(data, aes(x=factor(estacion), y = h, fill=factor(estacion))) +
geom_boxplot() +
labs(title="Humedad Relativa Promedio según estaciones meteorológicas, periodo 2014-2021",
subtitle="Con información imputada",
y="Humedad Relativa Promedio", x="Estaciones Meteorológicas GEA", color=NULL)
#
ggplot(data.sin, aes(x=factor(estacion), y = h, fill=factor(estacion))) +
geom_boxplot() +
labs(title="Humedad Relativa Promedio según estaciones meteorológicas, periodo 2014-2021",
subtitle="Sin información imputada",
y="Humedad Relativa Promedio", x="Estaciones Meteorológicas GEA", color=NULL)

# Presión Atmosférica Promedio
ggplot(data, aes(x=factor(estacion), y = pa, fill=factor(estacion))) +
geom_boxplot() +
labs(title="Presión Atmosférica Promedio según estaciones meteorológicas, periodo 2014-2021",
subtitle="Con información imputada",
y="Presión Atmosférica Promedio", x="Estaciones Meteorológicas GEA", color=NULL)
#
ggplot(data.sin, aes(x=factor(estacion), y = pa, fill=factor(estacion))) +
geom_boxplot() +
labs(title="Presión Atmosférica Promedio según estaciones meteorológicas, periodo 2014-2021",
subtitle="Sin información imputada",
y="Presión Atmosférica Promedio", x="Estaciones Meteorológicas GEA", color=NULL)

# Radiación Solar Difusa Promedio
ggplot(data, aes(x=factor(estacion), y = rsd, fill=factor(estacion))) +
geom_boxplot() +
labs(title="Radiación Solar Difusa Promedio según estaciones meteorológicas, periodo 2014-2021",
subtitle="Con información imputada",
y="Radiación Solar Difusa Promedio", x="Estaciones Meteorológicas GEA", color=NULL)
#
ggplot(data.sin, aes(x=factor(estacion), y = rsd, fill=factor(estacion))) +
geom_boxplot() +
labs(title="Radiación Solar Difusa Promedio según estaciones meteorológicas, periodo 2014-2021",
subtitle="Sin información imputada",

```

```

y="Radiación Solar Difusa Promedio", x="Estaciones Meteorológicas GEA", color=NULL)

# Velocidad de Viento Promedio
ggplot(data, aes(x=factor(estacion), y = vv, fill=factor(estacion))) +
  geom_boxplot() +
  labs(title="Velocidad de Viento Promedio según estaciones meteorológicas, periodo 2014-2021",
        subtitle="Con información imputada",
        y="Velocidad de Viento Promedio", x="Estaciones Meteorológicas GEA", color=NULL)
#
ggplot(data.sin, aes(x=factor(estacion), y = vv, fill=factor(estacion))) +
  geom_boxplot() +
  labs(title="Velocidad de Viento Promedio según estaciones meteorológicas, periodo 2014-2021",
        subtitle="Sin información imputada",
        y="Velocidad de Viento Promedio", x="Estaciones Meteorológicas GEA", color=NULL)
objects()
rm(aed, data.alao.sin, data.esPOCH.sin, data.matus.sin, data.multitud.sin, data.sin,
    data.tunshi.sin, estacion, estacion.sin, resumen, resumen.alao, resumen.esPOCH,
    resumen.matus, resumen.multitud, resumen.tunshi)
#### FIN Analisis exploratorio de datos ----

#### Modelo ARIMA ----
library("openxlsx")
library("forecast")
library("ggplot2")
library("fpp2")
library("urca")
library("cowplot")
# Incluyendo la variable tiempo en la matriz de datos
horas_completas <- data.frame(
  fh = seq(
    floor_date(min(data.tunshi$fh), unit = 'hour'),
    floor_date(max(data.multitud$fh), unit = 'hour'),
    by = 'hour'
  )
)
horas_completas <- data.frame(t = 1:dim(horas_completas)[1], horas_completas)
t <- c(seq(which(as.character(horas_completas$fh) == as.character(min(data.alao$fh))), which(as.character(horas_completas$fh) ==
as.character(max(data.alao$fh))) ),
  seq(which(as.character(horas_completas$fh) == "2015-01-01 00:00:00"), which(as.character(horas_completas$fh) == "2021-12-
14 23:00:00") ),
  seq(which(as.character(horas_completas$fh) == as.character(min(data.matus$fh))), which(as.character(horas_completas$fh) ==
as.character(max(data.matus$fh))) ),
  seq(which(as.character(horas_completas$fh) == as.character(min(data.multitud$fh))), which(as.character(horas_completas$fh)
== as.character(max(data.multitud$fh))) ),
  seq(which(as.character(horas_completas$fh) == as.character(min(data.tunshi$fh))), which(as.character(horas_completas$fh) ==
as.character(max(data.tunshi$fh))) )
)
data <- data.frame(t = t, data)

# Graficos de series de tiempo
ggplot(data, aes(x = t, y = ta)) +
  geom_line(aes(color = estacion), size = 1, alpha = 0.5) +
  labs(title="Gráfico de series de tiempo",
        subtitle="Temperatura Ambiente Promedio según estaciones meteorológicas, periodo 2014-2021",
        y="Temperatura Ambiente Promedio", x="Tiempo")

ggplot(data, aes(x = t, y = h)) +
  geom_line(aes(color = estacion), size = 1, alpha = 0.5) +
  labs(title="Gráfico de series de tiempo",
        subtitle="Humedad Relativa Promedio según estaciones meteorológicas, periodo 2014-2021",
        y="Humedad Relativa Promedio", x="Tiempo")

ggplot(data, aes(x = t, y = pa)) +
  geom_line(aes(color = estacion), size = 1, alpha = 0.5) +
  labs(title="Gráfico de series de tiempo",
        subtitle="Presión Atmosférica Promedio según estaciones meteorológicas, periodo 2014-2021",
        y="Presión Atmosférica Promedio", x="Tiempo")

ggplot(data, aes(x = t, y = rsd)) +
  geom_line(aes(color = estacion), size = 1, alpha = 0.5) +
  labs(title="Gráfico de series de tiempo",
        subtitle="Radiación Solar Difusa Promedio según estaciones meteorológicas, periodo 2014-2021",
        y="Radiación Solar Difusa Promedio", x="Tiempo")

ggplot(data, aes(x = t, y = vv)) +
  geom_line(aes(color = estacion), size = 1, alpha = 0.5) +

```

```
labs(title="Gráfico de series de tiempo",
      subtitle="Velocidad de Viento Promedio según estaciones meteorológicas, periodo 2014-2021",
      y="Velocidad de Viento Promedio", x="Tiempo")
```

```
# Pruebas de hipotesis de estacionariedad
library(tseries)
estacionariedad <- function(y){
  y <- ts(y, start=1, frequency=24)
  p <- adf.test(y)
  p$p.value
} # H1: La serie de tiempo es estacionaria
apply(data.alao[,-1], 2, estacionariedad)
apply(data.esepoch[,-1], 2, estacionariedad)
apply(data.matus[,-1], 2, estacionariedad)
apply(data.multitud[,-1], 2, estacionariedad)
apply(data.tunshi[,-1], 2, estacionariedad)
# Diferenciacion para la pa de tunshi
y <- ts(diff(data.tunshi$pa,1), start = 1, frequency = 24)
adf.test(y)
data.tunshi.est <- data.tunshi
data.tunshi.est$pa <- length(c(diff(data.tunshi$pa,1)[1],diff(data.tunshi$pa,1)))
```

```
# Modelos ARIMA
m.arima <- function(y){
  y <- ts(y, start=1, frequency=24)
  auto.arima(y)
}
arima.alao <- list()
for (i in 2:6) {
  arima.alao[[i]] <- m.arima(data.alao[,i])
}
arima.esepoch <- list()
for (i in 2:6) {
  arima.esepoch[[i]] <- m.arima(data.esepoch[,i])
}
arima.matus <- list()
for (i in 2:6) {
  arima.matus[[i]] <- m.arima(data.matus[,i])
}
arima.multitud <- list()
for (i in 2:6) {
  arima.multitud[[i]] <- m.arima(data.multitud[,i])
}
arima.tunshi <- list()
for (i in 2:6) {
  arima.tunshi[[i]] <- m.arima(data.tunshi[,i])
}
}
```

```
arima.alao[[2]]
arima.alao[[3]]
arima.alao[[4]]
arima.alao[[5]]
arima.alao[[6]]
```

```
arima.esepoch[[2]]
arima.esepoch[[3]]
arima.esepoch[[4]]
arima.esepoch[[5]]
arima.esepoch[[6]]
```

```
arima.matus[[2]]
arima.matus[[3]]
arima.matus[[4]]
arima.matus[[5]]
arima.matus[[6]]
```

```
arima.multitud[[2]]
arima.multitud[[3]]
arima.multitud[[4]]
arima.multitud[[5]]
arima.multitud[[6]]
```

```
arima.tunshi[[2]]
arima.tunshi[[3]]
arima.tunshi[[4]]
arima.tunshi[[5]]
```

```

arima.tunshi[[6]]

# Medidas de adecuacion
m.a <- function(data,fitted){
  #Medidas en escala
  MAD <- mean(abs(data - fitted) )
  MSE <- mean((data - fitted)^2 )
  #Medidas en porcentaje
  MAPE <- mean(abs(data - fitted) / data ) * 100
  MPE <- mean( (data - fitted) / data ) * 100
  result <- data.frame(MAD = round(MAD,2), MSE = round(MSE,2),
    MAPE = round(MAPE,2), MPE = round(MPE,2) )
  return(result)
}
resumen.m.a <- rbind(m.a(data = data.alao$ta, fitted = arima.alao[[2]]$fitted ),
  m.a(data = data.alao$h, fitted = arima.alao[[3]]$fitted ),
  m.a(data = data.alao$pa, fitted = arima.alao[[4]]$fitted ),
  m.a(data = data.alao$rsd, fitted = arima.alao[[5]]$fitted ),
  m.a(data = data.alao$vv, fitted = arima.alao[[6]]$fitted ),

  m.a(data = data.esepoch$ta, fitted = arima.esepoch[[2]]$fitted ),
  m.a(data = data.esepoch$h, fitted = arima.esepoch[[3]]$fitted ),
  m.a(data = data.esepoch$pa, fitted = arima.esepoch[[4]]$fitted ),
  m.a(data = data.esepoch$rsd, fitted = arima.esepoch[[5]]$fitted ),
  m.a(data = data.esepoch$vv, fitted = arima.esepoch[[6]]$fitted ),

  m.a(data = data.matus$ta, fitted = arima.matus[[2]]$fitted ),
  m.a(data = data.matus$h, fitted = arima.matus[[3]]$fitted ),
  m.a(data = data.matus$pa, fitted = arima.matus[[4]]$fitted ),
  m.a(data = data.matus$rsd, fitted = arima.matus[[5]]$fitted ),
  m.a(data = data.matus$vv, fitted = arima.matus[[6]]$fitted ),

  m.a(data = data.multitud$ta, fitted = arima.multitud[[2]]$fitted ),
  m.a(data = data.multitud$h, fitted = arima.multitud[[3]]$fitted ),
  m.a(data = data.multitud$pa, fitted = arima.multitud[[4]]$fitted ),
  m.a(data = data.multitud$rsd, fitted = arima.multitud[[5]]$fitted ),
  m.a(data = data.multitud$vv, fitted = arima.multitud[[6]]$fitted ),

  m.a(data = data.tunshi$ta, fitted = arima.tunshi[[2]]$fitted ),
  m.a(data = data.tunshi$h, fitted = arima.tunshi[[3]]$fitted ),
  m.a(data = data.tunshi$pa, fitted = arima.tunshi[[4]]$fitted ),
  m.a(data = data.tunshi$rsd, fitted = arima.tunshi[[5]]$fitted ),
  m.a(data = data.tunshi$vv, fitted = arima.tunshi[[6]]$fitted )
)
#openxlsx::write.xlsx(resumen.m.a,"Exports/Belen P/resumen.medidas.adecuacion.xlsx")

# Datos test y pronosticos ARIMA
arima.test.alao.ta <- data.alao$ta
arima.pronostico.alao.ta <- arima.alao[[2]]$fitted

arima.test.alao.h <- data.alao$h
arima.pronostico.alao.h <- arima.alao[[3]]$fitted

arima.test.alao.pa <- data.alao$pa
arima.pronostico.alao.pa <- arima.alao[[4]]$fitted

arima.test.alao.rsd <- data.alao$rsd
arima.pronostico.alao.rsd <- arima.alao[[5]]$fitted

arima.test.alao.vv <- data.alao$vv
arima.pronostico.alao.vv <- arima.alao[[6]]$fitted

arima.test.esepoch.ta <- data.esepoch$ta
arima.pronostico.esepoch.ta <- arima.esepoch[[2]]$fitted

arima.test.esepoch.h <- data.esepoch$h
arima.pronostico.esepoch.h <- arima.esepoch[[3]]$fitted

arima.test.esepoch.pa <- data.esepoch$pa
arima.pronostico.esepoch.pa <- arima.esepoch[[4]]$fitted

arima.test.esepoch.rsd <- data.esepoch$rsd
arima.pronostico.esepoch.rsd <- arima.esepoch[[5]]$fitted

```



```
arima.test.esPOCH.vv <- data.esPOCH$vv
arima.pronostico.esPOCH.vv <- arima.esPOCH[[6]]$fitted
```

```
arima.test.matus.ta <- data.matus$ta
arima.pronostico.matus.ta <- arima.matus[[2]]$fitted
```

```
arima.test.matus.h <- data.matus$h
arima.pronostico.matus.h <- arima.matus[[3]]$fitted
```

```
arima.test.matus.pa <- data.matus$pa
arima.pronostico.matus.pa <- arima.matus[[4]]$fitted
```

```
arima.test.matus.rsd <- data.matus$rsd
arima.pronostico.matus.rsd <- arima.matus[[5]]$fitted
```

```
arima.test.matus.vv <- data.matus$vv
arima.pronostico.matus.vv <- arima.matus[[6]]$fitted
```

```
arima.test.multitud.ta <- data.multitud$ta
arima.pronostico.multitud.ta <- arima.multitud[[2]]$fitted
```

```
arima.test.multitud.h <- data.multitud$h
arima.pronostico.multitud.h <- arima.multitud[[3]]$fitted
```

```
arima.test.multitud.pa <- data.multitud$pa
arima.pronostico.multitud.pa <- arima.multitud[[4]]$fitted
```

```
arima.test.multitud.rsd <- data.multitud$rsd
arima.pronostico.multitud.rsd <- arima.multitud[[5]]$fitted
```

```
arima.test.multitud.vv <- data.multitud$vv
arima.pronostico.multitud.vv <- arima.multitud[[6]]$fitted
```

```
arima.test.tunshi.ta <- data.tunshi$ta
arima.pronostico.tunshi.ta <- arima.tunshi[[2]]$fitted
```

```
arima.test.tunshi.h <- data.tunshi$h
arima.pronostico.tunshi.h <- arima.tunshi[[3]]$fitted
```

```
arima.test.tunshi.pa <- data.tunshi$pa
arima.pronostico.tunshi.pa <- arima.tunshi[[4]]$fitted
```

```
arima.test.tunshi.rsd <- data.tunshi$rsd
arima.pronostico.tunshi.rsd <- arima.tunshi[[5]]$fitted
```

```
arima.test.tunshi.vv <- data.tunshi$vv
arima.pronostico.tunshi.vv <- arima.tunshi[[6]]$fitted
```

```
# Comprobacion de supuestos
```

```
# Estacionariedad: H1: La serie es estacionaria
```

```
estacionariedad(arima.alao[[2]]$residuals )
```

```
estacionariedad(arima.alao[[3]]$residuals )
```

```
estacionariedad(arima.alao[[4]]$residuals )
```

```
estacionariedad(arima.alao[[5]]$residuals )
```

```
estacionariedad(arima.alao[[6]]$residuals )
```

```
estacionariedad(arima.esPOCH[[2]]$residuals )
```

```
estacionariedad(arima.esPOCH[[3]]$residuals )
```

```
estacionariedad(arima.esPOCH[[4]]$residuals )
```

```
estacionariedad(arima.esPOCH[[5]]$residuals )
```

```
estacionariedad(arima.esPOCH[[6]]$residuals )
```

```
estacionariedad(arima.matus[[2]]$residuals )
```

```
estacionariedad(arima.matus[[3]]$residuals )
```

```
estacionariedad(arima.matus[[4]]$residuals )
```

```
estacionariedad(arima.matus[[5]]$residuals )
```

```
estacionariedad(arima.matus[[6]]$residuals )
```

```
estacionariedad(arima.multitud[[2]]$residuals )
```

```
estacionariedad(arima.multitud[[3]]$residuals )
```

```
estacionariedad(arima.multitud[[4]]$residuals )
```

```

estacionariedad(arima.multitud[[5]]$residuals )
estacionariedad(arima.multitud[[6]]$residuals )

estacionariedad(arima.tunshi[[2]]$residuals )
estacionariedad(arima.tunshi[[3]]$residuals )
estacionariedad(arima.tunshi[[4]]$residuals )
estacionariedad(arima.tunshi[[5]]$residuals )
estacionariedad(arima.tunshi[[6]]$residuals )

# Independencia
# Prueba de Ljung-Box de Autocorrelación (Tendencia)
Ljung.Box <- function(x){
  a <- Box.test(x, lag=24, type="Ljung-Box") # H1: Existe autocorrelación
  a$p.value
}

Ljung.Box(arima.alao[[2]]$residuals )
Ljung.Box(arima.alao[[3]]$residuals )
Ljung.Box(arima.alao[[4]]$residuals )
Ljung.Box(arima.alao[[5]]$residuals )
Ljung.Box(arima.alao[[6]]$residuals )

Ljung.Box(arima.esepoch[[2]]$residuals )
Ljung.Box(arima.esepoch[[3]]$residuals )
Ljung.Box(arima.esepoch[[4]]$residuals )
Ljung.Box(arima.esepoch[[5]]$residuals )
Ljung.Box(arima.esepoch[[6]]$residuals )

Ljung.Box(arima.matus[[2]]$residuals )
Ljung.Box(arima.matus[[3]]$residuals )
Ljung.Box(arima.matus[[4]]$residuals )
Ljung.Box(arima.matus[[5]]$residuals )
Ljung.Box(arima.matus[[6]]$residuals )

Ljung.Box(arima.multitud[[2]]$residuals )
Ljung.Box(arima.multitud[[3]]$residuals )
Ljung.Box(arima.multitud[[4]]$residuals )
Ljung.Box(arima.multitud[[5]]$residuals )
Ljung.Box(arima.multitud[[6]]$residuals )

Ljung.Box(arima.tunshi[[2]]$residuals )
Ljung.Box(arima.tunshi[[3]]$residuals )
Ljung.Box(arima.tunshi[[4]]$residuals )
Ljung.Box(arima.tunshi[[5]]$residuals )
Ljung.Box(arima.tunshi[[6]]$residuals )

# Normalidad H1: No son normales
jarque.bera.test(arima.alao[[2]]$residuals )
jarque.bera.test(arima.alao[[3]]$residuals )
jarque.bera.test(arima.alao[[4]]$residuals )
jarque.bera.test(arima.alao[[5]]$residuals )
jarque.bera.test(arima.alao[[6]]$residuals )

jarque.bera.test(arima.esepoch[[2]]$residuals )
jarque.bera.test(arima.esepoch[[3]]$residuals )
jarque.bera.test(arima.esepoch[[4]]$residuals )
jarque.bera.test(arima.esepoch[[5]]$residuals )
jarque.bera.test(arima.esepoch[[6]]$residuals )

jarque.bera.test(arima.matus[[2]]$residuals )
jarque.bera.test(arima.matus[[3]]$residuals )
jarque.bera.test(arima.matus[[4]]$residuals )
jarque.bera.test(arima.matus[[5]]$residuals )
jarque.bera.test(arima.matus[[6]]$residuals )

jarque.bera.test(arima.multitud[[2]]$residuals )
jarque.bera.test(arima.multitud[[3]]$residuals )
jarque.bera.test(arima.multitud[[4]]$residuals )
jarque.bera.test(arima.multitud[[5]]$residuals )
jarque.bera.test(arima.multitud[[6]]$residuals )

jarque.bera.test(arima.tunshi[[2]]$residuals )
jarque.bera.test(arima.tunshi[[3]]$residuals )
jarque.bera.test(arima.tunshi[[4]]$residuals )
jarque.bera.test(arima.tunshi[[5]]$residuals )
jarque.bera.test(arima.tunshi[[6]]$residuals )

```

```
#### FIN Modelo ARIMA ----
```

```
#### Redes Neuronales ----
```

```
library(tseries)  
require(RSNNS)  
require(quantmod)
```

```
m.red.neuronal <- function(RawData) {
```

```
  tsData <- ts(RawData, start = 1, frequency = 24)  
  stsData <- as.ts(tsData,F)  
  stsDataN <- (stsData-min(stsData))/(max(stsData)-min(stsData))
```

```
  #definimos una conjunto de datos para train 70% y 30% para test  
  tamaño_total <- length(stsDataN)  
  tamaño_train <- round(tamaño_total * 0.7, digits = 0)  
  train <- 0:(tamaño_train-1)  
  test <- (tamaño_train):tamaño_total
```

```
  y<-as.zoo(stsDataN)  
  x1<-Lag(y,k=1)  
  x2<-Lag(y,k=2)  
  x3<-Lag(y,k=3)  
  x4<-Lag(y,k=4)  
  x5<-Lag(y,k=5)  
  x6<-Lag(y,k=6)  
  x7<-Lag(y,k=7)  
  x8<-Lag(y,k=8)  
  x9<-Lag(y,k=9)  
  x10<-Lag(y,k=10)  
  x11<-Lag(y,k=11)  
  x12<-Lag(y,k=12)  
  x13<-Lag(y,k=13)  
  x14<-Lag(y,k=14)  
  x15<-Lag(y,k=15)  
  x16<-Lag(y,k=16)  
  x17<-Lag(y,k=17)  
  x18<-Lag(y,k=18)  
  x19<-Lag(y,k=19)  
  x20<-Lag(y,k=20)  
  x21<-Lag(y,k=21)  
  x22<-Lag(y,k=22)  
  x23<-Lag(y,k=23)  
  x24<-Lag(y,k=24)  
  slogN<-cbind(y,x1,x2,x3,x4,x5,x6,x7,x8,x9,x10,x11,x12,x13,x14,x15,x16,x17,x18,x19,x20,x21,x22,x23,x24)
```

```
  stsDataN<-slogN[-(1:24),]
```

```
  inputs<-stsDataN[,2:25]  
  outputs<-stsDataN[,1]
```

```
  # Modelo de Elman  
  fit.elma <- elman(inputs[train],outputs[train], size=c(3,2), learnFuncParams=c(0.07), maxit=5000)  
  predictions<-predict(fit.elma,inputs[-train])  
  valuesPred.elma <- predictions * (max(stsData) - min(stsData)) + min(stsData)  
  values <- RawData[(tamaño_train+24):tamaño_total]
```

```
  # Modelo Jordan  
  fit.jordan <- jordan(inputs[train], outputs[train], size=4, learnFuncParams=c(0.07), maxit=5000)  
  predictions<-predict(fit.jordan,inputs[-train])  
  valuesPred.jordan <- predictions * (max(stsData) - min(stsData)) + min(stsData)
```

```
  y <- as.vector(outputs[-test])
```

```
  resultado <- list(fit.elma, valuesPred.elma, fit.jordan, valuesPred.jordan, inputs, y, test, values)  
  return(resultado)  
}
```

```
# Modelos Redes Neuronales
```

```
red.neuronal.alao <- list()  
for (i in 2:6) {  
  red.neuronal.alao[[i]] <- m.red.neuronal(data.alao[,i])  
}  
red.neuronal.epoch <- list()
```

```

for (i in 2:6) {
  red.neuronal.esepoch[[i]] <- m.red.neuronal(data.esepoch[,i])
}
red.neuronal.matus <- list()
for (i in 2:6) {
  red.neuronal.matus[[i]] <- m.red.neuronal(data.matus[,i])
}
red.neuronal.multitud <- list()
for (i in 2:6) {
  red.neuronal.multitud[[i]] <- m.red.neuronal(data.multitud[,i])
}
red.neuronal.tunshi <- list()
for (i in 2:6) {
  red.neuronal.tunshi[[i]] <- m.red.neuronal(data.tunshi[,i])
}

# Medidas de adecuacion
# Elman
i <- 2
m.a.elma <- data.frame()
while (i <= 6) {
  a <- red.neuronal.alao[[i]]
  m.a.elma <- rbind(m.a.elma, m.a(data = a[[8]], fitted = a[[2]]))
  i <- i + 1
}
i <- 2
while (i <= 6) {
  a <- red.neuronal.esepoch[[i]]
  m.a.elma <- rbind(m.a.elma, m.a(data = a[[8]], fitted = a[[2]]))
  i <- i + 1
}
i <- 2
while (i <= 6) {
  a <- red.neuronal.matus[[i]]
  m.a.elma <- rbind(m.a.elma, m.a(data = a[[8]], fitted = a[[2]]))
  i <- i + 1
}
i <- 2
while (i <= 6) {
  a <- red.neuronal.multitud[[i]]
  m.a.elma <- rbind(m.a.elma, m.a(data = a[[8]], fitted = a[[2]]))
  i <- i + 1
}
i <- 2
while (i <= 6) {
  a <- red.neuronal.tunshi[[i]]
  m.a.elma <- rbind(m.a.elma, m.a(data = a[[8]], fitted = a[[2]]))
  i <- i + 1
}
#openxlsx::write.xlsx(m.a.elma,"Exports/Belen P/resumen.m.a.elma.xlsx")

# Jordan
i <- 2
m.a.jordan <- data.frame()
while (i <= 6) {
  a <- red.neuronal.alao[[i]]
  m.a.jordan <- rbind(m.a.jordan, m.a(data = a[[8]], fitted = a[[4]]))
  i <- i + 1
}
i <- 2
while (i <= 6) {
  a <- red.neuronal.esepoch[[i]]
  m.a.jordan <- rbind(m.a.jordan, m.a(data = a[[8]], fitted = a[[4]]))
  i <- i + 1
}
i <- 2
while (i <= 6) {
  a <- red.neuronal.matus[[i]]
  m.a.jordan <- rbind(m.a.jordan, m.a(data = a[[8]], fitted = a[[4]]))
  i <- i + 1
}
i <- 2
while (i <= 6) {
  a <- red.neuronal.multitud[[i]]
  m.a.jordan <- rbind(m.a.jordan, m.a(data = a[[8]], fitted = a[[4]]))
  i <- i + 1
}

```

```

}
i <- 2
while (i <= 6) {
  a <- red.neuronal.tunshi[[i]]
  m.a.jordan <- rbind(m.a.jordan, m.a(data = a[[8]], fitted = a[[4]]) )
  i <- i + 1
}
#openxlsx::write.xlsx(m.a.jordan,"Exports/Belen P/resumen.m.a.jordan.xlsx")

```

```

# Datos test y pronosticos Redes Neuronales Elman

```

```

a <- red.neuronal.alao[[2]]
pred <- predict(a[[1]], a[[5]][-a[[7]])
rn.test.alao.ta <- a[[6]]
rn.pronostico.alao.ta <- pred

```

```

a <- red.neuronal.alao[[3]]
pred <- predict(a[[1]], a[[5]][-a[[7]])
rn.test.alao.h <- a[[6]]
rn.pronostico.alao.h <- pred

```

```

a <- red.neuronal.alao[[4]]
pred <- predict(a[[1]], a[[5]][-a[[7]])
rn.test.alao.pa <- a[[6]]
rn.pronostico.alao.pa <- pred

```

```

a <- red.neuronal.alao[[5]]
pred <- predict(a[[1]], a[[5]][-a[[7]])
rn.test.alao.rsd <- a[[6]]
rn.pronostico.alao.rsd <- pred

```

```

a <- red.neuronal.alao[[6]]
pred <- predict(a[[1]], a[[5]][-a[[7]])
rn.test.alao.vv <- a[[6]]
rn.pronostico.alao.vv <- pred

```

```

a <- red.neuronal.epoch[[2]]
pred <- predict(a[[1]], a[[5]][-a[[7]])
rn.test.epoch.ta <- a[[6]]
rn.pronostico.epoch.ta <- pred

```

```

a <- red.neuronal.epoch[[3]]
pred <- predict(a[[1]], a[[5]][-a[[7]])
rn.test.epoch.h <- a[[6]]
rn.pronostico.epoch.h <- pred

```

```

a <- red.neuronal.epoch[[4]]
pred <- predict(a[[1]], a[[5]][-a[[7]])
rn.test.epoch.pa <- a[[6]]
rn.pronostico.epoch.pa <- pred

```

```

a <- red.neuronal.epoch[[5]]
pred <- predict(a[[1]], a[[5]][-a[[7]])
rn.test.epoch.rsd <- a[[6]]
rn.pronostico.epoch.rsd <- pred

```

```

a <- red.neuronal.epoch[[6]]
pred <- predict(a[[1]], a[[5]][-a[[7]])
rn.test.epoch.vv <- a[[6]]
rn.pronostico.epoch.vv <- pred

```

```

a <- red.neuronal.matus[[2]]
pred <- predict(a[[1]], a[[5]][-a[[7]])
rn.test.matus.ta <- a[[6]]
rn.pronostico.matus.ta <- pred

```

```

a <- red.neuronal.matus[[3]]
pred <- predict(a[[1]], a[[5]][-a[[7]])
rn.test.matus.h <- a[[6]]
rn.pronostico.matus.h <- pred

```

```

a <- red.neuronal.matus[[4]]
pred <- predict(a[[1]], a[[5]][-a[[7]])
rn.test.matus.pa <- a[[6]]

```

```

rn.pronostico.matus.pa <- pred

a <- red.neuronal.matus[[5]]
pred <- predict(a[[1]], a[[5]][-a[[7]])
rn.test.matus.rsd <- a[[6]]
rn.pronostico.matus.rsd <- pred

a <- red.neuronal.matus[[6]]
pred <- predict(a[[1]], a[[5]][-a[[7]])
rn.test.matus.vv <- a[[6]]
rn.pronostico.matus.vv <- pred

a <- red.neuronal.multitud[[2]]
pred <- predict(a[[1]], a[[5]][-a[[7]])
rn.test.multitud.ta <- a[[6]]
rn.pronostico.multitud.ta <- pred

a <- red.neuronal.multitud[[3]]
pred <- predict(a[[1]], a[[5]][-a[[7]])
rn.test.multitud.h <- a[[6]]
rn.pronostico.multitud.h <- pred

a <- red.neuronal.multitud[[4]]
pred <- predict(a[[1]], a[[5]][-a[[7]])
rn.test.multitud.pa <- a[[6]]
rn.pronostico.multitud.pa <- pred

a <- red.neuronal.multitud[[5]]
pred <- predict(a[[1]], a[[5]][-a[[7]])
rn.test.multitud.rsd <- a[[6]]
rn.pronostico.multitud.rsd <- pred

a <- red.neuronal.multitud[[6]]
pred <- predict(a[[1]], a[[5]][-a[[7]])
rn.test.multitud.vv <- a[[6]]
rn.pronostico.multitud.vv <- pred

a <- red.neuronal.tunshi[[2]]
pred <- predict(a[[1]], a[[5]][-a[[7]])
rn.test.tunshi.ta <- a[[6]]
rn.pronostico.tunshi.ta <- pred

a <- red.neuronal.tunshi[[3]]
pred <- predict(a[[1]], a[[5]][-a[[7]])
rn.test.tunshi.h <- a[[6]]
rn.pronostico.tunshi.h <- pred

a <- red.neuronal.tunshi[[4]]
pred <- predict(a[[1]], a[[5]][-a[[7]])
rn.test.tunshi.pa <- a[[6]]
rn.pronostico.tunshi.pa <- pred

a <- red.neuronal.tunshi[[5]]
pred <- predict(a[[1]], a[[5]][-a[[7]])
rn.test.tunshi.rsd <- a[[6]]
rn.pronostico.tunshi.rsd <- pred

a <- red.neuronal.tunshi[[6]]
pred <- predict(a[[1]], a[[5]][-a[[7]])
rn.test.tunshi.vv <- a[[6]]
rn.pronostico.tunshi.vv <- pred

# Graficas de evolucion de error y grafica de valores predichos
# Elman
par(mfrow = c(3, 2))
a <- red.neuronal.multitud[[2]]
plotIterativeError(a[[1]], main = "Modelo de Red Neuronal de Elman: Evolución del error según el número de iteraciones (Multitud, tap)")
a <- red.neuronal.matus[[3]]
plotIterativeError(a[[1]], main = "Modelo de Red Neuronal de Elman: Evolución del error según el número de iteraciones (Matus, hrp)")
a <- red.neuronal.multitud[[4]]

```

```

plotIterativeError(a[[1]], main = "Modelo de Red Neuronal de Elman: Evolución del error según el número de iteraciones (Multitud, pap)")
a <- red.neuronal.matus[[5]]
plotIterativeError(a[[1]], main = "Modelo de Red Neuronal de Elman: Evolución del error según el número de iteraciones (Matus, rsdp)")
a <- red.neuronal.alao[[6]]
plotIterativeError(a[[1]], main = "Modelo de Red Neuronal de Elman: Evolución del error según el número de iteraciones (Alao, vvp)")

par(mfrow = c(3, 2))
a <- red.neuronal.multitud[[2]]
pred <- predict(a[[1]], a[[5]][-a[[7]])]
plot(a[[6]],type="l", main = "Mejores Modelo de Elman: Valores reales de la serie de tiempo y pronosticos vs tiempo (Multitud, tap)")
lines(pred,col="red")
a <- red.neuronal.matus[[3]]
pred <- predict(a[[1]], a[[5]][-a[[7]])]
plot(a[[6]],type="l", main = "Mejores Modelo de Elman: Valores reales de la serie de tiempo y pronosticos vs tiempo (Matus, hrp)")
lines(pred,col="red")
a <- red.neuronal.multitud[[4]]
pred <- predict(a[[1]], a[[5]][-a[[7]])]
plot(a[[6]],type="l", main = "Mejores Modelo de Elman: Valores reales de la serie de tiempo y pronosticos vs tiempo (Multitud, pap)")
lines(pred,col="red")
a <- red.neuronal.matus[[5]]
pred <- predict(a[[1]], a[[5]][-a[[7]])]
plot(a[[6]],type="l", main = "Mejores Modelo de Elman: Valores reales de la serie de tiempo y pronosticos vs tiempo (Matus, rsdp)")
lines(pred,col="red")
a <- red.neuronal.alao[[6]]
pred <- predict(a[[1]], a[[5]][-a[[7]])]
plot(a[[6]],type="l", main = "Mejores Modelo de Elman: Valores reales de la serie de tiempo y pronosticos vs tiempo (Alao, vvp)")
lines(pred,col="red")

# Jordan
par(mfrow = c(3, 2))
a <- red.neuronal.multitud[[2]]
plotIterativeError(a[[3]], main = "Modelo de Red Neuronal de Jordan: Evolución del error según el número de iteraciones (Multitud, tap)")
a <- red.neuronal.tunshi[[3]]
plotIterativeError(a[[3]], main = "Modelo de Red Neuronal de Jordan: Evolución del error según el número de iteraciones (Tunshi, hrp)")
a <- red.neuronal.multitud[[4]]
plotIterativeError(a[[3]], main = "Modelo de Red Neuronal de Jordan: Evolución del error según el número de iteraciones (Multitud, pap)")
a <- red.neuronal.multitud[[5]]
plotIterativeError(a[[3]], main = "Modelo de Red Neuronal de Jordan: Evolución del error según el número de iteraciones (Multitud, rsdp)")
a <- red.neuronal.alao[[6]]
plotIterativeError(a[[3]], main = "Modelo de Red Neuronal de Jordan: Evolución del error según el número de iteraciones (Alao, vvp)")

par(mfrow = c(3, 2))
a <- red.neuronal.multitud[[2]]
pred <- predict(a[[3]], a[[5]][-a[[7]])]
plot(a[[6]],type="l", main = "Mejores Modelo de Jordan: Valores reales de la serie de tiempo y pronosticos vs tiempo (Multitud, tap)")
lines(pred,col="red")
a <- red.neuronal.tunshi[[3]]
pred <- predict(a[[3]], a[[5]][-a[[7]])]
plot(a[[6]],type="l", main = "Mejores Modelo de Jordan: Valores reales de la serie de tiempo y pronosticos vs tiempo (Tunshi, hrp)")
lines(pred,col="red")
a <- red.neuronal.multitud[[4]]
pred <- predict(a[[3]], a[[5]][-a[[7]])]
plot(a[[6]],type="l", main = "Mejores Modelo de Jordan: Valores reales de la serie de tiempo y pronosticos vs tiempo (Multitud, pap)")
lines(pred,col="red")
a <- red.neuronal.multitud[[5]]
pred <- predict(a[[3]], a[[5]][-a[[7]])]
plot(a[[6]],type="l", main = "Mejores Modelo de Jordan: Valores reales de la serie de tiempo y pronosticos vs tiempo (Multitud, rsdp)")
lines(pred,col="red")
a <- red.neuronal.alao[[6]]
pred <- predict(a[[3]], a[[5]][-a[[7]])]
plot(a[[6]],type="l", main = "Mejores Modelo de Jordan: Valores reales de la serie de tiempo y pronosticos vs tiempo (Alao, vvp)")
lines(pred,col="red")
#### FIN Redes Neuronales -----

#### Teoria del Caos ----

```

```

# rm(data.alao.sin, data.esepoch.sin, data.matus.sin, data.multitud.sin, data.tunshi.sin)
library(devtools)
#devtools::install_github("antoniofabio/RTisean", subdir = "RTisean")
library(RTisean)
setTISEANpath(path = "D:/Documents/OneDrive - ESCUELA SUPERIOR POLITECNICA DE CHIMBORAZO/7. R -
copia/CODIGO R/science planners/Data/Belen P/Tisean_3.0.0/bin")
library(modeest)
library(moments)
library(openxlsx)
library(tidyverse)
library(tseriesChaos)
library(Metrics)
library(DescTools)
library(forecast)

t_r <- function(data){
  n <- length(data)
  i <- 1
  min <- data[i]
  while(data[i] <= min) {
    min <- data[i]
    i <- i + 1
    if (i==n) {
      data[i]=10^9
    }
  }
  pos <- i-2
  return(pos)
} # Ubicaciones
# Predicciones con rbj
detach("package:RSNNS", unload = TRUE)
# función para predicción
DP <- function(dat,thp,m,d){
  n1 <- length(dat)-thp
  res.rbf <- rbf(dat, m = m, d = d, p = 9, n=n1, L=thp)
  #res.rbf$error
  pred <- res.rbf$pred
  return(pred)
}

teoria.caos <- function(x.modelo){
  # Tiempo de retardo mutal
  Base.X1_X12 <- x.modelo
  MutualBase.X1_X12 <- sapply(Base.X1_X12, mutual, partitions =20,plot = F,lag.max = 24)

  d <- c()
  for (i in 1:length(MutualBase.X1_X12[1,])) {
    d[i] <- t_r(MutualBase.X1_X12[i])
  }
  d

  Pronosticos <- c()
  for (i in 1:4 ) {
    P <- DP(dat = Base.X1_X12[i], thp = thp, m = 24, d = d[i])
    Pronosticos <- cbind(Pronosticos,P)
  }
  colnames(Pronosticos) <- c("ta", "h", "pa", "rsd")
  pronosticos <- as.data.frame(Pronosticos)
}

# Definimos una conjunto de datos para ingresar al modelos (todos los datos menos las dos ultimas semanas) y para la prueba (ultimas
dos semanas)
thp <- 168
teoria.caos.f <- function(x){
  thp <- 24 * 7 #dos semanas
  modelo <- seq(1, dim(x)[1] - thp)
  test <- seq(dim(x)[1] - thp + 1, dim(x)[1])
  x.modelo <- x[modelo, -1]
  x.test <- x[test, -1]
  x.test[,5]

  Pronosticos <- teoria.caos(x.modelo)

  plot(Base.X1_X12$vy, type = "l")
  a <- which(Base.X1_X12$vv == 0)
  a <- sort(a)

```



```

plot(a, type = "l")
a <- a[a > 50000]
posicion <- a[1]-1
modelo <- seq(1,posicion - thp)
test <- seq(posicion - thp + 1, posicion)

P <- DP(dat = Base.X1_X12$vv[modelo], thp = thp, m = 24, d = d[i])
Pronosticos <- cbind(Pronosticos,P)
colnames(Pronosticos) <- c("ta", "h", "pa", "rsd", "vv")
Pronosticos <- as.data.frame(Pronosticos)
x.test$vv <- Base.X1_X12$vv[test]
resultado <- list(Pronosticos, x.test)
}

# Guardar
y <- teoria.caos.f(data.alao)
pronostico.alao <- y[[1]]
test.alao <- y[[2]]

y <- teoria.caos.f(data.esPOCH)
pronostico.esPOCH <- y[[1]]
test.esPOCH <- y[[2]]

y <- teoria.caos.f(data.matus)
pronostico.matus <- y[[1]]
test.matus <- y[[2]]

y <- teoria.caos.f(data.multitud)
pronostico.multitud <- y[[1]]
test.multitud <- y[[2]]

y <- teoria.caos.f(data.tunshi)
pronostico.tunshi <- y[[1]]
test.tunshi <- y[[2]]

# Medidas de adecuacion
m.a <- function(data,fitted){
  #Medidas en escala
  MAD <- mean(abs(data - fitted) )
  MSE <- mean((data - fitted)^2 )
  #Medidas en porcentaje
  MAPE <- mean(abs(data - fitted) / data ) * 100
  MPE <- mean( (data - fitted) / data ) * 100
  result <- data.frame(MAD = round(MAD,2), MSE = round(MSE,2),
    MAPE = round(MAPE,2), MPE = round(MPE,2) )
  return(result)
}

resumen.m.a <- rbind(m.a(data = test.alao$ta, fitted = pronostico.alao$ta ),
  m.a(data = test.alao$h, fitted = pronostico.alao$h ),
  m.a(data = test.alao$pa, fitted = pronostico.alao$pa ),
  m.a(data = test.alao$rsd, fitted = pronostico.alao$rsd ),
  m.a(data = test.alao$vv, fitted = pronostico.alao$vv ),

  m.a(data = test.esPOCH$ta, fitted = pronostico.esPOCH$ta ),
  m.a(data = test.esPOCH$h, fitted = pronostico.esPOCH$h ),
  m.a(data = test.esPOCH$pa, fitted = pronostico.esPOCH$pa ),
  m.a(data = test.esPOCH$rsd, fitted = pronostico.esPOCH$rsd ),
  m.a(data = test.esPOCH$vv, fitted = pronostico.esPOCH$vv ),

  m.a(data = test.matus$ta, fitted = pronostico.matus$ta ),
  m.a(data = test.matus$h, fitted = pronostico.matus$h ),
  m.a(data = test.matus$pa, fitted = pronostico.matus$pa ),
  m.a(data = test.matus$rsd, fitted = pronostico.matus$rsd ),
  m.a(data = test.matus$vv, fitted = pronostico.matus$vv ),

  m.a(data = test.multitud$ta, fitted = pronostico.multitud$ta ),
  m.a(data = test.multitud$h, fitted = pronostico.multitud$h ),
  m.a(data = test.multitud$pa, fitted = pronostico.multitud$pa ),
  m.a(data = test.multitud$rsd, fitted = pronostico.multitud$rsd ),
  m.a(data = test.multitud$vv, fitted = pronostico.multitud$vv ),

  m.a(data = test.tunshi$ta, fitted = pronostico.tunshi$ta ),
  m.a(data = test.tunshi$h, fitted = pronostico.tunshi$h ),
  m.a(data = test.tunshi$pa, fitted = pronostico.tunshi$pa ),

```

```

        m.a(data = test.tunshi$rsd, fitted = pronostico.tunshi$rsd ),
        m.a(data = test.tunshi$vv, fitted = pronostico.tunshi$vv )
    )
#openxlsx::write.xlsx(resumen.m.a,"Exports/Belen P/resumen.medidas.adecuacion.TC.xlsx")

# Graficas de resultados
par(mfrow = c(3, 2))
plot(test.multitud$ta, type="l", main = "Teoría del Caos: Valores reales de la serie de tiempo y pronosticos vs tiempo (Multitud, tap)")
lines(pronostico.multitud$ta, col="red")

plot(test.matus$h, type="l", main = "Teoría del Caos: Valores reales de la serie de tiempo y pronosticos vs tiempo (Matus, hrp)")
lines(pronostico.matus$h, col="red")

plot(test.multitud$pa, type="l", main = "Teoría del Caos: Valores reales de la serie de tiempo y pronosticos vs tiempo (Multitud, pap)")
lines(pronostico.multitud$pa, col="red")

plot(test.multitud$rsd, type="l", main = "Teoría del Caos: Valores reales de la serie de tiempo y pronosticos vs tiempo (Multitud, rsdp)")
lines(pronostico.multitud$rsd, col="red")

plot(test.esPOCH$vv, type="l", main = "Teoría del Caos: Valores reales de la serie de tiempo y pronosticos vs tiempo (ESPOCH, vvp)")
lines(pronostico.esPOCH$vv, col="red")

#### FIN Teoria del Caos ----

#### Analisis de precision de modelos ----
Utheil <- function(dat,pred){
#%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
  suppressWarnings(suppressMessages({
    library(DescTools)
  }))
#%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
  n1 <- length(dat)-8760
  n2 <- n1+length(pred)
  #Utheil 1 y 2 aunque el 2 es mas preciso
  U1 <- TheilU(dat[(n1+1):n2],pred,type = 1)
  U2 <- TheilU(dat[(n1+1):n2],pred,type = 2)
  result <- data.frame(U1,U2,row.names = variable.names(dat))
  return(result)
}

x <- arima.test.alao.vv
plot(x, type = "l")
a <- which(x == 0)
a <- sort(a)
plot(a, type = "l")
a <- a[a > 40000]
posicion <- a[1]-1
modelo <- seq(1,posicion - thp)
test <- seq(posicion - thp + 1, posicion)
resumen.Utheil.arima <- rbind(Utheil(tail(arima.test.alao.ta,168), tail(arima.pronostico.alao.ta,168) ),
  Utheil(tail(arima.test.alao.h,168), tail(arima.pronostico.alao.h,168) ),
  Utheil(tail(arima.test.alao.pa,168), tail(arima.pronostico.alao.pa,168) ),
  Utheil(tail(arima.test.alao.rsd,168), tail(arima.pronostico.alao.rsd,168) ),
  Utheil(tail(arima.test.alao.vv[test],168), tail(arima.pronostico.alao.vv[test],168) )
)

x <- arima.test.esPOCH.vv
plot(x, type = "l")
a <- which(x == 0)
a <- sort(a)
plot(a, type = "l")
a <- a[a > 50000]
posicion <- a[1]-1
modelo <- seq(1,posicion - thp)
test <- seq(posicion - thp + 1, posicion)
resumen.Utheil.arima <- rbind(resumen.Utheil.arima,
  Utheil(tail(arima.test.esPOCH.ta,168), tail(arima.pronostico.esPOCH.ta,168) ),
  Utheil(tail(arima.test.esPOCH.h,168), tail(arima.pronostico.esPOCH.h,168) ),
  Utheil(tail(arima.test.esPOCH.pa,168), tail(arima.pronostico.esPOCH.pa,168) ),
  Utheil(tail(arima.test.esPOCH.rsd,168), tail(arima.pronostico.esPOCH.rsd,168) ),
  Utheil(tail(arima.test.esPOCH.vv[test],168), tail(arima.pronostico.esPOCH.vv[test],168) )
)

```

```

x <- arima.test.matus.vv
plot(x, type = "l")
a <- which(x == 0)
a <- sort(a)
plot(a, type = "l")
a <- a[a > 30000]
posicion <- a[1]-1
modelo <- seq(1,posicion - thp)
test <- seq(posicion - thp + 1, posicion)
resumen.Utheil.arima <- rbind(resumen.Utheil.arima,
                             Utheil(tail(arima.test.matus.ta,168), tail(arima.pronostico.matus.ta,168) ),
                             Utheil(tail(arima.test.matus.h,168), tail(arima.pronostico.matus.h,168) ),
                             Utheil(tail(arima.test.matus.pa,168), tail(arima.pronostico.matus.pa,168) ),
                             Utheil(tail(arima.test.matus.rsd,168), tail(arima.pronostico.matus.rsd,168) ),
                             Utheil(tail(arima.test.matus.vv[test],168), tail(arima.pronostico.matus.vv[test],168) )
)

x <- arima.test.multitud.h
plot(x, type = "l")
a <- which(x == 0)
a <- sort(a)
plot(a, type = "l")
a <- a[a > 65000]
posicion <- a[1]-1
modelo <- seq(1,posicion - thp)
test.h <- seq(posicion - thp + 1, posicion)
x <- arima.test.multitud.vv
plot(x, type = "l")
a <- which(x == 0)
a <- sort(a)
plot(a, type = "l")
a <- a[a > 40000]
posicion <- a[1]-1
modelo <- seq(1,posicion - thp)
test <- seq(posicion - thp + 1, posicion)
resumen.Utheil.arima <- rbind(resumen.Utheil.arima,
                             Utheil(tail(arima.test.multitud.ta,168), tail(arima.pronostico.multitud.ta,168) ),
                             Utheil(tail(arima.test.multitud.h[test.h],168), tail(arima.pronostico.multitud.h[test.h],168) ),
                             Utheil(tail(arima.test.multitud.pa,168), tail(arima.pronostico.multitud.pa,168) ),
                             Utheil(tail(arima.test.multitud.rsd,168), tail(arima.pronostico.multitud.rsd,168) ),
                             Utheil(tail(arima.test.multitud.vv[test],168), tail(arima.pronostico.multitud.vv[test],168) )
)

resumen.Utheil.arima <- rbind(resumen.Utheil.arima,
                             Utheil(tail(arima.test.tunshi.ta,168), tail(arima.pronostico.tunshi.ta,168) ),
                             Utheil(tail(arima.test.tunshi.h,168), tail(arima.pronostico.tunshi.h,168) ),
                             Utheil(tail(arima.test.tunshi.pa,168), tail(arima.pronostico.tunshi.pa,168) ),
                             Utheil(tail(arima.test.tunshi.rsd,168), tail(arima.pronostico.tunshi.rsd,168) ),
                             Utheil(tail(arima.test.tunshi.vv,168), tail(arima.pronostico.tunshi.vv,168) )
)
#openxlsx::write.xlsx(resumen.Utheil.arima,"Exports/Belen P/resumen.Utheil.arima.xlsx")

x <- rn.test.alao.vv
plot(x, type = "l")
a <- which(x == 0)
a <- sort(a)
plot(a, type = "l")
a <- a[a > 40000]
posicion <- a[1]-1
modelo <- seq(1,posicion - thp)
test <- seq(posicion - thp + 1, posicion)
resumen.Utheil.RN <- rbind(Utheil(tail(rn.test.alao.ta,168), tail(rn.pronostico.alao.ta,168) ),
                          Utheil(tail(rn.test.alao.h,168), tail(rn.pronostico.alao.h,168) ),
                          Utheil(tail(rn.test.alao.pa,168), tail(rn.pronostico.alao.pa,168) ),
                          Utheil(tail(rn.test.alao.rsd,168), tail(rn.pronostico.alao.rsd,168) ),
                          Utheil(tail(rn.test.alao.vv[test],168), tail(rn.pronostico.alao.vv[test],168) ),

                          Utheil(tail(rn.test.epoch.ta,168), tail(rn.pronostico.epoch.ta,168) ),
                          Utheil(tail(rn.test.epoch.h,168), tail(rn.pronostico.epoch.h,168) ),
                          Utheil(tail(rn.test.epoch.pa,168), tail(rn.pronostico.epoch.pa,168) ),
                          Utheil(tail(rn.test.epoch.rsd,168), tail(rn.pronostico.epoch.rsd,168) ),
                          Utheil(tail(rn.test.epoch.vv,168), tail(rn.pronostico.epoch.vv,168) ),

                          Utheil(tail(rn.test.matus.ta,168), tail(rn.pronostico.matus.ta,168) ),

```

```

    Utheil(tail(rn.test.matus.h,168), tail(rn.pronostico.matus.h,168) ),
    Utheil(tail(rn.test.matus.pa,168), tail(rn.pronostico.matus.pa,168) ),
    Utheil(tail(rn.test.matus.rsd,168), tail(rn.pronostico.matus.rsd,168) ),
    Utheil(tail(rn.test.matus.vv,168), tail(rn.pronostico.matus.vv,168) ),

    Utheil(tail(rn.test.multitud.ta,168), tail(rn.pronostico.multitud.ta,168) ),
    Utheil(tail(rn.test.multitud.h,168), tail(rn.pronostico.multitud.h,168) ),
    Utheil(tail(rn.test.multitud.pa,168), tail(rn.pronostico.multitud.pa,168) ),
    Utheil(tail(rn.test.multitud.rsd,168), tail(rn.pronostico.multitud.rsd,168) ),
    Utheil(tail(rn.test.multitud.vv,168), tail(rn.pronostico.multitud.vv,168) ),

    Utheil(tail(rn.test.tunshi.ta,168), tail(rn.pronostico.tunshi.ta,168) ),
    Utheil(tail(rn.test.tunshi.h,168), tail(rn.pronostico.tunshi.h,168) ),
    Utheil(tail(rn.test.tunshi.pa,168), tail(rn.pronostico.tunshi.pa,168) ),
    Utheil(tail(rn.test.tunshi.rsd,168), tail(rn.pronostico.tunshi.rsd,168) ),
    Utheil(tail(rn.test.tunshi.vv,168), tail(rn.pronostico.tunshi.vv,168) )
)
#openxlsx::write.xlsx(resumen.Utheil.RN,"Exports/Belen P/resumen.Utheil.RN.xlsx")

resumen.Utheil.TC <- rbind(Utheil(test.alao$ta, pronostico.alao$ta ),
    Utheil(test.alao$h, pronostico.alao$h ),
    Utheil(test.alao$pa, pronostico.alao$pa ),
    Utheil(test.alao$rsd, pronostico.alao$rsd ),
    Utheil(test.alao$vv, pronostico.alao$vv ),

    Utheil(test.espoch$ta, pronostico.espoch$ta ),
    Utheil(test.espoch$h, pronostico.espoch$h ),
    Utheil(test.espoch$pa, pronostico.espoch$pa ),
    Utheil(test.espoch$rsd, pronostico.espoch$rsd ),
    Utheil(test.espoch$vv, pronostico.espoch$vv ),

    Utheil(test.matus$ta, pronostico.matus$ta ),
    Utheil(test.matus$h, pronostico.matus$h ),
    Utheil(test.matus$pa, pronostico.matus$pa ),
    Utheil(test.matus$rsd, pronostico.matus$rsd ),
    Utheil(test.matus$vv, pronostico.matus$vv ),

    Utheil(test.multitud$ta, pronostico.multitud$ta ),
    Utheil(test.multitud$h, pronostico.multitud$h ),
    Utheil(test.multitud$pa, pronostico.multitud$pa ),
    Utheil(test.multitud$rsd, pronostico.multitud$rsd ),
    Utheil(test.multitud$vv, pronostico.multitud$vv ),

    Utheil(test.tunshi$ta, pronostico.tunshi$ta ),
    Utheil(test.tunshi$h, pronostico.tunshi$h ),
    Utheil(test.tunshi$pa, pronostico.tunshi$pa ),
    Utheil(test.tunshi$rsd, pronostico.tunshi$rsd ),
    Utheil(test.tunshi$vv, pronostico.tunshi$vv )
)
resumen.Utheil.TC
#openxlsx::write.xlsx(resumen.Utheil.TC,"Exports/Belen P/resumen.Utheil.TC.xlsx")

# Errores ARIMA
x <- arima.test.alao.vv
plot(x, type = "l")
a <- which(x == 0)
a <- sort(a)
plot(a, type = "l")
a <- a[a > 40000]
posicion <- a[1]-1
modelo <- seq(1,posicion - thp)
test <- seq(posicion - thp + 1, posicion)
arima.alao.ta.error <- tail(arima.test.alao.ta,168) - tail(arima.pronostico.alao.ta,168)
arima.alao.h.error <- tail(arima.test.alao.h,168) - tail(arima.pronostico.alao.h,168)
arima.alao.pa.error <- tail(arima.test.alao.pa,168) - tail(arima.pronostico.alao.pa,168)
arima.alao.rsd.error <- tail(arima.test.alao.rsd,168) - tail(arima.pronostico.alao.rsd,168)
arima.alao.vv.error <- tail(arima.test.alao.vv[test],168) - tail(arima.pronostico.alao.vv[test],168)

x <- arima.test.espoch.vv
plot(x, type = "l")
a <- which(x == 0)
a <- sort(a)
plot(a, type = "l")

```

```

a <- a[a > 50000]
posicion <- a[1]-1
modelo <- seq(1,posicion - thp)
test <- seq(posicion - thp + 1, posicion)
arima.epoch.ta.error <- tail(arima.test.epoch.ta,168) - tail(arima.pronostico.epoch.ta,168)
arima.epoch.h.error <- tail(arima.test.epoch.h,168) - tail(arima.pronostico.epoch.h,168)
arima.epoch.pa.error <- tail(arima.test.epoch.pa,168) - tail(arima.pronostico.epoch.pa,168)
arima.epoch.rsd.error <- tail(arima.test.epoch.rsd,168) - tail(arima.pronostico.epoch.rsd,168)
arima.epoch.vv.error <- tail(arima.test.epoch.vv[test],168) - tail(arima.pronostico.epoch.vv[test],168)

x <- arima.test.matus.vv
plot(x, type = "l")
a <- which(x == 0)
a <- sort(a)
plot(a, type = "l")
a <- a[a > 30000]
posicion <- a[1]-1
modelo <- seq(1,posicion - thp)
test <- seq(posicion - thp + 1, posicion)
arima.matus.ta.error <- tail(arima.test.matus.ta,168) - tail(arima.pronostico.matus.ta,168)
arima.matus.h.error <- tail(arima.test.matus.h,168) - tail(arima.pronostico.matus.h,168)
arima.matus.pa.error <- tail(arima.test.matus.pa,168) - tail(arima.pronostico.matus.pa,168)
arima.matus.rsd.error <- tail(arima.test.matus.rsd,168) - tail(arima.pronostico.matus.rsd,168)
arima.matus.vv.error <- tail(arima.test.matus.vv[test],168) - tail(arima.pronostico.matus.vv[test],168)

x <- arima.test.multitud.h
plot(x, type = "l")
a <- which(x == 0)
a <- sort(a)
plot(a, type = "l")
a <- a[a > 65000]
posicion <- a[1]-1
modelo <- seq(1,posicion - thp)
test.h <- seq(posicion - thp + 1, posicion)
x <- arima.test.multitud.vv
plot(x, type = "l")
a <- which(x == 0)
a <- sort(a)
plot(a, type = "l")
a <- a[a > 40000]
posicion <- a[1]-1
modelo <- seq(1,posicion - thp)
test <- seq(posicion - thp + 1, posicion)
arima.multitud.ta.error <- tail(arima.test.multitud.ta,168) - tail(arima.pronostico.multitud.ta,168)
arima.multitud.h.error <- tail(arima.test.multitud.h[test.h],168) - tail(arima.pronostico.multitud.h[test.h],168)
arima.multitud.pa.error <- tail(arima.test.multitud.pa,168) - tail(arima.pronostico.multitud.pa,168)
arima.multitud.rsd.error <- tail(arima.test.multitud.rsd,168) - tail(arima.pronostico.multitud.rsd,168)
arima.multitud.vv.error <- tail(arima.test.multitud.vv[test],168) - tail(arima.pronostico.multitud.vv[test],168)

arima.tunshi.ta.error <- tail(arima.test.tunshi.ta,168) - tail(arima.pronostico.tunshi.ta,168)
arima.tunshi.h.error <- tail(arima.test.tunshi.h,168) - tail(arima.pronostico.tunshi.h,168)
arima.tunshi.pa.error <- tail(arima.test.tunshi.pa,168) - tail(arima.pronostico.tunshi.pa,168)
arima.tunshi.rsd.error <- tail(arima.test.tunshi.rsd,168) - tail(arima.pronostico.tunshi.rsd,168)
arima.tunshi.vv.error <- tail(arima.test.tunshi.vv,168) - tail(arima.pronostico.tunshi.vv,168)

# Errores Red Neuronal Elman
x <- rn.test.alao.vv
plot(x, type = "l")
a <- which(x == 0)
a <- sort(a)
plot(a, type = "l")
a <- a[a > 40000]
posicion <- a[1]-1
modelo <- seq(1,posicion - thp)
test <- seq(posicion - thp + 1, posicion)
rn.alao.ta.error <- tail(rn.test.alao.ta,168) - tail(rn.pronostico.alao.ta,168)
rn.alao.h.error <- tail(rn.test.alao.h,168) - tail(rn.pronostico.alao.h,168)
rn.alao.pa.error <- tail(rn.test.alao.pa,168) - tail(rn.pronostico.alao.pa,168)
rn.alao.rsd.error <- tail(rn.test.alao.rsd,168) - tail(rn.pronostico.alao.rsd,168)
rn.alao.vv.error <- tail(rn.test.alao.vv[test],168) - tail(rn.pronostico.alao.vv[test],168)

rn.epoch.ta.error <- tail(rn.test.epoch.ta,168) - tail(rn.pronostico.epoch.ta,168)
rn.epoch.h.error <- tail(rn.test.epoch.h,168) - tail(rn.pronostico.epoch.h,168)
rn.epoch.pa.error <- tail(rn.test.epoch.pa,168) - tail(rn.pronostico.epoch.pa,168)

```

```

rn.epoch.rsd.error <- tail(rn.test.epoch.rsd,168) - tail(rn.pronostico.epoch.rsd,168)
rn.epoch.vv.error <- tail(rn.test.epoch.vv,168) - tail(rn.pronostico.epoch.vv,168)

rn.matus.ta.error <- tail(rn.test.matus.ta,168) - tail(rn.pronostico.matus.ta,168)
rn.matus.h.error <- tail(rn.test.matus.h,168) - tail(rn.pronostico.matus.h,168)
rn.matus.pa.error <- tail(rn.test.matus.pa,168) - tail(rn.pronostico.matus.pa,168)
rn.matus.rsd.error <- tail(rn.test.matus.rsd,168) - tail(rn.pronostico.matus.rsd,168)
rn.matus.vv.error <- tail(rn.test.matus.vv,168) - tail(rn.pronostico.matus.vv,168)

rn.multitud.ta.error <- tail(rn.test.multitud.ta,168) - tail(rn.pronostico.multitud.ta,168)
rn.multitud.h.error <- tail(rn.test.multitud.h,168) - tail(rn.pronostico.multitud.h,168)
rn.multitud.pa.error <- tail(rn.test.multitud.pa,168) - tail(rn.pronostico.multitud.pa,168)
rn.multitud.rsd.error <- tail(rn.test.multitud.rsd,168) - tail(rn.pronostico.multitud.rsd,168)
rn.multitud.vv.error <- tail(rn.test.multitud.vv,168) - tail(rn.pronostico.multitud.vv,168)

rn.tunshi.ta.error <- tail(rn.test.tunshi.ta,168) - tail(rn.pronostico.tunshi.ta,168)
rn.tunshi.h.error <- tail(rn.test.tunshi.h,168) - tail(rn.pronostico.tunshi.h,168)
rn.tunshi.pa.error <- tail(rn.test.tunshi.pa,168) - tail(rn.pronostico.tunshi.pa,168)
rn.tunshi.rsd.error <- tail(rn.test.tunshi.rsd,168) - tail(rn.pronostico.tunshi.rsd,168)
rn.tunshi.vv.error <- tail(rn.test.tunshi.vv,168) - tail(rn.pronostico.tunshi.vv,168)

# Test de Diabolt Mariano
resumen.diabolt <- rbind(dm.test(arima.alao.ta.error, rn.alao.ta.error, alternative = "greater")$p.value ,
  dm.test(arima.alao.h.error, rn.alao.h.error, alternative = "greater")$p.value ,
  dm.test(arima.alao.pa.error, rn.alao.pa.error, alternative = "greater")$p.value ,
  dm.test(arima.alao.rsd.error, rn.alao.rsd.error, alternative = "greater")$p.value ,
  dm.test(arima.alao.vv.error, rn.alao.vv.error, alternative = "greater")$p.value ,

  dm.test(arima.epoch.ta.error, rn.epoch.ta.error, alternative = "greater")$p.value ,
  dm.test(arima.epoch.h.error, rn.epoch.h.error, alternative = "greater")$p.value ,
  dm.test(arima.epoch.pa.error, rn.epoch.pa.error, alternative = "greater")$p.value ,
  dm.test(arima.epoch.rsd.error, rn.epoch.rsd.error, alternative = "greater")$p.value ,
  dm.test(arima.epoch.vv.error, rn.epoch.vv.error, alternative = "greater")$p.value ,

  dm.test(arima.matus.ta.error, rn.matus.ta.error, alternative = "greater")$p.value ,
  dm.test(arima.matus.h.error, rn.matus.h.error, alternative = "greater")$p.value ,
  dm.test(arima.matus.pa.error, rn.matus.pa.error, alternative = "greater")$p.value ,
  dm.test(arima.matus.rsd.error, rn.matus.rsd.error, alternative = "greater")$p.value ,
  dm.test(arima.matus.vv.error, rn.matus.vv.error, alternative = "greater")$p.value ,

  dm.test(arima.multitud.ta.error, rn.multitud.ta.error, alternative = "greater")$p.value ,
  dm.test(arima.multitud.h.error, rn.multitud.h.error, alternative = "greater")$p.value ,
  dm.test(arima.multitud.pa.error, rn.multitud.pa.error, alternative = "greater")$p.value ,
  dm.test(arima.multitud.rsd.error, rn.multitud.rsd.error, alternative = "greater")$p.value ,
  dm.test(arima.multitud.vv.error, rn.multitud.vv.error, alternative = "greater")$p.value ,

  dm.test(arima.tunshi.ta.error, rn.tunshi.ta.error, alternative = "greater")$p.value ,
  dm.test(arima.tunshi.h.error, rn.tunshi.h.error, alternative = "greater")$p.value ,
  dm.test(arima.tunshi.pa.error, rn.tunshi.pa.error, alternative = "greater")$p.value ,
  dm.test(arima.tunshi.rsd.error, rn.tunshi.rsd.error, alternative = "greater")$p.value ,
  dm.test(arima.tunshi.vv.error, rn.tunshi.vv.error, alternative = "greater")$p.value
)
round(resumen.diabolt,2)

#### FIN Analisis de precision de modelos ----

#### Gráficas aplicativo ----
# Modelos Arima
x <- arima.test.alao.vv
plot(x, type = "l")
a <- which(x == 0)
a <- sort(a)
plot(a, type = "l")
a <- a[a > 40000]
posicion <- a[1]-1
modelo <- seq(1,posicion - thp)
test <- seq(posicion - thp + 1, posicion)
web.arima.test.alao.ta <- tail(arima.test.alao.ta,168)
web.arima.pronostico.alao.ta <- tail(arima.pronostico.alao.ta,168)
web.arima.test.alao.h <- tail(arima.test.alao.h,168)
web.arima.pronostico.alao.h <- tail(arima.pronostico.alao.h,168)
web.arima.test.alao.pa <- tail(arima.test.alao.pa,168)
web.arima.pronostico.alao.pa <- tail(arima.pronostico.alao.pa,168)
web.arima.test.alao.rsd <- tail(arima.test.alao.rsd,168)

```

```
web.arima.pronostico.alao.rsd <- tail(arima.pronostico.alao.rsd,168)
web.arima.test.alao.vv <- tail(arima.test.alao.vv[test],168)
web.arima.pronostico.alao.vv <- tail(arima.pronostico.alao.vv[test],168)
```

```
x <- arima.test.esepoch.vv
plot(x, type = "l")
a <- which(x == 0)
a <- sort(a)
plot(a, type = "l")
a <- a[a > 50000]
posicion <- a[1]-1
modelo <- seq(1,posicion - thp)
test <- seq(posicion - thp + 1, posicion)
web.arima.test.esepoch.ta <- tail(arima.test.esepoch.ta,168)
web.arima.pronostico.esepoch.ta <- tail(arima.pronostico.esepoch.ta,168)
web.arima.test.esepoch.h <- tail(arima.test.esepoch.h,168)
web.arima.pronostico.esepoch.h <- tail(arima.pronostico.esepoch.h,168)
web.arima.test.esepoch.pa <- tail(arima.test.esepoch.pa,168)
web.arima.pronostico.esepoch.pa <- tail(arima.pronostico.esepoch.pa,168)
web.arima.test.esepoch.rsd <- tail(arima.test.esepoch.rsd,168)
web.arima.pronostico.esepoch.rsd <- tail(arima.pronostico.esepoch.rsd,168)
web.arima.test.esepoch.vv <- tail(arima.test.esepoch.vv[test],168)
web.arima.pronostico.esepoch.vv <- tail(arima.pronostico.esepoch.vv[test],168)
```

```
x <- arima.test.matus.vv
plot(x, type = "l")
a <- which(x == 0)
a <- sort(a)
plot(a, type = "l")
a <- a[a > 30000]
posicion <- a[1]-1
modelo <- seq(1,posicion - thp)
test <- seq(posicion - thp + 1, posicion)
web.arima.test.matus.ta <- tail(arima.test.matus.ta,168)
web.arima.pronostico.matus.ta <- tail(arima.pronostico.matus.ta,168)
web.arima.test.matus.h <- tail(arima.test.matus.h,168)
web.arima.pronostico.matus.h <- tail(arima.pronostico.matus.h,168)
web.arima.test.matus.pa <- tail(arima.test.matus.pa,168)
web.arima.pronostico.matus.pa <- tail(arima.pronostico.matus.pa,168)
web.arima.test.matus.rsd <- tail(arima.test.matus.rsd,168)
web.arima.pronostico.matus.rsd <- tail(arima.pronostico.matus.rsd,168)
web.arima.test.matus.vv <- tail(arima.test.matus.vv[test],168)
web.arima.pronostico.matus.vv <- tail(arima.pronostico.matus.vv[test],168)
```

```
x <- arima.test.multitud.h
plot(x, type = "l")
a <- which(x == 0)
a <- sort(a)
plot(a, type = "l")
a <- a[a > 65000]
posicion <- a[1]-1
modelo <- seq(1,posicion - thp)
test.h <- seq(posicion - thp + 1, posicion)
x <- arima.test.multitud.vv
plot(x, type = "l")
a <- which(x == 0)
a <- sort(a)
plot(a, type = "l")
a <- a[a > 40000]
posicion <- a[1]-1
modelo <- seq(1,posicion - thp)
test <- seq(posicion - thp + 1, posicion)
web.arima.test.multitud.ta <- tail(arima.test.multitud.ta,168)
web.arima.pronostico.multitud.ta <- tail(arima.pronostico.multitud.ta,168)
web.arima.test.multitud.h <- tail(arima.test.multitud.h[test.h],168)
web.arima.pronostico.multitud.h <- tail(arima.pronostico.multitud.h[test.h],168)
web.arima.test.multitud.pa <- tail(arima.test.multitud.pa,168)
web.arima.pronostico.multitud.pa <- tail(arima.pronostico.multitud.pa,168)
web.arima.test.multitud.rsd <- tail(arima.test.multitud.rsd,168)
web.arima.pronostico.multitud.rsd <- tail(arima.pronostico.multitud.rsd,168)
web.arima.test.multitud.vv <- tail(arima.test.multitud.vv[test],168)
web.arima.pronostico.multitud.vv <- tail(arima.pronostico.multitud.vv[test],168)
```

```
web.arima.test.tunshi.ta <- tail(arima.test.tunshi.ta,168)
web.arima.pronostico.tunshi.ta <- tail(arima.pronostico.tunshi.ta,168)
web.arima.test.tunshi.h <- tail(arima.test.tunshi.h,168)
```

```
web.arima.pronostico.tunshi.h <- tail(arima.pronostico.tunshi.h,168)
web.arima.test.tunshi.pa <- tail(arima.test.tunshi.pa,168)
web.arima.pronostico.tunshi.pa <- tail(arima.pronostico.tunshi.pa,168)
web.arima.test.tunshi.rsd <- tail(arima.test.tunshi.rsd,168)
web.arima.pronostico.tunshi.rsd <- tail(arima.pronostico.tunshi.rsd,168)
web.arima.test.tunshi.vv <- tail(arima.test.tunshi.vv,168)
web.arima.pronostico.tunshi.vv <- tail(arima.pronostico.tunshi.vv,168)
```

```
# Modelos Redes Neuronales
```

```
x <- rn.test.alao.vv
plot(x, type = "l")
a <- which(x == 0)
a <- sort(a)
plot(a, type = "l")
a <- a[a > 40000]
posicion <- a[1]-1
modelo <- seq(1,posicion - thp)
test <- seq(posicion - thp + 1, posicion)
web.m.test.alao.ta <- tail(rn.test.alao.ta,168)
web.m.pronostico.alao.ta <- tail(rn.pronostico.alao.ta,168)
web.m.test.alao.h <- tail(rn.test.alao.h,168)
web.m.pronostico.alao.h <- tail(rn.pronostico.alao.h,168)
web.m.test.alao.pa <- tail(rn.test.alao.pa,168)
web.m.pronostico.alao.pa <- tail(rn.pronostico.alao.pa,168)
web.m.test.alao.rsd <- tail(rn.test.alao.rsd,168)
web.m.pronostico.alao.rsd <- tail(rn.pronostico.alao.rsd,168)
web.m.test.alao.vv <- tail(rn.test.alao.vv[test],168)
web.m.pronostico.alao.vv <- tail(rn.pronostico.alao.vv[test],168)
```

```
web.m.test.epoch.ta <- tail(rn.test.epoch.ta,168)
web.m.pronostico.epoch.ta <- tail(rn.pronostico.epoch.ta,168)
web.m.test.epoch.h <- tail(rn.test.epoch.h,168)
web.m.pronostico.epoch.h <- tail(rn.pronostico.epoch.h,168)
web.m.test.epoch.pa <- tail(rn.test.epoch.pa,168)
web.m.pronostico.epoch.pa <- tail(rn.pronostico.epoch.pa,168)
web.m.test.epoch.rsd <- tail(rn.test.epoch.rsd,168)
web.m.pronostico.epoch.rsd <- tail(rn.pronostico.epoch.rsd,168)
web.m.test.epoch.vv <- tail(rn.test.epoch.vv,168)
web.m.pronostico.epoch.vv <- tail(rn.pronostico.epoch.vv,168)
```

```
web.m.test.matus.ta <- tail(rn.test.matus.ta,168)
web.m.pronostico.matus.ta <- tail(rn.pronostico.matus.ta,168)
web.m.test.matus.h <- tail(rn.test.matus.h,168)
web.m.pronostico.matus.h <- tail(rn.pronostico.matus.h,168)
web.m.test.matus.pa <- tail(rn.test.matus.pa,168)
web.m.pronostico.matus.pa <- tail(rn.pronostico.matus.pa,168)
web.m.test.matus.rsd <- tail(rn.test.matus.rsd,168)
web.m.pronostico.matus.rsd <- tail(rn.pronostico.matus.rsd,168)
web.m.test.matus.vv <- tail(rn.test.matus.vv,168)
web.m.pronostico.matus.vv <- tail(rn.pronostico.matus.vv,168)
```

```
web.m.test.multitud.ta <- tail(rn.test.multitud.ta,168)
web.m.pronostico.multitud.ta <- tail(rn.pronostico.multitud.ta,168)
web.m.test.multitud.h <- tail(rn.test.multitud.h,168)
web.m.pronostico.multitud.h <- tail(rn.pronostico.multitud.h,168)
web.m.test.multitud.pa <- tail(rn.test.multitud.pa,168)
web.m.pronostico.multitud.pa <- tail(rn.pronostico.multitud.pa,168)
web.m.test.multitud.rsd <- tail(rn.test.multitud.rsd,168)
web.m.pronostico.multitud.rsd <- tail(rn.pronostico.multitud.rsd,168)
web.m.test.multitud.vv <- tail(rn.test.multitud.vv,168)
web.m.pronostico.multitud.vv <- tail(rn.pronostico.multitud.vv,168)
```

```
web.m.test.tunshi.ta <- tail(rn.test.tunshi.ta,168)
web.m.pronostico.tunshi.ta <- tail(rn.pronostico.tunshi.ta,168)
web.m.test.tunshi.h <- tail(rn.test.tunshi.h,168)
web.m.pronostico.tunshi.h <- tail(rn.pronostico.tunshi.h,168)
web.m.test.tunshi.pa <- tail(rn.test.tunshi.pa,168)
web.m.pronostico.tunshi.pa <- tail(rn.pronostico.tunshi.pa,168)
web.m.test.tunshi.rsd <- tail(rn.test.tunshi.rsd,168)
web.m.pronostico.tunshi.rsd <- tail(rn.pronostico.tunshi.rsd,168)
web.m.test.tunshi.vv <- tail(rn.test.tunshi.vv,168)
web.m.pronostico.tunshi.vv <- tail(rn.pronostico.tunshi.vv,168)
```

```
#Teoria del Caos
```



```
web.tc.test.alao.ta <- test.alao$ta
web.tc.pronostico.alao.ta <- pronostico.alao$ta
web.tc.test.alao.h <- test.alao$h
web.tc.pronostico.alao.h <- pronostico.alao$h
web.tc.test.alao.pa <- test.alao$pa
web.tc.pronostico.alao.pa <- pronostico.alao$pa
web.tc.test.alao.rsd <- test.alao$rsd
web.tc.pronostico.alao.rsd <- pronostico.alao$rsd
web.tc.test.alao.vv <- test.alao$vv
web.tc.pronostico.alao.vv <- pronostico.alao$vv
```

```
web.tc.test.esPOCH.ta <- test.esPOCH$ta
web.tc.pronostico.esPOCH.ta <- pronostico.esPOCH$ta
web.tc.test.esPOCH.h <- test.esPOCH$h
web.tc.pronostico.esPOCH.h <- pronostico.esPOCH$h
web.tc.test.esPOCH.pa <- test.esPOCH$pa
web.tc.pronostico.esPOCH.pa <- pronostico.esPOCH$pa
web.tc.test.esPOCH.rsd <- test.esPOCH$rsd
web.tc.pronostico.esPOCH.rsd <- pronostico.esPOCH$rsd
web.tc.test.esPOCH.vv <- test.esPOCH$vv
web.tc.pronostico.esPOCH.vv <- pronostico.esPOCH$vv
```

```
web.tc.test.matus.ta <- test.matus$ta
web.tc.pronostico.matus.ta <- pronostico.matus$ta
web.tc.test.matus.h <- test.matus$h
web.tc.pronostico.matus.h <- pronostico.matus$h
web.tc.test.matus.pa <- test.matus$pa
web.tc.pronostico.matus.pa <- pronostico.matus$pa
web.tc.test.matus.rsd <- test.matus$rsd
web.tc.pronostico.matus.rsd <- pronostico.matus$rsd
web.tc.test.matus.vv <- test.matus$vv
web.tc.pronostico.matus.vv <- pronostico.matus$vv
```

```
web.tc.test.multitud.ta <- test.multitud$ta
web.tc.pronostico.multitud.ta <- pronostico.multitud$ta
web.tc.test.multitud.h <- test.multitud$h
web.tc.pronostico.multitud.h <- pronostico.multitud$h
web.tc.test.multitud.pa <- test.multitud$pa
web.tc.pronostico.multitud.pa <- pronostico.multitud$pa
web.tc.test.multitud.rsd <- test.multitud$rsd
web.tc.pronostico.multitud.rsd <- pronostico.multitud$rsd
web.tc.test.multitud.vv <- test.multitud$vv
web.tc.pronostico.multitud.vv <- pronostico.multitud$vv
```

```
web.tc.test.tunshi.ta <- test.tunshi$ta
web.tc.pronostico.tunshi.ta <- pronostico.tunshi$ta
web.tc.test.tunshi.h <- test.tunshi$h
web.tc.pronostico.tunshi.h <- pronostico.tunshi$h
web.tc.test.tunshi.pa <- test.tunshi$pa
web.tc.pronostico.tunshi.pa <- pronostico.tunshi$pa
web.tc.test.tunshi.rsd <- test.tunshi$rsd
web.tc.pronostico.tunshi.rsd <- pronostico.tunshi$rsd
web.tc.test.tunshi.vv <- test.tunshi$vv
web.tc.pronostico.tunshi.vv <- pronostico.tunshi$vv
```

```
#### FIN Gráficas aplicativo ----
```

ANEXO C: CÓDIGO APLICATIVO WEB EN R

```
---
title: "GEAA: Aplicativo Web de información Meteorológica"
output:
  flexdashboard::flex_dashboard
---

```{r setup, include=FALSE}
library(flexdashboard)
library(highcharter)
library(tidyverse)
library(openxlsx)
library(shiny)
library(shinydashboard)
library(DT)
library("forecast")
library("fpp2")
library("urca")
library("cowplot")
load("D:/Documents/OneDrive - ESCUELA SUPERIOR POLITECNICA DE CHIMBORAZO/7. R - copia/CODIGO R/science
planners/Data/Belen P/data_GEAA.RData")
```
```

Gráficos de Series de Tiempo {.storyboard}

Temperatura Ambiente Promedio

```
```{r}
ggplot(data, aes(x = t, y = ta)) +
 geom_line(aes(color = estacion), size = 1, alpha = 0.5) +
 labs(title="Gráfico de series de tiempo",
 subtitle="Temperatura Ambiente Promedio según estaciones meteorológicas, periodo 2014-2021",
 y="Temperatura Ambiente Promedio", x="Tiempo")
```
```

Humedad Relativa Promedio

```
```{r}
ggplot(data, aes(x = t, y = h)) +
 geom_line(aes(color = estacion), size = 1, alpha = 0.5) +
 labs(title="Gráfico de series de tiempo",
 subtitle="Humedad Relativa Promedio según estaciones meteorológicas, periodo 2014-2021",
 y="Humedad Relativa Promedio", x="Tiempo")
```
```

Presión Atmosférica Promedio

```
```{r}
ggplot(data, aes(x = t, y = pa)) +
 geom_line(aes(color = estacion), size = 1, alpha = 0.5) +
 labs(title="Gráfico de series de tiempo",
 subtitle="Presión Atmosférica Promedio según estaciones meteorológicas, periodo 2014-2021",
 y="Presión Atmosférica Promedio", x="Tiempo")
```
```

Radiación Solar Difusa Promedio

```
```{r}
ggplot(data, aes(x = t, y = rsd)) +
 geom_line(aes(color = estacion), size = 1, alpha = 0.5) +
 labs(title="Gráfico de series de tiempo",
 subtitle="Radiación Solar Difusa Promedio según estaciones meteorológicas, periodo 2014-2021",
 y="Radiación Solar Difusa Promedio", x="Tiempo")
```
```

```
### Velocidad de Viento Promedio
```

```
```{r}
ggplot(data, aes(x = t, y = vv)) +
 geom_line(aes(color = estacion), size = 1, alpha = 0.5) +
 labs(title="Gráfico de series de tiempo",
 subtitle="Velocidad de Viento Promedio según estaciones meteorológicas, periodo 2014-2021",
 y="Velocidad de Viento Promedio", x="Tiempo")
```
```

```
ARIMA: Valores reales (negro) y pronosticados (rojo) {.storyboard}
```

```
### Alao
```

```
```{r}
par(mfrow = c(3, 2))
plot(web.arima.test.esepoch.ta, type="l", main = "Alao, tap", xlab = "Tiempo", ylab = "tap")
lines(web.arima.pronostico.esepoch.ta, col="red")

plot(web.arima.test.esepoch.h, type="l", main = "Alao, hrp", xlab = "Tiempo", ylab = "hrp")
lines(web.arima.pronostico.esepoch.h, col="red")

plot(web.arima.test.esepoch.pa, type="l", main = "Alao, pap", xlab = "Tiempo", ylab = "pap")
lines(web.arima.pronostico.esepoch.pa, col="red")

plot(web.arima.test.esepoch.rsd, type="l", main = "Alao, rsdp", xlab = "Tiempo", ylab = "rsdp")
lines(web.arima.pronostico.esepoch.rsd, col="red")

plot(web.arima.test.esepoch.vv, type="l", main = "Alao, vvp", xlab = "Tiempo", ylab = "vvp")
lines(web.arima.pronostico.esepoch.vv, col="red")
```
```

```
### Espoch
```

```
```{r}
par(mfrow = c(3, 2))
plot(web.arima.test.esepoch.ta, type="l", main = "Espoch, tap", xlab = "Tiempo", ylab = "tap")
lines(web.arima.pronostico.esepoch.ta, col="red")

plot(web.arima.test.esepoch.h, type="l", main = "Espoch, hrp", xlab = "Tiempo", ylab = "hrp")
lines(web.arima.pronostico.esepoch.h, col="red")

plot(web.arima.test.esepoch.pa, type="l", main = "Espoch, pap", xlab = "Tiempo", ylab = "pap")
lines(web.arima.pronostico.esepoch.pa, col="red")

plot(web.arima.test.esepoch.rsd, type="l", main = "Espoch, rsdp", xlab = "Tiempo", ylab = "rsdp")
lines(web.arima.pronostico.esepoch.rsd, col="red")

plot(web.arima.test.esepoch.vv, type="l", main = "Espoch, vvp", xlab = "Tiempo", ylab = "vvp")
lines(web.arima.pronostico.esepoch.vv, col="red")
```
```

```
### Matus
```

```
```{r}
par(mfrow = c(3, 2))
plot(web.arima.test.matus.ta, type="l", main = "Matus, tap", xlab = "Tiempo", ylab = "tap")
lines(web.arima.pronostico.matus.ta, col="red")

plot(web.arima.test.matus.h, type="l", main = "Matus, hrp", xlab = "Tiempo", ylab = "hrp")
lines(web.arima.pronostico.matus.h, col="red")

plot(web.arima.test.matus.pa, type="l", main = "Matus, pap", xlab = "Tiempo", ylab = "pap")
lines(web.arima.pronostico.matus.pa, col="red")

plot(web.arima.test.matus.rsd, type="l", main = "Matus, rsdp", xlab = "Tiempo", ylab = "rsdp")
lines(web.arima.pronostico.matus.rsd, col="red")
```
```

```

plot(web.arima.test.matus.vv, type="l", main = "Matus, vvp", xlab = "Tiempo", ylab = "vvp")
lines(web.arima.pronostico.matus.vv, col="red")

...

### Multitud

```{r}
par(mfrow = c(3, 2))
plot(web.arima.test.multitud.ta, type="l", main = "Multitud, tap", xlab = "Tiempo", ylab = "tap")
lines(web.arima.pronostico.multitud.ta, col="red")

plot(web.arima.test.multitud.h, type="l", main = "Multitud, hrp", xlab = "Tiempo", ylab = "hrp")
lines(web.arima.pronostico.multitud.h, col="red")

plot(web.arima.test.multitud.pa, type="l", main = "Multitud, pap", xlab = "Tiempo", ylab = "pap")
lines(web.arima.pronostico.multitud.pa, col="red")

plot(web.arima.test.multitud.rsd, type="l", main = "Multitud, rsdp", xlab = "Tiempo", ylab = "rsdp")
lines(web.arima.pronostico.multitud.rsd, col="red")

plot(web.arima.test.multitud.vv, type="l", main = "Multitud, vvp", xlab = "Tiempo", ylab = "vvp")
lines(web.arima.pronostico.multitud.vv, col="red")

...

```

```

tunshi

```{r}
par(mfrow = c(3, 2))
plot(web.arima.test.tunshi.ta, type="l", main = "Tunshi, tap", xlab = "Tiempo", ylab = "tap")
lines(web.arima.pronostico.tunshi.ta, col="red")

plot(web.arima.test.tunshi.h, type="l", main = "Tunshi, hrp", xlab = "Tiempo", ylab = "hrp")
lines(web.arima.pronostico.tunshi.h, col="red")

plot(web.arima.test.tunshi.pa, type="l", main = "Tunshi, pap", xlab = "Tiempo", ylab = "pap")
lines(web.arima.pronostico.tunshi.pa, col="red")

plot(web.arima.test.tunshi.rsd, type="l", main = "Tunshi, rsdp", xlab = "Tiempo", ylab = "rsdp")
lines(web.arima.pronostico.tunshi.rsd, col="red")

plot(web.arima.test.tunshi.vv, type="l", main = "Tunshi, vvp", xlab = "Tiempo", ylab = "vvp")
lines(web.arima.pronostico.tunshi.vv, col="red")

...

```

RN: Valores reales (negro) y pronosticados (rojo) {.storyboard}

```

### Alao

```{r}
par(mfrow = c(3, 2))
plot(web.m.test.esPOCH.ta, type="l", main = "Alao, tap", xlab = "Tiempo", ylab = "tap")
lines(web.m.pronostico.esPOCH.ta, col="red")

plot(web.m.test.esPOCH.h, type="l", main = "Alao, hrp", xlab = "Tiempo", ylab = "hrp")
lines(web.m.pronostico.esPOCH.h, col="red")

plot(web.m.test.esPOCH.pa, type="l", main = "Alao, pap", xlab = "Tiempo", ylab = "pap")
lines(web.m.pronostico.esPOCH.pa, col="red")

plot(web.m.test.esPOCH.rsd, type="l", main = "Alao, rsdp", xlab = "Tiempo", ylab = "rsdp")
lines(web.m.pronostico.esPOCH.rsd, col="red")

plot(web.m.test.esPOCH.vv, type="l", main = "Alao, vvp", xlab = "Tiempo", ylab = "vvp")
lines(web.m.pronostico.esPOCH.vv, col="red")

...

```

### ### Eepoch

```
```{r}
par(mfrow = c(3, 2))
plot(web.m.test.esepoch.ta, type="l", main = "Eepoch, tap", xlab = "Tiempo", ylab = "tap")
lines(web.m.pronostico.esepoch.ta, col="red")

plot(web.m.test.esepoch.h, type="l", main = "Eepoch, hrp", xlab = "Tiempo", ylab = "hrp")
lines(web.m.pronostico.esepoch.h, col="red")

plot(web.m.test.esepoch.pa, type="l", main = "Eepoch, pap", xlab = "Tiempo", ylab = "pap")
lines(web.m.pronostico.esepoch.pa, col="red")

plot(web.m.test.esepoch.rsd, type="l", main = "Eepoch, rsdp", xlab = "Tiempo", ylab = "rsdp")
lines(web.m.pronostico.esepoch.rsd, col="red")

plot(web.m.test.esepoch.vv, type="l", main = "Eepoch, vvp", xlab = "Tiempo", ylab = "vvp")
lines(web.m.pronostico.esepoch.vv, col="red")
```
```

### ### Matus

```
```{r}
par(mfrow = c(3, 2))
plot(web.m.test.matus.ta, type="l", main = "Matus, tap", xlab = "Tiempo", ylab = "tap")
lines(web.m.pronostico.matus.ta, col="red")

plot(web.m.test.matus.h, type="l", main = "Matus, hrp", xlab = "Tiempo", ylab = "hrp")
lines(web.m.pronostico.matus.h, col="red")

plot(web.m.test.matus.pa, type="l", main = "Matus, pap", xlab = "Tiempo", ylab = "pap")
lines(web.m.pronostico.matus.pa, col="red")

plot(web.m.test.matus.rsd, type="l", main = "Matus, rsdp", xlab = "Tiempo", ylab = "rsdp")
lines(web.m.pronostico.matus.rsd, col="red")

plot(web.m.test.matus.vv, type="l", main = "Matus, vvp", xlab = "Tiempo", ylab = "vvp")
lines(web.m.pronostico.matus.vv, col="red")
```
```

### ### Multitud

```
```{r}
par(mfrow = c(3, 2))
plot(web.m.test.multitud.ta, type="l", main = "Multitud, tap", xlab = "Tiempo", ylab = "tap")
lines(web.m.pronostico.multitud.ta, col="red")

plot(web.m.test.multitud.h, type="l", main = "Multitud, hrp", xlab = "Tiempo", ylab = "hrp")
lines(web.m.pronostico.multitud.h, col="red")

plot(web.m.test.multitud.pa, type="l", main = "Multitud, pap", xlab = "Tiempo", ylab = "pap")
lines(web.m.pronostico.multitud.pa, col="red")

plot(web.m.test.multitud.rsd, type="l", main = "Multitud, rsdp", xlab = "Tiempo", ylab = "rsdp")
lines(web.m.pronostico.multitud.rsd, col="red")

plot(web.m.test.multitud.vv, type="l", main = "Multitud, vvp", xlab = "Tiempo", ylab = "vvp")
lines(web.m.pronostico.multitud.vv, col="red")
```
```

### ### tunshi

```
```{r}
par(mfrow = c(3, 2))
plot(web.m.test.tunshi.ta, type="l", main = "Tunshi, tap", xlab = "Tiempo", ylab = "tap")
lines(web.m.pronostico.tunshi.ta, col="red")

plot(web.m.test.tunshi.h, type="l", main = "Tunshi, hrp", xlab = "Tiempo", ylab = "hrp")
```
```

```

lines(web.m.pronostico.tunshi.h, col="red")

plot(web.m.test.tunshi.pa, type="l", main = "Tunshi, pap", xlab = "Tiempo", ylab = "pap")
lines(web.m.pronostico.tunshi.pa, col="red")

plot(web.m.test.tunshi.rsd, type="l", main = "Tunshi, rsdp", xlab = "Tiempo", ylab = "rsdp")
lines(web.m.pronostico.tunshi.rsd, col="red")

plot(web.m.test.tunshi.vv, type="l", main = "Tunshi, vvp", xlab = "Tiempo", ylab = "vvp")
lines(web.m.pronostico.tunshi.vv, col="red")

...

```

TC: Valores reales (negro) y pronosticados (rojo) {.storyboard}

### Alao

```

...{r}
par(mfrow = c(3, 2))
plot(web.tc.test.esepoch.ta, type="l", main = "Alao, tap", xlab = "Tiempo", ylab = "tap")
lines(web.tc.pronostico.esepoch.ta, col="red")

plot(web.tc.test.esepoch.h, type="l", main = "Alao, hrp", xlab = "Tiempo", ylab = "hrp")
lines(web.tc.pronostico.esepoch.h, col="red")

plot(web.tc.test.esepoch.pa, type="l", main = "Alao, pap", xlab = "Tiempo", ylab = "pap")
lines(web.tc.pronostico.esepoch.pa, col="red")

plot(web.tc.test.esepoch.rsd, type="l", main = "Alao, rsdp", xlab = "Tiempo", ylab = "rsdp")
lines(web.tc.pronostico.esepoch.rsd, col="red")

plot(web.tc.test.esepoch.vv, type="l", main = "Alao, vvp", xlab = "Tiempo", ylab = "vvp")
lines(web.tc.pronostico.esepoch.vv, col="red")

...

```

### Espoch

```

...{r}
par(mfrow = c(3, 2))
plot(web.tc.test.esepoch.ta, type="l", main = "Espoch, tap", xlab = "Tiempo", ylab = "tap")
lines(web.tc.pronostico.esepoch.ta, col="red")

plot(web.tc.test.esepoch.h, type="l", main = "Espoch, hrp", xlab = "Tiempo", ylab = "hrp")
lines(web.tc.pronostico.esepoch.h, col="red")

plot(web.tc.test.esepoch.pa, type="l", main = "Espoch, pap", xlab = "Tiempo", ylab = "pap")
lines(web.tc.pronostico.esepoch.pa, col="red")

plot(web.tc.test.esepoch.rsd, type="l", main = "Espoch, rsdp", xlab = "Tiempo", ylab = "rsdp")
lines(web.tc.pronostico.esepoch.rsd, col="red")

plot(web.tc.test.esepoch.vv, type="l", main = "Espoch, vvp", xlab = "Tiempo", ylab = "vvp")
lines(web.tc.pronostico.esepoch.vv, col="red")

...

```

### Matus

```

...{r}
par(mfrow = c(3, 2))
plot(web.tc.test.matus.ta, type="l", main = "Matus, tap", xlab = "Tiempo", ylab = "tap")
lines(web.tc.pronostico.matus.ta, col="red")

plot(web.tc.test.matus.h, type="l", main = "Matus, hrp", xlab = "Tiempo", ylab = "hrp")
lines(web.tc.pronostico.matus.h, col="red")

plot(web.tc.test.matus.pa, type="l", main = "Matus, pap", xlab = "Tiempo", ylab = "pap")
lines(web.tc.pronostico.matus.pa, col="red")

```

```

plot(web.tc.test.matus.rsd, type="l", main = "Matus, rsdp", xlab = "Tiempo", ylab = "rsdp")
lines(web.tc.pronostico.matus.rsd, col="red")

plot(web.tc.test.matus.vv, type="l", main = "Matus, vvp", xlab = "Tiempo", ylab = "vvp")
lines(web.tc.pronostico.matus.vv, col="red")

...

Multitud

```{r}
par(mfrow = c(3, 2))
plot(web.tc.test.multitud.ta, type="l", main = "Multitud, tap", xlab = "Tiempo", ylab = "tap")
lines(web.tc.pronostico.multitud.ta, col="red")

plot(web.tc.test.multitud.h, type="l", main = "Multitud, hrp", xlab = "Tiempo", ylab = "hrp")
lines(web.tc.pronostico.multitud.h, col="red")

plot(web.tc.test.multitud.pa, type="l", main = "Multitud, pap", xlab = "Tiempo", ylab = "pap")
lines(web.tc.pronostico.multitud.pa, col="red")

plot(web.tc.test.multitud.rsd, type="l", main = "Multitud, rsdp", xlab = "Tiempo", ylab = "rsdp")
lines(web.tc.pronostico.multitud.rsd, col="red")

plot(web.tc.test.multitud.vv, type="l", main = "Multitud, vvp", xlab = "Tiempo", ylab = "vvp")
lines(web.tc.pronostico.multitud.vv, col="red")

...

### tunshi

```{r}
par(mfrow = c(3, 2))
plot(web.tc.test.tunshi.ta, type="l", main = "Tunshi, tap", xlab = "Tiempo", ylab = "tap")
lines(web.tc.pronostico.tunshi.ta, col="red")

plot(web.tc.test.tunshi.h, type="l", main = "Tunshi, hrp", xlab = "Tiempo", ylab = "hrp")
lines(web.tc.pronostico.tunshi.h, col="red")

plot(web.tc.test.tunshi.pa, type="l", main = "Tunshi, pap", xlab = "Tiempo", ylab = "pap")
lines(web.tc.pronostico.tunshi.pa, col="red")

plot(web.tc.test.tunshi.rsd, type="l", main = "Tunshi, rsdp", xlab = "Tiempo", ylab = "rsdp")
lines(web.tc.pronostico.tunshi.rsd, col="red")

plot(web.tc.test.tunshi.vv, type="l", main = "Tunshi, vvp", xlab = "Tiempo", ylab = "vvp")
lines(web.tc.pronostico.tunshi.vv, col="red")

...

```



epoch

Dirección de Bibliotecas y  
Recursos del Aprendizaje

UNIDAD DE PROCESOS TÉCNICOS Y ANÁLISIS BIBLIOGRÁFICO Y  
DOCUMENTAL

REVISIÓN DE NORMAS TÉCNICAS, RESUMEN Y BIBLIOGRAFÍA

Fecha de entrega: 22 / 06/ 2023

<b>INFORMACIÓN DEL AUTOR/A (S)</b>
<b>Nombres – Apellidos:</b> Belén Carolina Paucar Chavarrea
<b>INFORMACIÓN INSTITUCIONAL</b>
<b>Facultad:</b> Ciencias
<b>Carrera:</b> Carrera Estadística
<b>Título a optar:</b> Ingeniera en Estadística Informática
<b>f. Analista de Biblioteca responsable:</b> Ing. Rafael Inty Salto Hidalgo



0898-DBRA-UPT-2023