

**Universidad Internacional de La Rioja (UNIR)**

**ESIT**

**Máster Universitario en Inteligencia Artificial**

Prototipo experimental de IA  
para modelos biogeoquímicos e  
Hidrodinámicos  
oceanográficos en centros de  
cultivo para predicción de  
variables e impacto ambiental.

**Trabajo Fin de Máster**

**Presentado por:** Godoy Bonillo, Giocrisrai

**Director/a:** Emmi, Luis

Ciudad: Puerto Montt

Fecha: 03/03/2021

## Resumen

Este trabajo desarrolla la interacción de algoritmos de inteligencia artificial con modelos oceanográficos. Para ello se utilizo un piloto experimental, para la predicción de las series temporales de diferentes variables que pueden generar impactos ambientales en el ecosistema marítimo. Fue necesario apoyarse en el estudio de línea base oceanográfico, para obtener datos al sur de Chile y generar el estudio en una concesión marítima. El procesamiento de los datos provenientes del modelo Oceanográfico CROCO-OCEAN, se llevo a cabo mediante una transformación de las series temporales provenientes del formato NetCDF a CSV, permitiendo una mejor manipulación de los datos. Los resultados obtenidos muestran una comparación del rendimiento de diferentes redes neuronales: densas (DNN), convolucionales (CNN) y recurrentes (RNN), lo que permitió determinar cual método obtiene mejores métricas en problemáticas de series temporales, con la finalidad de poder monitorear y predecir las variables que afectan en el centro de cultivo.

**Palabras Clave:** Aprendizaje Profundo, Series Temporales, Redes Neuronales Recurrentes, Modelo Oceanográfico, Centro de Cultivo de Salmones.

## Abstract

This work develops the interaction of artificial intelligence algorithms with oceanographic models. For this, an experimental pilot was used to predict the time series of different variables that can generate environmental impacts on the maritime ecosystem. It was necessary to rely on the oceanographic baseline study to obtain data in southern Chile and generate the study in a maritime concession. The processing of the data from the CROCO-OCEAN Oceanographic model was carried out through a transformation of the time series from the NetCDF to CSV format, allowing a better manipulation of the data. The results obtained show a comparison of the performance of different neural networks: dense (DNN), convolutional (CNN) and recurrent (RNN), which allowed determining which method obtains better metrics in time series problems. In order to be able to monitor and predict the variables that affect the cultivation center.

**Keywords:** Deep Learning, Time Series, Recurrent Neural Networks, Oceanographic Model, Salmon Culture Center.

# Índice de contenidos

1. Introducción .....	1
1.1 Motivación.....	5
1.2 Planteamiento del trabajo .....	7
1.3 Estructura de la memoria.....	8
2. Contexto y estado del arte.....	10
2.1 Acuicultura chilena.....	10
2.2 Desafíos medioambientales en la acuicultura .....	11
2.3 Inteligencia Artificial en la acuicultura .....	12
2.4 IoT (Internet of Things) en la acuicultura .....	13
2.5 Sistemas híbridos en la acuicultura. ....	14
2.6 Inteligencia artificial.....	15
2.7 Redes neuronales artificiales.....	16
2.8 Aprendizaje profundo (Deep Learning).....	18
2.9 Red Neuronal convolucionales CNN .....	19
2.10 Red Neuronal recurrente RNN.....	20
3. Objetivos y metodología de trabajo .....	23
3.1. Objetivo general.....	23
3.2. Objetivos específicos .....	23
3.3. Metodología del trabajo .....	23
3.3.1. Etapa 1: Comprensión del negocio.....	24
3.3.2. Etapa 2: Enfoque analítico.....	24
3.3.3. Etapa 3: Requisitos de datos .....	25
3.3.4. Etapa 4: Recopilación de datos .....	25
3.3.5. Etapa 5: Comprensión de datos .....	25
3.3.6. Etapa 6: Preparación de datos .....	25
3.3.7. Etapa 7: Modelado.....	25
3.3.8. Etapa 8: Evaluación .....	26

3.3.9. Etapa 9: Implementación .....	26
3.3.10. Etapa 10: Retroalimentación.....	26
4. Descripción detallada del experimento.....	27
4.1 Definición de la problemática y descripción del prototipo como herramienta de solución.	27
4.2 Técnicas estadísticas y aprendizaje automático.....	27
4.3 Recursos informáticos .....	28
4.4 Comprensión del negocio .....	29
4.5 Enfoque analítico .....	30
4.6 Requisitos de datos .....	31
4.7 Recopilación de los datos .....	32
4.8 Comprensión de datos primera iteración .....	33
4.9 Preparación de datos primera iteración .....	37
4.10 Comprensión de datos segunda iteración .....	38
4.11 Preparación de datos segunda iteración .....	42
4.12 Modelado .....	48
4.12.1 Modelos de un solo paso: .....	48
4.12.1.1 Base:.....	48
4.12.1.2 Modelo lineal:.....	51
4.12.1.3 Denso:.....	53
4.12.1.4 Denso de varios pasos: .....	54
4.12.1.5 Red neuronal convolucionales:.....	57
4.12.1.6 Red neuronal recurrente: .....	59
4.12.2 Modelos de múltiples salidas: .....	60
4.12.2.1 Base:.....	61
4.12.2.2 Dense:.....	61
4.12.2.3 Red neuronal recurrente RNN: .....	62
4.12.2.4 Avanzado: conexiones residuales .....	62

4.12.3 Modelos de varios pasos: .....	63
4.12.3.1 Líneas base múltiples pasos.....	65
4.12.3.2 Modelos de disparo único con múltiples pasos .....	67
4.12.3.3 Lineal .....	67
4.12.3.4 Denso para múltiples pasos.....	68
4.12.3.65 Redes neuronales convolucionales CNN para múltiples pasos .....	70
4.12.3.6 Red neuronal recurrente RNN para múltiples pasos.....	71
4.12.3.7 Avanzado modelo auto regresivo. ....	72
4.12.3.8 Red neuronal recurrente auto regresivo. ....	73
5. Descripción de los resultados.....	75
6. Discusión .....	78
7. Conclusiones y trabajo futuro .....	80
7.1. Conclusiones .....	80
7.2. Líneas de trabajo futuro .....	82
8. Bibliografía.....	84
Anexos.....	91

## Índice de tablas

Tabla 1 LOS 10 PRINCIPALES EXPORTADORES E IMPORTADORES DE PESCADO Y PRODUCTOS PESQUEROS.....	10
Tabla 2 Coordenadas de trabajo .....	32
Tabla 3 Variables provenientes del modelo CROCO-OCEAN .....	34
Tabla 4 Dimensiones de las variables.....	36
Tabla 5 Variables dimensionales.....	36
Tabla 6 Variable temperatura configuración dimensional .....	37
Tabla 7 Variables con etiquetas modificadas incluyendo unidades .....	37
Tabla 8 Datos estadísticos del conjunto de datos .....	40
Tabla 9 Configuración de ventana para generar una salida de 24 horas a partir de 24 horas de entrada .....	44
Tabla 10 Configuración de ventana para generar una salida de una hora a partir de 6 horas de entrada .....	44
Tabla 11 Dimensiones de las ventanas ante el uso del método split_windows .....	45
Tabla 12 Ejemplo de estructura del dataset.....	48
Tabla 13 Configuración del WindowsGenerator para producir pares de un solo paso .....	48
Tabla 14 Estructura del dataset a partir de los conjuntos, entrenamiento, validación y prueba .....	48
Tabla 15 Datos del modelo de línea base (baseline) para una sola salida.....	49
Tabla 16 Ventana para la predicción de una hora en el futuro.....	49
Tabla 17 Estructura de entrada y salida del modelo baseline.....	50
Tabla 18 Estructura de entrada y salida del modelo lineal.....	51
Tabla 19 Datos que alimentan la función para compilar y entrenar el modelo.....	51
Tabla 20 Datos del entrenamiento y evaluación del modelo lineal .....	52
Tabla 21 Estructura de las ventanas para el modelo lineal.....	52
Tabla 22 Diseño del modelo dense secuencial para un solo paso .....	53
Tabla 23 Ventana para la predicción de una hora en el futuro de acuerdo con 3 horas de entrada .....	54

Tabla 24 Diseño del modelo multi_step_dense secuencial para un solo paso.....	55
Tabla 25 Configuración de ventanas de entrada para el enfoque multi-paso denso .....	56
Tabla 26 Dimensiones del modelo Conv model on conv_window .....	57
Tabla 27 Diseño del modelo convolucional para un solo paso .....	57
Tabla 28 Diseño del modelo convolucional para un solo paso. ....	59
Tabla 29 Datos del modelo de línea base (baseline) múltiples salidas un paso.....	61
Tabla 30 Datos del modelo dense múltiples salidas un paso.....	61
Tabla 31 Datos del modelo recurrente LSTM múltiples salidas un paso. ....	62
Tabla 32 Datos del modelo recurrente LSTM múltiples salidas un paso. ....	63
Tabla 33 Configuración del objeto Window dado 24 horas pasadas para predecir las próximas 24 horas futuras.....	64
Tabla 34 Datos del modelo de línea base (Last) repitiendo el último paso de entrada.....	65
Tabla 35 Datos del modelo de línea base (Repeat) repitiendo el día anterior asumiendo que mañana será igual.....	66
Tabla 36 Datos del modelo lineal para múltiples pasos .....	67
Tabla 37 Datos del modelo denso para múltiples pasos.....	68
Tabla 38 Datos del modelo denso para múltiples pasos.....	70
Tabla 39 Datos del modelo multi LSTM para una salida.....	71
Tabla 40 Datos del modelo autoregresivo RNN para una salida. ....	73
Tabla 41 Rendimiento de los modelos de un solo paso para una salida .....	75
Tabla 42 Rendimiento de los modelos de un solo paso para salidas múltiples .....	76
Tabla 43 Rendimiento de los modelos de varios pasos para una salida .....	77

## Índice de figuras

Figura 1 Producción acuícola según la FAO.(FAO Fisheries & Aquaculture - Acuicultura, 2021.) .....	1
Figura 2 Modelo ROMS (Regional Ocean Modeling System (ROMS), 2021).....	3
Figura 3 Modelo WRF(Weather Research and Forecasting Model (WRF)   NCAR Research Applications Laboratory   RAL, 2021.).....	4
Figura 4 Distinción entre IA como ciencia e IA como ingeniería(Peñaflie & Ávila, 2007).....	16
Figura 5 Neurona biológica (Inspiración Biológica de Las Redes Neuronales Artificiales   by Mario Campos Soberanis   SoldAI   Medium, 2021.) .....	16
Figura 6 Neurona artificial(Redes Neuronales Artificiales   Tablada   Revista de Educación Matemática, n.d.).....	17
Figura 7 Capas de red neuronal (Marín Diazaraque, 2007).....	17
Figura 8 Recorrido hacia el Deep Learning (Association of Artificial Intelligence, Machine Learning and Deep...   Download Scientific Diagram, n.d.).....	19
Figura 9 Red neuronal convolucionales (Albelwi & Mahmood, 2017).....	20
Figura 10 Red neuronal recurrente (Regueiro, 2018) .....	21
Figura 11 Caracterización de una de las etapas de las redes LSTM (De Lucio, n.d.) .....	22
Figura 12 Metodología Fundamental para la Ciencia de Datos (Rollins, 2020.).....	24
Figura 13 Batimetría del sector de estudio al sur de Chile Fuente: Elaboración propia.....	31
Figura 14 Grilla seleccionada para el estudio Fuente: Elaboración propia .....	32
Figura 15 Características a lo largo del tiempo completo Fuente: Elaboración propia .....	39
Figura 16 Características a lo largo del tiempo hasta 480 datos Fuente: Elaboración propia .....	39
Figura 17 Matriz de correlación del conjunto de datos completo Fuente: Elaboración propia .....	41
Figura 18 Matriz de correlación respecto a la temperatura Fuente: Elaboración propia.....	42
Figura 19 Chequeo de distribución de las funciones Fuente: Elaboración propia .....	43
Figura 20 Alineación de entradas, etiquetas y futuras predicciones en función al tiempo para la temperatura Fuente: Elaboración propia .....	46

Figura 21 Alineación de entradas, etiquetas y futuras predicciones en función al tiempo para la salinidad Fuente: Elaboración propia .....	47
Figura 22 Modelo baseline de un solo paso Fuente: Elaboración propia.....	50
Figura 23 Modelo lineal de un solo paso Fuente: Elaboración propia.....	52
Figura 24 Peso asignado a modelo lineal de un solo paso Fuente: Elaboración propia.....	53
Figura 25 Dado 3h de entrada, predice 1h en el futuro Fuente: Elaboración propia.....	55
Figura 26 Dado 3h de entrada, predice 1h en el futuro multi-paso denso Fuente: Elaboración propia.....	56
Figura 27 Predicción de una red neuronal convolucionales teniendo en cuenta 3 pasos de tiempo anteriores Fuente: Elaboración propia.....	58
Figura 28 Predicción de una red neuronal recurrente 24h de datos a la vez Fuente: Elaboración propia.....	60
Figura 29 Generación de predicción de múltiples puntos a futuro Fuente: Elaboración propia .....	65
Figura 30 Modelo de línea base (Last) de múltiples pasos Fuente: Elaboración propia.....	66
Figura 31 Modelo de línea base repetir el día anterior de múltiples pasos Fuente: Elaboración propia.....	67
Figura 32 Modelo lineal para múltiples pasos Fuente: Elaboración propia.....	68
Figura 33 Modelo denso para múltiples pasos Fuente: Elaboración propia.....	69
Figura 34 Modelo red neuronal convolucionales para múltiples pasos Fuente: Elaboración propia.....	71
Figura 35 Modelo red neuronal recurrente para múltiples pasos a partir de acumular 24h Fuente: Elaboración propia .....	72
Figura 36 Modelo red neuronal recurrente autoregresiva múltiples pasos Fuente: Elaboración propia.....	74
Figura 37 Actuación de modelos de un solo paso una salida Fuente: Elaboración propia....	75
Figura 38 Actuación de modelos de un solo paso para múltiples salidas Fuente: Elaboración propia.....	76
Figura 39 Actuación de modelos de varios pasos para una salida Fuente: Elaboración propia .....	77

## 1. Introducción

Para dar inicio a la investigación es importante mencionar que dentro del rubro de producción de alimentos tenemos la acuicultura, que es definida por la subsecretaría de pesca y acuicultura del Gobierno de Chile como el conjunto de actividades, técnicas y conocimientos de cultivo de especies acuáticas vegetales y animales (*¿Qué Es Acuicultura?* - SUBPESCA. Orientacion, 07/12/2020). Esta definición presentada es tomada en consideración debido a que la presente investigación se desarrollara con información obtenida de una empresa chilena del rubro acuícola.

Se debe señalar que este rubro es una importante actividad económica de producción de alimento para el país. Materias primas de uso industrial que van en crecimientos según la FAO (Food and Agriculture Organization) (FAO Fisheries & Aquaculture - Acuicultura, 2020) por lo tanto, se muestra la contribución de la acuicultura mundial a la producción pesquera mundial la que alcanzó el 46,0 % en 2018, frente al 25,7 % en 2000, y el 29,7 % en el resto del mundo, excepto China, quien en comparación obtuvo el 12,7 % en 2000.

Cabe destacar que dentro de las diferentes actividades acuícolas, Chile es el segundo país exportador y productor de salmón a nivel mundial detrás de Noruega (La Historia de La Acuicultura y El Salmón En Chile, 2021).

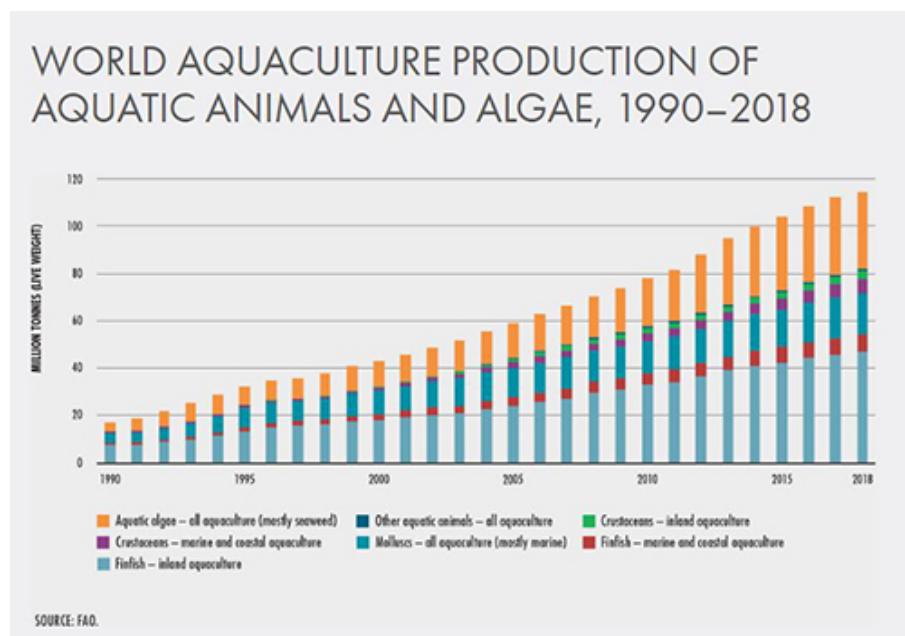


Figura 1 Producción acuícola según la FAO.(FAO Fisheries & Aquaculture - Acuicultura, 2021.)

Siguiendo en el ámbito chileno, una de las realidades del país es que en la actualidad el Estado está entregando concesiones (derechos por períodos determinados) a empresas para

<sup>1</sup> Prototipo experimental de IA para modelos biogeoquímicos e Hidrodinámicos oceanográficos en centros de cultivo para predicción de impactos ambientales

la explotación del recurso natural a través de las concesiones acuícolas. Se debe añadir a esto como parte importante que los espacios ocupados son capaces de generar un alto grado de contaminación (Espinosa Plascencia & Del Carmen Bermúdez Almada, 2012.) dado a la cantidad de desechos de alimentos no consumidos y otros factores que se depositan en el fondo marino. Por lo tanto, se debe tener en consideración que los factores de contaminación, hacen inviables la continuidad del negocio en las diferentes concesiones ya otorgadas, lo que genera consecuencias negativas tanto para el medioambiente, como para las empresas productoras del rubro acuícola, ya que a consecuencia de esto deben migrar a otros sectores y esperar a que el Estado Chileno les otorgue nuevas concesiones y lugares para retomar la producción, por otro lado se debe dar tiempo para que los espacios ya utilizados y contaminados puedan recuperar las condiciones óptimas para un nuevo ciclo productivo, lo cual es un tiempo y escenario incierto desde el punto de vista natural, ya que no se sabe con exactitud cuánto demoraran en recuperarse los espacios marítimos.

Cuando se trata de fiscalización por parte de los organismos reguladores, estos deben supervisar que se mantengan los parámetros óptimos para la continuidad de la producción, proceso que verifican visitando los centros de cultivo cada cierto periodo de tiempo para analizar las variables ambientales y a partir de esta evaluación permitir la continuidad operativa o por el contrario aplicar sanciones por el mal estado del espacio natural.

Cabe añadir que se realizan una diversidad de estudios dentro de una DIA (Declaración de Impacto Ambiental) (SEA Chile, 2021) o en su defecto una EIA (Estudio de Impacto Ambiental), según el nivel de impacto que producirá el proyecto y de acuerdo a los estatutos legales establecidos (Solicitud de Concesión de Acuicultura - SUBPESCA Tramites, n.d.), para que la concesión pueda producir. Estos estudios permiten dar una característica inicial en base a la cantidad de biomasa (salmones, mariscos, entre otros) que pueden producir de acuerdo con las modelaciones oceanográficas e hidrodinámicas del sector, además estos estudios pueden ser utilizados si se desea ampliar biomasa en las diferentes concesiones acuícolas. Sin embargo, este proceso al ser esporádico no permite un monitoreo constante que permita a los productores estar en alerta y así mantener tanto las condiciones necesarias para la operatividad como el buen trato con el medio ambiente.

Por lo anteriormente indicado, se busca dar una posible respuesta a esta necesidad de carácter ambiental y productiva, planteando un nuevo prototipo de inteligencia artificial, el cual se apoyará de la utilización de los recursos tecnológicos actuales, es decir, se pueden considerar los datos oceanográficos del sector con las características de los modelos ya existentes, trabajando con softwares de modelación como el ROMS (Regional Ocean Modeling System) (Lee, 2000) y el CROCO-OCEANS (Coastal and Regional Ocean

Community Model) que es una actualización del ROMS (Croco – Coastal and Regional Ocean COmmunity Model, 2021.), estos softwares son alimentados con información de los sectores meteorológicos y oceanográficos.

A continuación, a modo de ejemplo, se muestra la Figura 2, que hace referencia a la utilización del primer software mencionado. Este esquema nos presenta la recopilación de información que caracteriza un determinado espacio oceanográfico del sistema ROMS.

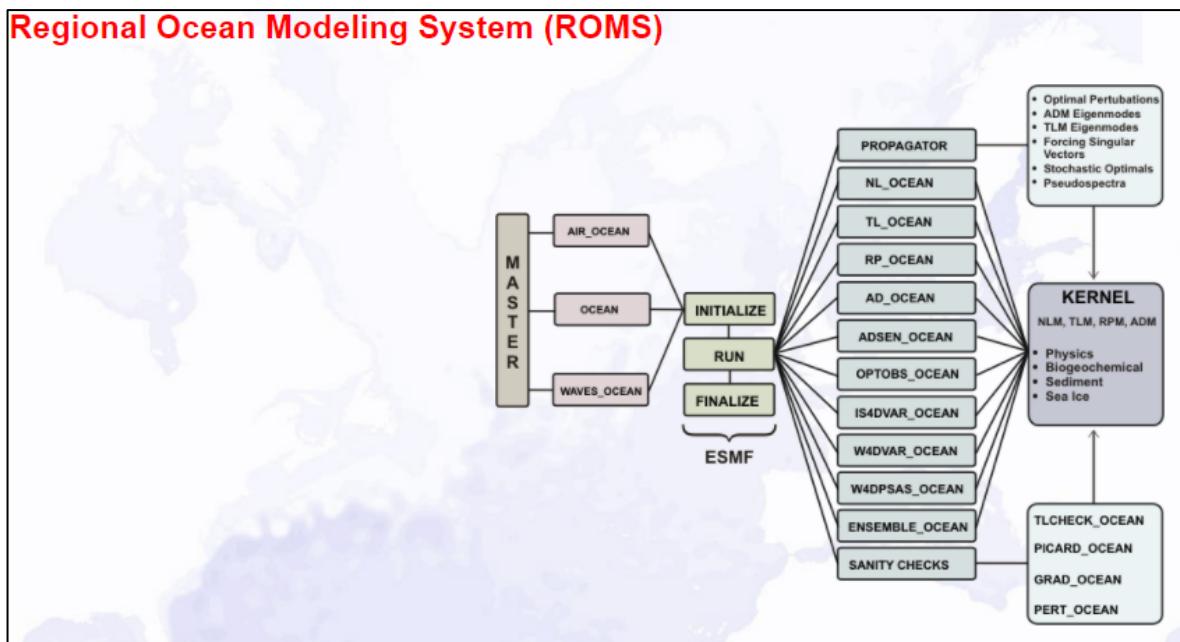
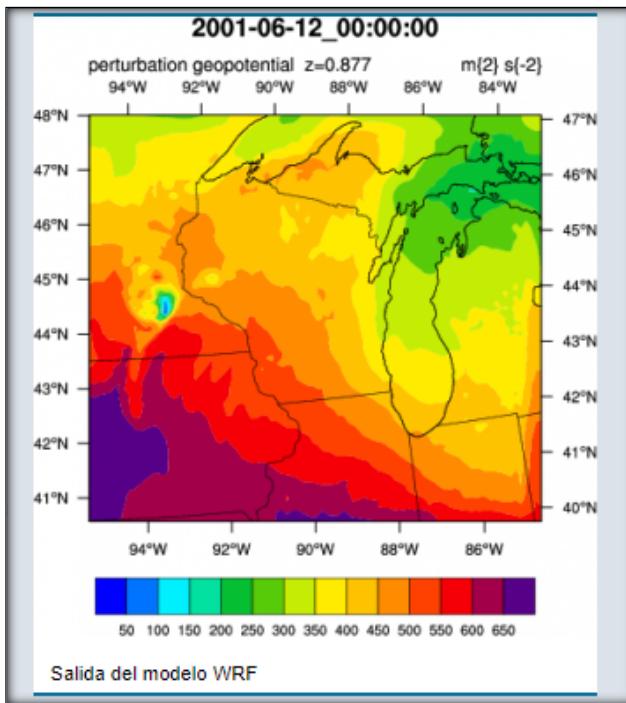


Figura 2 Modelo ROMS (Regional Ocean Modeling System (ROMS), 2021)

Para el prototipo igual se pueden considerar los modelos WRF (weather research and forecasting) que están disponibles en la web de libre acceso (Weather Research and Forecasting Model (WRF) | NCAR Research Applications Laboratory | RAL, 2021.), lo que permite tener una línea base del sector productivo, por lo tanto, una vez obtenida la información de los valores que provienen de los modelos mencionados, estos son ingresados al software de modelación para la dispersión del carbono y de los sólidos totales.

Podemos visualizar en la Figura 3 un ejemplo del modelo WRF el cual es un sistema de predicción meteorológica numérica de meso escala de próxima generación, diseñado para satisfacer las necesidades tanto de investigación atmosférica como de predicción operativa. Cuenta con dos núcleos dinámicos, un sistema de asimilación de datos y una arquitectura de software que facilita el cálculo paralelo y la extensibilidad del sistema. El modelo sirve para una amplia gama de aplicaciones meteorológicas en escalas de decenas de metros a miles de kilómetros.



*Figura 3 Modelo WRF(Weather Research and Forecasting Model (WRF) | NCAR Research Applications Laboratory | RAL, 2021.)*

Este proceso generado por los modelos meteorológicos WRF, oceanográficos CROCO-OCEAN, permiten la estimación del comportamiento del centro de cultivo, de acuerdo con condiciones ambientales recopiladas en periodos de tiempo establecidos. Algunas de las variables que se obtienen a través de información recolectada por instrumentación de campo y que ayuda a contrastar los modelos, son los datos de corrientes tanto velocidad como dirección del viento, pH, salinidad, oxígeno disuelto, turbidez, temperatura, etc.

Se desea aprovechar la información existente de estos modelos para luego poder integrarla en un sistema de inteligencia artificial que lleve a generar un sistema avanzado de acuicultura sostenible.

Para realizar el experimento, se utilizarán las salidas de información que proporcionan estos modelos, ya que es una fuente valiosa de información que generan el conjunto de datos necesarios para visualizar el comportamiento de un sistema productivo para la acuicultura de salmonidos.

Con este prototipo experimental de inteligencia artificial, se pretende incentivar la instalación de instrumentación de campo, que permita generar información del centro de cultivo, ya que este tipo de ecosistema necesita del monitoreo constante utilizando soluciones IoT (internet of thing), para rentabilidad del negocio en sintonía con el medio ambiente (IoT En Acuicultura y Seguridad Alimentaria | Telcel Empresas, 2021).

La interacción de las variables de medición, junto con los datos de alimentación de los centros acuícolas, permitirán verificar la correlación de los diferentes modelos y de este modo monitorear el comportamiento ambiental del sector de estudio, esperando que sea capaz de predecir las series temporales de las variables críticas para la operación en el centro de cultivo, pudiendo garantizar la eficiencia del proceso productivo, contribuyendo a la disminución del impacto ambiental.

Para este prototipo experimental se tomarán los datos de la salida de información del sector de las Clarence ubicada al sur de Chile, en el cual se cuenta con información derivada de la campaña verano e invierno del modelo oceanográfico CROCO-OCEAN, proveniente de un sistema construido por ROMS\_AGRIF. Esta información incluye diferentes componentes atmosféricos, sedimentos, información tomada en terreno y datos que son obtenidos durante los períodos de medición en este sector. También se tomarán los datos de los instrumentos que fueron colocados en este sector por estos períodos estacionales para tener información real del sector en el cual se pretende hacer el prototipo experimental, entendiendo que se tiene una serie de datos a diferentes profundidades de acuerdo con la grilla geográfica de la batimetría del sector.

Se añadirá un abordaje de las ventajas y limitaciones, creando una ventana de oportunidad para gestionar sistemas híbridos de modelamiento a través de técnicas de inteligencia artificial.

Se realizarán pruebas con diferentes algoritmos de aprendizaje profundo, tales como las redes neuronales densas, redes convolucionales, redes recurrentes, etc. Dado que se pueda generar una comparación y obtener el modelo que presente un mejor rendimiento en el experimento a realizar, determinando cual red se adapta mucho mejor para la resolución de esta problemática, que tiene como desafío el poder determinar la predicción de series temporales de las variables oceanográficas que afectan ambientalmente y también la producción del centro de cultivo.

## 1.1 Motivación

Como se señala en los párrafos anteriores, se plantea el gran desafío de generar un prototipo experimental a raíz de la necesidad imperante que surge de obtener un sistema que se mantenga de forma constante en la producción acuícola previniendo futuros sucesos que pueden ser poco beneficiosos no solo para las empresas sino también para el medio ambiente.

En este rubro económico se evidencian falencias como; falta de análisis de los datos que provienen de los modelos oceanográficos, biogeoquímicos e hidrodinámicos, así como también en los datos productivos para una integración inteligente en la toma de decisiones para operar dentro de los centros de cultivo acuícola, y que a su vez se encuentre en consonancia con el medio ambiente, ya que es una variable fundamental para la continuidad del negocio.

En lo que respecta a las condiciones necesarias para la producción, se espera que no se tenga que cerrar un centro de cultivo o gestionar procesos de relocalización por el hecho de llegar a estados de inoperatividad, en algunos casos por condiciones ambientales perjudiciales debido a no tener una correcta orientación acorde con un sistema de alertas o recomendaciones para poder gestionar una mejor toma de decisiones que garanticen la continuidad operativa.

También puede ocurrir el surgimiento de anaerobiosis, este concepto se utiliza para identificar las condiciones del ambiente, el que puede presentar carencia de oxígeno para la vida marina. Según el decreto n°320 reglamento ambiental para la acuicultura del Gobierno de Chile (Decreto-320 14-DIC-2001 MINISTERIO DE ECONOMÍA, FOMENTO Y RECONSTRUCCIÓN, SUBSECRETARIA DE PESCA - Ley Chile - Biblioteca Del Congreso Nacional, 2021), define lo siguiente sobre la anaerobiosis: Condición que indica la ausencia de oxígeno disuelto en el agua intersticial de los primeros tres centímetros del sedimento. En el caso de sustratos duros, semiduros o sitios con profundidades superiores a 60 metros, las condiciones anaeróbicas se constatarán medidas a una distancia máxima de 3 metros desde el fondo.

Es por esto que, al realizar esta investigación, surgen las siguientes preguntas:

- ¿Cómo lograr la interacción entre los modelos oceanográficos e hidrodinámicos con algoritmos de IA?
- ¿Cómo prevenir los impactos ambientales a través de los sistemas predictivos en tiempo real?
- ¿De qué manera los sistemas predictivos pueden aportar en la eficiencia de la producción de los centros de cultivo?

Si bien son preguntas que presentan un desafío, también incrementan la motivación de querer lograr un sistema que apoye la labor acuícola y medioambiental o por lo menos lograr un acercamiento positivo con el prototipo experimental de IA.

La principal causa que genera las dificultades antes ya mencionadas es el hecho de que no se posea una instrumentación de campo que permita el monitoreo en tiempo real, pero para lograr superar esta falencia, es necesario apoyarse en los modelos existentes donde se extrae con información meteorológica del sector, incorporando mediciones históricas del lugar en períodos de tiempo. Se espera que tomando en cuenta estas variables, sea posible realizar una mejor toma de decisión, por lo tanto, es necesario integrar lo que ya existe con técnicas de inteligencia artificial y de este modo encaminar a una mejor utilización de los recursos.

El problema es totalmente relevante y motivante ya que a través del sistema actual se finalizan concesiones y se compromete cada día más la sustentabilidad del negocio, debido al impacto ambiental que se está produciendo, lo que conlleva a pérdidas millonarias en los cierres de centros de cultivos.

Los cierres de centros acuícolas se realizan debido a que los espacios donde se lleva a cabo la producción no pueden continuar con el trabajo a causa de la contaminación de las aguas, provocando daño al medio ambiente. Esto es bastante negativo pues es una de las principales fuentes de suministro laboral y alimenticio, por lo que es necesario contar con un sistema avanzado de acuicultura sostenible, aprovechando las técnicas de inteligencia artificial.

## 1.2 Planteamiento del trabajo

Para dejar en claro lo que se plantea abordar, hay que tomar en cuenta que en el área de la acuicultura chilena, se generan situaciones negativas que afectan una producción a largo plazo, esto suele darse debido a la alta contaminación que ejercen las empresas de esta área y que impacta en el medio ambiente, lo que claramente no solo trae consecuencias en la naturaleza sino que además repercute de forma negativa en las empresas acuícolas, pues en muchos casos, estas deben hacer abandono del sector donde se está produciendo, lo que causa una disminución de los ingresos, por traslados de sector, espera de nuevas concesiones o en algunos casos más extremos el cierre de las compañías del área.

Como se viene repitiendo, lo que se pretende es poder llegar a tener un sistema de acuicultura avanzada sostenible. Para esto es necesario el análisis de los datos como la integración de los diferentes modelos que permiten generar una orientación del comportamiento de un centro

de cultivo de acuerdo a las condiciones ambientales, logrando la interacción de los modelos hidrodinámicos y oceanográficos a través de inteligencia artificial permitiendo la predicción de impactos ambientales en el ecosistema marítimo en tiempo real, lo que puede llegar a anticipar las externalidades negativas que pueden surgir debido a acciones provenientes del centro de cultivo, como de las variables físico ambientales.

Por lo que se propone generar un prototipo experimental tomando la información del sector de las Clarence al sur de Chile, la que será aportada por la empresa llamada Gestión acuícola, marítima y ambiental limitada (GEOGAMA). Se tomarán en cuenta las líneas base e información levantada en terreno, esto permitirá gestionar un acercamiento del comportamiento de los centros de cultivo que se instalaran en el sector.

## 1.3 Estructura de la memoria

Se muestra a continuación una descripción breve del contenido de cada uno de los capítulos que incluye este documento:

- **Capítulo 1:** Pertenece a la Introducción de la memoria en la cual se comienza con la presentación de la temática que se pretende abordar en este trabajo, a este capítulo se le agregan las motivaciones que llevaron a plantear esta temática, junto a una descripción del planteamiento del problema, que permita una mayor compresión de los temas que se abordaran en los próximos capítulos expresándolos en la estructura que llevara el documento.
- **Capítulo 2:** Se describe y documenta contexto relacionado con el estado del arte, los ámbitos tecnológicos que sirven de base para el desarrollo del trabajo, contextualizando diferentes métodos, técnicas e investigaciones que han utilizado en problemas similares, mostrando una visión general de las líneas de trabajos actuales, que permitan partir en una buena fundamentación en la solución a experimentar.
- **Capítulo 3:** Se mencionan los objetivos y metodología a utilizar para desarrollar el presente trabajo. El objetivo general se desglosa en un conjunto de objetivos específicos, analizables por separado. Seguidamente se detalla la metodología utilizada para la construcción y despliegue analítico en el desarrollo del prototipo experimental, siendo este el soporte fundamental para poder alcanzar los objetivos propuestos.

- **Capítulo 4:** Se genera la descripción detallada del experimento, desarrollando las etapas de la metodología que permitan verificar la funcionalidad y aplicabilidad de este. Dichos experimentos, deben venir acompañados de un análisis profundo que incidirá en las ventajas y contingencias que se presentan en las posibles soluciones.
- **Capítulo 5:** En este capítulo se genera, la descripción de los resultados obtenidos en los experimentos de manera objetiva, sin valorar, ni justificar los resultados obtenidos durante el experimento.
- **Capítulo 6:** Luego de presentar los resultados de manera objetiva, se aportará a través de la discusión la relevancia de este, presentando posibles explicaciones ante las anomalías, haciendo énfasis en los datos que sean significativos para el análisis del experimento.
- **Capítulo 7:** Se recogen las conclusiones alcanzadas tras la realización del trabajo, junto con la evaluación del cumplimiento de los objetivos y su aplicabilidad. Se debe evaluar el objetivo principal, para continuar con los objetivos específicos. Finalmente se incorporan las líneas de trabajo futuro que podrían aportar valor al trabajo realizado.
- **Bibliografía:** Se encuentran las referencias bibliográficas que fueron consultadas y que permitieron dar sustento a cada uno de los capítulos.
- **Anexo:** Incluye documentos adicionales, que ayudan en la comprensión del desarrollo del experimento realizado.

## 2. Contexto y estado del arte

Es necesario poder abordar esta investigación tomando en consideración los trabajos, referencias e investigaciones cercanas al ámbito que se pretende desarrollar. Ya que estos pueden tener una alta relevancia para el desarrollo del TFM.

### 2.1 Acuicultura chilena

Hace bastantes años la industria salmonera chilena ha conseguido situarse como grandes exportadores a nivel mundial, siguiendo a Noruega, generando más de 60.000 empleos, operando en más de 70 mercados. (La Historia de La Acuicultura y El Salmón En Chile, 2021).

Chile se encuentra entre los principales exportadores de pescado y productos pesqueros tal como lo presenta la siguiente tabla:

Tabla 1 LOS 10 PRINCIPALES EXPORTADORES E IMPORTADORES DE PESCADO Y PRODUCTOS PESQUEROS

	2004 (En millones de USD)	2014 (En millones de USD)	APR (Porcentaje)
EXPORTADORES	Canadá	3.487	4.503
	Chile	2.501	5.854
	China	6.637	20.980
	Dinamarca	3.566	4.765
	Estados Unidos de América	3.851	6.144
	India	1.409	5.604
	Noruega	4.132	10.803
	Países Bajos	2.452	4.555
	Tailandia	4.060	6.565
	Viet Nam	2.444	8.029
<b>SUBTOTAL 10 PRINCIPALES</b>		<b>34.539</b>	<b>77.801</b>
<b>SUBTOTAL RESTO DEL MUNDO</b>		<b>37.330</b>	<b>70.346</b>
<b>TOTAL MUNDIAL</b>		<b>71.869</b>	<b>148.147</b>
			<b>7,5</b>
IMPORTADORES	Alemania	2.805	6.205
	China	3.126	8.501
	España	5.222	7.051
	Estados Unidos de América	11.964	20.317
	Francia	4.176	6.670
	Italia	3.904	6.166
	Japón	14.560	14.844
	Reino Unido	2.812	4.638
	República de Corea	2.250	4.271
	Suecia	1.301	4.783
<b>SUBTOTAL 10 PRINCIPALES</b>		<b>52.119</b>	<b>83.447</b>
<b>SUBTOTAL RESTO DEL MUNDO</b>		<b>23.583</b>	<b>57.169</b>
<b>TOTAL MUNDIAL</b>		<b>75.702</b>	<b>140.616</b>
			<b>6,4</b>

(HLPE-CSA-FAO, 2015).

La acuicultura en Chile se encuentra regulada ambientalmente, este proceso viene dado por el Derecho ambiental, en la que se determinan sistemas que permiten evaluar el impacto de los proyectos que se encuentren dentro de sus procesos operativos o próximos a operar (Bermúdez Soto, 2007).

Debido al aumento de la producción acuícola chilena, por un periodo de transformación tanto a nivel tecnológico como regulatorio debido a la alta producción, existe un control estatal sobre las licencias y regulaciones relacionadas al cultivo del salmón (Fuentes Olmos, 2014). Sin embargo, los ambientalistas y habitantes están en conflicto, ya que existen algunas consecuencias negativas en el ambiente debido a su producción, las que ocasionan altos grados de contaminación en la vida marina (*¿Revolución Azul? El Impacto Regional de La Acuicultura Del Salmón En Chile, 2020*).

## 2.2 Desafíos medioambientales en la acuicultura

La producción acuícola va en aumento, lo que genera que se planteen soluciones ante los desafíos biológicos, económicos y las necesidades de ser cuidadosos con el medio ambiente. Se debe aprovechar los sistemas emergentes automatizados, que ayuden en temas de monitoreo de la biomasa, alimentación y gestión de la operación dentro de los centros de cultivo (Føre et al., 2018).

A través de los años se han ido incorporando nuevos elementos con el objetivo de lograr aminorar los impactos ambientales de la acuicultura, así como para generar automatización y mejora en los procesos manufactureros, usando como ejemplo investigaciones realizadas en otros rubros productivos.

Los sistemas de acuicultura se encuentran utilizando nuevas tecnologías para poder disminuir los peligros ambientales, tomando como experiencia los procesos agrícolas, generando una ingeniería acuícola de precisión, aprovechando la visión por computadora, una red de sensores, robótica y finalmente interpretación para la toma de decisiones, de este modo aumenta la producción y mejora la calidad de los productos (Antonucci & Costa, 2020).

El surgimiento de las tecnología del aprendizaje profundo han generado resultados favorables en diferentes campos en los cuales se incluye la acuicultura, creando una serie de desafíos en el procesamiento de los datos, en esta publicación nos comentan la clasificación de las especie, comportamiento, alimentación, estimación de biomasa y predicción de la calidad del agua, también se muestra el análisis de los rendimiento de los algoritmos ya que se requiere de una gran cantidad de etiquetado de datos, sin embargo, el aprendizaje profundo proporciona una forma de abordar estos datos complejos de la acuicultura, haciendo más fuerte el apoyo en la implementación de aplicaciones inteligentes (X. Yang et al., 2020).

Para lograr un buen almacenamiento de información es necesaria la colaboración de varios actores que trabajen en el rubro.

Dentro de la investigación oceanográfica, esta involucra un esfuerzo multidisciplinario en la que aplica poder extraer la mayor cantidad de datos in situ, aprovechando la teledetección. De igual forma se cuentan con repositorios con información en línea, que se deben integrar de manera automatizada, lo que representa un desafío clave en la integración de estos datos oceanográficos (Sagi et al., 2020).

No se debe negar que los desafíos o barreras que se van presentando a través del tiempo han servido de motivación para crear formas de protección para el medio ambiente. Poder mantener un ambiente ecológico con buena calidad del agua en la acuicultura, ha generado una nueva tendencia en la acuicultura inteligente para poder llevar a cabo el seguimiento de predicción, alertas y control de riesgos en tiempo real de los factores fisicoquímicos. (Hu et al., 2020).

## 2.3 Inteligencia Artificial en la acuicultura

Los avances científicos y aportes desde el área agrícola han servido de base para incentivar la incorporación de tecnologías al sector acuícola.

La automatización utilizando la inteligencia artificial en la acuicultura proveniente del rubro agrícola, puede ayudar en el control de la calidad del agua, procesamiento y alimentación, además de mantener la sostenibilidad en la pesca. Dentro del artículo, (Chrispin et al., 2020). Comenta que se puede reducir costes hasta un 30% con un sistema de control de producción utilizando la inteligencia artificial.

Los sistemas de acuicultura nuevos permiten tener en cuenta las condiciones ambientales y las variables de entrada al sistema, tales como; el agua, oxígeno, temperatura, velocidad de alimentación, pH y otros factores, que a través de inteligencia artificial podrán generar eficiencia en el proceso. Dado que se considera que es necesaria la interacción y compatibilidad en el software y en el hardware lo que nos puede llevar a tener un sistema mucho más confiable a través del uso de esta tecnología (Lee, 2000).

Dentro de las debilidades encontradas en el rubro pesquero se puede decir lo siguiente:

A pesar del crecimiento que se ha generado en los últimos años en la industria acuícola, se muestra la volatilidad en los resultados dados, ya que no es usado el análisis de datos para el desarrollo de modelos computacionales capaces de generar un mejor abordaje en la toma de decisión dentro de la industria acuícola (Luna & Manuel, 2020). Estos nos entregan una amplia gama de posibilidades para generar bienestar humano según las tecnologías

avanzadas (TA) que se nos ofrecen de la mano de la IA, la cual se entiende como la capacidad de aprendizaje de las máquinas para tomar decisiones.

Es importante que, durante el proceso de toma de decisiones, se consideren factores que permitan mantener activa la producción y vida marina.

En el cultivo de especies acuícolas, uno de los factores relevantes es la calidad del agua, según se menciona en el estudio presentado por (Hernández et al., 2017), en el cual demuestra que el uso de las redes neuronales es una buena opción para la evaluación y detección de estados óptimos o no deseados para un buen manejo del agua en los centros de cultivo de camarones, tomando en consideración los parámetros importantes para el hábitat en el centro de cultivo como; la temperatura del agua, pH, oxígeno disuelto y salinidad. Utilizando un modelo computacional capaz de ver la relación dinámica del ecosistema en el cultivo de las especies.

Entre los parámetros más importantes para determinar la calidad de agua en los sistemas de acuicultura, nos encontramos con el oxígeno disuelto, para ello es importante poder generar las predicciones de series temporales, ya que en los años más recientes se han utilizado las redes neuronales artificiales para la estimación en este tipo de problemática (Carlos Julián Torres González & Martín Flores Albino José Juan Carbajal Hernández, 2017).

Se puede apreciar que los sistemas de visión artificial permitieron el desarrollo de un software para poder realizar el conteo de alevinos de truchas y con esto poder determinar el nivel de mortalidad, mejorando la eficiencia de medición, así como el cálculo de alimentación a suministrar en la conservación y mejora del recurso ocupado (RAMOS, 2018).

## 2.4 IoT (Internet of Things) en la acuicultura

Existe un prototipo y prueba basado en un sistema de monitoreo de variables obtenidas del agua acuícola, la cual es de suma importancia ya que, comparada con la agricultura, la primera es un área tecnológica un poco abandonada. Con este sistema se busca solucionar problemas como la calidad del agua, desperdicio de recursos y pérdidas. Este sistema propuesto realiza seguimiento a la calidad del agua basado en redes de sensores inalámbricos y en Internet of Things (IoT) (Encinas et al., 2017).

El proceso de integración de la acuicultura y el IoT es algo que se encuentra en una etapa inicial, pero al poder mezclarlos es fácil de implementar siendo asequible, esto permite el

procesamiento de los datos a través de inteligencia artificial, enfrentando desafíos importantes, como ser eficiente con el medio ambiente (Dupont et al., 2018).

La integración de sistemas de monitoreo usando el internet de las cosas conjunto con la inteligencia artificial, se plantea que debe ser llevada en 4 capas: sensores, comunicaciones, servicios y aplicaciones. Lo que permite la mejor toma de decisiones en base a las variables relacionadas en los sistemas acuapónicos para el desarrollo del servicio de inteligencia artificial dado los datos capturados por los sistemas de IoT.

Las innovaciones de la ingeniería acuícola de precisión, lleva a que se incorporen elementos tales como; la robótica, sensores del internet de las cosas en conjunto de sistemas que apoyen en la toma de decisiones como lo hace la inteligencia artificial (Antonucci & Costa, 2020). Debido a que con los procedimientos de IoT se comunican con sistemas que cooperan entre sí, en miras de aumentar y mejorar la producción de los productos acuícolas consiguiendo mejorar la calidad, ya que es más importante monitorear que controlar el proceso productivo.

## 2.5 Sistemas híbridos en la acuicultura.

Es necesario poder comprender los sistemas híbridos y la integración de modelos a través de inteligencia artificial, ya que esto nos permitirá saber como a través de la información, podemos generar sistemas predictivos en ambientes oceanográficos en tiempo real (Corchado Rodríguez & Aiken, 2002), Por lo tanto, es fundamental comprender estos sistemas de acuerdo a diferentes variables, debido a que son totalmente complejos e involucran el funcionamiento de los espacios para la acuicultura.

Además, es necesario conseguir una disminución de efectos negativos en el medio ambiente para lograr generar sistemas de acuicultura sostenible.

Dentro de los métodos híbridos de inteligencia artificial en modelos de predicción oceanográfica, se puede pronosticar los valores de los parámetros y comportamientos característicos en tiempo real del océano, siendo este un ambiente dinámico y complejo. Es una tarea ardua por lo que se desarrolló un método que se basa en un sistema integrado de razonamiento, utilizando redes de neuronas, en donde se presentan resultados del experimento, en los que el sistema operó en tiempo real en dicho entorno (Corchado Rodríguez & Aiken, 2002).

Cuando se refieren al uso de elementos de predicción interdisciplinarios de manera rápida y eficiente, entendiendo que se encuentran en un ambiente dinámico como lo es el océano, se

muestra que como objetivo científico está el acoplar los elementos físicos, biológicos y acústicos, mediante la construcción distribuida de grillas que permitan visualizar información, con el cual se pueda disminuir la incertidumbre de los modelos y sus diferentes parametrizaciones (Patrikalakis et al., 2004).

Poder determinar la calidad del agua en los sistemas de acuicultura es un factor fundamental dentro del rubro y de la investigación, debido a la no linealidad de los parámetros. Es utilizado el vector de soporte de regresión en conjunto con algoritmos genéticos, ya que demuestra una eficacia mucho mayor que de forma individual (S. Liu et al., 2013).

Los parámetros importantes para determinar la calidad del agua se encuentran en la concentración de sólido disuelto. (Eze & Ajmal, 2020) menciona que hacer un pronóstico a corto plazo, utilizando una serie de tiempo de los sistemas acuícolas mediante un método híbrido de redes de neuronas basadas en la descomposición empírica, mejora la calidad de los datos en los cuales se adoptaron índices de medición de los resultados, para pronosticar los valores de oxígeno disuelto.

Tal como se presenta en la investigación, el monitoreo ambiental es de gran importancia porque busca verificar sistemas que son capaces de identificar contaminantes. Esto se presenta en el prototipo de bajo costo del trabajo de investigación de (José de Jesús et al., 2016). En el cual se crea un sistema que permite identificar contaminantes en el ambiente.

## 2.6 Inteligencia artificial

La inteligencia artificial se puede definir como; la habilidad de los ordenadores para hacer actividades que normalmente requieren inteligencia humana (Rouhiainen, 2018). Siendo más específico, se puede decir que la inteligencia artificial, es la capacidad en las máquinas para usar algoritmos que permiten aprender de los datos y tomar decisiones.

El término de inteligencia artificial propuesto por J. McCarthy, fue aceptado por aunar en un solo cuerpo de doctrina, las metodologías, filosofías y ambiciones de aquellos que trabajan en computadora de problemas, es decir, con demostraciones de teoremas, razonamiento general, procesamiento simbólico, juegos de intelecto, puzzles, entre otros (M. Delgado, 1997).

La inteligencia artificial se puede visualizar tanto como ciencia y como ingeniería, ya que se intenta mezclar para darle sentido en ambos puntos. Desde la ciencia pueden ser vistas las intenciones, la semántica, emociones y desde el punto de vista más computacional o ingenieril; modelos, inferencias, operadores lógicos (Peñafiel & Ávila, 2007).

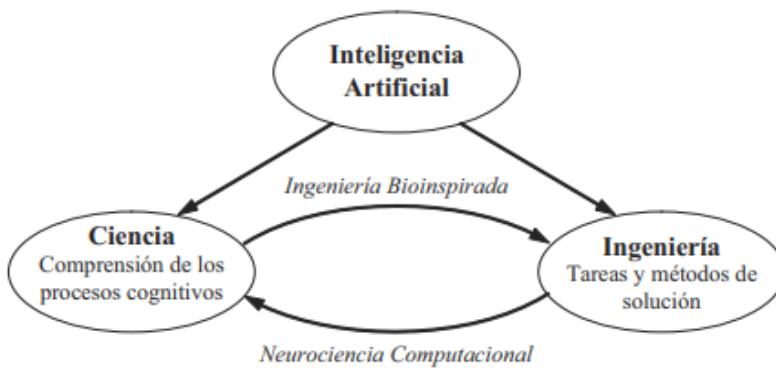


Figura 4 Distinción entre IA como ciencia e IA como ingeniería(Peñaflie & Ávila, 2007)

## 2.7 Redes neuronales artificiales.

Las redes neuronales artificiales nos ofrecen los medios para modelar de manera efectiva y eficiente problemas grandes y complejos. Son capaces de encontrar relaciones (patrones) de forma inductiva por medio de los algoritmos de aprendizaje basado en los datos existentes (Salas, n.d.). Se puede decir que son un método de resolver problemas, de forma individual o combinada con otros métodos, para aquellas tareas en las que el balance datos/conocimiento se inclina hacia los datos.

Las redes neuronales artificiales, están inspiradas en redes neuronales biológicas del cerebro humano (Basogain Olabe, 2005). Se constituyen por elementos que se comportan de manera similar a la neurona biológica al margen de parecerse. Presentan características propias del cerebro, en el cual podríamos tomar de ejemplo que aprenden de la experiencia, generalizando de conocimientos previos a ejemplos nuevos y abstraen características principales de una serie de datos.

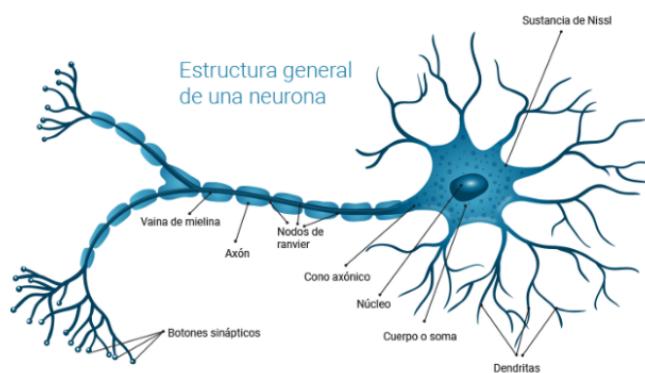
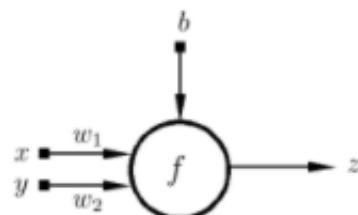


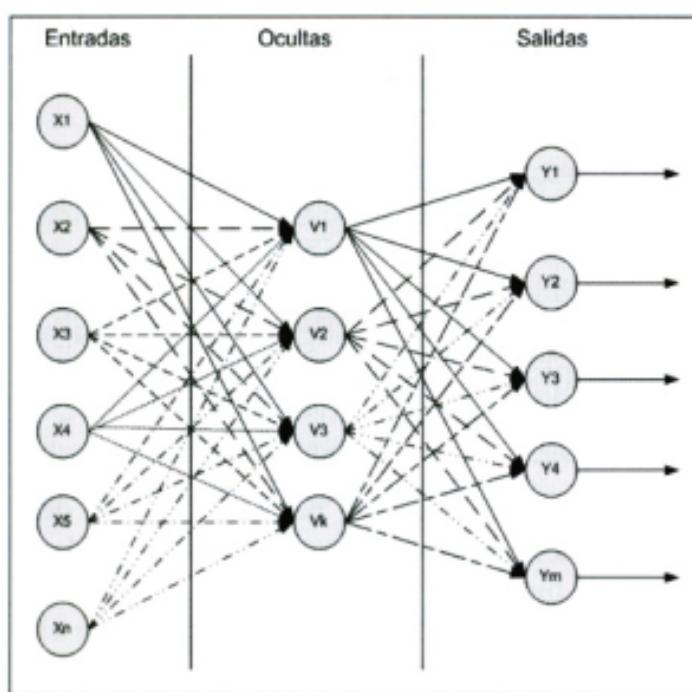
Figura 5 Neurona biológica (Inspiración Biológica de Las Redes Neuronales Artificiales | by Mario Campos Soberanis | SoldAI | Medium, 2021.)

El primer modelo matemático de una neurona artificial fue presentado en 1943. (*Redes Neuronales Artificiales | Tablada | Revista de Educación Matemática*, n.d.). El mismo consta de las entradas, los pesos sinápticos, el término aditivo, una función de activación y la salida.



*Figura 6 Neurona artificial(**Redes Neuronales Artificiales | Tablada | Revista de Educación Matemática*, n.d.)

La estructura de una red neuronal está compuesta por varios elementos procesales, en las que tenemos las neuronas de entradas, las ocultas, las salidas y las interconexiones (Marín Diazaraque, 2007). Las neuronas de entrada sólo dejan pasar la información, las neuronas ocultas reciben las entradas y tienen la función de proporcionar un mejor aprendizaje o separación de lo aprendido y las neuronas de salida se encargan de proporcionar la escapatoria del sistema indicado según el aprendizaje.



*Figura 7 Capas de red neuronal (Marín Diazaraque, 2007)*

Una de las características fundamentales de las redes neuronales es su adaptabilidad y su susceptibilidad de aprendizaje, a través de la modificación de los pesos de las interconexiones

en las diferentes neuronas (Lara Rosano, 2017). Entre las formas fundamentales de aprendizaje tenemos; el aprendizaje supervisado, en el cual se conoce el vector de respuesta correcto ante cada vector de entrada, y con la respuesta real se genera un vector de error, que actualiza los pesos de las interconexiones de la red, de manera que el error tienda a desaparecer. El aprendizaje no supervisado, utiliza los datos de entrenamientos no etiquetados previamente en el cual se comprimen los datos de entradas en ciertas categorías de clasificación.

El algoritmo de backpropagation o retropropagación a través del tiempo es una herramienta muy útil en el reconocimiento de patrones, modelización dinámica y control de sistemas en el tiempo (Werbos, 1990). Es un método muy utilizado en el aprendizaje supervisado, intentando adaptar la red neuronal artificial para que sus salidas reales se acerquen a algunas salidas objetivo para un conjunto de entrenamiento.

## 2.8 Aprendizaje profundo (Deep Learning)

Las observaciones oceanográficas están limitadas por las tasas de muestreo, mientras que los modelos oceánicos están limitados por una resolución finita con altos coeficientes de viscosidad y difusión (Ducournau & Fablet, 2017). Por lo que con el uso de los algoritmos de aprendizaje profundo, son capaces de replicar con éxito la variabilidad espacio-temporal (Jiang et al., 2018). Además como menciona (Bolton & Zanna, 2019) existen cuadriculas sin una disminución significativa en la precisión. Poder generar predicciones con varios pasos de tiempo en modelos climáticos, son programas informáticos muy complejos que representan la física de la atmósfera, que a través de las redes neuronales profundas abre nuevas posibilidades para el modelado climático como el oceanográfico (Scher, 2018).

En los últimos años se han producido grandes avances en la inteligencia artificial con las técnicas de aprendizaje profundo de redes neuronales artificiales, dentro de los campos que ha tenido alto impacto en los reconocimientos de imágenes, procesamiento de lenguaje natural y series temporales (Vázquez & Constable, 2019).

Esta tercera era de las redes neuronales comenzaron en 2006 de la mano de Geoffrey Hilton, el cual pudo desarrollar una forma mas eficiente de entrenar modelos de aprendizaje profundo, utilizando aprendizaje por capas (Muñiz, 2018), en donde la primera capa aprende características primitivas, que se envían a la siguiente capa, en la cual se reconocen características mas complejas entre cada capa haciéndolas cada vez mas complejas . Este avance permitió entrenar arquitecturas mas profundas que las utilizadas en las redes

neuronales convencionales, lo cual popularizó el término de aprendizaje profundo o deep learning, permitiendo que ante esta situación grandes empresas de tecnología estén apostando por el aprendizaje profundo para el desarrollo de sus aplicaciones.

En el año 2009, el grupo de Jürgen ganó una competencia internacional con una red neuronal profunda, convirtiéndose en la primera red neuronal y sistema de aprendizaje profundo en ganar una competencia de reconocimiento de patrones (Jhan & Arteaga, n.d.). El grupo siguió trabajando en 2012 con las GPUs, mostrando excelentes rendimientos y rompiendo récords mundiales en tasas de error hasta 0,37%.

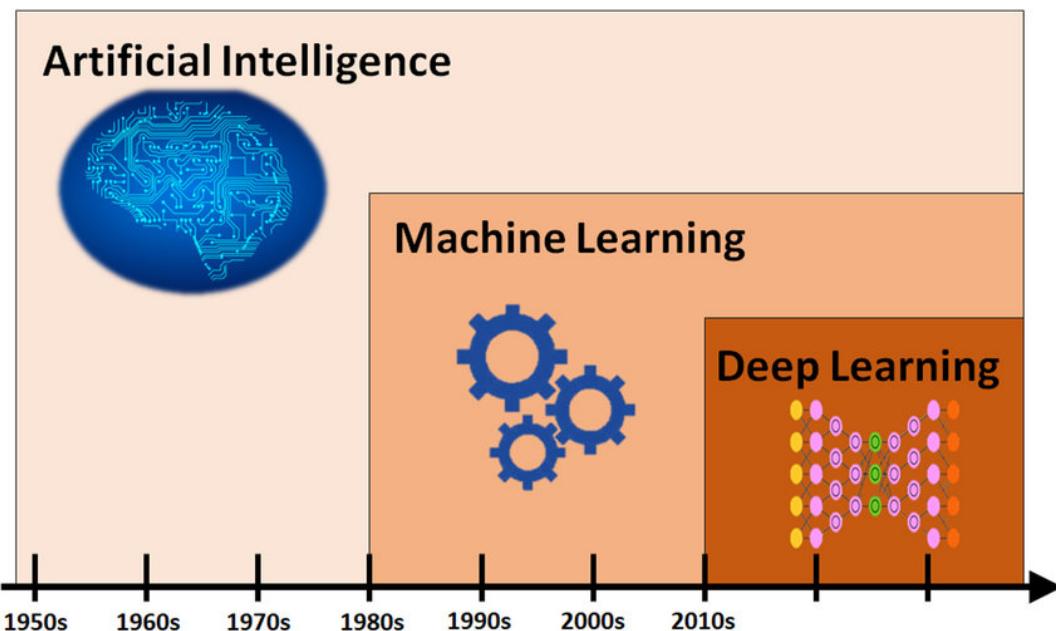


Figura 8 Recorrido hacia el Deep Learning (Association of Artificial Intelligence, Machine Learning and Deep... | Download Scientific Diagram, n.d.)

## 2.9 Red Neuronal convolucionales CNN

Las redes neuronales convolucionales son un tipo alternativo de red neuronal que se puede utilizar para reducir las variaciones y modelar las correcciones espectrales que existen en las señales (Sainath et al., 2013). Las redes neuronales convolucionales han logrado un rendimiento en muchas tareas de reconocimiento de patrones, sin embargo, el desarrollo de los modelos profundos de alta calidad, generalmente se basan en una cantidad sustancial de prueba y error, ya que todavía no hay una compresión clara de cuando y porque funciona un modelo profundo (M. Liu et al., 2017).

La red neuronal convolucionales (CNN) han mostrado un rendimiento ejemplar en varias competiciones relacionadas con la visión por computadora y procesamiento de imágenes,

dentro de las mas interesantes incluye; la clasificación y segmentación de imágenes, detección de objetos, procesamiento de video, procesamiento de lenguaje natural y reconocimiento de voz (Khan et al., 2020).

Los modelos de redes neuronales convolucionales tienen una estructura estándar que constan de capas convolucionales alternas y capas agrupadas, las ultimas capas son unas pequeñas cantidades de neuronas completamente conectadas, y la capa final si es un clasificador, lleva una función de activación softmax (Albelwi & Mahmood, 2017). La capa convolucional esta compuesta por un conjunto de kernels o filtros que se pueden aprender y que tiene como objetivo, extraer características locales de la entrada. En donde cada kernel se utiliza para calcular un mapa de características. Por lo general, se genera formando un producto escalar seguido de una función de activación no lineal.

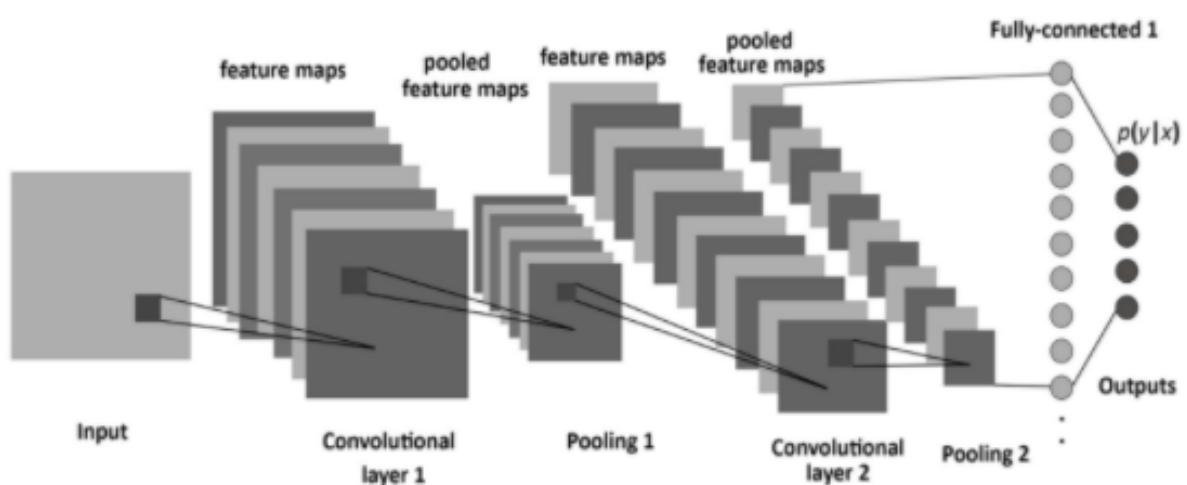


Figura 9 Red neuronal convolucionales (Albelwi & Mahmood, 2017)

Dentro de las ventajas de las redes neuronales convolucionales se puede decir que son buenas para poder aprender características en señales de series de tiempo, lo que permite poder automatizar el aprendizaje de funciones de algún sistema en particular (J. B. Yang et al., 2015).

## 2.10 Red Neuronal recurrente RNN

Las redes neuronales recurrentes (RNN) son modelos conexionistas de datos secuenciales que son naturalmente aplicables en el análisis del proceso de lenguaje natural (Irsoy & Cardie, 2014). Se han convertido recientemente en una popular elección para modelar secuencias de longitud variable, ya que pueden expresarse como una composición de múltiples capas no lineales cuando se despliegan en el tiempo, pudiendo ser utilizadas en predicción de series temporales (Pascanu et al., 2014).

Para las redes neuronales recurrentes, la función principal de las capas es introducir memoria, no un procesamiento jerárquico que es algo común en las distribuciones temporales jerárquicas, lo que hace es agregar nueva información en cada capa de interacción de la red y así pueda transmitir información para un número indefinido de actualizaciones de la red que generalmente van recibiendo entrada y produciendo la salida con cada paso de tiempo (Hermans & Schrauwen, 2013).

La red neuronal recurrente puede ser capaz de realizar una amplia variedad de tareas computacionales, incluyendo; el tratamiento de secuencias, la continuación de una trayectoria, la predicción no lineal y modelación de sistemas dinámicos (Bonet Cruz et al., 2007). También son conocidas como redes de espaciotemporales o dinámicas que intentan corresponder entre la secuencia de entrada y de salida que no son más que patrones temporales.

Se dice que una red es recurrente cuando en el flujo de datos entre las neuronas que consideramos de entradas y las que consideramos de salida existen retroalimentación (Regueiro, 2018).

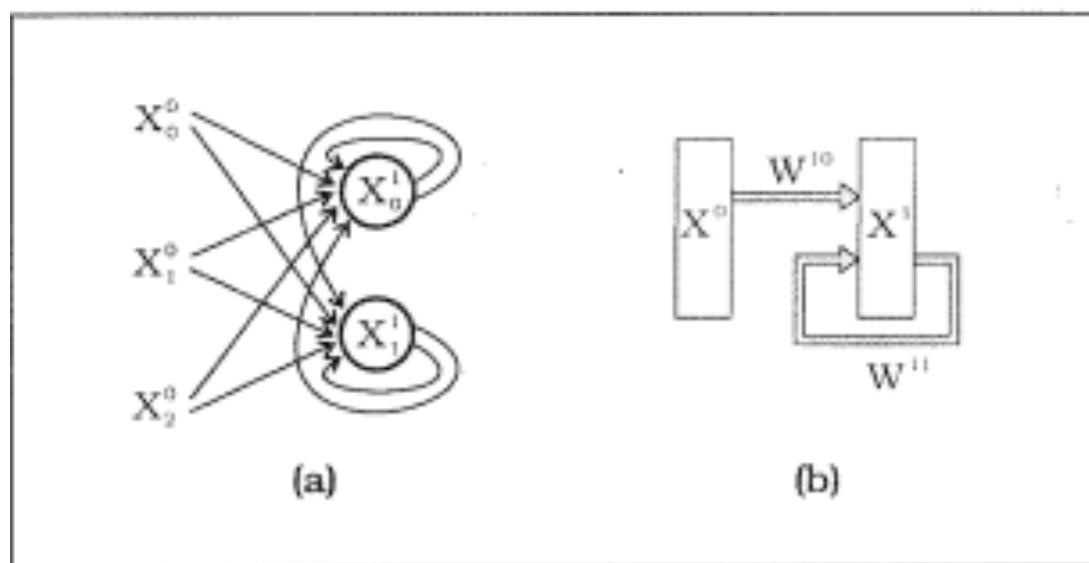


Figura 10 Red neuronal recurrente (Regueiro, 2018)

Las predicciones de series temporales, es una de las aplicaciones más habituales de las redes neuronales recurrentes, debido que a partir de la historia pasada de una o más variables, la red neuronal debe proporcionar una predicción lo más correcta posible de su valor futuro (Zoraida Díaz Burgos Bajo la dirección del Doctor et al., 2005). La mayor parte de los estudios de este tipo son centradas en series económicas o tomadas de fenómenos naturales.

Las redes recurrentes de tipo memoria a corto y largo plazo (LSTM) son capaces de proporcionar una alta precisión en el modelado de secuencia en varios dominios de aplicación

(Ramírez Alcocer et al., 2018). Este tipo de red poseen bucles de retroalimentación compuestos por células de memoria que poseen un sistema de decisiones basado en compuertas, las cuales se encargan de recordar patrones históricos y olvidar patrones no relevantes y es por esta razón que son capaces de asimilar dependencia a corto y largo plazo atendiendo el problema del gradiente de fuga (*View of Ethanol Fuel Demand Forecasting in Brazil Using a LSTM Recurrent Neural Network Approach*, n.d.).

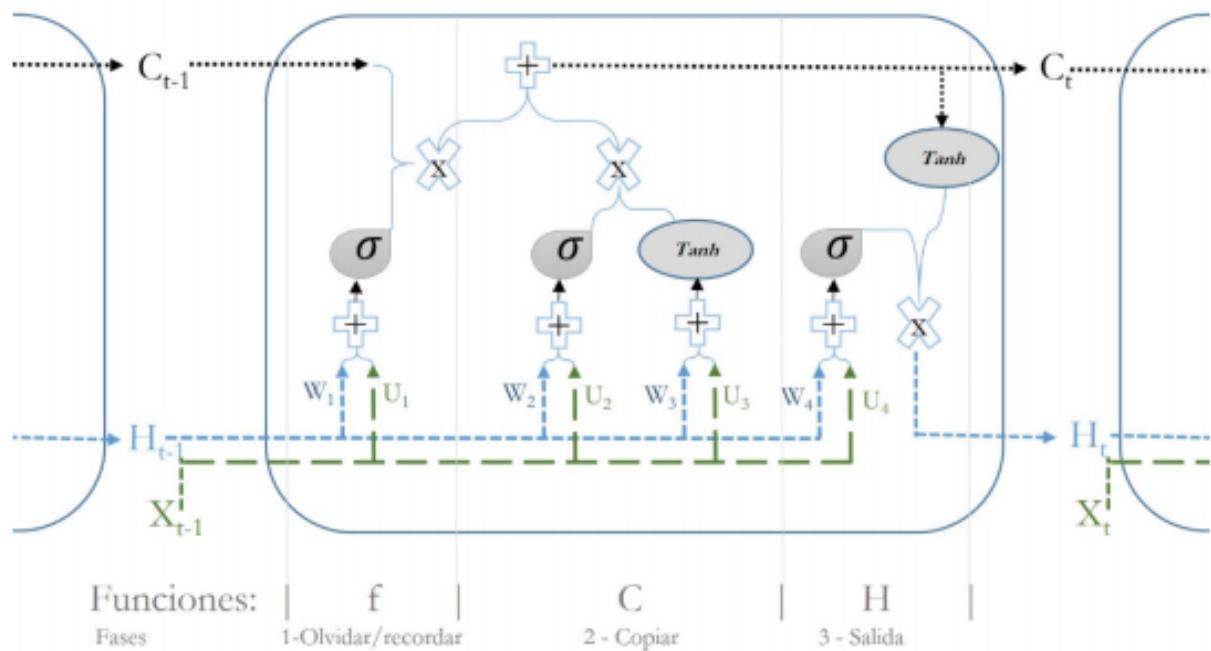


Figura 11 Caracterización de una de las etapas de las redes LSTM (De Lucio, n.d.)

Lo primero que hace una red LSTM es decidir que información va a desechar de la memoria, cuya decisión es tomada por la puerta llamada “*Forgot gate*” o puerta de olvido (Montesdeoca Santana, 2016). Seguidamente se decide que información nueva se va a almacenar en el estado de la memoria, en la puerta de entrada o “*Input gate*”, la que decide cuales valores vamos a actualizar. Por ultimo se decide que sea la salida de la célula LSTM que se obtienen por el producto de dos elementos, el primero de ellos será la salida de la red con sigmoide que servirá para decidir que elementos de la memoria se combinarán y el segundo elemento será filtrado de datos desde la memoria por una tangente hiperbólica (tanh) para empujar los valores entre -1 y 1.

### 3. Objetivos y metodología de trabajo

De acuerdo con la revisión del estado del arte y trabajos anteriores, en el siguiente TFM se ha planteado alcanzar diversos objetivos, utilizando la metodología que se presentara a continuación.

#### 3.1. Objetivo general

Desarrollar un prototipo experimental para la predicción de impactos ambientales en un ecosistema marítimo en tiempo real integrando los datos obtenidos de los modelos oceanográficos, hidrodinámicos y biogeoquímicos a través de algoritmos de inteligencia artificial, comparando diferentes modelos para verificar cual es la mejor métrica.

#### 3.2. Objetivos específicos

- Explorar los datos obtenidos de los diferentes modelos oceanográficos, hidrodinámicos y biogeoquímicos.
- Analizar las diferentes variables que involucran los diferentes modelos oceanográficos de acuerdo a la información obtenida en el sector piloto.
- Establecer las variables más relevantes para poder realizar el prototipo experimental.
- Comparar diferentes algoritmos que permitan la integración de los datos a través de inteligencia artificial.
- Medir la interacción efectiva de entre los modelos existentes con el algoritmo de inteligencia artificial.
- Indagar de que forma pueden interactuar la información con los centros de cultivo en miras a predicción de impacto ambientales
- Determinar el grado de precisión de los algoritmos ante las variables seleccionadas

#### 3.3. Metodología del trabajo

De cara a alcanzar los objetivos específicos y gestionar un trabajo ágil, se utilizará la metodología fundamental para la ciencia de datos de IBM (International Business Machines Corporation) (Rollins, 2015) el que consiste en 10 etapas que forman parte de un proceso iterativo en el cual cada paso es trascendental en el contexto de la metodología general.

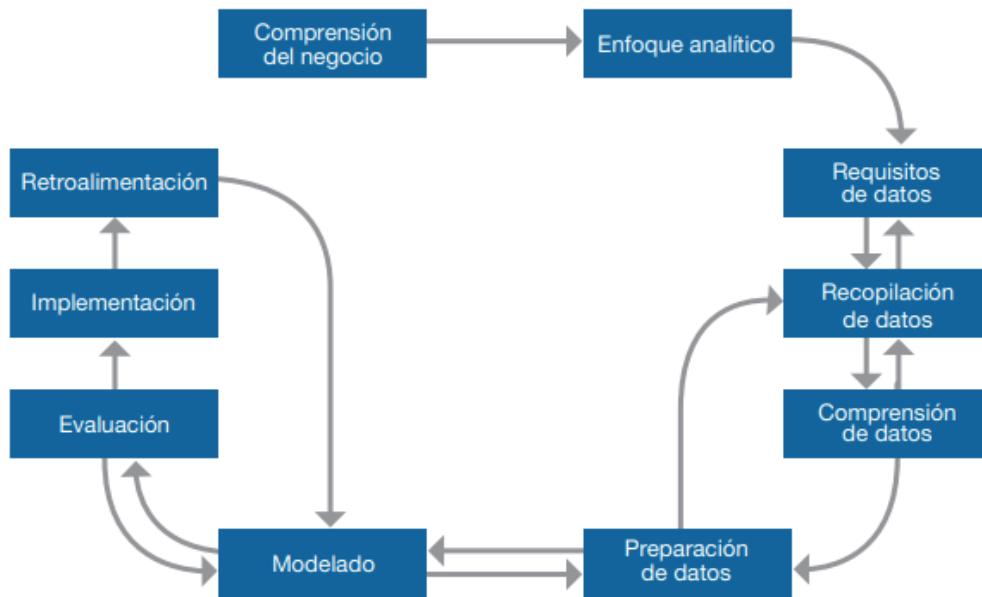


Figura 12 Metodología Fundamental para la Ciencia de Datos (Rollins, 2020.)

A continuación, se explicará cada etapa de la metodología de IBM para ciencia de datos, tomando como referencia su propia secuencia, que ayudará a guiar el desarrollo de esta investigación.

### 3.3.1. Etapa 1: Comprensión del negocio

Es la etapa de análisis, en el cual los pasos principales son; Definir el problema, los objetivos de un proyecto y establecer los requisitos de solución. En este paso se fijan las bases para lograr desarrollar exitosamente los objetivos planteados. Se deben obtener experiencias que permitan analizar los hallazgos, siguiendo un ritmo programado y componer las soluciones que sean necesarias.

### 3.3.2. Etapa 2: Enfoque analítico

Una vez establecido el problema, se debe tener claridad respecto al enfoque analítico que se empleará para la solución del problema, expresando las técnicas estadísticas utilizadas y de aprendizaje automático, escogiendo las óptimas para los resultados que se esperan conseguir.

### **3.3.3. Etapa 3: Requisitos de datos**

Mediante un enfoque analítico se determinan los requisitos de los datos como; formatos, contenido, y representaciones por el conocimiento.

### **3.3.4. Etapa 4: Recopilación de datos**

En la recopilación de los datos, se identifican y reúnen los recursos disponibles que son relevantes para el dominio del problema, visualizando si se debe invertir en la compilación de datos menos accesibles, pero esta decisión se plantea hasta que se pueda saber más de los datos y el modelo. Si se presentan lagunas, es necesario revisar los requisitos y recopilar nuevos datos. Al poder incorporar más datos disponibles e incluso todos, pueden ser capaces de representar mejor los eventos raros del sistema.

### **3.3.5. Etapa 5: Comprensión de datos**

Seguidamente al tener los antecedentes iniciales, los científicos de datos utilizan técnicas estadísticas descriptivas, como su visualización, con la finalidad de comprender la calidad de la información e identificar los datos faltantes.

### **3.3.6. Etapa 6: Preparación de datos**

En esta etapa se debe realizar los procesos de limpieza de los datos, combinación de los datos de múltiples fuentes y transformación de los datos en variables mas útiles. Los científicos de datos deben generar variables explicativas, en el caso de información no estructurada o semi-estructurada, se analiza en miras de ver si puede enriquecer el conjunto de información que permitan potenciar el modelo. Este proceso suele ser el más extendido, entendiendo que, en varios dominios, los pasos de preparación de los datos suelen generar la mayor cantidad de problemas. El poder automatizar la preparación de los datos acelera el proceso. Los sistemas paralelos y la funcionalidad analítica es una ventaja en la preparación de la información al utilizar conjunto de datos muy grandes.

### **3.3.7. Etapa 7: Modelado**

En esta etapa se aprovecha la primera versión de los datos preparados en que se desarrollará el modelo descriptivo o predictivo dependiendo del enfoque de análisis definido anteriormente.

Para el caso de los modelos predictivos se utiliza un conjunto de capacitación para poder construir el modelo. Estos procesos suelen ser muy iterativo debido a las percepciones intermedias, que conlleva a ajustar la preparación de los datos y las especificaciones. Se suelen probar diferentes algoritmos para una técnica, con sus diferentes variables hasta encontrar un mejor modelo.

### **3.3.8. Etapa 8: Evaluación**

En el proceso de desarrollo como previo a la implementación, el científico de datos evalúa el modelo para comprender su calidad, además de garantizar que se aborda el problema de la manera adecuada y completa. Para poder evaluar, es necesario medir el cálculo de diferentes diagnósticos, resultados, gráficas, tablas, esto permitirá determinar la eficacia de los modelos en la resolución de problemas. En oportunidades se utilizan un conjunto de validación para realizar evaluación final. Se pueden asignar modelos de prueba de significancia estadística como prueba adicional, cuando hay en juego algo muy costoso o un sistema crítico.

### **3.3.9. Etapa 9: Implementación**

Al saber que el modelo desarrollado ha sido satisfactorio y aprobado, se implementa en un entorno de pruebas comparables. El cual se hace en una forma limitada hasta que su rendimiento se haya evaluado por completo. Dicha implementación puede generar un informe con recomendaciones, o incrustar el modelo en un proceso complejo administrado por una aplicación personalizada.

### **3.3.10. Etapa 10: Retroalimentación**

Cuando se consiguen los resultados de la implementación del modelo, la organización obtiene la retroalimentación sobre el rendimiento y el impacto en su entorno. Los científicos de datos pueden analizar la retroalimentación en miras de ajustar y gestionar una reimplementación del modelo y de este modo obtener mejores resultados.

## 4. Descripción detallada del experimento

En este capítulo se dará información relevante y detallada de la problemática existente hasta llegar a la solución planteada, explicando cada paso con detención, mencionando tecnologías usadas, organización, instrumentos, entre otras.

### 4.1 Definición de la problemática y descripción del prototipo como herramienta de solución.

En esta investigación se ha tomado en consideración datos existentes en modelos biogeoquímicos e Hidrodinámicos en los procesos oceanográficos, evidenciando que existe una falta de análisis de los datos e información oceanográfica, asociada a los métodos operativos de los centros de cultivo, tanto en las variables de proceso como en las variables físicas del entorno, ya que estos no son entregados en tiempo real, lo que produce un desfase de información, generando que no se logren anticipar en el momento preciso los impactos en el ecosistema marítimo, dando como resultado consecuencias negativas en el medio ambiente y a su vez ineficiencia de los recursos a nivel productivo.

Por tal motivo, se propone dar solución a esta problemática diseñando un prototipo experimental, tomando las líneas base de datos oceanográficos como también los modelados hidrodinámicos y biogeoquímicos que garanticen la conservación del medioambiente, los que interactuaran con algoritmos de IA utilizando redes de neuronas profundas, intentando que se aprendan los modelos ya existentes para generar sistemas predictivos en tiempo real. Aportando de esta manera en la prevención de los impactos en el medio ambiente, anticipando las externalidades negativas que pueden surgir debido a acciones provenientes desde el centro de cultivo como de las variables físico-ambientales.

### 4.2 Técnicas estadísticas y aprendizaje automático.

A nivel estadístico se utilizará análisis exploratorio de los datos, para poder verificar la calidad de la información, es decir, cual es el dato mayor o menor, determinar la media, mediana y su desviación estándar.

Además de esto se seleccionará los algoritmos de aprendizaje automático que mejor se adapten a nuestra problemática, entendiendo que la situación a la que nos enfrentamos

consiste en la predicción de series temporales siguientes, es decir, procesar información ya obtenida y almacenada para luego emplearla a modo de predecir escenarios futuros.

Por lo anteriormente señalado nos apoyaremos en algoritmos de regresión Lineal, redes neuronales Denso, CNN (redes de neuronas convolucionales), RNN (redes de neuronas recurrentes) y algoritmos de aprendizaje automático que puedan generar un aporte comparativo en el desarrollo del experimento.

Dentro de las tecnologías a utilizar es necesario comprender el manejo de datos con los algoritmos que se realizaran las comparativas dentro del experimento del prototipo. En primer lugar, se debe tener en cuenta que la fuente de información que se posee es proveniente del modelo CROCO-OCEAN, cuyo formato es NetCDF (Network Common Data Form) que contiene diversas variables, en formatos geográficos con las componentes principales son la latitud, longitud, profundidad y tiempo por cada componente que se desee revisar e integrar en los algoritmos previamente mencionados.

Se escogen porque son las técnicas más apropiadas para la predicción de series temporales, entendiendo que el océano es un ambiente complejo. Las RNN serán usadas porque son capaces de poder almacenar datos o información en una secuencia lineal, dado que, si se quiere analizar datos provenientes de alguna serie o patrón de secuencia, permite generar la predicción a la variable subsiguiente que se quiere predecir en la línea temporal.

### 4.3 Recursos informáticos

Para la creación del prototipo se aprovecharán recursos informáticos, como *google colaboratory*, ya que nos permite trabajar en un entorno en la web similar a *Jupyter notebook*, que es un entorno de programación que permite documentar el código de una manera más ordenada, además de proveer los recursos de aceleración por hardware que entrega google en el uso de las GPU (graphics processing unit).

De igual forma se utilizará un computador portátil Macbook pro-2012, el cual cuenta con un procesador de 2,5Ghz Dual-Core Intel Core i5, dos discos duros de estados sólidos sumando 640 GB de almacenamiento, además de 16GB de memoria RAM, el cual se utiliza para el manejo de los datos brutos que fueron suministrados por la empresa GEOGAMA (Gestión Acuícola, Marítima y Ambiental Limitada), los que tienen un peso superior a los 300 GB por campaña (invierno y verano) con ese peso imposibilita el uso del recurso de *google colaboratory* gratuito, por lo que es necesario usar el recurso local, usando el IDE (*Integrated*

*(Development Environment) Jupyter Notebook* para el tratamiento de los datos y cuyos conjuntos resultantes si serán procesados en el recurso de la nube previamente nombrado.

Se usa lenguaje de programación Python, debido a que cuenta con una gran comunidad que se dedica a la ciencia de datos y a la inteligencia artificial, esto permite aprovechar de mejor manera los recursos ya existentes y librerías que nos ayuden en el tratamiento y manejo de los datos.

Las librerías que se destacan en la ayuda del manejo de los datos, son; numpy, pandas, netCDF4 (Nadh, 2020) que es una librería que permite manejar la información que se nos proporciona de la data obtenida de los modelos oceanográficos CROCO-OCEANS con Python, además de usar la librería de TensorFlow desarrollada por google, capaz de entrenar redes neuronales, además que ya viene instalado en el entorno de *google colaboratory* facilitando el entorno de trabajo.

Se puede mencionar que, la información proporcionada para este piloto experimental se obtuvo de un análisis sistemático de las variables, que vienen del modelo oceanográfico que nos indica información del sector que estaremos evaluando, en el cual se utiliza Matlab, como un primer acercamiento de los datos que fueron suministrados, para poder conocer inicialmente los conjuntos de datos que fueron proporcionados para este trabajo.

## 4.4 Comprensión del negocio

El rubro acuícola en Chile genera uno de los mayores ingresos del país, pero cuenta con un gran desafío de sostenibilidad en el tiempo debido al impacto ambiental, por lo que es necesario aprovechar los recursos de la tecnología, en especial la inteligencia artificial, para que permita generar la predicción del espacio oceanográfico donde se desarrollan los procedimientos del centro de cultivo y todo lo inherente al proceso de acuicultura. También nos puede anticipar ante posibles hechos que generen cambios en el entorno en el cual se desarrolla la actividad, pudiendo ayudar a que los sectores involucrados en la producción se encuentren dentro de los parámetros ambientales que permitan la continuidad del negocio.

Para poder llegar a esto, es necesario generar un piloto experimental que ayude en la predicción de los datos o variables del centro de cultivo, otorgando series temporales que ayuden a generar una mejor comprensión de la operatividad del sector y de este modo crear una solución que pueda servir como alternativa para prevenir sucesos negativos según el comportamiento de las variables medidas en el tiempo.

Permitiendo que se pueda operar dentro del centro de cultivo de salmonidos durante un tiempo prolongado, sin tener multas y cierre de operaciones debido al no cumplimiento de los parámetros óptimos para la producción, también podría adelantarse con este sistema predictivo algunas externalidades negativas del entorno que puedan afectar parámetros críticos, como el uso de los recursos de una manera mucho mas eficiente haciendo uso de las bondades de la ciencia de datos, inteligencia artificial y la industria 4.0 para este rubro productivo.

## 4.5 Enfoque analítico

Para desarrollar esta etapa es necesaria la comprensión del negocio y entender que se pueden obtener mejoras utilizando los modelos oceanográficos ya existentes apoyados con técnicas de inteligencia artificial para lograr predecir variables ambientales, que afectan en este caso directamente a la acuicultura chilena.

Para este prototipo experimental se utilizará la información de un centro de cultivo acuícola de salmonidos, ubicado al sur de Chile, la información es proporcionada por la empresa GEOGAMA en las cuales se suministran datos de muestreo de las campañas de invierno verano del sector, en el cual se seleccionará una grilla particular que será el centro de cultivo para el estudio.

Para llevar a cabo este prototipo experimental, es necesario comprender que el desafío enfrentado, es lograr la predicción de series temporales de variables ambientales oceanográficas, todo apoyado con información obtenida del modelo CROCO-OCEAN y de esta manera, poder corroborar estos datos con la información existente de alguno de los componentes o variables que se utilizaron en la misma época, es decir, instrumentación de campo o sensores que estuvieron instalados en el sector de estudio, datos con los cuales se podrán entrenar los diferentes modelos de inteligencia artificial.

Se debe tomar en cuenta, que se seleccionara una variable como prueba, ya que esto se puede trasladar a las diferentes variables que están involucradas en el espacio ambiental oceanográfico, y probar conceptualmente cual es la mejor vía para poder solucionar esta problemática. Esta sería una fuente valiosa para poder generar una predicción de múltiples variables que atentan contra la operatividad del centro de cultivo generando modelos más robustos y así fortalecer el desarrollo de una solución que puede traer beneficios para el productor y para el medio ambiente.

## 4.6 Requisitos de datos

La fuente de datos suministrada que se utilizarán para esta evaluación se encuentra en formato NetCDF, estos datos se tendrán que procesar debido a que ellos son proporcionados en información de múltiples dimensiones, principalmente están divididos en los siguientes componentes; tiempo, grilla en eje x, grilla eje y, profundidad. Para el conjunto de datos va a ser necesario realizar transformaciones, para que puedan quedar como series temporales guardadas en formato de valores separados por coma o por sus siglas en inglés CSV.

Para guardar las series temporales en el formato CSV se debe tener conocimiento de la unidad de tiempo a la cual hacen referencias las variables que se estarán trabajando, por si es necesario realizar algún tipo de transformación que permita una mejor comprensión de los datos.

Otro punto importante con estos datos es que, se debe considerar que tienen una referenciación de un espacio geográfico, al igual que los componentes de latitud y longitud. Para esto, es necesario realizar una transformación y analizar solo la grilla de estudio donde se simula el centro de cultivo utilizado para este prototipo experimental, ya que sino el espacio de trabajo sería muy amplio y no estaríamos visualizando que es lo que pasa directamente en el centro de cultivo.

Como referencia se muestra la Figura 13, ésta es la batimetría del sector en el cual se debe tomar la grilla de trabajo.

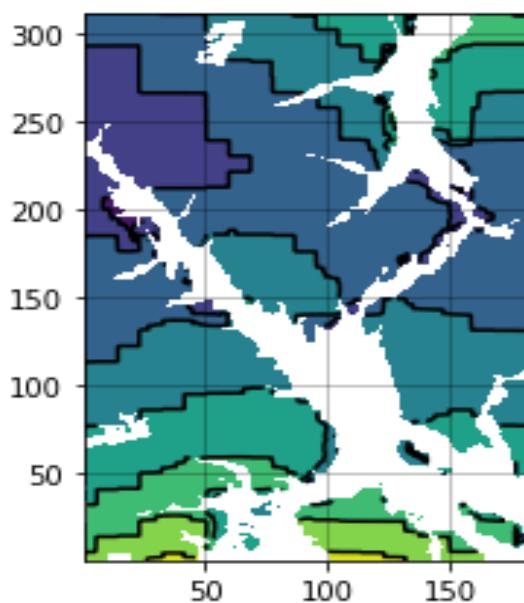


Figura 13 Batimetría del sector de estudio al sur de Chile Fuente: Elaboración propia

## 4.7 Recopilación de los datos

Los datos suministrados por la empresa GEOGAMA, específicamente su departamento de oceanografía, fueron subidos en la nube bajo la plataforma de google drive. Estos datos son obtenidos de las campañas de verano e invierno al sur de Chile, archivos que se encuentran en formato NetCDF, con un peso de por lo menos 300GB por campaña, estos datos son provenientes de la salida del modelo CROCO-OCEAN.

Se recopila también la información de la batimetría del sector y de la grilla con la cual se trabajará. Un dato importante son las coordenadas de trabajo, pues ellas nos indicaran la grilla que usaremos para el desarrollo del prototipo.

Tabla 2 Coordenadas de trabajo

Coordinada X	Coordinada Y
Representación del valor grilla $xj$	Representación del valor grilla $yz$
Representación del valor grilla CROCO-OCEAN $x_rho$	Representación del valor grilla CROCO-OCEAN $etha_rho$
70	36

Fuente: Elaboración propia

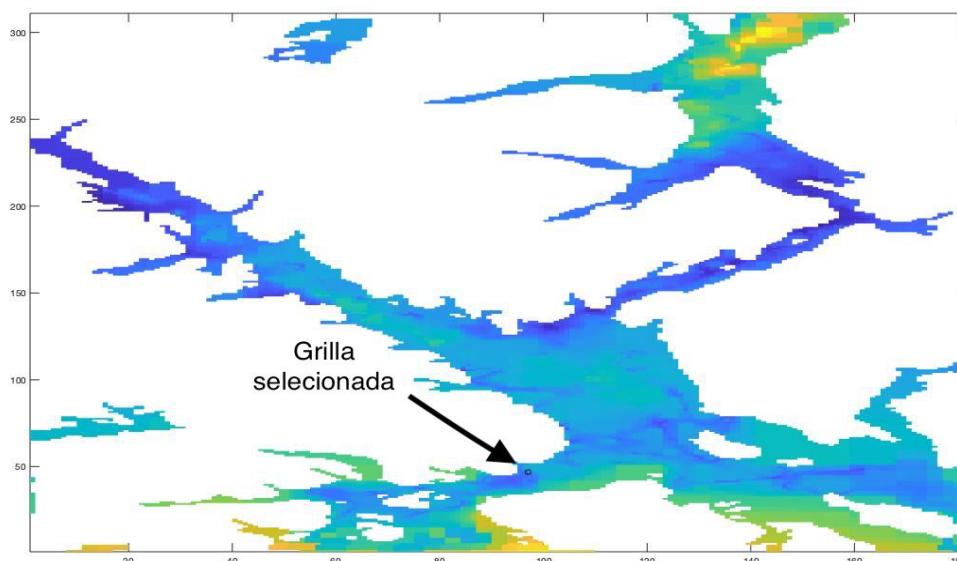


Figura 14 Grilla seleccionada para el estudio Fuente: Elaboración propia

Al tener esos datos en google drive y ser archivos muy pesados que se iban a trabajar de manera local, fue necesario utilizar la herramienta para copias de seguridad *google takeout*, la cual permite exportar una copia del contenido de la cuenta google creando una copia de seguridad, permitiendo descargar los archivos en secciones comprimidas y así bajarlos en el equipo que se utilizó para trabajar. Se debió utilizar 7Zip para poder descomprimir y reunificar en un solo archivo por campaña, y así evitar perdida de información.

De igual forma, fue necesario realizar modificaciones del hardware al portátil donde se iba a trabajar, debido a la capacidad de almacenamiento base que se tenía, el cual era un disco de estado sólido de tan solo 240GB de almacenamiento, en el cual estaban los programas básicos y el sistema operativo, por lo que se tuvo que agregar una unidad de 500GB SSD, sustituyendo el puerto de unidad de disco para ser ocupado como almacenamiento, sumando 740GB con el cual se podría hacer la exploración y revisión de cada set de datos de manera paulatina. Además, se debió incorporar 8GB de memoria RAM extra, para sumar 16GB en total, debido a que al explorar los datos en la portátil surgían problemas por el volumen de información manejada.

## 4.8 Comprepción de datos primera iteración

Luego de poder recopilar los datos que fueron suministrados, solucionando los problemas de hardware que se presentaron al inicio, además de los formatos básicos para poder manejar los datos dentro de un entorno de programación conocido, se utilizó el lenguaje de programación para la ciencia de datos Python, ya que cuenta con una gran comunidad y una amplia librería para poder continuar con el desarrollo del prototipo.

Se procedió a cargar los datos de verano inicialmente y así poder saber con qué características e información se estaba trabajando, fue necesario instalar la librería netCDF4 para Python. Seguidamente ya se podría leer este tipo de archivos, empezando el proceso de exploración y de este modo poder comprender los datos con los que se trabajarían.

Posteriormente se procedió a verificar cuáles son las variables que posee el archivo de trabajo, dando como resultado un diccionario de Python en el cual se muestran las etiquetas que se observan a continuación:

Tabla 3 Variables provenientes del modelo CROCO-OCEAN

Nombre	Nombre largo	Unidades
spherical	grid type logical switch	-
xl	domain length in the XI-direction	meter
el	domain length in the ETA-direction	-
xi_rho	x-dimension of the grid	-
xi_u	x_grid_index at u location	-
eta_rho	y-dimension of the grid	-
eta_v	y-dimension of the grid at v location	-
Vtransform	vertical terrain-following transformation equatio	-
s_rho	S-coordinate at RHO-points	-
s_w	S-coordinate at W-points	-
sc_r	ocean s roms coordinate at rho point	-
sc_w	ocean s roms coordinate at w point	-
Cs_r	S-coordinate stretching curves at RHO-points	-
Cs_w	S-coordinate stretching curves at W-points	-
hc	S-coordinate parameter, critical depth	meter
h	bathymetry at RHO-points	meter
f	Coriolis parameter at RHO-points	second-1
pm	curvilinear coordinates metric in X	meter-1
pn	curvilinear coordinates metric in ET	meter-1
lon_rho	longitude of RHO-points	degree_east
lat_rho	latitude of RHO-points	degree_north
lon_u	longitude of U-points	degree_east
lat_u	latitude of U-points	degree_north
lon_v	longitude of V-points	degree_east
lat_v	latitude of V-points	degree_north
angle	angle between XI-axis and EAST	radians
mask_rho	mask on RHO-points	-
time_step	time step and record numbers from initialization	-
scrum_time	time since initialization	second
time	time since initialization	second
zeta	free-surface	meter
ubar	vertically integrated u-momentum component	meter second-1
vbar	vertically integrated v-momentum component	meter second-1
u	u-momentum component	meter second-1
v	v-momentum component	meter second-1
temp	potential temperature	Celsius
salt	salinity	PSU
DIC	Dissolved inorganic carbon	umol C L-1
TALK	Total Alkalinity	umol C L-1
O2	Oxygen	umol L-1
CACO3	Calcite	umol C L-1
PO4	Phosphate	umol P L-1
POC	Small particulate organic carbon	umol C L-1
Si	Silicate	umol Si L-1
NANO	Nanophytoplankton	umol C L-1
ZOO	Microzooplankton	umol C L-1
DOC	Dissolved organic carbon	umol C L-1
DIA	Diatoms	umol C L-1

MESO	Mesozooplankton	umol C L-1
BSI	Diatoms silicon	umol Si L-1
FER	Dissolved iron	umol Fe L-1
BFE	Iron in big particles	umol Fe L-1
GOC	Big particulate organic carbon	umol C L-1
SFE	Iron in nanophytoplankton	umol Fe L-1
DFE	Iron in diatoms	umol Fe L-1
DSI	Biogenic silica	umol Si L-1
NFE	Iron in small particles	umol Fe L-1
NCHL	Chlorophyll in nano	mg Chl m-3
DCHL	Chlorophyll in diatoms	mg Chl m-3
NO3	Nitrate	umol N L-1
NH4	Ammonium	umol N L-1
w	vertical momentum component	meter second-1
bostr	Kinematic bottom stress	N/m2
wstr	Kinematic wind stress	N/m2
sustr	Kinematic u wind stress component	N/m2
svstr	Kinematic v wind stress component	N/m2
AKt	temperature vertical diffusion coefficient	meter2 second-1
hbl	depth of planetary boundary layer	meter
hbbl	depth of bottom boundary layer	meter
shflux	surface net heat flux	Watts meter-2
swflux	surface freshwater flux (E-P)	centimeter day-1
swrad	Short-wave surface radiation	Watts meter-2

Fuente: Elaboración propia

Estas etiquetas representan variables dentro del conjunto de datos, las cuales caracterizan todo el sector. Estas son importantes dado que con ellas se puede acceder a información dentro del conjunto de datos y hacer una exploración en profundidad, según sea el interés para la investigación.

En el conjunto de datos al trabajar con información de diferentes coordenadas oceanográficas, es vital poder tener una comprensión acerca de las dimensiones que posee este conjunto, debido a que, con la exploración de este se podrá determinar como están compuestas internamente las variables y como se deben tratar.

Se observa a continuación, una diversidad de dimensiones del conjunto de datos, lo que llevará a combinaciones para cada variable, de acuerdo con la composición que tiene cada una de ellas.

*Tabla 4 Dimensiones de las variables*

Nombre de las dimensiones	Tamaño de las dimensiones
xi_rho	182
xi_u	181
eta_rho	311
eta_v	310
s_rho	30
s_w	31
time	1023
auxil	4

Fuente: Elaboración propia

Entre los datos que se deben tener en cuenta sobre las dimensiones que se utilizaran para el presente estudio son; los datos de la etiqueta ‘time’, el cual posee 1023 datos por cada punto, la etiqueta ‘s\_rho’, que representa 30 diferentes profundidades para estudiar, la etiqueta ‘xi\_rho’, que representa el punto xj, que es una de las coordenadas de ubicación de la grilla, en la cual se encuentra el centro de cultivo que servirá de prueba en este prototipo experimental y la etiqueta ‘etha\_rho’, la cual muestra la ubicación en la grilla yj, que representa la otra coordenada del centro de cultivo para realizar el estudio

Procediendo a una exploración aún mayor se puede verificar como se encuentran cada una de las variables, tomando en cuenta, cuales son sus dimensiones internas.

*Tabla 5 Variables dimensionales*

Variables	Nombre completo	Eje	Datos por dimensión
xi_rho	x-dimension of the grid	X	182
etha_rho	y-dimension of the grid	Y	311
s_rho	S-coordinate at RHO-points	Z	30
time	time since initialization	-	1023

Fuente: Elaboración propia

Luego de haber explorado las dimensiones de las variables que forman parte de la estructura principal, se procede a verificar la estructura de una variable, que se componga de esta configuración, ya que esta va a permitir visualizar los datos que están involucrados en alguna condición ambiental que se explorará para ser usado en el prototipo.

*Tabla 6 Variable temperatura configuración dimensional*

Variable	Nombre completo	Unidad	Datos por dimensión
temp	potential temperature	Celsius	temp(time, s_rho, eta_rho, xi_rho)
			(1023, 30, 311, 182)

Fuente: Elaboración propia

Con la exploración de la variable de temperatura, se puede corroborar que posee la estructura dimensional, la que se requiere para armar el conjunto de datos con el cual se generará la estructura de trabajo que se estableció en la etapa metodológica.

Así como se realizó la indagación anterior con la temperatura, también se procedió a realizar la exploración con cada una de las variables presentes en el archivo de datos, para establecer cuales poseían estas dimensiones y así obtener el conjunto de datos con el cual se iba a trabajar en la futura construcción de los modelos.

## 4.9 Preparación de datos primera iteración

Al haber realizado la primera iteración en la comprensión de los datos, fue necesario pasar por una preparación de la información, utilizando un conjunto mucho más cerrado de acuerdo con las características que se dieron en la comprensión inicial.

Se pudieron obtener 30 variables con las mismas características dimensionales del conjunto inicial de datos, para esto fue necesario guardar cada variable con su etiqueta, seguida de la unidad ingenieril de medida y así poder facilitar la próxima comprensión de los datos, quedando con las siguientes etiquetas:

*Tabla 7 Variables con etiquetas modificadas incluyendo unidades*

Nombre Original	Nombre Modificado
time	time second
u	u meter second-1
v	v meter second-1
temp	temp Celsius
salt	salt PSU
DIC	DIC umol C L-1
TALK	TALK umol C L-1
O2	O2 umol L-1
CACO3	CACO3 umol C L-1
PO4	PO4 umol P L-1
POC	POC umol C L-1
Si	Si umol Si L-1
NANO	NANO umol C L-1
ZOO	ZOO umol C L-1

DOC	DOC umol C L-1
DIA	DIA umol C L-1
MESO	MESO umol C L-1
BSI	BSI umol Si L-1
FER	FER umol Fe L-1
BFE	BFE umol Fe L-1
GOC	GOC umol C L-1
SFE	SFE umol Fe L-1
DFE	DFE umol Fe L-1
DSI	DSI umol Si L-1
NFE	NFE umol Fe L-1
NCHL	NCHL mg Chl m-3
DCHL	DCHL mg Chl m-3
NO3	NO3 umol N L-1
NH4	NH4 umol N L-1
w	w meter second-1

Fuente: Elaboración propia

Luego de haber etiquetado era necesario crear el conjunto de datos, en lo que se procedió a ir tomando las coordenadas  $x_j$  junto con la coordenada  $y_j$ , con el valor fijo establecido para el centro de cultivo o grilla de estudio e ir generando a través de una función los archivos con extensión CSV por cada capa de profundidad, es decir, las 30 capas que se tenían establecidos como  $s\_rho$ .

A partir de ello se tienen 30 archivos de datos, los cuales son leídos, para generar un conjunto de información consolidado con la unión de las capas de las mismas coordenadas, obteniendo un *dataframe* de 30.690 filas y 30 columnas, con la cual se procedió a generar un nuevo archivo CSV con el consolidado de datos.

De esta forma el conjunto de datos ya se pudo cargar en el notebook de *google colaboratory*, para esto se determinó un conjunto de referencias del *dataframe* principal y un set de datos donde se tiene la variable del tiempo.

## 4.10 Comprensión de datos segunda iteración

Luego de tener el grupo de datos ya tratados del formato NetCDF a CSV y con la separación de dos conjuntos, una serie de tiempo y el *dataframe* restante, se procede a verificar las primeras filas y las últimas para corroborar que, entre el proceso de gestionar el conjunto de datos no se consiga algún error que se estuviera arrastrando en la unión o en el tratamiento previo de los datos.

Fue necesario explorar como queda el conjunto de datos, mencionando cuales son las dimensiones que se tienen.

En la exploración se pudo observar, que se poseen 29 columnas debido a que fue tomada la columna de tiempo aparte, y también se conservan las 30.690 filas generadas con el conjunto tratado previamente.

Luego se procede a pintar el comportamiento de algunas variables y verificar que características tienen a lo largo del tiempo.

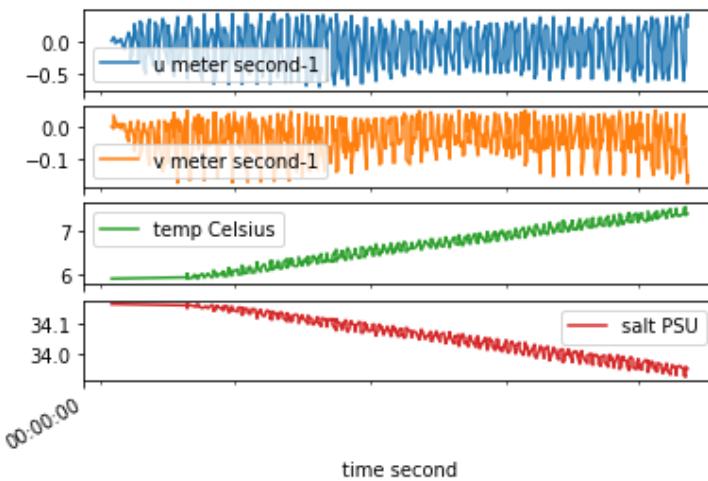


Figura 15 Características a lo largo del tiempo completo Fuente: Elaboración propia

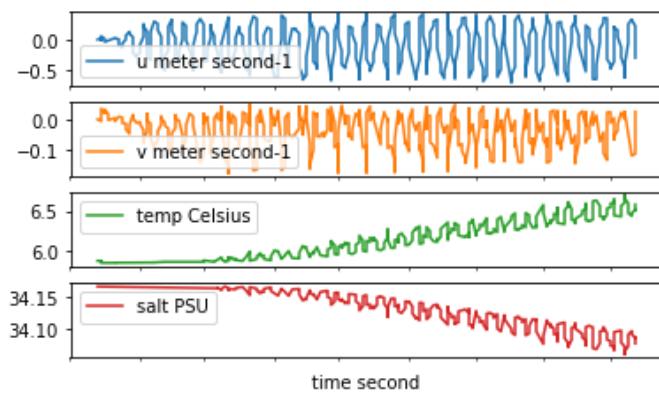


Figura 16 Características a lo largo del tiempo hasta 480 datos Fuente: Elaboración propia

Esta exploración se realizó con el resto de los conjuntos de datos para visualizar como es el comportamiento de cada variable en su serie de tiempo, visualizando el set completo como también los fragmentos durante esta etapa de comprensión de los datos.

Continuando con la comprensión estadística del conjunto de datos, se procedió a utilizar la función *describe* de la librería de pandas, la cual permite mostrar las estadísticas del conjunto de datos.

Tabla 8 Datos estadísticos del conjunto de datos

	<b>count</b>	<b>mean</b>	<b>std</b>	<b>min</b>	<b>25%</b>	<b>50%</b>	<b>75%</b>	<b>max</b>
BFE umol Fe L-1	30690.0	0.000305	0.000373	4,96E-02	1,87E+00	6,52E+01	0.000648	0.001100
BSI umol Si L-1	30690.0	0.029474	0.032706	1,50E+03	2,88E+03	1,37E+04	0.051302	0.128013
CACO3 umol C L-1	30690.0	0.061126	0.061966	2,32E+03	5,99E+03	2,69E+04	0.117461	0.182655
DCHL mg Chl m-3	30690.0	0.051405	0.047779	2,18E+03	7,39E+03	3,65E+04	0.090006	0.176629
DFE umol Fe L-1	30690.0	0.000001	0.000001	5,00E-02	1,83E-01	9,10E-01	0.000002	0.000004
DIA umol C L-1	30690.0	0.206453	0.190553	1,00E+04	2,83E+04	1,49E+05	0.363773	0.696160
DIC umol C L-1	30690.0	2.123.942.395	13.290.242	2,11E+09	2,11E+09	2,12E+09	2.136.934.082	2.144.519.287
DOC umol C L-1	30690.0	11.358.100	0.827818	1,03E+07	1,06E+07	1,13E+07	12.024.643	13.151.708
DSI umol Si L-1	30690.0	0.013139	0.019375	2,78E+02	7,07E+02	2,41E+03	0.018067	0.074651
FER umol Fe L-1	30690.0	0.094627	0.066701	3,58E+02	2,98E+04	9,32E+04	0.152804	0.227091
GOC umol C L-1	30690.0	0.063141	0.063063	2,07E+03	6,05E+03	2,85E+04	0.137173	0.165633
MESO umol C L-1	30690.0	0.012666	0.004333	9,90E+03	9,96E+03	1,03E+04	0.013759	0.029755
NANO umol C L-1	30690.0	0.436246	0.349823	1,00E+04	5,14E+04	4,77E+05	0.733460	1.148.098
NCHL mg Chl m-3	30690.0	0.078619	0.062857	2,18E+03	9,86E+03	8,42E+04	0.132732	0.206810
NFE umol Fe L-1	30690.0	0.000002	0.000002	5,00E-02	2,47E-01	2,13E+00	0.000003	0.000005
NH4 umol N L-1	30690.0	0.052590	0.047860	1,00E+04	1,80E+04	2,95E+04	0.075647	0.181861
NO3 umol N L-1	30690.0	16.776.885	1.391.658	1,41E+07	1,56E+07	1,67E+07	18.094.217	18.812.498
O2 umol L-1	30690.0	297.505.318	1.556.480	2,95E+08	2,96E+08	2,98E+08	298.832.428	299.779.419
PO4 umol P L-1	30690.0	1.327.842	0.059719	1,23E+06	1,28E+06	1,31E+06	1.382.847	1.425.893
POC umol C L-1	30690.0	0.402766	0.425665	1,00E+04	3,25E+04	1,88E+05	0.802609	1.244.324
SFE umol Fe L-1	30690.0	0.003100	0.004133	5,00E-02	3,48E+01	6,31E+02	0.005630	0.012760
Si umol Si L-1	30690.0	9.499.086	3.314.882	5,74E+06	6,24E+06	8,84E+06	12.181.413	17.567.022
TALK umol C L-1	30690.0	2.290.528.549	11.844.105	2,28E+09	2,28E+09	2,29E+09	2.302.051.025	2.310.431.885
ZOO umol C L-1	30690.0	0.112749	0.172676	9,72E+03	9,87E+03	1,29E+04	0.152436	0.754237
salt PSU	30690.0	34.070.363	0.072456	3,39E+07	3,40E+07	3,41E+07	34.139.332	34.167.885

temp Celsius	30690.0	6.591.963	0.509164	5,85E+06	6,11E+06	6,60E+06	7.029.170	7.580.384
u meter second -1	30690.0	-0.065995	0.290208	-6,93E+05	-3,28E+05	-3,86E+03	0.188195	0.411113
v meter second -1	30690.0	-0.036890	0.051150	-1,77E+05	-7,03E+04	-3,32E+04	0.004857	0.050560
w meter second -1	30690.0	0.003391	0.005803	-1,37E+04	-8,29E+02	2,49E+03	0.006707	0.019413

Fuente: elaboración propia.

En la tabla anterior se visualizan diferentes características importantes del conjunto de datos, entre los cuales se puede tener información acerca de los valores mínimos, máximos, promedio y desviación estándar, cuyos datos nos pueden dar una buena aproximación de como esta el conjunto de datos.

Seguidamente se procede a realizar la matriz de correlación que permite generar una mayor comprensión de las características del conjunto de datos.

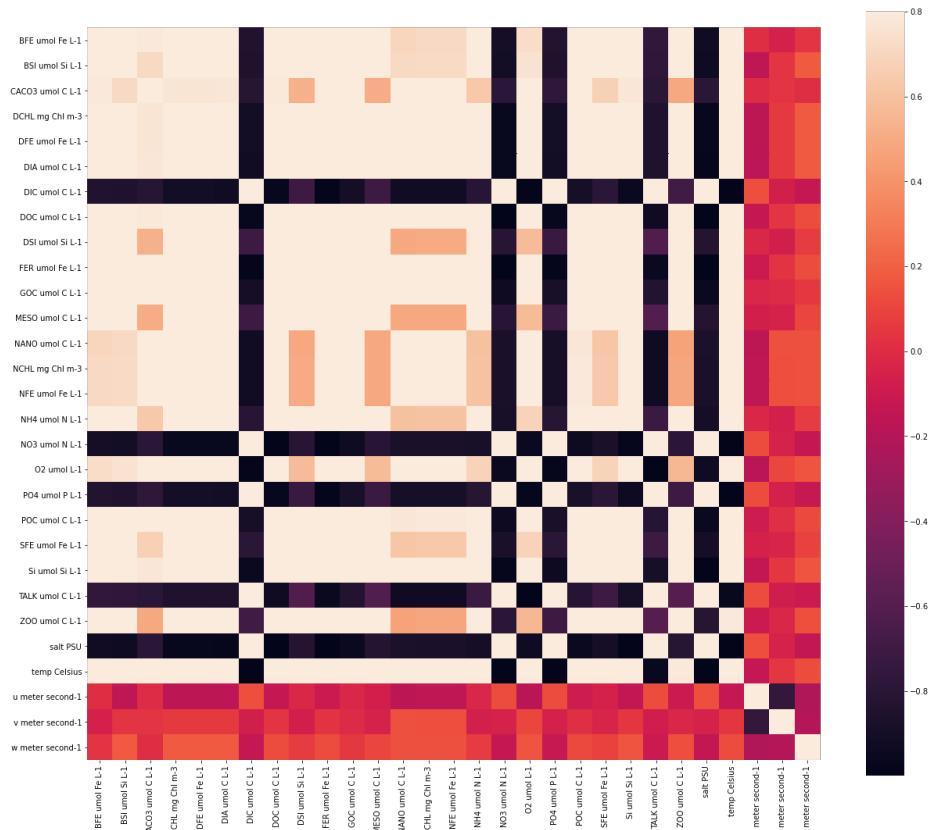


Figura 17 Matriz de correlación del conjunto de datos completo Fuente: Elaboración propia

En esta matriz se puede observar que, a medida que los colores son mucho más claros tienen una mayor correlación entre dichas variables, a diferencia de los colores más oscuros.

Para poder continuar con la exploración, se aprovecha de tomar la variable piloto con la que se realizarán las pruebas de concepto, por tal motivo se considera la variable temperatura para realizar la matriz de correlación, teniendo en cuenta las 10 variables que tienen una mayor correlación con esta y así poder tener un panorama mucho más claro acerca de su comportamiento.

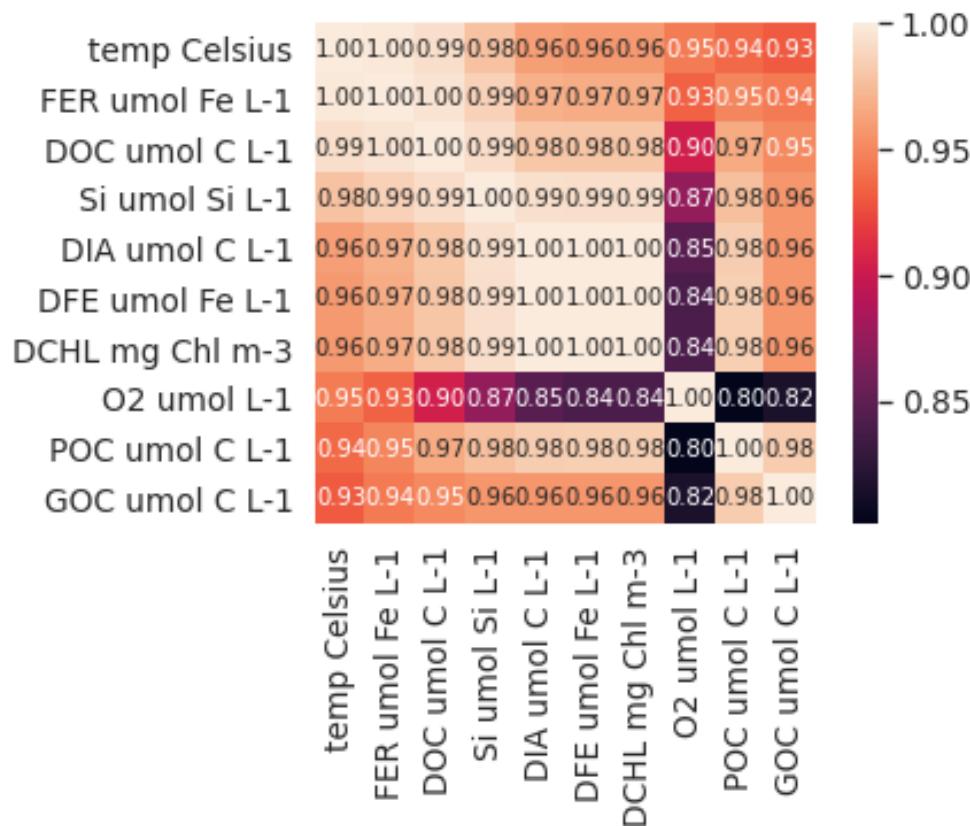


Figura 18 Matriz de correlación respecto a la temperatura Fuente: Elaboración propia

Se procedió a realizar una matriz de covarianza, para poder identificar otras características del conjunto de datos, este proceso ayudara si es necesario a eliminar algunas variables, que puedan sesgar los modelos con los cuales se van a realizar pruebas experimentales.

## 4.11 Preparación de datos segunda iteración

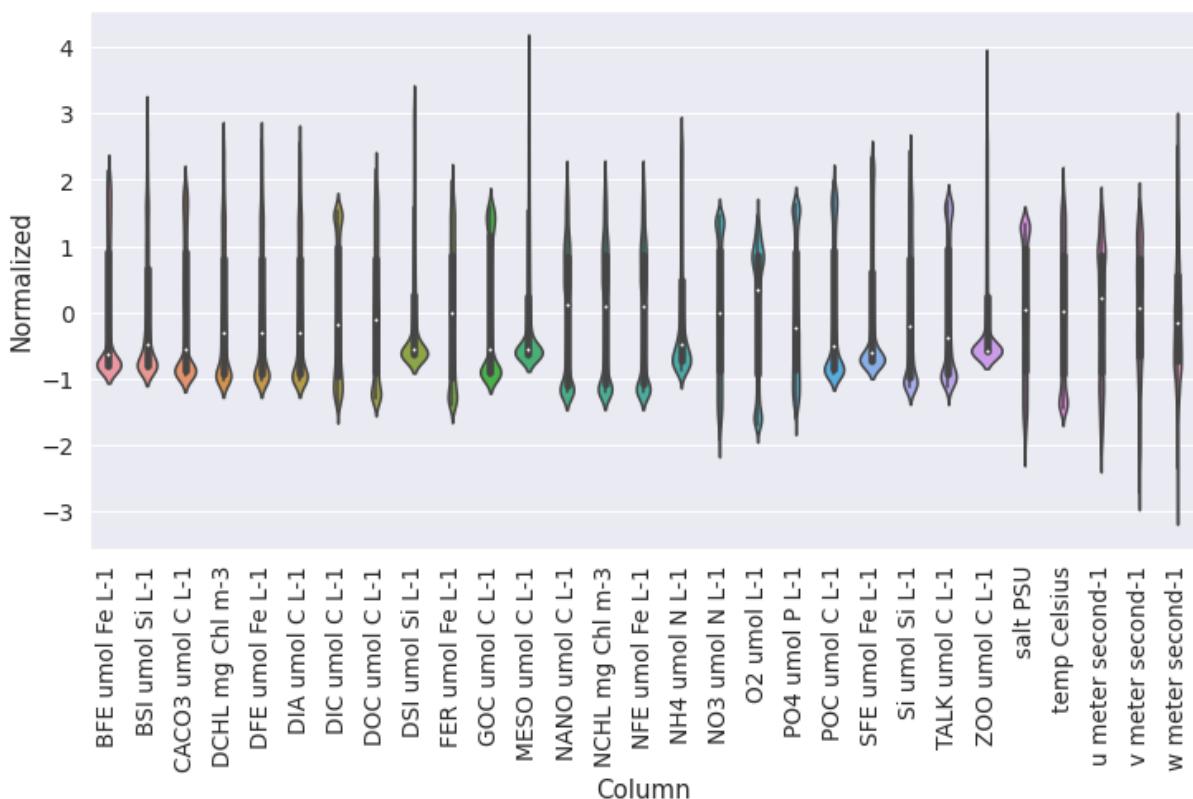
Al haber realizado la segunda iteración en la comprensión de la información, ya se tienen las características del conjunto de datos con el que se va a trabajar, por lo que es necesario generar la división de los datos en 3 conjuntos distribuidos de la siguiente manera: 70% de los datos serán utilizados para el entrenamiento de los modelos, 20% será utilizado para el

proceso de validación de los modelos y un 10% para la prueba o test de los modelos que se analizarán.

En primer lugar, es necesario cerciorarse que sea posible dividir los datos en ventanas de muestras consecutivas. De igual forma hay que garantizar que los resultados de la validación y las pruebas sean más realistas, siendo evaluadas en función de los datos recopilados después de que el modelo sea entrenado.

Seguidamente, es importante generar una normalización de la información, ya que como se pudo observar cuando se realizó la exploración de los datos, cada uno de ellos poseían valores máximos y mínimos distintos, así como el manejo a unas escalas totalmente diferentes, por lo que es necesario generar una modificación para que se pueda trabajar en rangos similares con las diferentes variables

El proceso de normalización se realiza restándole el valor promedio y dividiendo por la desviación estándar, con lo que podemos observar como queda la distribución normalizada por el conjunto de variables, en la Figura 19.



*Figura 19 Chequeo de distribución de las funciones Fuente: Elaboración propia*

**Ventanas de datos:** Los modelos que se evaluarán están basados en una venta de muestras consecutivas de datos, cuyas principales características de estas son las siguientes:

- El ancho (número de pasos en el tiempo) de las ventanas de entrada y etiqueta.
- El tiempo de compensación entre ellos.
- Las características se utilizan como entradas, etiquetas o ambas.

La creación de la clase WindowsGenerator puede ayudar facilitando el uso de las siguientes características:

En primer lugar, manejar los índices y compensaciones, siendo capaz de tomar los marcos de los datos de entrenamiento, evaluación y prueba.

Si se quieren generar 24 horas de salida a partir de 24 horas de entrada, se podría representar una configuración de este modo:

Tabla 9 Configuración de ventana para generar una salida de 24 horas a partir de 24 horas de entrada

<b>Tamaño total de la ventana</b>	48
<b>Indices de entrada</b>	[ 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23]
<b>Indice de Salida</b>	[47]
<b>Etiqueta de columna</b>	['temp Celsius']

Fuente: elaboración propia.

En la tabla mostrada se puede visualizar que la ventana de entrada esta representada de 0 a 23, el índice de salida es 24 horas posterior a esta, mostrándose como el índice 47 para la variable temperatura.

Como segundo ejemplo podemos considerar que, si se desea crear una ventana de predicción teniendo como dato 6 horas de historia y queremos predecir la hora posterior, es decir, la hora 7 se puede mostrar de la siguiente manera:

Tabla 10 Configuración de ventana para generar una salida de una hora a partir de 6 horas de entrada

<b>Tamaño total de la ventana</b>	7
<b>Indices de entrada</b>	[ 0 1 2 3 4 5 ]
<b>Indice de Salida</b>	[6]
<b>Etiqueta de columna</b>	['temp Celsius']

Fuente: elaboración propia.

En segundo lugar, se pueden dividir los datos en listas consecutivas, utilizando el método `split_windows` el cual convertirá esta información en una ventada de entrada y una ventana de etiqueta.

Tabla 11 Dimensiones de las ventanas ante el uso del método `split_windows`

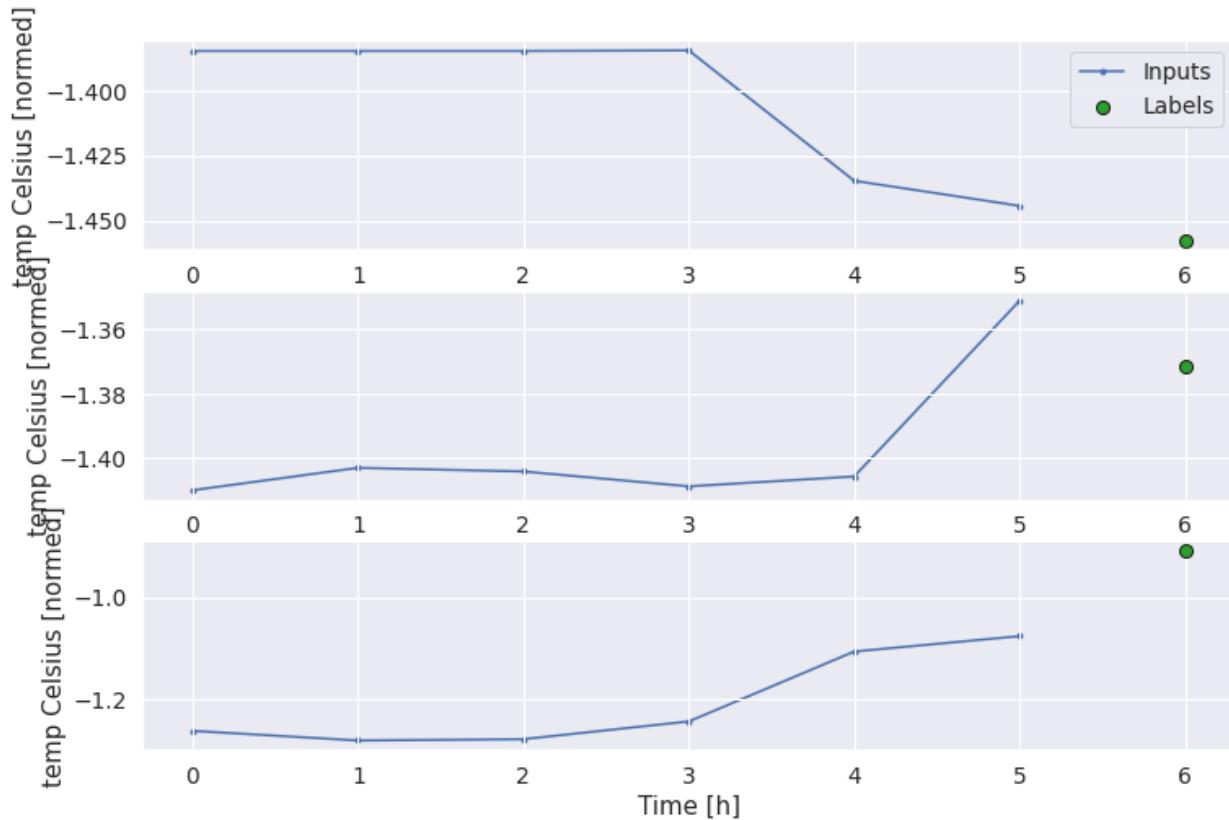
Dimensiones	lote	tiempo	Características
Dimensione de la ventana	3	7	29
Dimensiones de entrada	3	6	29
Dimension de la etiqueta	3	1	1

Fuente: elaboración propia.

Al generar la división se aprovecha la librería de TensorFlow, específicamente el método `stack`, el que es capaz de empaquetar los arreglos teniendo en cuenta que en el índice más externo se encuentran las dimensiones del lote. Los índices intermedios son las dimensiones de tiempo, y los índices internos son las características.

En la expresión anterior se toma un lote de 3 ventanas, de 7 movimientos, con 29 funciones en cada paso de tiempo, el que luego se dividió en un lote de 6 movimientos, 29 entradas y una etiqueta de paso para un tiempo y una función. Cuya etiqueta solo tiene una característica dado que al inicializar `WindowsGenerator` se realizó con la etiqueta = ['temp Celsius'], que representa una sola característica.

Como tercer punto, al trazar el contenido de las ventanas resultantes, utilizando un método que permite una visualización simple de la ventana dividida, el gráfico alinea las entradas, las etiquetas y las predicciones posteriores en función al tiempo al que se refiere el elemento.



*Figura 20 Alineación de entradas, etiquetas y futuras predicciones en función al tiempo para la temperatura* Fuente: Elaboración propia

De igual manera se pueden trazar otras columnas, incluso cuando la configuración de la ventana solo tiene etiquetas destinadas a la columna de temperatura (temp Celsius) como ocurre en la siguiente figura, en la que se muestra la etiqueta de salinidad (salt PSU).

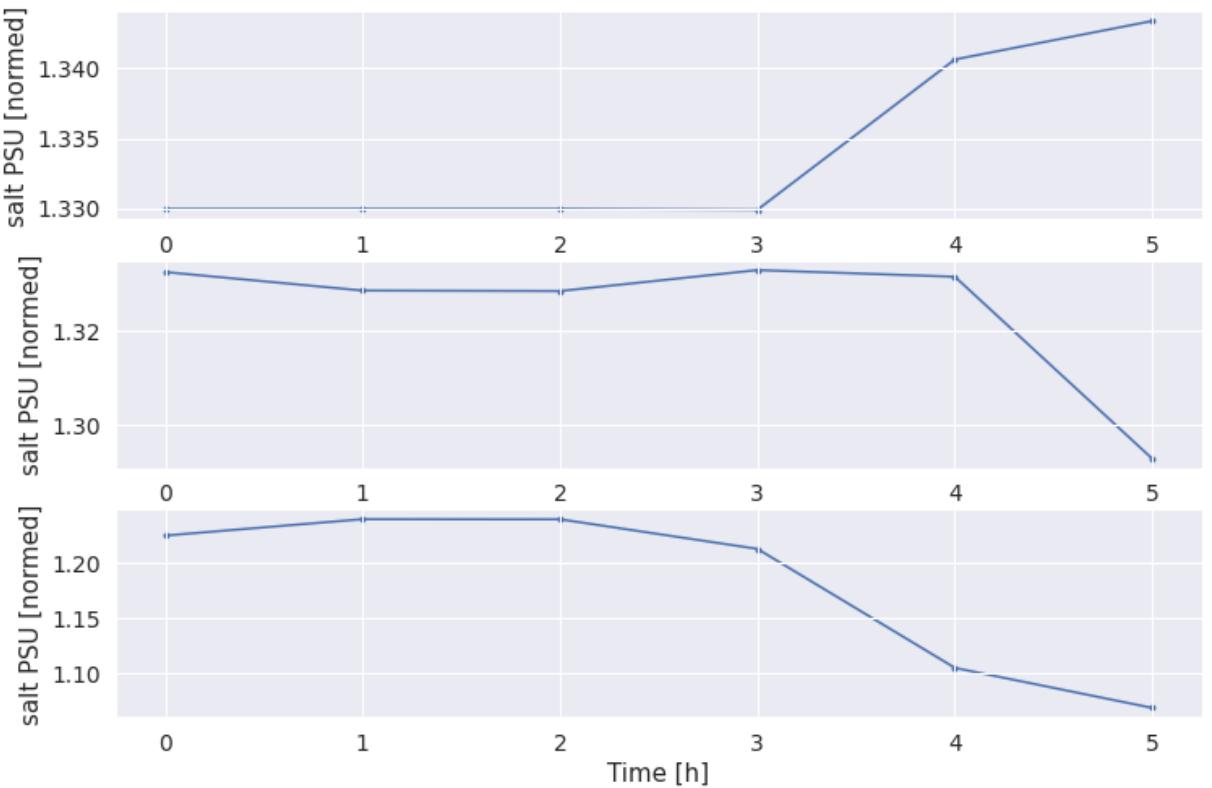


Figura 21 Alineación de entradas, etiquetas y futuras predicciones en función al tiempo para la salinidad Fuente: Elaboración propia

Como cuarto paso se generan lotes de estas ventanas para lograr mayor eficiencia, a partir de los datos de entrenamiento, evaluación y prueba, utilizando `tf.data.Dataset`, con el que finalmente se creará un método llamado `make_dataset`. Luego se toma un `DataFrame` convirtiéndolo en un `tf.data.Dataset` de `(input_window, label_window)` usando la función `preprocessing.timeseries_dataset_from_array`.

El objeto `WindowsGenerator` contiene datos de entrenamiento, validación y test, en el que se le agregan propiedades para acceder a ellos como `tf.data.Dataset`, usando el método `make_dataset` anterior. También se agrega un lote de ejemplo estándar para facilitar el acceso y el trazado.

Posteriormente el objeto `WindowsGenerator`, les da acceso a los objetos `tf.data.Dataset`, para que puedan iterar fácilmente sobre los datos. La propiedad `Dataset.element_spec` le indica la estructura, los `dtype` y las formas de los elementos del conjunto de datos.

La iteración sobre un `Dataset` produce lotes concretos como:

Tabla 12 Ejemplo de estructura del dataset

Dimensiones	lote	tiempo	Características
Dimensiones de entrada	32	6	29
Dimension de la etiqueta	32	1	1

Fuente: elaboración propia.

## 4.12 Modelado

### 4.12.1 Modelos de un solo paso:

El modelo más simple que se puede construir a partir de este tipo de dato es uno que predice el valor de una sola característica, 1 paso de tiempo (1h) en el futuro, basado solo en las condiciones actuales. Así que se comenzará por construir modelos para predecir el valor de la temperatura 1h en el futuro. Seguidamente se configura un objeto WindowsGenerator para producir estos pares de un solo paso (input, label).

Tabla 13 Configuración del WindowsGenerator para producir pares de un solo paso

Tamaño total de la ventana	2
Indices de entrada	[ 0 ]
Indice de Salida	[ 1 ]
Etiqueta de columna	['temp Celsius']

Fuente: elaboración propia.

En el objeto Windows se crea tf.data.Dataset, a partir de los conjuntos de entrenamiento, validación y prueba, lo que le permite iterar fácilmente sobre lotes de datos.

Tabla 14 Estructura del dataset a partir de los conjuntos, entrenamiento, validación y prueba

Dimensiones	lote	tiempo	Características
Dimensiones de entrada	32	1	29
Dimension de la etiqueta	32	1	1

Fuente: elaboración propia.

#### 4.12.1.1 Base:

Antes de construir un modelo entrenable, es bueno tener una línea base de rendimiento como punto de comparación con los modelos posteriores más complicados.

Esta primera tarea consiste, en predecir la temperatura 1h en el futuro dado el valor actual de todas las características (los valores incluyen la temperatura actual). Así que se comienza con un modelo que simplemente devuelva la temperatura actual como pronóstico, prediciendo sin cambios. Esta sería una línea de base razonable, ya que la temperatura cambia lentamente. Por supuesto, esta línea de base no funcionará tan bien si se realiza una predicción adicional en el futuro.

Se crea una instancia y se evalúa este modelo:

Tabla 15 Datos del modelo de línea base (*baseline*) para una sola salida.

<b>baseline</b>	<b>Indice de la etiqueta</b>	<b>Función de perdida</b>	<b>Función de metrica</b>
	temp Celsius	MeanSquaredError	MeanAbsoluteError
		0.0184	0.0691

Fuente: elaboración propia.

Este modelo base imprimió algunas métricas de rendimiento, pero estas no entregan una idea de qué tan bien está funcionando.

WindowGenerator tiene un método de gráfico, pero estos no serán muy interesantes con una sola muestra. Por lo tanto, se crea un WindowsGenerator que generará ventanas de 24 horas de entradas y etiquetas consecutivas a la vez (wide\_windows no cambia la forma en que opera el modelo) este todavía hace predicciones 1h en el futuro basadas en un solo paso de tiempo de entrada. Aquí el eje de time actúa como el eje de batch, cada predicción se realiza de forma independiente sin interacción entre los pasos de tiempo.

Tabla 16 Ventana para la predicción de una hora en el futuro

<b>Tamaño total de la ventana</b>	25
<b>Indices de entrada</b>	[ 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23]
<b>Indices de Salida</b>	[ 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24]
<b>Etiqueta de columna</b>	['temp Celsius']

Fuente: elaboración propia.

Esta ventana expandida se puede pasar directamente al mismo modelo de *baseline* sin ningún cambio de código. Esto es posible porque las entradas y etiquetas tienen la misma cantidad de pasos de tiempo, y la línea de base simplemente reenvía la entrada de información a la salida.

Tabla 17 Estructura de entrada y salida del modelo *baseline*

Dimensiones	lote	tiempo	Características
Dimensiones de entrada	32	24	29
Dimension de la salida	32	24	1

Fuente: elaboración propia.

Al trazar las predicciones del modelo de línea base, se puede ver que las etiquetas son desplazadas a la derecha en 1h.

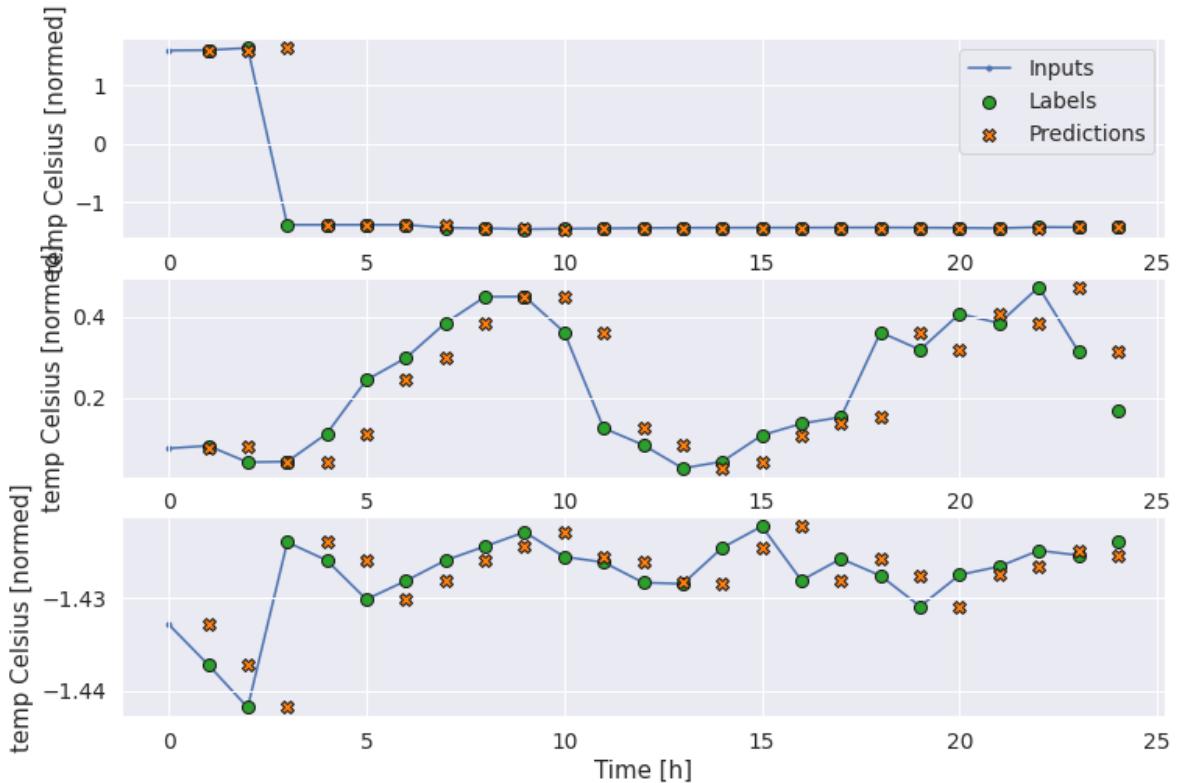


Figura 22 Modelo baseline de un solo paso Fuente: Elaboración propia

En las gráficas mostradas de 3 ejemplos, el modelo de un solo paso se ejecuta en el transcurso de 24 horas. Esto se explica de la siguiente forma:

La línea azul “entradas”, muestra la temperatura de ingreso en cada paso de tiempo. El modelo recibe todas las características, este gráfico solo muestra la temperatura.

Los puntos verdes de “etiquetas”, muestra el valor de predicción que se tiene como objetivo. Estos puntos se muestran en el momento de la predicción, no en el momento de la entrada. Es por eso, que el rango de etiquetas se desplaza 1 paso con respecto a las nuevas entradas.

Las cruces naranjas de “predicciones”, muestran los pronósticos del modelo para cada paso de tiempo de salida. Si el modelo estuviera prediciendo perfectamente, las predicciones aterrizarían directamente en las “etiquetas”.

#### 4.12.1.2 Modelo lineal:

El modelo entrenable más simple que se puede aplicar a esta tarea, es insertar una transformación lineal entre la entrada y la salida. En este caso, la salida de un paso de tiempo solo depende de ese movimiento.

La capa layers.Dense sin un conjunto de activación es un modelo lineal. La capa solo transforma el último eje de los datos de; batch, time, inputs a batch,time,units se aplica de forma independiente a cada elemento en los ejes de batch y time.

Tabla 18 Estructura de entrada y salida del modelo lineal

Dimensiones	lote	tiempo	Características
<b>Dimensiones de entrada</b>	32	1	29
<b>Dimension de la salida</b>	32	1	1

Fuente: elaboración propia.

Se genera una función para compilar y entrenar el modelo.

Tabla 19 Datos que alimentan la función para compilar y entrenar el modelo

<b>MAX_EPOCHS</b>	20	
<b>earling_stopping</b>	monitor	val_loss
	patience	2
	mode	min
<b>model</b>	loss	MeanSquaredError
	optimizers	Adam
	metrics	MeanAbsoluteError

Fuente: elaboración propia.

Se entrena el modelo y se evalúa su desempeño:

Tabla 20 Datos del entrenamiento y evaluación del modelo lineal

Lineal	Función de perdida	Función de metrica
	MeanSquaredError	MeanAbsoluteError
	0,0141	0,0722

Fuente: elaboración propia.

Al igual que el modelo de *baseline*, el modelo lineal se puede utilizar en lotes de ventanas amplias. Usado de esta manera, el modelo hace un conjunto de predicciones independientes en pasos de tiempo consecutivos. El eje de time actúa como otro eje de batch. No hay interacciones entre las predicciones en cada paso de tiempo.

Tabla 21 Estructura de las ventanas para el modelo lineal

Dimensiones	lote	tiempo	Características
Dimensiones de entrada	32	24	29
Dimension de salida	32	24	1

Fuente: elaboración propia.

En la Figura 23 las predicciones de ejemplo en wide\_window, muestra cómo en muchos casos la predicción es mejor que devolver el valor de la temperatura de entrada, pero en algunos casos no es tan acertada.

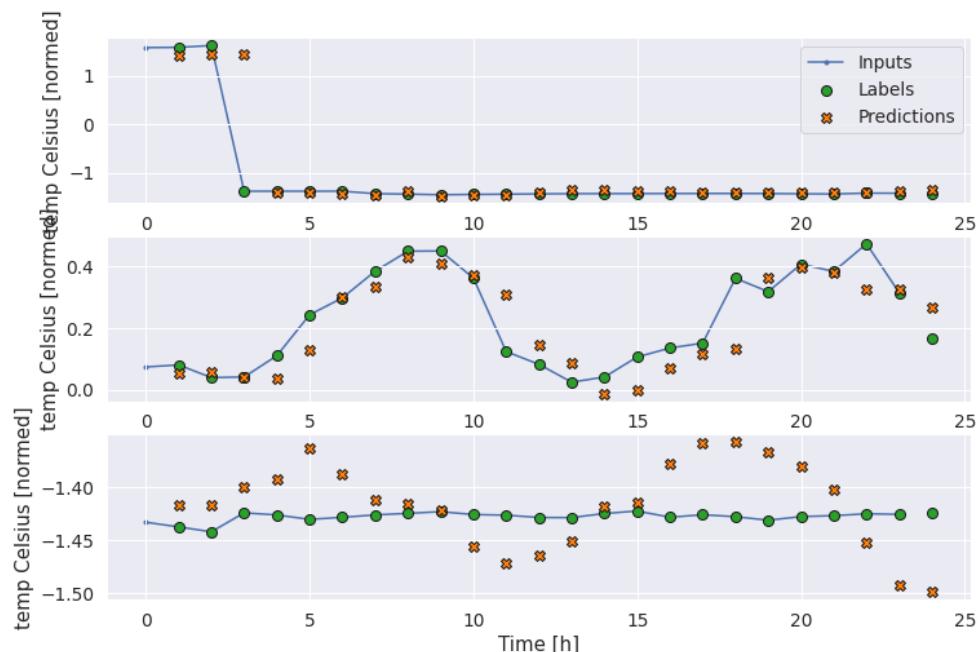


Figura 23 Modelo lineal de un solo paso Fuente: Elaboración propia

Una ventaja de los modelos lineales es que son relativamente simples de interpretar. Puede extraer los pesos de la capa y ver el peso asignado a cada entrada.

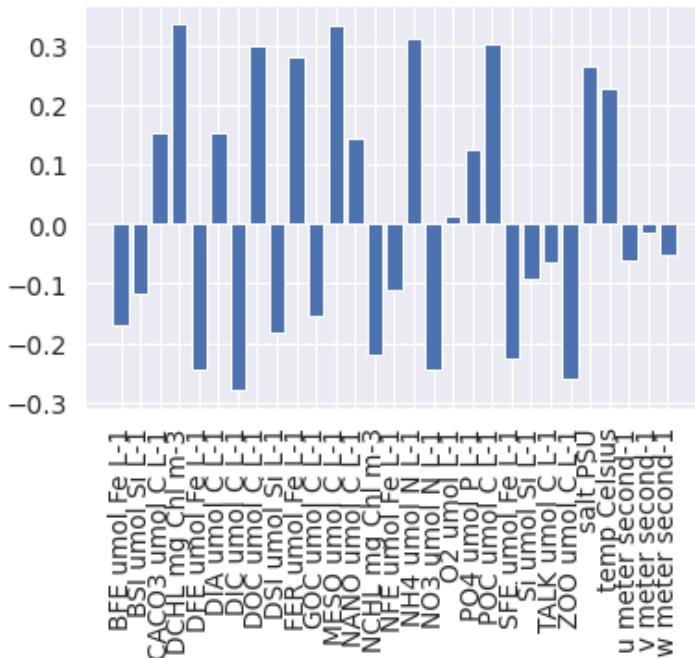


Figura 24 Peso asignado a modelo lineal de un solo paso Fuente: Elaboración propia

#### **4.12.1.3 Denso:**

Antes de aplicar modelos que realmente operan en múltiples pasos de tiempo, es necesario verificar el rendimiento de los modelos de pasos de entrada únicos más profundos y potentes.

Aquí hay un modelo similar al modelo lineal, excepto que apila varias capas Dense entre la entrada y la salida.

Tabla 22 Diseño del modelo dense secuencial para un solo paso

Modelo	dense secuential		
	capa	units	activation
primera capa	Dense	64	relu
segunda capa	Dense	64	relu
Capa de salida	Dense	1	

Configuración del modelo			
MAX_EPOCHS	20		
earling_stopping	monitor	val_loss	
	patience	2	

	mode	min
<b>model configuration</b>	loss	MeanSquaredError
	optimizers	Adam
	metrics	MeanAbsoluteError
<b>Metricas del modelo</b>		
Función de perdida	Función de metrica	
MeanSquaredError	MeanAbsoluteError	
0,0037	0,0318	

Fuente: elaboración propia.

#### 4.12.1.4 Denso de varios pasos:

Un modelo de paso de tiempo único no tiene contexto para los valores actuales de sus entradas. No puede ver como cambian las características de entrada con el tiempo. Para abordar este problema, el modelo necesita acceso a múltiples pasos de tiempo al hacer predicciones, los modelos baseline, lineal y dense manejan cada paso de tiempo de forma independiente. Aquí el modelo tomará múltiples pasos de tiempo como entrada para producir una única salida.

Se crea un WindowGenerator que producirá lotes de 3h de entradas y 1h de etiqueta, teniendo en cuenta que el parámetro de shift de Window es relativo al final de las dos ventanas.

Tabla 23 Ventana para la predicción de una hora en el futuro de acuerdo con 3 horas de entrada

Tamaño total de la ventana	4
Indices de entrada	[ 0 1 2 ]
Indicce de Salida	[3]
Etiqueta de columna	["temp Celsius"]

Fuente: elaboración propia.

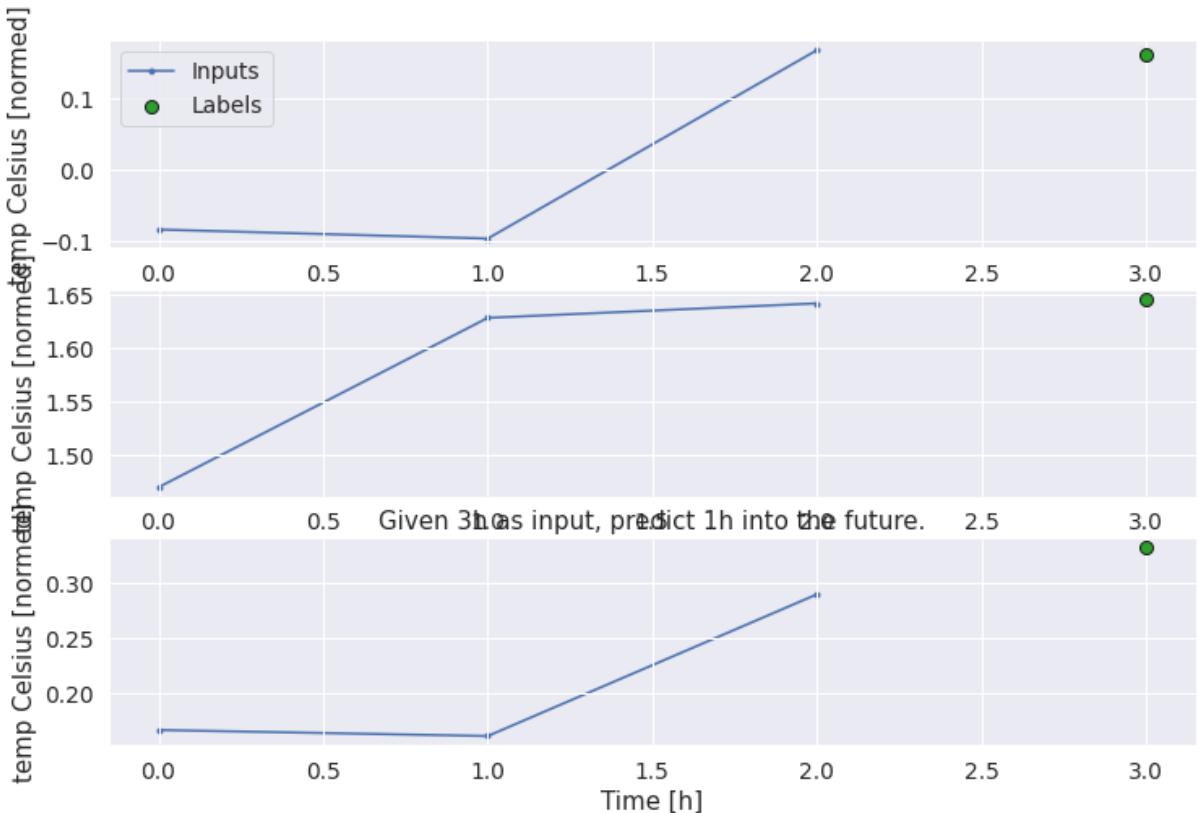


Figura 25 Dado 3h de entrada, predice 1h en el futuro Fuente: Elaboración propia

Se puede entrenar un modelo dense en una ventana de pasos de entrada múltiple, agregando layers.Flatten como la primera capa del modelo.

Tabla 24 Diseño del modelo *multi\_step\_dense* secuencial para un solo paso

Modelo	multi_step_dense		
	capa	units	activation
primera capa	Flatten		
segunda capa	Dense	32	relu
tercera capa	Dense	32	relu
cuarta capa	Dense	1	
capa de salida	Reshape	[1, -1]	
<b>Configuración del modelo</b>			
<b>MAX_EPOCHS</b>	20		
<b>earling_stopping</b>	monitor	val_loss	
	patience	2	
	mode	min	
<b>model configuration</b>	loss	MeanSquaredError	
	optimizers	Adam	
	metrics	MeanAbsoluteError	

Metricas del modelo	
Función de perdida	Función de metrica
MeanSquaredError	MeanAbsoluteError
0,0084	0,0277

Fuente: elaboración propia.

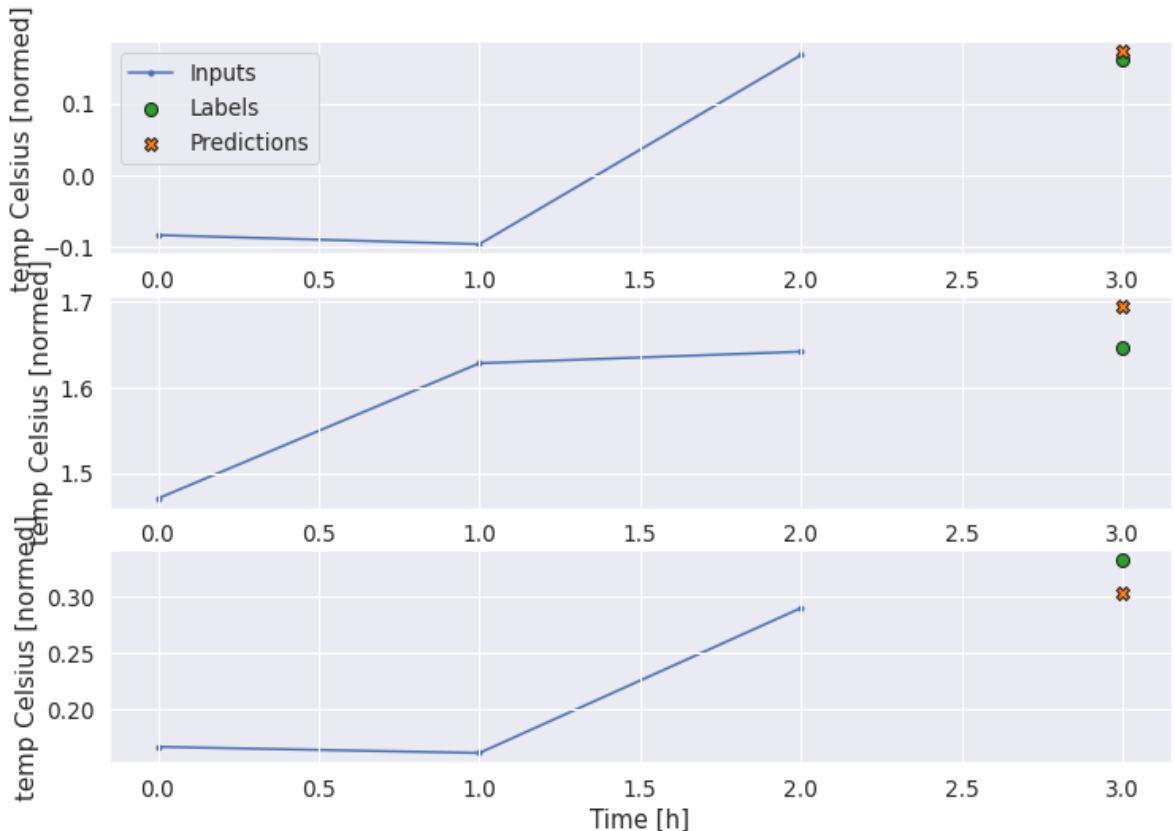


Figura 26 Dado 3h de entrada, predice 1h en el futuro multi-paso denso Fuente: Elaboración propia

El principal inconveniente de este enfoque es que el modelo resultante solo se puede ejecutar en ventanas de entrada de esta forma:

Tabla 25 Configuración de ventanas de entrada para el enfoque multi-paso denso

Dimensiones	lote	tiempo	Características
Dimensiones de entrada	32	24	29

Fuente: elaboración propia.

Error en el valor; la entrada 0 de la capa dense\_4 es incompatible con la capa: se esperaba que el eje -1 de la forma de entrada tuviera el valor 87 pero se recibió la entrada con la forma (32, 696). Los modelos convolucionales solucionan este problema.

#### 4.12.1.5 Red neuronal convolucionales:

Una capa de convolución (layers.Conv1D), también toma varios pasos de tiempo como entrada para cada predicción. A continuación, se muestra el mismo modelo que multi\_step\_dense, reescrito con una convolución.

Teniendo en cuenta los cambios:

- Las capas layers.Flatten y las primeras capas layers.Dense se reemplazan por capas layers.Conv1D.
- Las capas layers.Reshape ya no son necesarias debido a que la convolución mantiene el eje de tiempo en su salida.

Se ejecuta un lote para ver que el modelo produce salidas con la forma esperada.

Tabla 26 Dimensiones del modelo *Conv model on conv\_window*

Dimensiones	lote	tiempo	Características
Dimensiones de entrada	32	3	29
Dimension de salida	32	1	1

Fuente: elaboración propia.

La conv\_model debería dar un rendimiento similar al modelo multi\_step\_dense.

Tabla 27 Diseño del modelo convolucional para un solo paso

Modelo	Convolucional				
	capa	units	activation	filters	Kernel_size
primera capa	Conv1D	-	relu	32	(CONV_WIDTH,)
segunda capa	Dense	32	relu		
Capa de salida	Dense	1			
Configuración del modelo					
MAX_EPOCHS	20				
earling_stopping	monitor			val_loss	
	patience			2	
	mode			min	
model configuration	loss			MeanSquaredError	
	optimizers			Adam	
	metrics			MeanAbsoluteError	
Metricas del modelo					

Función de perdida	Función de metrica
MeanSquaredError	MeanAbsoluteError
0,0041	0,0251

Fuente: elaboración propia.

La diferencia entre conv\_model y el modelo multi\_step\_dense es que conv\_model se puede ejecutar en entradas de cualquier longitud. La capa convolucional se le aplica en una ventana deslizante de entradas, de este modo ver si se ejecuta en una entrada más amplia, produciendo una salida más amplia.

Teniendo en cuenta que la salida es más corta que la entrada, para que el entrenamiento o el trazado funcionen, se necesita que las etiquetas y la predicción tengan la misma longitud. Por lo tanto, es necesario producir ventanas amplias con algunos pasos de tiempo de entrada adicionales para que coincida con las longitudes de la etiqueta y la predicción.

Ahora se pueden trazar las predicciones del modelo en una ventana más amplia, teniendo en cuenta los 3 pasos de tiempo de entrada antes del primer pronóstico.

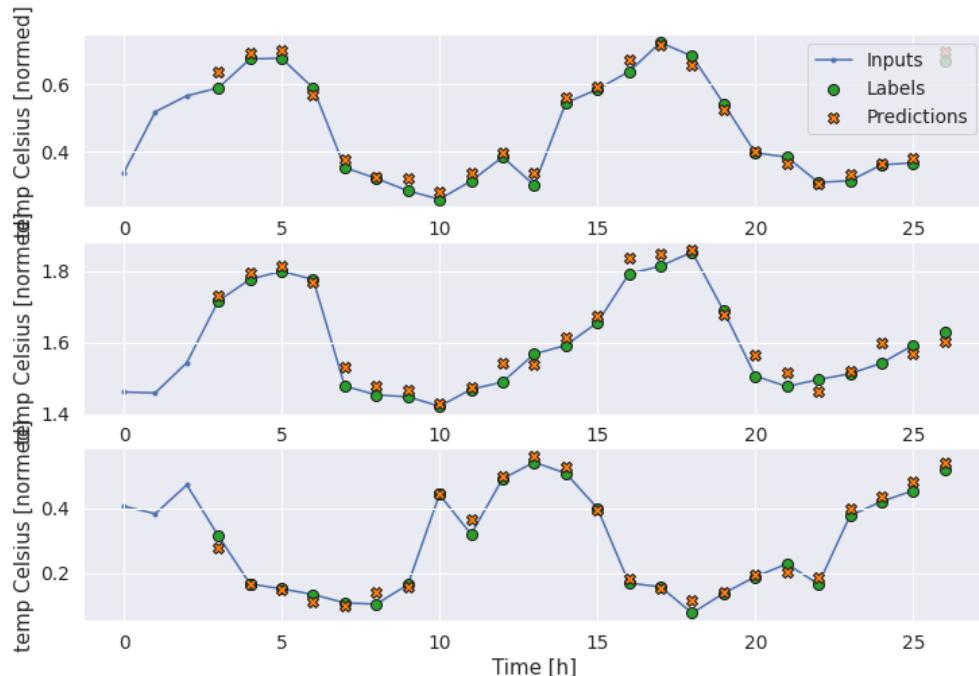


Figura 27 Predicción de una red neuronal convolucionales teniendo en cuenta 3 pasos de tiempo anteriores Fuente: Elaboración propia

#### 4.12.1.6 Red neuronal recurrente:

Una red neuronal recurrente (RNN) es un tipo de red neuronal adecuada para datos de series de tiempo. Las RNN procesan una serie de tiempo paso a paso, manteniendo un estado interno guardado al paso del tiempo.

Se usará una capa RNN llamada Long Short Term Memory (LSTM).

Un argumento importante en la construcción de todas las capas en keras RNN es el argumento `return_sequences`. Esta representación puede configurar la capa de una de dos formas.

- Si es `False`, el valor predeterminado, la capa solo devuelve el resultado del paso de tiempo final, lo que le da tiempo al modelo para calentar su estado interno antes de hacer una sola predicción:
- Si es `True` la capa devuelve una salida para cada entrada. Esto es útil para:
  - El apilamiento de capas RNN.
  - Entrenamiento de un modelo en múltiples pasos de tiempo simultáneamente.

Con `return_sequences = True`, el modelo se puede entrenar en 24 horas de datos a la vez.

Tabla 28 Diseño del modelo convolucional para un solo paso.

Modelo	RNN LSTM		
	capa	units	return_sequences
primera capa	LSTM	32	True
capa de salida	Dense	1	
<b>Configuración del modelo</b>			
<b>MAX_EPOCHS</b>	20		
<b>earling_stopping</b>	monitor	val_loss	
	patience	2	
	mode	min	
<b>model configuration</b>	loss	MeanSquaredError	
	optimizers	Adam	
	metrics	MeanAbsoluteError	
<b>Metricas del modelo</b>			
Función de perdida		Función de metrica	

MeanSquaredError	MeanAbsoluteError
8,2347e-04	0,0149

Fuente: elaboración propia.

Nota: Esto le dará una visión pesimista del desempeño del modelo. En el primer paso de tiempo, el modelo no tiene acceso a los pasos anteriores, por lo que no puede funcionar mejor que los modelos simples lineal y dense mostrados anteriormente.

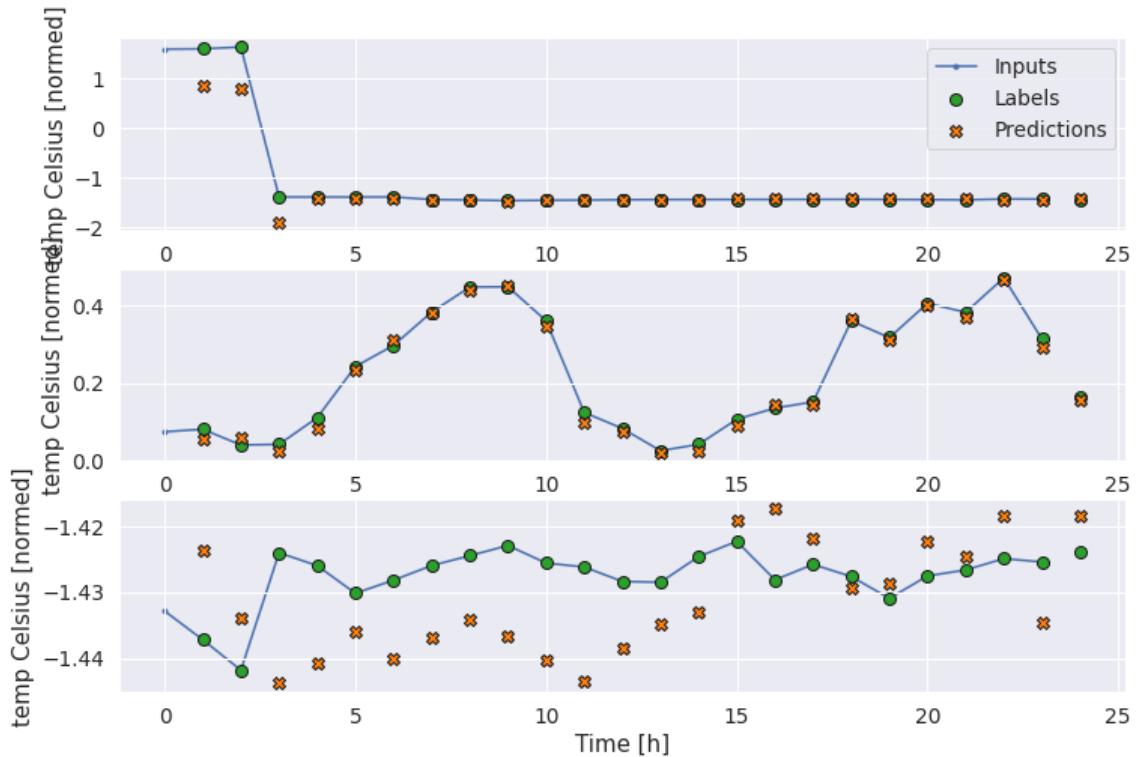


Figura 28 Predicción de una red neuronal recurrente 24h de datos a la vez Fuente: Elaboración propia

#### 4.12.2 Modelos de múltiples salidas:

Hasta ahora, todos los modelos predijeron una característica de salida única (temp Celsius), para un solo paso de tiempo.

Todos estos modelos se pueden convertir para predecir múltiples características, simplemente cambiando el número de unidades en la capa de salida y ajustando las ventanas de entrenamiento para incluir todas las características en las etiquetas.

Teniendo en cuenta que el eje de *features* o características de las etiquetas ahora tiene la misma profundidad que las entradas, en lugar de 1.

#### 4.12.2.1 Base:

Aquí se puede utilizar el mismo modelo de referencia, pero esta vez repitiendo todas las funciones en lugar de seleccionar un *label\_index* específico.

Tabla 29 Datos del modelo de línea base (baseline) múltiples salidas un paso.

<b>base</b>	Función de perdida	Función de metrica
	MeanSquaredError	MeanAbsoluteError
	0,0862	0,1212

Fuente: elaboración propia.

#### 4.12.2.2 Dense:

Tabla 30 Datos del modelo dense múltiples salidas un paso.

<b>Modelo</b>	<b>Dense multi_features</b>				
	capa	units	activation		
primera capa	Dense	64	relu		
segunda capa	Dense	32	relu		
Capa de salida	Dense	num_features			
<b>Configuración del modelo</b>					
<b>MAX_EPOCHS</b>	20				
<b>earling_stopping</b>	monitor	val_loss			
	patience	2			
	mode	min			
<b>model configuration</b>	loss	MeanSquaredError			
	optimizers	Adam			
	metrics	MeanAbsoluteError			
<b>Metricas del modelo</b>					
Función de perdida	Función de metrica				
MeanSquaredError	MeanAbsoluteError				
0,0047	0,0353				

Fuente: elaboración propia.

#### 4.12.2.3 Red neuronal recurrente RNN:

Tabla 31 Datos del modelo recurrente LSTM múltiples salidas un paso.

Modelo	RNN LSTM				
	capa	units	return_sequences		
primera capa	LSTM	32	True		
capa de salida	Dense	num_features			
<b>Configuración del modelo</b>					
<b>MAX_EPOCHS</b>	20				
<b>earling_stopping</b>	monitor	val_loss			
	patience	2			
	mode	min			
<b>model configuration</b>	loss	MeanSquaredError			
	optimizers	Adam			
	metrics	MeanAbsoluteError			
<b>Metricas del modelo</b>					
Función de perdida	Función de metrica				
MeanSquaredError	MeanAbsoluteError				
8,2347e-04	0,0149				

Fuente: elaboración propia.

#### 4.12.2.4 Avanzado: conexiones residuales

El modelo de Baseline anterior aprovechó el hecho de que la secuencia no cambia drásticamente de un paso de tiempo a otro. Todos los modelos entrenados hasta ahora, se inicializaron aleatoriamente y luego tuvieron que aprender que el resultado es un pequeño cambio con respecto al paso de tiempo anterior.

Si bien se puede solucionar este problema con una inicialización cuidadosa, es más sencillo integrarlo en la estructura del modelo.

Es común en el análisis de series de tiempo, construir modelos que en lugar de predecir el siguiente valor, predicen cómo cambiará el valor en el siguiente paso de tiempo. De manera similar, las "redes residuales" o "ResNets" en el aprendizaje profundo, se refieren a arquitecturas en las que cada capa se suma al resultado acumulado del modelo.

Así es como se aprovecha el conocimiento utilizando el cambio que debe ser pequeño. Básicamente, esto inicializa el modelo para que coincida con la *baseline*. Esta tarea ayuda a

que los modelos converjan más rápido, con un rendimiento ligeramente mejor. Este enfoque se puede utilizar junto con cualquier modelo analizado.

Aquí se está aplicando al modelo LSTM, teniendo en cuenta el uso de `tf.initializers.zeros` permite garantizar que los cambios iniciales previstos sean pequeños y no dominen la conexión residual. Aquí no hay preocupaciones de ruptura de simetría para los gradientes, ya que los ceros solo se usan en la última capa.

Tabla 32 Datos del modelo recurrente LSTM múltiples salidas un paso.

Modelo	ResidualWrappe			
	Sequential			
	capa	units	return_sequences	Kernel_initializer
primera capa	LSTM	32	True	
capa de salida	Dense	num_features		<code>tf.initializers.zeros</code>
Configuración del modelo				
MAX_EPOCHS			20	
earling_stopping	monitor		val_loss	
	patience		2	
	mode		min	
model configuration	loss		MeanSquaredError	
	optimizers		Adam	
	metrics		MeanAbsoluteError	
Metricas del modelo				
Función de perdida			Función de metrica	
MeanSquaredError			MeanAbsoluteError	
0,0036			0,0311	

Fuente: elaboración propia.

#### 4.12.3 Modelos de varios pasos:

Tanto el modelo de salida única como el de salida múltiple de las secciones anteriores, hicieron predicciones de paso de tiempo único, 1h en el futuro. Esta sección analiza cómo expandir estos modelos para hacer predicciones de múltiples pasos de tiempo.

En una predicción de varios pasos, el modelo debe aprender a predecir un rango de valores futuros. Por lo tanto, a diferencia de un modelo de un solo paso, donde solo se predice un único punto futuro, un modelo de varios pasos predice una secuencia de los valores futuros.

Hay dos enfoques aproximados para esto:

- Predicciones de un solo disparo donde se predice toda la serie de tiempo a la vez.
- Predicciones autoregresivas donde el modelo solo hace anticipaciones de un solo paso y su salida se retroalimenta como su entrada.

En esta sección, todos los modelos predecirán las características en todos los pasos de tiempo de salida. Para el modelo de varios pasos, los datos de entrenamiento nuevamente consisten en muestras por hora. Sin embargo, aquí los modelos aprenderán a predecir las 24 horas del futuro, dadas las 24 horas del pasado.

Se muestra un objeto Window que genera estos cortes a partir del conjunto de datos.

Tabla 33 Configuración del objeto Window dado 24 horas pasadas para predecir las próximas 24 horas futuras

<b>Tamaño total de la ventana</b>	48
<b>Indices de entrada</b>	[ 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23]
<b>Indicce de Salida</b>	[24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47]
<b>Etiqueta de columna</b>	Ninguna

Fuente: elaboración propia.

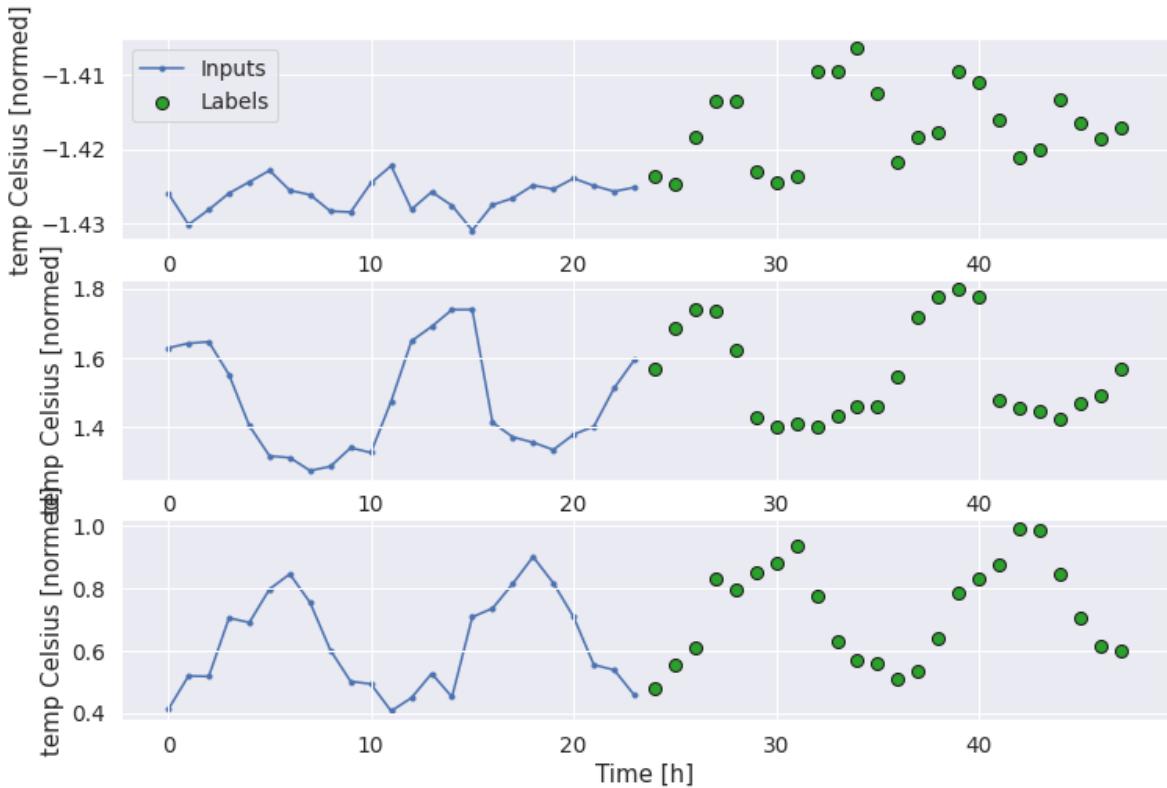


Figura 29 Generación de predicción de múltiples puntos a futuro Fuente: Elaboración propia

#### 4.12.3.1 Líneas base múltiples pasos

Una línea de base simple para esta tarea es repetir el último paso de tiempo de entrada para el número requerido de pasos de tiempo en la salida.

Tabla 34 Datos del modelo de línea base (*Last*) repitiendo el último paso de entrada.

<b>Last</b>	Función de perdida	Función de metrica
	MeanSquaredError	MeanAbsoluteError
	0, 3527	0, 2790

Fuente: elaboración propia.

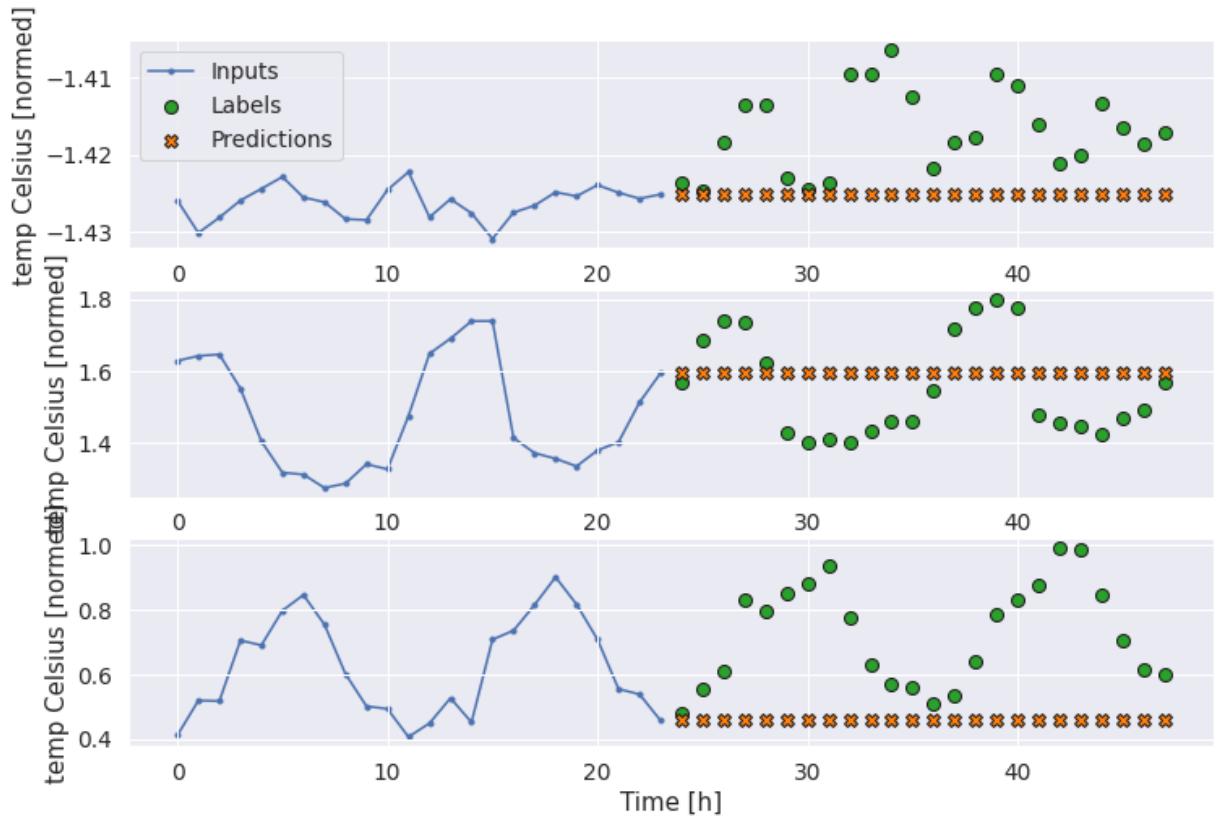


Figura 30 Modelo de línea base (Last) de múltiples pasos Fuente: Elaboración propia

Esta tarea es predecir 24 horas futuras dadas 24 horas pasadas. Otro enfoque simple es repetir el día anterior, asumiendo que mañana será similar.

Tabla 35 Datos del modelo de línea base (*Repeat*) repitiendo el día anterior asumiendo que mañana será igual.

Repeat	Función de perdida	Función de metrica
	MeanSquaredError	MeanAbsoluteError
	0,2144	0,1868

Fuente: elaboración propia.

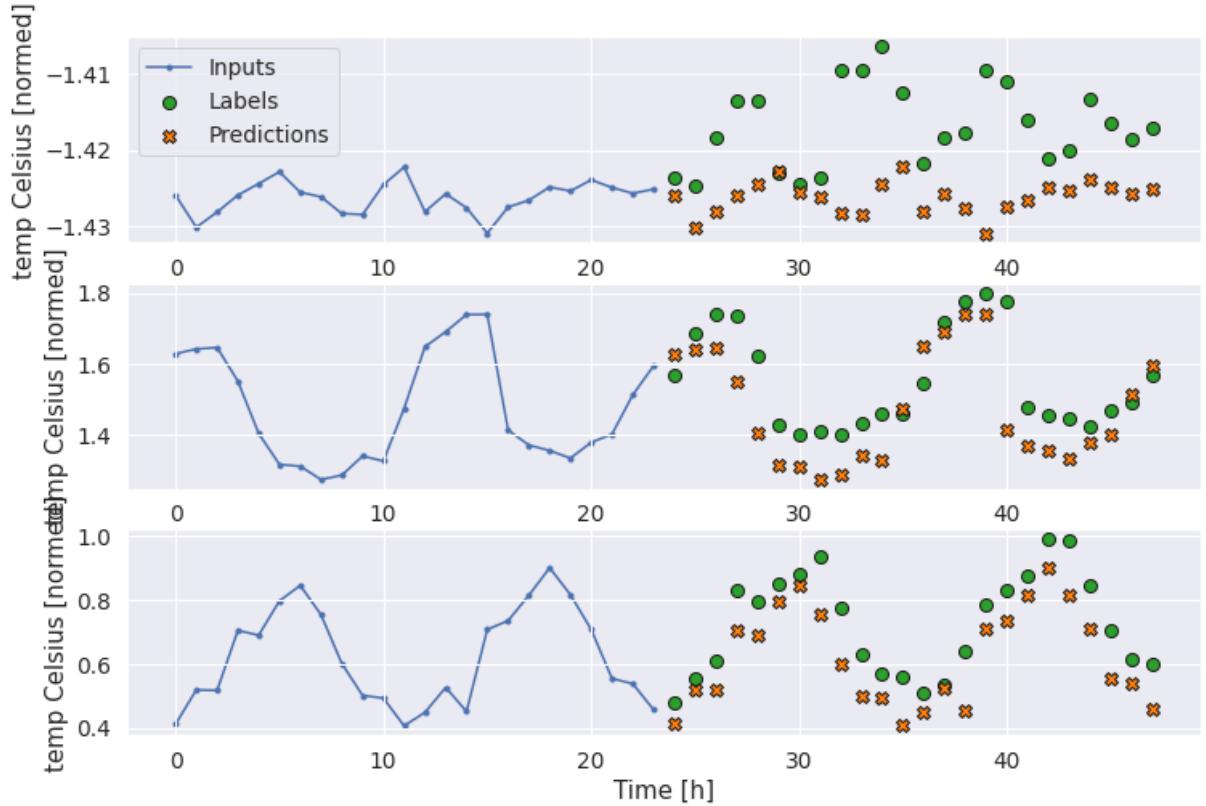


Figura 31 Modelo de línea base repetir el día anterior de múltiples pasos Fuente: Elaboración propia

#### 4.12.3.2 Modelos de disparo único con múltiples pasos

Un enfoque de alto nivel para este problema es, utilizar un modelo de "disparo único" en el cual el modelo hace la predicción de la secuencia completa en un solo paso. Esto se puede implementar de manera eficiente con la capa layers.Dense con OUT\_STEPS\*features para las unidades de salida. El modelo solo necesita remodelar esa salida a la requerida (OUTPUT\_STEPS, features).

#### 4.12.3.3 Lineal

Un modelo lineal simple basado en el último paso de tiempo de entrada funciona mejor que cualquier línea de base, pero no tiene suficiente potencia. El modelo necesita predecir los pasos de tiempo OUTPUT\_STEPS , a partir de un solo paso de tiempo de entrada con una proyección lineal. Solo puede capturar un segmento de baja dimensión del comportamiento, probablemente basado principalmente en la hora del día y la época del año.

Tabla 36 Datos del modelo lineal para múltiples pasos

Modelo	multi_linear_model		
	capa	batch	activation
primera capa	Lambda	lambda x: x[:, -1:, :]	
Segunda capa	Dense	OUT_STEPS*num_features	tf.initializers.zeros
capa de salida	Reshape	[OUT_STEPS, num_features]	
Configuración del modelo			
MAX_EPOCHS		20	
earling_stopping	monitor		val_loss
	patience		2
	mode		min
model configuration	loss		MeanSquaredError
	optimizers		Adam
	metrics		MeanAbsoluteError
Metricas del modelo			
Función de perdida		Función de metrica	
MeanSquaredError		MeanAbsoluteError	
0,1281		0,1882	

Fuente: elaboración propia.

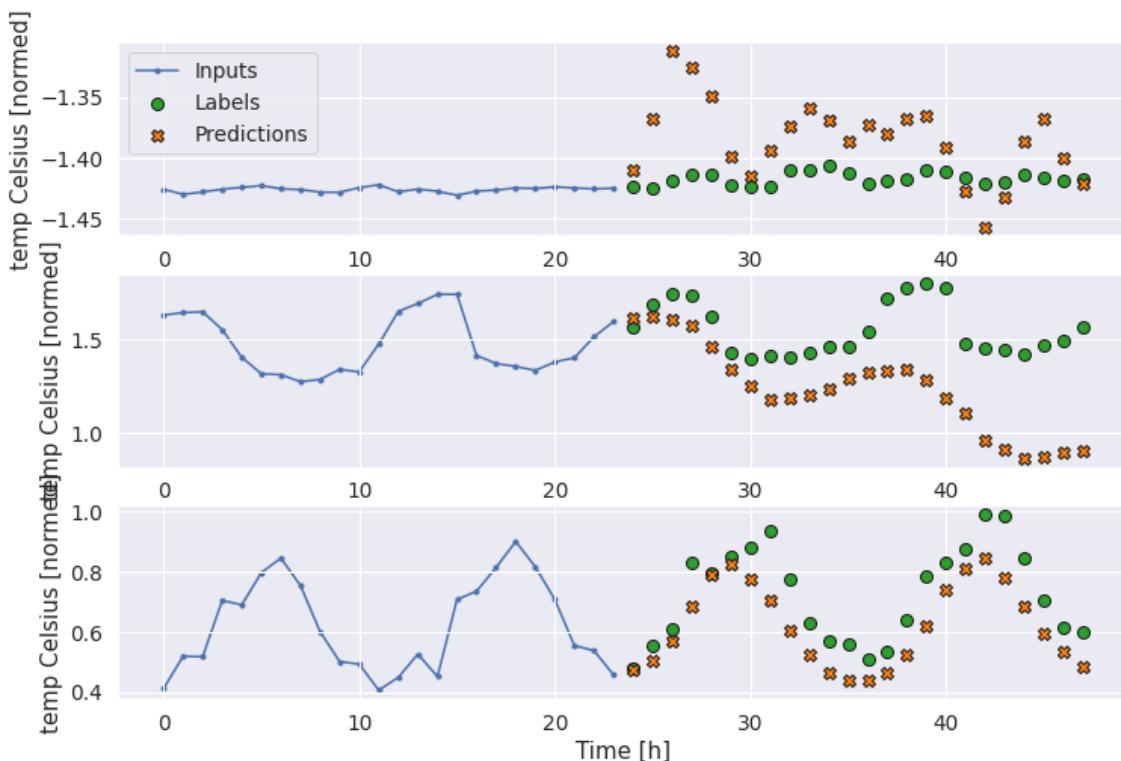


Figura 32 Modelo lineal para múltiples pasos Fuente: Elaboración propia

#### 4.12.3.4 Dense para múltiples pasos

Se agrega un layers.Dense entre la entrada y la salida, esto le da al modelo lineal más potencia, pero aún se basa en un solo paso de tiempo de entrada.

Tabla 37 Datos del modelo denso para múltiples pasos

Modelo	multi_dense_model				
	capa	batch	activation		
primera capa	Lambda	lambda x: x[:, -1:, :]			
Segunda capa	Dense	512	relu		
tercera capa	Dense	OUT_STEPS*num_features	tf.initializers.zeros		
capa de salida	Reshape	[OUT_STEPS, num_features]			
<b>Configuración del modelo</b>					
<b>MAX_EPOCHS</b>	20				
earling_stopping	monitor	val_loss			
	patience	2			
	mode	min			
model configuration	loss	MeanSquaredError			
	optimizers	Adam			
	metrics	MeanAbsoluteError			
<b>Metricas del modelo</b>					
Función de perdida	Función de metrica				
MeanSquaredError	MeanAbsoluteError				
0,0429	0,1129				

Fuente: elaboración propia.

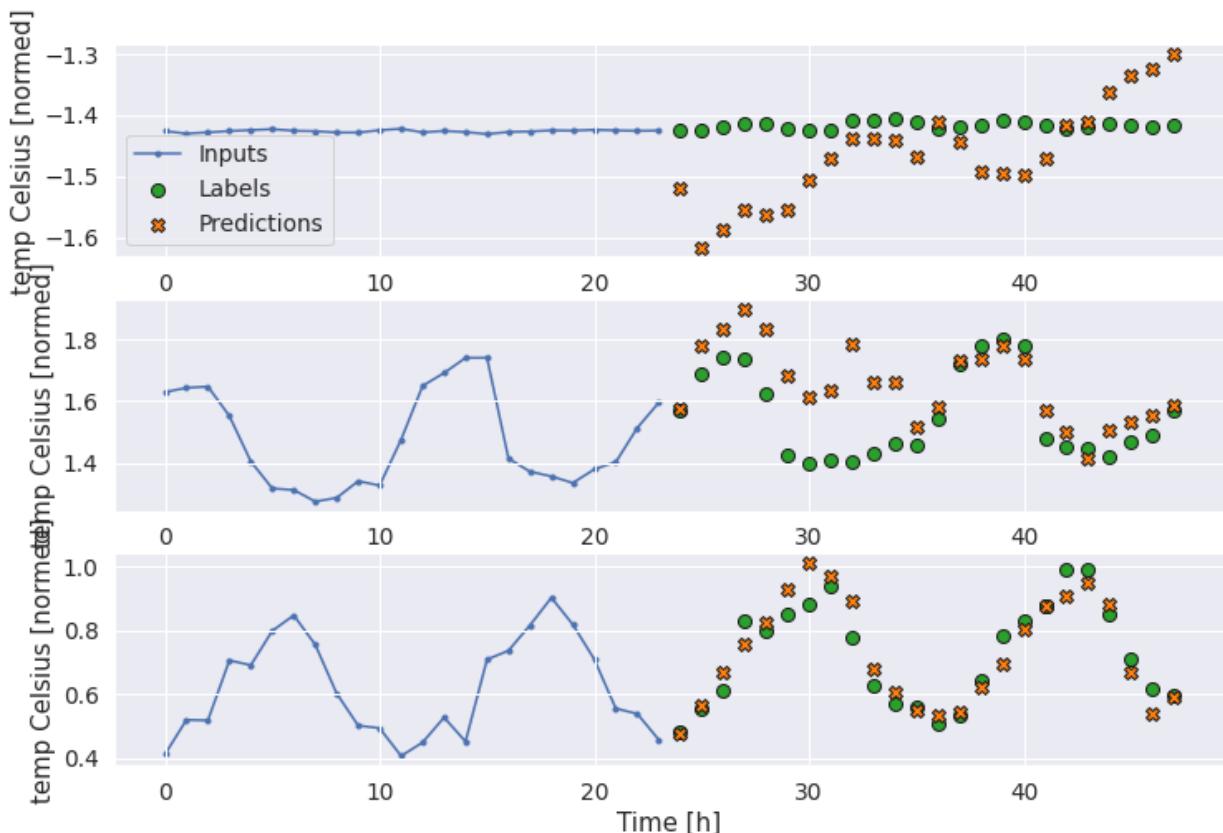


Figura 33 Modelo denso para múltiples pasos Fuente: Elaboración propia

#### 4.12.3.65 Redes neuronales convolucionales CNN para múltiples pasos

Un modelo convolucional hace predicciones basadas en un historial de ancho fijo, lo que puede conducir a un mejor rendimiento que el modelo *dense*, ya que puede ver cómo cambian las cosas con el tiempo.

Tabla 38 Datos del modelo denso para múltiples pasos

Modelo	multi_conv_model						
	capa		activation	kernel			
primera capa	Lambda	lambda x: x[:, - CONV_WIDTH:, :]					
Segunda capa	Conv1D	256	relu	Kernel_size=3			
tercera capa	Dense	OUT_STEPS*num_features		tf.initializers.zeros			
capa de salida	Reshape	[OUT_STEPS, num_features]					
Configuración del modelo							
MAX_EPOCHS	20						
earling_stopping	monitor	val_loss					
	patience	2					
	mode	min					
model configuration	loss	MeanSquaredError					
	optimizers	Adam					
	metrics	MeanAbsoluteError					
Metricas del modelo							
Función de perdida	Función de metrica						
MeanSquaredError	MeanAbsoluteError						
0,0259	0,0864						

Fuente: elaboración propia.

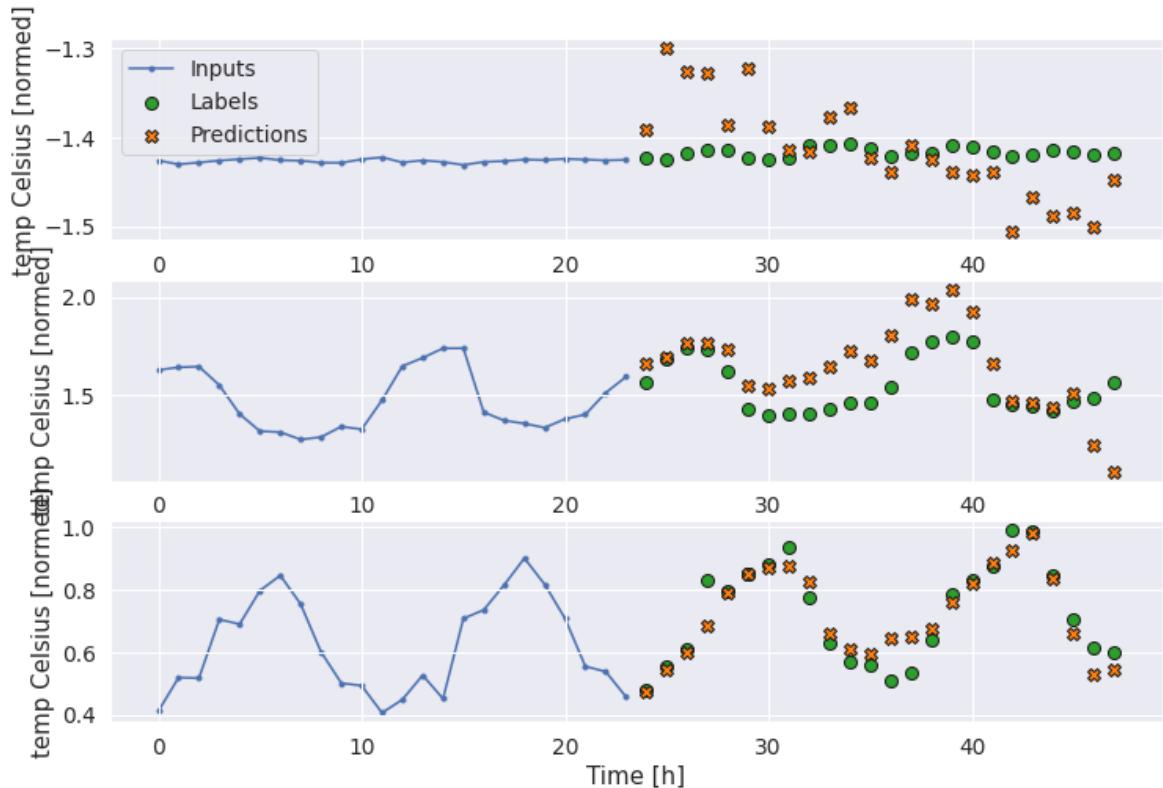


Figura 34 Modelo red neuronal convolucionales para múltiples pasos Fuente: Elaboración propia

#### 4.12.3.6 Red neuronal recurrente RNN para múltiples pasos.

Un modelo recurrente, puede aprender a usar un largo historial de entradas si es relevante para las predicciones que hace el modelo. Aquí, el modelo acumulará el estado interno durante 24 h, antes de realizar una única predicción para las próximas 24 h.

En este formato de un solo disparo, el LSTM solo necesita producir una salida en el último paso de tiempo, así que configure `return_sequences=False`.

Tabla 39 Datos del modelo multi LSTM para una salida.

Modelo	Multi RNN LSTM			
	capa	units	return_sequence s	kernel_initializer
primera capa	LSTM	32	False	
Segunda capa	Dense	OUT_STEPS*num_features		tf.initializers.zeros
capa de salida	Reshape	num_features		
Configuración del modelo				
MAX_EPOCHS		20		
	monitor		val_loss	

<b>earling_stoppin g</b>	patience	2
	mode	min
<b>model configuration</b>	loss	MeanSquaredError
	optimizer	Adam
	metrics	MeanAbsoluteError
<b>Metricas del modelo</b>		
Función de perdida	Función de metrica	
MeanSquaredError	MeanAbsoluteError	
0,0192	0,0670	

Fuente: elaboración propia.

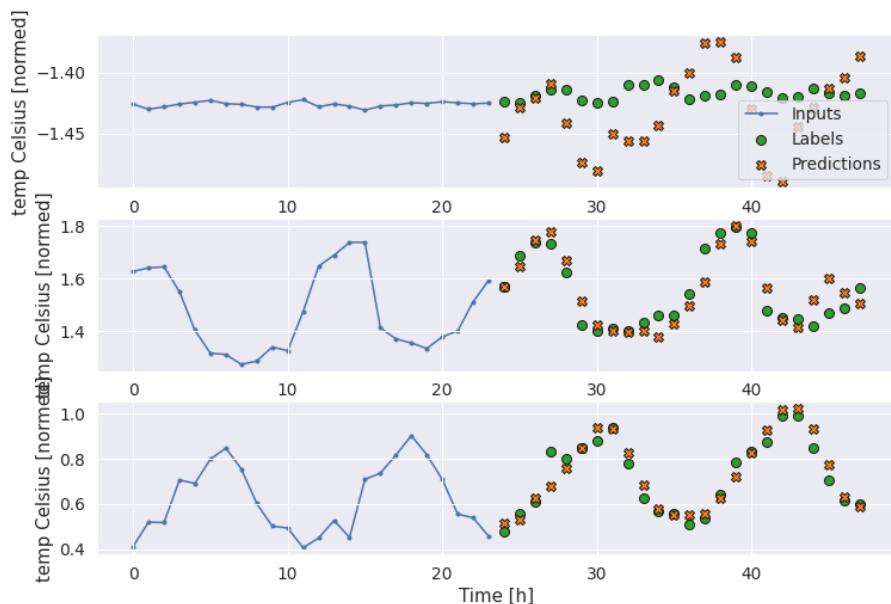


Figura 35 Modelo red neuronal recurrente para múltiples pasos a partir de acumular 24h Fuente: Elaboración propia

#### 4.12.3.7 Avanzado modelo auto regresivo.

Todos los modelos anteriores predicen la secuencia de salida completa en un solo paso. En algunos casos, puede ser útil que el modelo descomponga esta predicción en pasos de tiempo individuales. Luego, la salida de cada modelo se puede retroalimentar en cada paso y se pueden hacer predicciones condicionadas al anterior como el modelo clásico y generar secuencias con redes neuronales recurrentes.

Una clara ventaja de este estilo de modelo es que, se puede configurar para producir resultados con una longitud variable. Puede tomar cualquier modelo de salida múltiple de un solo paso, entrenados en la primera mitad de este tutorial y ejecutarlo en un ciclo de retroalimentación auto regresivo, pero aquí se enfocará en construir un modelo que haya sido

entrenado explícitamente para hacer eso. Retroalimentando la salida desde un modelo a su entrada.

#### 4.12.3.8 Red neuronal recurrente auto regresivo.

Solo se crea un modelo RNN auto regresivo, pero este patrón se puede aplicar a cualquier modelo que haya sido diseñado para generar un solo paso de tiempo. El modelo tendrá la misma forma básica como las de un solo paso LSTM, seguido por un layers.Dense que convierte los LSTM salidas a las predicciones del modelo.

Un layers.LSTM es un layers.LSTMCell envuelto en las layers.RNN que administra el estado y los resultados de la secuencia por usted (consulte los RNN de Keras para obtener más detalles). En este caso, el modelo tiene que administrar manualmente las entradas para cada paso, por lo que utiliza layers.LSTMCell directamente para el nivel inferior, interfaz de paso de tiempo único.

Lo que necesita este modelo es un método de warmup para inicializar su estado interno en función de las entradas. Una vez entrenado, este estado capturará las partes relevantes del historial de entrada. Esto es equivalente al modelo LSTM solo paso anterior.

Este método devuelve una única predicción de paso de tiempo y el estado interno del LSTM. Con el estado del RNN y una predicción inicial, ahora puede continuar iterando el modelo alimentando las predicciones en cada paso hacia atrás como entrada.

El enfoque más simple para recopilar las predicciones de salida es usar una lista de Python y tf.stack después del ciclo.

Entrenando el modelo muestra las siguientes métricas:

Tabla 40 Datos del modelo autoregresivo RNN para una salida.

Modelo	RNN autoregressive	
	units	out_steps
feedback_model	32	OUT_STEPS
<b>Configuración del modelo</b>		
MAX_EPOCHS	20	
earling_stopping	monitor	val_loss
	patience	2
	mode	min

model configuration	loss	MeanSquaredError
	optimizers	Adam
	metrics	MeanAbsoluteError
<b>Metricas del modelo</b>		
Función de perdida	Función de metrica	
MeanSquaredError	MeanAbsoluteError	
0,0183	0,0703	

Fuente: elaboración propia.

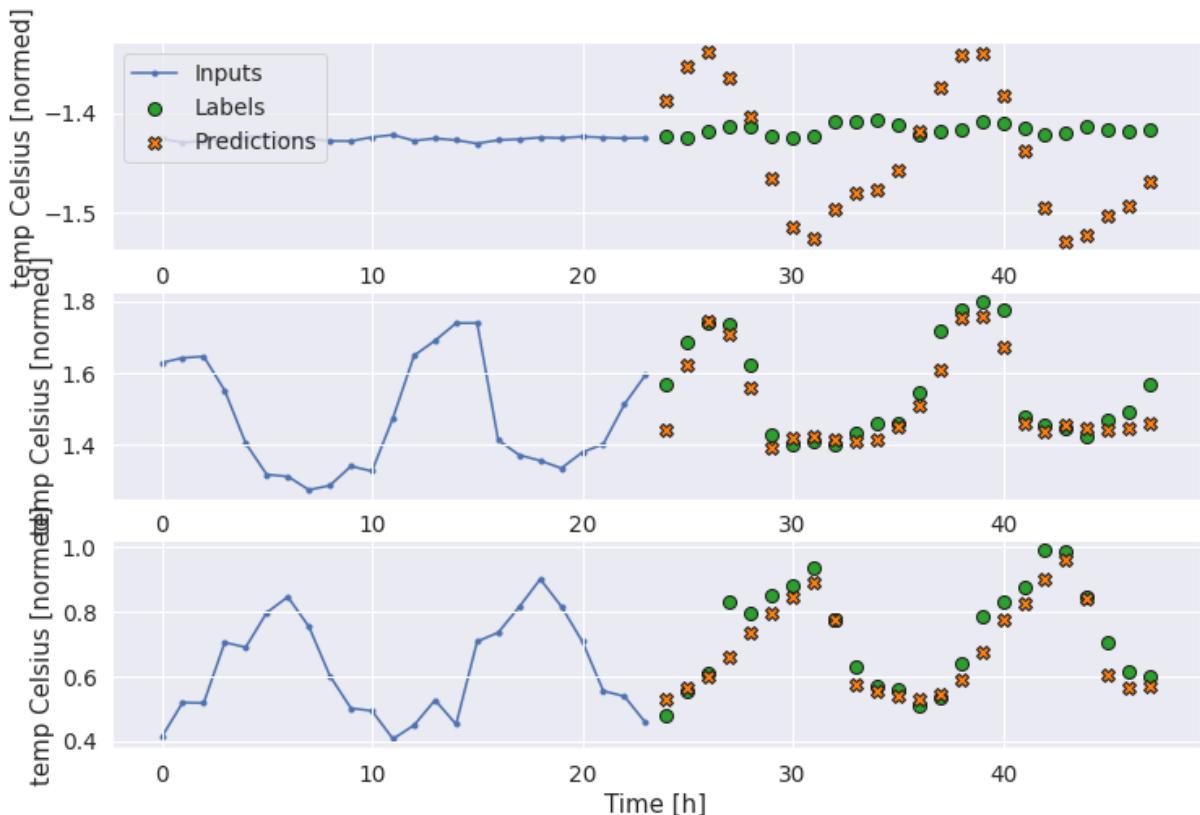


Figura 36 Modelo red neuronal recurrente autoregresiva múltiples pasos Fuente: Elaboración propia

## 5. Descripción de los resultados

Para los modelos de una sola salida se puede verificar su actuación utilizando los datos de validación y de prueba de los modelos de una sola salida.

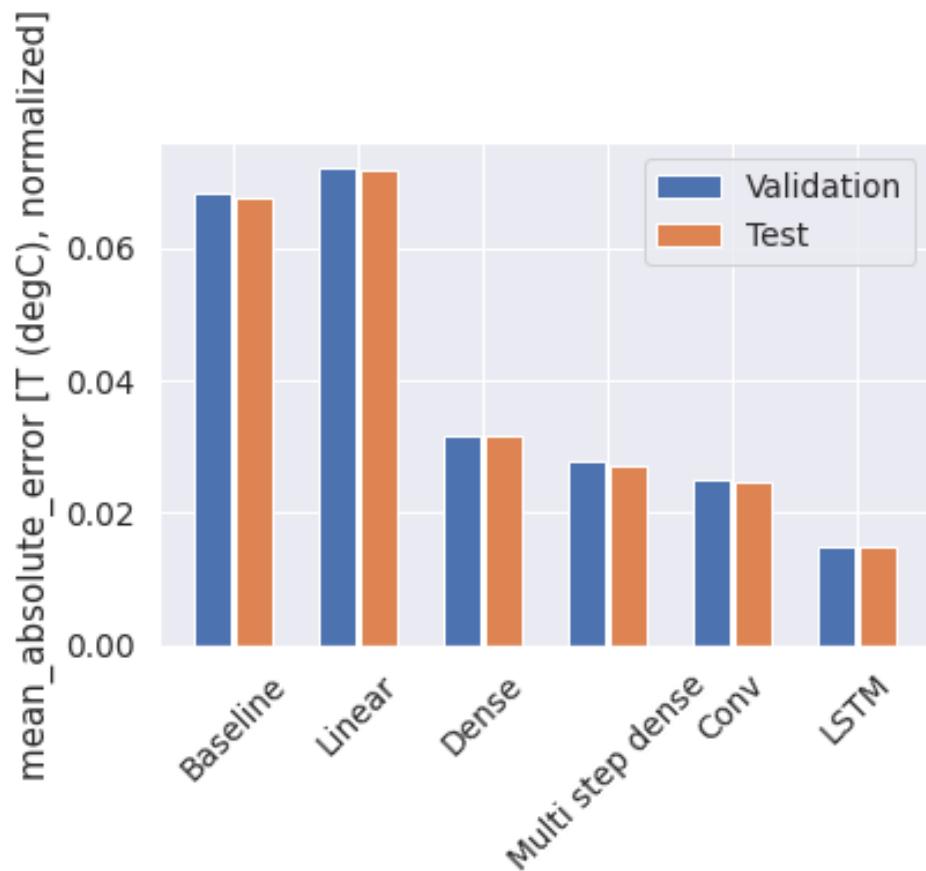


Figura 37 Actuación de modelos de un solo paso una salida Fuente: Elaboración propia

De esta forma se presentan los datos de los rendimientos de los diferentes modelos en la siguiente tabla:

Tabla 41 Rendimiento de los modelos de un solo paso para una salida

Modelos	Mean Abosulte Error (MAE)
Baseline	0.0677
Linear	0.0717
Dense	0.0317
Multi step dense	0.02713
Conv	0.0248
LSTM	0.0148

Fuente elaboración propia.

A continuación, se muestra el rendimiento general de estos modelos de múltiples salidas, para esto se exponen los promedios de todas las salidas en base a los modelos que se probaron en esta ocasión.

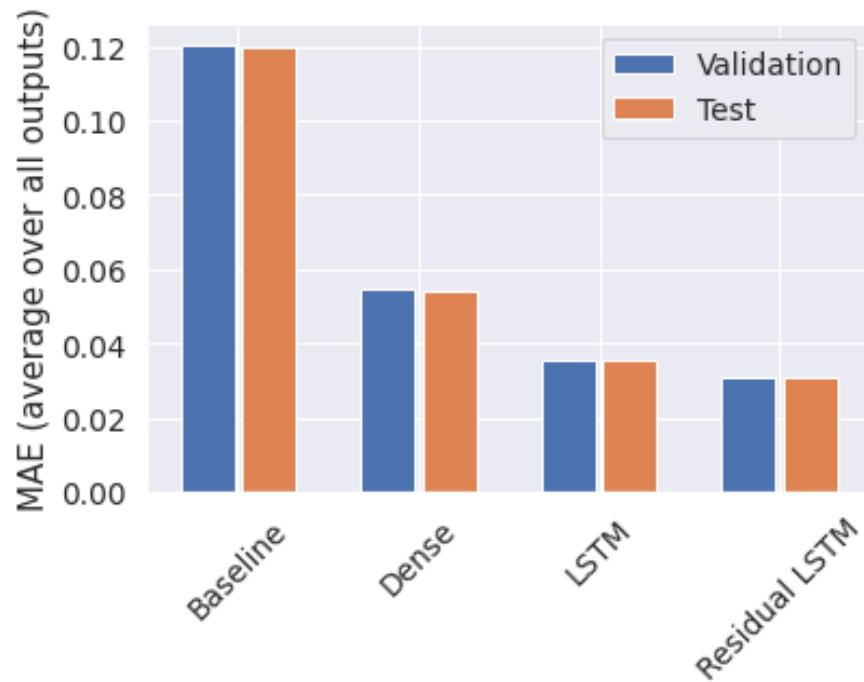


Figura 38 Actuación de modelos de un solo paso para múltiples salidas Fuente: Elaboración propia

En la siguiente tabla se muestran los rendimientos de los modelos de un solo paso para múltiples salidas.

Tabla 42 Rendimiento de los modelos de un solo paso para salidas múltiples

Modelo	Mean Absolute Error (MAE)
Baseline	0.1200
Dense	0.0543
LSTM	0.0353
Residual LSTM	0.0311

Fuente elaboración propia.

Los rendimientos anteriores se promedian en todos los resultados del modelo.

La actuación de los modelos de múltiples pasos para una salida, muestran los rendimientos claramente decrecientes en función de la complejidad del modelo sobre este problema, por lo cual se presenta la validación y la prueba en la siguiente grafica.

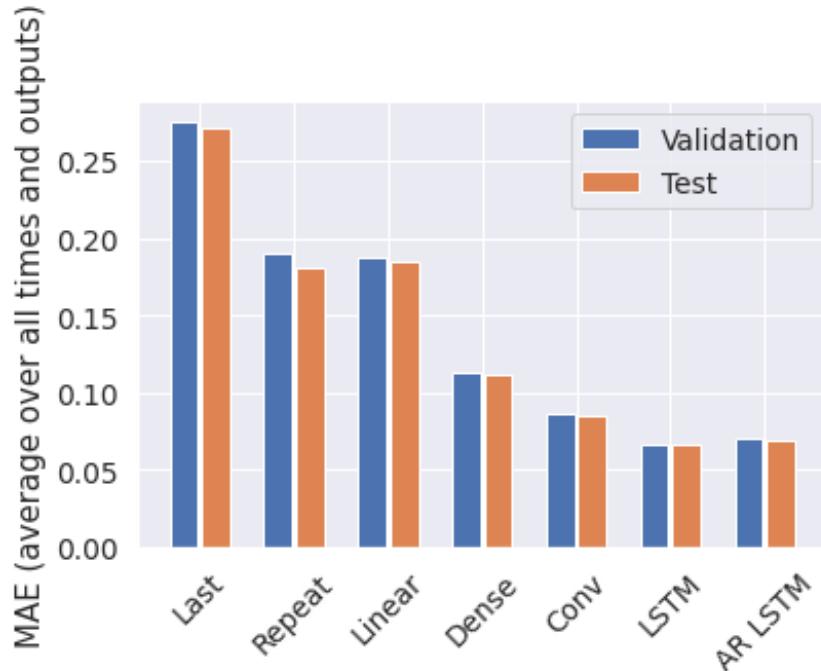


Figura 39 Actuación de modelos de varios pasos para una salida Fuente: Elaboración propia

Las métricas para los modelos de múltiples salidas, en la primera mitad de este trabajo muestran el rendimiento promedio en todas las funciones de salida. Estos desempeños son similares pero también promediados en los pasos de tiempo de salida.

Tabla 43 Rendimiento de los modelos de varios pasos para una salida

Modelos	Mean Abosulte Error (MAE)
Last	0.2715
Repeat	0.1808
Linear	0.1848
Dense	0.1111
Conv	0.0847
LSTM	0.0664
AR LSTM	0.0696

Fuente elaboración propia.

Las ganancias logradas al pasar de un modelo denso a modelos convolucionales y recurrentes son solo un pequeño porcentaje, el modelo autoregresivo tuvo un desempeño ligeramente más bajo que la red recurrente. Por lo tanto, es posible que estos enfoques más complejos no mejoren a gran escala lo que con otros modelos más sencillos ya se había conseguido, pero no había forma de saberlo sin intentarlo. De todas maneras estos modelos podrían ser útiles para la resolución del problema.

## 6. Discusión

En el capítulo 5 se ha presentado la implementación y evaluación de diferentes modelos de redes de neuronas profundas, para resolver el problema de predicción de serie temporales de variables oceanográficas, presentes en un centro de cultivo de salmonidos, las cuales pueden ser usadas para prevenir impactos ambientales dentro de sus procesos operativos, así como detectar la presencia de externalidades negativas en alguna variable que pueda afectar el proceso.

Para el experimento se llevaron a cabo 3 procesos principales que internamente se derivaron en otros modelos a evaluar. Esta serie de modelos que se desarrollaron tenían diferentes finalidades; la predicción de un solo paso para una salida, la predicción de un solo paso para múltiples salidas y la predicción de varios pasos para una salida. Cabe destacar que los diferentes experimentos, cuya finalidad era obtener la predicción de una sola salida, consideró la variable de temperatura como piloto y así poder validar conceptualmente el experimento.

En primer lugar, se realizaron una serie de pruebas para llevar a cabo la predicción de un solo paso para una salida. Se generó una línea base para poder tener una referencia de partida, pasando a evaluar un modelo lineal, el cual no mejoró la línea base previa que se había desarrollado.

Se va generando una mejoría cuando ya se empiezan a explorar las redes profundas, es aquí cuando comienza a mejorar el rendimiento del modelo, pasando a una red convolucional donde sigue mejorando el modelo hasta llegar a la red neuronal recurrente LSTM, con la cual se consigue el mejor rendimiento obteniendo, el *mean absolute error* de 0,0148. Con la información que se puede analizar para generar la predicción de la variable piloto “temperatura”, se puede mencionar que hay un mejor comportamiento, al utilizar este tipo de red en la predicción de un valor en el tiempo.

Como segundo experimento se procedió a probar una serie de modelos que permitieron evaluar el comportamiento de las predicciones a un salto de tiempo para múltiples salidas, es decir, se le añade complejidad al experimento anterior. De la misma manera, se parte con una línea base de referencia que ayuda a comprender el comportamiento de los modelos a evaluar. En este caso se inicia probando una red profunda, con la cual empiezan a mejorar las métricas obtenidas por la línea base.

En miras de aprovechar la experiencia de los modelos pasados, se procede a realizar pruebas con una red neural recurrente LSTM, que también mejora las métricas obtenidas.

En este experimento se logró mejorar el resultado con el diseño de una red neuronal recurrente residual LSTM, con la cual se obtuvo una métrica del *mean absolute error* de 0,0311. Lo que igual determina una mejoría entre los modelos explorados y entrega una perspectiva ante múltiples salidas que puedan aportar en una predicción con más información, la que puede ser aprovechada dentro de los centros acuícolas.

Como último experimento, se probaron diferentes modelos para poder determinar varios pasos para una salida. En este caso se realizaron pruebas como; ver el pasado, repetir los valores, realizar un modelo lineal, hasta ir llegando a los modelos profundos, pasando por una red convolucional, luego se prueba una red neuronal recurrente y una red neuronal recurrente auto regresiva.

Se puede decir que las ganancias logradas al pasar de un modelo denso a modelos convolucionales y recurrentes son un pequeño porcentaje. El modelo autoregresivo no mejoró mucho respecto al modelo recurrente LSTM el cual tuvo un rendimiento del 0,0664.

Se pudo observar que, con los diferentes comportamientos de los modelos, las redes neuronales recurrentes en sus diferentes variantes ayudaron en las predicciones temporales.

Por lo tanto, si se quiere realizar un análisis dentro de algún centro de cultivo acuícola, estos modelos pueden aportar información valiosa para los procesos productivos, así como también contribuir en la prevención del impacto ambiental dentro del espacio acuícola.

## 7. Conclusiones y trabajo futuro

A continuación, se presenta el resumen de este trabajo de fin de máster. En primer lugar, se presentan las conclusiones, relacionándolas con los objetivos planteados. Luego, se comentan las posibles líneas de trabajo futuro que pueden aportar un valor añadido.

### 7.1. Conclusiones

El presente trabajo de fin de máster fue direccionado por su objetivo general:

*Desarrollar un prototipo experimental para la predicción de impactos ambientales en un ecosistema marítimo en tiempo real integrando los datos obtenidos de los modelos oceanográficos, hidrodinámicos y biogeoquímicos a través de algoritmos de inteligencia artificial, comparando diferentes modelos para verificar cual es la mejor métrica.*

El primer objetivo específico, fue explorar los datos obtenidos de los diferentes modelos oceanográficos, hidrodinámicos y biogeoquímicos. Esto se ha cumplido, debido a que se generó un proceso de exploración inicial de los datos, provenientes de la salida de los modelos oceanográficos CROCO-OCEAN, el cual era fundamental para comprender con que datos se contaba, tanto en unidades de medición, dimensiones, formato, peso de los archivos, como para poder tratar los archivos. Con estos se pudo determinar que los datos que provienen de los modelos oceanográficos están diseñados en NetCDF, un formato que es utilizado para guardar información geográfica, de igual manera fue relevante conocer el peso de los archivos, ya que permitió acomodar los recursos que se necesitaban para poder analizar a profundidad las variables que están dentro de los archivos.

El segundo objetivo específico fue, analizar las diversas variables que involucran los diferentes modelos oceanográficos, de acuerdo con la información obtenida en el sector piloto. Este objetivo es ampliamente cumplido, ya que se pudo analizar inicialmente los archivos de trabajo. Se verificó que se poseía información extensa de todo un sector, por lo que se decidió tomar una grilla para el trabajo. Dentro de la misma se seleccionaron unas variables provenientes de las salidas del modelo CROCO-OCEAN, de las cuales configuran sus dimensiones; el tiempo que se poseen 1023 datos cada punto, las profundidades que son 30 puntos, y además de los dos componentes  $x_j$  tanto  $y_j$  que determinan el punto geográfico en el espacio determinado para el estudio. Con este proceso se realizaron los análisis estadísticos de las diferentes variables y verificar que características estuvieron presentes en

el conjunto de datos, logrando una mayor comprensión para determinar los modelos que se utilizarán.

El tercer objetivo específico, pretende establecer las variables más relevantes para poder realizar el prototipo experimental. Este objetivo fue ampliamente abordado, ya que, al poseer las características de las variables de los modelos oceanográficos, se procedió a seleccionar el conjunto de datos con el que se trabajaría para la construcción de los modelos de inteligencia artificial. Se obtiene un dataset en formato csv para un mejor manejo del conjunto de datos, utilizando técnicas descriptivas estadísticas, además de realizar matriz de correlación tanto del conjunto inicialmente seleccionado, como de las 10 variables con una mayor correlación respecto a la variable seleccionada como piloto (temperatura).

El cuarto objetivo específico que es, comparar los diferentes algoritmos que permitan la integración de los datos a través de inteligencia artificial, fue abordado de forma significativa, pudiendo realizar las exploraciones entre diferentes algoritmos de inteligencia artificial, en primer lugar se realizó una comparativa entre los modelos que permitieran generar la predicción de un solo paso para una salida, partiendo en el diseño de una línea base, pasando por un modelo lineal, seguidamente un modelo denso, hasta probar con redes neuronales convolucionales y luego con las recurrentes LSTM. Se procedió a una evaluación de los modelos de un solo paso y múltiples salidas, para poder llevar a cabo esta comparación, se parte por una línea base y luego se incorpora una red neuronal densa, se prueba con una red neuronal recurrente LSTM para finalmente incluir una red neuronal recurrente LSTM residual.

Por último se realizó una comparativa de los modelos de varios pasos para una salida, con los cuales se partía visualizando una línea base de 24 horas antes de conseguir la predicción de las siguientes 24 horas, pasando por los modelos lineales para finalmente probar con la red densa, la red convolucional, una red recurrente LSTM y una red neuronal recurrente con auto regresión.

El quinto objetivo específico que es, medir la interacción efectiva, entre los modelos existentes con el algoritmo de inteligencia artificial. Este objetivo fue abordado mediante la interacción de los datos obtenidos por los modelos, así como los resultados obtenidos en los diferentes experimentos obteniendo métricas, que permiten dar una buena interacción entre los datos de los modelos junto con la inteligencia artificial para potenciar, los modelos oceanográficos ya existentes.

El sexto objetivo específico; Indagar de que forma puede interactuar la información con los centros de cultivo en miras a la predicción de impactos ambientales, fue abordado

esencialmente en el uso de la determinación de la predicción de los datos ambientales oceanográficos, aprovechando las técnicas de inteligencia artificial, generando una buena precisión en la determinación de la salida de un dato, para el experimento se tomo la temperatura, así como la determinación de un paso para múltiples salidas. De igual forma con la predicción de una salida ante múltiples saltos de tiempo, llevando a generar información valiosa que puede ser aprovechada en los centros de cultivo, tanto en la operatividad como en la disminución de efectos adversos en medio ambiente pudiendo este sistema integrarse a un sistema de monitoreo en línea que pueda aportar información valiosa para el centro de cultivo acuícola.

El séptimo objetivo específico; Determinar el grado de precisión de los algoritmos ante las variables seleccionadas. Fue llevado a cabalidad debido a que se utilizaron un conjunto de validación y un conjunto a parte de prueba, utilizando como métrica de precisión el *mean absolute error* con los diferentes modelos evaluados, tanto para la determinación de un salto para una variable, un salto para múltiples variables como los múltiples saltos para las diferentes variables.

## 7.2. Líneas de trabajo futuro

A continuación, se presentan las futuras líneas de trabajo que se proponen en base al término del TFM:

Al obtener los diferentes modelos explorados para la determinación de las predicciones, tanto de un salto de tiempo en una salida como las predicciones de múltiples salidas, en las variables ambientales oceanográficas, a su vez la determinación de una salida para múltiples pasos. Sería interesante poder explorar la determinación de varios pasos para múltiples salidas, pudiendo generar una configuración de modelos que pueden desarrollar un mayor aporte de información en el centro de cultivo acuícola.

Se recomienda poder probar el diseño con otras grillas en diferentes sectores y así realizar réplicas reduciendo la probabilidad de generar errores, se recomienda no solo obtener los datos de campaña de invierno y verano, sino poder tener información de todo el año de medición, así como todo el ciclo productivo de salmonidos que por lo general son alrededor de 22 a 24 meses.

Un posible trabajo futuro, puede integrar un nuevo diseño de experimentos, directamente en el centro de cultivo, alimentándose no solo con los datos históricos previos, sino con la

alimentación en tiempo real de las variables y esto permita ir ajustando las salidas de los modelos ante cualquier interacción validando con los datos in situ.

Finalmente se recomienda aumentar la exploración del sexto objetivo específico, ya que se podría utilizar una mayor cantidad de información de los centros de cultivo en miras a predicción de impactos ambientales y de este modo robustecer aún más los modelos que se exploraron en esta investigación.

## 8. Bibliografía

¿Qué es Acuicultura? - SUBPESCA. *Orientacion.* (n.d.). Retrieved December 7, 2020, from <http://www.subpesca.cl/orientacion/604/w3-article-80537.html>

¿Revolución Azul? *El impacto regional de la acuicultura del salmón en Chile.* (n.d.). Retrieved February 16, 2021, from <https://repositorio.uc.cl/xmlui/bitstream/handle/11534/3622/000157853.pdf>

Albelwi, S., & Mahmood, A. (2017). A Framework for Designing the Architectures of Deep Convolutional Neural Networks. *Entropy*, 19(6), 242. <https://doi.org/10.3390/e19060242>

Antonucci, F., & Costa, C. (2020). Precision aquaculture: a short review on engineering innovations. In *Aquaculture International* (Vol. 28, Issue 1, pp. 41–57). Springer. <https://doi.org/10.1007/s10499-019-00443-w>

*Association of artificial intelligence, machine learning and deep... | Download Scientific Diagram.* (n.d.). Retrieved February 17, 2021, from [https://www.researchgate.net/figure/Association-of-artificial-intelligence-machine-learning-and-deep-learning-Artificial\\_fig3\\_331608031](https://www.researchgate.net/figure/Association-of-artificial-intelligence-machine-learning-and-deep-learning-Artificial_fig3_331608031)

Basogain Olabe, X. (2005). Redes Neuronales Artificiales Y Sus Aplicaciones. *Medicina Intensiva*, 29(1), 13–20. <http://linkinghub.elsevier.com/retrieve/pii/S021056910574198X>

Bermúdez Soto, J. (2007). *307 Política y regulación ambiental de la acuicultura chilena PolíTCa y RegUlaClón amblental De la aCUICUlURa Chllena [Chilean aquaculture environmental Policy and Regulation].*

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. <https://doi.org/10.1029/2018MS001472>

Bonet Cruz, I., Salazar Martínez, S., Rodríguez Abed, A., Grau Ábalos, R., & García Lorenzo, M. M. (2007). *Redes neuronales recurrentes para el análisis de secuencias.*

Carlos Julián Torres González, I., & Martín Flores Albino José Juan Carbajal Hernández, J. (2017). *Modelo Computacional para la Estimación de Oxígeno Disuelto en Estanques de Producción Acuícola Empleando Redes Neuronales Artificiales.* Universidad Autónoma del Estado de México. <http://ri.uaemex.mx/handle/20.500.11799/65613>

Chrispin, L. C., V, J. V, & Daney Angela, A. S. (2020). Application of Artificial Intelligence in Fisheries and Aquaculture. In *Research Today* (Vol. 2, Issue 6).  
[www.bioticinternational.com](http://www.bioticinternational.com)

Corchado Rodríguez, J. M., & Aiken, J. (2002). *Hybrid artificial intelligence methods in oceanographic forecast models*. Institute of Electrical & Electronics Engineers (IEEE).  
<https://gredos.usal.es/handle/10366/134466>

Croco – Coastal and Regional Ocean COmmunity model. (n.d.). Retrieved October 8, 2020, from <https://www.croco-ocean.org/>

De Lucio, J. (n.d.). Estimación adelantada del crecimiento regional mediante redes neuronales LSTM. *Investigaciones Regionales-Journal of Regional Research*, 2021(1).  
<https://doi.org/10.38191/iirr-jorr.21.007>

Decreto-320 14-DIC-2001 MINISTERIO DE ECONOMÍA, FOMENTO Y RECONSTRUCCIÓN, SUBSECRETARIA DE PESCA - Ley Chile - Biblioteca del Congreso Nacional. (n.d.). Retrieved December 8, 2020, from <https://www.bcn.cl/leychile/navegar?idNorma=192512>

Ducournau, A., & Fablet, R. (2017, February 28). Deep learning for ocean remote sensing: An application of convolutional neural networks for super-resolution on satellite-derived SST data. *2016 9th IAPR Workshop on Pattern Recognition in Remote Sensing, PRRS 2016*. <https://doi.org/10.1109/PRRS.2016.7867019>

Dupont, C., Cousin, P., & Dupont, S. (2018, November 13). IoT for aquaculture 4.0 smart and easy-to-deploy real-time water monitoring with IoT. *2018 Global Internet of Things Summit, GIoTS 2018*. <https://doi.org/10.1109/GIOTS.2018.8534581>

Encinas, C., Ruiz, E., Cortez, J., & Espinoza, A. (2017, June 7). Design and implementation of a distributed IoT system for the monitoring of water quality in aquaculture. *Wireless Telecommunications Symposium*. <https://doi.org/10.1109/WTS.2017.7943540>

Espinosa Plascencia, A., & Del Carmen Bermúdez Almada, M. (n.d.). *La acuicultura y su impacto al medio ambiente Aquaculture and environmental impact Dirección para correspondencia: cbermudez@ciad.mx (María del Carmen Bermúdez)*.

Eze, E., & Ajmal, T. (2020). *Dissolved Oxygen Forecasting in Aquaculture: A Hybrid Model Approach*. <https://doi.org/10.20944/preprints202009.0518.v1>

FAO Fisheries & Aquaculture - Acuicultura. (n.d.). Retrieved February 15, 2021, from  
<http://www.fao.org/fishery/aquaculture/es>

Føre, M., Frank, K., Norton, T., Svendsen, E., Alfredsen, J. A., Dempster, T., Eguiraun, H., Watson, W., Stahl, A., Sunde, L. M., Schellewald, C., Skøien, K. R., Alver, M. O., & Berckmans, D. (2018). Precision fish farming: A new framework to improve production in aquaculture. In *Biosystems Engineering* (Vol. 173, pp. 176–193). Academic Press.  
<https://doi.org/10.1016/j.biosystemseng.2017.10.014>

Fuentes Olmos, J. (2014). Evolución del régimen ambiental de la acuicultura en Chile. *Revista de Derecho (Valparaíso)*, 42, 441–477. <https://doi.org/10.4067/s0718-68512014000100013>

Hermans, M., & Schrauwen, B. (2013). Training and analyzing deep recurrent neural networks. *Advances in Neural Information Processing Systems*.

Hernández, J. J. C., Fernández, L. P. S., Bautista, I. H., & López, J. H. (2017). Modelo basado en redes neuronales artificiales para la evaluación de la calidad del agua en sistemas de cultivo extensivo de camarón. *Tecnología y Ciencias Del Agua*, 8(5), 71–89. <https://doi.org/10.24850/j-tyca-2017-05-05>

HLPE-CSA-FAO. (2015). Contribución del agua a la seguridad alimentaria y la nutrición. *Contribución Del Agua a La Seguridad Alimentaria y La Nutrición*, 155. [www.fao.org/](http://www.fao.org/)

Hu, Z., Li, R., Xia, X., Yu, C., Fan, X., & Zhao, Y. (2020). A method overview in smart aquaculture. In *Environmental Monitoring and Assessment* (Vol. 192, Issue 8, pp. 1–25). Springer. <https://doi.org/10.1007/s10661-020-08409-9>

*Inspiración biológica de las redes neuronales artificiales | by Mario Campos Soberanis | SoldAI | Medium.* (n.d.). Retrieved February 16, 2021, from  
<https://medium.com/soldai/inspiraci%C3%B3n-biol%C3%B3gica-de-las-redes-neuronales-artificiales-9af7d7b906a>

*IoT en acuicultura y seguridad alimentaria | Telcel Empresas.* (n.d.). Retrieved February 15, 2021, from <https://www.telcel.com/empresas/tendencias/notas/acuicultura-y-seguridad-alimentaria-con-iot>

Irsoy, O., & Cardie, C. (2014). Opinion mining with deep recurrent neural networks. *EMNLP 2014 - 2014 Conference on Empirical Methods in Natural Language Processing*,

*Proceedings of the Conference*, 720–728. <https://doi.org/10.3115/v1/d14-1080>

Jhan, G., & Arteaga, P. R. (n.d.). *APLICACIÓN DEL APRENDIZAJE PROFUNDO (“DEEP LEARNING”) AL PROCESAMIENTO DE SEÑALES DIGITALES UNIVERSIDAD AUTÓNOMA DE OCCIDENTE FACULTAD DE INGENIERÍA DEPARTAMENTO DE AUTOMÁTICA Y ELECTRÓNICA PROGRAMA DE INGENIERÍA MECATRÓNICA SANTIAGO DE CALI 2015.*

Jiang, G. Q., Xu, J., & Wei, J. (2018). A Deep Learning Algorithm of Neural Network for the Parameterization of Typhoon-Ocean Feedback in Typhoon Forecast Models. *Geophysical Research Letters*, 45(8), 3706–3716.  
<https://doi.org/10.1002/2018GL077004>

José de Jesús, R., Hernández-Aguilar José, A., Ávila-Camacho Francisco, J., Stein-Carrillo Juan, M., & Meléndez-Ramírez, A. (2016). Sistema sensor para el monitoreo ambiental basado en redes Neuronales. *Ingeniería, Investigación y Tecnología*, 17(2), 211–222.  
<https://doi.org/10.1016/j.riit.2016.06.006>

Khan, A., Sohail, A., Zahoor, U., & Qureshi, A. S. (2020). A survey of the recent architectures of deep convolutional neural networks. *Artificial Intelligence Review*, 53(8), 5455–5516. <https://doi.org/10.1007/s10462-020-09825-6>

*La historia de la Acuicultura y el Salmón en Chile*. (n.d.). Retrieved December 8, 2020, from <https://aquafeed.co/entrada/la-historia-de-la-acuicultura-y-el-salm-n-en-chile-20239/>

Lara Rosano, F. (2017). Fundamentos De Redes Neuronales Artificiales. *Laboratorio de Cibernetica Aplicada Centro de Instrumentos UNAM*, 1, 1–37.  
[http://conceptos.sociales.unam.mx/conceptos\\_final/598trabajo.pdf](http://conceptos.sociales.unam.mx/conceptos_final/598trabajo.pdf)

Lee, P. G. (2000). Process control and artificial intelligence software for aquaculture. *Aquacultural Engineering*, 23(1–3), 13–36. [https://doi.org/10.1016/S0144-8609\(00\)00044-3](https://doi.org/10.1016/S0144-8609(00)00044-3)

Liu, M., Shi, J., Li, Z., Li, C., Zhu, J., & Liu, S. (2017). Towards Better Analysis of Deep Convolutional Neural Networks. *IEEE Transactions on Visualization and Computer Graphics*, 23(1), 91–100. <https://doi.org/10.1109/TVCG.2016.2598831>

Liu, S., Tai, H., Ding, Q., Li, D., Xu, L., & Wei, Y. (2013). A hybrid approach of support vector regression with genetic algorithm optimization for aquaculture water quality prediction.

- Mathematical and Computer Modelling*, 58(3–4), 458–465.  
<https://doi.org/10.1016/j.mcm.2011.11.021>
- Luna, M., & Manuel. (2023). *Aplicaciones de ciencia de datos al proceso de toma de decisiones en la gestión de empresas de acuicultura*.  
<https://repositorio.unican.es/xmlui/handle/10902/19221>
- M.Delgado. (n.d.). *La inteligencia artificial Realidad de un mito moderno*. Retrieved February 16, 2021, from <http://hera.ugr.es/honoriscausa/16912512.pdf>
- Marín Diazaraque, J. M. (2007). Introducción a las redes neuronales aplicadas. *Manual Data Mining*, 1–31. <http://www.redicces.org.sv/jspui/bitstream/10972/2158/1/4. Introduccion a redes neuronales artificiales.pdf>
- Montesdeoca Santana, B. (2016). *Estudios de predicción en series temporales de datos meteorológicos utilizando redes neuronales recurrentes*. 1–50.  
<https://accedacris.ulpgc.es/jspui/handle/10553/18829>
- Muñiz, A. G. (2018). *Aplicaciones de Técnicas de Inteligencia Artificial Basadas en Aprendizaje Profundo (deep learning) al Análisis y Mejora de la Eficiencia de Procesos Industriales*. February 2018, 75. <https://digibuo.uniovi.es/dspace/handle/10651/45097>
- Nadh, K. (2020). *netCDF4 API documentation*. NetCDF4 API Documentation.  
<https://unidata.github.io/netcdf4-python/netCDF4/index.html>
- Pascanu, R., Gulcehre, C., Cho, K., & Bengio, Y. (2014). How to construct deep recurrent neural networks. *2nd International Conference on Learning Representations, ICLR 2014 - Conference Track Proceedings*.
- Patrikalakis, N. M., McCarthy, J. J., Robinson, A. R., Schmidt, H., Evangelinos, C., Haley, P. J., Lalis, S., Lermusiaux, P. F. J., Tian, R., Leslie, W. G., & Cho, W. (2004). *TOWARDS A DYNAMIC DATA DRIVEN SYSTEM FOR RAPID ADAPTIVE INTERDISCIPLINARY OCEAN FORECASTING*.
- Peñafiel, C., & Ávila, R. (2007). Inteligencia Artificial: Métodos, técnicas y aplicaciones. In *Inteligencia Artificial* (Vol. 2, Issue 6).  
<http://scholar.google.com/scholar?hl=en&btnG=Search&q=intitle:Inteligencia+Artificial:+un+enfoque+moderno#0>

- Ramírez Alcocer, U. M., Tello Leal, E., & Ríos Alvarado, A. B. (2018). MODELO BASADO EN REDES NEURONALES RECURRENTES LSTM PARA LA PREDICCIÓN DE LA SIGUIENTE ACTIVIDAD EN PROCESOS DE NEGOCIO. *Tecnológico Nacional de México En Celaya Pistas Educativas*, 40(130), 962–974.  
<http://itcelaya.edu.mx/ojs/index.php/pistas>
- RAMOS, J. (2018). SISTEMA DE VISIÓN ARTIFICIAL PARA EL CONTEO Y MEDICIÓN DE ALEVINOS DE TRUCHA “ARCOÍRIS”, PARA LA DIRECCIÓN SUBREGIONAL DE LA PRODUCCIÓN ANDAHUAYLA. *Agroindustrial Science*, 2(2), 126–131.  
<http://20.20.9.10/handle/123456789/431>

*Redes Neuronales Artificiales | Tablada | Revista de educación matemática*. (n.d.). Retrieved February 16, 2021, from  
<https://revistas.psi.unc.edu.ar/index.php/REM/article/view/10280/10979>

*Regional Ocean Modeling System (ROMS)*. (2021). <https://www.myroms.org/>

Regueiro, C. V. (2018). Modelos Básicos de Redes Neuronales Artificiales. *Journal of Materials Processing Technology*, 1(1), 1–8.  
<https://minerva.usc.es/xmlui/bitstream/handle/10347/12132/8cc86neuronal.pdf?sequence=1&isAllowed=y>

Rollins, J. B. (n.d.). *Metodología Fundamental para la Ciencia de Datos*. Retrieved October 7, 2020, from <https://www.ibm.com/downloads/cas/6RZMKDN8>

Rouhiainen, L. (2018). *INTELIGENCIA ARTIFICIAL 101 COSAS QUE DEBES SABER HOY SOBRE NUESTRO FUTURO INTELIGENCIA ARTIFICIAL*. www.planetadelibros.com

SAE Chile. (2018). *SEA Chile. SEA*. <https://www.sea.gob.cl/sea/declaracion-estudio-impacto-ambiental>

Sagi, T., Lehahn, Y., & Bar, K. (2020). Artificial intelligence for ocean science data integration: Current state, gaps, and way forward. *Elementa*, 8(1).  
<https://doi.org/10.1525/ELEMENTA.418>

Sainath, T. N., Mohamed, A. R., Kingsbury, B., & Ramabhadran, B. (2013). Deep convolutional neural networks for LVCSR. *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, 8614–8618.  
<https://doi.org/10.1109/ICASSP.2013.6639347>

Salas, R. (n.d.). *Redes Neuronales Artificiales-Rodrigo Salas.*

Scher, S. (2018). Toward Data-Driven Weather and Climate Forecasting: Approximating a Simple General Circulation Model With Deep Learning. *Geophysical Research Letters*, 45(22), 12,616-12,622. <https://doi.org/10.1029/2018GL080704>

*Solicitud de concesión de acuicultura - SUBPESCA Tramites.* (n.d.). Retrieved February 15, 2021, from <https://www.subpesca.cl/portal/619/w3-article-12182.html>

Vázquez, J., & Constable, L. (2019). Redes Neurales Artificiales con Aprendizaje Profundo. *Wicc*, 5. <http://sedici.unlp.edu.ar/handle/10915/76941>

*View of Ethanol Fuel Demand Forecasting in Brazil Using a LSTM Recurrent Neural Network Approach.* (n.d.). Retrieved February 18, 2021, from <https://latamt.ieeer9.org/index.php/transactions/article/view/3713>

*Weather Research and Forecasting Model (WRF) | NCAR Research Applications Laboratory | RAL.* (n.d.). Retrieved December 10, 2020, from <https://ral.ucar.edu/solutions/products/weather-research-and-forecasting-model-wrf>

Werbos, P. J. (1990). Backpropagation Through Time: What It Does and How to Do It. *Proceedings of the IEEE*, 78(10), 1550–1560. <https://doi.org/10.1109/5.58337>

Yang, J. B., Nguyen, M. N., San, P. P., Li, X. L., & Krishnaswamy, S. (2015). Deep convolutional neural networks on multichannel time series for human activity recognition. *IJCAI International Joint Conference on Artificial Intelligence, 2015-Janua*, 3995–4001.

Yang, X., Zhang, S., Liu, J., Gao, Q., Dong, S., & Zhou, C. (2020). Deep learning for smart fish farming: applications, opportunities and challenges. *Reviews in Aquaculture*. <https://doi.org/10.1111/raq.12464>

Zoraida Díaz Burgos Bajo la dirección del Doctor, C., García García, E., & Vida Y Satisfaccion Laboral, E. DE. (2005). *MEMORIA PRESENTADA PARA OPTAR AL GRADO DE DOCTOR POR.*

## Anexos

# Prototipo experimental de IA para modelos biogeoquímicos e Hidrodinámicos oceanográficos en centros de cultivo para predicción de variables e impacto ambiental.



Giocrisrai Godoy Bonillo

Universidad Internacional de la Rioja, Logroño (España)

Fecha 03/03/2021

## RESUMEN

Este trabajo desarrolla la interacción de algoritmos de inteligencia artificial con modelos oceanográficos. Para ello se utilizó un piloto experimental, para la predicción de las series temporales de diferentes variables que pueden generar impactos ambientales en el ecosistema marítimo. Fue necesario apoyarse en el estudio de línea base oceanográfico, para obtener datos al sur de Chile y generar el estudio en una concesión marítima. El procesamiento de los datos provenientes del modelo Oceanográfico CROCO-OCEAN, se llevó a cabo mediante una transformación de las series temporales provenientes del formato NetCDF a CSV, permitiendo una mejor manipulación de los datos. Los resultados obtenidos muestran una comparación del rendimiento de diferentes redes neuronales: densas (DNN), convolucionales (CNN) y recurrentes (RNN), lo que permitió determinar cuál método obtiene mejores métricas en problemáticas de series temporales, con la finalidad de poder monitorear y predecir las variables que afectan en el centro de cultivo.

## I. INTRODUCCIÓN

Para dar inicio a la investigación, se debe señalar que la acuicultura es una actividad económica tanto a nivel mundial como para el ámbito Chileno, ya que Chile es el segundo país exportador y productor de salmón a nivel mundial detrás de Noruega [1]. Siguiendo en el ámbito chileno, el Estado se encuentra otorgando concesiones a empresas para la explotación del recurso natural, pero cabe destacar que parte de estos espacios son capaces de generar un alto grado de contaminación, dado a la cantidad de desechos de alimentos no consumidos y otros factores ambientales que hace que dichos residuos se depositen en el fondo marino [2].

Por lo anteriormente indicado, se busca dar una posible respuesta a la necesidad de la posibilidad de otorgar concesiones evitando impactos que eviten la continuidad operativa de los centros de cultivos, por lo que se plantea un nuevo prototipo de inteligencia artificial, el cual se apoyará de utilización de recursos tecnológicos actuales, es decir, se pueden considerar datos oceanográficos del sector con las características de los modelos ya existentes, trabajando con softwares de modelación CROCO-OCEAN (Coastal and Regional Ocean Community Model), que es una actualización del modelo ROMS (Regional Ocean Modeling System), ya que estos softwares se alimentan de información meteorológicas y oceanográficas del sector de estudio [3].

Se desea aprovechar la información existente de estos modelos para luego poder integrarla en un sistema de inteligencia artificial que lleve a generar un sistema avanzado de acuicultura sostenible. Para realizar el experimento, se utilizarán las salidas de información que proporcionan los modelos, ya que permitirá incentivar a la instalación de dispositivos IoT que permita recolectar una mayor cantidad de información que alimente a los modelos, de este modo entrar en sintonía con el medio ambiente

y de igual manera una mejor rentabilidad para el negocio y la producción del mismo [4].

Al ser un prototipo experimental se tomarán los datos de la salida de información de los modelos previamente nombrados, tomando los datos de un sector ubicado al sur de Chile, en el cual se suministraron datos de la campaña de verano e invierno, con su correspondiente grilla geográfica.

Para llevar a cabo el experimento se realizarán pruebas con diferentes algoritmos de aprendizaje profundo, tales como; redes neuronales densas, redes convolucionales, redes recurrentes, etc. Dado que se pueda generar una comparación y presentar cuáles han dado un mejor rendimiento en la resolución de la problemática, que tiene como desafío el poder determinar la predicción de series temporales de las variables oceanográficas que afecta ambientalmente como la producción del centro de cultivo.

## II. ESTADO DEL ARTE

### 2.1 Acuicultura chilena

La industria del salmón chilena ha conseguido situarse como grandes exportadores y productores a nivel mundial, siguiendo a Noruega [5]. Debido al aumento de la producción acuícola chilena, se encuentra en un proceso de transformación tecnológico como regulatorio [6] [7]. Sin embargo, los ambientalistas y habitantes están en conflicto, ya que existen algunas consecuencias negativas en el medio ambiente debido a su producción, las que ocasionan altos grados de contaminación en la vida marina [8].

### 2.2 Desafíos medioambientales en la acuicultura

La producción acuícola va en un aumento, lo que genera que se planteen soluciones ante los desafíos biológicos, económicos y el cuidado con el medioambiente, por tal motivo es importante aprovechar los sistemas automatizados emergentes que ayuden en

el monitoreo de la biomasa, alimentación y de las gestiones de la operación de los centros de cultivo [9] [10].

El surgimiento de las tecnologías del aprendizaje profundo ha generado grandes resultados favorables en diferentes campos en los que se incluye la acuicultura, creando una serie de desafíos en el procesamiento de datos, clasificación de las especies, comportamiento, alimentación, estimación de biomasa y predicción de calidad de agua [11] [12] [13].

### 2.3 Inteligencia Artificial en la acuicultura

La automatización utilizando inteligencia artificial proveniente del rubro agrícola, puede ayudar en el control de la calidad de agua, procesamiento y alimentación [14].

Los sistemas de acuicultura nuevos permiten tener en cuenta las condiciones ambientales y las variables de entrada del sistema tales como; el agua, el oxígeno, la temperatura, velocidad alimentación, pH y otros factores que a través de inteligencia artificial podrán generar eficiencia del proceso [15].

En el centro de cultivo uno de los factores relevantes es la calidad del agua, en el cual se demuestra que el uso de las redes neuronales es una buena opción para la evaluación y detección de estados óptimos o no deseados para un buen manejo de agua en los centros de cultivos de camarones utilizando un modelo computacional capaz de ver la relación dinámica del ecosistema en el cultivo de las especies [16] [17].

### 2.4 IoT (Internet of Things) en la acuicultura

El proceso de integración de la acuicultura y el IoT es algo que se encuentra en una etapa inicial, pero al poder mezclarlos es fácil de implementar, siendo asequible, esto permite el procesamiento de los datos a través de inteligencia artificial, enfrentando desafíos como ser eficiente con el medio ambiente, cuyo artículo se sienta en las bases de la acuicultura 4.0 en los proyectos de investigación europeos [18] [19].

Las innovaciones de la ingeniería acuícola de precisión, lleva a que se incorporen elementos tales como; la robótica, sensores del internet de las cosas en conjunto de sistemas que apoyen en la toma de decisiones como lo hace la inteligencia artificial [20].

### 2.5 Sistemas híbridos en la acuicultura.

Es necesario poder comprender los sistemas híbridos y la integración de modelos a través de inteligencia artificial, ya que esto nos permitirá saber como a través de la información podemos generar sistemas predictivos en ambientes oceanográficos en tiempo real [21] Por lo tanto, es fundamental comprender estos sistemas de acuerdo con diferentes variables, debido a que son totalmente complejos e involucran el funcionamiento de los espacios para la acuicultura.

Dentro de los métodos híbridos de inteligencia artificial en modelos de predicción oceanográfica, se puede pronosticar los valores de los parámetros y comportamientos característicos en tiempo real del océano, siendo este un ambiente dinámico y complejo [22] [23] [24].

Los parámetros importantes para determinar la calidad del agua se encuentran en la concentración de sólido disuelto [25]. Hacer un pronóstico a corto plazo utilizando una serie de tiempo de los sistemas acuícolas, mediante un método híbrido de redes de neuronas basadas en la descomposición empírica, mejorando la calidad de los datos en los cuales se adoptaron índices de medición de los resultados para pronosticar los valores de oxígeno disuelto [26].

### 2.6 Inteligencia artificial

La inteligencia artificial se puede definir como la habilidad de los ordenadores para hacer actividades que normalmente requieren inteligencia humana [27]. Siendo más específico se puede decir que la inteligencia artificial, es la capacidad en las

máquinas para usar algoritmos que permiten aprender de los datos y tomar decisiones [28].

La inteligencia artificial se puede visualizar tanto como ciencia y como ingeniería, ya que se intenta mezclar para darle sentido en ambos puntos. Desde la ciencia pueden ser vistas las intenciones, la semántica, emociones; y desde el punto de vista más computacional o ingenieril; modelos, inferencias, operadores lógicos [29].

### 2.7 Redes neuronales artificiales.

Las redes neuronales artificiales nos ofrecen los medios para modelar de manera efectiva y eficiente problemas grandes y complejos. Son capaces de encontrar relaciones (patrones) de forma inductiva por medio de los algoritmos de aprendizaje basado en los datos existentes [30] [31] [32].

Una de las características fundamentales de las redes neuronales es su adaptabilidad y su susceptibilidad de aprendizaje, a través de la modificación de los pesos de las interconexiones entre las diferentes neuronas [33] [34].

El algoritmo de backpropagation o retropropagación a través del tiempo es una herramienta muy útil en el reconocimiento de patrones, modelización dinámica y control de sistemas en el tiempo [35].

### 2.8 Aprendizaje profundo (Deep Learning)

Las observaciones oceanográficas están limitadas por las tasas de muestreo, mientras que los modelos oceánicos están limitados por una resolución finita con altos coeficientes de viscosidad y difusión. Por lo que con el uso de los algoritmos de aprendizaje profundo son capaces de replicar con éxito la variabilidad espaciotemporal. Poder generar predicciones con varios pasos de tiempo en modelos climáticos, son programas informáticos muy complejos que representan la física de la atmósfera, que a través de las redes neuronales profundas abre nuevas posibilidades para el modelado climático como el oceanográfico[36][37][38][39].

### 2.9 Red Neuronal convolucionales CNN

Las redes neuronales convolucionales han logrado un rendimiento en muchas tareas de reconocimiento de patrones, sin embargo, el desarrollo de los modelos profundos de alta calidad generalmente se basa en una cantidad sustancial de prueba y error, ya que todavía no hay una comprensión clara de cuando y porque funciona un modelo profundo [40] [41].

Los modelos de redes neuronales convolucionales tienen una estructura estándar que consta de capas convolucionales alternas y capas agrupadas, las últimas capas son unas pequeñas cantidades de neuronas completamente conectadas y la capa final si es un clasificador lleva una función de activación softmax. La capa convolucional esta compuesta por un conjunto de kernels o filtros que se pueden aprender y que tiene como objetivo extraer características locales de la entrada. En donde cada kernel se utiliza para calcular un mapa de características [42] [43].

### 2.10 Red Neuronal recurrente RNN

Las redes neuronales recurrentes (RNN) son modelos conexiónistas de datos secuenciales que son naturalmente aplicables en el análisis del proceso de lenguaje natural [44]. Se han convertido recientemente en una popular elección para modelar secuencias de longitud variable, ya que pueden expresarse como una composición de múltiples capas no lineales cuando se despliegan en el tiempo, pudiendo ser utilizadas en predicción de series temporales [45] [46].

Las redes recurrentes de tipo memoria a corto y largo plazo (LSTM) son capaces de proporcionar una alta precisión en el modelado de secuencia en varios dominios de aplicación. Este tipo de red poseen bucles de retroalimentación compuestos por células de memoria que poseen un sistema de decisiones basado en compuertas, las cuales se encargan de recordar patrones

históricos y olvidar patrones no relevantes y es por esta razón que son capaces de asimilar dependencia a corto y largo plazo atendiendo el problema del gradiente de fuga [47].

### III. OBJETIVOS Y METODOLOGÍA

#### Objetivo general

Desarrollar un prototipo experimental para la predicción de impactos ambientales en un ecosistema marítimo en tiempo real integrando los datos obtenidos de los modelos oceanográficos, hidrodinámicos y biogeoquímicos a través de algoritmos de inteligencia artificial, comparando diferentes modelos tomando el modelo con la mejor métrica.

#### Objetivos específicos

- Explorar los datos obtenidos de los diferentes modelos oceanográficos, hidrodinámicos y biogeoquímicos.
- Analizar las diferentes variables que involucran los diferentes modelos oceanográficos de acuerdo a la información obtenida en el sector piloto.
- Establecer las variables más relevantes para poder realizar el prototipo experimental.
- Comparar diferentes algoritmos que permitan la integración de los datos a través de inteligencia artificial.
- Medir la interacción efectiva de entre los modelos existentes con el algoritmo de inteligencia artificial.
- Indagar de que forma pueden interactuar la información con los centros de cultivo en miras a predicción de impacto ambientales
- Determinar el grado de precisión de los algoritmos ante las variables seleccionadas

### IV. CONTRIBUCIÓN

#### Modelos de un solo paso:

El modelo más simple que se puede construir a partir de este tipo de dato es uno que predice el valor de una sola característica, 1 paso de tiempo (1h) en el futuro, basado solo en las condiciones actuales. Así que se comenzara por construir modelos para predecir el valor de la temperatura 1h en el futuro.

#### Base:

Esta primera tarea consiste en predecir la temperatura 1h en el futuro dado el valor actual de todas las características. Así que se comienza con un modelo que simplemente que devuelva la temperatura actual como pronóstico. Esta sería una línea de base razonable, ya que la temperatura cambia lentamente. Por supuesto, esta línea de base no funcionará tan bien si se realiza una predicción adicional en el futuro.

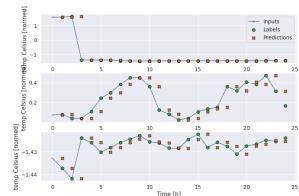


Fig 1 Modelo baseline de un solo paso Fuente: Elaboración propia

En las gráficas mostradas de 3 ejemplos, el modelo de un solo paso se ejecuta en el transcurso de 24 horas. Esto se explica de la siguiente forma:

La línea azul “entradas”, muestra la temperatura de ingreso en cada paso de tiempo. El modelo recibe todas las características, este gráfico solo muestra la temperatura.

Los puntos verdes de “etiquetas”, muestra el valor de predicción que se tiene como objetivo. Estos puntos se muestran en el momento de la predicción, no en el momento de la entrada.

Es por eso, que el rango de etiquetas se desplaza 1 paso con respecto a las nuevas entradas.

Las cruces naranjas de “predicciones”, muestran los pronósticos del modelo para cada paso de tiempo de salida. Si el modelo estuviera prediciendo perfectamente, las predicciones aterrizarían directamente en las “etiquetas”.

#### Modelo lineal:

El modelo entrenable más simple que se puede aplicar a esta tarea, es insertar una transformación lineal entre la entrada y la salida. En este caso, la salida de un paso de tiempo solo depende de ese movimiento.

Al igual que el modelo de *baseline*, el modelo lineal se puede utilizar en lotes de ventanas amplias. Usado de esta manera, el modelo hace un conjunto de predicciones independientes en pasos de tiempo consecutivos. El eje de time actúa como otro eje de batch. No hay interacciones entre las predicciones en cada paso de tiempo.

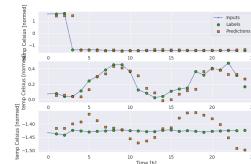


Fig 2 Modelo lineal de un solo paso Fuente: Elaboración propia

#### Denso:

Antes de aplicar modelos que realmente operan en múltiples pasos de tiempo, vale la pena verificar el rendimiento de los modelos de pasos de entrada únicos más profundos y potentes.

Aquí hay un modelo similar al modelo lineal, excepto que apila varias capas Dense entra la entrada y la salida.

#### Denso de varios pasos:

Un modelo de paso de tiempo único no tiene contexto para los valores actuales de sus entradas. Para abordar este problema, el modelo necesita acceso a múltiples pasos de tiempo al hacer predicciones, los modelos baseline, linear y dense manejan cada paso de tiempo de forma independiente. Aquí el modelo tomará múltiples pasos de tiempo como entrada para producir una única salida.

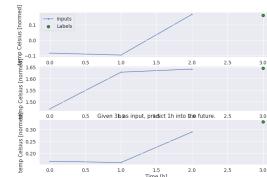


Fig 3 Dado 3h de entrada, predice 1h en el futuro Fuente: Elaboración propia

Se puede entrenar un modelo dense en una ventana de pasos de entrada múltiple agregando layers.Flatten como la primera capa del modelo.

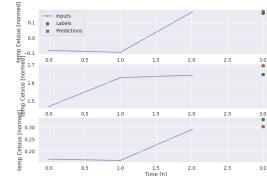


Fig 4 Dado 3h de entrada, predice 1h en el futuro multi-paso denso Fuente: Elaboración propia

#### Red neuronal convolucionales:

Una capa de convolución (layers.Conv1D), también toma varios pasos de tiempo como entrada para cada predicción. A

continuación, se muestra el mismo modelo que `multi_step_dense`, reescrito con una convolución.

Teniendo en cuenta los cambios:

- Las capas `layeres.Flatten` y las primeras capas `layeres.Dense` se reemplazan por capas `layers.Conv1D`.

La diferencia entre `conv_model` y el modelo `multi_step_dense` es que `conv_model` se puede ejecutar en entradas de cualquier longitud. La capa convolucional se le aplica una venta deslizante de entradas, de este modo ver si se ejecuta en una entrada más amplia, produciendo una salida más amplia.

Teniendo en cuenta que la salida es más corta que la entrada. Para que el entrenamiento o el trazado funcionen, se necesita que las etiquetas y la predicción tengan la misma longitud.

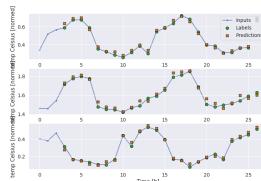


Fig 5 Predicción de una red neuronal convolucionales teniendo en cuenta 3 pasos de tiempo anteriores Fuente: Elaboración propia

#### Red neuronal recurrente:

Los RNN procesan una serie de tiempo paso a paso, manteniendo un estado interno de paso de tiempo a paso de tiempo.

Se usará una capa RNN llamada Long Short Term Memory (LSTM).

Un argumento de constructor importante para todas las capas de keras RNN es el argumento `return_sequences`. Esta configuración puede configurar la capa de una de dos formas.

- Si es `False`, el valor predeterminado, la capa solo devuelve el resultado del paso de tiempo final, lo que le da tiempo al modelo para calentar su estado interno antes de hacer una sola predicción:
- Si es `True` la capa devuelve una salida para cada entrada. Esto es útil para:
  - El apilamiento de capas RNN.
  - Entrenamiento de un modelo en múltiples pasos de tiempo simultáneamente.



Fig 6 Predicción de una red neuronal recurrente 24h de datos a la vez Fuente: Elaboración propia

#### Modelos de múltiples salidas:

Hasta ahora, todos los modelos predijeron una característica de salida única (`temp Celsius`), para un solo paso de tiempo.

Todos estos modelos se pueden convertir para predecir múltiples características simplemente cambiando el número de unidades en la capa de salida y ajustando las ventanas de entrenamiento para incluir todas las características en las etiquetas.

Teniendo en cuenta que el eje de features de las etiquetas ahora tiene la misma profundidad que las entradas, en lugar de 1.

#### Avanzado: conexiones residuales

El modelo de Baseline anterior aprovechó el hecho de que la secuencia no cambia drásticamente de un paso de tiempo a otro. Todos los modelos entrenados hasta ahora se inicializaron aleatoriamente y luego tuvieron que aprender que el resultado es un pequeño cambio con respecto al paso de tiempo anterior.

Es común en el análisis de series de tiempo construir modelos que en lugar de predecir el siguiente valor, predicen cómo cambiará el valor en el siguiente paso de tiempo.

Así es como se aprovecha el conocimiento que el cambio debe ser pequeño. Básicamente, esto inicializa el modelo para que coincida con la *baseline*. Para esta tarea, ayuda a que los modelos converjan más rápido, con un rendimiento ligeramente mejor. Este enfoque se puede utilizar junto con cualquier modelo analizado.

Aquí se está aplicando al modelo LSTM, teniendo en cuenta el uso de `tf.initializers.zeros` para garantizar que los cambios iniciales previstos sean pequeños y no dominen la conexión residual. Aquí no hay preocupaciones de ruptura de simetría para los gradientes, ya que los ceros solo se usan en la última capa.

#### Modelos de varios pasos:

En una predicción de varios pasos, el modelo debe aprender a predecir un rango de valores futuros. Por lo tanto, a diferencia de un modelo de un solo paso, donde solo se predice un único punto futuro, un modelo de varios pasos predice una secuencia de los valores futuros.

Hay dos enfoques aproximados para esto:

- Predicciones de un solo disparo donde se predice toda la serie de tiempo a la vez.
- Predicciones autoregresivas donde el modelo solo hace predicciones de un solo paso y su salida se retroalimenta como su entrada.

Para el modelo de varios pasos, los datos de entrenamiento nuevamente consisten en muestras por hora. Sin embargo, aquí los modelos aprenderán a predecir las 24 horas del futuro, dadas las 24 horas del pasado.

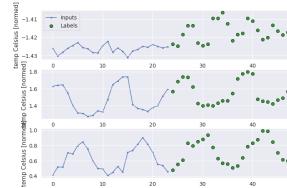


Fig 7 Generación de predicción de múltiples puntos a futuro Fuente: Elaboración propia

#### Líneas base múltiples pasos

Una línea de base simple para esta tarea es repetir el último paso de tiempo de entrada para el número requerido de pasos de tiempo de salida

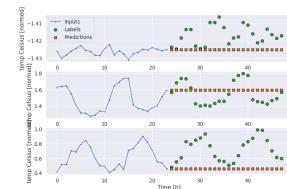


Fig 8 Modelo de línea base last de múltiples pasos Fuente: Elaboración propia

Dado que esta tarea es predecir 24 horas dadas 24 horas, otro enfoque simple es repetir el día anterior, asumiendo que mañana será similar.

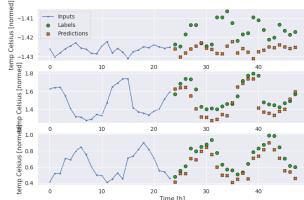


Fig 9 Modelo de línea base repetir el día anterior de múltiples pasos  
Fuente: Elaboración propia

### Modelos de disparo único múltiples pasos

Un enfoque de alto nivel para este problema es utilizar un modelo de "disparo único", donde el modelo hace la predicción de la secuencia completa en un solo paso. Esto se puede implementar de manera eficiente como layers.Dense con OUT\_STEPS\*features unidades de salida. El modelo solo necesita remodelar esa salida a la requerida (OUTPUT\_STEPS, features).

#### Lineal

Un modelo lineal simple basado en el último paso de tiempo de entrada funciona mejor que cualquier línea de base, pero no tiene suficiente potencia. El modelo necesita predecir los pasos de tiempo OUTPUT\_STEPS, a partir de un solo paso de tiempo de entrada con una proyección lineal. Solo puede capturar un segmento de baja dimensión del comportamiento, probablemente basado principalmente en la hora del día y la época del año.

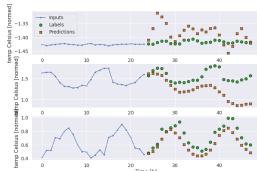


Fig 10 Modelo lineal para múltiples pasos Fuente: Elaboración propia

#### Denso para múltiples pasos

Se agrega un layers.Dense La layers.Dense entre la entrada y la salida le da al modelo lineal más potencia, pero aún se basa en un solo paso de tiempo de entrada.

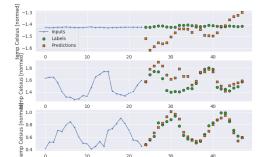


Fig 11 Modelo denso para múltiples pasos Fuente: Elaboración propia

#### Redes neuronales convolucionales CNN para múltiples pasos

Un modelo convolucional hace predicciones basadas en un historial de ancho fijo, lo que puede conducir a un mejor rendimiento que el modelo denso, ya que puede ver cómo cambian las cosas con el tiempo.

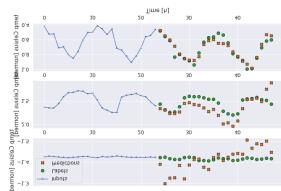


Fig 13 Modelo red neuronal convolucionales para múltiples pasos  
Fuente: Elaboración propia

#### Red neuronal recurrente RNN para múltiples pasos

Un modelo recurrente puede aprender a usar un largo historial de entradas, si es relevante para las predicciones que hace el modelo. Aquí, el modelo acumulará el estado interno durante 24 h, antes de realizar una única predicción para las próximas 24 h.

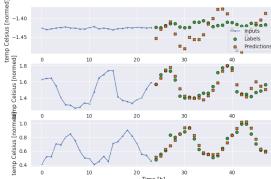


Fig 14 Modelo red neuronal recurrente para múltiples pasos a partir de acumular 24h Fuente: Elaboración propia

#### Avanzado modelo auto regresivo

Todos los modelos anteriores predicen la secuencia de salida completa en un solo paso. En algunos casos, puede ser útil que el modelo descomponga esta predicción en pasos de tiempo individuales. Luego, la salida de cada modelo se puede retroalimentar en cada paso y se pueden hacer predicciones condicionadas al anterior, como en el clásico Generar secuencias con redes neuronales recurrentes.

Una clara ventaja de este estilo de modelo es que se puede configurar para producir resultados con una longitud variable. Retroalimentar la salida de un modelo a su entrada.

#### Red neuronal recurrente auto regresivo

Solo se crea un modelo RNN auto regresivo, pero este patrón se puede aplicar a cualquier modelo que haya sido diseñado para generar un solo paso de tiempo. El modelo tendrá la misma forma básica como las de un solo paso LSTM modelos: Un LSTM seguido por un layers.Dense que convierte los LSTM salidas a las predicciones del modelo.

Un layers.LSTM es un layers.LSTMCell envuelto en las layers.RNN nivel layers.RNN que administra el estado y los resultados de la secuencia por usted. En este caso, el modelo tiene que administrar manualmente las entradas para cada paso, por lo que utiliza layers.LSTMCell directamente para el nivel inferior, interfaz de paso de tiempo único.

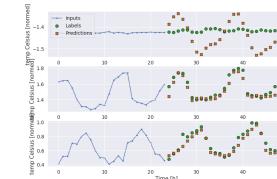


Fig 15 Modelo red neuronal recurrente autoregresivo múltiples pasos  
Fuente: Elaboración propia

## V. RESULTADOS

Para los modelos de una sola salida se puede verificar su actuación utilizando los datos de validación y de prueba de los modelos de una sola salida.

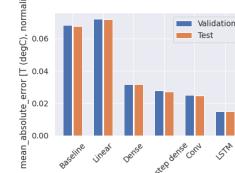


Fig 16 Actuación de modelos de un solo paso una salida Fuente: Elaboración propia

De esta forma se presentan los datos de los rendimientos de los diferentes modelos en la siguiente tabla:

TABLA I

RENDIMIENTO DE LOS MODELOS DE UN SOLO PASO PARA UNA SALIDA

Modelos	Mean Absolute Error (MAE)
Baseline	0.0677
Linear	0.0717
Dense	0.0317
Multi step dense	0.02713
Conv	0.0248
LSTM	0.0148

FUENTE ELABORACIÓN PROPIA.

A continuación, se muestra el rendimiento general de estos modelos de múltiples salidas, para esto se exponen los promedios de todas las salidas en base a los modelos que se probaron en esta ocasión.

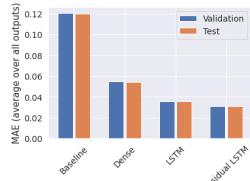


Fig 17 Actuación de modelos de un solo paso para múltiples salidas  
Fuente: Elaboración propia

En la siguiente tabla se muestran los rendimientos de los modelos de un solo paso para múltiples salidas.

TABLA 2

RENDIMIENTO DE LOS MODELOS DE UN SOLO PASO PARA SALIDAS MÚLTIPLES

Modelo	Mean Absolute Error (MAE)
Baseline	0.1200
Dense	0.0543
LSTM	0.0353
Residual LSTM	0.0311

FUENTE ELABORACIÓN PROPIA.

Los rendimientos anteriores se promedian en todos los resultados del modelo.

La actuación de los modelos de múltiples pasos para una salida, muestran los rendimientos claramente decrecientes en función de la complejidad del modelo sobre este problema, por lo cual se presenta la validación y la prueba en la siguiente grafica.

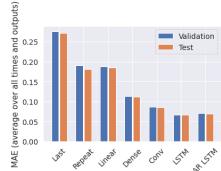


Fig 18 Actuación de modelos de varios pasos para una salida  
Fuente: Elaboración propia

Las métricas para los modelos de múltiples salidas en la primera mitad de este trabajo muestran el rendimiento promedio en todas las funciones de salida. Estos desempeños son similares, pero también promediados en los pasos de tiempo de salida.

TABLA 3

RENDIMIENTO DE LOS MODELOS DE VARIOS PASOS PARA UNA SALIDA

Modelos	Mean Absolute Error (MAE)
Last	0.2715
Repeat	0.1808
Linear	0.1848
Dense	0.1111
Conv	0.0847
LSTM	0.0664
AR LSTM	0.0696

FUENTE ELABORACIÓN PROPIA.

Las ganancias logradas al pasar de un modelo denso a modelos convolucionales y recurrentes son solo un pequeño porcentaje y el modelo autoregresivo tuvo un desempeño ligeramente más

bajo que la red recurrente.

## VI. DISCUSIÓN

Para el experimento se llevaron a cabo 3 procesos principales que internamente se derivaron en otros modelos a evaluar. Esta serie de modelos que se desarrollaron tenían diferentes finalidades; la predicción de un solo paso para una salida, la predicción de un solo paso para múltiples salidas y la predicción de varios pasos para una salida.

En primer lugar, se realizaron una serie de pruebas para llevar a cabo la predicción de un solo paso para una salida. Se generó una línea base para poder tener una referencia de partida, pasando a evaluar un modelo lineal, el cual no mejoró la línea base previa que se había desarrollado.

Se va generando una mejoría cuando ya se empiezan a explorar las redes profundas, es aquí cuando comienza a mejorar el rendimiento del modelo, pasando a una red convolucional donde sigue mejorando el modelo hasta llegar a la red neuronal recurrente LSTM, con la cual se consigue el mejor rendimiento obteniendo, el mean absolute error de 0,0148.

Como segundo experimento se procedió a probar una serie de modelos que permitieron evaluar el comportamiento de las predicciones a un salto de tiempo para múltiples salidas, es decir, se le añade complejidad al experimento anterior. En este caso se inicia probando una red profunda, con la cual empiezan a mejorar las métricas obtenidas por la línea base.

En miras de aprovechar la experiencia de los modelos pasados, se procede a realizar pruebas con una red neural recurrente LSTM, que también mejora las métricas obtenidas.

En este experimento se logró mejorar el resultado con el diseño de una red neuronal recurrente residual LSTM, con la cual se obtuvo una métrica del mean absolute error de 0,0311. Lo que igual determina una mejoría entre los modelos explorados y entrega una perspectiva ante múltiples salidas que puedan aportar en una predicción con más información, la que puede ser aprovechada dentro de los centros acuícolas.

Como último experimento, se probaron diferentes modelos para poder determinar varios pasos para una salida. En este caso se realizaron pruebas como; ver el pasado, repetir los valores, realizar un modelo lineal, hasta ir llegando a los modelos profundos, pasando por una red convolucional, luego se prueba una red neuronal recurrente y una red neuronal recurrente auto regresiva.

Se pudo observar que, con los diferentes comportamientos de los modelos, las redes neuronales recurrentes en sus diferentes variantes ayudaron en las predicciones temporales.

Por lo tanto, si se quiere realizar un análisis dentro de algún centro de cultivo acuícola, estos modelos pueden aportar información valiosa para los procesos productivos, así como también contribuir en la prevención del impacto ambiental dentro del espacio acuícola.

## VII. CONCLUSIONES

El presente trabajo de fin de máster fue direccionado por su objetivo general:

Desarrollar un prototipo experimental para la predicción de impactos ambientales en un ecosistema marítimo en tiempo real integrando los datos obtenidos de los modelos oceanográficos, hidrodinámicos y biogeocíquicos a través de algoritmos de inteligencia artificial, comparando diferentes modelos para verificar cual es la mejor métrica.

El primer objetivo específico se ha cumplido, debido a que se generó un proceso de exploración inicial de los datos, provenientes de la salida de los modelos oceanográficos CROCO-OCEAN, el cual era fundamental para comprender con que datos

se contaba, tanto en unidades de medición, dimensiones, formato, peso de los archivos, como para poder tratar los archivos. Con estos se pudo determinar que los datos que provienen de los modelos oceanográficos están diseñados en NetCDF, un formato que es utilizado para guardar información geográfica, de igual manera fue relevante conocer el peso de los archivos, ya que permitió acomodar los recursos que se necesitaban para poder analizar a profundidad las variables que están dentro de los archivos.

El segundo objetivo específico esta ampliamente cumplido, ya que se pudo analizar inicialmente los archivos de trabajo. Se verificó que se poseía información extensa de todo un sector, por lo que se decidió tomar una grilla para el trabajo. Con este proceso se realizaron los análisis estadísticos de las diferentes variables y verificar que características estuvieron presentes en el conjunto de datos, logrando una mayor comprensión para determinar los modelos que se utilizarán.

El tercer objetivo específico, fue ampliamente abordado, ya que, al poseer las características de las variables de los modelos oceanográficos, se procedió a seleccionar el conjunto de datos con el que se trabajaría para la construcción de los modelos de inteligencia artificial. Se obtienen un dataset en formato csv para un mejor manejo del conjunto de datos, utilizando técnicas descriptivas estadísticas, además de realizar matriz de correlación tanto del conjunto inicialmente seleccionado, como de las 10 variables con una mayor correlación respecto a la variable seleccionada como piloto (temperatura).

El cuarto objetivo específico fue abordado de forma significativa, pudiendo realizar las exploraciones entre diferentes algoritmos de inteligencia artificial, en primer lugar, se realizó una comparativa entre los modelos que permitieran generar la predicción de un solo paso para una salida, partiendo en el diseño de una línea base, pasando por un modelo lineal, seguidamente un modelo denso, hasta probar con redes neuronales convolucionales y luego con las recurrentes LSTM.

El quinto objetivo fue abordado mediante la interacción de los datos obtenidos por los modelos, así como los resultados obtenidos en los diferentes experimentos obteniendo métricas, que permiten dar una buena interacción entre los datos de los modelos junto con la inteligencia artificial para potenciar, los modelos oceanográficos ya existentes.

El sexto objetivo específico, fue abordado esencialmente en el uso de la determinación de la predicción de los datos ambientales oceanográficos, aprovechando las técnicas de inteligencia artificial, generando una buena precisión en la determinación de la salida de un dato, para el experimento se tomo la temperatura, así como la determinación de un paso para múltiples salidas.

El séptimo objetivo específico. Fue llevado a cabalidad debido a que se utilizaron un conjunto de validación y un conjunto a parte de prueba, utilizando como métrica de precisión el mean absolute error con los diferentes modelos evaluados, tanto para la determinación de un salto para una variable, un salto para múltiples variables como los múltiples saltos para las diferentes variables.

## REFERENCIAS

- [1] La historia de la Acuicultura y el Salmón en Chile. (n.d.). Retrieved December 8, 2020, from <https://aquafeed.co/entrada/la-historia-de-la-acuicultura-y-el-salm-n-en-chile-20239/> (Espinosa Plascencia & Del Carmen Bermúdez Almada, 2012.)
- [2] Lee, P. G. (2000). Process control and artificial intelligence software for aquaculture. *Aquacultural Engineering*, 23(1–3), 13–36. [https://doi.org/10.1016/S0144-8609\(00\)00044-3](https://doi.org/10.1016/S0144-8609(00)00044-3) (IoT En Acuicultura y Seguridad Alimentaria | Telcel Empresas, 2021)
- [3] La historia de la Acuicultura y el Salmón en Chile. (n.d.). Retrieved December 8, 2020, from <https://aquafeed.co/entrada/la-historia-de-la-acuicultura-y-el-salm-n-en-chile-20239/> (Bermúdez Soto, 2007)
- [4] Fuentes Olmos, J. (2014). Evolución del régimen ambiental de la acuicultura en Chile. *Revista de Derecho (Valparaíso)*, 42, 441–477. <https://doi.org/10.4067/s0718-68512014000100013> (¿Revolución Azul? El Impacto Regional de La Acuicultura Del Salmón En Chile, 2020)
- [5] Føre, M., Frank, K., Norton, T., Svendsen, E., Alfredsen, J. A., Dempster, T., Eguiraun, H., Watson, W., Stahl, A., Sunde, L. M., Schellewald, C., Skoien, K. R., Alver, M. O., & Berckmans, D. (2018). Precision fish farming: A new framework to improve production in aquaculture. In *Biosystems Engineering* (Vol. 173, pp. 176–193). Academic Press. <https://doi.org/10.1016/j.biosystemseng.2017.10.014>
- [6] Antonucci, F., & Costa, C. (2020). Precision aquaculture: a short review on engineering innovations. In *Aquaculture International* (Vol. 28, Issue 1, pp. 41–57). Springer. <https://doi.org/10.1007/s10499-019-00443-w>
- [7] Yang, X., Zhang, S., Liu, J., Gao, Q., Dong, S., & Zhou, C. (2020). Deep learning for smart fish farming: applications, opportunities and challenges. *Reviews in Aquaculture*. <https://doi.org/10.1111/raq.12464>
- [8] Sagi, T., Lehahn, Y., & Bar, K. (2020). Artificial intelligence for ocean science data integration: Current state, gaps, and way forward. *Elementa*, 8(1). <https://doi.org/10.1525/ELEMENTA.418>
- [9] Hu, Z., Li, R., Xia, X., Yu, C., Fan, X., & Zhao, Y. (2020). A method overview in smart aquaculture. In *Environmental Monitoring and Assessment* (Vol. 192, Issue 8, pp. 1–25). Springer. <https://doi.org/10.1007/s10661-020-08409-9>
- [10] Chrispin, L. C., V, J. V., & Daney Angela, A. S. (2020). Application of Artificial Intelligence in Fisheries and Aquaculture. In *Research Today* (Vol. 2, Issue 6). [www.bioticainternational.com](http://www.bioticainternational.com)
- [11] Lee, P. G. (2000). Process control and artificial intelligence software for aquaculture. *Aquacultural Engineering*, 23(1–3), 13–36. [https://doi.org/10.1016/S0144-8609\(00\)00044-3](https://doi.org/10.1016/S0144-8609(00)00044-3) (Carlos Julián Torres González & Martín Flores Albino José Juan Carbajal Hernández, 2017).
- [12] RAMOS, J. (2018). SISTEMA DE VISIÓN ARTIFICIAL PARA EL CONTEO Y MEDICIÓN DE ALEVINOS DE TRUCHA “ARCOÍRIS”, PARA LA DIRECCIÓN SUBREGIONAL DE LA PRODUCCIÓN ANDAHUAYLA. *Agroindustrial Science*, 2(2), 126–131. <http://20.20.9.10/handle/123456789/431>
- [13] Encinas, C., Ruiz, E., Cortez, J., & Espinoza, A. (2017, June 7). Design and implementation of a distributed IoT system for the monitoring of water quality in aquaculture. *Wireless Telecommunications Symposium*. <https://doi.org/10.1109/WTS.2017.7943540>
- [14] Dupont, C., Cousin, P., & Dupont, S. (2018, November 13). IoT for aquaculture 4.0 smart and easy-to-deploy real-time water monitoring with IoT. *2018 Global Internet of Things Summit, GIOTS 2018*. <https://doi.org/10.1109/GIOTS.2018.8534581>
- [15] Antonucci, F., & Costa, C. (2020). Precision aquaculture: a short review on engineering innovations. In *Aquaculture International* (Vol. 28, Issue 1, pp. 41–57). Springer. <https://doi.org/10.1007/s10499-019-00443-w>
- [16] Corchado Rodríguez, J. M., & Aiken, J. (2002). Hybrid artificial intelligence methods in oceanographic forecast models. *Institute of Electrical & Electronics Engineers (IEEE)*.

- https://gredos.usal.es/handle/10366/134466
- [17] Corchado Rodríguez, J. M., & Aiken, J. (2002). Hybrid artificial intelligence methods in oceanographic forecast models. Institute of Electrical & Electronics Engineers (IEEE). https://gredos.usal.es/handle/10366/134466
- [18] Patrikalakis, N. M., McCarthy, J. J., Robinson, A. R., Schmidt, H., Evangelinos, C., Haley, P. J., Lalis, S., Lemusiaux, P. F. J., Tian, R., Leslie, W. G., & Cho, W. (2004). TOWARDS A DYNAMIC DATA DRIVEN SYSTEM FOR RAPID ADAPTIVE INTERDISCIPLINARY OCEAN FORECASTING.
- [19] Liu, S., Tai, H., Ding, Q., Li, D., Xu, L., & Wei, Y. (2013). A hybrid approach of support vector regression with genetic algorithm optimization for aquaculture water quality prediction. Mathematical and Computer Modelling, 58(3–4), 458–465. https://doi.org/10.1016/j.mcm.2011.11.021
- [20] Eze, E., & Ajmal, T. (2020). Dissolved Oxygen Forecasting in Aquaculture: A Hybrid Model Approach. https://doi.org/10.20944/preprints202009.0518.v1
- [21] José de Jesús, R., Hernández-Aguilar José, A., Ávila-Camacho Francisco, J., Stein-Carrillo Juan, M., & Meléndez-Ramírez, A. (2016). Sistema sensor para el monitoreo ambiental basado en redes Neuronales. Ingeniería, Investigación y Tecnología, 17(2), 211–222. https://doi.org/10.1016/j.riit.2016.06.006
- [22] Rouhiainen, L. (2018). INTELIGENCIA ARTIFICIAL 101 COSAS QUE DEBES SABER HOY SOBRE NUESTRO FUTURO INTELIGENCIA ARTIFICIAL. www.planetadelibros.com
- [23] M.Delgado. (n.d.). La inteligencia artificial Realidad de un mito moderno. Retrieved February 16, 2021, from http://hera.ugr.es/honoriscausa/16912512.pdf
- [24] Peñafiel, C., & Ávila, R. (2007). Inteligencia Artificial: Métodos, técnicas y aplicaciones. In Inteligencia Artificial (Vol. 2, Issue 6). http://scholar.google.com/scholar?hl=en&btnG=Search&q=intitle:Inteligencia+Artificial:+un+enfoque+moderno#0
- [25] Salas, R. (n.d.). Redes Neuronales Artificiales-Rodrigo Salas.
- [26] Basogain Olabe, X. (2005). Redes Neuronales Artificiales Y Sus Aplicaciones. Medicina Intensiva, 29(1), 13–20. http://linkinghub.elsevier.com/retrieve/pii/S021056910574198X( Redes Neuronales Artificiales | Tablada | Revista de Educación Matemática, n.d.)
- [27] Marín Diazaraque, J. M. (2007). Introducción a las redes neuronales aplicadas. Manual Data Mining, 1–31. http://www.redicces.org.sv/jspui/bitstream/10972/2158/1/4. Introduccion a redes neuronales artificiales.pdf
- [28] Lara Rosano, F. (2017). Fundamentos De Redes Neuronales Artificiales. Laboratorio de Cibernetica Aplicada Centro de Instrumentos UNAM, 1, 1–37. http://conceptos.sociales.unam.mx/conceptos\_final/598trabajo.pdf
- [29] Werbos, P. J. (1990). Backpropagation Through Time: What It Does and How to Do It. Proceedings of the IEEE, 78(10), 1550–1560. https://doi.org/10.1109/5.58337
- [30] Ducomteau, A., & Fablet, R. (2017, February 28). Deep learning for ocean remote sensing: An application of convolutional neural networks for super-resolution on satellite-derived SST data. 2016 9th IAPR Workshop on Pattern Recognition in Remote Sensing, PRRS 2016. https://doi.org/10.1109/PRRS.2016.7867019
- [31] Jiang, G. Q., Xu, J., & Wei, J. (2018). A Deep Learning Algorithm of Neural Network for the Parameterization of Typhoon-Ocean Feedback in Typhoon Forecast Models. Geophysical Research Letters, 45(8), 3706–3716. https://doi.org/10.1002/2018GL077004
- [32] 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. https://doi.org/10.1029/2018MS001472
- [33] Scher, S. (2018). Toward Data-Driven Weather and Climate Forecasting: Approximating a Simple General Circulation Model With Deep Learning. Geophysical Research Letters, 45(22), 12,616–12,622. https://doi.org/10.1029/2018GL080704
- [34] Sainath, T. N., Mohamed, A. R., Kingsbury, B., & Ramabhadran, B. (2013). Deep convolutional neural networks for LVCSR. ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings, 8614–8618. https://doi.org/10.1109/ICASSP.2013.6639347
- [35] (M. Liu et al., 2017)
- [36] Albelwi, S., & Mahmood, A. (2017). A Framework for Designing the Architectures of Deep Convolutional Neural Networks. Entropy, 19(6), 242. https://doi.org/10.3390/e1906024
- [37] Yang, J. B., Nguyen, M. N., San, P. P., Li, X. L., & Krishnaswamy, S. (2015). Deep convolutional neural networks on multichannel time series for human activity recognition. IJCAI International Joint Conference on Artificial Intelligence, 2015-Janua, 3995–4001.
- [38] Irsoy, O., & Cardie, C. (2014). Opinion mining with deep recurrent neural networks. EMNLP 2014 - 2014 Conference on Empirical Methods in Natural Language Processing, Proceedings of the Conference, 720–728. https://doi.org/10.3115/v1/d14-1080
- [39] Pascanu, R., Gulcehre, C., Cho, K., & Bengio, Y. (2014). How to construct deep recurrent neural networks. 2nd International Conference on Learning Representations, ICLR 2014 - Conference Track Proceedings.
- [40] Hermans, M., & Schrauwen, B. (2013). Training and analyzing deep recurrent neural networks. Advances in Neural Information Processing Systems.
- [41] Zoraida Díaz Burgos Bajo la dirección del Doctor, C., García García, E., & Vida Y Satisfaccion Laboral, E. DE. (2005). MEMORIA PRESENTADA PARA OPTAR AL GRADO DE DOCTOR POR.
- [42] Ramírez Alcocer, U. M., Tello Leal, E., & Ríos Alvarado, A. B. (2018). MODELO BASADO EN REDES NEURONALES RECURRENTES LSTM PARA LA PREDICCIÓN DE LA SIGUIENTE ACTIVIDAD EN PROCESOS DE NEGOCIO. Tecnológico Nacional de México En Celaya Pistas Educativas, 40(130), 962–974. http://itcelaya.edu.mx/ojs/index.php/pistas
- [43] View of Ethanol Fuel Demand Forecasting in Brazil Using a LSTM Recurrent Neural Network Approach. (n.d.). Retrieved February 18, 2021, from https://latamt.ieeer9.org/index.php/transactions/article/view/3713
- [44] Montesdeoca Santana, B. (2016). Estudios de predicción en series temporales de datos meteorológicos utilizando redes neuronales recurrentes. 1–50. https://accedacris.ulpgc.es/jspui/handle/10553/18829
- [45] Rollins, J. B. (n.d.). Metodología Fundamental para la Ciencia de Datos. Retrieved October 7, 2020, from https://www.ibm.com/downloads/cas/6RZMKDN8