UNIVERSITAT POLITÈCNICA DE VALÈNCIA

UNIVERSIDAD POLITÉCNICA DE VALENCIA

DEPARTAMENTO DE ESTADÍSTICA E INVESTIGACIÓN OPERATIVA APLICADA Y CALIDAD

TRABAJO DE FIN DE MÁSTER

ELABORACIÓN DE UNA METODOLOGÍA DE TRABAJO PARA EL TRATAMIENTO Y LA PREDICCIÓN DE SERIES TEMPORALES DE CONSUMO DE AGUA POTABLE

DIRECTOR:
JUAN CARLOS GARCÍA-DÍAZ

AUTOR:
FIDAE EL MORER

MÁSTER EN INGENIERÍA DE ANÁLISIS DE DATOS, MEJORA DE PROCESOS Y TOMA DE DECISIONES

A 23 DE NOVIEMBRE DE 2020

RESUMEN

El presente trabajo consiste en la elaboración de una metodología para el análisis de una serie temporal de caudal de agua potable en un sector hidráulico de tipo domiciliario de una ciudad de la provincia de Valencia (España). Esta metodología aborda la reconstrucción de la serie temporal mediante la imputación de valores faltantes, la corrección de valores anómalos y la predicción del consumo a corto plazo mediante el uso de técnicas de *machine learning* y *deep learning*. La investigación llevada a cabo propone una metodología novedosa, puesto que en la literatura científica relacionada con este ámbito no se ha abordado el problema del tratamiento de este tipo de series temporales de manera integral. La metodología desarrollada, por lo tanto, pretende ser la semilla de un sistema de ayuda para la toma de decisiones que permita decidir, para cada tipo de serie temporal de caudal de agua potable o similares, cuál es la estrategia idónea que debe seguir el analista para optimizar la predicción del consumo en un sector hidráulico, y por ende, la operación del propio sistema de distribución asociado.

ABSTRACT

The following research consists in the elaboration of a methodology for the analysis of a time series of drinking water flow rate in a domestic hydraulic sector of a city in the province of Valencia (Spain). This methodology deals with the reconstruction of the time series through the imputation of missing values, the correction of outliers and the forecasting of short-term consumption using machine learning and deep learning techniques. The conducted research proposes a novel methodology since the treatment of this kind of time series has not been addressed in a comprehensive way in the scientific literature related to this field. The developed methodology, aims to be the seed of a decision support system that allows to decide, for each kind of time series of drinking water flow rate or similar, which is the ideal strategy to be followed by the analyst to optimize the forecast of the flow rate in a hydraulic sector, and therefore, the operation of the associated distribution system.

ÍNDICE

ÍNDICE DE ILUSTRACIONES

ÍNDICE DE TABLAS

1 INTRODUCCIÓN Y OBJETIVOS

La operación en el ámbito del suministro en las redes de agua potable supone la gestión de un bien escaso que se distribuye a través de una infraestructura que en numerosas ocasiones presenta deficiencias debidas al mal estado de los activos que la componen o de una gestión no basada en los datos.

Uno de los mayores problemas derivados de estas deficiencias es el bajo rendimiento hidráulico de las infraestructuras. Según las estimaciones de la International Water Association, cada día se pierden aproximadamente 346 millones de metros cúbicos de agua por diversos motivos [1].

Un conocimiento preciso y basado en técnicas estadísticas de los hábitos de consumo de agua potable en un sector hidráulico, en conjunto con la capacidad de predecir la demanda de este bien, puede ayudar a una gestión más racional de los recursos hídricos, y del mismo modo, detectar anomalías que sean imperceptibles a ojos del operador.

Algunos ejemplos de la mejora de la operación de las redes de abastecimiento basadas en datos son la detección de fugas debidas a la rotura de tuberías [2] [3] [4] [5] [6], la optimización del funcionamiento de las bombas para el suministro con el menor coste posible [7] [8] [9] o la optimización de la operación de las plantas de tratamiento de aguas residuales [10] [11]. Todos estos problemas requieren disponer de datos de demanda de agua potable que no contengan datos faltantes ni extremadamente anómalos, así como predicciones precisas para operar correctamente.

Algunos autores como Antunes *et al.* [12], Rodríguez Rangel *et al.* [13], Mouatadid y Adamowski [14] o Duerr *et al.* [15] tratan de atajar el problema mediante el uso de diversas técnicas de predicción de series temporales con el objetivo de que la empresa operadora del servicio pueda contar con elementos suficientes para la toma de decisiones a corto plazo, y determinar la necesidad de actuación en la red partiendo de las diferencias entre las predicciones realizadas y las mediciones tomadas en campo.

Sin embargo, en estudios como los mencionados se obvian las dimensiones de la reconstrucción de los valores faltantes o de la detección de valores anómalos, y, por lo tanto, no atienden al estudio de manera detallada del comportamiento de las series temporales de consumo, sino que tienen como única finalidad la predicción de manera precisa de la demanda en un periodo de tiempo definido.

Por lo tanto, y una vez expuesta la problemática asociada al tratamiento de las series temporales de consumo de agua potable en sectores hidráulicos domiciliarios, se propone como objetivo del presente trabajo establecer una metodología integral que aborde todas las fases del tratamiento de los datos desde su extracción de las bases de datos en las que se encuentran hasta la predicción a corto plazo de la demanda.

El fin último del presente trabajo es la elaboración de un sistema de ayuda a la toma de decisiones (DSS o *Decision Support System*, por sus siglas en inglés) que ayude a los operadores de las redes de abastecimiento a determinar cuál es la estrategia óptima para la predicción de este tipo de series temporales, y facilitar así el establecimiento de flujos de trabajo que permitan la mejora de la operación de los sistemas de distribución de agua potable.

Para este fin, el documento plantea en primer lugar una revisión bibliográfica de las metodologías utilizadas para la reconstrucción de series temporales similares a la estudiada en caso de valores faltantes o anómalos, y se analizará el estado del arte de los métodos de predicción de consumos de agua potable a corto plazo, prestando especial atención a las técnicas de *Machine Learning* y *Deep Learning*.

A continuación, se realizará una descripción somera de la serie objeto de estudio, y se detallará la metodología seguida para llevar a cabo el trabajo. Finalmente, se describe la estructura del repositorio en el cual los lectores pueden acceder al paquete de cálculo que resulta como producto del proyecto elaborado, y se discuten los resultados obtenidos.

.

2 REVISIÓN BIBLIOGRÁFICA

2.1 RECONSTRUCCIÓN DE SERIES TEMPORALES

Los datos obtenidos de los sensores desplegados en las redes de distribución de agua potable cuentan a menudo con la presencia de datos faltantes o de valores anómalos que pueden comprometer la integridad de las series temporales y la calidad de los análisis, así como de las predicciones realizadas a partir de ellas.

Para una mejor comprensión de la problemática presentada, cabe definir en primer lugar qué se entiende por datos faltantes y valores anómalos, términos que servirán de referencia para el desarrollo del presente trabajo.

2.1.1 Datos faltantes

Los datos faltantes pueden estar regidos fundamentalmente por tres mecanismos: faltantes de manera completamente aleatoria (MCAR por sus siglas en inglés), faltantes de manera aleatoria (MAR) y datos faltantes de manera no aleatoria (MNAR) [16].

Se dice que los datos se encuentran en una situación MCAR (*Missing Completely At Random*) cuando la probabilidad de que existan datos faltantes en la variable *X* no está relacionada con otras variables medidas ni con los valores de la propia variable *X* [17]. En el caso de una serie temporal de consumo de agua potable, los datos faltantes de este tipo se pueden deber a errores humanos a la hora de introducir los datos, a errores en la lectura debidos a fallos en el sistema de comunicaciones, errores en la instrumentación, etc. [16].

En cuanto a los datos faltantes de tipo MAR (*Missing At Random*), estos se derivan de una correlación con otras variables medidas, pero no a valores hipotéticos de la variable en caso de que no fueran faltantes [17]. Como ejemplo de este tipo de datos faltantes se puede plantear una situación en la que no se sepa el nivel de ingresos de un cliente, pero sí se pueda inferir mediante otras variables conocidas como su profesión, su experiencia o su nivel académico [16]. Puesto que en el presente trabajo únicamente se cuenta con una variable

observada y no se contará con variables exógenas, este tipo de datos faltantes será desestimado.

Por último, los datos faltantes de tipo MNAR (*Missing Not At Random*), son aquellos en los que la probabilidad de que existan datos faltantes está sistemáticamente relacionada con los valores hipotéticos que tomaría la variable observada en caso de no tener datos faltantes [17].

Dado que la variable observada se recoge directamente de sensores dispuestos en la red de distribución de agua potable y que no se contempla la manipulación de dichos datos, se asumirá que el único mecanismo operante como causa de la falta de datos es el primero que se ha descrito. Por lo tanto, se partirá de la premisa de que los datos faltantes se deberán a las siguientes razones:

- Fallos en la comunicación con los *dataloggers*
- Errores de configuración o de instalación de los caudalímetros
- Fallos en la recepción en las bases de datos.

2.1.2 Valores anómalos

Una vez determinada la naturaleza de los datos faltantes que se deberán tratar en el presente trabajo, se procede a la definición del concepto de dato anómalo en el contexto del tratamiento de series temporales univariantes.

Para definir qué es un dato anómalo, se tomará como referencia el trabajo de Loureiro *et al.* [18], en el que se aborda el problema considerando las regiones de valores anómalos.

$$OR(\mu, \sigma, \varphi_1, \varphi_2) = \{w: dist(w, \mu) > r(\varphi_1, \varphi_2)\sigma\} \quad (1)$$

En dicho trabajo se establece que aquellos valores del caudal que tengan una distancia de un valor de referencia mayor que $r(\varphi_1, \varphi_2)\sigma$, se deben considerar anómalos, siendo $r(\varphi_1, \varphi_2)$ un valor umbral que depende de los parámetros φ_1 y φ_2, que a su vez hacen referencia a los porcentajes de falsos positivos y de falsos negativos.

Los autores proponen la sustitución de los parámetros μ y σ de (1), que se refieren a la media y a la desviación típica de la serie temporal, por estimadores

más robustos. Para ello, se adopta como parámetro de localización la mediana de la muestra, y como parámetro de escala el estimador propuesto por Rousseeuw y Croux [19].

$$Q_n = d\{|x_i - x_j|; i < j\}_{(k)} \quad (2)$$

En la ecuación (2) se muestra la expresión mediante la cual se obtiene el parámetro de escala citado anteriormente. Para calcularlo, se toma un conjunto de valores de tamaño n y se calcula la diferencia entre todos los valores dos a dos. Posteriormente, se ordenan de menor a mayor y se extrae el estadístico de orden k, donde $k = \left(\frac{n}{2}\right)/4$. Según los autores que proponen este estimador, el parámetro d es una constante de valor 2,2219.

$$OR(w_n, Q_n, \varphi_1, \varphi_2) = \{w: |w - w_n| > r(\varphi_1, \varphi_2) Q_n(w)\} \quad (3)$$

El resultado una vez realizadas las modificaciones anteriores es el mostrado en la ecuación (3). Cabe destacar que, según señalan los autores, para una mayor precisión del método, en primer lugar, se debe realizar un análisis *cluster*, de manera que no se tenga como referencia la mediana de toda la serie temporal para cada momento, sino que cada día deberá ser comparado con un conjunto de días similares. En el apartado 4.1.2 se recoge el planteamiento desarrollado en este trabajo para la elaboración de segmentos de consumo.

2.1.3 Técnicas de reconstrucción de series temporales

Una vez analizados los principales problemas que pueden plantear las series temporales de consumo de agua potable, se procede a realizar una revisión de las diferentes técnicas mediante las cuales se ha abordado esta problemática.

En el estudio realizado por Quevedo *et al.* [20], se utilizan modelos ARIMA para el cálculo de caudales agregados de manera diaria en un sector hidráulico. De este modo, los autores plantean dos sistemas de predicción y reconstrucción: el ya mencionado y otro basado en el análisis de correlaciones y en el algoritmo LAMDA para la reconstrucción de series con una granularidad de 10 minutos. Estos últimos modelos explotan la correlación entre días similares para determinar patrones de consumo, de modo que los datos que se imputan son los valores medios asociados a los patrones hallados.

Barrela *et al.* [21] proponen una metodología basada en el uso de un modelo ARIMA con estacionalidad simple y de un modelo TBATS con estacionalidades simple y doble. Plantean 3 posibles modos de imputación: uno mediante la realización de predicciones en el sentido de avance del tiempo (*forecasts*), otro en el sentido contrario (*backcasts*) para ser utilizado cuando la falta de datos no permita usar el primer método, y un tercer modelo basado en la combinación de los dos primeros, de modo que el valor que se imputa en el instante *i* viene dado por la ecuación (4).

$$c_i = \delta_i x forecast_i + (1 - \delta_i) x backcast_i, i = 1, \dots, l \quad (4)$$

Siendo *l* el tamaño de la secuencia de valores faltantes y δ un parámetro que toma los siguientes valores:

$$\delta_i = \begin{cases} \frac{1}{2} l, l = 1 \\ \frac{l - i}{l - 1}, l > 1 \end{cases} \quad (5)$$

Los resultados arrojados por el estudio muestran un mejor rendimiento por parte de los modelos TBATS frente al ARIMA, y que no se aprecian diferencias significativas entre usar una o dos estacionalidades a la hora de realizar las predicciones con el modelo TBATS. Además, los autores concluyen que la eficacia del método combinado (es decir, la ponderación de resultados obtenidos por los enfoques *forecast* y *backcast*), es superior al abordaje del problema usando únicamente uno de los dos sistemas.

Cabe traer a colación otros estudios realizados con el objetivo de reconstruir series temporales pese a no estar referidos específicamente al tipo de series que se pretenden tratar en el presente trabajo.

En el trabajo realizado por Bennis *et al.* [22], los autores plantean el uso de modelos autorregresivos para la imputación de datos faltantes realizando barridos en el sentido de avance del tiempo y en el sentido contrario, de manera similar a la propuesta por Barrela *et al.* [21].

En el trabajo de Golyandina & Osipov [23] los autores presentan la aplicación del análisis de espectro singular o SSA por sus siglas en inglés para imputar datos

faltantes. La metodología planteada consiste en la descomposición de la serie temporal en componentes aditivos, tales como la tendencia, la estacionalidad y el ruido, para luego reconstruirla proyectando las componentes extraídas en un subespacio en el que también se incluirán los datos faltantes.

En la Tabla 1 se recogen las principales características y la metodología empleada de los trabajos revisados en esta sección.

2.2 PREDICCIÓN

La revisión de la bibliografía en relación con la predicción de series temporales mediante el uso de técnicas de *machine learning* y d*eep learning* toma como base el trabajo realizado por Antunes *et al.* [12] en el que se realiza una revisión de la literatura existente en relación con el uso de dichos marcos de trabajo para la predicción de la demanda de agua a corto plazo. En dicho trabajo, además, se evalúa una serie de técnicas de *machine learning* y se comparan tomando como referencia un modelo ARIMA.

La revisión bibliográfica que se ha llevado a cabo en el presente estudio tiene en consideración los trabajos publicados posteriormente al año 2017 (año de redacción del trabajo de Antunes *et al.*), y se encuentra resumida en la Tabla 2.

Rodríguez Rangel *et al.* [13] proponen la predicción de la demanda de agua potable a corto plazo mediante el uso de 24 modelos diferentes, cada uno especializado en la predicción de una hora al día. Estos modelos pueden consistir en redes neuronales artificiales, Holt-Winters, ARIMA, etc. Cada modelo utiliza un número diferente de observaciones previas para realizar la predicción, y la red neuronal se entrena mediante un algoritmo genético. El enfoque propuesto por los autores se evalúa sobre 4 caudalímetros de la red de distribución de Barcelona, realizando una comparación con otros métodos como una red neuronal artificial, un suavizado de Holt-Winters, un modelo de vecinos cercanos (KNN) y un método ingenuo (la predicción en $y_{t+1} = y_{t-2}$). De los resultados obtenidos se observa que el método propuesto devuelve las mejores predicciones en 2 de los 4 caudalímetros, y que los modelos basados en Holt-Winters o en el algoritmo KNN arrojan resultados mejores o similares.

Trabajo	Ámbito	Metodología	Comentarios
Loureiro *et al.* [18]	Valores anómalos	Definición de regiones anómalos	Se utilizan estadísticos robustos para establecer un valor umbral a partir del cual un valor se considera anómalo
Quevedo *et al.* [20]	Reconstrucción de series temporales	ARIMA para datos diarios y LAMDA para datos cada 10 minutos	Con ARIMA se realiza la predicción de valores faltantes para datos diarios, mientras que LAMDA identifica patrones e imputa la mediana de dichos patrones. LAMDA solo se puede utilizar en series con pocos valores faltantes
Barrela *et al.* [21]	Reconstrucción de series temporales	ARIMA para estacionalidad simple y TBATS para estacionalidades simple y doble	Se usa un método híbrido de *forecasts* y *backcasts* ponderado que muestra mejores resultados que el uso de uno u otro por separado, aunque el consumo de recursos es intensivo
Bennis *et al* [22]	Reconstrucción de series temporales	AR	Similar al método anterior
Golyandina & Osipov [23]	Reconstrucción de series temporales	Análisis de espectro singular (SSA)	La serie es descompuesta en componentes aditivos (tendencia, estacionalidad y ruido) y se reconstruye mediante la proyección de las componentes extraídas

Tabla 1. Resumen de características de los trabajos revisados

Mouatadid & Adamowski [14] exploran el uso de las máquinas de aprendizaje extremo (*Extreme Learning Machines*) o ELM para predecir la demanda de agua potable diaria para un horizonte de 1 y 3 días en la ciudad de Montreal (Canadá). Para evaluar la precisión de este modelo, los autores comparan sus resultados con las técnicas de regresión lineal múltiple, una red neuronal artificial y una máquina de soporte vectorial. Los mejores resultados se obtienen mediante el uso de la técnica de ELM, seguida por la máquina de soporte vectorial, la regresión lineal múltiple y finalmente la red neuronal artificial. Cabe destacar que el trabajo citado no aborda el problema de predicción desde un punto de vista univariante, sino que incluye la demanda media diaria de agua, la temperatura máxima, la precipitación total y la ocurrencia de precipitación.

Duerr *et al.* [15] comparan diversos métodos de *machine learning* para predecir la demanda mensual extraída de la agregación del consumo de los contadores de agua de la región de Tampa Bay en Florida (Estados Unidos). Los modelos utilizados en el trabajo son el algoritmo *Random Forest*, los árboles de regresión aditivos bayesianos (BART por sus siglas en inglés) y algoritmos de *Gradient boosting*. Estos modelos se comparan con métodos clásicos como el ARIMA, un AR(1) o la regresión lineal. Según las pruebas realizadas por los autores, los modelos clásicos de series temporales arrojan mejores resultados que los obtenidos por algoritmos de *machine learning*, siendo el AR(1) el que mejores pronósticos realiza.

En el trabajo realizado por Zubaidi *et al.* [24] analizan la influencia de factores exógenos climáticos en la predicción de la demanda a corto plazo, utilizando una serie temporal de consumo de agua con una granularidad diaria. Los datos exógenos utilizados son las temperaturas máxima, media y mínima, la precipitación, la evaporación, la radiación solar, la presión y la humedad relativa máxima. Los autores emplean una red neuronal artificial, que además combinan con dos algoritmos heurísticos para su afinación: *Backtracking Search Optimisation* (BSA) y *Gravitational Search* (GSA). Las conclusiones indican que el mejor modelo para la predicción de la demanda a corto plazo es el híbrido GSA-ANN. Sin embargo, no se realizan comparaciones con otras metodologías

de predicción de series temporales, ni se compara la precisión del pronóstico utilizando un enfoque univariante.

Guancheng *et al.* [25] utilizan un modelo de *deep learning* para la predicción de la demanda de agua a corto plazo con una serie de tiempo quinceminutal. El objetivo del trabajo es obtener la demanda de las próximas 24 horas comparando un modelo denominado *Gated Recurrent Unit Network* (GRUN) con una red neuronal artificial y un modelo SARIMA. Los resultados obtenidos muestran que el modelo GRUN realiza las mejores predicciones y las más robustas, aunque es el que más tiempo necesita para la realización de los cálculos.

Lee & Derrible [26] estudian la demanda agregada del consumo residencial y utilizan variables exógenas como la superficie de las viviendas, datos sobre las piscinas, el número de dormitorios, etc. Para la predicción utilizan múltiples modelos como el *Random Forest*, *Gradient Boosting Regression* (GBR), una máquina de soporte vectorial o una red neuronal artificial. Las conclusiones halladas por los autores indican que el modelo que mejores resultados arroja es el GBR, aunque alcanza un R^2_{adj} de 0.60 ± 0.13. Dejando a un lado el hecho de que los resultados obtenidos no gozan de una alta precisión, se observa que al igual que en el trabajo Zubaidi *et al.*, no se compara el enfoque multivariante con el univariante, de modo que no se puede apreciar el impacto de añadir nuevas variables a la predicción.

Bata *et al.* [27] realizan la predicción del consumo con datos horarios de 4 meses del año 2017. Los autores comparan un modelo ARIMA con un modelo híbrido basado en un árbol de regresión y el algoritmo SOM o *Kohonen Neural Networks* [28], una técnica de *clustering* para la reducción de la dimensionalidad. Los autores señalan que el modelo híbrido funciona mejor que el ARIMA, y que la precisión de la predicción aumenta conforme se incrementa el número de clústeres del algoritmo SOM.

Los mismos autores del último trabajo citado elaboraron otro estudio en el que compararon un modelo ARIMA con una red neuronal artificial y un modelo autorregresivo no lineal con factores exógenos o NARX [29]. Los autores usan factores climáticos y, al igual que antes, realizan la predicción a 24 horas

utilizando los datos horarios de los 4 meses previos, comparándolos esta vez con la realización de los cálculos con los datos de 1 y 5 años. Los resultados muestran que las redes neuronales y el modelo NARX mejora las predicciones frente al modelo ARIMA, y que, a su vez, la consideración de factores exógenos aumenta la precisión del pronóstico.

Li *et al.* [30] comparan diferentes modelos de *machine learning* con la técnica ARIMA para predecir la demanda de agua a 24 horas vista en la ciudad de Hefei (China) con diferentes resoluciones temporales (15 minutos, 1 hora y 24 horas), haciendo una comparación entre la predicción con series univariantes y multivariantes. Los autores también incorporan el modelo LSTM (*long short-term memory*), una técnica de *deep learning* capaz de detectar patrones en datos que muestran dependencias temporales con un rango amplio. Los resultados obtenidos muestran que los modelos basados en LSTM devuelven mejores predicciones que el resto, y se concluye que los factores exógenos tienen un impacto reducido sobre la mejora de la precisión de los modelos.

En general, de los trabajos revisados en este apartado se observa lo siguiente:

- De las investigaciones que incluyen factores exógenos, únicamente en una se concluye que el tratamiento del problema estudiado desde el punto de vista multivariante mejora la precisión de los pronósticos, por lo que no se puede concluir que la inclusión de variables adicionales mejore los resultados obtenidos de manera consistente.
- Los modelos de series temporales clásicos (AR o Holt-Winters) siguen ofreciendo resultados muy competitivos respecto a los obtenidos por los algoritmos de *machine learning.*
- Por norma general, los modelos de *deep learning* devuelven mejores resultados que los modelos clásicos o los de *machine learning*, con el inconveniente de requerir un mayor coste computacional.
- La variedad de conclusiones alcanzadas por los diferentes autores citados parece indicar que no existe un único modelo o técnica que sea capaz de predecir cualquier tipo de serie temporal de consumo de agua potable, por

lo cual el problema se debe abordar de una perspectiva más amplia, decidiendo cuál es la metodología aplicable a cada tipo de serie temporal

2.3 OPTIMIZACIÓN DE HIPERPARÁMETROS

De cara a realizar predicciones a corto plazo con una alta precisión se debe apostar por el uso de técnicas que cuenten, para cada problema, con la configuración óptima de parámetros. La optimización de estos parámetros o *hyperparameter tuning* es un problema que se ha tratado usando diferentes metodologías que se procederán a describir a continuación.

En el trabajo realizado por Bergstra & Bengio [31] se realiza un estudio detallado de las diferencias existentes entre las técnicas de *grid search* y *random search*. El primer método consiste en definir una serie de valores que deberán tomar los parámetros que se dispone a optimizar, y las pruebas realizadas consistirán en calcular todas las combinaciones posibles de dichos parámetros.

El problema principal que presenta este enfoque reside en el alto coste de computación, dado que, si se dispone de K variables L, el número de pruebas a realizar es de $S = \prod_{k=1}^{K} L^{(k)}$, que crecerá exponencialmente conforme se aumente el número de parámetros a optimizar. A esto hay que añadir que el espacio de búsqueda introducido puede no incluir la solución óptima para el problema, lo cual convierte a la técnica de *grid search* en una opción con importantes deficiencias.

En cuanto a la búsqueda aleatoria, los autores defienden que la eficiencia del método reside precisamente en aquello de lo que carece la búsqueda en malla, es decir, se pueden realizar menos pruebas para evaluar la precisión de las soluciones, a la vez que existe la posibilidad de explorar espacios de soluciones más amplios.

Si bien la búsqueda aleatoria puede presentar mejores resultados que la búsqueda en malla, ambas técnicas carecen de fiabilidad a la hora de explorar un espacio de soluciones, ya que el analista no tiene ninguna manera de verificar que la solución obtenida con alguno de los dos métodos sea un posible óptimo.

Trabajo	Modelos	Resumen	Factores exógenos	Comentarios
Antunes *et al.* [12]	SVR, ANN, KNN, RF, ARIMA	Predicción con un horizonte de 24 horas a partir de datos horarios en 3 sectores hidráulicos		Los hiperparámetros utilizados no se justifican. Los modelos de *machine learning* arrojan mejores resultados en las pruebas realizadas
Rodríguez Rangel *et al* [13].	24 modelos	Cada modelo se utiliza para la predicción de una hora concreta en 4 caudalímetros		Holt-Winters da los mejores resultados en 2 de 4 cuadalímetros. KNN devuelve resultados similares y mejores en el resto.
Mouatadid & Adamowski [14]	Extreme Learning Machines	Predicción del consumo diario para un horizonte de 1 y 3 días	X	Mejores resultados que la regresión lineal, SVR o ANN.
Duerr *et al.* [15]	RF, BART, Gradient boosting	Predicción de la demanda mensual agregada		AR(1) arroja los mejores resultados
Zubaidi *et al.* [24]	ANN	Influencia de factores exógenos climáticos sobre la predicción	X	No se compara la precisión de la predicción con un modelo univariante, ni se utilizan otros métodos de predicción

Guancheng *et al.* [25]	Gated Recurrent Unit Network (GRUN), ANN y SARIMA	Predicción con un horizonte de 24 horas de una serie con granularidad de 15 minutos		El método GRUN realiza las mejores predicciones y la más robustas, aunque su consumo de recursos es más intensivo
Bata *et al.* [27]	ARIMA, KNN+SOM	Predicción a 24 horas con datos de los últimos 4 meses.		El modelo KNN+SOM tiene un mejor rendimiento que el ARIMA, y su precisión aumenta conforme se incrementa el número de clústeres
Bata *et al.* [29]	ARIMA, ANN, NARX	Incorporación de factores climáticos. Predicción a 24 horas con datos de los últimos 4 meses, 1 año y 5 años	X	ANN y NARX realizan mejores predicciones que ARIMA. Los factores exógenos mejoran la precisión del pronóstico
Li *et al.* [30]	ARIMA, SVR, RF y LSTM	Predicción a 24 horas con diferentes resoluciones temporales, incluyendo factores climáticos	X	LSTM muestra mejores resultados que el resto de modelos, especialmente en resoluciones bajas (15 minutos). Los factores exógenos no varían la precisión de manera significativa

Tabla 2. Revisión bibliográfica de los trabajos realizados a partir de 2017

Por ello, se procederá a realizar un análisis de trabajos que hayan tratado la optimización de parámetros mediante el uso de algoritmos metaheurísticos.

Los algoritmos metaheurísticos son métodos que tratan de conseguir la mejor solución factible de todas las posibles soluciones de un problema de optimización [32]. Se trata de un conjunto de técnicas utilizadas para abordar problemas que no se pueden resolver mediante el uso de métodos exactos debido a que son altamente exigentes desde un punto de vista computacional.

Shih-Wei *et al.* [33] recurren al algoritmo *particle swarm* o enjambre de partículas para la optimización de los hiperparámetros de una máquina de soporte vectorial. Para evaluar su efectividad, abordan múltiples problemas de clasificación comunes en la literatura y comparan los resultados obtenidos con otros métodos, incluyendo la búsqueda en malla y la aplicación de un algoritmo genético. Los autores concluyen que el algoritmo utilizado mejora los resultados obtenidos mediante el uso de la búsqueda por malla, y obtiene precisiones similares a aquellas arrojadas por el algoritmo genético.

De manera similar al estudio realizado por Shih-Wei *et al.*, Aljarah *et al.* [34] abordan el problema de la optimización de una máquina de soporte vectorial sobre los mismos conjuntos de datos, aunque en esta ocasión se compara un mayor número de algoritmos. Los autores señalan que el algoritmo propuesto (*Grasshopper Optimization Algorithm*) mejora los resultados obtenidos por otros algoritmos como el enjambre de partículas o el algoritmo genético.

Ji-Hoon *et al.* [35] se optimizan los hiperparámetros de una red neuronal convolucional mediante el uso de un algoritmo genético para abordar los problemas de identificación de números manuscritos y la clasificación de motores defectuosos. Los parámetros objeto de optimización son la tasa de aprendizaje, el factor de inercia, el tamaño del filtro, el número de filtros, el número de neuronas y el tamaño de lote. En el estudio se observa que los parámetros obtenidos mediante el uso del algoritmo genético arrojan mejores resultados que aquellos derivados de realizar búsquedas en malla o aleatorias, introduciendo también una mejora en el tiempo de cálculo y del entrenamiento.

Algunos de los algoritmos y las técnicas mencionadas en los párrafos anteriores están implementadas en paquetes de software de código abierto, y se encuentran a disposición de los usuarios y de los analistas para su uso.

Debido a que el código del presente trabajo se encuentra redactado en Python, únicamente se mencionarán aquellas librerías presentes en este lenguaje. De las librerías más prominentes, cabe destacar algunas como Optunity [36], Hyperopt [37], Sherpa [38] u Optuna [39].

Una vez realizado un análisis de las librerías mencionadas, se observan lo siguiente:

- La documentación ofrecida por los autores se centra fundamentalmente en la optimización de redes neuronales, no haciendo hincapié en algoritmos de *machine learning*.
- Dado el punto anterior, la adaptación de este tipo de herramientas a problemas específicos requiere la modificación del código fuente, extremo que queda fuera del alcance de presente trabajo.
- Los trabajos que respaldan las librerías mencionadas no detallan qué tipo de algoritmos son utilizados, por lo que la metodología de optimización no es transparente a ojos del analista.

Por lo tanto, de la revisión bibliográfica efectuada en este punto se desprende que la optimización de parámetros de los algoritmos usados para alcanzar los objetivos de este trabajo requiere el uso de algoritmos metaheurísticos, ya que mejoran los resultados de predicción obtenidos por las otras técnicas analizadas, y realizan los cálculos en menos tiempo. Además, dados los inconvenientes planteados por las librerías de optimización de uso común en Python, se propone la elaboración de un algoritmo genético específico para el problema abordado.

3 DESCRIPCIÓN DE LOS DATOS

Los datos utilizados para la realización del presente trabajo consisten en una serie temporal de consumo de agua potable con registros disponibles entre el 1 de enero de 2018 y el 29 de febrero de 2020, y han sido proporcionados por la empresa Global Omnium Idrica.

Estos datos provienen de la lectura de un caudalímetro ubicado en el sector hidráulico de Calvari, en el municipio de Benetússer (Valencia), y contienen lecturas del caudal y de la presión cada 15 minutos. Dado el enfoque de este trabajo, únicamente serán considerados los datos de caudal, desechando así los de presión. La serie cuenta principalmente con dos estacionalidades: una diaria que consta de 96 instantes de tiempo y otra semanal con 673 instantes.

La serie temporal estudiada proviene de las bases de datos de la empresa suministradora, por lo que ya se han realizado algunas correcciones o rectificaciones que incluyen la eliminación de valores fuera del rango del caudalímetro, así como de aquellos registros imposibles (datos negativos, cadenas de caracteres, etc.). Sin embargo, los datos estudiados cuentan con una serie de fallos y de particularidades que deberán ser tratados para poder realizar las predicciones posteriores.

En primer lugar, la serie presenta datos faltantes en múltiples instantes de tiempo. En la Ilustración 1 se pueden observar los momentos en los que se carece de información. Como podrá apreciar el lector, los bloques de datos perdidos presentan diferentes tamaños, siendo algunos de ellos de un mes completo. Es el caso de los meses de mayo y agosto de 2018 y abril y octubre de 2019. Estos datos serán reconstruidos de acuerdo con la metodología propuesta en el punto 4.1.1.

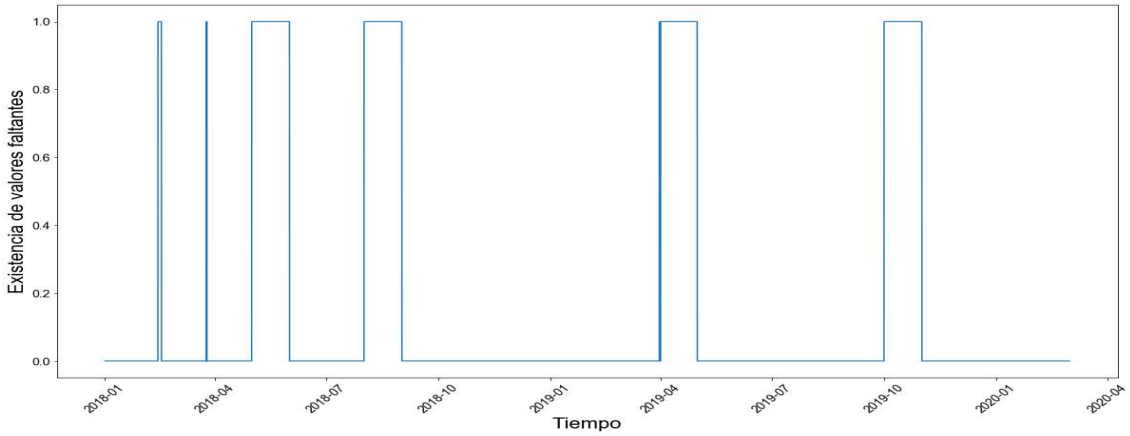

Ilustración 1. Detección de valores faltantes en la serie temporal estudiada. Los espacios que alcanzan el valor de 1 son aquellos en los que se han hallado valores faltantes.

Por otro lado, la serie cuenta con numerosos datos anómalos, a los que se deberá atender de manera especial. Un ejemplo de día con valores especialmente anómalos es el 12 de junio de 2018 (Ilustración 2). En este día se puede apreciar una bajada repentina en el consumo que se aproxima a valores prácticamente nulos, y que posteriormente desemboca en una subida en el caudal que llega hasta los 120 m^3/h. No existen elementos de juicio suficientes para dictaminar que este consumo anómalo sea un error de lectura, por la que la naturaleza de este y otros datos anómalos no será analizada de manera pormenorizada, sino que se les aplicará la metodología propuesta en el punto 4.1.2.

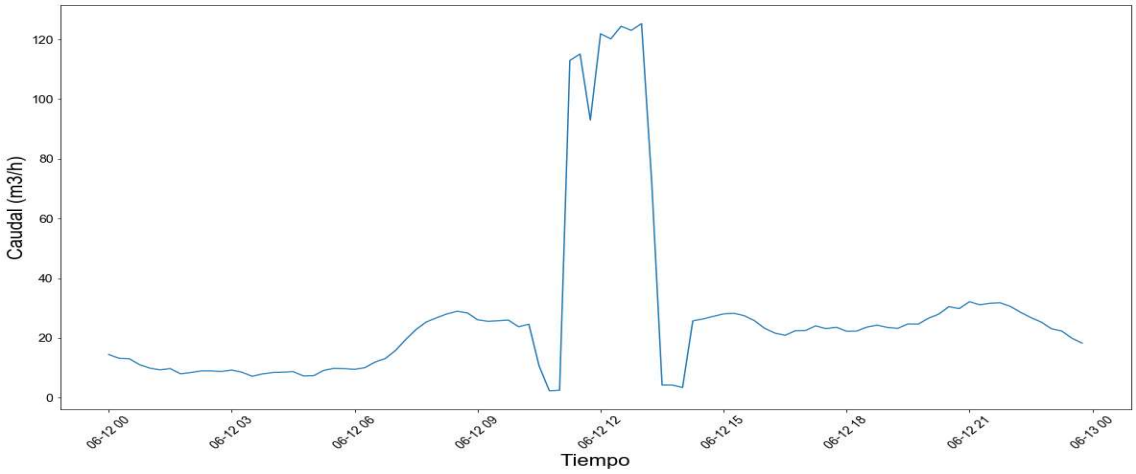

Ilustración 2. Consumo de agua potable del día 12 de junio de 2018

4 METODOLOGÍA

La metodología empleada para el desarrollo de este trabajo consiste en la aplicación de una serie de técnicas cuantitativas para la reconstrucción, validación y predicción de series temporales de consumo de agua potable, atajando así la problemática expuesta en el apartado 2.

Para llevar a cabo esta tarea, se ha utilizado el software Python 3.8 [40], el cual es un lenguaje de programación multipropósito que en los últimos años ha adquirido una gran relevancia en los campos del *machine learning*, el *deep learning* y el cálculo técnico y científico.

La metodología desarrollada en el presente apartado se ha implementado mediante un paquete de cálculo que se ha denominado *waTS*, y comprende las clases y los métodos necesarios para la aplicación de dicha metodología. Este paquete contiene código elaborado por el autor de este trabajo, así como librerías creadas por terceros.

El repositorio que contiene el código y los datos utilizados en este trabajo puede ser consultado en el siguiente enlace: https://github.com/Fidaeic/TFM, y la descripción de su contenido se puede encontrar en el apartado 5 de este documento, así como en el propio enlace adjunto.

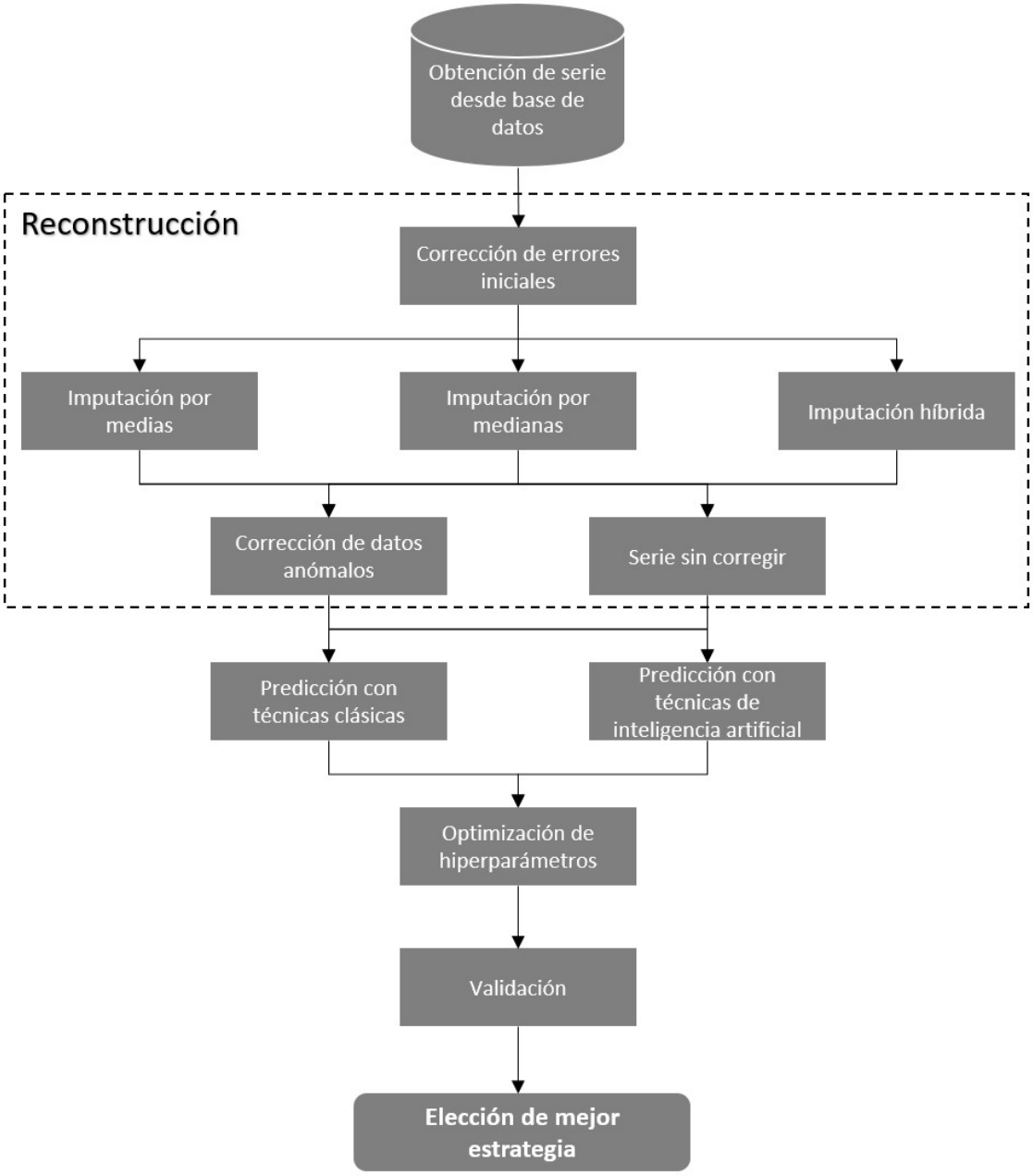

Ilustración 3. Metodología general

4.1 RECONSTRUCCIÓN DE SERIES TEMPORALES

4.1.1 Datos faltantes

Para la imputación de datos faltantes en la serie temporal estudiada se han escogido diferentes estrategias que se proceden a describir a continuación.

En primer lugar, se establece una regla en la cual para cada instante *t* en el que se disponga de un dato faltante, se introduce la media de los mismos instantes de las *n* últimas semanas, de modo que el valor imputado viene dado por la expresión (6).

$$y_t = \frac{\sum_{i=1}^{n} y_{t-96*i}}{n}; i = \{1, 2, \dots, n\} \quad (6)$$

Del mismo modo, se ha desarrollado un segundo método en el que en lugar de tener en cuenta la media de las últimas *n* semanas, se toma en consideración la mediana por considerarlo un estimador más robusto e insesgado del comportamiento de la serie temporal.

Finalmente, se elabora un tercer método que, en caso de tener un conjunto de datos faltantes inferior a una estacionalidad, se utilizan métodos de predicción de series temporales basados en modelos estadísticos clásicos o en algoritmos de *machine learning*. En el caso contrario, es decir, si se dispone de un conjunto de datos cuyo tamaño es superior a la estacionalidad definida, se utiliza uno de los dos métodos descritos inicialmente. En la Ilustración 4 se muestra un resumen esquemático del funcionamiento del método descrito.

Ilustración 4. Cuadro resumen del método de imputación híbrido.

La idoneidad del método elegido para la imputación de datos vendrá dada por la precisión que tengan en última instancia las diferentes técnicas de predicción evaluadas.

4.1.2 Valores anómalos

Una vez realizada la imputación de los datos faltantes, se procede al estudio de los valores anómalos, para lo cual se ha tomado como referencia el trabajo realizado por Loureiro *et al.* [18], y se utilizará la expresión (1) para determinar la región de anómalos.

Previamente al análisis de los valores anómalos se debe realizar una segmentación de los días que componen la serie temporal para tener referencias robustas con las que comparar cada día. Si bien en el trabajo citado en el párrafo anterior se propone la división de los datos en 2 tipos (laborables y fines de semana), en este trabajo se ha optado por utilizar una metodología diferente basada en la complejidad computacional de los cálculos.

En caso de contar con un número reducido de segmentos, se da el problema de la complejidad de cálculo. Como puede apreciar el lector, en la expresión (2) aparece un término en el que se realiza una diferencia por pares de todos los elementos que componen un clúster. Para ilustrar este caso, se propone un

segmento en el que se dispone de *n* valores del caudal, para el cual se deben calcular las restas de los pares sin repetición:

$$\binom{n}{r} = \frac{n!}{r!\,(n-r)!} = \frac{n!}{2!\,(n-2)!} \quad (7)$$

Si se toma *n* como el número de instantes presentes en un día (se dispone de 96 valores del caudal por día), es posible conocer el número de operaciones necesarias para cada día que se añadiese, tal y como queda reflejado en la Tabla 3.

Número de días	Número de operaciones elementales
1	4560
2	18.336
3	41.328
4	73.536
5	114.960
6	165.600

Tabla 3. Incremento del número de operaciones necesarias en función de los días contenidos en cada clúster

Se puede observar que la relación existente entre el número de operaciones en función de los días que componen un segmento sigue un patrón cuadrático que se apoya en la expresión $\frac{n^2}{2} - \frac{n}{2}$, por lo que se puede afirmar que la complejidad del cálculo es $O(n^2)$.

Asumiendo un número de segmentos igual al utilizado por Loureiro *et al.* [18] en su estudio, y suponiendo que en un año se cuenta con aproximadamente 260 días laborables, en el caso de este trabajo el número de operaciones a realizar teniendo en cuenta que se dispone de 2 años de datos es de 1.245.978.240, lo cual queda fuera de las capacidades computacionales del equipo con el que se ha desarrollado el presente proyecto.

Por lo tanto, dadas las condiciones expuestas en este apartado y con el fin de efectuar un cálculo de valores anómalos basado en patrones de consumo, se realizará una división aun mayor de los días analizados, para lo cual a la segmentación propuesta por los autores anteriores se añadirá la relativa a las

estaciones del año. De este modo, se pasará de 2 a 8 clústeres, simplificando de manera significativa los cálculos realizados.

Una vez detectados los valores anómalos, se aplicará la metodología planteada en este apartado con y sin corrección de dichos valores. La corrección, en caso de ser aplicada, consistirá en la sustitución de los consumos extremos por la mediana de su segmento correspondiente.

De esta manera, el objetivo es conseguir series temporales más suavizadas que ayuden a mejorar las cualidades de la predicción. La idoneidad de esta estrategia vendrá dada, al igual que se expresó en el apartado anterior, por la precisión conseguida en la predicción.

4.2 PREDICCIÓN

Una vez reconstruida la serie y tratados los datos anómalos según los criterios establecidos en el punto 4.2, se procede a la evaluación de los diferentes métodos de predicción de series temporales.

El algoritmo utilizado para la realización de la predicción es como sigue, siendo la variable *horizonte* el número de instantes que deberán ser predichos por el algoritmo.

Algoritmo para la predicción de series temporales

Inicio del algoritmo

Creación de matriz con *t* instantes desfasados

Separación de la matriz en conjuntos de entrenamiento, validación y predicción

Ajuste de modelo a conjunto de entrenamiento

Validación del modelo

Para *i*=1 hasta *horizonte:*

    Generación de nueva fila de datos partiendo de la anterior desfasando 1 instante de tiempo

    Predicción del instante *i*

    Imputación de la predicción en el nuevo registro

Fin del bucle

Evaluación de la predicción frente a datos reales

Fin del algoritmo

La matriz para la realización de los cálculos de predicción tendrá tantas columnas como instantes de tiempo se hayan desfasado para tener en consideración la autocorrelación de los datos. Por otro lado, se plantea la variable *horizonte* como el número de predicciones *out of bag* o fuera del conjunto muestreado sobre el cual se realizará la predicción.

En los siguientes apartados se procede a realizar una descripción somera de los modelos utilizados en la fase de predicción.

Árboles de decisión

Son técnicas de regresión que consisten en la toma de decisiones de manera secuencial en función de operaciones lógicas a las que se someten las variables explicativas del conjunto de datos. Los árboles se componen de:

- Raíz. Es el nodo inicial. De este empiezan a surgir nodos en los que se van tomando decisiones, que se denominan hijos. Este nodo no tiene padre.
- Nodo interno. Es similar a la raíz, pero se diferencia en que sí tiene padre

- Hoja. Son nodos finales ya que no tienen hijos, y en ellos se encuentran las predicciones [41].

Lo que se busca es que las hojas sean lo más puras posible, lo cual se puede medir con la entropía o el índice de Gini, que son indicadores que representan el coste de realizar un error en la predicción [42].

Random Forest

La técnica de *Random Forest* utiliza un conjunto de árboles de predicción de manera que cada árbol depende de los valores de un vector aleatorio muestreado de manera independiente y con la misma distribución para todos los árboles del bosque.

Se trata de una técnica útil tanto para la clasificación como para la regresión. En cuanto a esto último, un modelo de regresión basado en *Random Forest* consiste en un conjunto de árboles crecientes que dependen de un vector Θ tal que cada árbol de predicción $h(x, \Theta)$ utiliza valores numéricos. El bosque de predicción se forma tomando la media de *k* árboles $\{h(x, \Theta_k)\}$ [43].

Los *k* árboles utilizados provienen del uso de *k* muestras obtenidas mediante la técnica de *bagging* (**B**ootstrap **agg**regat**ing**). Este método consiste en tomar un conjunto de datos de entrenamiento L de tamaño *n* y replicarlo *m* veces, de manera que se consiguen una serie de conjuntos de datos de entrenamiento L^B con remuestreo y sin sustitución con el objetivo de tener para cada muestra un grupo de datos incluido o *in bag* y otro grupo excluido o *out of bag* [44].

Esta técnica se incluye dentro del dominio de los *ensemble methods* o métodos combinados, que funcionan como una amalgama de diferentes algoritmos de regresión o clasificación para mejorar los resultados arrojados por cada uno de ellos [45]. Los valores devueltos por el modelo consisten en una combinación lineal de los resultados de cada uno de los regresores que componen el conjunto (habitualmente la media de estos) [46].

Como ventaja de esta técnica sobre otras de regresión cabe destacar que el algoritmo devuelve la importancia de las variables para la obtención de las

mejores predicciones, lo cual es útil para la identificación de factores con alta significación estadística para la realización de predicciones.

Support Vector Machines

Las técnicas de Máquinas de Soporte Vectorial o *Support Vector Machines* (SVM) tienen como origen los problemas de clasificación, en los que el algoritmo resuelve problemas de clasificación binaria mediante el cálculo de límites de separación óptimos para las dos clases del conjunto de datos estudiado.

La técnica de Support Vector Regression es una generalización de la técnica de clasificación, que introduce un parámetro ε mediante el cual se define una región alrededor de dicho límite de separación, y una función de coste que debe ser minimizada para encontrar el ε óptimo. De esta manera, se construye una función multiobjetivo a partir de la función de coste y las propiedades geométricas de la región del parámetro ε [47].

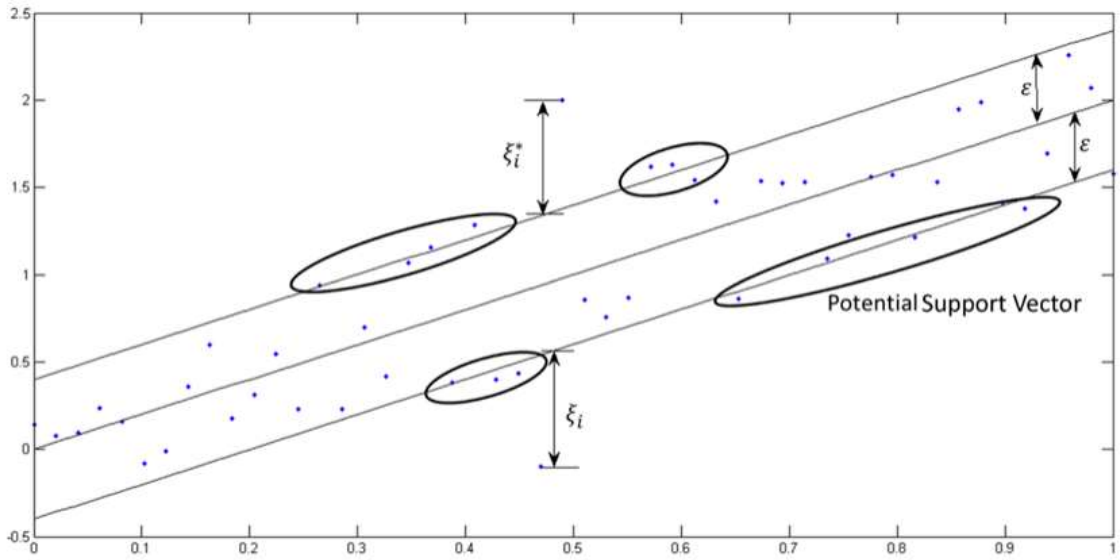

Ilustración 5. SVM para la regresión lineal unidimensional. Fuente: Awwad & Khanna [47]

Gradient Boosting

El modelo *Gradient Boosting* es una técnica propuesta por Friedman *et al.* [48] que plantea un sistema en el que se utilizan múltiples regresores "débiles" de manera secuencial que son combinados para producir resultados mejores mediante la minimización de una función de coste *L* que se puede simplificar según la expresión (8), donde se tienen unos regresores base b caracterizados por un conjunto de parámetros γ y unas variables de entrada x_i , unos coeficientes de expansión β_m, con m = 1,2,…,M y una variable de salida y_i [49].

$$\min_{\beta,\gamma} \sum_{i=1}^{N} L\big(y_i, \beta b(x_i;\gamma)\big) \qquad (8)$$

A pesar del parecido que pueden tener los modelos de *Gradient Boosting* con los *ensemble methods*, existe una diferencia sustancial: en estos últimos se evalúan diferentes modelos y se obtiene un resultado a partir de la combinación lineal del conjunto de los resultados obtenidos, mientras que en los primeros se ajustan nuevos modelos de manera consecutiva que buscan la máxima correlación con el gradiente negativo de la función de coste asociada al conjunto de modelos. La ventaja en este sentido es que el analista puede decidir cuál es la función de coste adecuada para cada problema tratado [50].

Redes neuronales artificiales

Las redes neuronales artificiales constituyen un modelo inspirado en el funcionamiento del cerebro de los animales en la medida en la que existen múltiples nodos interconectados que pueden procesar información en paralelo [51].

Un modelo de este tipo se basa en una capa de entrada compuesta por tantos nodos como variables de entrada se tengan, una o varias capas ocultas o intermedias, y una capa de salida en la que se recogen los resultados de la predicción. Cada conexión existente entre un nodo y todos los posteriores lleva asociada un peso.

El resultado del procesamiento realizado por la neurona *i* en una capa oculta viene dado por la expresión (9), donde $\sigma(\,)$ es la función de activación o transferencia, *N* es el número de neuronas de entrada, V_{ij} es el conjunto de los pesos, x_j son los datos de entrada a las neuronas y T_i^{hid} es el umbral de los términos de las neuronas ocultas [52].

$$h_i = \sigma(\sum_{j=1}^{N}(V_{ij}x_j + T_i^{hid}) \tag{9}$$

Las funciones σ son usadas para transformar el nivel de activación de una neurona en una señal de salida, por lo que actúan como funciones de transferencia entre capas. Para una definición en profundidad de las diferentes funciones de activación existentes, se recomienda el trabajo de Karlik & Olgac. [53].

En el presente trabajo se utilizará *rectified linear units* (ReLU) como función de activación [54], dados sus buenos resultados a la hora de mejorar la velocidad de aprendizaje de diferentes redes neuronales [55]. El alto rendimiento de esta función de activación la ha convertido en el método utilizado por defecto para la confección de redes neuronales [56].

4.3 OPTIMIZACIÓN DE HIPERPARÁMETROS

Tal y como se adelantó en el punto 2.3, en este trabajo se procederá a generar una versión propia del algoritmo genético con el objetivo de optimizar los hiperparámetros de los algoritmos de predicción descritos en el apartado 4.2.

Los genéticos son un conjunto de algoritmos metaheurísticos que se basan en la evolución y la supervivencia de las especies para la selección de los individuos más adecuados entre una población [57]. Partiendo de una población inicial, se aplican mecanismos de cruce y mutación entre individuos que se evalúan mediante una función de coste para determinar cuáles de ellos pasan a la siguiente generación y así formar una nueva población de tamaño más reducido. El proceso es iterativo y el individuo resultante es la solución que optimiza la función de coste seleccionada [58]. En este caso, la función de coste a minimizar es la descrita en el apartado 4.4.

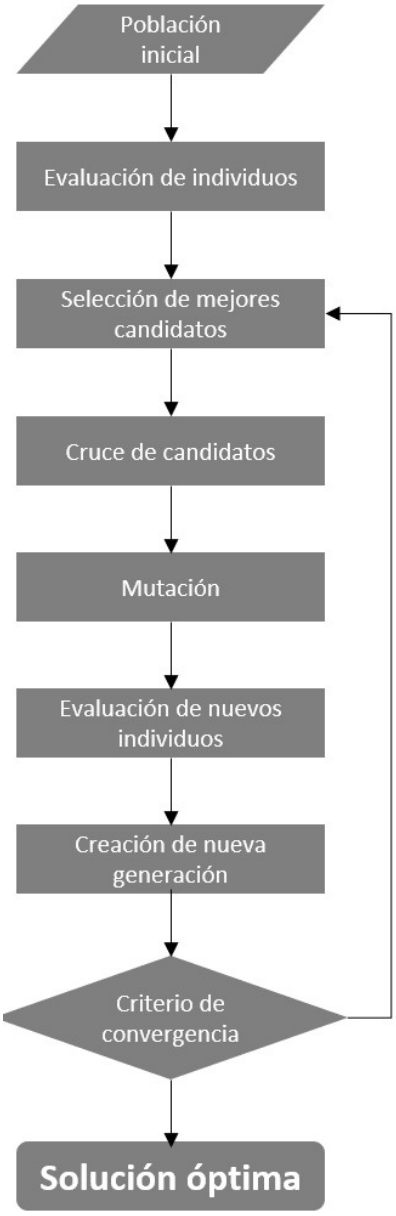

Ilustración 6. Flujo de trabajo del algoritmo genético. Fuente [58]

Para una descripción en profundidad del funcionamiento de los algoritmos genéticos, se recomienda el trabajo elaborado por Whitley [59], en el que se analiza la motivación, así como la metodología utilizada en este tipo de algoritmos.

En el Anexo II del presente trabajo se incluye la relación de hiperparámetros optimizados, así como los valores resultantes de la aplicación del algoritmo genético.

4.4 EVALUACIÓN

La evaluación de las diferentes fases del modelo consistirá en la comparación de los resultados con base en las siguientes métricas:

R^2:

$$1 - \frac{SCR}{SCT} \tag{10}$$

SMAPE:

$$SMAPE = \frac{|x - y|}{\frac{|x| + |y|}{2}} \tag{11}$$

RMSE:

$$\sqrt{\frac{\sum_{i=1}^{n}(\hat{y}_i - y_i)}{n}} \tag{12}$$

Durante la fase de optimización de parámetros descrita en el apartado 4.3, se utilizará el RMSE como función de coste, y el algoritmo genético buscará su minimización para cada modelo predictivo, teniendo en consideración únicamente los conjuntos de entrenamiento y validación, y no el de predicción.

Cabe destacar SMAPE (*Symmetric mean absolute percentage error*) como métrica de validación. Este método presenta una ventaja frente a las métricas basadas en los mínimos cuadrados ya que no se ve afectado por problemas de escala de los datos, y a que no se producen situaciones de indeterminación cuando el valor real de una observación es nulo [60].

Una vez obtenidos los resultados de todos los modelos de predicción, se procederá a realizar una comparación entre los métodos usados en las diferentes fases que componen la metodología descrita con el objetivo de determinar, para cada una de ellas, cuál es la estrategia de cálculo óptima. En el caso de que los resultados de la comparación no presenten diferencias estadísticamente significativas, se procederá a seleccionar aquella técnica que menos tiempo haya requerido para el cálculo, y que por lo tanto menos consumo de recursos exige.

Para determinar si existen diferencias significativas entre los resultados obtenidos por un método u otro se recurrirá a las pruebas de hipótesis no paramétricas de Levene [61] y de Kruskal-Wallis [62], mediante las cuales se compararán las varianzas y las medianas, respectivamente, de las poblaciones resultantes, tomando como criterio de comparación el RMSE resultante.

Las pruebas mencionadas toman como hipótesis nula la igualdad de las varianzas o de las medianas, según el caso, de las poblaciones comparadas. Un P-valor menor que 0.05 implica que existen diferencias estadísticamente significativas entre las poblaciones estudiadas, por lo que, ante esta situación, se analizará el signo de esta diferencia para decidir qué población devuelve mejores resultados.

En el caso de que dos poblaciones no presenten diferencias significativas en términos del error de predicción, se utilizará como criterio de desempate el tiempo requerido para la ejecución del cálculo.

5 DESCRIPCIÓN DEL REPOSITORIO DE RESULTADOS

El paquete de cálculo desarrollado para la realización de este trabajo recoge los elementos descritos en los apartados anteriores, y contiene las estructuras de datos necesarias para el análisis, la reconstrucción, la validación y la predicción de la serie temporal estudiada o de otras similares.

La estructura del repositorio, que se puede encontrar en el enlace https://github.com/Fidaeic/TFM, es la que se detalla a continuación:

- Código:
    - **main.py:** Script principal, en el que se ejecutan todas las posibles combinaciones de métodos desarrollados en waTS.py, y que sirve para decidir la estrategia óptima de imputación, reconstrucción y predicción.
    - **waTS.py:** Se trata del fichero principal, en el que se encuentran implementados todos los objetos, así como los métodos y atributos asociados para el desarrollo de los cálculos. La descripción de los objetos contenidos en este fichero se encuentra en el Anexo I.
    - **ga.py:** En este fichero se encuentra la implementación del algoritmo genético desarrollado para la optimización de los hiperparámetros de los algoritmos de predicción.
    - **ga_example:** Ejemplo de la optimización de un algoritmo mediante el uso del script desarrollado en ga.py.
    - **metrics.py:** Fichero en el que se recogen algunas métricas de validación utilizadas a lo largo del trabajo.
    - **results.py:** Script en el que se analizan los resultados mediante gráficos y pruebas de hipótesis.
- Data: Carpeta en la que se almacenan los datos de la serie temporal original, previamente a su manipulación
- Predictions: Carpetas en las que se almacenan los resultados de las predicciones realizadas para las 144 combinaciones calculadas
- Results: Carpeta en la que se almacenan los datos reales que se intentan predecir, los resultados de todas las combinaciones y los tiempos de

ejecución. También se incluye un fichero en el que se encuentran los resultados del mejor modelo obtenido, junto con un intervalo de confianza del 95%.

Adicionalmente, se ha utilizado una serie de paquetes desarrollados por terceros, que son los siguientes: Numpy [63], Pandas [64], Scikit Learn [65], Keras [66], Statsmodels [67], Pmdarima [68] y Scipy [69].

6 RESULTADOS

6.1 RECONSTRUCCIÓN

Las diferentes técnicas aplicadas para la reconstrucción de la serie temporal en materia de datos faltantes varían tanto en la precisión de las predicciones resultantes como en los tiempos de cálculo. A la hora de tomar la decisión de cuál es la estrategia óptima para efectuar esta tarea, en caso de que las diferencias entre los modelos aplicados no sean significativas, se utilizará como criterio de desempate el tiempo requerido para la ejecución del cálculo.

6.1.1 Datos faltantes

En la Ilustración 7 se puede ver cómo, a priori, no existen diferencias significativas en los métodos de imputación de valores faltantes de cara a los resultados de la predicción. Todos los métodos propuestos se caracterizan por presentar una variabilidad alta, destacando los modelos *hybrid-KNN-mean* y *hybrid-ARIMA-mean* por dar resultados relativamente más robustos.

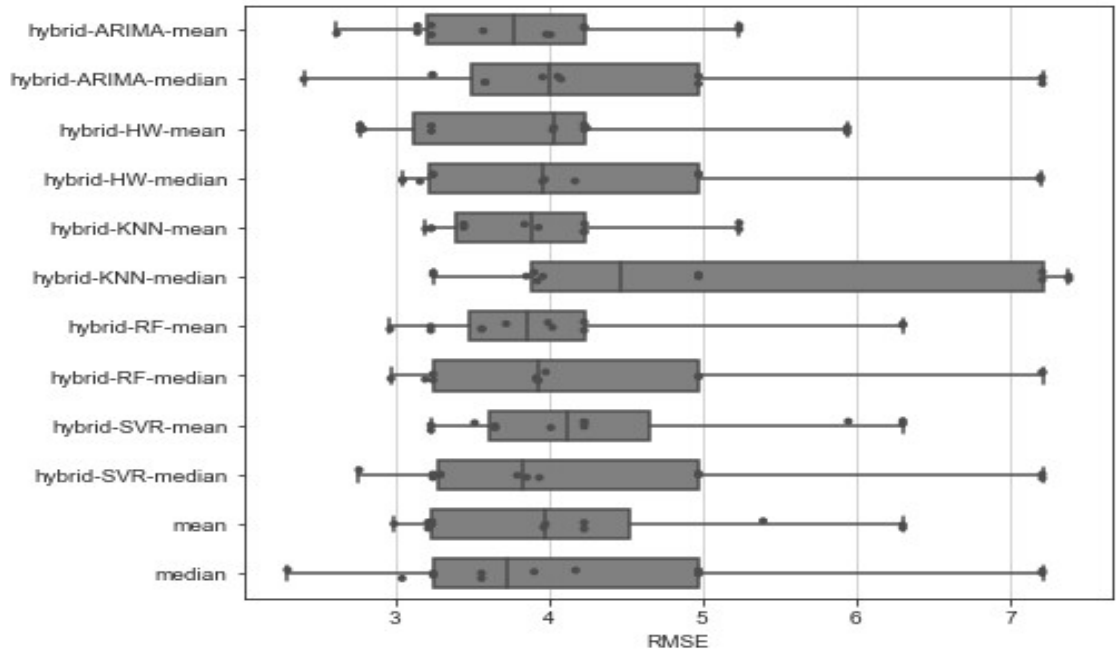

Ilustración 7. Comparación de los resultados finales de la predicción en función del método de imputación de valores faltantes

Para determinar si existen diferencias estadísticamente significativas entre las poblaciones, se procede a usar la prueba de Levene para la comparación de

varianzas. En el caso de que exista homocedasticidad entre las pruebas realizadas, se procederá a analizar la diferencia entre las medianas mediante la prueba de Kruskal-Wallis.

	1	2	3	4	5	6	7	8	9	10	11	12
1	1.00											
2	0.29	1.00										
3	0.54	0.56	1.00									
4	0.25	0.96	0.51	1.00								
5	0.56	0.17	0.29	0.14	1.00							
6	**0.04**	0.41	0.12	0.43	**0.02**	1.00						
7	0.80	0.45	0.80	0.41	0.52	0.10	1.00					
8	0.38	0.90	0.67	0.86	0.24	0.35	0.55	1.00				
9	0.54	0.60	0.97	0.55	0.31	0.14	0.78	0.71	1.00			
10	0.30	0.99	0.57	0.96	0.19	0.42	0.46	0.90	0.60	1.00		
11	0.42	0.69	0.83	0.64	0.23	0.18	0.66	0.81	0.87	0.69	1.00	
12	0.22	0.86	0.45	0.90	0.13	0.53	0.36	0.77	0.48	0.87	0.56	1.00

1	*hybrid-ARIMA-mean*	***7***	*hybrid-RF-mean*
2	*hybrid-ARIMA-median*	***8***	*hybrid-RF-median*
3	*hybrid-HW-mean*	***9***	*hybrid-SVR-mean*
4	*hybrid-HW-median*	***10***	*hybrid-SVR-median*
5	*hybrid-KNN-mean*	***11***	*mean*
6	*hybrid-KNN-median*	***12***	*median*

Tabla 4. P-valores resultantes de la realización de la prueba de hipótesis de Levene para la comparación de varianzas según el método de imputación de datos faltantes

Como se puede comprobar, únicamente el modelo 6 (*hybrid-KNN-median*) presenta diferencias significativas en la varianza con los métodos 1 (*hybrid-ARIMA-mean*) y 3 (*hybrid-HW-mean*).

	1	2	3	4	5	6	7	8	9	10	11	12
1	1.00											
2	0.36	1.00										
3	0.77	0.52	1.00									
4	0.69	0.73	0.64	1.00								
5	0.69	0.60	0.91	0.95	1.00							
6	0.08	0.27	0.13	0.15	0.12	1.00						
7	0.64	0.56	0.91	0.95	0.95	0.13	1.00					
8	0.56	0.86	0.64	0.77	0.82	0.22	0.82	1.00				
9	0.18	0.86	0.39	0.56	0.45	0.30	0.35	0.73	1.00			
10	0.52	0.75	0.69	0.77	0.91	0.12	0.91	0.73	0.69	1.00		
11	0.52	0.60	0.69	0.95	0.91	0.17	0.95	0.86	0.39	0.77	1.00	
12	0.69	0.77	0.69	0.86	0.95	0.20	0.95	0.88	0.60	0.91	0.86	1.00

Tabla 5. P-valores resultantes de la realización de la prueba de hipótesis de Kruskal-Wallis para la comparación de medianas según el método de imputación de datos faltantes

En cuanto a la diferencia de medianas, ningún modelo muestra diferencias significativas en los resultados finales de la predicción, por lo que el criterio que se utilizará para determinar cuál es el mejor modelo es el tiempo necesario para el cálculo. Como se puede observar en la Ilustración 8, los modelos que presentan tiempos de cálculo más reducidos son la mediana y la media, con un tiempo de ejecución de 17.94 segundos y 21.39 segundos respectivamente.

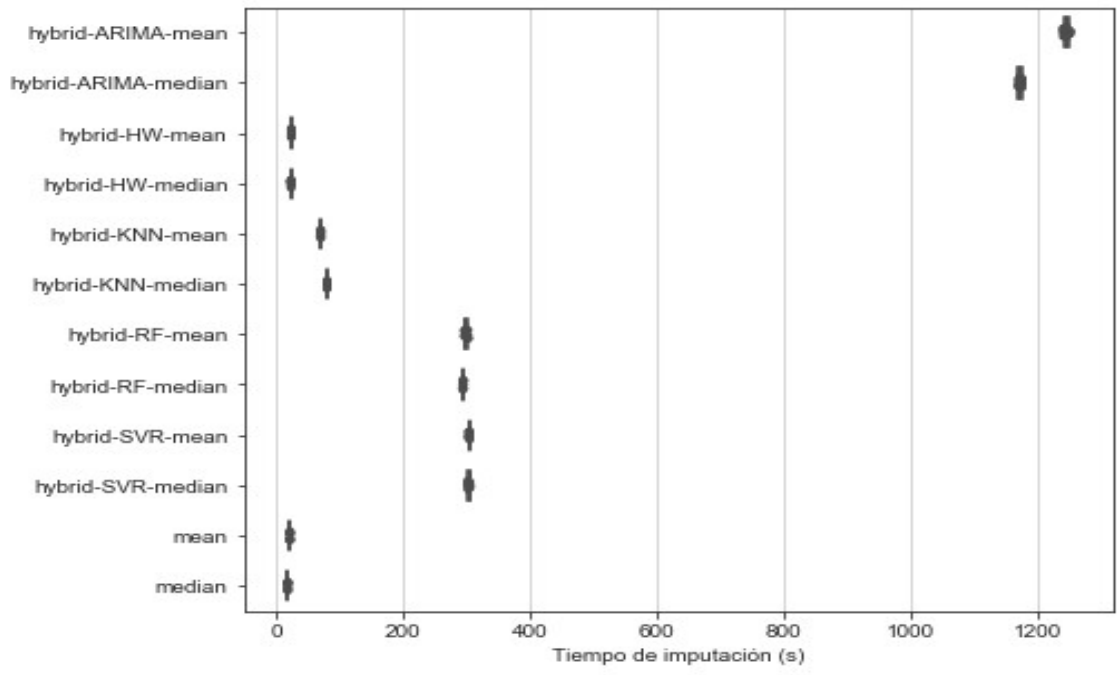

Ilustración 8. Tiempo necesario para los cálculos de imputación de valores faltantes

Por lo tanto, en términos de elección de un modelo para la imputación de valores faltantes, **el método de la mediana se muestra como un candidato robusto en cuanto a los resultados de predicción y rápido en lo que respecta a los tiempos de cálculo.**

6.1.2 Valores anómalos

La Ilustración 9 recoge un ejemplo en el cual se ha realizado la corrección por valores anómalos en un día en el que se producen lecturas significativamente diferentes a la mediana del clúster al que pertenece dicho día. Se puede observar cómo, tras la aplicación de la metodología propuesta, han sido eliminados aquellos valores superiores o inferiores a los límites de la región de anómalos [18], y posteriormente han sido sustituidos por la mediana del clúster preceptivo.

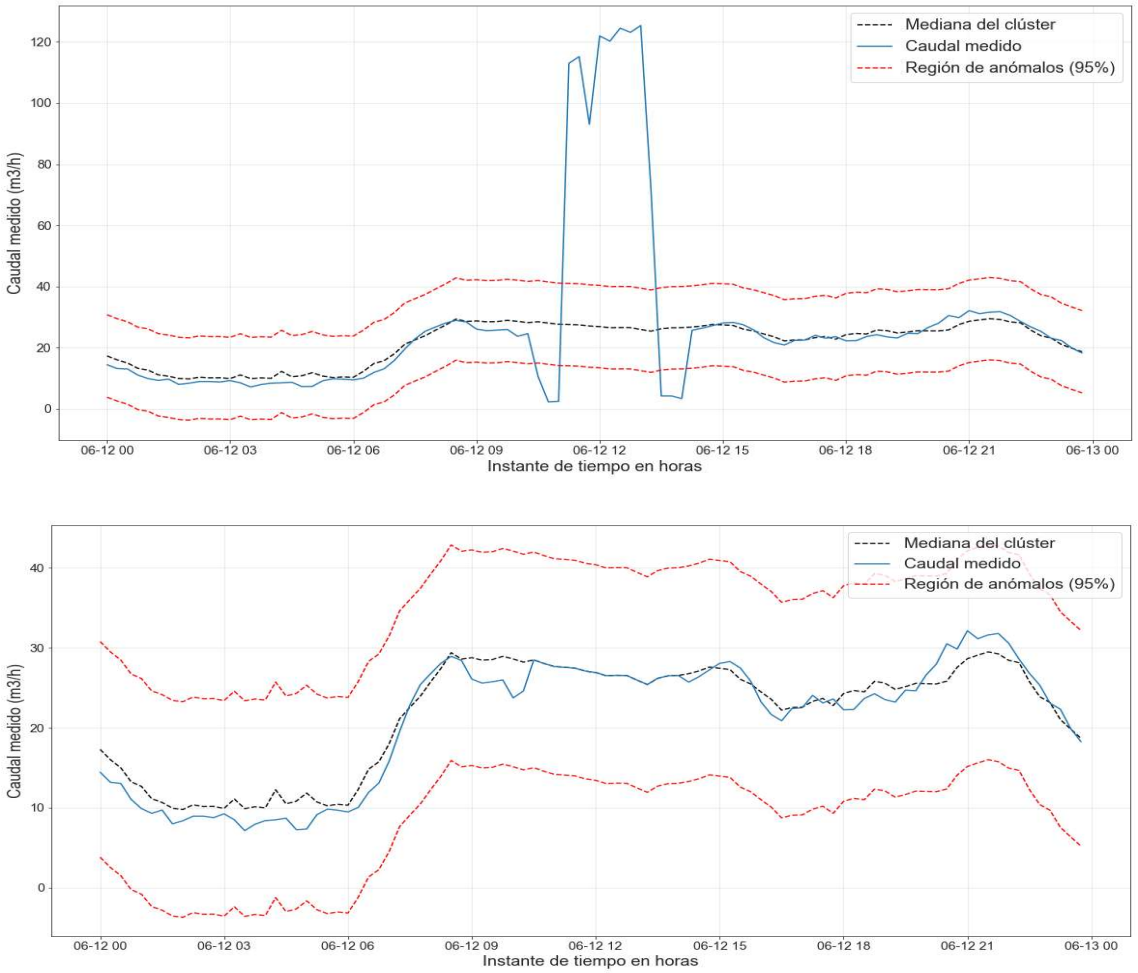

Ilustración 9. Ejemplo de corrección por valores anómalos del día 12 de junio de 2018 para una región de anómalos con un riesgo de primera especie del 95%

La Ilustración 10 muestra una comparación gráfica de los resultados obtenidos en los casos en los que se ha corregido la serie por valores anómalos y aquellos en los que no se ha realizado esta corrección. Como se puede apreciar, no parecen existir diferencias significativas entre los dos modelos, lo cual se confirma mediante las pruebas de hipótesis de Levene y de Kruskal-Wallis, que devuelven unos P-valores de 0.85358 y de 0.57159 respectivamente, llevando a aceptar la hipótesis nula de que no existen diferencias significativas entre las varianzas ni las medianas de las dos poblaciones estudiadas.

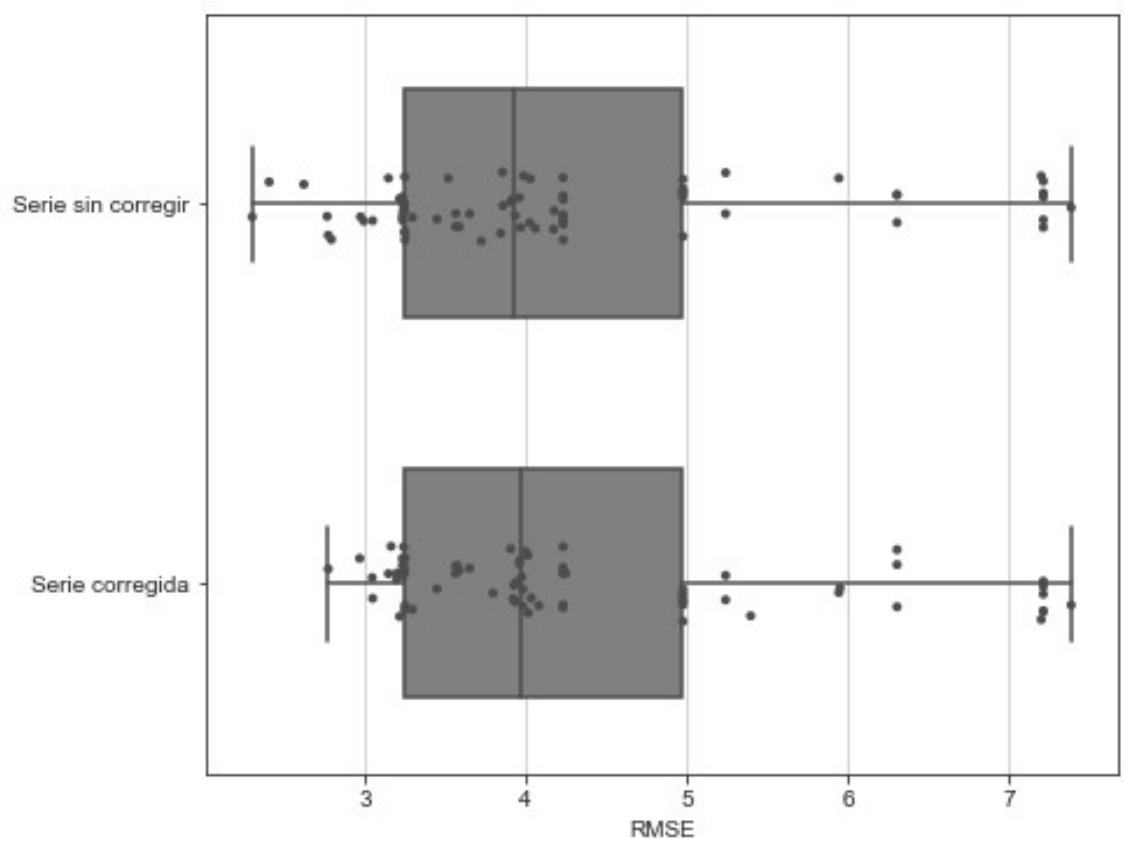

Ilustración 10. Comparación de los resultados finales de la predicción en función de si la serie ha sido corregida o no por valores anómalos.

Por lo tanto, a la luz de lo estudiado en este apartado, se puede concluir que para el problema abordado **no es necesaria la corrección de los valores anómalos de la serie para su sustitución por la mediana de los días similares.**

6.2 PREDICCIÓN

En el presente apartado se procede a comparar los resultados de predicción arrojados por los modelos estudiados con base en el error de predicción. Al igual que en el apartado anterior, en caso de que no se aprecien diferencias significativas entre dos modelos, se usará como criterio de selección el tiempo necesario para la realización de los cálculos.

En la Ilustración 11 se muestran los pronósticos de las combinaciones que han devuelto menores errores de predicción. A simple vista, se puede ver que las redes neuronales artificiales y el algoritmo *Gradient Boosting Regressor* arrojan resultados muy parecidos al consumo real, a diferencia de los árboles de decisión, cuya mejor predicción se distingue sensiblemente de los datos reales y no es capaz de captar el patrón de consumo.

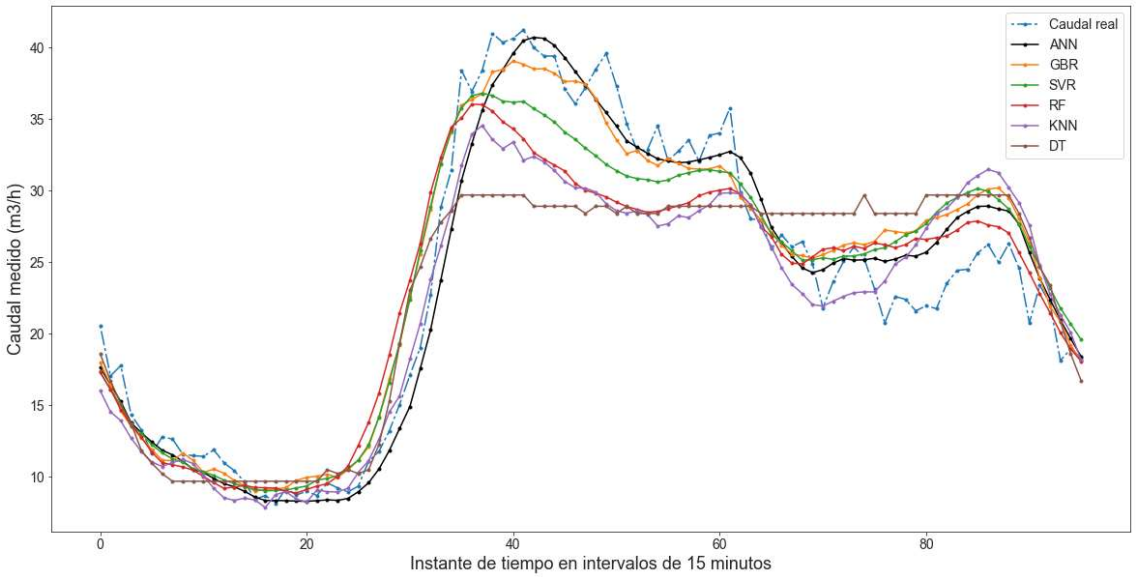

Ilustración 11. Mejores pronósticos realizados por cada uno de los modelos estudiados

Una vez calculadas todas las combinaciones de las estrategias definidas para la predicción del consumo de agua potable con un horizonte de 24 horas, se procede a comparar los resultados agrupados por cada fase de las que componen el trabajo. En la Tabla 6 se recogen los 5 mejores resultados para cada modelo predictivo. La relación completa de los resultados para las 144 combinaciones calculadas se encuentra en el Anexo III.

Modelo predictivo	Imputación	Corrección de valores anómalos	R2 (%)	SMAPE (%)	RMSE
ANN	median	No	99.20	8.03	2.302
	hybrid-ARIMA-median	No	99.12	8.18	2.407
	hybrid-ARIMA-mean	No	98.96	12.79	2.622
	hybrid-SVR-median	No	98.84	9.10	2.767
	hybrid-HW-mean	No	98.82	8.96	2.794
GBR	hybrid-HW-mean	No	98.84	9.39	2.774
	hybrid-HW-mean	Sí	98.84	9.39	2.774
	hybrid-HW-median	Sí	98.59	10.65	3.050
	hybrid-HW-median	No	98.59	10.65	3.050
	hybrid-ARIMA-mean	No	98.50	10.49	3.148
SVR	hybrid-RF-mean	No	98.42	10.88	3.234
	hybrid-RF-mean	Sí	98.42	10.88	3.234
	hybrid-SVR-mean	No	98.42	10.89	3.236
	hybrid-SVR-mean	Sí	98.42	10.89	3.236
	hybrid-KNN-mean	No	98.42	10.90	3.238
RF	hybrid-KNN-mean	No	97.29	13.28	4.234
	hybrid-SVR-mean	No	97.29	14.07	4.234
	median	No	97.37	13.22	4.176
	hybrid-ARIMA-median	Sí	97.48	13.20	4.081
	hybrid-ARIMA-median	No	97.51	13.05	4.060
DT	hybrid-KNN-mean	No	95.85	16.43	5.241
	hybrid-KNN-mean	Sí	95.85	16.43	5.241
	hybrid-ARIMA-mean	No	95.85	16.43	5.241
	hybrid-ARIMA-mean	Sí	95.85	16.43	5.241
	hybrid-HW-mean	No	94.66	19.97	5.944
KNN	hybrid-ARIMA-mean	No	97.30	13.12	4.232
	hybrid-ARIMA-mean	Sí	97.30	13.12	4.232
	hybrid-HW-mean	No	97.30	13.12	4.232
	hybrid-HW-mean	Sí	97.30	13.12	4.232
	hybrid-KNN-mean	No	97.30	13.12	4.232

Tabla 6. Resultados de las predicciones realizadas por las cinco mejores combinaciones para cada modelo predictivo

Como se puede observar en la Tabla 6, el modelo que devuelve el mejor resultado de predicción de acuerdo con las tres métricas utilizadas (R^2, SMAPE y RMSE) es el de redes neuronales artificiales, con imputación por medianas y sin corrección por valores anómalos. En la Ilustración 12 se muestra la predicción realizada por este modelo, incluyendo los límites inferior y superior para un intervalo de confianza del 95%.

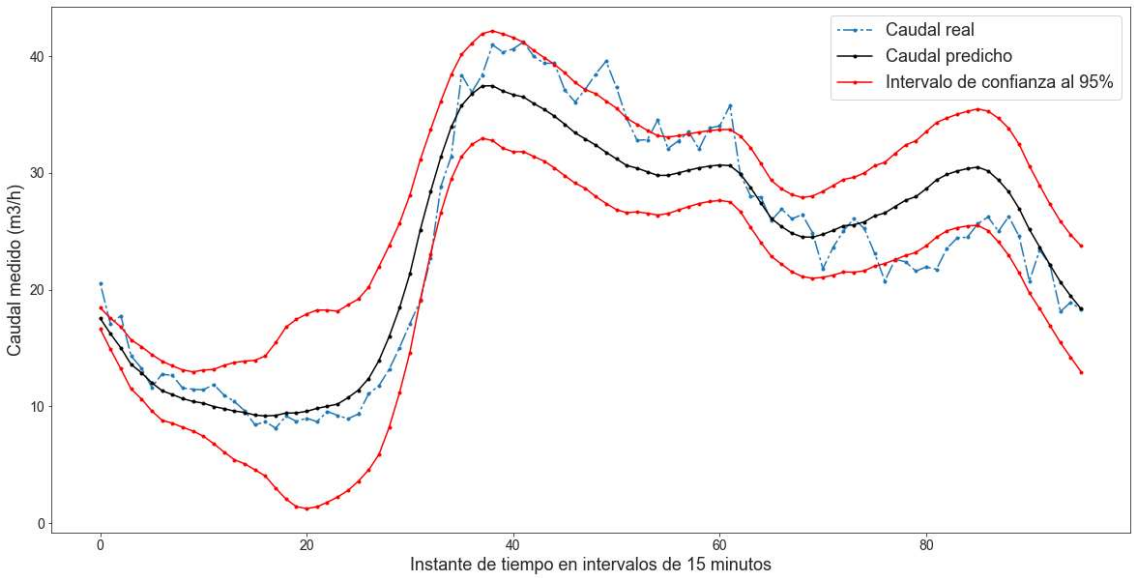

Ilustración 12. Resultado de la predicción con el modelo con la mayor precisión resultante (red neuronal artificial con imputación por medianas y sin corrección de valores anómalos)

La Ilustración 13 muestra la comparación gráfica de los modelos empleados para realizar las predicciones. Las redes neuronales artificiales devuelven los mejores resultados de predicción, pero presentan una gran variabilidad con respecto al resto de modelos.

Algo similar sucede con las pruebas realizadas con el algoritmo *Gradient Boosting Regressor*, cuyos resultados se encuentran mayoritariamente en un RMSE de entre 3 y 4, pero presenta algunas observaciones con un error de predicción muy alto. Estas observaciones se corresponden con el método híbrido de imputación que utiliza el algoritmo KNN para los periodos menores que una estacionalidad y las medianas para los periodos superiores.

También se puede observar que, a pesar de no ofrecer los menores resultados en cuanto al error de predicción, los modelos que mayor robustez muestran en el error resultante son las máquinas de soporte vectorial (SVR) y el algoritmo *Random Forest*.

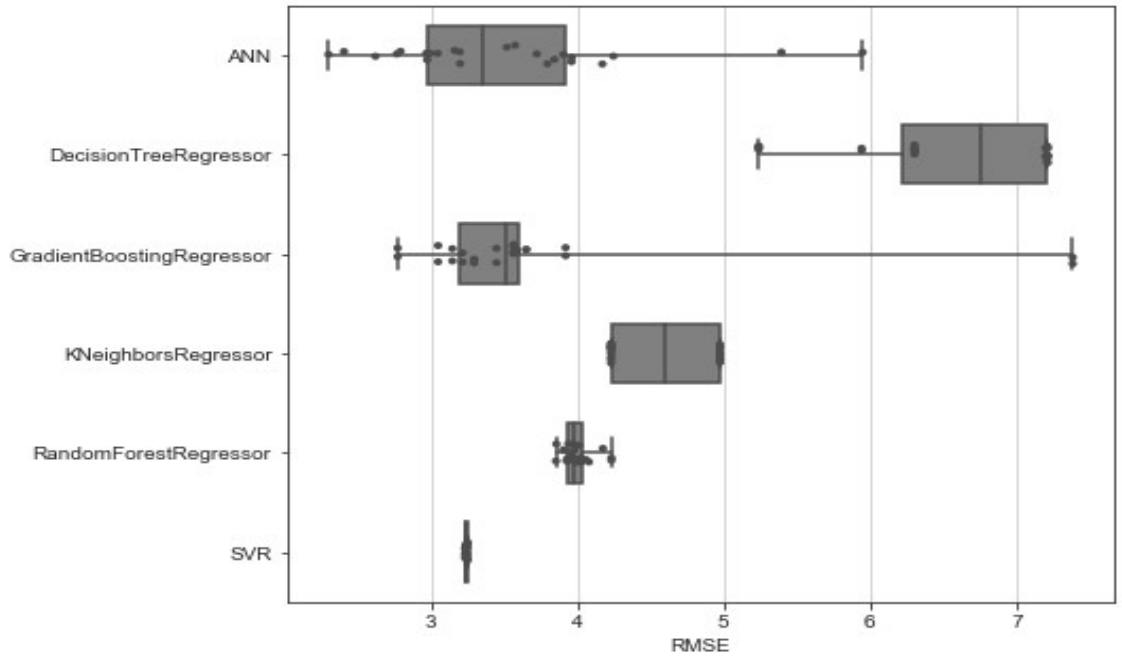

Ilustración 13. Comparación de los resultados finales de la predicción en función del método de predicción utilizado.

Dada esta gran variabilidad, únicamente se compararán los resultados obtenidos por las redes neuronales artificiales (ANN), las máquinas de soporte vectorial (SVR) y el algoritmo *Gradient Boosting Regressor* (GBR), puesto que son los modelos que mejores resultados presentan.

	ANN	GBR	SVR
ANN	1		
GBR	0.716	1	
SVR	**1.75E-6**	**0.012**	1

Tabla 7.. P-valores resultantes de la realización de la prueba de hipótesis de Levene para la comparación de varianzas según el modelo de predicción

	ANN	GBR	SVR
ANN	1		
GBR	0.680	1	
SVR	1	**0.048**	1

Tabla 8.. P-valores resultantes de la realización de la prueba de hipótesis de Kruskal-Wallis para la comparación de medianas según el modelo de predicción

En las tablas 7 y 8 se puede observar que, dada su robustez, el SVR muestra diferencias significativas con el resto de los modelos observados en cuanto a la

variabilidad de los errores de predicción, mientras que su mediana no presenta diferencias significativas con ninguno de los otros modelos.

Por lo tanto, se puede concluir que, el modelo que mejores pronósticos realiza, por regla general, es el basado en redes neuronales artificiales. Sin embargo, debido a la robustez de los resultados y al bajo valor de los errores de predicción, **las máquinas de soporte vectorial son el mejor modelo de predicción para abordar el problema estudiado.**

7 CONCLUSIONES

El presente trabajo recoge una metodología que aborda el tratamiento de una serie temporal del caudal medido en un sector hidráulico del municipio de Benetússer (Valencia) cada 15 minutos.

La metodología elaborada estudia diferentes sistemas para la reconstrucción de la serie temporal, incluyendo la imputación de valores faltantes y la sustitución de valores anómalos, y finalmente realiza la predicción de las lecturas de caudal en las 24 horas posteriores al último dato disponible en la serie.

El objetivo del trabajo, por lo tanto, consiste en aplicar la metodología elaborada para determinar cuál es la mejor estrategia a seguir para reconstruir una serie temporal y predecir la demanda a corto plazo, generando así una herramienta de ayuda a la toma de decisiones que permita tratar los problemas de análisis y pronóstico de series temporales de consumo de agua potable de la manera más precisa posible.

En relación con la reconstrucción de la serie temporal estudiada, se ha observado que los modelos que llevan a cabo la imputación de datos faltantes en espacios de tiempo menores que la estacionalidad mediante técnicas clásicas de predicción de series temporales o a través de algoritmos de *machine learning* no devuelven resultados significativamente diferentes de los conseguidos mediante el uso de técnicas más sencillas como la media o la mediana del mismo instante en semanas anteriores.

Por otro lado, la corrección de los valores anómalos mediante la mediana de los días similares, técnica que se ha probado para tratar de facilitar los pronósticos posteriores bajo la premisa de que con una serie temporal suavizada aumenta la posibilidad de obtener predicciones más precisas, no ha mostrado ninguna mejora en los resultados de predicción, si bien supone un mayor coste computacional.

En cuanto a los resultados de los modelos de predicción, se observan importantes diferencias en los resultados obtenidos mediante los algoritmos puestos a prueba. Los mejores pronósticos se han obtenido mediante un modelo

de imputación por medianas, sin corrección de valores anómalos y con el uso de redes neuronales artificiales.

Si bien las redes neuronales artificiales devuelven resultados muy satisfactorios, es de notar la variabilidad de los errores obtenidos en las pruebas en las que se han utilizado otras técnicas de reconstrucción. En este sentido, cabe destacar la robustez de las máquinas de soporte vectorial y del algoritmo *Random Forest*, que no solo permiten obtener errores de predicción muy bajos, sino que son insensibles a los métodos de imputación y corrección utilizados en fases anteriores del cálculo.

Este último hecho resulta de especial interés, pues la realización de pronósticos con modelos que no se vean significativamente afectados por el tratamiento previo que hayan tenido los datos tiene una gran relevancia de cara a extrapolar los resultados obtenidos en el presente trabajo a series de tiempo similares.

Por lo tanto, a la luz de los resultados obtenidos en esta investigación, y dada la precisión de las predicciones realizadas, se considera alcanzado el objetivo de elaborar una metodología que sirva de base para un sistema de apoyo a la toma de decisiones, especialmente orientado a empresas operadoras de servicios de distribución de agua potable.

Las conclusiones planteadas en el presente trabajo deben ser consideradas con cierta cautela, pues existen limitaciones computacionales asociadas al equipo con el que se ha llevado a cabo, así como restricciones temporales y de alcance de la investigación dada la naturaleza del trabajo.

Esto lleva a plantear que futuras investigaciones relacionadas con este ámbito deben abordar temas que han quedado fuera del alcance de este trabajo. Algunos aspectos que cabe considerar son el rango temporal de datos disponibles que es necesario para efectuar predicciones precisas, la posibilidad de utilizar otros modelos de *machine learning* o de *deep learning* que no hayan sido utilizados en este proyecto, la evaluación de la necesidad de utilizar algoritmos de optimización de hiperparámetros, y, en caso de estimarlo

conveniente, plantear el uso de diferentes algoritmos metaheurísticos al usado en este trabajo.

La aplicabilidad de esta metodología debe ser comprobada y contrastada con el fin de comprobar su validez, así como determinar si las conclusiones aquí obtenidas se mantienen para otras series temporales de consumo de agua potable que presenten otra tipología (sectores industriales, sectores domiciliarios en zonas cuya población se ve afectada por la estacionalidad, etc.), o incluso para otro tipo de series de tiempo.

REFERENCIAS

[1] R. Liemberger y A. Wyatt, «Quantifying the global non-revenue water problem,» *Water Supply,* pp. 831-837, 2019.

[2] C. Palau, F. J. Arregui y A. Ferrer, «Using multivariate principal component analysis of injected water flows to detect anomalous behaviors in a water supply system-a case study,» *Water Science and Technology: Water Supply,* vol. 4, nº 3, pp. 169-182, 2004.

[3] S. J. Lee, G. Lee, J. C. Suh y J. M. Lee, «Online Burst Detection and Location of Water Distribution Systems and Its Practical Applications,» *Journal of Water Resources Planning and Management,* vol. 142, nº 1, p. 04015033, 2016.

[4] D. Jung y K. Lansey, «Water Distribution System Burst Detection Using a Nonlinear Kalman Filter,» *Journal of Water Resources Planning and Management,* vol. 141, nº 5, p. 04014070, 2015.

[5] D. Jung, D. Kang, J. Liu y K. Lansey, «Improving the rapidity of responses to pipe burst in water distribution systems: a comparison of statistical process control methods,» *Journal of Hydroinformatics,* pp. 307-328, 2015.

[6] K. Nam, P. Ifaei, S. Heo, G. Rhee, S. Lee y C. Yoo, «An Efficient Burst Detection and Isolation Monitoring System for Water Distribution Networks Using Multivariate Statistical Techniques,» *Sustainability ,* 2019.

[7] G. Cembrano, G. Wells, J. Quevedo, R. Pérez y R. Argelaguet, «Optimal control of a water distribution network in a supervisory control system,» *Control Engineering Practice,* vol. 8, nº 10, pp. 1177-1188, 2000.

[8] E. Salomons, A. Goryashko, U. Shamir, Z. Rao y S. Alvisi, «Optimizing the operation of the Haifa-A water-distribution network,» *Journal of Hydroinformatics,* vol. 9, nº 1, pp. 51-64, 2007.

[9] H.-S. Kang, H. Kim, J. Lee, I. Lee, B.-Y. Kwak y H. Im, «Optimization of pumping schedule based on water demand forecasting using a combined model of autoregressive integrated moving average and exponential smoothing,» *Water Supply,* vol. 15, nº 1, pp. 188-195, 2015.

[10] D. Elixmann, J. Busch y W. Marquardt, «Integration of model-predictive scheduling, dynamic real-time optimization and output tracking for a wastewater treatment process,» *IFAC Proceedings Volumes,* vol. 43, nº 6, pp. 90-95, 2010.

[11] M. Simon-Várhelyi, V. M. Cristea y A. V. Luca, «Reducing energy costs of the wastewater treatment plant by improved scheduling of the periodic influent load,» *Journal of Environmental Management,* vol. 262, p. 110294, 2020.

[12] A. Antunes, A. Andrade-Campos, A. Sardinha-Lourenço y M. S. Oliveira, «Short-term water demand forecasting using machine learning techniques,» *Journal of Hydroinformatics,* vol. 20, nº 6, pp. 1343-1366, 2018.

[13] H. Rodríguez Rangel, V. Puig, R. López Farias y J. J. Flores, «Short-term demand forecast using a bank of neural network models trained using genetic algorithms for the optimal management of drinking water networks,» *Journal of Hydroinformatics,* vol. 19.1, pp. 1-16, 2017.

[14] S. Mouatadid y J. Adamowski, «Using extreme learning machines for short-term urban water demand forecasting,» *Urban Water Journal,* vol. 14, nº 6, pp. 630-638, 2017.

[15] I. Duerr, H. R. Merrill, C. Wang, R. Bai, M. Boyer, M. D. Dukes y N. Bliznyuk, «Forecasting urban household water demand with statistical and machine learning methods using large space-time data: A Comparative study,» *Environmental Modelling & Software,* vol. 102, pp. 29-38, 2018.

[16] M. S. Osman, A. M. Abu-Mahfouz y P. R. Page, «A Survey on Data Imputation Techniques: Water Distribution System as a Use Case,» *IEEE Access,* vol. 6, pp. 63279-63291, 2018.

[17] A. N. Baraldi y C. K. Enders, «An introduction to modern missing data analyses,» *Journal of School Psychology,* vol. 48, pp. 5-37, 2010.

[18] D. Loureiro, C. Amado, A. Martins, D. Vitorino, A. Mamade y S. Teixeira Coelho, «Water distribution systems flow monitoring and anomalous event detection: A practical approach,» *Urban Water Journal,* vol. 13, nº 3, pp. 242-22, 2015.

[19] P. Rousseeuw y C. Croux, «Alternatives to the median absolute deviation,» *Journal of the American Statistical Association,* vol. 88, nº 424, pp. 1273-1283, 1993.

[20] J. Quevedo, V. Puig, G. Cembrano, J. Blanch, J. Aguilar, D. Saporta, G. Benito, M. Hedo y A. Molina, «Validation and reconstruction of flow meter data in the Barcelona water distribution network,» *Control Engineering Practice,* nº 18, pp. 640-651, 2010.

[21] R. Barrela, C. Amado, D. Loureiro y A. Mamade, «Data reconstructiion of flow time series in water distribution systems - a new method that accomodatess multiple seasonality,» *Journal of Hydroinformatics,* 2017.

[22] S. Bennis, F. Berrada y N. Kang, «Improving single-variable and multivariable techniqes for estimating missing hydrological data,» *Journal of Hydrology,* nº 191, pp. 87-105, 1997.

[23] N. Golyandina y E. Osipov, «The "Caterpillar"-SSA method for analysis of time series with missing values,» *Journal of Statistical Planning and Inference,* vol. 137, pp. 2642-2653, 2007.

[24] S. L. Zubaidi, S. K. Gharghan, J. Dooley, R. M. Alkhaddar y M. Abdellatif, «Short-Term Urban Water Demand Prediction Considering Weather Factors,» *Water Resources Management,* vol. 32, pp. 4527-4542, 2018.

[25] G. Guancheng, L. Shuming, W. Yipeng, L. Junyu, Z. Ren y Z. Xiaoyun, «Short-Term Water Demand Forecast Based on Deep Learning Method,» *Journal of Water Resources Planning and Management,* vol. 12, nº 144, p. 04018076, 2018.

[26] D. Lee y S. Derrible, «Predicting Residential Water Demand with Machine-Based Statistical Learning,» *Journal of Water Resources Planning and Management,* vol. 146, nº 1, p. 04019067, 2020.

[27] M. Bata, R. Carriveau y D. . S.-K. Ting, «Short-term water demand forecasting using hybrid supervised and unsupervised machine learning model,» *Smart Water,* vol. 5, pp. 1-18, 2020.

[28] T. Kohonen, «The self-organizing map,» *Proceedings of the IEEE,* vol. 78, nº 9, pp. 1464-1480, 1990.

[29] M. Bata, R. Carriveau y D. S.-K. Ting, «Short-Term Water Demand Forecasting Using Nonlinear Autoregressive Artficial Neural Networks,» *Journal of Water Resources Planning and Management,* vol. 146, nº 3, p. 04020008, 2020.

[30] M. Li, Z. Feifei, T. Ruoling y Z. Qingzhou, «Hourly and Daily Urban Water Demand Predictions Using a Long Short-Term Memory Based Model,» *Journal of Water Resources Planning and Management,* vol. 146, nº 9, p. 05020017, 2020.

[31] J. Bergstra y Y. Bengio, «Random Search for Hyper-Parameter Optimization,» *Journal of Machine Learning Research,* vol. 13, pp. 281-305, 2012.

[32] K. Sorensen y F. Glover, «Metaheuristics,» de *Encyclopedia of Operations Research and Management Science*, Springer US, 2013, pp. 960-970.

[33] L. Shih-Wei, Y. Kuo-Ching, C. Shih-Chieh y L. Zne-Jung, «Particle swarm optimization for parameter determination and featur selection of support vector machines,» *Expert Systems with Applications,* vol. 35, pp. 1817-1824, 2008.

[34] I. Aljarah, A. M. Al-Zoubi, H. Faris, M. A. Hassonah, S. Mirjalili y H. Saadeh, «Simultaneous feature selection and Support Vector Machine optimization using the Grasshopper Optimization Algorithm,» *Cognitive Computation,* vol. 10, pp. 478-495, 2018.

[35] H. Ji-Hoon, C. Dong-Jin, P. Sang-Uk y H. Sun-Ki, «Hyperparameter Optimization Using a Genetic Algorithm Considering Verification Time in a Convolutional Neural Network,» *Journal of Electrical Engineering & Technology,* vol. 15, pp. 721-726, 2020.

[36] M. Claesen, J. Simm, D. Popovic y B. D. Moor, «Hyperparameter tuning in Python using Optunity,» *Proceedings of the International Workshop on Technical Computing for Machine Learning and Mathematical Engineering,* vol. 1, p. 3, 2014.

[37] J. Bergstra, D. Yamins y D. D. Cox, «Hyperopt: A Python library for optimizing the hyperparameters of machine learning algorithms,» de *Proceedings of the 12th Python in science conference*, Austin, TX, 2013.

[38] L. Hertel, J. Collado, P. Sadowski y P. Baldi, «Sherpa: hyperparameter optimization for machine learning models.».

[39] T. Akiba, S. Sano, T. O. T. Yanase y M. Koyama, «Optuna: A next-generation hyperparameter optimization framework,» de *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, Anchorage, AK, 2019.

[40] G. Van Rossum y F. L. Drake Jr, Python reference manual, Amsterdam: Centrum voor Wiskunde en Informatica, 1995.

[41] J. R. Quinlan, «Decision Trees and Multi-Valued Attributes,» *Machine Intelligence,* 1985.

[42] L. Breiman, J. Friedman, R. Olshen y C. Stone, Classification and Regression Trees, Belmont, CA: Wadsworth International Group, 1984.

[43] L. Breiman, «Random forests,» *Machine Learning,* vol. 45, pp. 5-32, 2001.

[44] L. Breiman, «Bagging Predictors,» *Machine Learning,* vol. 24, pp. 123-140, 1996.

[45] G. Valentini y F. Masulli, «Ensembles of learning machines,» *Italian workshop on neural nets,* pp. 3-20, 2002.

[46] P. Bühlmann, «Bagging, boosting and ensemble methods,» *Handbook of Computational Statistics,* pp. 985-1022, 2012.

[47] K. R. Awad M., «Support Vector Regression,» de *Efficient Learning Machines*, Berkeley, CA, Apress, 2015.

[48] J. H. Friedman, «Greedy function approximation: a gradient boosting machine,» *Annals of statistics,* pp. 1189-1232, 2001.

[49] J. Friedman, T. Hastie y R. Tibshirani, The elements of statistical learning, 2 ed., vol. 1, New York: Springer series in statistics, 2008.

[50] A. Natekin y K. Alois, «Gradien boosting machines, a tutorial,» *Frontiers in Neurorobotics,* pp. 7-21, 2013.

[51] A. K. Jain, J. Mao y K. M. Mohiuddin, «Jain, A. K., Mao, J., & Mohiuddin, K. M. (1996). Artificial neural networks: A tutorial,» *Computer,* vol. 29, nº 3, pp. 31-44, 1996.

[52] W. Sun-Chong, Interdisciplinary Computing in Java Programming, New Yotk: Springer, 2003.

[53] B. Karlik y A. V. Olgac, «Performance Analysis of Various Activation Functions in Generalized MLP Architectures of Neural Networks,» *International Journal of Artificial Intelligence and Expert Systems,* vol. 1, nº 4, pp. 111-122, 2011.

[54] V. Nair y G. E. Hinton, «Rectified linear units improve restricted boltzmann machines,» *ICML-10,* 2010.

[55] X. Glorot, A. Bordes y Y. Bengio, «Deep sparse rectifier neural networks,» de *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, Fort Lauderdale, FL, USA, 2011.

[56] H. Ide y T. Kurita, «Improvement of Learning for CNN with ReLU Activation by Sparse Regularization,» de *2017 International Joint Conference on Neural Networks*, Anchorage, AK, USA, 2017.

[57] M. Mitchell, «Genetic algorithms: An overview,» *Complexity,* pp. 31-39, 1995.

[58] C. Di Francescomarino, M. Dumas, M. Federici, C. Ghidini, F. M. Maggi, W. Rizzi y L. Simonetto, «Genetic algorithms for hyperparameter optimization in predictive business process monitoring,» *Information Systems,* pp. 67-83, 2018.

[59] D. Whitley, «A genetic algorithm tutorial,» *Statistics and computing,* vol. 4, nº 2, pp. 65-85, 1994.

[60] R. J. Hyndman y A. B. Koehler, «Another look at measures of forecast accuracy,» *International Journal of Forecasting,* pp. 679-688, 2006.

[61] H. Levene, «Robust tests for equality of variances,» de *Contributions to Probability and Statistics*, I. Olkin, Ed., Palo Alto, California, Stanford University Press, 1960, pp. 278-292.

[62] W. H. Kruskal y W. A. & Wallis, «Use of ranks in one-criterion variance analysis,» *Journal of the American statistical Association,* vol. 47, nº 260, pp. 583-621, 1952.

[63] C. Harris, K. Millman y S. van der Walt, «Array programming with NumPy.,» *Nature,* vol. 585, nº 7825, p. 357–362, 2020.

[64] W. McKinney, «Data Structures for Statistical Computing in Python,» de *Proceedings of the 9th Python in Science Conference*, J. M. Stéfan van der Walt, Ed., Austin, Texas, 2010, pp. 56-61.

[65] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot y E. Duchesnay, «Scikit-learn: Machine Learning in Python,» *Journal of Machine Learning Research,* vol. 12, pp. 2825-2830, 2011.

[66] F. Chollet, «Keras,» 2015. [En línea]. Available: https://github.com/fchollet/keras.

[67] S. Seabold y J. Perktold, «statsmodels: Econometric and statistical modeling with python,» de *9th Python in Science Conference*, Austin, Texas, 2010.

[68] T. G. Smith, «Pmdarima: ARIMA estimators for Python,» 2017. [En línea]. Available: http://www.alkaline-ml.com/pmdarima. [Último acceso: 16 11 2020].

[69] P. Virtanen, R. Gommers, T. Oliphant, M. Haberland, T. Reddy, D. Cournapeau, E. Burovski, P. Peterson, W. Weckesser y J. Bright, «Scipy

1.0: Fundamental Algorithms for Scientific Computing in Python,» *Nature Methods,* vol. 17, pp. 261-272, 2020.

ANEXO I. CARACTERÍSTICAS DEL PAQUETE DE CÁLCULO

Objeto	Métodos	Ámbito	Descripción
waTS			En este objeto se desarrollan todos los métodos necesarios para el desarrollo de los cálculos
	data_wrangler	Reconstrucción	Toma la serie temporal original y realiza las tareas de limpieza y ordenación de los datos. El método elimina valores duplicados, introduce datos faltantes en las partes de la serie en la que hubiese saltos temporales y almacena aquellos instantes en los que existan dichos datos faltantes.
	recon_median:	Datos faltantes	Método para la imputación de valores faltantes mediante el uso de la mediana del mismo instante de tiempo durante las *w* semanas anteriores
	recon_mean	Datos faltantes	Ídem de lo anterior, utilizando la media en lugar de la mediana
	recon_hybrid	Datos faltantes	Método de imputación que considera la media o la mediana de las *w* semanas anteriores cuando el bloque de datos faltantes es mayor que la menor estacionalidad, y que utiliza modelos de *machine learning* o de series temporales cuando dichos bloques son menores que la estacionalidad
	shifting	Valores anómalos/Predicción	Se trata del método que realiza los decalajes de la serie temporal en *t* instantes de tiempo. Se utiliza en partes posteriores de los cálculos para realizar tanto la reconstrucción como las predicciones.

	matrix	Valores anómalos	Este método transforma los datos de la serie temporal en la que las filas se corresponden con los días y las columnas con los consumos en cada instante a lo largo del día. Se utiliza para la obtención de clústeres en fases posteriores del cálculo.
	Qn	Valores anómalos	Este método toma la matriz obtenida mediante el método anterior y calcula el estimador Qn [19].
	clusters	Valores anómalos	Toma la matriz resultante del método *Matrix* y obtiene clústeres en función del tipo de día (laborable/fin de semana) y de la época del año.
	outlier_region	Valores anómalos	A partir de la matriz anterior, los clústeres y el estadístico Qn, se obtiene la región de anómalos según los criterios de Loureiro *et al.* [18].
	corrected	Valores anómalos	Este método corrige la serie temporal, sustituyendo aquellos valores que quedan en la región de anómalos por la mediana correspondiente al clúster al que pertenece el día estudiado.
	forecast_ML	Predicción	Se trata del método utilizado para la predicción de la serie a partir de modelos de *machine learning*.
	forecast_ANN	Predicción	Método análogo al anterior, pero en este caso con redes neuronales artificiales
Pipeline			Este objeto hereda las propiedades del anterior con el objetivo de tener una herramienta más sencilla para probar diferentes estrategias

	wrangle	Reconstrucción	Reconstruye la serie mediante el método Data_wrangler del objeto anterior
	recon	Datos faltantes	Imputa los datos faltantes de la serie de acuerdo con los parámetros introducidos por el usuario
	outliers	Valores anómalos	Permite el reemplazo de los valores anómalos con la mediana del clúster al que pertenece el día estudiado
	predict	Predicción	Predice la serie temporal con el modelo, la estacionalidad y el horizonte de predicción elegidos por el usuario

ANEXO II. PARÁMETROS OPTIMIZADOS Y VALORES RESULTANTES

Modelo	Hiperparámetros	Tipo	Rango	Valor óptimo
Random Forest	n_estimators	Numérico	1-100	380
	max_depth	Numérico	1-100	62
	min_samples_split	Numérico	2-20	13
	min_samples_leaf	Numérico	1-20	9
K Nearest Neighbors	n_neighbors	Numérico	1-50	22
	weights	Categórico	uniform, distance	Uniform
	algorithm	Categórico	auto, ball_tree, kd_tree, brute	Auto
	leaf-size	Numérico	1-100	73.76
	p	Categórico	1-2	1.09
Support Vector Regression	kernel	Categórico	linear, ply, rbf, sigmoid, precomputed	rbf
	degree	Numérico	1-10	-
	gamma	Categórico	scale, auto	scale
	coef0	Numérico	0-10	-
	shrinking	Booleano	Verdadero, Falso	Verdadero

Decision Tree	criterion	Categórico	mse, friedman_mse, mae	mse
	splitter	Categórico	best, random	best
	max_depth	Numérico	1-500	40
	min_samples_split	Numérico	1-50	34
	min_samples_leaf	Numérico	1-50	0.014596969
Gradient Boosting Regression	loss	Numérico	1-4	ls
	learning_rate	Numérico	0-1	0.3
	n_estimators	Numérico	1-500	58
	criterion	Categórico	mse, friedman_mse, mae	mse
	min_samples_split	Numérico	2-50	16
	min_samples_leaf	Numérico	1-50	0.0187
	max_depth	Numérico	1-500	32
ANN	nodes	Numérico	1-250	100
	epochs	Numérico	50-500	100
	layers	Numérico	1-10	5
	kernel_initializer	Categórico	normal, random_normal	normal
	activation	Categórico	relu	relu
	optimizer	Categórico	adam	adam

ANEXO III. RELACIÓN DE RESULTADOS OBTENIDOS

Imputación de valores faltantes			Corrección de valores anómalos	Modelo predictivo	Métricas de validación		
Método	Periodos menores que la estacionalidad	Periodos mayores que la estacionalidad			R2 (%)	SMAPE (%)	RMSE
median				ANN	99.20	8.03	2.302
hybrid	ARIMA	median		ANN	99.12	8.18	2.407
hybrid	ARIMA	mean		ANN	98.96	12.79	2.622
hybrid	SVR	median		ANN	98.84	9.10	2.767
hybrid	HW	mean		GradientBoostingRegressor	98.84	9.39	2.774
hybrid	HW	mean	X	GradientBoostingRegressor	98.84	9.39	2.774
hybrid	HW	mean		ANN	98.82	8.96	2.794
hybrid	RF	mean	X	ANN	98.67	9.54	2.969
hybrid	RF	median		ANN	98.66	11.45	2.975
mean				ANN	98.65	10.88	2.995
median			X	ANN	98.60	9.81	3.047
hybrid	HW	median	X	GradientBoostingRegressor	98.59	10.65	3.050
hybrid	HW	median		GradientBoostingRegressor	98.59	10.65	3.050
hybrid	ARIMA	mean		GradientBoostingRegressor	98.50	10.49	3.148
hybrid	ARIMA	mean	X	GradientBoostingRegressor	98.50	10.49	3.148
hybrid	HW	median	X	ANN	98.49	11.62	3.164
hybrid	RF	median	X	ANN	98.45	11.72	3.198
hybrid	KNN	mean	X	ANN	98.45	10.43	3.200
mean				GradientBoostingRegressor	98.44	10.39	3.217
mean			X	GradientBoostingRegressor	98.44	10.39	3.217
hybrid	RF	mean		SVR	98.42	10.88	3.234

hybrid	RF	mean	X	SVR	98.42	10.88	3.234
hybrid	SVR	mean		SVR	98.42	10.89	3.236
hybrid	SVR	mean	X	SVR	98.42	10.89	3.236
hybrid	KNN	mean		SVR	98.42	10.90	3.238
hybrid	KNN	mean	X	SVR	98.42	10.90	3.238
hybrid	HW	mean		SVR	98.42	10.87	3.238
hybrid	HW	mean	X	SVR	98.42	10.87	3.238
hybrid	ARIMA	mean		SVR	98.42	10.87	3.238
hybrid	ARIMA	mean	X	SVR	98.42	10.87	3.238
mean				SVR	98.41	10.90	3.241
mean			X	SVR	98.41	10.90	3.241
hybrid	ARIMA	median		SVR	98.41	10.87	3.246
hybrid	ARIMA	median	X	SVR	98.41	10.87	3.246
hybrid	HW	median		SVR	98.41	10.88	3.248
hybrid	HW	median	X	SVR	98.41	10.88	3.248
hybrid	SVR	median		SVR	98.41	10.90	3.248
hybrid	SVR	median	X	SVR	98.41	10.90	3.248
hybrid	RF	median		SVR	98.41	10.90	3.249
hybrid	RF	median	X	SVR	98.41	10.90	3.249
hybrid	KNN	median		SVR	98.40	10.91	3.250
hybrid	KNN	median	X	SVR	98.40	10.91	3.250
median				SVR	98.40	10.91	3.250
median			X	SVR	98.40	10.91	3.250
hybrid	SVR	median		GradientBoostingRegressor	98.36	11.88	3.297
hybrid	SVR	median	X	GradientBoostingRegressor	98.36	11.88	3.297
hybrid	KNN	mean		GradientBoostingRegressor	98.20	12.15	3.449
hybrid	KNN	mean	X	GradientBoostingRegressor	98.20	12.15	3.449

hybrid	SVR	mean		ANN	98.13	12.08	3.519
median				GradientBoostingRegressor	98.08	12.38	3.564
median			X	GradientBoostingRegressor	98.08	12.38	3.564
hybrid	RF	mean	X	GradientBoostingRegressor	98.08	12.08	3.566
hybrid	RF	mean		GradientBoostingRegressor	98.08	12.08	3.566
hybrid	ARIMA	mean	X	ANN	98.07	11.99	3.576
hybrid	ARIMA	median		GradientBoostingRegressor	98.06	12.10	3.588
hybrid	ARIMA	median	X	GradientBoostingRegressor	98.06	12.10	3.588
hybrid	SVR	mean		GradientBoostingRegressor	97.99	12.33	3.652
hybrid	SVR	mean	X	GradientBoostingRegressor	97.99	12.33	3.652
hybrid	RF	mean		ANN	97.91	10.69	3.724
hybrid	SVR	median	X	ANN	97.82	13.23	3.797
hybrid	KNN	mean		ANN	97.77	17.24	3.844
hybrid	KNN	median		RandomForestRegressor	97.75	12.56	3.856
hybrid	SVR	median		RandomForestRegressor	97.75	12.65	3.859
hybrid	KNN	median		ANN	97.70	14.07	3.906
median			X	RandomForestRegressor	97.69	12.57	3.907
hybrid	RF	median		GradientBoostingRegressor	97.68	13.04	3.921
hybrid	RF	median	X	GradientBoostingRegressor	97.68	13.04	3.921
hybrid	KNN	median	X	RandomForestRegressor	97.67	12.68	3.927
hybrid	KNN	mean	X	RandomForestRegressor	97.66	12.50	3.934
hybrid	RF	median		RandomForestRegressor	97.66	12.70	3.935
hybrid	SVR	median	X	RandomForestRegressor	97.65	12.71	3.940
hybrid	HW	median		RandomForestRegressor	97.63	12.82	3.960
hybrid	ARIMA	median	X	ANN	97.63	10.47	3.961
hybrid	KNN	median	X	ANN	97.63	16.56	3.963
mean				RandomForestRegressor	97.62	12.72	3.968

hybrid	HW	median	X	RandomForestRegressor	97.61	12.92	3.975
mean			X	RandomForestRegressor	97.61	12.81	3.980
hybrid	RF	median	X	RandomForestRegressor	97.61	12.84	3.981
hybrid	ARIMA	mean		RandomForestRegressor	97.60	12.76	3.986
hybrid	RF	mean	X	RandomForestRegressor	97.59	12.88	3.997
hybrid	ARIMA	mean	X	RandomForestRegressor	97.57	12.80	4.014
hybrid	SVR	mean	X	RandomForestRegressor	97.56	12.79	4.015
hybrid	RF	mean		RandomForestRegressor	97.55	12.85	4.025
hybrid	HW	mean		RandomForestRegressor	97.55	13.00	4.028
hybrid	HW	mean	X	RandomForestRegressor	97.54	13.09	4.034
hybrid	ARIMA	median		RandomForestRegressor	97.51	13.05	4.060
hybrid	ARIMA	median	X	RandomForestRegressor	97.48	13.20	4.081
hybrid	HW	median		ANN	97.37	15.38	4.173
median				RandomForestRegressor	97.37	13.22	4.176
hybrid	ARIMA	mean		KNeighborsRegressor	97.30	13.12	4.232
hybrid	ARIMA	mean	X	KNeighborsRegressor	97.30	13.12	4.232
hybrid	HW	mean		KNeighborsRegressor	97.30	13.12	4.232
hybrid	HW	mean	X	KNeighborsRegressor	97.30	13.12	4.232
hybrid	KNN	mean		KNeighborsRegressor	97.30	13.12	4.232
hybrid	KNN	mean	X	KNeighborsRegressor	97.30	13.12	4.232
hybrid	RF	mean		KNeighborsRegressor	97.30	13.12	4.232
hybrid	RF	mean	X	KNeighborsRegressor	97.30	13.12	4.232
hybrid	SVR	mean		KNeighborsRegressor	97.30	13.12	4.232
hybrid	SVR	mean	X	KNeighborsRegressor	97.30	13.12	4.232
mean				KNeighborsRegressor	97.30	13.12	4.232
mean			X	KNeighborsRegressor	97.30	13.12	4.232
hybrid	SVR	mean		RandomForestRegressor	97.29	14.07	4.234

hybrid	KNN	mean		RandomForestRegressor	97.29	13.28	4.234
hybrid	HW	mean	X	ANN	97.27	13.84	4.248
hybrid	ARIMA	median		KNeighborsRegressor	96.26	15.36	4.976
hybrid	ARIMA	median	X	KNeighborsRegressor	96.26	15.36	4.976
hybrid	HW	median		KNeighborsRegressor	96.26	15.36	4.976
hybrid	HW	median	X	KNeighborsRegressor	96.26	15.36	4.976
hybrid	KNN	median		KNeighborsRegressor	96.26	15.36	4.976
hybrid	KNN	median	X	KNeighborsRegressor	96.26	15.36	4.976
hybrid	RF	median		KNeighborsRegressor	96.26	15.36	4.976
hybrid	RF	median	X	KNeighborsRegressor	96.26	15.36	4.976
hybrid	SVR	median		KNeighborsRegressor	96.26	15.36	4.976
hybrid	SVR	median	X	KNeighborsRegressor	96.26	15.36	4.976
median				KNeighborsRegressor	96.26	15.36	4.976
median			X	KNeighborsRegressor	96.26	15.36	4.976
hybrid	KNN	mean		DecisionTreeRegressor	95.85	16.43	5.241
hybrid	KNN	mean	X	DecisionTreeRegressor	95.85	16.43	5.241
hybrid	ARIMA	mean		DecisionTreeRegressor	95.85	16.43	5.241
hybrid	ARIMA	mean	X	DecisionTreeRegressor	95.85	16.43	5.241
mean			X	ANN	95.60	19.60	5.396
hybrid	HW	mean		DecisionTreeRegressor	94.66	19.97	5.944
hybrid	HW	mean	X	DecisionTreeRegressor	94.66	19.97	5.944
hybrid	SVR	mean	X	ANN	94.65	22.16	5.950
hybrid	RF	mean	X	DecisionTreeRegressor	94.00	22.85	6.305
hybrid	RF	mean		DecisionTreeRegressor	94.00	22.85	6.305
mean				DecisionTreeRegressor	94.00	22.85	6.305
mean			X	DecisionTreeRegressor	94.00	22.85	6.305
hybrid	SVR	mean		DecisionTreeRegressor	94.00	22.85	6.305

hybrid	SVR	mean	X	DecisionTreeRegressor	94.00	22.85	6.305
hybrid	HW	median		DecisionTreeRegressor	92.17	29.90	7.200
hybrid	HW	median	X	DecisionTreeRegressor	92.17	29.90	7.200
hybrid	ARIMA	median		DecisionTreeRegressor	92.14	30.33	7.213
hybrid	KNN	median		DecisionTreeRegressor	92.14	30.33	7.213
hybrid	RF	median	X	DecisionTreeRegressor	92.14	30.33	7.213
hybrid	SVR	median		DecisionTreeRegressor	92.14	30.33	7.213
hybrid	SVR	median	X	DecisionTreeRegressor	92.14	30.33	7.213
median				DecisionTreeRegressor	92.14	30.33	7.213
hybrid	ARIMA	median	X	DecisionTreeRegressor	92.14	30.33	7.213
hybrid	RF	median		DecisionTreeRegressor	92.14	30.33	7.213
median			X	DecisionTreeRegressor	92.14	30.33	7.213
hybrid	KNN	median	X	DecisionTreeRegressor	92.14	30.33	7.213
hybrid	KNN	median		GradientBoostingRegressor	91.76	20.54	7.386
hybrid	KNN	median	X	GradientBoostingRegressor	91.76	20.54	7.386