Facultad de Ciencias

DESARROLLO DE METODOLOGÍAS BASADAS EN APRENDIZAJE AUTOMÁTICO PARA EL MODELADO HIDRODINÁMICO A ESCALA LOCAL Y SU APLICACIÓN EN LA BAHÍA DE SANTANDER

(Development of Methodologies Based on Machine Learning for Local-Scale Hydrodynamic Modeling and its Application in the Bay of Santander)

Trabajo de Fin de Máster
para acceder al

MÁSTER EN CIENCIA DE DATOS

Autor: Mirko Rupani

Director\es: Ana Julia Abascal Santillana
Rodrigo García Manzanas

UC UIMP

DESARROLLO DE METODOLOGÍAS BASADAS EN APRENDIZAJE AUTOMÁTICO PARA EL MODELADO HIDRODINÁMICO A ESCALA LOCAL Y SU APLICACIÓN EN LA BAHÍA DE SANTANDER

Autor: Mirko Rupani
Directores: Ana Julia Abascal Santillana y Rodrigo García Manzanas
Trabajo Final de Máster Universitario en Ciencia de Datos | Master in Data Science

RESUMEN

La conservación y el uso sostenible de los océanos y mares son prioridades reconocidas a nivel internacional para alcanzar los Objetivos de Desarrollo Sostenible establecidos en la Agenda 2030. Para la gestión de los ecosistemas costeros, es fundamental contar con una estimación de las corrientes de alta resolución, tanto de periodos históricos (*hindcast*) como futuros (*forecast*). Obtenerla mediante modelado numérico (enfoque tradicional) resulta muy costoso computacionalmente, y puede ser inviable en ciertos casos.

Para abordar la necesidad de contar con estimaciones precisas y de alta resolución de corrientes marinas, reduciendo el coste computacional, este Trabajo de Fin de Máster (TFM) se centra en el desarrollo de metodologías para el modelado hidrodinámico a escala local (p. ej. puertos, bahías, estuarios), utilizando diferentes técnicas de aprendizaje automático, y el análisis comparativo de las mismas. Se ha seleccionado como caso de estudio para la aplicación de las metodologías la Bahía de Santander, y se han considerado como variables objetivo el nivel del mar y las velocidades superficiales de la corriente (ambas componentes).

Las tareas realizadas incluyen (i) la revisión exhaustiva del estado del conocimiento, (ii) la selección de forzamientos relevantes (predictores), (iii) el modelado numérico con el *software* Delft3D para obtener los predictandos, y (iv) la implementación y validación de diferentes técnicas de aprendizaje automático, prestando especial atención a la optimización de hiperparámetros.

Todas las técnicas evaluadas han demostrado ser competitivas en el modelado de corrientes y nivel del mar, con las ventajas que eso conlleva respecto al uso de modelos numéricos tradicionales en términos de coste computacional. Entre ellas, destacan las redes neuronales recurrentes *long short-term memory* (RNN-LSTM) por su precisión en la predicción, y por la capacidad de extrapolación espacial de las configuraciones óptimas de hiperparámetros obtenidas.

Palabras clave: modelado hidrodinámico, *downscaling*, aprendizaje automático, análogos, árboles de decisión, AdaBoost, *deep learning*, redes neuronales recurrentes, RNN-LSTM, Bahía de Santander.

INTRODUCCIÓN

El enfoque tradicional para obtener corrientes de alta resolución consiste en la implementación de técnicas de *downscaling* dinámico, basadas en el anidamiento de modelos numéricos a diferentes escalas para aumentar la resolución espacial. Sin embargo, la aplicación de estas técnicas es compleja y computacionalmente costosa.

Como alternativa al *downscaling* dinámico, en otros campos como la meteorología y para otras variables marinas, como el oleaje (Camus et al., 2014) y el nivel del mar (Cid et al., 2017), se han desarrollado y aplicado satisfactoriamente técnicas de *downscaling* estadístico, que establecen relaciones (estadísticas) entre las variables a escala regional (predictores) y observaciones de alta resolución en costa (predictandos). Con el desarrollo generalizado del *big data*, varios autores han introducido técnicas de *machine learning* e inteligencia artificial a la oceanografía. No obstante, el grado de avance en la implementación de estas técnicas para modelado hidrodinámico es muy limitado, y no se han encontrado en la literatura científica aplicaciones en el refinamiento de la resolución espacial del campo de corrientes.

OBJETIVOS

El objetivo general del TFM es generar el conocimiento científico que permita desarrollar un marco metodológico basado en técnicas de aprendizaje automático para el modelado hidrodinámico a escala local y su aplicación en la Bahía de Santander.

Para ello, se han planteado como objetivos específicos: (i) analizar las metodologías existentes, (ii) seleccionar las variables océano-meteorológicas predictoras más importantes, (iii) recopilar las bases de datos disponibles, (iv) examinar la variabilidad de los predictores y seleccionar un periodo que explique la máxima variabilidad, (v) modelar numéricamente ese periodo y obtener los predictandos, (vi) implementar el código para entrenar y optimizar diferentes técnicas de aprendizaje automático, (vii) aplicar en puntos representativos de la Bahía de Santander y validar los resultados, y (viii) realizar un análisis comparativo de las distintas técnicas implementadas.

ZONA DE ESTUDIO

La metodología desarrollada se ha aplicado en la Bahía de Santander, que conforma el principal estuario de la costa norte de España, situado en el litoral cántabro (Figura 1). Se trata de un estuario de aguas poco profundas, macromareal (*i.e.* la carrera de marea supera los 4 m) y de mareas semidiurnas. Su principal aportación fluvial es la del río Miera a través de la Ría de Cubas, la cual resulta muy pequeña frente al prisma de marea (*i.e.* el volumen de agua intercambiado entre el estuario y el mar abierto durante un ciclo de marea completo). Por esta razón, las dinámicas mareales suelen prevalecer sobre las fluviales y puede considerarse un estuario bien mezclado en vertical.

Figura 1. Bahía de Santander y principales rías.

METODOLOGÍA

La secuencia de tareas seguida para cumplir con los objetivos planteados es la que se muestra en la Figura 2.

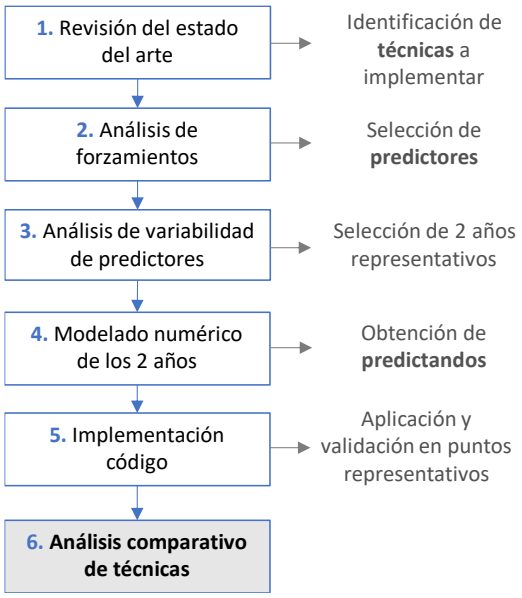


Figura 2. Esquema de la metodología planteada.

SELECCIÓN DE TÉCNICAS

Con base en la revisión del estado del arte y los objetivos, se ha considerado una selección ilustrativa de técnicas de aprendizaje automático de distinto grado de complejidad:

- Predicción condicionada a análogos
- AdaBoost *regression*
- Redes neuronales recurrentes *long short-term memory* (RNN-LSTM)

PREDICTORES Y PREDICTANDOS

Existen muchos elementos forzadores de corrientes y nivel del mar en la Bahía. No obstante, dadas las características de la zona de estudio (estuario macromareal, gran prisma de marea, aportación fluvial pequeña, etc.), estarán determinados principalmente por la marea astronómica. Adicionalmente, se verán influenciados por el caudal del río Miera y el viento. En consecuencia, los predictores que se han considerado son:

- Carrera de marea
- Caudal del río Miera
- Velocidad del viento (componentes zonal y meridional)

Por otro lado, como el objetivo es relacionar las condiciones fuera de la Bahía (escala regional), con condiciones locales en el interior, también es necesario incluir como *inputs*:

- Nivel del mar a escala regional
- Corrientes (ambas componentes) a escala regional

Los predictandos (nivel del mar y corrientes a escala local) se han obtenido a través de modelado numérico con Delft3D.

OPTIMIZACIÓN DE HIPERPARÁMETROS

Obtenidos los datos de entrenamiento, se ha realizado una optimización de hiperparámetros, fundamental en el desarrollo de modelos de aprendizaje automático, ya que puede mejorar significativamente su rendimiento. El proceso implica la exploración de diferentes combinaciones de hiperparámetros y encontrar la configuración que maximiza alguna métrica de rendimiento, para lo cual existen diferentes métodos.

En el caso de análogos, no se ha implementado una optimización automatizada, sino que se han explorado manualmente diferentes configuraciones seleccionadas con criterio de experto. Para el modelo AdaBoost, se ha utilizado una búsqueda en grilla, que permite explorar sistemáticamente una cuadrícula de combinaciones y evaluar cada una mediante validación cruzada. En el caso de las RNN-LSTM, se ha implementado una estrategia de búsqueda basada en bandas hiperparamétricas que evalúa muchas configuraciones y elimina las menos prometedoras de manera progresiva (Li et al., 2018).

APLICACIÓN Y VALIDACIÓN

Los modelos optimizados se han aplicado para reconstruir la hidrodinámica en puntos representativos de la Bahía, y se han validado mediante la comparación de los valores predichos con los obtenidos en el modelado numérico (*ground truth*) y la obtención de métricas que permiten evaluar diferentes aspectos de las predicciones realizadas.

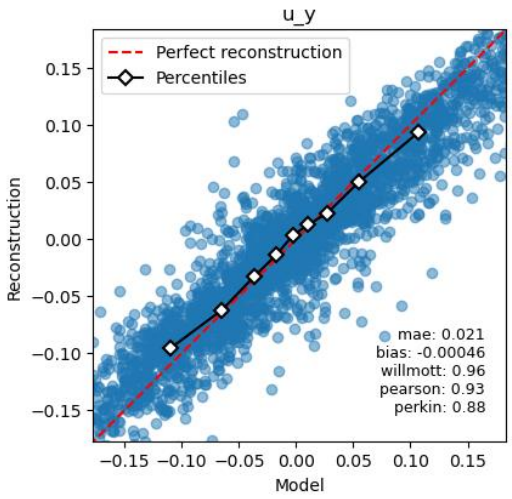


Figura 3. Resultados obtenidos en un punto en el interior de la Bahía para la componente meridional de la velocidad superficial de la corriente, con RNN-LSTM.

En la Figura 3 se muestra un ejemplo de validación mediante *scatter plot*, y las métricas obtenidas.

ANÁLISIS COMPARATIVO

Obtenidas las métricas en todas las estaciones, se ha llevado a cabo un análisis comparativo. En la Figura 4 se muestran los valores de MAE y correlación de Pearson obtenidos en 5 puntos característicos (estaciones) de la Bahía con diferentes técnicas.

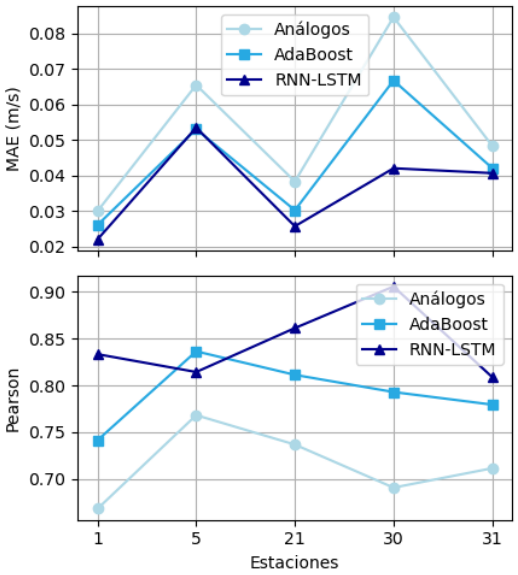


Figura 4. Métricas obtenidas para la componente zonal de la velocidad superficial de la corriente.

Además de la calidad de los resultados, también se ha considerado la eficiencia en cuanto a recursos computacionales necesarios para el entrenamiento y la predicción.

CONCLUSIONES

Del análisis, surge que todas las técnicas han demostrado ser efectivas en la predicción de corrientes y nivel del mar. Además, presentan ventajas respecto a los modelos numéricos tradicionales en términos de coste computacional, ya que una vez entrenadas, son más rápidas y menos exigentes en recursos. Esto sugiere la potencialidad del aprendizaje automático para el modelado hidrodinámico.

Las RNN-LSTM han presentado la mayor precisión en la predicción, aunque la optimización de hiperparámetros adoptada para ese método tiene un coste computacional elevado. No obstante, las configuraciones óptimas obtenidas han demostrado ser extrapolables espacialmente. Es decir, los hiperparámetros óptimos obtenidos para un punto funcionan bien en otros puntos, garantizando la escalabilidad del método. Por lo tanto, de las técnicas evaluadas, la basada

en RNN-LSTM resulta la más adecuada (por su precisión y escalabilidad) para la predicción hidrodinámica de alta resolución a escala local.

REFERENCIAS

Camus, P., Menendez, M., Mendez, F. J., Izaguirre, C., Espejo, A., Canovas, V., Perez, J., Rueda, A., Losada, I. J., & Medina, R. (2014). A weather-type statistical downscaling framework for ocean wave climate. *Journal of Geophysical Research: Oceans*, *119*(11), 7389–7405.

Cid, A., Camus, P., Castanedo, S., Méndez, F. J., & Medina, R. (2017). Global reconstructed daily surge levels from the 20th Century Reanalysis (1871–2010). *Global and Planetary Change*, *148*, 9–21.

Li, L., Jamieson, K., DeSalvo, G., Rostamizadeh, A., & Talwalkar, A. (2018). Hyperband: A novel bandit-based approach to hyperparameter optimization. *Journal of Machine Learning Research*, *18*(185), 1–52.

UC UIMP

DEVELOPMENT OF METHODOLOGIES BASED ON MACHINE LEARNING FOR LOCAL-SCALE HYDRODYNAMIC MODELING AND ITS APPLICATION IN THE BAY OF SANTANDER

Author: Mirko Rupani
Directors: Ana Julia Abascal Santillana and Rodrigo García Manzanas
Final Project for the Master's Degree in Data Science

ABSTRACT

The conservation and sustainable use of oceans are internationally recognized priorities for achieving the Sustainable Development Goals established in the 2030 Agenda. For the management of coastal ecosystems, it is essential to have high-resolution current estimates for both historical periods (hindcast) and future periods (forecast). Obtaining these through numerical modeling (traditional approach) is computationally very expensive and can be unfeasible in certain cases.

To address the need for accurate and high-resolution marine current estimates while reducing computational costs, this Final Master's Project (TFM) focuses on developing methodologies for local-scale hydrodynamic modeling (e.g., ports, bays, estuaries) using various machine learning techniques, and performing a comparative analysis. The Bay of Santander was selected as a case study for the application of these methodologies, and sea level and surface current velocities (both components) were considered as target variables.

The tasks carried out include (i) a comprehensive review of the state of knowledge, (ii) the selection of relevant forcings (predictors), (iii) numerical modeling with Delft3D software to obtain the predictands, and (iv) the implementation and validation of different machine learning techniques, paying special attention to hyperparameter optimization.

All evaluated techniques have demonstrated competitiveness in modeling currents and sea level, offering advantages over traditional numerical models in terms of computational cost. Among them, long short-term memory recurrent neural networks (RNN-LSTM) stand out for their accuracy in prediction and the capacity for spatial extrapolation of the optimal hyperparameter configurations obtained.

Keywords: hydrodynamic modeling, downscaling, machine learning, analogs, decision trees, AdaBoost, deep learning, recurrent neural networks, RNN-LSTM, Bay of Santander.

INTRODUCTION

The traditional approach to obtaining high-resolution currents involves implementing dynamic downscaling techniques, based on nesting numerical models at different scales to increase spatial resolution. However, the application of these techniques is complex and computationally expensive.

As an alternative to dynamic downscaling, statistical downscaling techniques have been successfully developed and applied in other fields such as meteorology and for other marine variables like waves (Camus et al., 2014) and sea level (Cid et al., 2017). These techniques establish statistical relationships between regional-scale variables (predictors) and high-resolution coastal observations (predictands). With the

widespread development of big data, several authors have introduced machine learning and artificial intelligence techniques into oceanography. However, the progress in implementing these techniques for hydrodynamic modeling is very limited, and no applications have been found in the scientific literature for refining the spatial resolution of the current field.

OBJECTIVES

The general objective of this TFM is to generate scientific knowledge that enables the development of a methodological framework based on machine learning techniques for local-scale hydrodynamic modeling and its application in the Bay of Santander.

To achieve this, the specific objectives are: (i) to analyze existing methodologies, (ii) to select the most important metocean predictor variables, (iii) to gather the available databases, (iv) to examine the variability of the predictors and select a period that explains the maximum variability, (v) to numerically model this period and obtain the predictands, (vi) to implement the code to train and optimize different machine learning techniques, (vii) to apply these techniques to representative points in the Bay of Santander and validate the results, and (viii) to perform a comparative analysis of the different techniques implemented.

STUDY AREA

The developed methodology has been applied in the Bay of Santander, which forms the main estuary on the northern coast of Spain and is located on the Cantabrian coast (Figure 1). It is a shallow, macrotidal estuary (i.e., the tidal range exceeds 4 meters) with semidiurnal tides. Its main fluvial contribution comes from the Miera River through the Cubas Estuary, which is very small compared to the tidal prism (i.e., the volume of water exchanged between the estuary and the open sea during a complete tidal cycle). For this reason, tidal dynamics usually prevail over fluvial dynamics, and it can be considered a vertically well-mixed estuary.

Figure 1. Bay of Santander and main estuaries.

METHODOLOGY

The sequence of tasks followed to meet the stated objectives is shown in Figure 2.

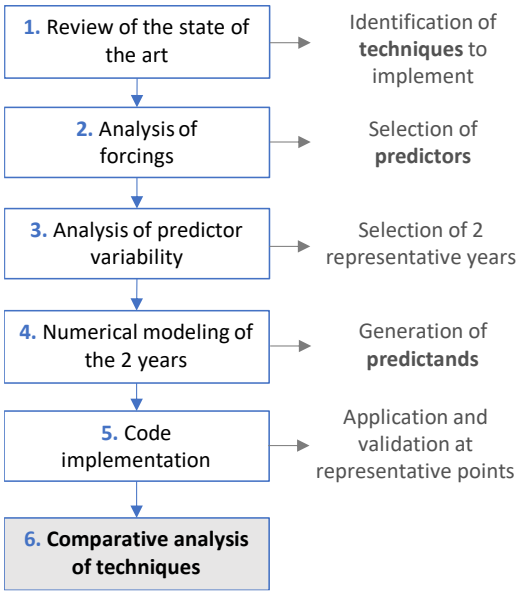


Figure 2. Diagram of the proposed methodology.

SELECTION OF TECHNIQUES

Based on the review of the state of the art and the objectives, an illustrative

selection of machine learning techniques of varying complexity has been considered:

- Prediction conditioned on analogs
- AdaBoost regression
- Long short-term memory recurrent neural networks (RNN-LSTM)

PREDICTORS AND PREDICTANDS

There are many forcing elements of currents and sea level in the Bay. However, given the characteristics of the study area (macrotidal estuary, large tidal prism, small river discharge, etc.), they will primarily be determined by astronomical tides. Additionally, they will be influenced by the flow of the Miera River and the wind. Consequently, the predictors that have been considered are:

- Tidal range
- Miera River discharge
- Wind speed (zonal and meridional components)

On the other hand, since the objective is to relate conditions outside the Bay (on a large scale) to local conditions inside, it is also necessary to include as inputs:

- Large-scale sea level
- Currents (both components) on a large scale

The predictands (local scale sea level and currents) have been obtained through numerical modeling with Delft3D.

HYPERPARAMETER OPTIMIZATION

Once the training data has been obtained, hyperparameter optimization has been performed, which is crucial in the development of machine learning models as it can significantly improve their performance. This process involves exploring different combinations of hyperparameters to find the configuration that maximizes a chosen performance metric, using various methods.

In the case of analogs, automated optimization has not been implemented; instead, different configurations selected by expert judgment have been manually explored. For the AdaBoost model, a grid search has been utilized, allowing systematic exploration of a grid of parameter combinations and evaluating each using cross-validation. For RNN-LSTM models, a tuning strategy based on hyperparametric bands has been implemented, assessing numerous configurations and progressively discarding less promising ones (Li et al., 2018).

APPLICATION AND VALIDATION

The optimized models have been applied to reconstruct the hydrodynamics at representative points within the Bay and have been validated by comparing predicted values with those obtained from numerical modeling (ground truth). Metrics have been obtained to evaluate various aspects of the predictions made.

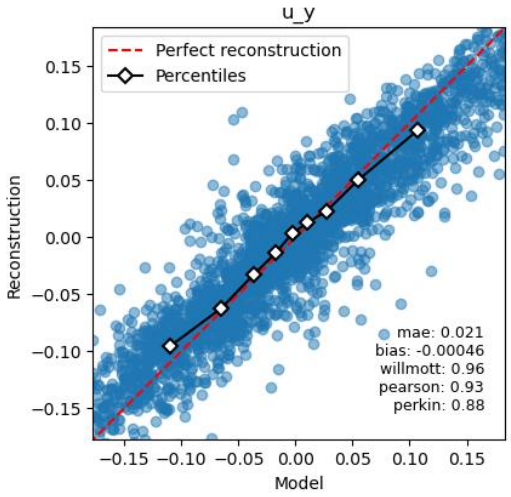


Figure 3. Results obtained at a point inside the Bay for the meridional component of surface current velocity using RNN-LSTM.

Figure 3 shows an example of validation through a scatter plot and the metrics obtained.

COMPARATIVE ANALYSIS

After obtaining metrics at all stations, a comparative analysis has been conducted. Figure 4 displays Mean Absolute Error (MAE) and Pearson correlation values obtained at 5 characteristic points (stations) within the Bay using different techniques.

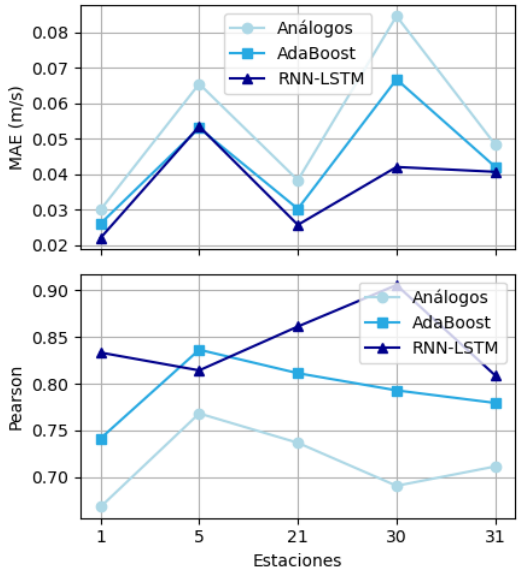


Figure 4. Metrics obtained for the zonal component of surface current velocity.

In addition to assessing result quality, efficiency in terms of computational resources required for training and prediction has also been considered.

CONCLUSIONS

From the analysis, it is evident that all the techniques have proven effective in predicting currents and sea level. Furthermore, they offer advantages over traditional numerical models in terms of computational cost; once trained, they are faster and less resource-intensive. This highlights the potential of machine learning for hydrodynamic modeling.

Although RNN-LSTM has shown the highest accuracy in prediction, the hyperparameter optimization required for this method comes with a high computational cost. However, the optimal configurations obtained have demonstrated spatial extrapolation capability. This means that the optimal hyperparameters found for one point perform well at other points, ensuring the scalability of the method. Therefore, among the evaluated techniques, the RNN-LSTM-based approach appears to be the most suitable for high-resolution hydrodynamic prediction at a local scale.

REFERENCES

Camus, P., Menendez, M., Mendez, F. J., Izaguirre, C., Espejo, A., Canovas, V., Perez, J., Rueda, A., Losada, I. J., & Medina, R. (2014). A weather-type statistical downscaling framework for ocean wave climate. *Journal of Geophysical Research: Oceans*, *119*(11), 7389–7405.

Cid, A., Camus, P., Castanedo, S., Méndez, F. J., & Medina, R. (2017). Global reconstructed daily surge levels from the 20th Century Reanalysis (1871–2010). *Global and Planetary Change*, *148*, 9–21.

Li, L., Jamieson, K., DeSalvo, G., Rostamizadeh, A., & Talwalkar, A. (2018). Hyperband: A novel bandit-based approach to hyperparameter optimization. *Journal of Machine Learning Research*, *18*(185), 1–52.

ÍNDICE

1 INTRODUCCIÓN

En el presente informe se expone el Trabajo de Fin de Máster (TFM) realizado por el alumno Mirko Rupani, estudiante del Máster Universitario en Ciencia de Datos / *Master in Data Science* 2023/2024, dictado por la Universidad de Cantabria y la Universidad Internacional Menéndez Pelayo. Los profesores Ana Julia Abascal Santillana y Rodrigo García Manzanas han actuado como codirectores del Trabajo.

En el marco del TFM se ha realizado un modelado hidrodinámico de alta resolución en la Bahía de Santander, utilizando diferentes técnicas de aprendizaje automático. El análisis se ha enfocado principalmente en el modelado de las corrientes y el nivel del mar. Para ello, se ha llevado a cabo una revisión del estado del conocimiento, seguida de un análisis y selección de forzamientos relevantes para las corrientes en la Bahía de Santander, basado en el conocimiento físico del problema a modelar. Se ha estudiado la variabilidad de los predictores, considerando las extensiones de las bases de datos disponibles, y se han seleccionado los dos años que explican la máxima variabilidad posible. Estos 2 años representativos se han modelado utilizando el *software* Delft3D, de manera de obtener los datos de entrenamiento correspondientes.

A continuación, se ha implementado el código necesario para entrenar diversas técnicas de aprendizaje automático, intentado cubrir todo el espectro en términos de complejidad, comenzando desde enfoques menos complejos hasta los más avanzados. En todos los casos se ha realizado el correspondiente ajuste de hiperparámetros mediante técnicas de validación cruzada.

En primer lugar, se ha implementado la metodología en puntos representativos de la Bahía de Santander, y se han validado los resultados utilizando datos del propio modelado numérico (subconjunto reservado para *test*). Se han obtenido gráficas de validación y diferentes indicadores estadísticos en puntos representativos, de forma de poder evaluar qué tan bien se comporta cada una de las técnicas en diferentes sectores de la Bahía.

A partir de los resultados obtenidos, se ha evaluado individualmente la efectividad de cada técnica para el modelado de las variables consideradas. Asimismo, se ha realizado un análisis comparativo de las distintas técnicas, haciendo especial énfasis en la relación entre la precisión de los resultados, y la complejidad y recursos computacionales necesarios para la implementación de cada técnica.

1.1 MOTIVACIÓN

La conservación y el uso sostenible de los océanos y mares son prioridades reconocidas a nivel internacional para alcanzar los Objetivos de Desarrollo Sostenible establecidos en la Agenda 2030. Sin embargo, la contaminación marina (derrames accidentales de hidrocarburos o químicos, contaminación por plásticos, etc.) es un riesgo común en el entorno costero. La creciente presión ejercida hacia los océanos por las diferentes actividades antrópicas ha llevado a la necesidad de desarrollar en las últimas décadas herramientas para la prevención y respuesta ante contaminación en la costa.

Estas herramientas están basadas en la utilización de modelos numéricos que permiten, por un lado, mejorar la respuesta mediante la predicción de la trayectoria de los contaminantes en tiempo real y, por otro lado, mejorar la planificación mediante el análisis probabilístico de la peligrosidad y el riesgo en la costa ante un posible evento de contaminación. En ambos casos, el modelado numérico se basa en la utilización de información océano-meteorológica para prever el transporte, dispersión y, de corresponder, degradación, utilizando como forzamientos el viento, las corrientes y/o el oleaje.

En las últimas décadas, se han realizado esfuerzos importantes en el desarrollo de bases de datos océano-meteorológicos que proporcionan datos de corrientes que permiten aplicar estas metodologías a escala regional (3 – 10 km). Sin embargo, esta resolución resulta insuficiente para modelar la evolución del contaminante cuando se produce en el interior de un puerto, bahía o estuario. En estos casos, es necesario aumentar la resolución espacial, en un proceso conocido como *downscaling*.

El enfoque tradicional para aumentar la resolución espacial consiste en la implementación de técnicas de *downscaling* dinámico, basadas en el anidamiento de modelos a diferentes escalas o el anidamiento de mallas, que resuelven de manera numérica las ecuaciones diferenciales en derivadas parciales que explican el flujo. Sin embargo, la aplicación de estas técnicas es compleja y computacionalmente costosa. Como alternativa al *downscaling* dinámico, en otros campos como la meteorología y para otras variables marinas, como el oleaje y el nivel del mar, se han desarrollado y aplicado satisfactoriamente técnicas de *downscaling* estadístico. Estas técnicas establecen relaciones entre las variables a escala regional (predictores) y observaciones de alta resolución en la costa (predictandos) mediante técnicas estadísticas, lo cual, en general, es más rápido y menos costoso computacionalmente. Sin embargo, su aplicación para variables hidrodinámicas, y especialmente para corrientes, es prácticamente inexistente.

Contar con metodologías eficientes para el modelado de periodos históricos (*hindcast*) y futuros (*forecast*) no solo es beneficioso para la investigación científica, sino que podría generar un impacto significativo en la gestión y protección de los ecosistemas marinos. El desarrollo de sistemas operacionales de predicción de corrientes de alta resolución (costera y local), tanto a nivel nacional como europeo, es reciente y limitado hoy en día, debido, entre otros aspectos, a la complejidad de la implementación y al coste computacional de ejecución y mantenimiento de estos modelos. Sin embargo, a través de técnicas como las desarrolladas en el TFM, podrían implementarse modelos operacionales a un coste mucho menor, lo que permitiría extender su alcance y cubrir fácilmente cualquier sitio.

En consecuencia, se concluye que es necesario generar nuevo conocimiento en el uso de técnicas híbridas y estadísticas, que permitan: i) optimizar el proceso de *downscaling* de corrientes para obtener la información de alta resolución requerida a escala local; y ii) optimizar el modelado de la evolución de la contaminación marina en zonas de bahías, puertos y estuarios (escala local), donde la alta resolución de las corrientes es un factor determinante (véase la Sección 3.1 IMPORTANCIA DEL MODELADO DE CORRIENTES).

2 OBJETIVOS

2.1 OBJETIVO GENERAL

El objetivo general del TFM es generar el conocimiento científico que permita desarrollar un marco metodológico basado en técnicas de aprendizaje automático para el modelado hidrodinámico a escala local (bahías y estuarios) y su aplicación en la Bahía de Santander. Las técnicas analizadas se aplicarán para la simulación de corrientes y nivel del mar. Los resultados obtenidos contribuirán a mejorar la comprensión y gestión de este tipo de entornos, y tienen el potencial de beneficiar a comunidades costeras y a la sociedad en su conjunto, promoviendo un desarrollo más sostenible y resiliente frente al cambio climático y a eventos de contaminación antrópica.

2.2 OBJETIVOS ESPECÍFICOS

Para la concreción del objetivo general, se han establecido una serie de objetivos específicos, que se alinean con las tareas que se presentan en la Sección 4 METODOLOGÍA. A saber:

- analizar las metodologías existentes para el modelado de variables oceanográficas mediante técnicas basadas en aprendizaje automático;
- determinar las variables océano-meteorológicas que podrían actuar como forzamientos de las variables a predecir y seleccionar las más importantes, que actuarán como predictores;
- recopilar las bases de datos disponibles de los predictores seleccionados;
- examinar la variabilidad de los predictores, excluyendo la marea astronómica (debido a su naturaleza determinista), considerando las extensiones de las bases de datos disponibles, y seleccionar un periodo continuo de dos años que explique la máxima variabilidad posible;
- modelar numéricamente los dos años representativos utilizando el *software* Delft3D, para obtener los datos de entrenamiento correspondientes (predictandos);
- implementar el código necesario para entrenar y optimizar diferentes técnicas de aprendizaje automático, de manera de cubrir todo el espectro en términos de complejidad, comenzando desde enfoques menos complejos hasta los más avanzados;
- aplicar en puntos representativos de la Bahía de Santander, y validar los resultados utilizando datos provenientes de modelado numérico;
- realizar un análisis comparativo de las distintas técnicas implementadas, y evaluar cómo se comporta cada una en diferentes sectores de la Bahía.

3 ZONA DE ESTUDIO

Como caso de estudio para el desarrollo del TFM se ha seleccionado la Bahía de Santander, situada en el litoral cántabro, que conforma el principal estuario de la costa norte de España. Cuenta con una extensión de 22,42 km^2, y un perímetro de aproximadamente 90 km. En las costas de la Bahía confluyen 5 municipios, que totalizan una población de 231.079 habitantes (Instituto Cántabro de Estadística, 2023). Se trata de un estuario macromareal (*i.e.* la carrera de marea supera los 4 m), de mareas semidiurnas y aguas poco profundas. Dadas estas características, aproximadamente dos tercios de su superficie resulta intermareal.

Teniendo en cuenta sus patrones de estratificación, el estuario puede considerarse bien mezclado. Esto se debe a que su principal aportación fluvial es la proveniente del río Miera, que resulta muy pequeña frente al prisma de marea (*i.e.* el volumen de agua intercambiado entre el estuario y el mar abierto durante un ciclo de marea completo). Por esta razón, las dinámicas mareales suelen prevalecer sobre las fluviales.

El río Miera, cuya descarga fluvial en la Bahía a través de la ría de Cubas se ha considerado en el presente estudio, es un curso fluvial de aproximadamente 45 km de longitud, los cuales recorre desde su nacimiento a 1.300 m de altura, hasta su desembocadura. La cuenca hidrográfica del Miera posee unos 297 km^2, y aporta un caudal medio anual de 5,24 m^3/s (Confederación Hidrográfica del Cantábrico). No obstante, la variabilidad en su aportación es muy elevada, como puede observarse en la Figura 6, en la que se representa una serie temporal de caudales medios diarios.

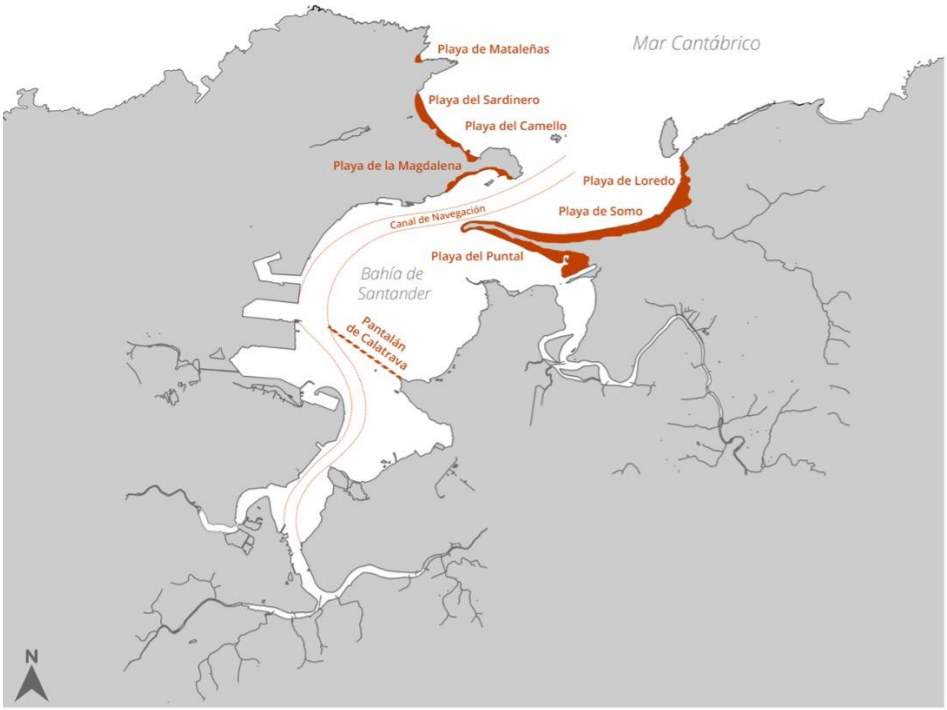


Figura 1. Principales zonas de baño en el área de la Bahía de Santander.

La desembocadura de la ría de Cubas está parcialmente cerrada en su margen derecha por el Puntal de Somo, una formación dunar que actúa como abrigo natural contra el oleaje y forma parte del Lugar de Importancia Comunitaria (LIC) Dunas del Puntal y Estuario del Miera.

Además de este LIC, que es parte de la Red Natura 2000, destacan 7 zonas de producción de moluscos ubicadas en los páramos intermareales frente a Pedreña y Elechas. Cerca de la desembocadura del río Miera, se encuentra el punto de vertido de la EDAR de Suesa, que trata aguas residuales de varios municipios. Además, las playas del Puntal, Somo, Loredo, La Magdalena, El Camello, El Sardinero y Mataleñas están designadas como zonas de baño, por lo que son sometidas a controles de calidad sanitaria.

Desde la bocana y hacia el interior del estuario, destaca la presencia de una canal de navegación contigua a la margen oeste (Figura 1), que sirve de acceso a la zona portuaria de Santander (Raos) y a los muelles de Astillero, cuyo mantenimiento se realiza a través de dragados periódicos. Además, desde 1966, del Pantalán de Calatrava se ubica sobre la margen derecha del estuario, salvando 1800 m de zonas de baja profundidad hasta la canal de navegación, de manera de permitir la carga y descarga de líquidos y gases.

3.1 IMPORTANCIA DEL MODELADO DE CORRIENTES

Visto lo anterior, se puede concluir que contar con una estimación de las corrientes de alta resolución en la Bahía de Santander, ya sea de periodos históricos (*hindcast*) o futuros (*forecast*), bajo ciertos forzamientos actuantes en una situación real o hipotética, resulta de especial importancia para:

- la gestión de espacios protegidos (LIC Dunas del Puntal y Estuario del Miera y zonas de producción de moluscos) y zonas de baño;
- la elaboración de estudios de calidad del agua;
- el mantenimiento de la canal de navegación;
- la gestión de posibles derrames accidentales en las zonas portuarias y en el Pantalán de Calatrava;
- el desarrollo de estudios de dispersión y acumulación de residuos;
- el análisis del riesgo frente a derrames de hidrocarburos o acumulación de residuos plásticos;
- la elaboración de estudios morfodinámicos teniendo en cuenta la morfología actual del estuario, o bien cualquier situación pasada o futura, determinada por diferentes alteraciones a las que podría verse sometido.

4 METODOLOGÍA

Con el objetivo de, por un lado, establecer una metodología para modelar corrientes de alta resolución mediante la aplicación de técnicas de aprendizaje automático; y, por el otro, estudiar la eficacia de cada técnica en la Bahía de Santander, se han planteado una serie de pasos. A continuación, se detalla la secuencia de tareas seguida, que también se esquematiza en la Figura 2.

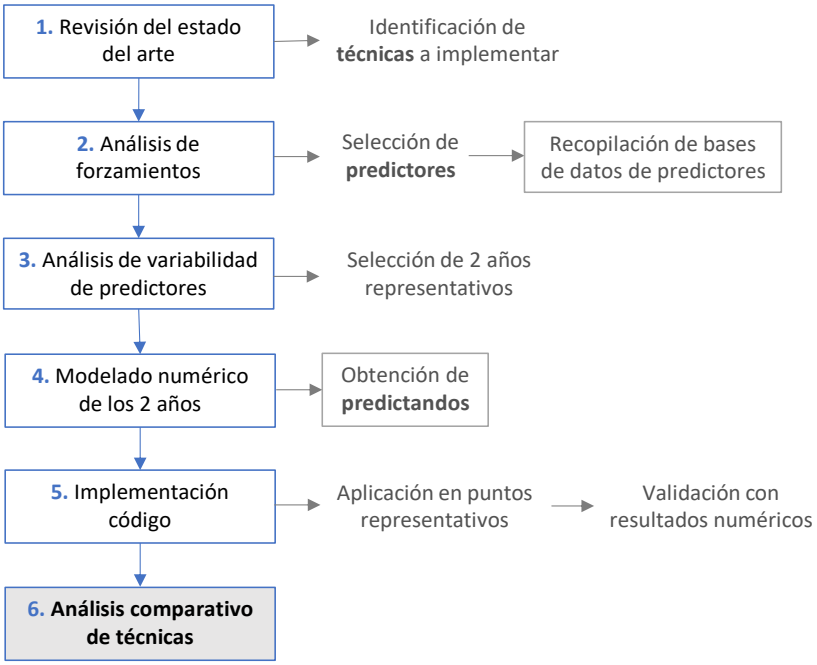


Figura 2. Esquema de la metodología seguida en el TFM

1. Revisión del estado del arte en la aplicación de técnicas de aprendizaje automático para la simulación de variables meteorológicas y oceanográficas. Como resultado de esta tarea, se han identificado las técnicas de *downscaling* que se consideran más adecuadas para alcanzar los objetivos del TFM.
2. Análisis de forzamientos, basado en el conocimiento físico del problema a modelar. Se han seleccionado los forzamientos más relevantes para las corrientes en la Bahía de Santander, que se tomarán como predictores para el desarrollo de los modelos de *downscaling*.
3. Excluyendo la marea astronómica (debido a su naturaleza determinista), se ha evaluado la variabilidad de los predictores, considerando las extensiones de las bases de datos disponibles, y se han seleccionado dos años que expliquen la máxima variabilidad.
4. Modelado numérico de los 2 años representativos utilizando el *software* Delft3D, de manera de obtener los predictandos para los modelos de *downscaling*.
5. Implementación del código en Python, utilizando programación orientada a objetos. Se ha trabajado con GitHub para llevar un control de versiones eficiente, facilitando la gestión y el seguimiento de los cambios realizados en el código a lo largo del desarrollo del proyecto, así como su compartición y distribución.

Esta tarea incluye, por un lado, el desarrollo, optimización y aplicación de las técnicas en puntos representativos de la Bahía de Santander, para lo que se han utilizado principalmente las librerías scikit-learn y Keras. Por otro lado, la tarea comprende la validación de los resultados utilizando datos provenientes de modelado numérico (*dataset* de *test*).

6. Análisis comparativo de las distintas técnicas, haciendo especial énfasis en la relación entre la calidad de los resultados obtenidos, y la complejidad y recursos computacionales necesarios para el entrenamiento y la predicción en cada caso.

4.1 REVISIÓN DEL ESTADO DEL ARTE

En las últimas décadas, se han desarrollado bases de datos oceanográficas, como las de NOAA en Estados Unidos o CMEMS (Copernicus) en Europa, que proveen información sobre corrientes a escala regional, con una resolución espacial de 3 a 10 kilómetros. Para analizar con mayor precisión la hidrodinámica en los ámbitos costero y local resulta necesario incrementar esta resolución y alcanzar niveles de detalle de aproximadamente 50 metros (escala local) a 300 – 500 metros (escala costera). El enfoque tradicional para hacerlo consiste en la implementación de técnicas de *downscaling* dinámico, basadas en simulaciones anidadas de modelos numéricos hasta alcanzar la resolución deseada. No obstante, esto requiere de un modelado complejo y costoso computacionalmente, especialmente para generar series históricas de amplia cobertura temporal (más de 10 años), como las necesarias para analizar la peligrosidad y el riesgo ante episodios de contaminación marina.

Como alternativa al *downscaling* dinámico, en otros campos como la meteorología y para otras variables marinas, como el oleaje y el nivel del mar, se han desarrollado y aplicado satisfactoriamente técnicas de *downscaling* estadístico y *downscaling* híbrido. En el caso del *downscaling* estadístico, se establecen relaciones entre las variables a escala regional (predictores) y observaciones de alta resolución en costa (predictandos) mediante técnicas estadísticas. Estas técnicas han sido desarrolladas inicialmente en el campo de la meteorología y en los últimos años extendidas a variables marinas, como el oleaje (Camus et al., 2014; Hegermiller et al., 2017) y el nivel del mar (Cid et al., 2017, 2018).

Con el desarrollo generalizado del *big data*, varios autores han introducido técnicas de *machine learning* e inteligencia artificial a la oceanografía. Por ejemplo, Chiri et al. (2019) realizan un modelado estadístico de patrones de corrientes en el Golfo de México, y ejercicios exploratorios de predicción, mediante técnicas de regresión logística autorregresiva. Sarkar et al. (2019) utilizan regresión en procesos Gaussianos para predecir corrientes, a partir de mediciones separadas espaciotemporalmente. Bayindir (2019), Bolton & Zanna (2019), Immas et al. (2021), Liu et al. (2022), Muhamed Ali et al. (2022) y Qian et al. (2022) presentan diferentes metodologías basadas en *deep learning* para la predicción de corrientes y/o dinámicas oceánicas. Si bien los anteriores son ejemplos de implementación de modelado estadístico de corrientes, en ningún caso se plantea un refinamiento de la resolución espacial.

En lo que respecta a la aplicación de técnicas estadísticas de *downscaling* de corrientes, se puede citar el trabajo de Thiria et al. (2023), que desarrollan un mecanismo con redes neuronales para combinar datos de nivel del mar de baja resolución (120 x 122 km) y de

temperatura de la superficie de alta resolución (13 x 14 km), para obtener corrientes a la resolución de la temperatura superficial (13 x 14 km).

En comparación con el *downscaling* dinámico, la implementación de *downscaling* estadístico es más rápida y mucho menos costosa computacionalmente. No obstante, este tipo de modelos podría no comportarse adecuadamente en ciertas situaciones. Por ejemplo, cuando se extrapola hacia condiciones climáticas futuras, ya que, en general, las metodologías propuestas suponen estacionarias las relaciones entre predictores y predictandos (Hernanz et al., 2022).

El *downscaling* híbrido combina técnicas numéricas y estadísticas y se ha aplicado principalmente para obtener oleaje de alta resolución en la franja costera. La metodología más común consiste en el desarrollo de una función de transferencia para la transformación de las condiciones en mar abierto a las condiciones en la costa, a través del modelado numérico de una selección de casos representativos y la posterior reconstrucción en la costa (Camus et al., 2011). Tal y como destacan Camus et al. (2011), los métodos híbridos implican un mayor coste computacional que los estadísticos, porque requieren modelar numéricamente algunos escenarios o periodos representativos, pero tienen la ventaja de que permiten una mejor representación de la variabilidad espacial, parámetro fundamental para el modelado de la contaminación marina. Además de su aplicación para obtener oleaje, las técnicas de *downscaling* híbridas se han aplicado satisfactoriamente en meteorología para modelar viento (Acevedo et al., 2019) y temperatura del aire (Wang et al., 2020).

Particularmente en el modelado de corrientes, la aplicación de técnicas híbridas es muy limitada y enfocada principalmente en la predicción a corto plazo en un punto (Immas et al., 2021; Saha et al., 2016) y a la corrección del error (Hollinger et al., 2012; Saha et al., 2016; Zhang et al., 2022). No se han encontrado en la literatura científica ejemplos de implementación de técnicas híbridas para el refinamiento de la resolución espacial del campo de corrientes.

Como puede observarse, la aplicación de técnicas basadas en aprendizaje automático presenta un gran potencial para la simulación de variables oceanográficas, y concretamente, de variables hidrodinámicas en el ámbito costero. Sin embargo, su aplicación es reciente y limitada, por lo que se requiere investigar y profundizar en su uso para la simulación de variables hidrodinámicas en diferentes escalas espaciotemporales.

De la revisión del estado del arte, se detectan las siguientes lagunas del conocimiento, algunas de las cuales dan lugar a los objetivos del TFM:

- Las técnicas estadísticas basadas en aprendizaje automático se han aplicado satisfactoriamente para el *downscaling* de variables meteorológicas, y recientemente para simular el nivel del mar y algunas variables oceanográficas a escala regional (p. ej. temperatura). Sin embargo, su aplicación para el *downscaling* de variables hidrodinámicas es prácticamente inexistente. Este punto constituye al objetivo principal del TFM.
- Las técnicas de *downscaling* híbridas han demostrado ser efectivas en áreas como la meteorología y variables marinas como el oleaje. Sin embargo, su aplicación en variables hidrodinámicas, especialmente corrientes, es muy limitada. Este punto se ha abordado parcialmente, al plantear las bases metodológicas de la técnica de predicción condicionada a análogos.

- Los estudios recientes muestran el potencial de las nuevas técnicas basadas en aprendizaje automático para la simulación de variables marinas. No obstante, dado el reciente uso de estas técnicas y la escasez de trabajos en el ámbito marino, no se conocen sus ventajas y limitaciones y su idoneidad para las diferentes variables hidrodinámicas. Este punto se ha abordado mediante la aplicación de diferentes técnicas a diferentes variables marinas.

4.1.1 IDENTIFICACIÓN DE TÉCNICAS

Con base en la revisión del estado del arte y los objetivos planteados en el TFM, se ha considerado una selección ilustrativa de técnicas de aprendizaje automático de distinto grado de complejidad. A continuación, se detalla cada una de ellas.

4.1.1.1 PREDICCIÓN CONDICIONADA A ANÁLOGOS

La predicción condicionada a análogos es una técnica no paramétrica de modelado predictivo, desarrollada por Lorenz (1969), que se basa en encontrar casos similares (análogos) en los datos históricos para hacer predicciones sobre nuevos casos. Es una técnica simple que históricamente ha sido muy efectiva en meteorología (Toth, 1989). Si bien su uso ha disminuido en las últimas décadas debido al desarrollo de los modelos numéricos (Van den Dool, 2007), está volviendo a ser relevante gracias al rápido incremento en la disponibilidad de datos y la aparición de nuevas técnicas de apoyo (Chattopadhyay et al., 2020; Zhao & Giannakis, 2016). Además de en meteorología, también se han obtenido muy buenos resultados en variables oceanográficas como el oleaje (Cagigal et al., 2024; Camus et al., 2014; Hegermiller et al., 2017). Puede ser particularmente útil en situaciones donde los datos presentan patrones complejos que son difíciles de capturar con modelos lineales simples.

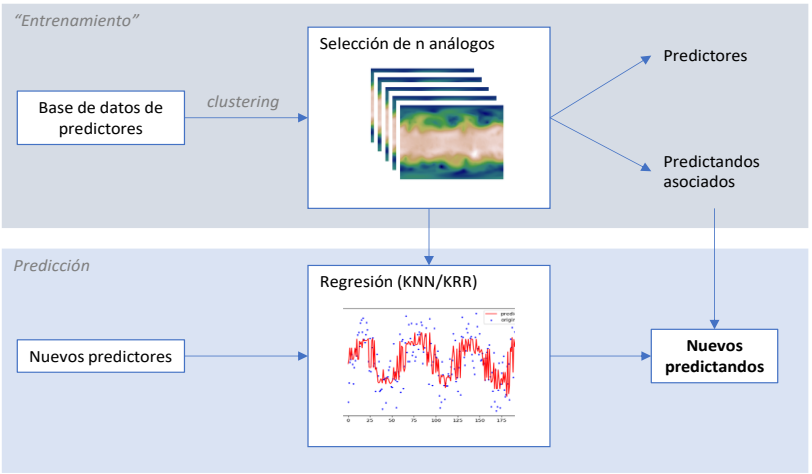


Figura 3. Esquema de la predicción condicionada a análogos.

Al reducir el conjunto total de datos disponibles a un pequeño grupo de casos representativos (a través de un proceso de *clustering*, como se explica más adelante), la regresión condicionada a análogos ofrece una forma eficiente y comprensible de modelar relaciones predictor-predictando, ideal para conjuntos de datos extensos, o cuando los recursos computacionales son limitados. Además, el modelo resultante tiende a ser más robusto frente a datos ruidosos o atípicos.

Por otro lado, debido a la sencillez propia de la técnica, es fácil entender cómo se llegó a una determinada predicción, ya que pueden identificarse claramente los casos utilizados para el análisis. La alta interpretabilidad y explicabilidad puede ser especialmente útil en aplicaciones donde se requiere transparencia y justificación en la toma de decisiones (p. ej. a la hora de decidir cómo actuar frente a un derrame de hidrocarburos).

Finalmente, otra ventaja de la técnica es que permite, a partir de un único modelo (siempre que los predictores sean los mismos), predecir múltiples variables de salida al mismo tiempo, lo cual resulta especialmente útil en el caso de la hidrodinámica.

La implementación de la predicción condicionada a análogos sigue tres pasos principales, que se explican a continuación.

4.1.1.1.1 CLUSTERING

Se realiza un agrupamiento de los datos de entrenamiento para identificar grupos similares. Para esto, se han seleccionado varias técnicas de *clustering,* lineales y no lineales:

- K-*means*: es un algoritmo de agrupamiento no supervisado que particiona un conjunto de datos en K grupos o *clusters*, de modo que cada dato pertenezca al *cluster* a cuyo centroide (representante del *cluster*) se parezca más. El proceso comienza con la selección (aleatoria en principio) de K centroides iniciales. Luego, cada dato se asigna al centroide más cercano, y los centroides se actualizan como la media de los datos asignados a cada *cluster*. Este proceso de asignación y actualización se repite iterativamente hasta que los centroides ya no cambian significativamente o se alcanza un número máximo de iteraciones. K-*means* es popular por su simplicidad y eficiencia en la clasificación de grandes conjuntos de datos. Se ha implementado utilizando la librería scikit-learn, una herramienta robusta y ampliamente utilizada en el ecosistema de Python para aprendizaje automático.
- Algoritmo de máxima disimilitud (MaxDiss): es una técnica utilizada para seleccionar un número específico de centroides "reales" (es decir, datos que pertenecen al conjunto de entrenamiento), máximamente disímiles, que actúan como representantes de los correspondientes grupos. Este algoritmo es especialmente útil en escenarios donde se necesita crear subconjuntos diversos de datos para análisis o para inicializar centroides en algoritmos de *clustering*. Ha sido implementado utilizando un código desarrollado por IHCantabria, disponible en GitHub (García Pérez & IHCantabria - Instituto de Hidráulica Ambiental de la Universidad de Cantabria, 2024).
- *Spectral clustering*: es una técnica de agrupamiento que utiliza el espectro (valores propios) de la matriz de similitud del conjunto de datos para realizar la reducción dimensional antes de aplicar un algoritmo de agrupamiento tradicional como K-*means*. Esta técnica es especialmente útil para identificar grupos en datos que no son necesariamente esféricos y para manejar formas complejas de agrupamiento. Se ha implementado utilizando la biblioteca scikit-learn.

4.1.1.1.2 SELECCIÓN DE ANÁLOGOS

En el caso de MaxDiss, el método devuelve directamente puntos de entrenamiento como representativos (i.e. los análogos). K-*means* y *spectral clustering*, en cambio, identifican centroides "sintéticos" de los grupos, que en general no se corresponden con puntos de entrenamiento. En consecuencia, es necesario seleccionar los análogos de algún modo.

Siguiendo la práctica habitual al aplicar este método, se ha definido como análogo al punto de entrenamiento más cercano al centroide de cada grupo.

4.1.1.1.3 REGRESIÓN

Utilizando únicamente los análogos seleccionados previamente, se entrena un modelo de aprendizaje automático basado en regresión. Se han implementado dos tipos de regresores:

- Regresión por vecinos cercanos (KNN *regression*): es una técnica de regresión que forma parte de los algoritmos de aprendizaje supervisado basado en instancias. A diferencia de los métodos tradicionales de regresión que construyen un modelo a partir de los datos de entrenamiento, KNN *regression* predice el valor de una nueva instancia basándose directamente en los valores de las instancias más cercanas en el conjunto de datos de entrenamiento. Se ha implementado utilizando la biblioteca scikit-learn, lo que ha permitido optimizar el número de vecinos (análogos) a considerar, cuya contribución se ha ponderado con la inversa de la distancia al punto a predecir.
- *Kernel Ridge Regression* (KRR): es una técnica de regresión que combina la *ridge regression* con el truco del *kernel* (*kernel trick*) para manejar relaciones no lineales en los datos. La *ridge regression* añade un término de regularización a la regresión lineal ordinaria, lo que ayuda a prevenir el sobreajuste. Al utilizar el truco del *kernel*, KRR puede capturar relaciones complejas entre las variables de entrada y la variable de salida sin tener que transformar explícitamente los datos a un espacio de características de mayor dimensión. Se ha implementado a través de scikit-learn, utilizando un *kernel* RBF (*radial basis function*). Se ha optimizado el parámetro de regularización y el parámetro gamma del *kernel* RBF, que controla el alcance de la influencia de cada punto de entrenamiento.

Una vez que el regresor está entrenado, se puede utilizar para hacer predicciones sobre nuevos datos.

4.1.1.2 ADABOOST

Las técnicas de *boosting* entrenan modelos secuencialmente. AdaBoost (*Adaptive Boosting*) *regression* es una técnica avanzada de modelado predictivo que combina múltiples modelos simples (*weak models*), típicamente árboles de decisión, con el objetivo de mejorar la precisión y reducir la tendencia al sobreajuste de estos últimos.

Los árboles de decisión son modelos "sencillos" formados por una colección de nodos que permiten representar relaciones no lineales y estructuras complejas en conjuntos de datos heterogéneos y pueden utilizarse tanto en problemas de clasificación como en problemas de regresión. En esencia, en cada nodo se selecciona el predictor que mejor separa el predictando en base a algún criterio de interés (por ejemplo, la reducción de entropía en el caso de árboles de clasificación o la reducción de la varianza en el de árboles de regresión). Este proceso de partición del espacio de predictores se va repitiendo iterativamente, dando lugar a ramificaciones que acaban determinando una estructura de árbol (invertido; es decir, con la raíz en la parte superior y las hojas en la inferior).

Para ello se trabaja iterativamente, ajustando cada árbol de decisión a los errores del anterior, para lo cual se les da un mayor peso a los datos mal predichos en cada paso (comportamiento adaptativo), tal y como se explica más adelante.

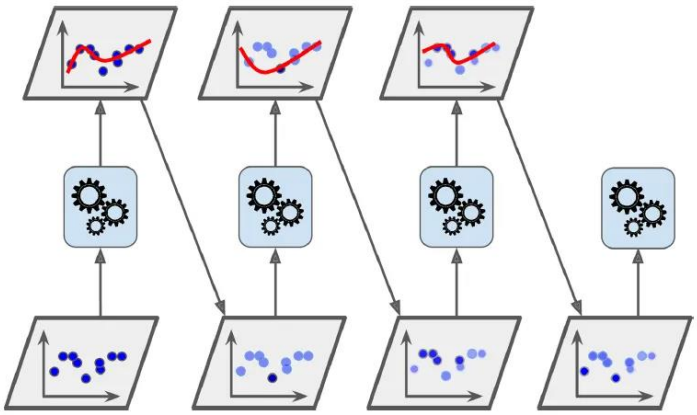


Figura 4. Entrenamiento secuencial AdaBoost. Fuente: (Géron, 2019).

Esta técnica es especialmente útil en el caso de conjuntos de datos grandes y complejos, pues una de sus principales ventajas es la capacidad para modelar relaciones predictor-predictando no lineales. Por este motivo, AdaBoost ha demostrado ser particularmente eficaz en campos como la meteorología (Asadollah et al., 2022; Ratnam et al., 2023).

Además, a pesar de tratarse de una técnica más compleja que las basadas en regresión, AdaBoost sigue siendo eficiente en términos de coste computacional y puede implementarse fácilmente con bibliotecas como scikit-learn (la que se ha utilizado en el TFM), que se basa en el algoritmo Adaboost.R2 (Drucker, 1997). AdaBoost.R2 usa una ponderación de los datos de entrenamiento basada en los errores, de modo que los datos más difíciles de predecir tienen mayor probabilidad de ser seleccionados en las iteraciones posteriores.

Los árboles de decisión son rápidos de entrenar y no paramétricos. No obstante, la técnica AdaBoost puede ser sensible a ruido en los datos, dado que el *boosting* incrementa el peso de los datos difíciles de predecir, que pueden incluir ruidos o valores anómalos. Asimismo, la interpretabilidad de los árboles de decisión puede reducirse debido a la combinación de múltiples modelos.

Es importante destacar que, a diferencia de otros enfoques (como la predicción condicionada a análogos), AdaBoost no puede predecir múltiples variables de salida simultáneamente, por lo que es necesario entrenar un modelo separado para cada variable objetivo.

4.1.1.3 RNN-LSTM

Las redes neuronales recurrentes (RNN, por sus siglas en inglés) son un tipo de red neuronal diseñada para procesar secuencias de datos, como series temporales, texto o datos de audio. A diferencia de las redes neuronales tradicionales, las RNN tienen la capacidad de mantener una memoria interna de la información procesada previamente, lo que les permite capturar dependencias temporales y patrones secuenciales en los datos. Sin embargo, las RNN estándar tienen limitaciones cuando se trata de aprender dependencias a largo plazo debido a problemas como el desvanecimiento y la explosión del gradiente. Para superar estas limitaciones, se desarrollaron las redes *Long Short-Term Memory* (LSTM), un tipo especial de RNN que incluye mecanismos de control más sofisticados, que permiten la retención y el uso selectivo de información relevante durante largos períodos de tiempo.

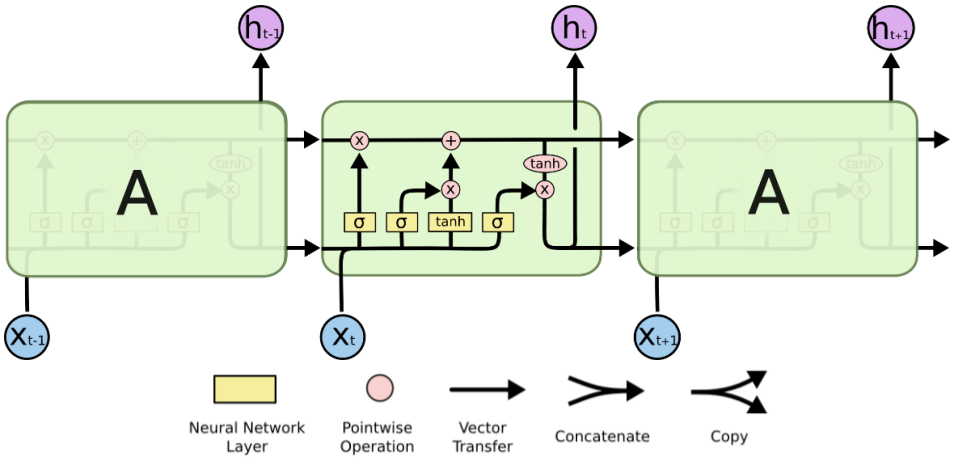


Figura 5. Arquitectura de una RNN-LSTM. Fuente: (Olah, 2015).

En el contexto de la predicción de corrientes, las LSTM sobresalen porque pueden capturar la dinámica compleja y las variaciones estacionales de las series temporales de datos ambientales (Memarzadeh & Keynia, 2020; Salman et al., 2018). La arquitectura de las LSTM permite modelar de manera más precisa las relaciones no lineales y las dependencias a largo plazo entre las observaciones pasadas y futuras, lo que es crucial para obtener predicciones precisas en contextos donde los cambios en las corrientes pueden depender de múltiples factores, pasados y contextuales. La robustez y la precisión de las LSTM en estos escenarios han sido respaldadas por numerosos estudios e implementaciones prácticas, consolidando su reputación como una herramienta potente y confiable para la predicción de series temporales complejas. En el TFM, se han implementado estas redes utilizando la librería Keras, que se basa en los principios descritos en el trabajo de Hochreiter & Schmidhuber (1997), que introdujeron la arquitectura LSTM.

4.2 ANÁLISIS DE FORZAMIENTOS

Existen muchos elementos forzadores de corrientes y nivel del mar en la Bahía. No obstante, dadas las características de la zona de estudio (estuario macromareal, gran prisma de marea, aportación fluvial pequeña, etc.), tanto la velocidad y dirección de la corriente en cada punto, como el nivel de la lámina de agua, van a estar determinados principalmente por la marea astronómica. Adicionalmente, la hidrodinámica en la Bahía de Santander está influenciada por el caudal del río Miera y el viento. La relevancia de estos forzamientos en la hidrodinámica va a depender de diferentes factores, como la variable objeto de estudio (corrientes, niveles), las condiciones atmosféricas y la localización en la Bahía de Santander.

De esta forma, si bien el caudal del río Miera puede variar mucho a lo largo del año (alcanzando picos que superan valores diarios de 150 m^3/s, como se aprecia en la Figura 6), el volumen introducido al estuario entre una bajamar y una pleamar es despreciable frente al prisma de marea del estuario. Por consiguiente, el efecto del río en las corrientes y en el nivel del agua del estuario será pequeño y estará concentrado en las proximidades de la desembocadura, en la zona de la ría de Cubas. Sí será de importancia, no obstante, el efecto del río sobre los valores de salinidad, especialmente cuando se produzcan avenidas (*i.e*. crecidas temporales y excepcionales en el caudal) significativas.

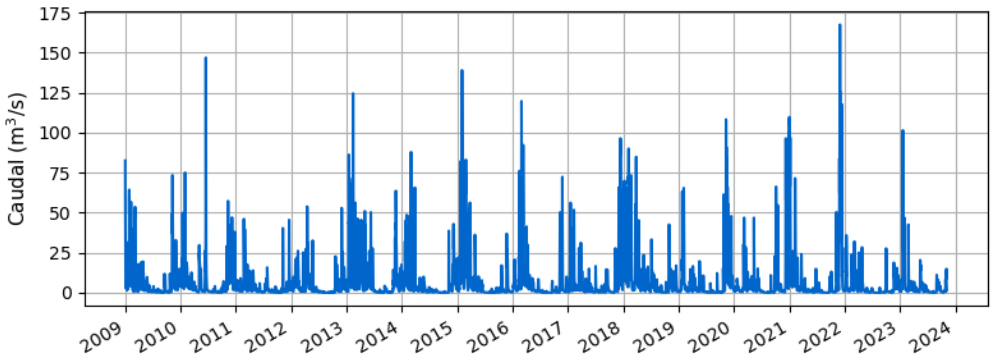


Figura 6. Serie temporal de caudales diarios usada en el TFM.

El aporte de otros cursos fluviales es, a su vez, despreciable frente al caudal del río Miera. Las rías de Boó, Tijero y San Salvador, por ejemplo, son de agua salada y su aporte es mayoritariamente pluvial.

El viento, por su parte, también es muy variable, tanto en intensidad como en dirección, lo cual puede apreciarse en la rosa de los vientos de la Figura 7. Su influencia en las corrientes será importante en las capas más superficiales, que son las que se han analizado en el TFM debido a su relevancia en el análisis de episodios de contaminación marina. En la misma línea, el efecto será considerable en zonas de menor profundidad y durante la bajamar, ya que la fricción ejercida se concentra en la parte superior de la columna de agua.

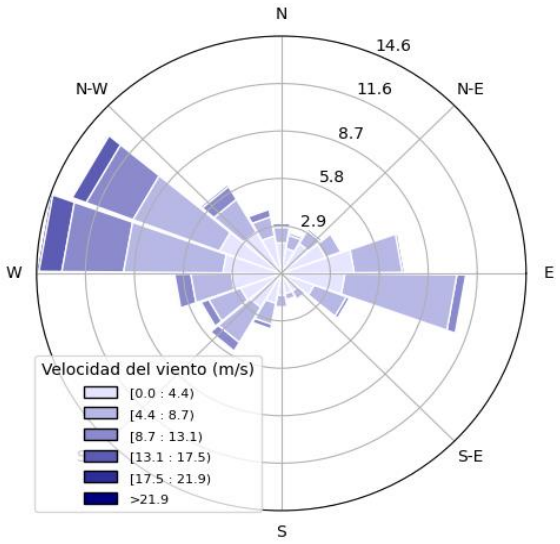


Figura 7. Rosa de los vientos para la Bahía de Santander. Fuente: elaboración propia con base en información obtenida de la base de datos SeaWindNCEP-HR (Acevedo et al., 2019).

Si bien podrían existir otros forzamientos, se asume que su influencia es baja o sus efectos son muy localizados, como es el caso del oleaje en la zona de la bocana. Estos elementos estarían en un tercer escalón en cuanto a importancia y, por consiguiente, se consideran poco relevantes para este estudio.

En consecuencia, los elementos forzadores de corrientes y nivel del mar en la Bahía de Santander que se han considerado en el análisis son:

- marea astronómica;
- caudal del río Miera;

- intensidad y dirección del viento.

Por otro lado, es importante recordar que el objetivo de las técnicas de aprendizaje automático es relacionar las condiciones fuera de la Bahía, a escala regional, con condiciones locales en el interior de la misma. Por lo tanto, también es necesario incluir como *inputs* del modelo:

- datos de nivel del mar a escala regional;
- datos de corrientes a escala regional.

4.2.1 SELECCIÓN DE PREDICTORES

A partir de la identificación de forzamientos actuantes, se han seleccionado los predictores que se presentan en la Tabla 1.

Tabla 1. Selección de predictores.

	Forzamiento	Predictor	Descripción
1	**Marea astronómica**	tidalRange	Carrera de marea
2	**Nivel del mar a gran escala**	zos	Nivel del mar
3	**Corriente a gran escala**	ubar	Componente zonal de la velocidad barotrópica
4		vbar	Componente meridional de la velocidad barotrópica
5	**Viento**	u_10	Componente zonal de la velocidad a 10 m
6		v_10	Componente meridional de la velocidad a 10 m
7	**Presión atmosférica**	slp	Presión a nivel del mar
8	**Caudal del río Miera**	discharge	Caudal del río

La marea astronómica ha decidido caracterizarse a través de la carrera de marea, que es la diferencia en altura entre el nivel de marea alta y el de marea baja.

Por otro lado, tanto la corriente como el viento son magnitudes vectoriales, en este caso simplificadas a dos dimensiones. En consecuencia, ambos se han caracterizado mediante sus componentes zonales (dirección oeste-este) y meridionales (dirección sur-norte). Se ha considerado el viento a una altura de 10 m, y la presión atmosférica a nivel del mar.

Para explicar las corrientes a gran escala, fuera de la Bahía, se ha seleccionado la velocidad barotrópica, que podría describirse de manera simplificada como la velocidad promediada en profundidad.

4.2.2 RECOPILACIÓN DE BASES DE DATOS

Una vez determinados los predictores, se ha realizado una recopilación de las bases de datos disponibles de las variables océano-meteorológicas correspondientes. En todos los casos, se ha considerado:

- el tipo de producto (reanálisis, *hindcast*, observación o *forecast*);
- la extensión y resolución espacial;
- la extensión y resolución temporal;
- las variables disponibles;
- la frecuencia de actualización;
- y la forma de acceso a los datos (API, Thredds, OpenDAP, etc.)

Los resultados de esta recopilación pueden consultarse en la carpeta *annex* del repositorio GitHub del proyecto (Rupani, 2024). En la Tabla 2 se presenta un resumen de las bases de datos seleccionadas.

Tabla 2. Bases de datos seleccionadas.

	Proveedor	Base de datos	Producto	Predictor
1	**Puertos del Estado**	Nivel del mar Santander	-	tidalRange
2	**Copernicus Marine Service**	Atlantic-Iberian Biscay Irish- Ocean Physics Analysis and Forecast	cmems_mod_ibi_phy_anfc_0.027deg-2D_PT1H-m	zos
3				ubar
4				vbar
5	**Metogalicia**	WRF_HIST/d02		u_10
6				v_10
7				slp
8	**IHCantabria**	Estudio de Recursos Hídricos de las Cuencas de la Vertiente Norte de Cantabria		discharge

4.3 ANÁLISIS DE VARIABILIDAD

Excluyendo la marea astronómica debido a su naturaleza determinista, se ha examinado la variabilidad de los predictores, considerando las extensiones de las bases de datos disponibles. Se han considerado dos años de datos de las variables predictoras. De esta forma, se tiene en cuenta la variabilidad estacional y la variabilidad interanual, al menos con dos años de datos.

La selección de los dos años se ha realizado considerando el caudal del río Miera, por ser la variable que presenta mayor variabilidad interanual. Al tratarse de un río, se ha realizado un análisis considerando años hidrológicos, que es un periodo de 12 meses en el que se miden las precipitaciones sobre una determinada cuenca hidrográfica. En España, el año hidrológico comienza el 1 de octubre, y termina el 30 de septiembre, de acuerdo con el Ministerio para la Transición Ecológica y el Reto Demográfico.

Por un lado, hay años que resultan mucho más húmedos que otros. Por este motivo, se ha analizado la aportación anual (volumen total anual). Por otro lado, hay años en los que, durante las avenidas, se alcanzan valores de caudal mucho más elevados que otros. Estos podrían coincidir con los años más húmedos (en términos de precipitaciones), pero no tiene por qué ser así. También se ha analizado la serie total de caudales diarios. Los resultados se presentan en la Figura 8 y la Figura 9.

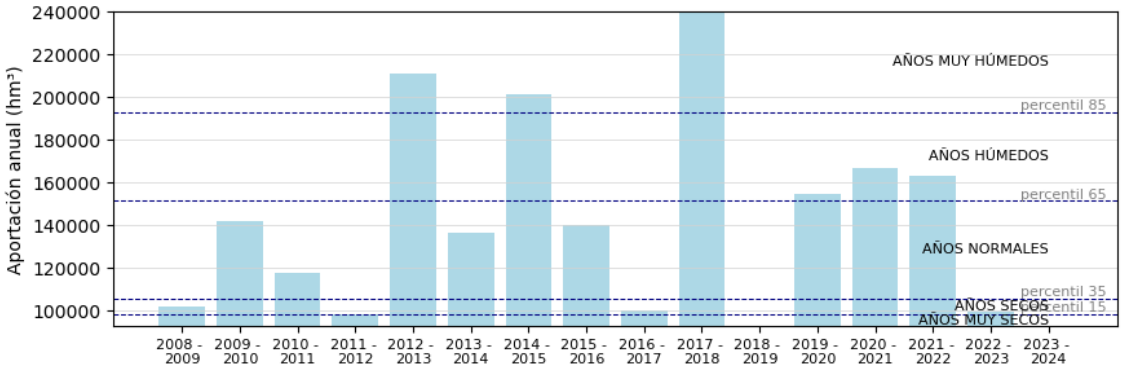


Figura 8. Aportación total anual por cada año hidrológico y clasificación en años muy secos, secos, normales, húmedos y muy húmedos. Para la clasificación, se han usado como límites los percentiles 15, 35, 65 y 85.

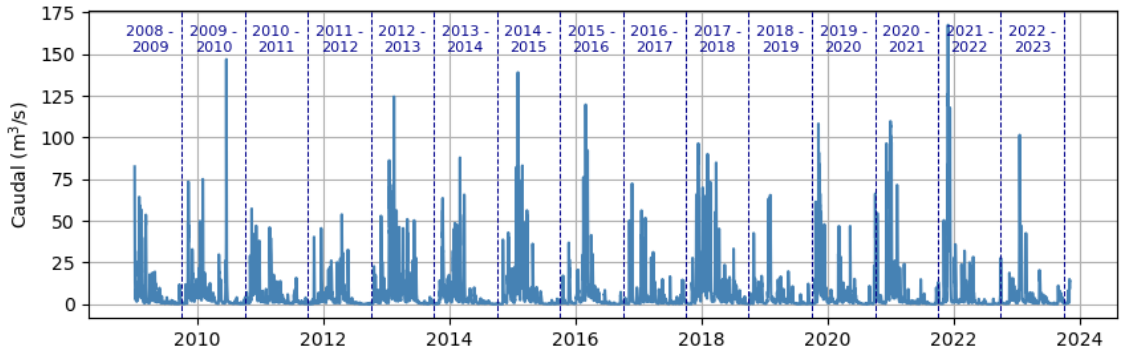


Figura 9. Serie de caudales diarios para cada año hidrológico.

El objetivo del análisis ha sido el de encontrar un periodo de la menor duración posible, en el que se pueda explicar la máxima variabilidad posible en términos de caudal. Así, se ha decidido considerar periodos de dos años, lo cual es un lapso razonable para modelar numéricamente y obtener los predictandos y, al mismo tiempo, para que la variabilidad de los predictores (y, en consecuencia, también la de los predictandos) esté suficientemente bien representada.

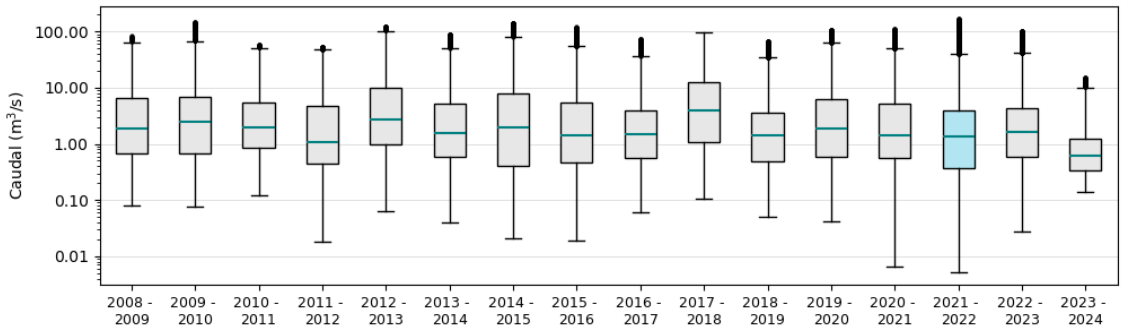


Figura 10. Diagrama de cajas (boxplot) de los caudales diarios para cada año hidrológico. Se ha optado por una escala logarítmica en el eje y, y los bigotes se extienden hasta 10 veces el rango intercuartílico (IQR), para representar mejor la variabilidad (durante las avenidas, pueden darse valores de caudal muy elevado).

Por tanto, se han seleccionado finalmente los años hidrológicos 2021-2022, y 2022-2023. De este modo, se obtienen un año húmedo (2021-2022) y un año seco (2022-2023) en términos de aportación total; y, simultáneamente, el año 2021-2022 cubre todo el espectro de variabilidad en cuanto a caudales diarios (Figura 10).

4.4 MODELADO NUMÉRICO

Con los dos años seleccionados, se ha realizado una simulación numérica utilizando el modelo de cálculo en aguas someras Delft3D-*FLOW* (Roelvink & Van Banning, 1995) cuyas características se detallan en el Manual de Usuario (Deltares, 2021). El objetivo del modelado es el de obtener resultados horarios de corrientes y nivel del mar, a utilizar como predictandos de las técnicas estadísticas a entrenar.

A la hora de ejecutar el modelo Delft3D-*FLOW*, además de los datos de entrada (forzamientos), resulta necesario definir el dominio de cálculo, discretizarlo, definir las condiciones de contorno y las condiciones iniciales, y llevar a cabo el proceso de calibración y validación. Además, es preciso determinar el tiempo necesario para que el modelo se estabilice (periodo de calentamiento). Para ello, deben realizarse simulaciones con diferentes forzamientos, y evaluar en puntos característicos del estuario el tiempo requerido para obtener valores estacionarios de las variables de salida.

Se ha establecido un periodo de calentamiento de un mes. Para agilizar la simulación, se ha paralelizado la simulación en 4 periodos de 7 meses (6 meses efectivos, más uno de calentamiento en cada caso), que se han lanzado en el *cluster* de supercomputación Neptuno perteneciente al Instituto de Hidráulica Ambiental de la Universidad de Cantabria (IHCantabria). El tiempo de cómputo de cada caso ha sido de entre 7 y 10 días.

Si bien el modelado numérico se ha realizado *ad hoc* para el presente TFM, su ejecución ha sido asumida por miembros del grupo de investigación de Oceanografía, Estuarios y Calidad del Agua de IHCantabria.

4.5 IMPLEMENTACIÓN CÓDIGO

Todo el código desarrollado en el TFM se ha implementado utilizando el lenguaje de programación Python. La elección se debe a su amplio uso en el ámbito de la ciencia de datos y el aprendizaje automático, así como a la disponibilidad de una gran variedad de librerías de código abierto que facilitan el desarrollo y la implementación de modelos complejos. Se ha seguido una metodología meticulosa y rigurosa, con el objetivo de garantizar tanto la eficiencia del código como la reproducibilidad del trabajo realizado.

Las librerías principales utilizadas incluyen:

- Numpy (Harris et al., 2020): para operaciones numéricas y manejo eficiente de arreglos y matrices.
- Pandas (McKinney, 2010): para manipulación y análisis de datos estructurados.
- Scipy (Virtanen et al., 2020): para cálculos científicos y técnicos avanzados.
- Matplotlib (Hunter, 2007), Plotly y Seaborn (Waskom, 2021): para la visualización de datos y la generación de gráficos.
- Scikit-learn (Pedregosa et al., 2011): para el desarrollo y la evaluación de modelos de aprendizaje automático.

- Keras (Chollet & others, 2015), con backend TensorFlow: para la creación y el entrenamiento de redes neuronales.

La implementación del código ha seguido el paradigma de la programación orientada a objetos (POO), lo que permite una estructura más modular y reusable, facilitando su extensión y mantenimiento. Para asegurar la consistencia en el estilo de codificación, se ha utilizado la convención lowerCamelCase para nombrar variables, *scripts* y demás elementos, exceptuando los casos en los que se llama a librerías externas que no utilizan esa convención.

Además, para asegurar un control riguroso sobre las versiones del código, y en un futuro facilitar la colaboración, se ha utilizado Git como sistema de control de versiones. Se tiene planeado implementar *Semantic Versioning* (Preston-Werner, 2013) en caso de que el proyecto continúe, lo cual permitirá una gestión más clara y estructurada de las versiones del *software*.

Todo el código desarrollado, junto con ejemplos de casos, documentación (*readme files*), y otros recursos relevantes, se encuentra disponible en un repositorio de GitHub (Rupani, 2024). Este repositorio es de acceso libre y abierto, siguiendo los principios FAIR (*findable, accessible, interoperable, reusable*), garantizando que cualquier persona pueda acceder, reutilizar y contribuir al proyecto.

En el futuro, si se generan *datasets* de relevancia, estos se pondrán a disposición en repositorios abiertos como Zenodo, asegurando así que los datos también cumplan con los principios FAIR.

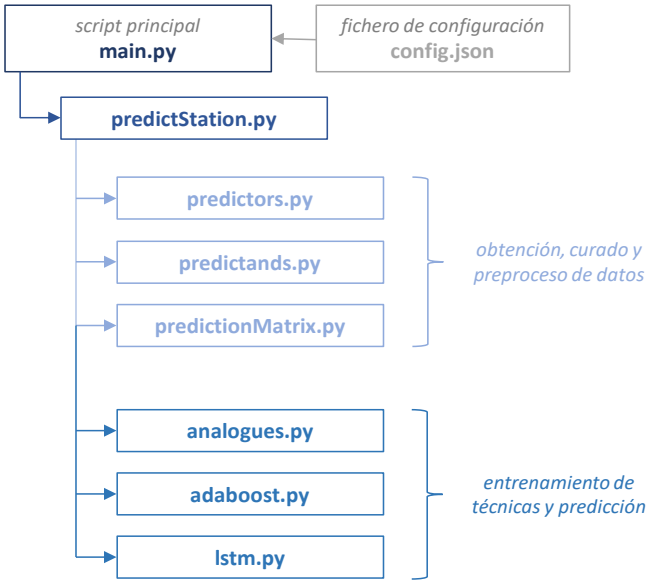


Figura 11. Estructura del código implementado.

La estructura del proyecto está diseñada para maximizar la flexibilidad y la facilidad de uso. Todo el código se ejecuta a partir de un único *script* principal (main.py), que toma sus parámetros de un fichero de configuración (config.json). Este fichero JSON contiene todas las variables y parámetros que pueden ser ajustados para diferentes casos de estudio. De este modo, no es necesario modificar el código directamente para adaptarlo a diferentes escenarios; cualquier usuario puede simplemente modificar el fichero de configuración y ejecutar el *script* principal para llevar a cabo su propio análisis.

En la Figura 11 se presenta un diagrama que esquematiza la estructura del código implementado, destacando la relación entre los diferentes componentes y su integración en el flujo de trabajo. Esta estructura modular y clara permite una fácil comprensión y modificación del código, facilitando su extensión para futuros trabajos y aplicaciones.

4.5.1 OPTIMIZACIÓN DE HIPERPARÁMETROS

La optimización de hiperparámetros es un proceso fundamental en el desarrollo de modelos de aprendizaje automático. Los hiperparámetros son aquellos parámetros que no se aprenden directamente del proceso de entrenamiento del modelo, sino que deben ser establecidos de antemano. Ejemplos comunes de hiperparámetros incluyen la tasa de aprendizaje, el número de árboles en un modelo de bosques aleatorios, o el número de capas y neuronas en una red neuronal.

La importancia de la optimización de hiperparámetros radica en su capacidad para mejorar significativamente el rendimiento del modelo. Un conjunto de hiperparámetros bien escogido puede hacer que el modelo sea más preciso, eficiente y generalizable a datos no vistos. Por el contrario, una mala elección de hiperparámetros puede llevar a un rendimiento deficiente, sobreajuste o infraajuste del modelo.

El proceso de optimización de hiperparámetros implica la exploración de diferentes combinaciones de estos para encontrar la configuración que maximiza alguna métrica de rendimiento. Existen diferentes métodos para llevar a cabo esta búsqueda. A continuación, se detallan los implementados para cada técnica de aprendizaje automático evaluada.

4.5.1.1 PREDICCIÓN CONDICIONADA A ANÁLOGOS

Para el método de análogos, no se ha implementado una optimización automatizada de hiperparámetros. En su lugar, se han explorado manualmente diferentes configuraciones para determinar la configuración que mejor se adapte a los datos y objetivos. Este enfoque manual se justifica por la naturaleza específica del método de análogos, donde la selección de parámetros puede depender en gran medida del conocimiento experto y las características particulares del conjunto de datos.

Los hiperparámetros que se han optimizado manualmente son:

- *nAnalogues*: número de análogos (*i.e.* centroides) considerados.
- *clustering*: método de agrupamiento utilizado para identificar análogos (K-*means*, MaxDiss o *spectral clustering*).
- *regressor*: tipo de regresor utilizado después de identificar los análogos (KNN o KRR).
- Hiperparámetros propios de KNN
    - *n_neighbors*: número de vecinos más cercanos considerados.
    - *weights*: peso asignado a los vecinos (p. ej., "*distance*").
    - *metric*: métrica utilizada para calcular la distancia hasta los vecinos cercanos (p. ej., "minkowski").
- Hiperparámetros propios de KRR
    - *kernel*: tipo de *kernel* utilizado (por ejemplo, "rbf").

- *alpha*: parámetro de regularización.
- *gamma*: parámetro específico para *kernels* RBF, *laplacian*, *polynomial*, *exponential chi2*, y *sigmoidADABOOST*. La interpretación es específica de cada *kernel*.

4.5.1.2 ADABOOST

Para el modelo AdaBoost, se ha utilizado GridSearchCV de la biblioteca scikit-learn. GridSearchCV es una técnica de búsqueda de hiperparámetros que permite explorar sistemáticamente una cuadrícula de combinaciones de hiperparámetros y evalúa cada combinación utilizando validación cruzada. Este enfoque garantiza que se prueben todas las posibles combinaciones de los hiperparámetros especificados, y se selecciona la combinación que maximiza el rendimiento del modelo según una métrica de evaluación específica.

A continuación, se describen los hiperparámetros que se han configurado para optimizar en el modelo AdaBoost:

- *estimator* (estimador): cada estimador tiene sus hiperparámetros (propios de cada árbol de regresión), que también se han optimizado:
  - *maxDepth* (profundidad máxima): este hiperparámetro se refiere a la profundidad máxima de cada árbol de decisión utilizado como estimador base en el modelo AdaBoost. La profundidad del árbol controla hasta qué nivel de detalle el árbol puede aprender los datos. Valores mayores permiten que el árbol aprenda interacciones más complejas, pero también pueden llevar a sobreajuste. Las técnicas de *boosting* se basan en combinar modelos débiles (*weak models*), por lo que suelen usarse profundidades pequeñas.
  - *criterion* (criterio): indica cómo evaluar la calidad de una división en los árboles de decisión, y así determinar la mejor división en cada nodo. Valores posibles: {*"squared_error", "friedman_mse", "absolute_error", "poisson"*}.
  - *minSamplesSplit* (mínimo de muestras por división): este hiperparámetro indica el número mínimo de muestras requeridas para dividir un nodo. Un valor mayor previene la creación de nodos con pocas muestras.
  - *minSamplesLeaf* (mínimo de muestras por hoja): este hiperparámetro define el número mínimo de muestras que debe tener un nodo hoja. Al igual que el parámetro anterior, ayuda a controlar el sobreajuste asegurando que las hojas no sean demasiado pequeñas.
- *nEstimators* (número de estimadores): este hiperparámetro controla el número de árboles de decisión que se entrenan y ensamblan en el modelo AdaBoost. Un número mayor de estimadores puede mejorar el rendimiento del modelo al reducir el sesgo, pero también puede aumentar el riesgo de sobreajuste y el tiempo de entrenamiento.
- *learningRate* (tasa de aprendizaje): la tasa de aprendizaje es un factor de escalado que se aplica a las contribuciones de cada árbol adicional en el modelo AdaBoost. Una tasa de aprendizaje más baja puede hacer que el modelo sea más robusto al ruido, pero puede requerir más estimadores para alcanzar un buen rendimiento.

- *loss* (función de pérdida): especifica la función de pérdida que el modelo AdaBoost debe minimizar. Se han considerado las funciones de pérdida "*square*", "*exponential*" y "*linear*", que afectan la manera en que se ponderan los errores durante el entrenamiento.

La métrica de evaluación utilizada para seleccionar la mejor combinación de hiperparámetros ha sido el error cuadrático medio negativo (*neg_mean_squared_error*) en todos los casos. Esta métrica evalúa el rendimiento del modelo penalizando los errores de predicción de manera cuadrática, lo que pone mayor peso en errores más grandes.

4.5.1.3 RNN-LSTM

En el caso de las redes neuronales recurrentes LSTM, la optimización de hiperparámetros se ha implementado a través de la librería KerasTuner (O'Malley et al., 2019), diseñada para facilitar la búsqueda de hiperparámetros óptimos en modelos de aprendizaje profundo creados con Keras y TensorFlow. Se utiliza para ajustar hiperparámetros tales como la cantidad de neuronas en cada capa, el número de capas, la tasa de aprendizaje, entre otros. Optimizar estos hiperparámetros manualmente puede ser un proceso tedioso y propenso a errores, especialmente en modelos complejos. KerasTuner facilita este proceso automatizándolo y proporcionando una búsqueda más estructurada y eficiente. Los pasos generales de la búsqueda son:

1. Definición del modelo: se define una función que construye y compila el modelo de Keras, incluyendo los hiperparámetros a ajustar.
2. Configuración del espacio de búsqueda: se define el rango de valores (o los valores discretos) posibles para cada hiperparámetro.
3. Selección del *tuner*: se selecciona un método específico para explorar el espacio de búsqueda.
4. Ejecución del *tuner*: se prueban diferentes combinaciones de hiperparámetros y se evalúa el rendimiento del modelo en un conjunto de validación (para elegir el mejor modelo).

Cómo se determinan las combinaciones y cómo se prueban cada una de ellas, depende del *tuner* seleccionado. KerasTuner ofrece las siguientes opciones:

- RandomSearch *Tuner*: realiza simplemente una búsqueda aleatoria en el espacio de hiperparámetros. Es útil cuando se tiene un espacio de búsqueda grande y se desea una búsqueda rápida y general.
- GridSearch *Tuner*: explora exhaustivamente todas las combinaciones posibles de hiperparámetros en una cuadrícula definida. Es útil cuando se tiene un espacio de búsqueda pequeño y bien definido (si no, se vuelve inviable por su coste computacional).
- BayesianOptimization *Tuner*: utiliza optimización bayesiana para seleccionar los hiperparámetros. Es más eficiente que la búsqueda aleatoria y la búsqueda en cuadrícula, ya que intenta encontrar las mejores combinaciones de hiperparámetros en menos iteraciones.
- Hyperband *Tuner*: utiliza una estrategia de búsqueda basada en bandas hiperparamétricas (*hyperbands*). Evalúa muchas configuraciones de hiperparámetros rápidamente y elimina las menos prometedoras de manera progresiva.

Se ha optado por utilizar el Hyperband Tuner, ya que es extremadamente eficiente en términos de tiempo y recursos computacionales. Resulta de 5 a 30 veces más rápido que los algoritmos de optimización bayesiana en una variedad de problemas de aprendizaje profundo y aprendizaje basado en *kernel* (Li et al., 2018). KerasTuner implementa el algoritmo desarrollado por Li et al. (2018), que formula la optimización de hiperparámetros como un problema de asignación adaptativa de recursos, determinando cómo distribuir recursos entre configuraciones de hiperparámetros muestreadas aleatoriamente.

4.5.2 APLICACIÓN

Se ha llevado a cabo la aplicación de la metodología para reconstruir la hidrodinámica en puntos que se consideran representativos de toda la Bahía. Utilizando los resultados obtenidos para estos puntos representativos, se ha realizado un análisis comparativo de las técnicas de modelado implementadas.

4.6 ANÁLISIS COMPARATIVO DE TÉCNICAS

4.6.1 SELECCIÓN DE MÉTRICAS

Para llevar a cabo un análisis comparativo exhaustivo de las diferentes técnicas de modelado implementadas, se ha seleccionado un conjunto de métricas que permiten evaluar múltiples aspectos de las predicciones realizadas. Se describen a continuación.

- *Mean Absolute Error* (MAE): mide el promedio de los errores absolutos entre las predicciones del modelo y los valores observados. Es una métrica sencilla que proporciona una idea clara de la magnitud promedio del error.

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i|$$

donde:

y_i son los valores observados.

$\hat{y}_i$ son los valores predichos por el modelo.

n es el número total de observaciones.

El MAE se interpreta como la magnitud promedio del error en las predicciones, con valores más bajos indicando un mejor ajuste del modelo a los datos observados.

- *Bias*: indica la tendencia promedio de las predicciones del modelo a ser mayores o menores que los valores observados. Un *bias* positivo indica una sobreestimación del modelo, mientras que un *bias* negativo indica una subestimación.

$$\text{Bias} = \frac{1}{n}\sum_{i=1}^{n}(\hat{y}_i - y_i)$$

- Pearson *Correlation Coefficient*: evalúa la correlación lineal entre las predicciones del modelo y los valores observados. El coeficiente de Pearson varía entre -1 y 1, donde 1 indica

una correlación perfecta positiva, -1 una correlación perfecta negativa, y 0 indica que no hay correlación.

$$r = \frac{\sum_{i=1}^{n}(y_i - \overline{y})(\hat{y}_i - \overline{\hat{y}})}{\sqrt{\sum_{i=1}^{n}(y_i - \overline{y})^2}\sqrt{\sum_{i=1}^{n}(\hat{y}_i - \overline{\hat{y}})^2}}$$

donde:

$\overline{y}$ es el valor promedio de los observados.

$\overline{\hat{y}}$ es el valor promedio de las predicciones.

- Willmott *Skill Index*: este índice, propuesto por Willmott (1981), refleja la destreza con la que las variables reconstruidas estiman las variables originales. Es ampliamente utilizado para evaluar la precisión en modelos oceanográficos. Un valor del índice igual a 1 indica una concordancia perfecta entre predicción y observación, mientras que un valor de 0 indica una disimilitud total. La fórmula del índice es:

$$s = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(|y_i - \overline{y}| + |\hat{y}_i - \overline{y}|)^2}$$

- Kolmogorov-Smirnov *Statistic* (Kolmogorov, 1933): es un *test* no paramétrico que compara la igualdad de distribuciones de probabilidad continuas (o discontinuas). El estadístico de Kolmogorov-Smirnov cuantifica la distancia máxima entre la función de distribución acumulativa empírica de dos variables (en este caso, predicción y observación). La fórmula del estadístico es:

$$D_n = max|F_n(x) - F(x)|$$

donde:

$F_n(x)$ es la función de distribución de la predicción y

$F(x)$ es la función de distribución de la observación.

- Perkins *Score* (Perkins et al., 2007): mide la similitud entre dos funciones de densidad de probabilidad (PDFs), en este caso correspondientes a la predicción y la observación. Un valor de 1 para este índice indicaría un solape perfecto entre ambas PDFs, mientras que un valor de 0 indicaría que ambas PDFs no solapan en absoluto.

4.6.2 CONSIDERACIONES DEL ANÁLISIS

El análisis comparativo entre las diferentes técnicas se basa no solo en el desempeño cuantitativo, evaluado a través de las métricas descritas, sino también en la eficiencia de los recursos computacionales necesarios para el entrenamiento y la predicción.

En concreto, se ha llevado un registro detallado de los recursos computacionales utilizados por cada técnica, incluyendo el tiempo de procesamiento, el uso de memoria y la eficiencia en el uso de otros recursos del sistema. Este aspecto es crucial para determinar la viabilidad práctica de implementar cada técnica en aplicaciones reales, especialmente en escenarios donde los recursos pueden ser limitados.

5 DATOS

En este apartado, se presentan los datos utilizados en el desarrollo del TFM, que se detallan en la Tabla 3. Se describirán cada uno de los conjuntos de datos, comenzando por los predictores y continuando con los predictandos. En la sección de predictores, también se presentan las técnicas de preproceso aplicadas a los datos.

Tabla 3. Datos utilizados para el desarrollo del TFM.

Nivel del mar	
Proveedor	Puertos del Estado
Variable	carrera de marea
Ubicación	mareógrafo de Santander (lat. 43.466, lon. -3.79)
Resolución temporal	horaria
Resolución espacial	puntual
IBI Ocean Analysis and Forecast	
Proveedor	CMEMS (*Copernicus Marine Environment Monitoring Service*)
Ubicación	punto más cercano
Dataset	cmems_mod_ibi_phy_anfc_0.027deg-2D_PT1H-m
Variables	velocidad barotrópica zonal velocidad barotrópica meridional nivel del mar
Resolución temporal	horaria
Resolución espacial	1/36° (aproximadamente 2-3 km)
WRF d02	
Proveedor	Meteogalicia
Variables	viento a 10 m zonal viento a 10 m meridional presión atmosférica (nivel del mar)
Ubicación	punto más cercano
Resolución temporal	horaria
Resolución espacial	12 km
Caudal Río Miera	
Proveedor	IHCantabria
Variable	caudal
Ubicación	límite de influencia mareal del río Miera
Resolución temporal	diaria
Resolución espacial	puntual
Modelado hidrodinámico	
Proveedor	IHCantabria
Variables	velocidad superficial zonal velocidad superficial meridional nivel del mar temperatura salinidad
Ubicación	toda la extensión de la malla
Resolución temporal	horaria
Resolución espacial	variable (10 a 900 m aprox.)

5.1 PREDICTORES

5.1.1 NIVEL DEL MAR

Para el análisis del nivel del mar, se han utilizado datos provenientes de Puertos del Estado, específicamente del mareógrafo de Santander, perteneciente a REDMAR (Puertos del Estado, 2020). El conjunto de datos REDMAR está formado por las medidas procedentes de la Red de Mareógrafos de Puertos del Estado. Esta red tiene como finalidad medir, grabar, analizar y almacenar de forma continua el nivel del mar en los puertos, permitiendo el acceso a los datos en tiempo real. Las estaciones más antiguas de la red proporcionan datos desde julio de 1992. Actualmente, la red cuenta con más de 30 estaciones en funcionamiento distribuidas a lo largo de la costa española.

En términos de parámetros disponibles, REDMAR ofrece datos de nivel del mar en series de 5 minutos y horarias, residuo meteorológico horario, niveles extremos diarios, mensuales y anuales, niveles medios diarios, mensuales y anuales, carreras de marea y constantes armónicas anuales y promedio. Para este TFM, se han utilizado datos horarios de carrera de marea, abarcando un periodo de dos años correspondiente al modelado numérico generado para obtener los predictandos (véase la Sección 4.3 ANÁLISIS DE VARIABILIDAD). En la Figura 12 se muestra una gráfica de la serie temporal.

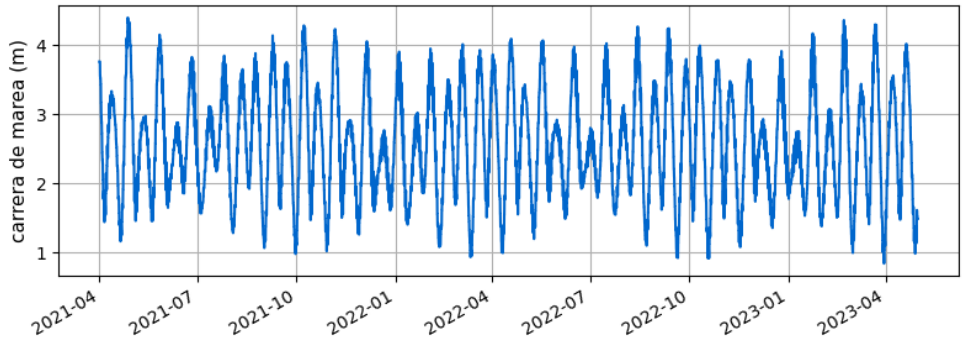


Figura 12. Serie temporal de carrera de marea usada en el TFM.

5.1.2 IBI OCEAN ANALYSIS AND FORECAST

El IBI-MFC *Iberia–Biscay–Ireland Monitoring and Forecasting Centre* (CMEMS Marine Data Store) es un producto de CMEMS (*Copernicus Marine Environment Monitoring Service*) que se ejecuta diariamente con el apoyo de Nologin y el CESGA (Centro de Supercomputación de Galicia) en términos de recursos de computación. Cubre las aguas europeas, y más específicamente, el área de la Península Ibérica, el Golfo de Vizcaya e Irlanda (IBI). El catálogo incluye tanto las mejores estimaciones históricas de los últimos 2 años como pronósticos de diferentes resoluciones temporales con un horizonte de 5 días, actualizados diariamente. El dominio del modelo cubre desde 19°W a 5°E y desde 26°N a 56°N, definido en una malla regular con resolución de 1/36 de grado (aproximadamente 2-3 km), resultando en 1081 x 865 puntos, y 50 niveles verticales.

Para el TFM, se ha empleado el *dataset* cmems_mod_ibi_phy_anfc_0.027deg-2D_PT1H-m, que incluye la velocidad barotrópica zonal y meridional, y la altura de la superficie del mar, con una resolución temporal horaria. Se han utilizado los datos correspondientes al punto

disponible de la malla más cercano al punto que se desea predecir. En la Figura 13, se presentan series temporales de ambas componentes de la velocidad barotrópica, a modo de ejemplo.

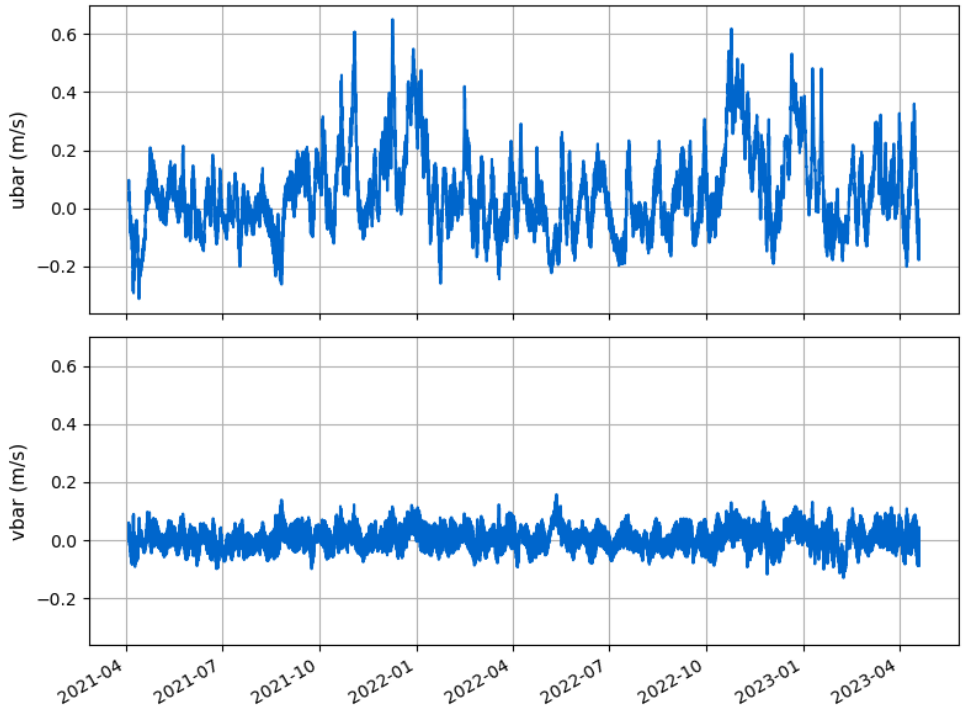


Figura 13. Series temporales de cada componente de la velocidad barotrópica de la corriente usadas en el TFM.

5.1.3 WRF D02 METEOGALICIA

Este *dataset* proviene del modelo atmosférico WRF (*Weather Research and Forecasting*) operado por MeteoGalicia. MeteoGalicia es el servicio meteorológico regional dependiente de la Consellería de Medio Ambiente, Territorio e Infraestruturas de la Xunta de Galicia. Este organismo realiza predicciones diarias mediante la ejecución de varios modelos numéricos meteorológicos y oceanográficos, además de disponer de una extensa red de observación meteorológica y oceanográfica.

El modelo WRF de MeteoGalicia integra observaciones meteorológicas recogidas en puntos distribuidos por la región de estudio, así como estimaciones numéricas generadas por otros modelos de predicción de alcance global y menor resolución espacial. Se utiliza como *input* el modelo *Global Forecast System* (GFS) de la *National Oceanic and Atmospheric Administration* (NOAA). Cuenta con tres dominios anidados de simulación, con resoluciones espaciales de 36 km, 12 km y 4 km. Para el TFM, se ha utilizado el dominio de 12 km (d02), que abarca toda la Península Ibérica y el Golfo de Vizcaya.

Las predicciones proporcionadas por el modelo incluyen variables atmosféricas en tres dimensiones, tales como temperatura, viento, precipitación, humedad, presión, flujos térmicos y flujos de radiación, entre otros. Para el presente trabajo, se han utilizado los datos de viento a 10 metros (componentes zonal y meridional, que se muestran en la Figura 14) y la presión atmosférica, obtenidos del punto de la malla más cercano al punto de predicción.

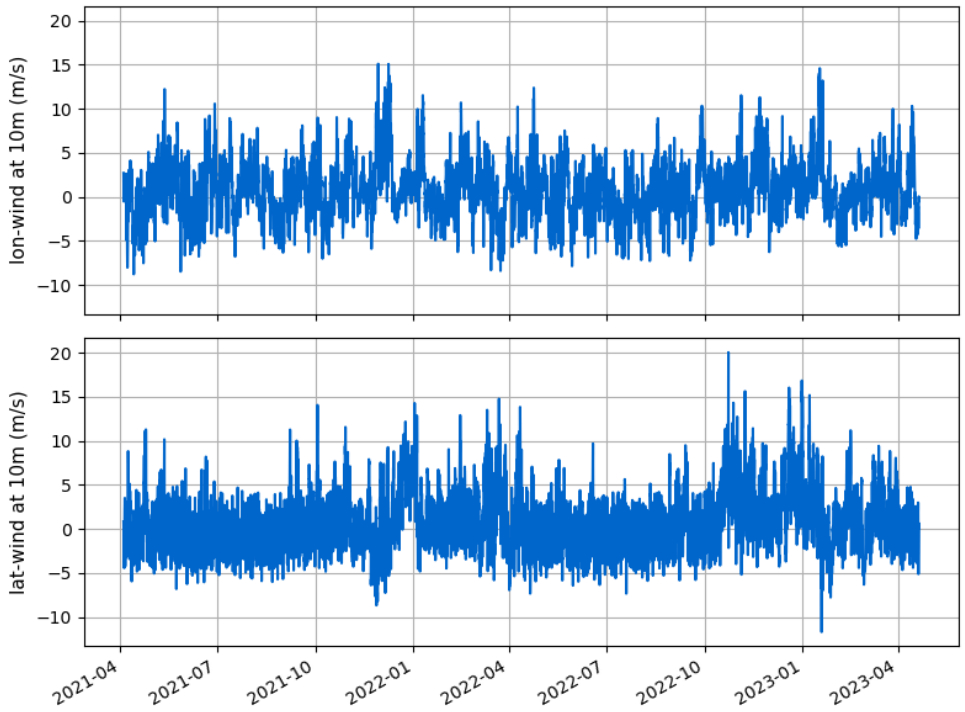


Figura 14. Series temporales de ambas componentes de la velocidad del viento a 10 m, usadas en el TFM.

5.1.4 CAUDAL DEL MIERA

Se han utilizado datos de caudal del río Miera que han sido obtenidos por IHCantabria mediante la aplicación de un modelo de equilibrio logístico (*logistic equilibrium model*, LEM), que se describe en Díaz et al. (2015), a datos de precipitaciones de Meteogalicia. El modelo se ha calibrado y validado en el periodo 2008-2012 con la información generada a través del modelo hidrológico HEC-HMS en el marco del "Estudio de Recursos Hídricos de las Cuencas de la Vertiente Norte de Cantabria Periodo 1987-2012", realizado por IHCantabria dentro del convenio Investigación, Desarrollo e Innovación en Materia de Agua (2012-2013) con la Consejería de Medio Ambiente, Ordenación del Territorio y Urbanismo del Gobierno de Cantabria. Las coordenadas de los datos corresponden aproximadamente con el límite de influencia mareal del río Miera.

Los datos tienen resolución temporal diaria. Para ajustar a la resolución horaria necesaria para entrenar los modelos, se ha realizado una interpolación lineal, tomando el dato diario como representativo de las 12 horas de cada día.

5.1.5 PREPROCESO

El preprocesamiento de los datos es una etapa crucial para asegurar que los modelos de predicción funcionen de manera eficiente. En este trabajo, se realiza en tres pasos principales: división de los datos en conjuntos de entrenamiento y *test*, escalado de los datos, y reducción de la dimensionalidad. Cada uno de estos pasos se detalla a continuación.

5.1.5.1 DIVISIÓN EN *TRAIN* Y *TEST*

El primer paso del preprocesamiento consiste en separar los datos en conjuntos de entrenamiento y prueba. Esto es esencial para contar con un subconjunto independiente que permita realizar la validación final del modelo. Dado que en este caso se trata con series

temporales, la división no puede realizarse de manera aleatoria. En su lugar, se utiliza una división secuencial, manteniendo la última parte del conjunto de datos para la validación. Esta división de datos no debe confundirse con el proceso de validación cruzada, que se presenta a continuación.

5.1.5.1.1 VALIDACIÓN CRUZADA

En el proceso de construcción y evaluación de modelos predictivos, es esencial garantizar que el modelo no solo se ajuste bien a los datos de entrenamiento, sino también que generalice adecuadamente a datos no vistos durante el mismo. Para esto, se utilizan diversas técnicas de validación cruzada, siendo *holdout* la opción más sencilla. Consiste en dividir el conjunto de datos en dos subconjuntos: uno para el entrenamiento y otro para la prueba (*test*). Esto es lo que se ha implementado en los métodos de análogos y RNN-LSTM.

La validación *leave-one-out* (LOO) es una técnica extrema de validación cruzada. En LOO, cada observación del conjunto de datos se utiliza una vez como conjunto de prueba y todas las demás observaciones se utilizan como conjunto de entrenamiento. Así, maximiza el tamaño del conjunto de entrenamiento y proporciona una estimación imparcial del rendimiento del modelo (no depende de cómo se ha realizado la división en *train* y *test*). No obstante, es impracticable en la mayoría de los casos debido a su elevado coste computacional (debe entrenarse un modelo por cada dato).

La validación cruzada por *folds* (*K-fold cross-validation*) es una técnica que equilibra la necesidad de una evaluación robusta con la eficiencia computacional. En este método, el conjunto de datos se divide en K subconjuntos (*folds*) de aproximadamente igual tamaño. Se realizan K iteraciones, en las que uno de los K *folds* se utiliza como subconjunto de *test* y los K−1 *folds* restantes se utilizan como subconjunto de entrenamiento. Así, cada *fold* se utiliza exactamente una vez como subconjunto de *test*. Es computacionalmente más eficiente (se entrenan K modelos), y se ha comprobado que con un número relativamente bajo de *folds* (p. ej. K=5 o K=10) se aproxima a los resultados de *leave-one-out* (Kohavi, 1995).

Para la predicción de series temporales, no es adecuado utilizar una validación cruzada por *folds* estándar debido a la naturaleza secuencial de los datos. En su lugar, se instancia la clase TimeSeriesSplit de scikit-learn, que permite realizar la validación cruzada respetando el orden temporal de los datos. Este proceso es el que sea ha implementado para el modelo AdaBoost, y se ilustra en la Figura 15.

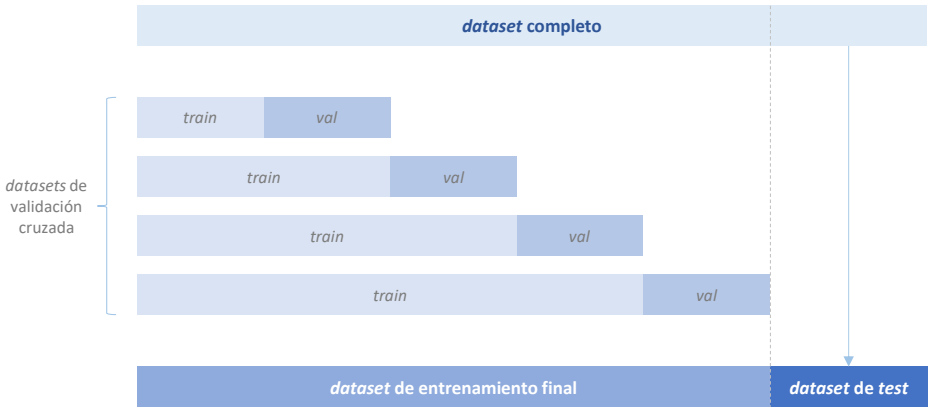


Figura 15. Esquema del proceso de validación cruzada implementado con AdaBoost.

En el caso de las redes LSTM, la validación cruzada está implementada internamente en Keras, donde los datos de validación (únicos) se pasan como un parámetro al método de entrenamiento de la red. Es decir, implementa una validación cruzada tipo *holdout*.

5.2 PREDICTANDOS

Como predictandos para el desarrollo de las técnicas de *downscaling* estadístico, se han utilizado los resultados del modelado numérico realizado *ad hoc* por IHCantabria, detallado en el apartado 4.4 MODELADO NUMÉRICO. En la Figura 16 puede apreciarse la extensión de la malla modelada.

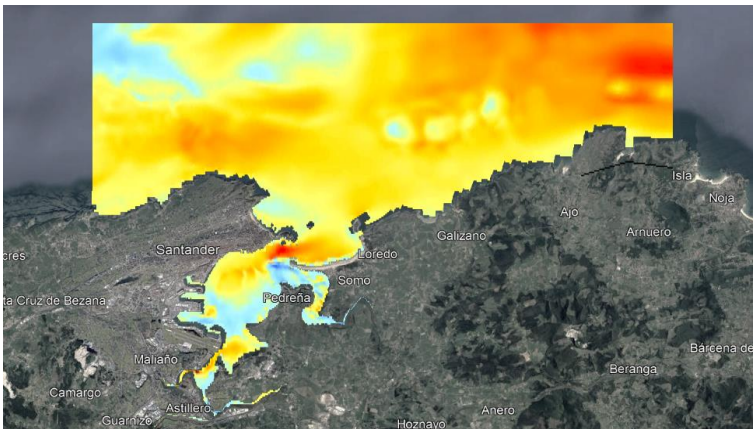


Figura 16. Ejemplo de resultado del modelado hidrodinámico (componente zonal de la velocidad).

Si bien se cuenta con datos para poder aplicar los métodos a todo el dominio de cálculo de la Figura 16, esto ha sido inviable dentro del alcance del TFM debido a dos limitaciones:

- Tamaño de los ficheros de salida espaciales: los resultados del modelado numérico generan ficheros espaciales de gran tamaño, prácticamente inmanejables. Son 4 ficheros de 195 GB cada uno, que solo pueden leerse con una *toolbox* específica para MATLAB de Deltares (por lo que requieren de mucho tiempo de preprocesamiento).
- Tiempos de entrenamiento: los tiempos necesarios para entrenar cada técnica también representan un desafío significativo. Aunque el método de análogos es relativamente rápido, tomando aproximadamente 30 segundos por punto, la optimización con *grid search* de AdaBoost y la optimización de las redes LSTM con Hyperband requieren considerables recursos computacionales y tiempo. Con los recursos informáticos con los que se ha contado para el TFM, el proceso de entrenamiento de un solo punto utilizando AdaBoost lleva más de 10 minutos, mientras que la optimización de las LSTM puede requerir varias horas.

Sin embargo, se ha encontrado una solución viable al utilizar un fichero de salida alternativo, más liviano, que contempla únicamente estaciones seleccionadas. Esta opción ofrece resultados puntuales mucho más manejables en términos de tamaño de archivo y tiempo de procesamiento. De esas estaciones, se han seleccionado 10 que resultan representativas de diferentes zonas de la Bahía, cada una con características distintas, como la zona exterior, la bocana, la canal de navegación, los bajos mareales y la desembocadura del río Miera. Estas estaciones fueron utilizadas para entrenar todos los modelos y se muestran en la Figura 17.

Figura 17. Estaciones de salida del modelado hidrodinámico. La numeración es simplemente identificativa.

En el caso de las redes LSTM, el principal desafío en cuanto a recursos computacionales reside en la optimización con Hyperband. Una vez seleccionados los hiperparámetros óptimos, el tiempo de entrenamiento de la red en sí se vuelve asumible, requiriendo tan solo unos pocos minutos. Por lo tanto, una alternativa viable sería entrenar una red en cada punto utilizando los hiperparámetros óptimos seleccionados para otro punto. Idealmente, esta estrategia se aplicaría en todo el espacio de interés. Para lograrlo, sería necesario clasificar todos los puntos del espacio según sus características (p ej., mediante técnicas de *clustering* de predictandos). Esto permitiría identificar los puntos representativos para los cuales ya se han optimizado los hiperparámetros con Hyperband.

Aunque esta clasificación aún no se ha realizado para todos los puntos del espacio, sí se ha llevado a cabo para las estaciones no seleccionadas. Inicialmente, la clasificación se ha realizado manualmente con criterio de experto, dada la cantidad asumible de puntos y las limitaciones de tiempo. Con el objetivo de facilitar el análisis de resultados, se ha implementado una interfaz de programación de aplicaciones (API). Esta herramienta interactiva permite seleccionar en un mapa la estación de interés (Figura 18). Si el modelo correspondiente a la estación seleccionada ya existe, la API lo carga automáticamente. En caso contrario, el modelo se entrena con los hiperparámetros óptimos "representativos" previamente asignados. A continuación, se realizan las predicciones correspondientes sobre el conjunto de datos de *test* y se generan gráficas de validación para facilitar la interpretación de los resultados.

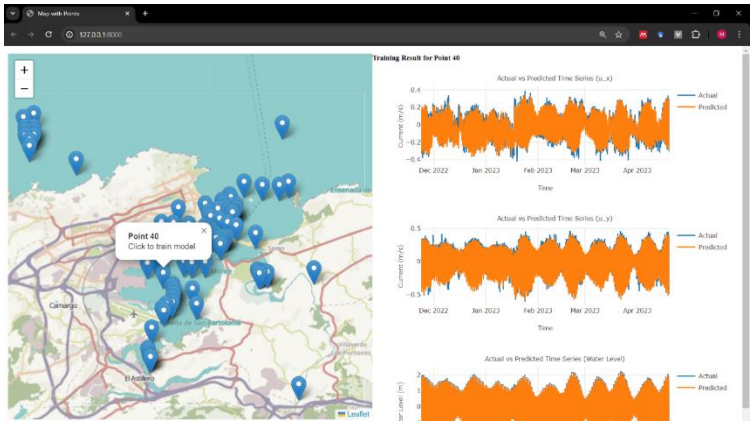


Figura 18. Ejemplo de validación de la estación 40 en la interfaz gráfica desarrollada.

6 RESULTADOS

En este apartado, se presentan los resultados obtenidos a partir de la validación sobre el periodo de *test* independiente definido en la Sección 5.1.5.1 DIVISIÓN EN *TRAIN* Y *TEST*. Se han validado los tres métodos aplicados en los 10 puntos representativos de la Bahía seleccionados, para cada una de las variables estudiadas: velocidad superficial de la corriente (ambas componentes) y nivel del mar. Los resultados consisten en:

- Las métricas de validación definidas en la Sección 4.6.1 SELECCIÓN DE MÉTRICAS. Se obtienen en forma de fichero JSON, uno para cada técnica, con ordenación jerárquica en variables.
- Gráficas de series temporales, para visualizar el comportamiento de las técnicas, comparando las series modeladas numéricamente (consideradas como *ground truth*) con las series predichas para cada variable. Estas gráficas permiten una evaluación visual de la precisión de las predicciones.
- Gráficas *scatter plot* que comparan los valores predichos por los modelos con los valores de *ground truth*. Se representan tanto las predicciones individuales como percentiles (P_{10}, P_{20}, P_{30}, P_{40}, P_{50}, P_{60}, P_{70}, P_{80} y P_{90}), lo cual facilita la comparación de las distribuciones.

En todos los casos, se ha guardado un archivo de configuración (config.json) que permite consultar los parámetros utilizados para configurar el caso y entrenar las técnicas. Adicionalmente, cuando se ha realizado la optimización de hiperparámetros, se han guardado los hiperparámetros óptimos obtenidos. Para el método AdaBoost, estos se han almacenado en un archivo CSV, mientras que para las redes neuronales RNN-LSTM, en un archivo JSON que puede utilizarse para entrenar una red en otro punto sin necesidad de optimizar nuevamente los hiperparámetros (como se detalla en la Sección 5.2 PREDICTANDOS).

Durante el proceso de entrenamiento, se han registrado manualmente los tiempos necesarios para entrenar cada técnica. Los tiempos medios obtenidos se presentan en la Tabla 4. Estos tiempos se han medido utilizando el siguiente equipo:

- Procesador: 13th Gen Intel(R) Core(TM) i7-13850HX 2.10 GHz
- RAM instalada: 32.0 GB
- GPU utilizada para RNN-LSTM: NVIDIA RTX A1000 6GB laptop GPU

Tabla 4. Tiempos medios de entrenamiento para cada técnica. Los tiempos marcados con (*) incluyen la optimización de hiperparámetros.

Técnica	Tiempo medio de entrenamiento
Análogos	2 minutos
AdaBoost	10 minutos (*)
RNN-LSTM	12 horas (*)

Se debe tener en cuenta que estos son tiempos de entrenamiento que incluyen la optimización de hiperparámetros en el caso de AdaBoost y RNN-LSTM. Sin incluir la optimización, los tiempos de entrenamiento se reducen a aproximadamente 1 minuto para

AdaBoost y 2 minutos para las RNN-LSTM. Los tiempos de predicción son pequeños y asumibles para las tres técnicas evaluadas.

Se presenta a continuación una serie de gráficas representativas de los resultados obtenidos en tres estaciones clave: estación 1 (interior), estación 31 (canal de navegación) y estación 5 (exterior). Adicionalmente, se incluyen gráficas para tres métricas representativas (MAE, coeficiente de correlación de Pearson y *Perkin's score*) que permiten comparar los resultados de cada una de las técnicas en todas las estaciones evaluadas (Figura 28, Figura 29 y Figura 30).

Aunque aquí se incluyen ejemplos representativos, los resultados completos están disponibles en la carpeta *annex* del repositorio GitHub del proyecto (Rupani, 2024).

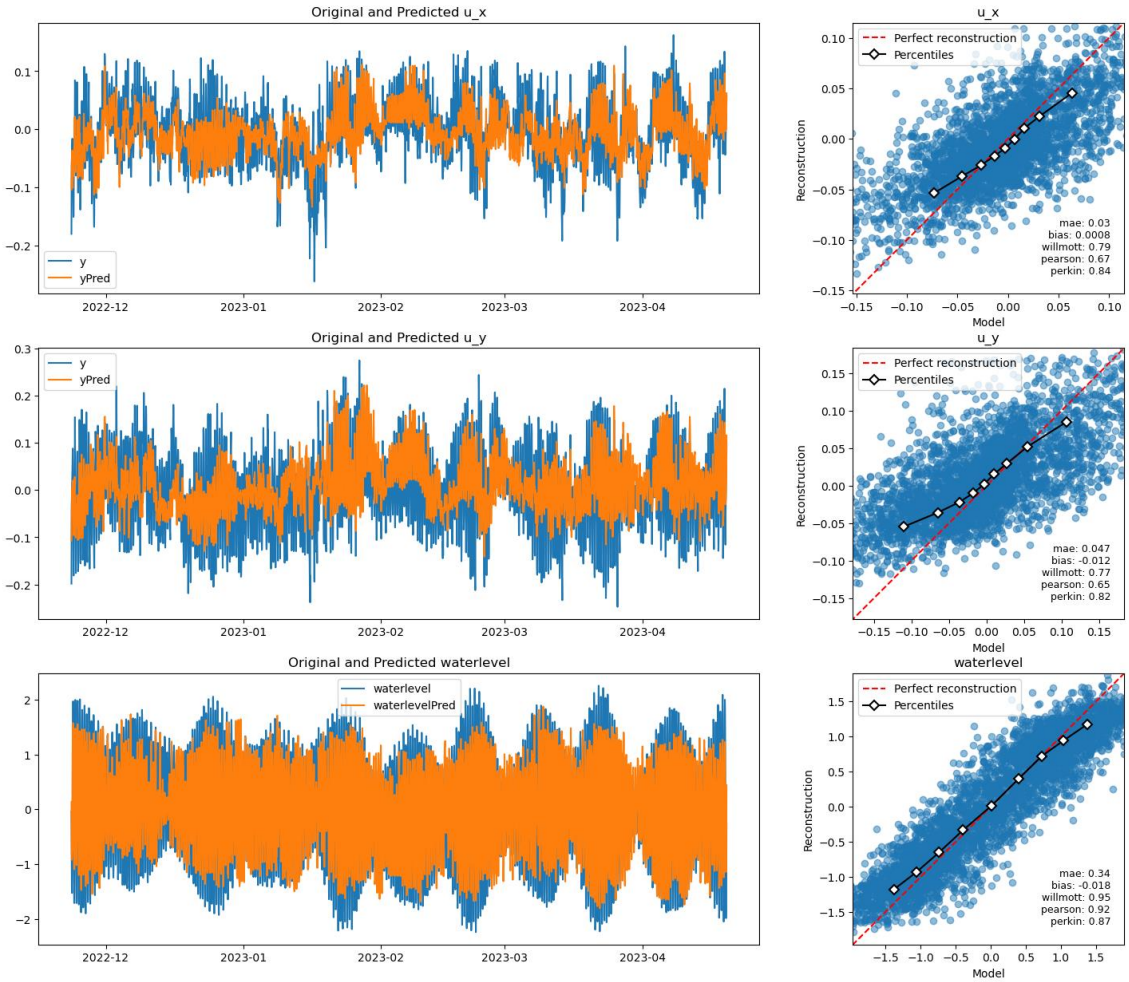
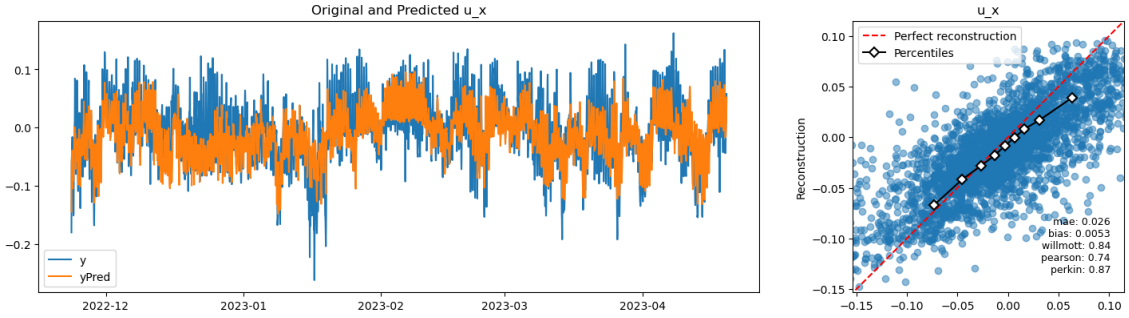


Figura 19. Técnica: análogos. Estación 1 (interior).



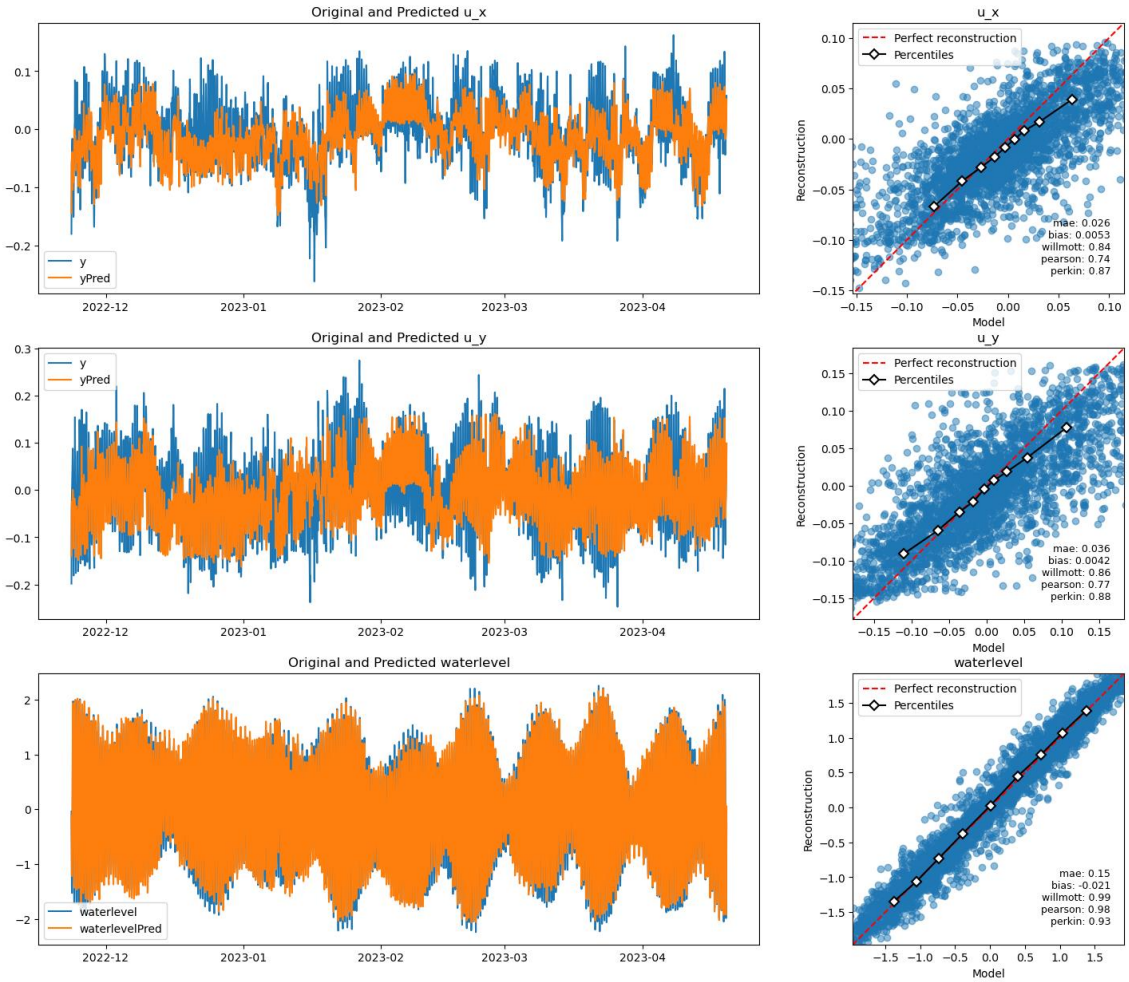
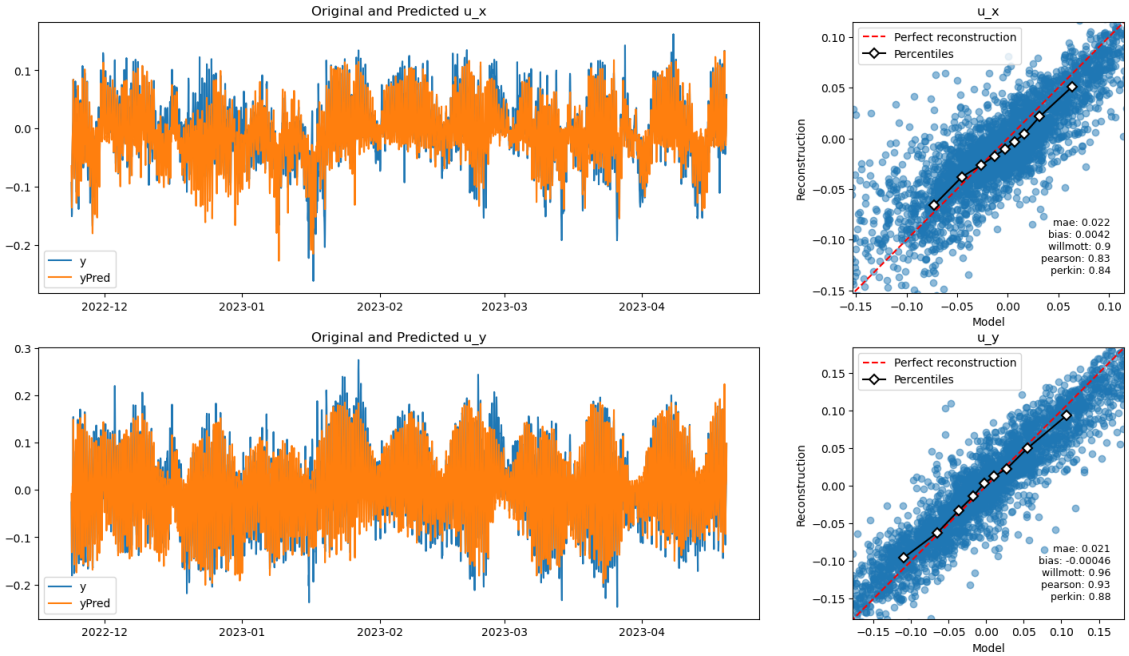


Figura 20. Técnica: AdaBoost. Estación 1 (interior).



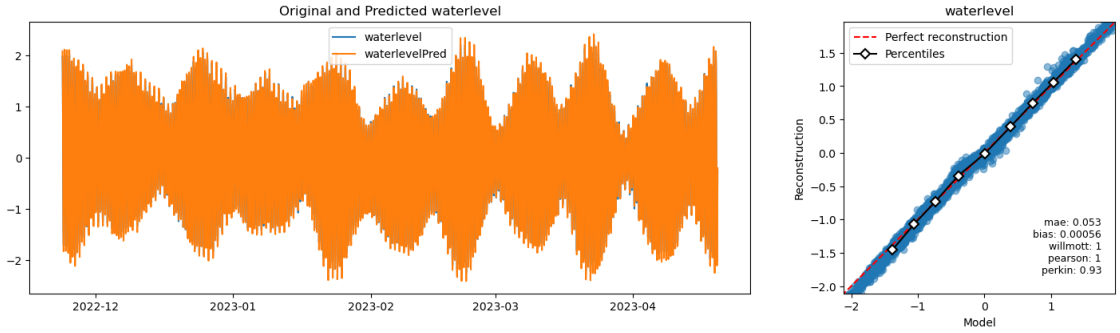


Figura 21. Técnica: RNN-LSTM. Estación 1 (interior).

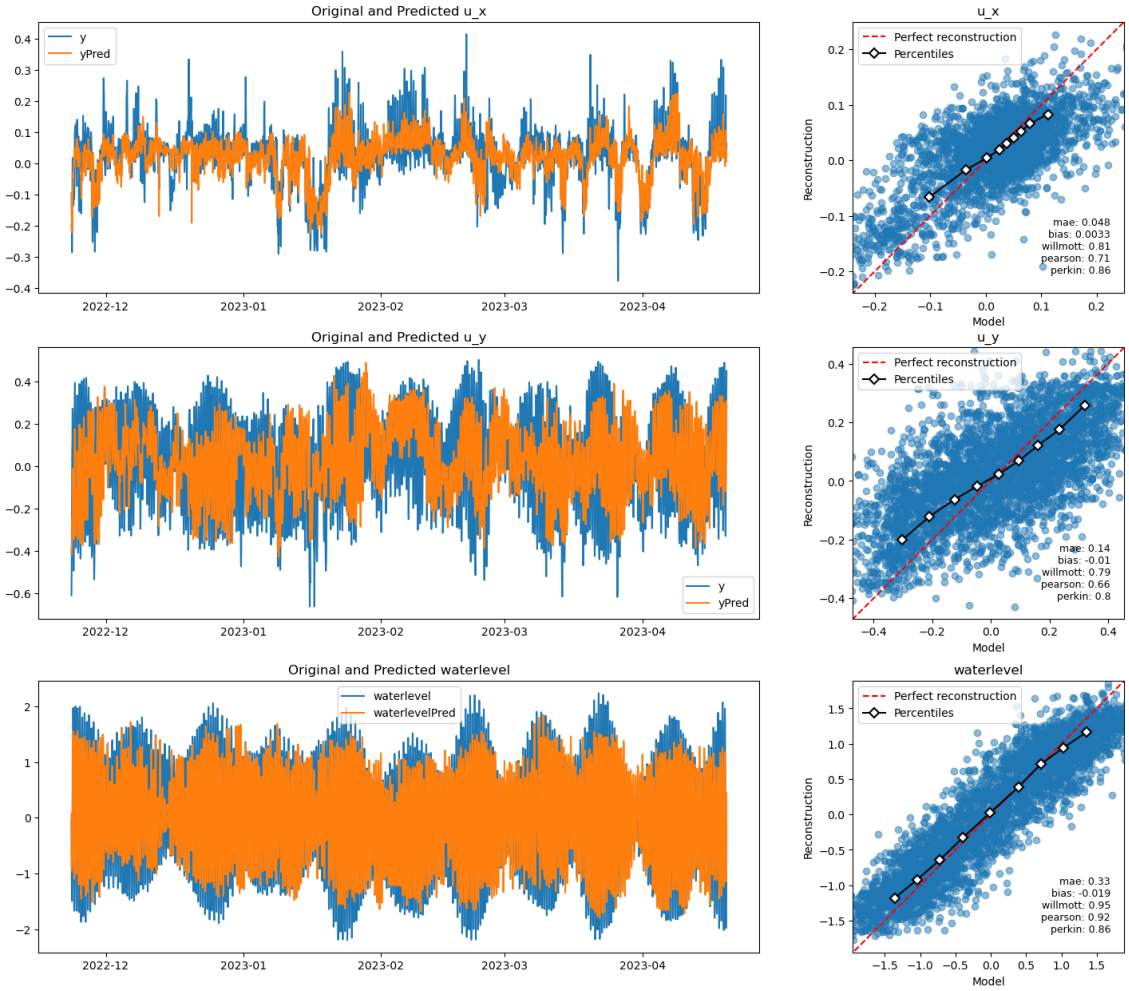
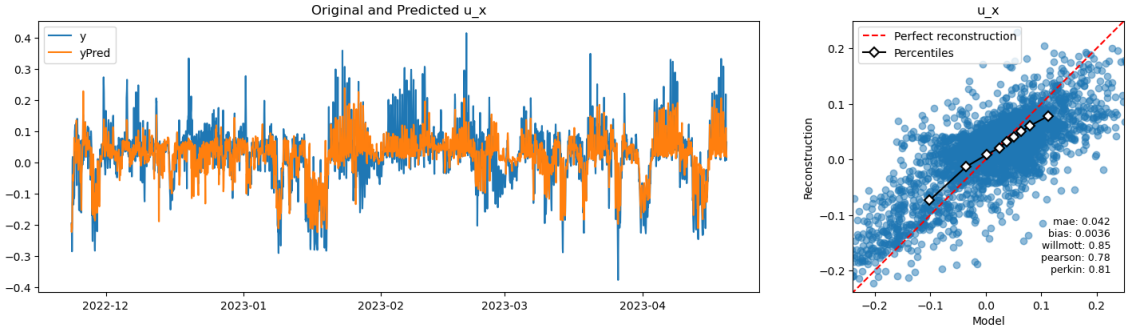


Figura 22. Técnica: análogos. Estación 31 (canal de navegación).



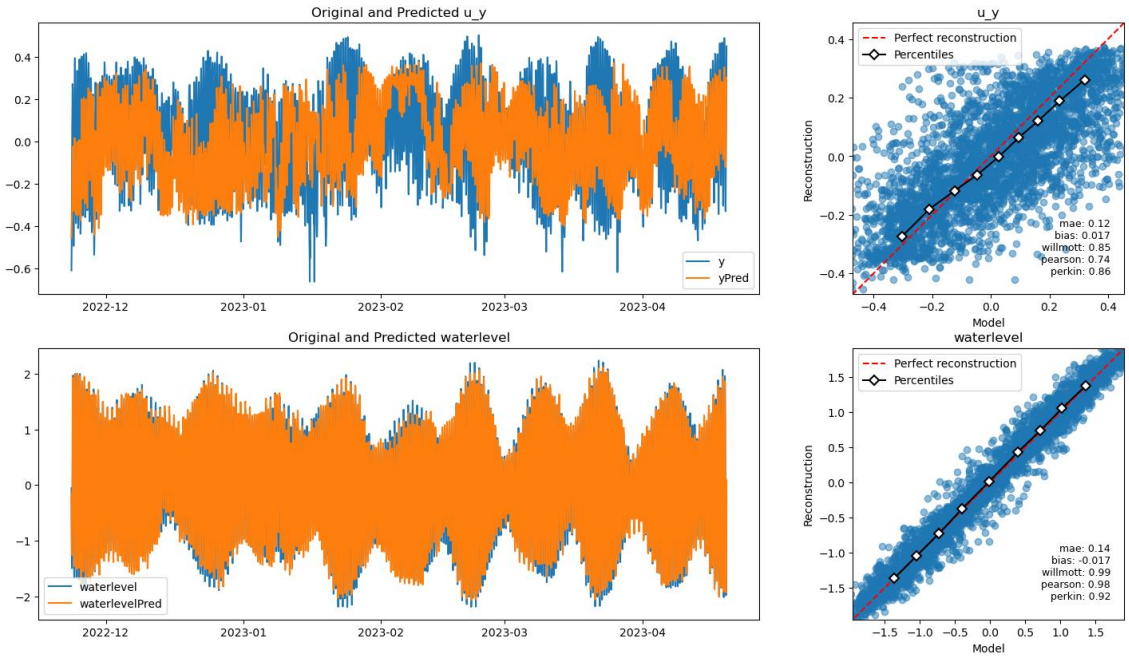


Figura 23. Técnica: AdaBoost. Estación 31 (canal de navegación).

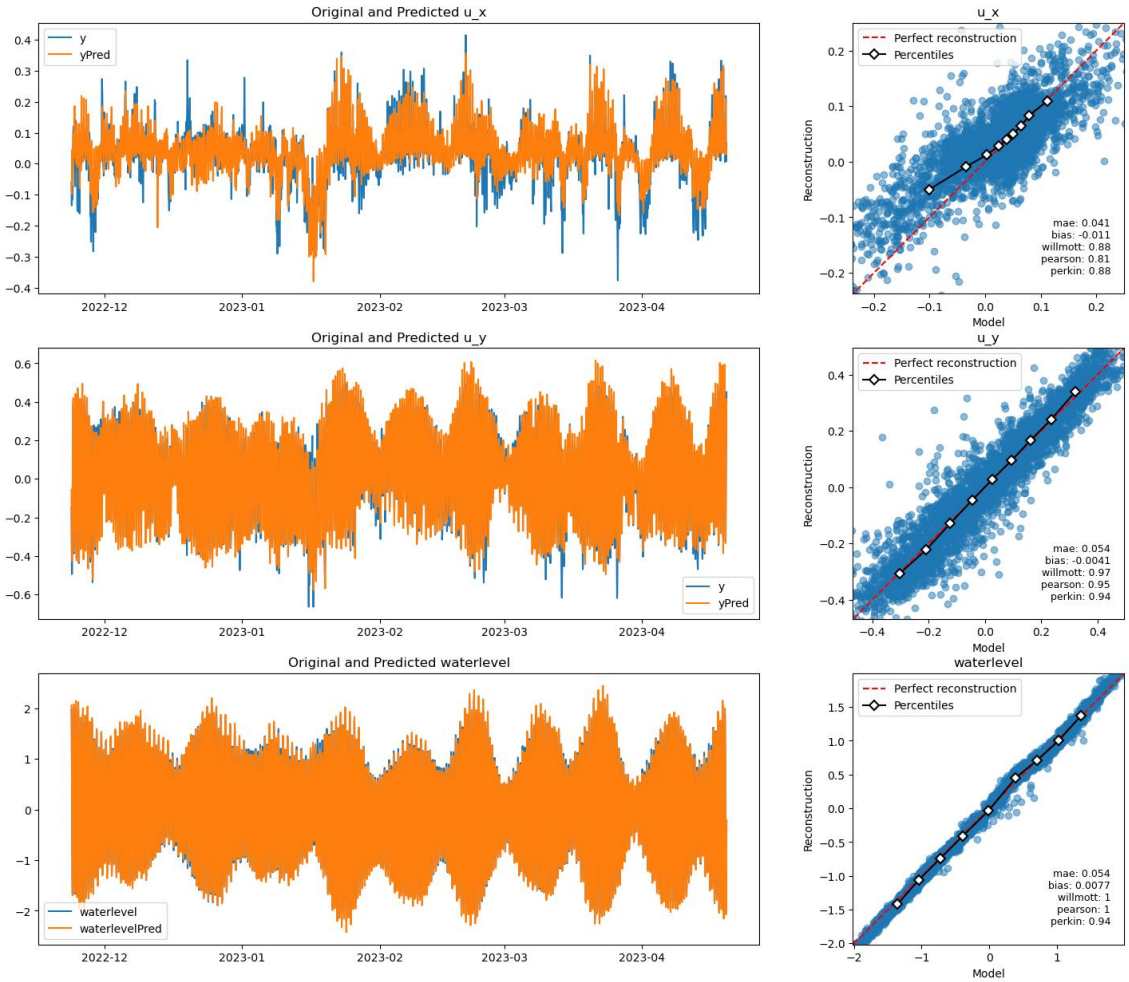


Figura 24. Técnica: RNN-LSTM. Estación 31 (canal de navegación).

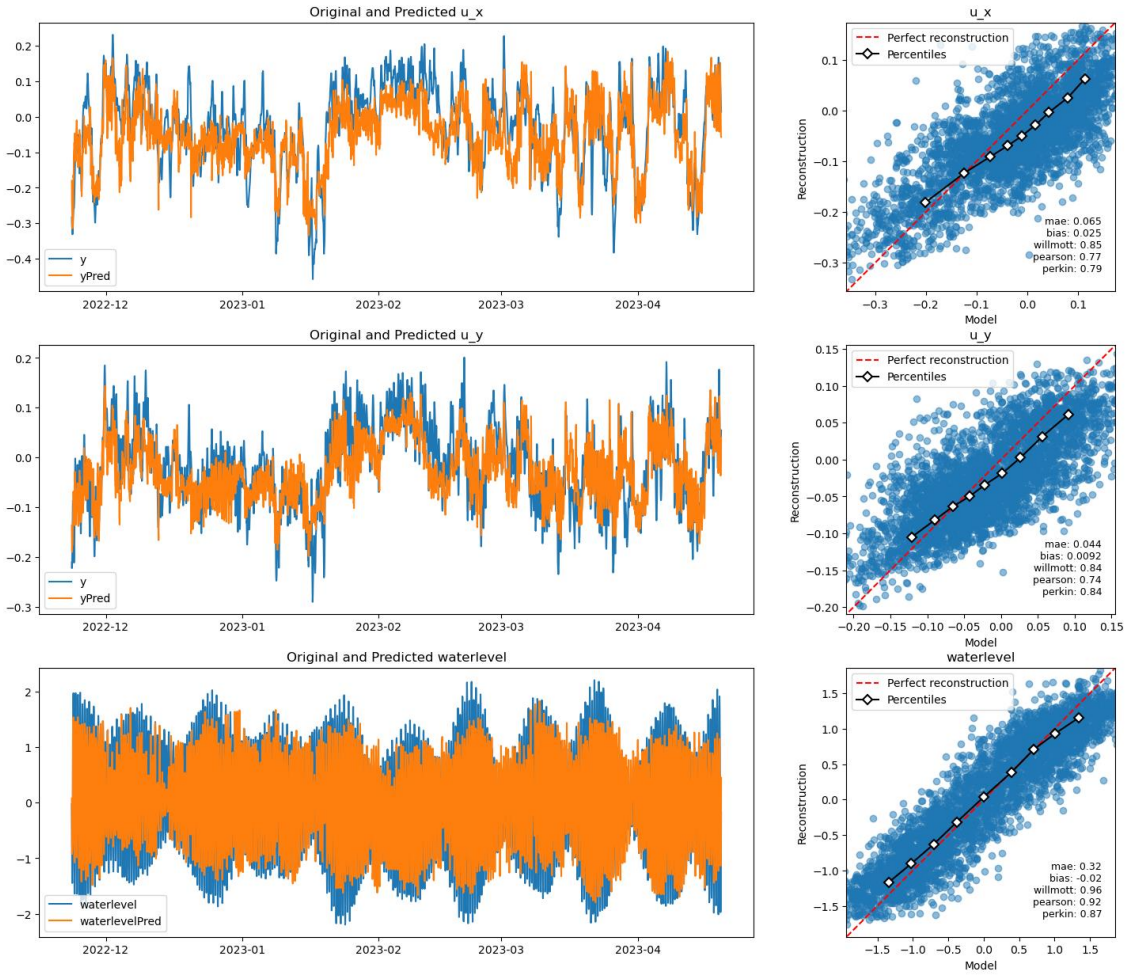
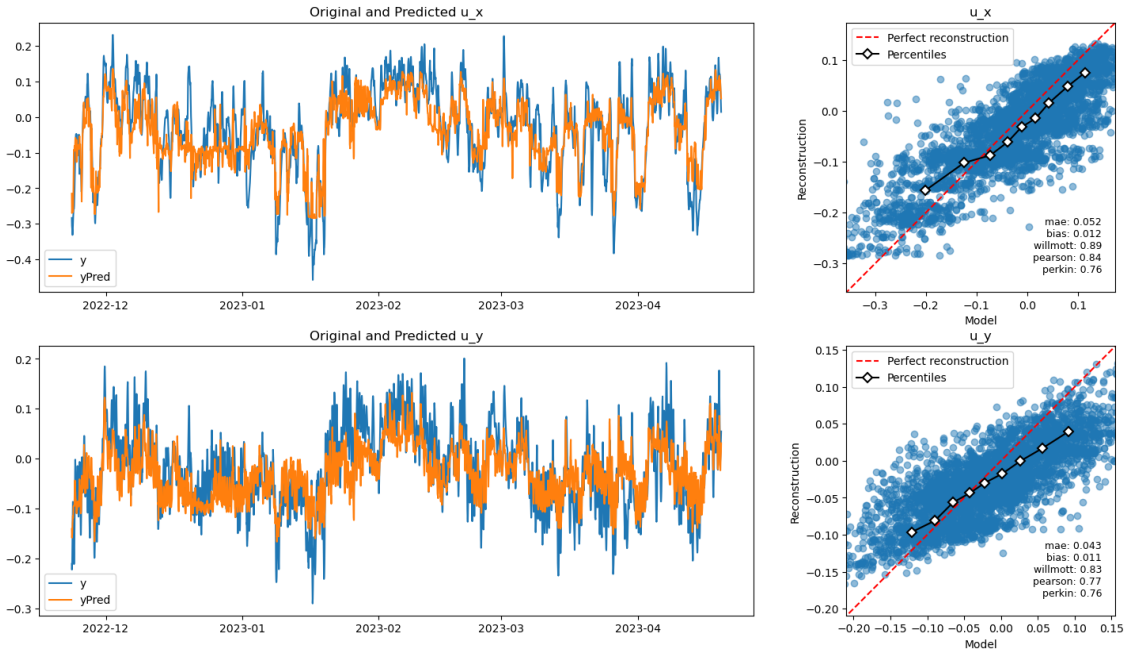


Figura 25. Técnica: análogos. Estación 5 (exterior).



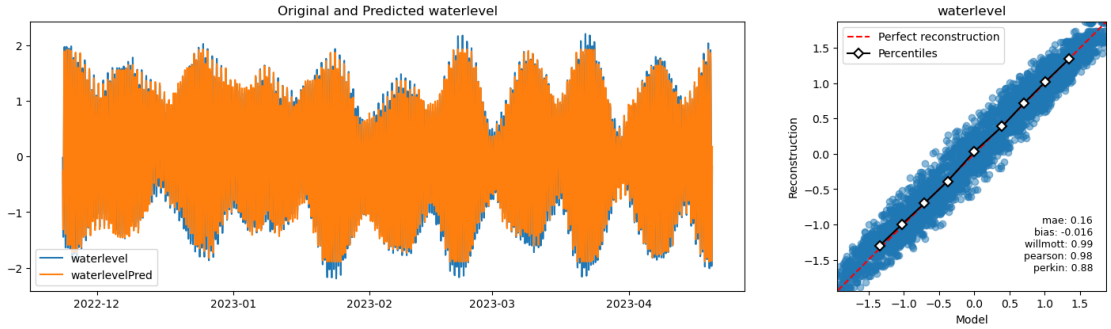


Figura 26. Técnica: AdaBoost. Estación 5 (exterior).

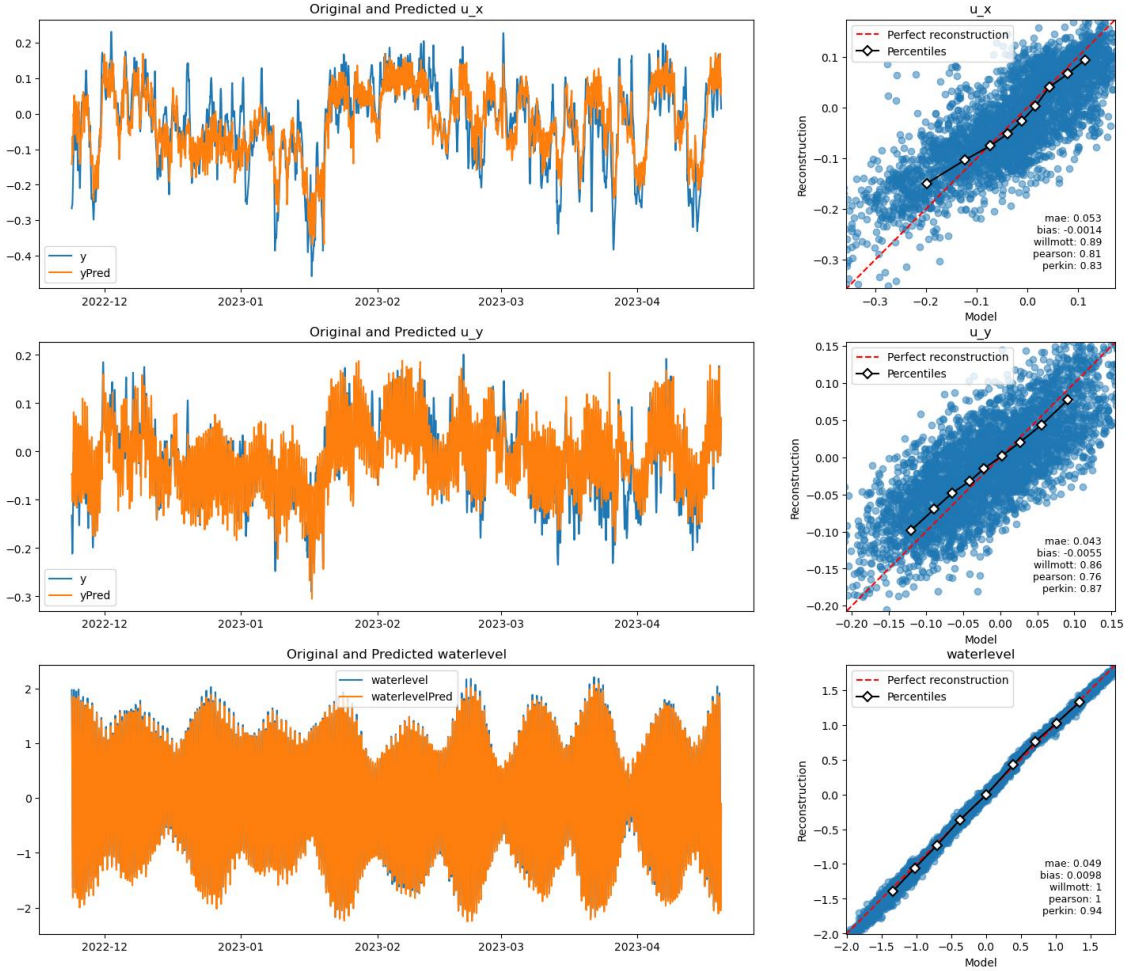
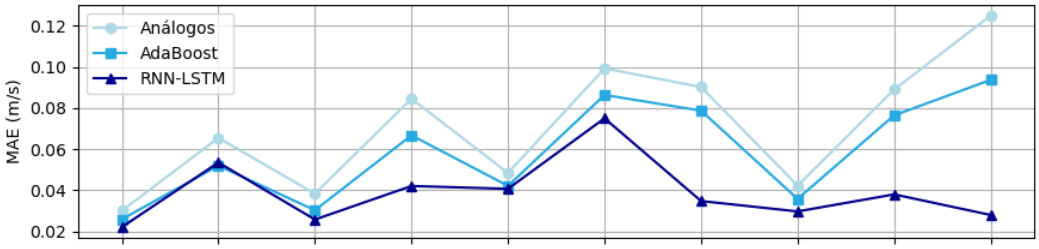


Figura 27. Técnica: RNN-LSTM. Estación 5 (exterior).



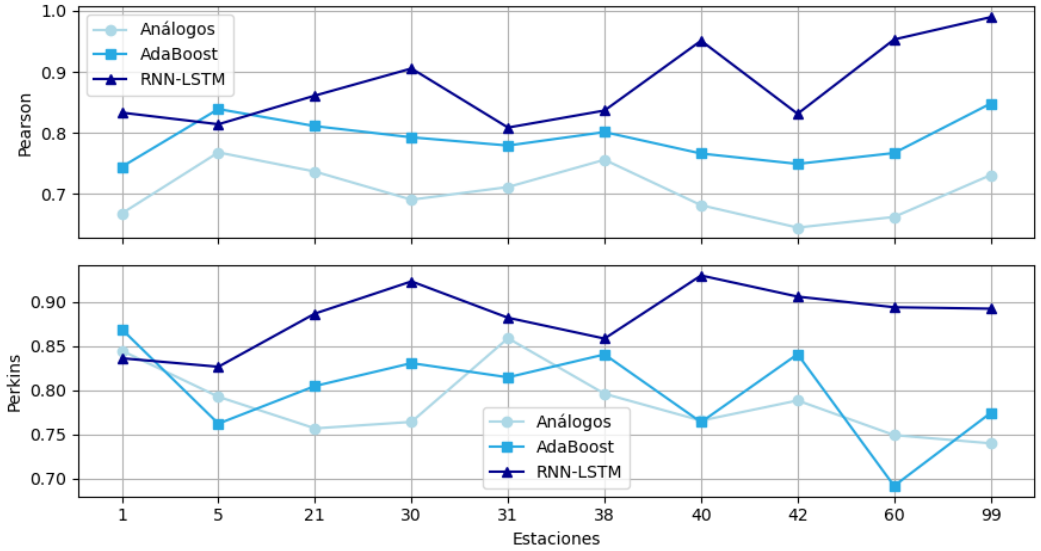


Figura 28. Mean absolute error (MAE), coeficiente de correlación de Pearson y Perkin's score para la componente zonal de la velocidad superficial de la corriente (u_x).

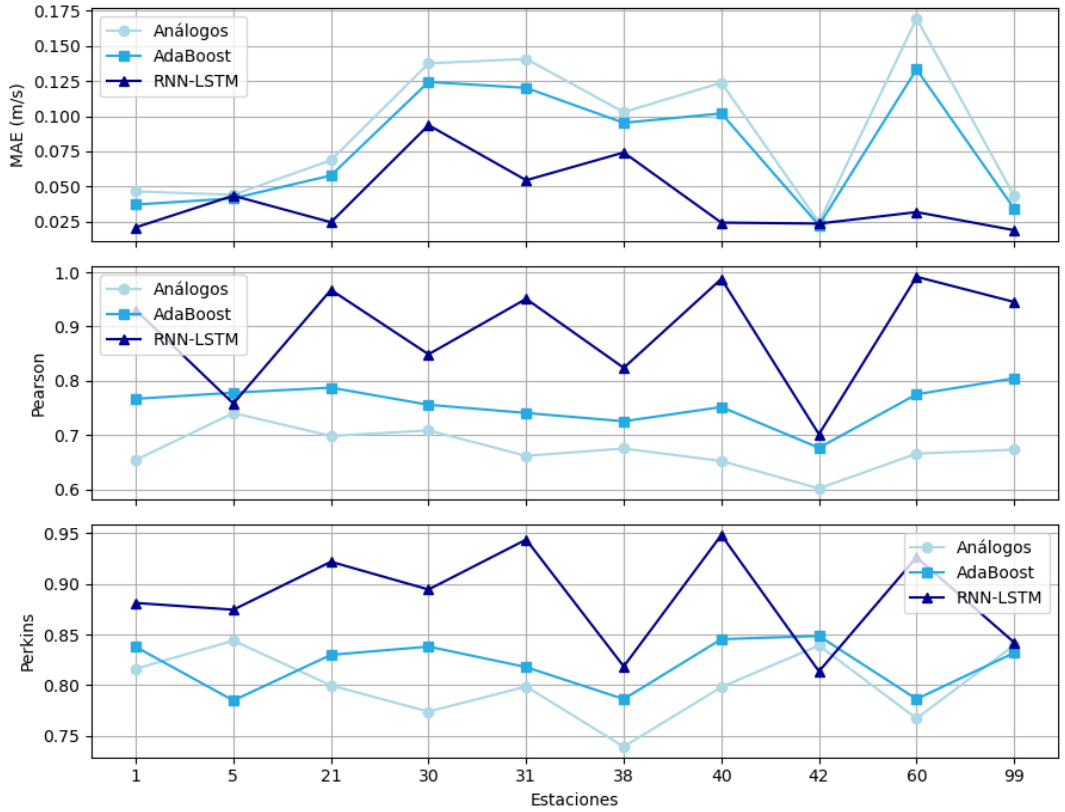
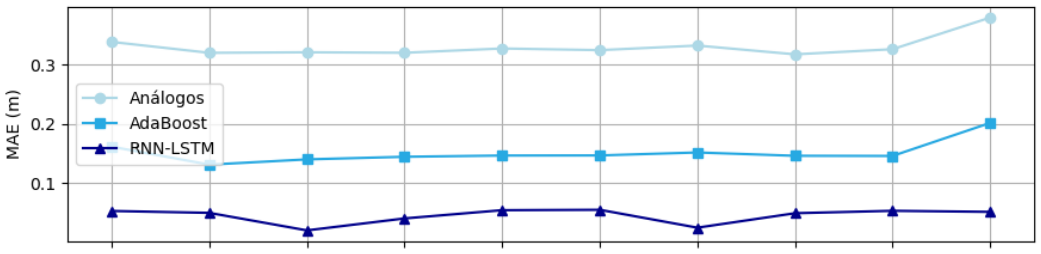


Figura 29. Mean absolute error (MAE), coeficiente de correlación de Pearson y Perkin's score para la componente meridional de la velocidad superficial de la corriente (u_y).



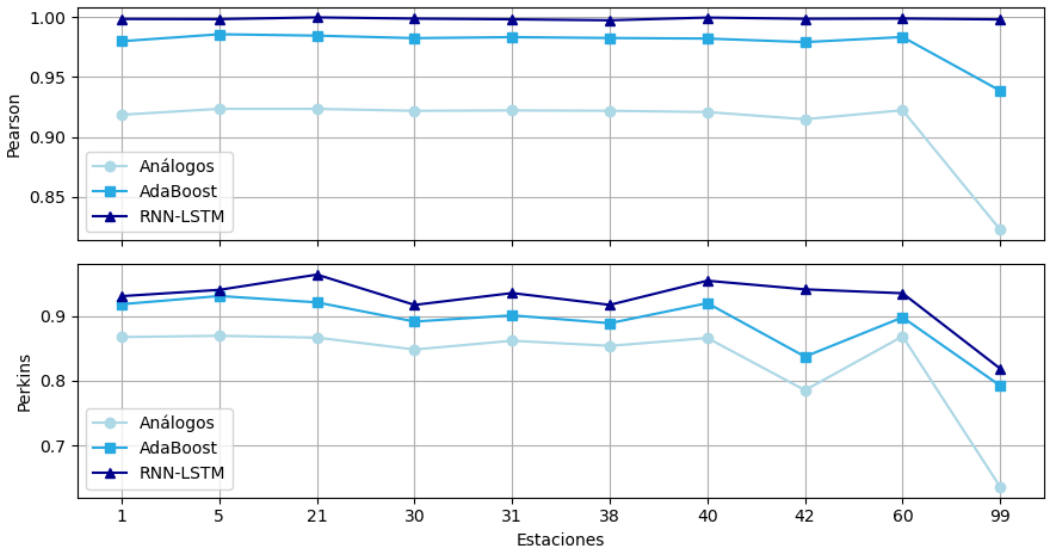


Figura 30. Mean absolute error (MAE), coeficiente de correlación de Pearson y Perkin's score para el nivel del mar.

6.1 DISCUSIÓN

6.1.1 OPTIMIZACIÓN DE HIPERPARÁMETROS

Después de realizar varias pruebas y considerar tanto los resultados como los tiempos de cómputo, se ha llegado a una configuración óptima para el método de análogos, que es común para todas las estaciones:

- Técnica de obtención de análogos: se ha seleccionado *spectral clustering*, aunque los resultados son similares al utilizar K-*means* o MaxDiss (el método no resulta muy sensible a este parámetro).
- Número de análogos: 500.
- Técnica de reconstrucción: KNN, utilizando 5 vecinos ponderados inversamente por la distancia. KRR aumenta el tiempo de cómputo, sin contribuir significativamente a la calidad de los resultados.

Al incrementar el número de vecinos, se observa una mejora en las métricas de correlación, pero a costa de suavizar los resultados y reducir la capacidad del modelo para predecir valores extremos.

En el caso de AdaBoost, la configuración óptima es única para cada estación y para cada variable. Así, se ha llegado a configuraciones que varían dentro de las siguientes horquillas:

- Número de estimadores (*i.e.* modelos *débiles;* en este caso, árboles de regresión): la técnica muestra una preferencia por números elevados (100 o 150 estimadores).
- Profundidad máxima de cada estimador: las profundidades óptimas suelen ser de 7 y 9 niveles, y ocasionalmente de 5. En ningún caso se han seleccionado como óptima una configuración basada en *stumps* (árboles con un único nivel de profundidad). Esta tendencia puede observarse en el ejemplo de la Figura 31.

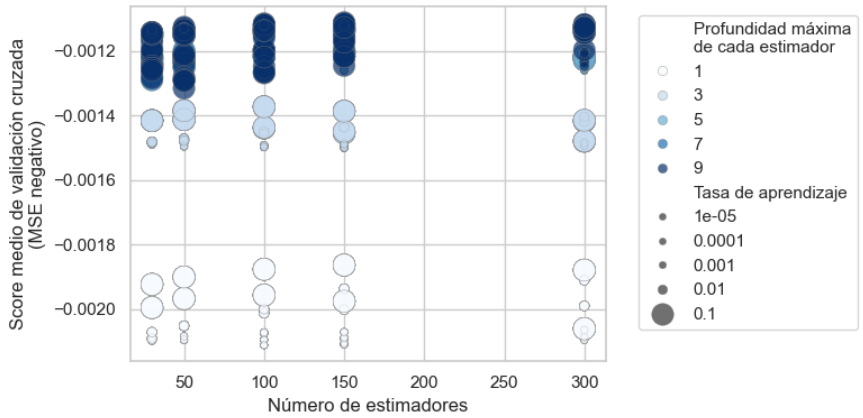


Figura 31. Resultados del proceso de optimización de hiperparámetros para la estación 1 y la variable u_x (componente zonal de la velocidad superficial de la corriente).

De todos modos, el impacto de estos dos hiperparámetros sobre los resultados es en líneas generales muy limitado. Sin embargo, se ha identificado que AdaBoost es muy sensible a la reducción de la dimensionalidad. En concreto, se obtienen resultados significativamente mejores utilizando directamente los campos crudos, incluso comparando con casos en los que se ha considerado un alto porcentaje de varianza explicada (99%). Conviene tener en cuenta que en este TFM los predictores son puntuales y no campos espaciales, por lo que se elimina el efecto de la redundancia y/o correlación espacial. En consecuencia, se ha decidido utilizar los campos crudos (sin reducción de dimensionalidad) en todas las técnicas.

En el caso de las redes LSTM, se obtiene una configuración óptima por cada estación. Se observa que los hiperparámetros óptimos obtenidos varían considerablemente de un punto a otro:

- Número de capas LSTM: siempre 1 o 2 (en ningún caso se ha seleccionado como óptima una arquitectura de 3 capas).
- Unidades LSTM por capa: varían mucho, tomando valores de 50, 100, 150, y 200.
- *El batch size* (tamaño del lote a partir del cual se realiza una actualización de los pesos de la red) óptimo es de 256 datos, salvo en una estación.

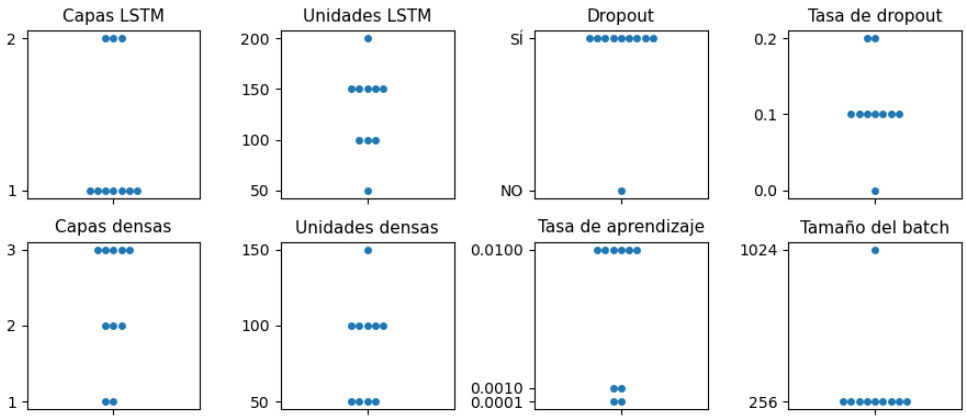


Figura 32. Configuraciones óptimas obtenidas para las diferentes estaciones, optimizando el MSE.

Otros hiperparámetros, tanto de arquitectura (número de capas densas y neuronas) como de aprendizaje (*learning rate*, *dropout rate*), cubren un amplio espectro de valores, lo que se aprecia en la Figura 32. A pesar de esta variabilidad, los resultados obtenidos son

consistentemente buenos, lo que permite aplicar con éxito los hiperparámetros óptimos de un punto en el entrenamiento de redes en otros puntos. A partir de la API desarrollada, se ha comprobado que entrenar una red LSTM en un punto utilizando los hiperparámetros óptimos de otro punto proporciona resultados comparables a los obtenidos al optimizar los hiperparámetros específicamente para cada punto. Esto hace que la predicción sobre todo el dominio utilizando RNN-LSTM sea viable, considerando que el entrenamiento de la red lleva solo un par de minutos.

6.1.2 VALIDACIÓN

6.1.2.1 NIVEL DEL MAR

La predicción del nivel del mar debe tomarse como un punto de referencia fundamental, especialmente en un estuario macromareal como el estudiado, donde el nivel está fuertemente influenciado por la marea, que tiene una naturaleza determinista. En general, los resultados son buenos, capturando adecuadamente las ondas de marea e incluso prediciendo correctamente las zonas intermareales que quedan secas en ciertos momentos.

Sin embargo, los resultados obtenidos con el método de análogos no llegan a ser completamente satisfactorios. Se observa una dispersión significativa en los *scatter plots* y una subestimación de los valores extremos. A pesar de alcanzar valores de correlación de Pearson del orden de 0.92, los métodos AdaBoost y RNN-LSTM logran valores de 0.98 y casi 1, respectivamente. En las gráficas comparativas de MAE y coeficiente de correlación de Pearson, se observa que los resultados de AdaBoost y LSTM son marcadamente mejores a los de análogos, con LSTM desempeñándose un poco mejor que AdaBoost. En cuanto al *Perkin's score*, el método de análogos presenta también presenta el peor rendimiento, pero la diferencia es menos marcada.

Es importante señalar que la optimización de los hiperparámetros del método de análogos se ha realizado manualmente. La automatización de este proceso permitiría evaluar más configuraciones de manera eficiente, lo que podría conducir a una mejora en los resultados.

6.1.2.2 CORRIENTES

Los resultados para las componentes de la velocidad superficial de la corriente son, en general, muy buenos con todas las técnicas utilizadas. Se representan adecuadamente los valores extremos y se predicen bien las corrientes en puntos con comportamientos muy distintos (algunos están influenciados predominantemente por la marea, y otros más por el viento). En casi todos los casos, el coeficiente de correlación de Pearson se mantiene por encima de 0.65 para análogos, 0.75 para AdaBoost y 0.82 en las RNN-LSTM. Si se analiza el MAE, se mantiene la misma tendencia: RNN-LSTM es el mejor de los tres, seguido por AdaBoost y luego por análogos. En cuanto al *Perkin's score*, análogos y AdaBoost presentan resultados comparables (especialmente para la componente zonal), mientras que RNN-LSTM es superior (salvo excepciones).

Estos resultados demuestran que, aunque los métodos de análogos pueden ser útiles en ciertas circunstancias y presentan otras ventajas (p. ej. interpretablidad), las técnicas más avanzadas como AdaBoost y RNN-LSTM ofrecen una precisión significativamente mayor para las variables analizadas en este estudio.

7 CONCLUSIONES

Para este TFM se ha desarrollado una metodología para el modelado hidrodinámico de alta resolución en la Bahía de Santander, utilizando diversas técnicas de aprendizaje automático. Se han considerado como variables objetivo el nivel del mar y las velocidades superficiales de la corriente (ambas componentes). Se ha hecho una revisión exhaustiva del estado del conocimiento, la selección de forzamientos relevantes, el modelado numérico con el *software* Delft3D para obtener datos de entrenamiento, y la implementación y validación de varias técnicas de aprendizaje automático. Las técnicas evaluadas abarcan desde enfoques menos complejos hasta los más avanzados. Se ha prestado especial atención a la optimización de hiperparámetros y se ha realizado un análisis comparativo de los resultados obtenidos.

Del análisis realizado se extraen las siguientes conclusiones:

1. Con respecto al modelado de las variables hidrodinámicas:
    - Todas las técnicas han demostrado ser efectivas en la predicción de corrientes y nivel del mar con resultados satisfactorios, siendo las que demuestran mayor precisión las basadas en RNN-LSTM.
    - Los resultados sugieren la potencialidad de las técnicas de aprendizaje automático para modelado hidrodinámico, presentando ventajas respecto a los modelos numéricos tradicionales en términos de coste computacional, ya que una vez entrenadas, son más rápidas y menos exigentes en recursos computacionales.
2. Con respecto a las técnicas:
    - La mayor precisión en la reconstrucción de las variables hidrodinámicas se ha obtenido con las RNN-LSTM y las mayores diferencias se han observado al aplicar la predicción condicionada a análogos.
    - El método de análogos es más sencillo desde el punto de vista metodológico, pero requiere un conocimiento mayor de la dinámica de la zona para la definición de predictores y, por tanto, mayor esfuerzo de configuración. Aunque los resultados obtenidos muestran mayor incertidumbre en la simulación de las variables con este método, especialmente en el nivel del mar, existe margen de mejora en la automatización del proceso de optimización de hiperparámetros, lo que podría conducir a mejores resultados.
    - La exhaustiva optimización de hiperparámetros con Hyperband en el caso de las RNN-LSTM, aunque tiene un coste computacional elevado, ha demostrado ser extrapolable espacialmente. Los hiperparámetros óptimos obtenidos para un punto funcionan bien en otros puntos, garantizando la escalabilidad del método.
    - La capacidad de extrapolación de hiperparámetros óptimos probablemente sea también válida para AdaBoost, pero los resultados finales no son tan buenos como los obtenidos con las RNN-LSTM y el tiempo de entrenamiento requerido es del mismo orden.
3. Con respecto a la metodología de aplicación:

- Para desarrollar un modelo de predicción de corrientes y nivel del mar a escala local (*i.e.* en toda la Bahía de Santander), lo más adecuado entre los métodos evaluados sería utilizar RNN-LSTM, y seguir los pasos que se presentan a continuación:
  a. Clasificar los puntos en la Bahía según sus características y el comportamiento de las variables a predecir (p. ej. mediante técnicas de *clustering*).
  b. Optimizar los hiperparámetros de una red LSTM para un punto representativo de cada grupo.
  c. Con los hiperparámetros óptimos, entrenar una red LSTM para cada punto, lo cual resulta asumible en términos de recursos computacionales. Con una tarjeta gráfica básica, el proceso de entrenamiento lleva un par de minutos.
  d. Realizar las predicciones. Una vez entrenadas las redes para todos los puntos, la predicción es muy rápida.
- Debería verificarse que las predicciones mantengan cierta coherencia espacial, lo cual es esperable dado que los predictores son los mismos y los predictandos utilizados (salidas de un modelo dinámico) tienen coherencia espacial. Análisis preliminares en puntos cercanos sugieren que los resultados espaciales tienen sentido, pero idealmente debería evaluarse el mapa completo de predicciones.

Se ha demostrado que las metodologías de modelado hidrodinámico basadas en aprendizaje automático desarrolladas proporcionan buenos resultados y son escalables. Su principal ventaja reside en la significativa reducción del coste computacional, al eliminar la necesidad de utilizar métodos numéricos mucho más demandantes de recursos. Esto hace que el modelado hidrodinámico sea más accesible para una variedad de aplicaciones. Por ejemplo, permite implementar modelos operacionales de corrientes a un coste mucho menor, lo cual es crucial para la gestión de ecosistemas costeros como la Bahía de Santander. Estos modelos operacionales son esenciales para diversas aplicaciones, incluyendo la navegación segura, la gestión de recursos marinos y la respuesta a eventos extremos, contribuyendo así a la sostenibilidad y protección de estos entornos naturales.

8 TRABAJO FUTURO

A partir del TFM, se han identificado varios aspectos prometedores para continuar desarrollando, y posibles mejoras en la metodología implementada, que se presentan a continuación. Se mencionan primero los puntos que aportarían una contribución significativa, y que al mismo tiempo pueden implementarse de manera relativamente sencilla.

- Validar los resultados obtenidos con los datos medidos en las campañas hidrodinámicas que han sido realizadas *ad hoc*. No se ha conseguido realizar en esta instancia debido a limitaciones computacionales asociadas con el tamaño de los archivos de predictandos.
- Implementar una variante de K-*fold cross-validation* con Keras, diseñada específicamente para series temporales. Esta técnica permitirá una evaluación más robusta (la validación ya no dependería de la partición en subconjuntos de entrenamiento y validación).
- Automatizar del proceso de optimización de hiperparámetros en el método de análogos. Esto agilizaría el ajuste y permitiría explorar un espacio de hiperparámetros más amplio.
- Mejora del modelado de temperatura y salinidad. En el TFM se han realizado pruebas iniciales, pero no se ha conseguido predecir con suficiente precisión estas variables. Es necesario explorar otras variables predictoras y técnicas de modelado.
- Análisis espacial y generación de mapas: realizar un entrenamiento punto a punto con RNN-LSTM, previa asignación de hiperparámetros óptimos mediante técnicas de *clustering*.
- Investigar la capacidad de las redes ya entrenadas (o, al menos, sus hiperparámetros óptimos) para adaptarse a diferentes ubicaciones y condiciones ambientales, especialmente en el contexto de un entorno cambiante debido al cambio climático.
- Explorar la aplicación de la metodología desarrollada a otras escalas (p. ej., escala costera: golfo de Vizcaya), para evaluar su efectividad en diferentes contextos geográficos.
- Explorar otras técnicas de aprendizaje automático (p. ej. XGBoost), y evaluar su desempeño.
- Investigar la viabilidad de utilizar otras técnicas (p. ej. redes convolucionales LSTM, como las implementadas por Joshi (2021)) para realizar predicciones directamente en el espacio, lo que podría resultar más preciso y garantizaría coherencia espacial.
- Implementación de AI explicativa: explorar técnicas de *explainable AI* para comprender mejor qué aspectos de los predictores utilizan los modelos en sus decisiones de predicción.
- Entrenamiento con datos medidos: investigar la posibilidad de entrenar los modelos utilizando datos medidos en lugar de resultados de modelado numérico. Esto podría aplicarse a datos puntuales de campañas hidrodinámicas, pero es especialmente interesante su aplicación para predecir corrientes superficiales a partir de datos de radar.
- Evaluar la adaptación de la metodología desarrollada para la predicción del transporte y la dispersión de contaminantes marinos.

Estas futuras líneas de investigación representan oportunidades muy interesantes para ampliar y mejorar el trabajo realizado hasta la fecha, así como para abordar nuevos desafíos en el campo de la modelización oceanográfica y la predicción de variables ambientales.

9 REFERENCIAS

Acevedo, A., Menéndez, M., Tomas, A., & Iturrioz, A. (2019). A hybrid downscaling approach for short-term forecasting offshore surface atmospheric variables in coastal areas. *Geophysical Research Abstracts*, *21*.

Asadollah, S. B. H. S., Khan, N., Sharafati, A., Shahid, S., Chung, E.-S., & Wang, X.-J. (2022). Prediction of heat waves using meteorological variables in diverse regions of Iran with advanced machine learning models. *Stochastic Environmental Research and Risk Assessment*, 1–16.

Autoridad Portuaria de Santander, & Instituto de Hidráulica Ambiental de la Universidad de Cantabria. (2009). *Informe de sostenibilidad ambiental del plan director de infraestructuras del Puerto de Santander - Anejo II*.

Bayindir, C. (2019). Predicting the Ocean Currents using Deep Learning. *ArXiv Preprint ArXiv:1906.08066*.

Bolton, T., & Zanna, L. (2019). Applications of deep learning to ocean data inference and subgrid parameterization. *Journal of Advances in Modeling Earth Systems*, *11*(1), 376–399.

Cagigal, L., Méndez, F. J., Ricondo, A., Gutiérrez-Barceló, D., Bosserelle, C., & Hoeke, R. (2024). BinWaves: An additive hybrid method to downscale directional wave spectra to nearshore areas. *Ocean Modelling*, *189*, 102346. https://doi.org/https://doi.org/10.1016/j.ocemod.2024.102346

Camus, P., Mendez, F. J., Medina, R., & Cofiño, A. S. (2011). Analysis of clustering and selection algorithms for the study of multivariate wave climate. *Coastal Engineering*, *58*(6), 453–462.

Camus, P., Menendez, M., Mendez, F. J., Izaguirre, C., Espejo, A., Canovas, V., Perez, J., Rueda, A., Losada, I. J., & Medina, R. (2014). A weather-type statistical downscaling framework for ocean wave climate. *Journal of Geophysical Research: Oceans*, *119*(11), 7389–7405.

Cendrero Uceda, A., & Diaz de Terán, J. R. (1977). Caracterización cuantitativa del desarrollo histórico del relleno de la bahía de Santander; Un proceso natural activado por el hombre. *Revista de Obras Públicas*, *124*(3150), 797–808. http://ropdigital.ciccp.es/pdf/publico/1977/1977_octubre_3150_04.pdf

Chattopadhyay, A., Nabizadeh, E., & Hassanzadeh, P. (2020). Analog forecasting of extreme-causing weather patterns using deep learning. *Journal of Advances in Modeling Earth Systems*, *12*(2), e2019MS001958.

Chiri, H., Abascal, A. J., Castanedo, S., Antolínez, J. A. A., Liu, Y., Weisberg, R. H., & Medina, R. (2019). Statistical simulation of ocean current patterns using autoregressive logistic regression models: A case study in the Gulf of Mexico. *Ocean Modelling*, *136*, 1–12.

Chollet, F., & others. (2015). *Keras*.

Cid, A., Camus, P., Castanedo, S., Méndez, F. J., & Medina, R. (2017). Global reconstructed daily surge levels from the 20th Century Reanalysis (1871–2010). *Global and Planetary Change*, *148*, 9–21.

Cid, A., Wahl, T., Chambers, D. P., & Muis, S. (2018). Storm surge reconstruction and return water level estimation in Southeast Asia for the 20th century. *Journal of Geophysical Research: Oceans*, *123*(1), 437–451.

Confederación Hidrográfica del Cantábrico. (n.d.). *Ficha Río Miera*. Retrieved September 9, 2021, from www.chcantabrico.es

Deltares. (2021). *Delft3D 3D/2D modelling suite for integral water solutions Hydro-Morphodynamics - User Manual*.

Díaz, C. Á., Alonso, E. G., Gómez, C. P. S. J. R., & Vega, B. T. (2015). *Aplicación de un modelo logístico triparamétrico a la estimación de caudales diarios en la cuenca del río Nam Ngum (Laos)*.

Drucker, H. (1997). Improving regressors using boosting techniques. *Icml*, *97*(107), e115.

E.U. Copernicus Marine Service Information (CMEMS). Marine Data Store (MDS). (n.d.). *Atlantic-Iberian Biscay Irish-Ocean Physics Analysis and Forecast*. https://doi.org/10.48670/moi-00027

García Pérez, P., & IHCantabria - Instituto de Hidráulica Ambiental de la Universidad de Cantabria. (2024). *The Maximum Dissimilarity Algorithm*. https://github.com/IHCantabria/mdapy

Géron, A. (2019). *Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow : concepts, tools, and techniques to build intelligent systems* (Second edition). O'Reilly. https://search.ebscohost.com/login.aspx?direct=true&scope=site&db=nlebk&db=nlabk&AN=2245240

Harris, C. R., Millman, K. J., Van Der Walt, S. J., Gommers, R., Virtanen, P., Cournapeau, D., Wieser, E., Taylor, J., Berg, S., & Smith, N. J. (2020). Array programming with NumPy. *Nature*, *585*(7825), 357–362.

Hegermiller, C. A., Antolinez, J. A. A., Rueda, A., Camus, P., Perez, J., Erikson, L. H., Barnard, P. L., & Mendez, F. J. (2017). A multimodal wave spectrum–based approach for statistical downscaling of local wave climate. *Journal of Physical Oceanography*, *47*(2), 375–386.

Hernanz, A., García-Valero, J. A., Domínguez, M., & Rodríguez-Camino, E. (2022). A critical view on the suitability of machine learning techniques to downscale climate change projections: Illustration for temperature with a toy experiment. *Atmospheric Science Letters*, *23*(6), e1087.

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, *9*(8), 1735–1780.

Hollinger, G., Pereira, A., Ortenzi, V., & Sukhatme, G. (2012). Towards improved prediction of ocean processes using statistical machine learning. *Robotics: Science and Systems Workshop on Robotics for Environmental Monitoring*.

Hunter, J. D. (2007). Matplotlib: A 2D graphics environment. *Computing in Science & Engineering*, *9*(03), 90–95.

Immas, A., Do, N., & Alam, M.-R. (2021). Real-time in situ prediction of ocean currents. *Ocean Engineering*, *228*, 108922.

Instituto Cántabro de Estadística. (2023). *Población por sexo y municipio*. https://www.icane.es/data/padron-municipal-habitantes-genero-municipio

Joshi, A. (2021). *Next-frame video prediction with convolutional lstms*. https://keras.io/examples/vision/conv_lstm/

Kohavi, R. (1995). A study of cross-validation and bootstrap for accuracy estimation and model selection. *Ijcai*, *14*(2), 1137–1145.

Kolmogorov, A. N. (1933). Sulla determinazione empirica di una legge di distribuzione. *Giornale Dell'Istituto Italiano Degli Attuari*, *4*, 83–91.

Li, L., Jamieson, K., DeSalvo, G., Rostamizadeh, A., & Talwalkar, A. (2018). Hyperband: A novel bandit-based approach to hyperparameter optimization. *Journal of Machine Learning Research*, *18*(185), 1–52.

Liu, J., Yang, J., Liu, K., & Xu, L. (2022). Ocean current prediction using the weighted pure attention mechanism. *Journal of Marine Science and Engineering*, *10*(5), 592.

Lorenz, E. N. (1969). Atmospheric predictability as revealed by naturally occurring analogues. *Journal of Atmospheric Sciences*, *26*(4), 636–646.

Madec, G., Bell, M., Blaker, A., Bricaud, C., Bruciaferri, D., Castrillo, M., Calvert, D., Chanut, J., Clementi, E., Coward, A., Epicoco, I., Éthé, C., Ganderton, J., Harle, J., Hutchinson, K., Iovino, D., Lea, D., Lovato, T., Martin, M., … Wilson, C. (2023). *NEMO Ocean Engine Reference Manual*. Zenodo. https://doi.org/10.5281/zenodo.8167700

McKinney, W. (2010). Data structures for statistical computing in Python. *SciPy*, *445*(1), 51–56.

Memarzadeh, G., & Keynia, F. (2020). A new short-term wind speed forecasting method based on fine-tuned LSTM neural network and optimal input sets. *Energy Conversion and Management*, *213*, 112824.

Muhamed Ali, A., Zhuang, H., Ibrahim, A. K., Wang, J. L., & Chérubin, L. M. (2022). Deep learning prediction of two-dimensional ocean dynamics with wavelet-compressed data. *Frontiers in Artificial Intelligence*, *5*, 923932.

Olah, C. (2015). *Understanding LSTM Networks*. https://colah.github.io/posts/2015-08-Understanding-LSTMs/

O'Malley, T., Bursztein, E., Long, J., Chollet, F., Jin, H., Invernizzi, L., & others. (2019). *KerasTuner*.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., & Dubourg, V. (2011). Scikit-learn: Machine learning in Python. *The Journal of Machine Learning Research*, *12*, 2825–2830.

Perkins, S. E., Pitman, A. J., Holbrook, N. J., & McAneney, J. (2007). Evaluation of the AR4 Climate Models' Simulated Daily Maximum Temperature, Minimum Temperature, and Precipitation over Australia Using Probability Density Functions. *Journal of Climate*, *20*(17), 4356–4376. https://doi.org/https://doi.org/10.1175/JCLI4253.1

Preston-Werner, T. (2013). Semantic Versioning 2.0. 0. *L\'\inea]. Available: Http://Semver. Org*.

Puertos del Estado. (2020). *CONJUNTO DE DATOS: REDMAR*. https://bancodatos.puertos.es/BD/informes/INT_3.pdf

Qian, P., Feng, B., Liu, X., Zhang, D., Yang, J., Ying, Y., Liu, C., & Si, Y. (2022). Tidal current prediction based on a hybrid machine learning method. *Ocean Engineering*, *260*, 111985.

Ratnam, J. V, Behera, S. K., Nonaka, M., Martineau, P., & Patil, K. R. (2023). Predicting maximum temperatures over India 10-days ahead using machine learning models. *Scientific Reports*, *13*(1), 17208.

Roelvink, J. A., & Van Banning, G. (1995). Design and development of DELFT3D and application to coastal morphodynamics. *Oceanographic Literature Review*, *11*(42), 925.

Rupani, M. (2024). *Desarrollo de metodologías basadas en aprendizaje automático para el modelado hidrodinámico*. https://github.com/mirkorupani/TFM

Saha, D., Deo, M. C., Joseph, S., & Bhargava, K. (2016). A combined numerical and neural technique for short term prediction of ocean currents in the Indian Ocean. *Environmental Systems Research*, *5*, 1–14.

Salman, A. G., Heryadi, Y., Abdurahman, E., & Suparta, W. (2018). Single layer & multi-layer long short-term memory (LSTM) model with intermediate variables for weather forecasting. *Procedia Computer Science*, *135*, 89–98.

Sarkar, D., Osborne, M. A., & Adcock, T. A. A. (2019). Spatiotemporal prediction of tidal currents using Gaussian processes. *Journal of Geophysical Research: Oceans*, *124*(4), 2697–2715.

Thiria, S., Sorror, C., Archambault, T., Charantonis, A., Bereziat, D., Mejia, C., Molines, J.-M., & Crépon, M. (2023). Downscaling of ocean fields by fusion of heterogeneous observations using deep learning algorithms. *Ocean Modelling*, *182*, 102174.

Toth, Z. (1989). Long-range weather forecasting using an analog approach. *Journal of Climate*, *2*(6), 594–607.

Van den Dool, H. M. (2007). *Empirical methods in short-term climate prediction*. Oxford University Press.

Virtanen, P., Gommers, R., Oliphant, T. E., Haberland, M., Reddy, T., Cournapeau, D., Burovski, E., Peterson, P., Weckesser, W., & Bright, J. (2020). SciPy 1.0: fundamental algorithms for scientific computing in Python. *Nature Methods*, *17*(3), 261–272.

Wang, J., Fonseca, R. M., Rutledge, K., Martín-Torres, J., & Yu, J. (2020). A hybrid statistical-dynamical downscaling of air temperature over Scandinavia using the WRF model. *Advances in Atmospheric Sciences*, *37*, 57–74.

Waskom, M. L. (2021). seaborn: statistical data visualization. *Journal of Open Source Software*, *6*(60), 3021. https://doi.org/10.21105/joss.03021

Willmott, C. J. (1981). On the validation of models. *Physical Geography*, *2*(2), 184–194.

Zhang, K., Wang, X., Wu, H., Zhang, X., Fang, Y., Zhang, L., & Wang, H. (2022). Study of the Performance of Deep Learning Methods Used to Predict Tidal Current Movement. *Journal of Marine Science and Engineering*, *11*(1), 26.

Zhao, Z., & Giannakis, D. (2016). Analog forecasting with dynamics-adapted kernels. *Nonlinearity*, *29*(9), 2888.