



Universidad de Valladolid

**ESCUELA DE INGENIERÍA INFORMÁTICA
DE SEGOVIA**

**Grado en Ingeniería Informática
de Servicios y Aplicaciones**

**Obtención de patrones meteorológicos
con machine learning**

Alumno: Antía Fernández Marcos

**Tutores: M^a Luisa Martín Pérez
Daniel Santos Muñoz**

Obtención de patrones meteorológicos mediante *Machine Learning*

Antía Fernández Marcos

02-04-2025

Resumen

Los CTs representan una de las amenazas más significativas para las regiones costeras alrededor del mundo, debido a su capacidad de causar daños extensos y poner en riesgo vidas humanas. Este Trabajo de Fin de Grado se enfoca en el desarrollo de un modelo de predicción de CTs, utilizando datos meteorológicos de alta resolución de ERA5, con el fin de mejorar la precisión y anticipación de estos fenómenos naturales.

En esta memoria se revisan las principales técnicas y arquitecturas actuales utilizadas en la predicción meteorológica y el análisis de grandes volúmenes de datos.

Se emplearon métodos de deep learning para desarrollar un modelo predictivo que analiza datos históricos y ofrece predicciones sobre la formación y trayectoria de CTs. La investigación incluye un análisis detallado de los datos, la selección de características relevantes y la implementación de algoritmos avanzados de aprendizaje automático.

Este trabajo pone de manifiesto la importancia de la inteligencia artificial y el aprendizaje automático en la mejora de las predicciones meteorológicas y ofrece una base sólida para el desarrollo de sistemas avanzados de alerta temprana.

Palabras clave: predicción de ciclones, deep learning, ERA5, inteligencia artificial, meteorología, CTs.

Abstract

Tropical cyclones represent one of the most significant threats to coastal regions around the world due to their ability to cause extensive damage and endanger human lives. This Bachelor's Thesis focuses on the development of a tropical cyclone prediction model using high-resolution meteorological data from ERA5, with the aim of improving the accuracy and anticipation of these natural phenomena.

In this thesis, the main techniques and current architectures used in meteorological prediction and the analysis of large volumes of data are reviewed. Deep learning methods were employed to develop a predictive model that analyzes historical data and provides predictions on the formation and trajectory of tropical cyclones. The research includes a detailed analysis of the data, the selection of relevant features, and the implementation of advanced machine learning algorithms.

This work highlights the importance of artificial intelligence and machine learning in improving meteorological predictions and offers a solid foundation for the development of advanced early warning systems.

Keywords: cyclone prediction, deep learning, ERA5, artificial intelligence, meteorology, tropical cyclones.

Índice de imágenes

Imagen 1: Formación de un CT	21
Imagen 2: Imagen térmica del huracán María de categoría 5 en 2017. En amarillo y naranja, aguas oceánicas cálidas, en azul y blanco, las cimas de las nubes del huracán con temperaturas mucho más bajas. (Satélite Terra, NASA)	25
Imagen 3: Comparación neurona biológica y un nodo de una red neuronal	27
Imagen 4: Capas de una red neuronal	27
Imagen 5: Comparativa de conexiones entre una red neuronal biológica y el concepto de red neuronal en tecnología	28
Imagen 6: Como ClimeLab simplifica el tratamiento de datos	32
Imagen 7: Diagrama de Gantt	41
Imagen 8: Grupo de Teams	44
Imagen 9: Canal de Teams	44
Imagen 10: Tablero de Trello	45
Imagen 11: Detalle de tarea en Trello	45
Imagen 12: Anaconda navigator	46
Imagen 13: Ilustración de una distribución de presión a nivel del mar	51
Imagen 14: Recorrido o track del huracán Katrina	52
Imagen 14: Muestra Notebook 1 - Carga de librerías y petición de dataset	55
Imagen 15: Muestra Notebook 2 - Muestra gráfica	55
Imagen 16: Muestra Notebook 3 - División conjunto muestra/test y creación del modelo	56
Imagen 17: Muestra Notebook 4 - Entrenamiento del modelo, evaluación, predicción y muestra de predicción	56
Imagen 18: Ejemplo de clasificación	57
Imagen 19: Perfil vertical de la atmósfera	59
Imagen 20: Relación entre espesor geopotencial y temperatura	60
Imagen 21: Anomalías en el espesor geopotencial	61
Imagen 22: Espesor geopotencial	62
Imagen 24: Función que resta los dos datasets para obtener el espesor geopotencial ...	63
Imagen 25: Función que divide el dataset	63
Imagen 25: Muestra Notebook 5 - Entrenamiento diferencia geopotencial	64
Imagen 26: Predicciones exitosas	65
Imagen 27: Muestra de HURDAT	67
Imagen 28: Ejemplo de predicción exitosa en casos dudosos	71

Índice de tablas

Tabla 1: Presupuesto recursos humanos	48
Tabla 2: Formato del Excel con los datos de los centros.....	53

Índice general

Resumen.....	5
Abstract	6
Índice de imágenes	8
Índice de tablas.....	10
<i>Parte I Descripción del proyecto</i>	<i>14</i>
Capítulo 1 Descripción del proyecto.....	15
1.1. Introducción	15
1.2. Objetivos del trabajo	17
1.3. Entorno de aplicación	19
Capítulo 2 Introducción a los Ciclones Tropicales	21
2.1. Ciclones Tropicales	21
2.2. Métodos de Detección y Seguimiento.....	24
2.3. Desafíos en la Predicción de Ciclones Tropicales	24
Capítulo 3 Contexto del trabajo	26
3.1. Contexto científico-técnico	26
3.2. Técnicas utilizadas	26
3.3. Tendencias actuales en el uso de ML para la predicción climática.....	30
Capítulo 4 Planificación	35
4.1. Metodología.....	35
4.2. Proceso de desarrollo	36
4.3. Diagrama de Gantt	40
4.4. Obstáculos.....	42
4.5. Tiempo total dedicado	42
4.6. Herramientas utilizadas	43
4.7. Presupuestos.....	46
<i>Parte II Desarrollo de la propuesta y resultados</i>	<i>49</i>
Capítulo 5 Cuadernos de aprendizaje.....	50
5.1. Caso de estudio elegido	50
5.2. Conjuntos de datos utilizados	50
5.2.1 Reanálisis ERA5.....	50
5.2.2 Base de Datos de Huracanes del Atlántico HURDAT	51
5.3. Datasets previos	52
5.3.1 Informe 1	55
5.3.2 Informe 2	58
Capítulo 6 Caso de estudio	67
6.1. Dataset final.....	67
Capítulo 7 Conclusiones y trabajo futuro.....	72
<i>Parte III Apéndices</i>	<i>75</i>
<i>Apéndice A Manual de Instalación.....</i>	<i>76</i>
1. Descarga e Instalación de Anaconda	76
<i>Apéndice B Contenido adjunto</i>	<i>78</i>
<i>Definición de siglas y abreviaturas.....</i>	<i>79</i>
<i>Bibliografía y Webgrafía</i>	<i>80</i>

Parte I Descripción del proyecto

Capítulo 1 Descripción del proyecto

1.1. Introducción

La predicción del tiempo, campo crucial para la seguridad y el bienestar humano, ha sido testigo de avances significativos desde la introducción de las computadoras en la década de 1950. A pesar de estos avances, sigue siendo una tarea desafiante que requiere una cantidad considerable de recursos y una capacidad de procesamiento aún insuficiente para abordar la abrumadora cantidad de datos generados por diversas fuentes meteorológicas. Estas fuentes, que incluyen observatorios de superficie, aviones, satélites y boyas marítimas, proporcionan datos continuos y heterogéneos desde todas las regiones del mundo, lo que plantea desafíos significativos en términos de adquisición, procesamiento y análisis de datos.

La asimilación de estas fuentes de datos en los modelos climatológicos, es decir, la inclusión de estos datos en la mejora de la predicción realizada por los modelos, es crucial para generar predicciones precisas y oportunas. Sin embargo, situaciones extraordinarias, como la pandemia global de COVID-19, han destacado la fragilidad de este sistema al interrumpir el suministro de datos de fuentes tradicionales como aviones y barcos. Esta interrupción ha puesto de relieve la necesidad de desarrollar enfoques alternativos y tecnologías innovadoras para garantizar la continuidad y la calidad de las predicciones meteorológicas.

En este contexto, el Machine Learning (ML) es una rama de la inteligencia artificial que ha surgido como una herramienta prometedora para abordar los desafíos en la predicción del tiempo. Algoritmos de ML pueden analizar grandes volúmenes de datos meteorológicos y aprender patrones complejos, lo que los hace ideales para mejorar la precisión y la eficiencia de las predicciones meteorológicas, ahorrando tiempo de computación. En particular, el enfoque en la identificación de ciclones, incluidos los CTs, representa un área de investigación crucial debido a la naturaleza única y la complejidad de estos fenómenos meteorológicos.

La predicción de CTs es un campo de investigación en constante desarrollo, impulsado por la necesidad de comprender y prever estos eventos climáticos críticos y dañinos. Las condiciones físicas y ambientales específicas requeridas para la formación de CTs hacen que la predicción precisa de estos eventos sea especialmente desafiante. Aunque el método de Análisis Dvorak en Etapa Temprana (EDA) [2] ha sido una herramienta valiosa para el seguimiento de CTs, la detección inicial sigue siendo un proceso que depende en gran medida de la experiencia y la intuición de los meteorólogos.

En este proyecto, nos proponemos explorar y aprovechar el potencial del ML para mejorar la identificación de CTs y facilitar así su seguimiento. Al desarrollar y aplicar modelos de ML a conjuntos de datos meteorológicos, buscamos avanzar en la comprensión y la predicción de estos fenómenos críticos, con el objetivo final de mejorar la capacidad de preparación y respuesta ante eventos climáticos extremos.

1.2. Motivación

La introducción de ML en los modelos meteorológicos actuales tiene el potencial de desempeñar un papel crucial en la reducción de los tiempos de respuesta ante catástrofes climáticas y en la elaboración de planes de actuación para mitigar algunas de sus devastadoras consecuencias. Por ejemplo, las olas de calor representan una de las principales causas de mortalidad en todo el mundo, y en España, los fallecimientos relacionados con desastres naturales, como inundaciones, olas de calor y temporales marinos, continúan siendo una preocupación significativa para la seguridad pública [12].

Desde una perspectiva económica, la implementación efectiva de modelos de ML en la predicción del tiempo podría traducirse en una reducción sustancial de los recursos financieros empleados por los gobiernos para hacer frente a los estragos causados por desastres meteorológicos. Según datos recopilados en el informe anual "Coste de las Catástrofes", elaborado por la Fundación Aon [4], el gobierno español ha asignado una cantidad considerable para cubrir los daños derivados de desastres naturales como DANA, inundaciones y temporales, alcanzando los 12.067 millones de euros entre los años 2016 y 2020. Esta cifra representa un gasto promedio anual de aproximadamente 2.413 millones de euros, una carga financiera que podría ser significativamente reducida con la implementación efectiva de tecnologías innovadoras en la predicción y prevención de desastres climáticos.

Un ejemplo ilustrativo de los impactos económicos adversos provocados por desastres meteorológicos se evidencia en el caso del temporal Filomena, que azotó España en enero de 2021. Se estima que este fenómeno climático causó un asombroso coste económico de 1.157 millones de euros, de los cuales menos de la mitad estaban asegurados, según se destaca en el informe mencionado anteriormente. Estas cifras ponen de manifiesto la urgente necesidad de adoptar enfoques más avanzados y efectivos en la predicción del tiempo, con el fin de reducir los impactos negativos tanto en términos de pérdidas humanas como económicas.

Ejemplos Adicionales de Desastres Naturales Recientes:

- **Huracanes y tormentas tropicales en el Atlántico:** La temporada de huracanes en el Atlántico ha sido particularmente activa en los últimos años, con eventos como el huracán Harvey en 2017, que causó graves inundaciones en Texas y Luisiana, y el huracán María en 2017, que devastó Puerto Rico, dejando una estela de destrucción y causando miles de muertes. Por otra parte, Europa Occidental se está viendo amenazada por CTs anómalos en los últimos 15 años. El huracán Vince y la tormenta tropical Delta en 2005, el huracán Alex en 2016, el huracán Ofelia en 2017 o recientemente el huracán Leslie en octubre de 2018 siguieron trayectorias inusuales y se intensificaron llegando a afectar severamente algunos de ellos a territorio español, en particular a las islas Canarias.
- **Incendios Forestales en Australia:** En los últimos años, Australia ha experimentado una serie de incendios forestales catastróficos, como los ocurridos en 2019 y 2020. Estos incendios arrasaron millones de hectáreas de bosques, destruyeron miles de viviendas y causaron la pérdida de vidas humanas y animales, además de tener un impacto duradero en el medio ambiente y la calidad del aire.

- **Inundaciones en Asia:** Las inundaciones son un fenómeno recurrente en países como India, Bangladesh y China, donde las fuertes precipitaciones monzónicas pueden provocar desbordamientos de ríos y devastar comunidades enteras. En 2020, por ejemplo, las inundaciones en el sur de Asia afectaron a millones de personas, dejando a su paso un saldo trágico de muertes y desplazamientos.

Estos ejemplos ilustran la diversidad y la gravedad de los desastres naturales que ocurren en todo el mundo, y subrayan la importancia de mejorar las técnicas de predicción meteorológica para prevenir pérdidas humanas y materiales. Es fundamental que la investigación en este campo avance rápidamente para proporcionar a las comunidades y a las autoridades los conocimientos y las herramientas necesarias para anticiparse y mitigar los efectos de estos eventos climáticos extremos.

1.2. Objetivos del trabajo

En este proyecto, se abordan tanto aspectos teóricos como prácticos relacionados con el uso de modelos de ML para la identificación de fenómenos atmosféricos específicos asociados con CTs. Se han identificado los siguientes objetivos principales:

- **OBJ-1: Estudiar cómo se realizan las predicciones meteorológicas actuales y cuál es la mejor aproximación para abordarlo desde un modelo predictivo.**

Este objetivo implica una revisión exhaustiva de los métodos y técnicas utilizados en la predicción meteorológica convencional. Se analizan las limitaciones y los desafíos actuales, así como las oportunidades potenciales para mejorar mediante el uso de modelos de ML.

- **OBJ-2: Desarrollar un modelo predictivo entrenado con datos clasificados previamente.**

Se diseña y entrena un modelo de ML utilizando conjuntos de datos previamente clasificados. El objetivo es desarrollar un modelo que pueda identificar eficazmente los patrones asociados con las altas y bajas presiones para llegar a la identificación de los CTs.

- **OBJ-3: Adquisición de Datos.**

Se utilizan datos climáticos del reanálisis ERA5 del European Centre for Medium-Range Weather Forecasts (ECMWF), específicamente de los niveles de presión de 500 y 925 hPa. La calidad y la cantidad de los datos adquiridos son fundamentales para el éxito del proyecto.

- **OBJ-4: Preprocesamiento de los datos.**

Se realiza un exhaustivo preprocesamiento de los datos para prepararlos para su posterior análisis y entrenamiento del modelo. Esto incluirá la creación de una muestra de datos con información esencial, como fecha, hora, ubicación y etiquetas de tipo de ciclón.

- **OBJ-5: Diseño y Entrenamiento del Modelo.**

Se implementa un modelo de red neuronal utilizando la biblioteca Keras. La arquitectura del modelo incluirá capas de entrada y salida, así como capas ocultas, diseñadas para capturar y aprender los patrones complejos presentes en los datos climáticos.

- **OBJ-6: Evaluación del Modelo.**

El modelo entrenado se evalúa exhaustivamente utilizando una muestra de datos independiente. Se realizan predicciones y se comparan con las etiquetas reales para determinar la eficacia y precisión del modelo calculando diversas métricas de rendimiento.

Relevancia Práctica

La consecución de los objetivos planteados en este proyecto tiene implicaciones significativas y beneficios concretos tanto para la sociedad en general como para las instituciones meteorológicas específicamente.

Impacto en la Sociedad

El desarrollo de modelos de ML más precisos y eficientes para la identificación y predicción de CTs contribuirá directamente a la seguridad y el bienestar de las comunidades vulnerables a estos fenómenos. Al mejorar la capacidad de prever la trayectoria y la intensidad de los ciclones, se podrán tomar medidas preventivas con mayor antelación, lo que reducirá el riesgo de pérdidas humanas y materiales asociadas a estos eventos extremos. Además, una mejor comprensión de la evolución de los CTs permitirá una planificación más efectiva de las operaciones de evacuación y respuesta de emergencia, lo que ayudará a minimizar el impacto en la infraestructura y los recursos públicos.

Beneficios para las Instituciones Meteorológicas

Para las instituciones meteorológicas y los organismos encargados de la gestión de desastres naturales, el acceso a modelos de ML avanzados representa una herramienta invaluable para mejorar la precisión de sus pronósticos y alertas tempranas. Al integrar estas herramientas en sus sistemas de monitoreo y predicción, podrán ofrecer servicios meteorológicos más confiables y oportunos a la población, fortaleciendo así su capacidad de respuesta ante eventos climáticos extremos. Además, el desarrollo de capacidades en el ámbito del ML posicionará a estas instituciones en la vanguardia de la innovación científica y tecnológica, permitiéndoles mantenerse a la altura de los desafíos climáticos en constante evolución.

Problemas que pueden surgir y que puedan afectar a la consecución de estos objetivos

- Disponibilidad de los participantes: este proyecto se desarrolla con la ayuda de profesionales del entorno de la meteorología; por este motivo, el buen ritmo de

este proyecto puede verse condicionado a su vida laboral o disponibilidad dentro de sus labores diarias, así como mi propia vida laboral.

- Clasificación de datos: el modelo de este proyecto se entrena proporcionándole datos clasificados, dichos datos necesitan ser clasificados previamente para no introducir errores en el modelo.

1.3. Entorno de aplicación

Los modelos numéricos meteorológicos empleados en la actualidad son herramientas que resuelven o aproximan la solución de ecuaciones diferenciales en derivadas parciales no lineales necesarias para generar pronósticos meteorológicos con base en condiciones iniciales conocidas. Estos modelos dividen la esfera terrestre en una malla tridimensional que abarca longitud, latitud y altitud, recopilando datos iniciales de numerosas variables atmosféricas en cada punto de la malla relevante y su evolución en el tiempo.

En el ámbito de los modelos meteorológicos, existen múltiples enfoques desarrollados y utilizados por diversas instituciones, diferenciándose en términos de tamaño y área de aplicación:

- **Modelos regionales:** Estos modelos se centran en áreas geográficas específicas, lo que permite una mayor precisión debido a la menor escala de la cuadrícula utilizada. Ejemplos incluyen el modelo HARMONIE de AEMET, con una resolución horizontal de 2.5 km; el modelo AROME empleado por METEOFRANCE, con una resolución horizontal de 1.3 km; y el modelo WRF utilizado por METEOGALICIA, con una resolución horizontal de 1 km.
- **Modelo global:** Estos modelos ofrecen pronósticos a escala global. En este estudio vamos a destacar dos de ellos:
 - GFS modelo que usa Estados Unidos en la NOAA (National Oceanic and Atmospheric Administration).
 - IFS modelo que usa en Europa en el ECMWF.

Los datos utilizados en este estudio provendrán principalmente del ECMWF, mientras que se emplearán datos de NOAA para identificar los centros de los fenómenos atmosféricos.

Se han llevado a cabo numerosos estudios sobre el uso de ML en la predicción de precipitaciones, destacando investigaciones como MetNet y MetNet2, realizadas por el equipo de Google Research. En el contexto de la predicción de ciclones utilizando ML, la mayoría de los estudios se centran en la detección, el seguimiento y la predicción de estos fenómenos. Los resultados obtenidos con deep learning han demostrado mejorar considerablemente la fiabilidad de las predicciones en comparación con los modelos numéricos tradicionales.

Desafíos en la Identificación y Predicción de Ciclones Tropicales

En esta sección se discuten algunos de los desafíos a los que se enfrentan los pronosticadores meteorológicos en relación con la identificación y predicción de los CTs:

1. Detección Temprana y Monitoreo Continuo:

La identificación temprana de la formación de un CT es fundamental para alertar a las comunidades y tomar medidas de precaución. Sin embargo, la detección inicial de estos sistemas puede ser difícil, especialmente en áreas oceánicas apartadas donde la cobertura de observación es limitada. Además, una vez que se ha formado un ciclón, su monitoreo continuo presenta desafíos debido a su movimiento irregular y a las variaciones en su intensidad.

2. Modelado de Trayectoria e Intensidad:

La predicción precisa de la trayectoria y la intensidad de un CT es crucial para la planificación de evacuaciones, la preparación de infraestructuras y la asignación de recursos de ayuda. Sin embargo, la trayectoria de un ciclón puede ser influenciada por múltiples factores, incluyendo la topografía local, la interacción con otros sistemas atmosféricos y la variabilidad en la intensidad del sistema. Además, la predicción de la intensidad de un CT es particularmente desafiante debido a la falta de comprensión completa de los procesos físicos que controlan su desarrollo y fortalecimiento.

3. Pronóstico de Impactos:

Además de predecir la trayectoria e intensidad de un CT, los pronosticadores también enfrentan el desafío de prever los impactos específicos que este fenómeno puede tener en áreas afectadas. Esto incluye la posibilidad de fuertes lluvias, inundaciones costeras, marejadas ciclónicas, vientos destructivos y tornados. La incertidumbre en la predicción de estos impactos puede dificultar la toma de decisiones por parte de las autoridades y la población en riesgo.

Capítulo 2 Introducción a los Ciclones Tropicales

2.1. Ciclones Tropicales

En el vasto escenario de la meteorología, los CTs emergen como una de las manifestaciones más impactantes y dinámicas de los sistemas atmosféricos. Estas poderosas tormentas, que incluyen huracanes, tifones y tormentas tropicales, no solo ejercen una influencia significativa en las regiones que tocan tierra, sino que también desempeñan un papel crucial en la regulación del clima global [16]

Antes de adentrarnos en la clasificación de los CTs veamos en qué consisten.

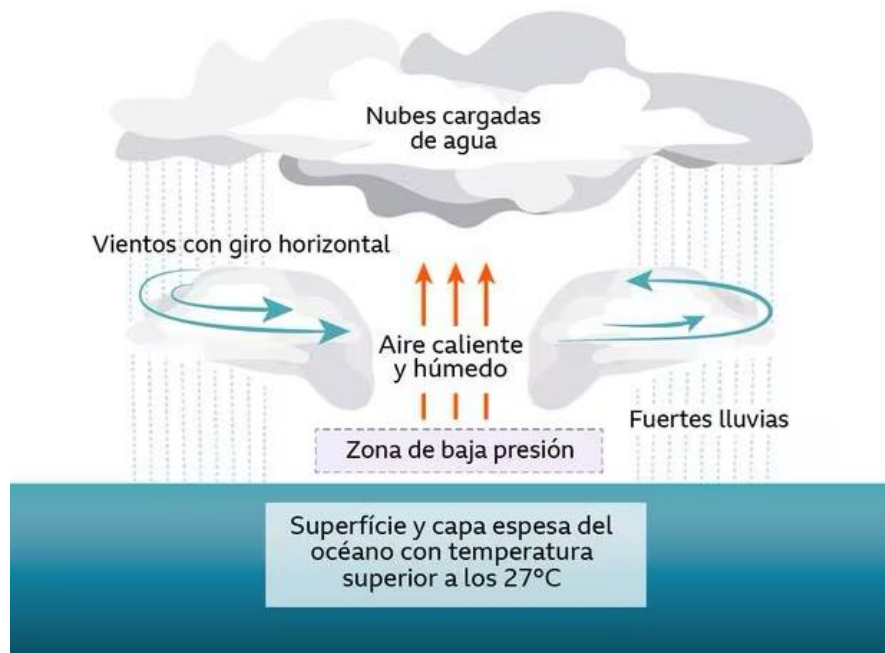


Imagen 1: Formación de un CT

Un CT es un fenómeno meteorológico caracterizado por una circulación cerrada de vientos fuertes y una baja presión atmosférica en el centro, con una formación sobre aguas cálidas tropicales. Estos sistemas pueden generar lluvias intensas, vientos violentos y marejadas ciclónicas, y se clasifican en varias categorías según su intensidad [7].

La formación y la intensificación de los CTs están influenciadas por diversos factores meteorológicos, incluida la temperatura superficial del mar, que debe superar los 26.5 °C para proporcionar la energía necesaria para su desarrollo. Además, la presencia de vientos débiles en la atmósfera superior y la ausencia de cizalladura del viento vertical son condiciones favorables para el fortalecimiento de los CTs [17].

Según [7] se tienen depresiones tropicales, tormentas tropicales y huracanes.

La Depresión Tropical (TD) es el estado más temprano de un CT, caracterizado por vientos sostenidos de hasta 62 km/h (38 mph).

1. **La Tormenta Tropical (TS):** se forma cuando los vientos sostenidos alcanzan velocidades entre 63 km/h (39 mph) y 118 km/h (73 mph). Las tormentas tropicales pueden producir lluvias fuertes y vientos dañinos.
2. **El Huracán o Tifón (HU)** es la forma más intensa de un CT, con vientos sostenidos de al menos 119 km/h (74 mph). Los huracanes y tifones pueden ser extremadamente destructivos, causando inundaciones, deslizamientos de tierra, marejadas ciclónicas y devastación generalizada.

Además de los CTs, también es importante tener en cuenta otros tipos de fenómenos atmosféricos que pueden ser relevantes en el contexto de este estudio:

1. **Ciclón Extratropical (EX):** Un ciclón que ha perdido sus características tropicales y se ha convertido en un sistema de tormenta típico de las latitudes medias.
2. **Depresión Subtropical (SD):** Un ciclón subtropical con vientos máximos sostenidos de menos de 34 nudos.
3. **Tormenta Subtropical (SS):** Un ciclón subtropical con vientos máximos sostenidos que exceden los 34 nudos.
4. **Baja (LO):** Una perturbación meteorológica que no cumple con los criterios de un CT, subtropical o extratropical.
5. **Onda Tropical (WV):** Un tipo de vaguada atmosférica, o curvatura ciclónica, en las latitudes tropicales.
6. **Perturbación (DB):** Un término amplio utilizado para describir cualquier fenómeno meteorológico que altera los patrones climáticos normales.

Un sistema atmosférico está caracterizado además por su ciclo de vida. El ciclo de vida de un CT comprende varias etapas distintivas, que incluyen la formación, la intensificación, la madurez y la disipación.

1. **Formación:** Todo comienza con la aparición de una perturbación atmosférica sobre cálidas aguas tropicales. Esta perturbación puede originarse por una variedad de factores, como ondas tropicales, áreas de baja presión, o perturbaciones atmosféricas preexistentes. Cuando las condiciones son propicias, la perturbación comienza a organizarse y se forma una depresión tropical [7] [19]. Un proceso clave en esta etapa es la **convección atmosférica**, donde el aire cálido y húmedo asciende, generando nubes de tormenta y liberación de calor latente, lo que proporciona la energía necesaria para el desarrollo inicial del ciclón.
2. **Depresión Tropical:** En esta etapa inicial, la perturbación atmosférica adquiere una circulación cerrada de vientos débiles y una baja presión central definida. Los vientos sostenidos oscilan entre 25 y 62 km/h (16 a 38 mph). Si la depresión tropical continúa fortaleciéndose y las condiciones son favorables, puede evolucionar hacia una tormenta tropical [7] [19]. La liberación de calor latente juega un papel crucial, ya que la condensación del vapor de agua libera energía adicional, que alimenta la convección y contribuye a la intensificación del sistema.

3. **Tormenta Tropical:** Cuando los vientos sostenidos alcanzan velocidades entre 63 y 118 km/h (39 a 73 mph), la depresión tropical se clasifica como tormenta tropical. En esta etapa, la circulación de vientos se organiza aún más, formando bandas nubosas y lluvias intensas alrededor del centro del sistema [7] [19].

Las tormentas tropicales pueden producir fuertes vientos, lluvias torrenciales y marejadas ciclónicas en las áreas costeras. La convergencia de vientos en la atmósfera inferior es fundamental, ya que los vientos que se mueven hacia el centro del ciclón aumentan la elevación del aire, promoviendo el desarrollo de una estructura más organizada.

4. **Huracán o Tifón:** Si las condiciones continúan siendo favorables y la tormenta tropical experimenta un mayor fortalecimiento, puede alcanzar la categoría de huracán (en el Atlántico y el noreste del Pacífico) o tifón (en el noroeste del Pacífico). Los huracanes y tifones se caracterizan por vientos sostenidos de al menos 119 km/h (74 mph) y pueden clasificarse en diferentes categorías según su intensidad, como la escala Saffir-Simpson para huracanes en el Atlántico [7] [19]. La fuerza de Coriolis es crítica en esta fase, ya que la rotación de la Tierra desvía los vientos, estableciendo la estructura de vórtice que define a los CTs.
5. **Madurez:** Durante esta etapa, el CT alcanza su máxima intensidad y se convierte en una fuerza poderosa de la naturaleza. Los vientos son extremadamente fuertes, las lluvias son torrenciales y las marejadas ciclónicas pueden inundar extensas áreas costeras. Esta fase puede durar varias horas o días, dependiendo de las condiciones atmosféricas y del entorno [7] [19]. El ciclón se mantiene por la continua liberación de calor latente y la convección que alimentan el sistema.
6. **Disipación:** Eventualmente, el CT encuentra condiciones menos favorables, como aguas más frías, cizalladura del viento o interacción con la tierra, lo que provoca su debilitamiento gradual. La circulación de vientos se desorganiza, las bandas de lluvia se dispersan y el sistema pierde su estructura característica. Finalmente, el CT se disipa por completo, convirtiéndose en una perturbación remanente o integrándose en sistemas meteorológicos más grandes [7] [19]. Factores como la cizalladura del viento y la interacción con tierra firme contribuyen a la pérdida de la organización estructural del ciclón.

Este ciclo dinámico es fundamental para comprender la evolución y el comportamiento de los CTs, y es muy importante para el desarrollo y la evaluación de nuestro modelo predictivo. Al estudiar cada etapa del ciclo de vida, podremos identificar patrones y factores clave que influyen en la intensidad y el trayecto de estos sistemas meteorológicos, lo que nos permitirá mejorar la precisión de nuestras predicciones y aumentar nuestra capacidad para anticipar y mitigar los impactos de los CTs.

2.2. Métodos de Detección y Seguimiento

En este apartado vamos a enumerar los principales dispositivos que permiten la recolección de datos meteorológicos actualmente y por ende, permiten el estudio y seguimiento de los fenómenos que nos ocupan.

Las imágenes satelitales son una herramienta fundamental para la detección y seguimiento de CTs. Permiten visualizar la evolución de la estructura de la tormenta, identificar la formación de bandas nubosas, detectar la presencia de un ojo en el caso de huracanes bien desarrollados y monitorizar cambios en la temperatura de la superficie del mar, lo que puede indicar la intensificación o debilitamiento del ciclón.

Los radares meteorológicos terrestres y marinos son muy importantes para el seguimiento de la precipitación asociada con los CTs. Permiten detectar la ubicación y la intensidad de las lluvias, así como la presencia de posibles tornados dentro del sistema. Además, los radares Doppler [12] pueden proporcionar información sobre la velocidad y dirección del viento dentro de la tormenta.

Las boyas marinas son dispositivos flotantes equipados con sensores que recopilan datos sobre la temperatura del agua, la presión atmosférica y la velocidad y dirección del viento en la superficie del mar. Estos datos son cruciales para comprender las condiciones ambientales que afectan la evolución de un CT y pueden ayudar a mejorar las predicciones de su trayectoria e intensidad.

Las aeronaves de reconocimiento, como los aviones cazahuracanes, son utilizadas por agencias meteorológicas para realizar mediciones directas dentro y alrededor de los CTs. Estas mediciones incluyen la velocidad y dirección del viento, la presión atmosférica y la temperatura, proporcionando datos invaluable para la validación de modelos de predicción y la mejora de la precisión de las predicciones.

2.3. Desafíos en la Predicción de Ciclones Tropicales

Uno de los desafíos más significativos en la predicción de CTs es la incertidumbre propia en los modelos meteorológicos utilizados para prever su trayectoria e intensidad. Los CTs son sistemas atmosféricos altamente complejos y dinámicos, y su evolución puede estar influenciada por una variedad de factores difíciles de modelar con precisión, como ya hemos comentado. La variabilidad natural en la atmósfera también representa un desafío para la predicción de CTs. Los cambios en los patrones de circulación atmosférica, como el fenómeno El Niño-La Niña, pueden alterar las condiciones ambientales y afectar la formación y evolución de los CTs en diferentes regiones del mundo [21]. Esta variabilidad dificulta la predicción a largo plazo y puede conducir a errores en las proyecciones de trayectoria e intensidad.

Por otra parte, la influencia del Cambio Climático está siendo crítica en la frecuencia e intensidad de los CTs [26]. El cambio climático plantea nuevos desafíos para la predicción de CTs, ya que se espera que altere las condiciones ambientales en las que se forman y desarrollan estos sistemas. Si bien aún existe debate sobre el impacto exacto del cambio climático en la frecuencia e intensidad de los CTs, es probable que el aumento de

la temperatura de la superficie del mar modifique los patrones de circulación atmosférica, lo que podría tener consecuencias significativas para su predicción (BIBLIO).

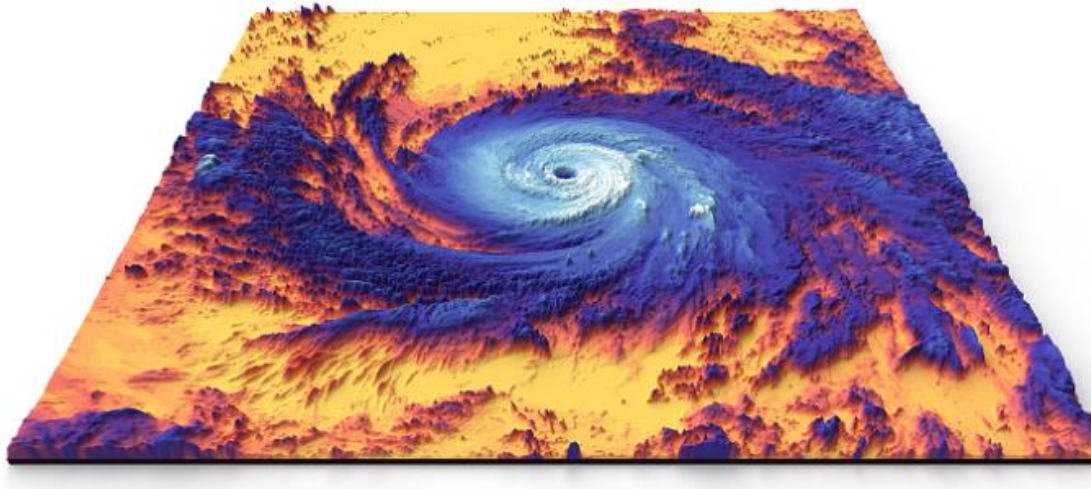


Imagen 2: Imagen térmica del huracán María de categoría 5 en 2017. En amarillo y naranja, aguas oceánicas cálidas, en azul y blanco, las cimas de las nubes del huracán con temperaturas mucho más bajas. (Satélite Terra, NASA)

Capítulo 3 Contexto del trabajo

3.1. Contexto científico-técnico

Como se ha mencionado la predicción climatológica es un campo crucial en la meteorología y la ciencia del clima, con aplicaciones que van desde la planificación agrícola hasta la gestión de desastres naturales... En los últimos años, el enfoque tradicional basado en modelos físicos ha sido complementado y enriquecido por técnicas de ML, que han demostrado ser eficaces para mejorar la precisión y la resolución temporal de las predicciones.

El ML aplicado a la predicción climatológica tiene como objetivo desarrollar algoritmos capaces de aprender relaciones complejas entre variables atmosféricas y producir pronósticos precisos a diferentes escalas temporales. A diferencia de los modelos físicos que dependen de ecuaciones complejas y de parámetros difíciles de ajustar, los modelos de ML pueden capturar patrones no lineales en los datos y adaptarse dinámicamente a las condiciones cambiantes del clima.

Además, el proyecto implica el análisis exhaustivo de grandes conjuntos de datos meteorológicos y climáticos para identificar tendencias, patrones y anomalías significativas. Esto implica el uso de técnicas estadísticas y de aprendizaje automático para procesar y analizar datos complejos, así como la visualización de datos para comunicar de manera efectiva los resultados.

3.2. Técnicas utilizadas

Aprendizaje Supervisado y No Supervisado

En el contexto de la predicción climatológica, se emplean tanto técnicas de aprendizaje supervisado como no supervisado [12]. El aprendizaje supervisado, utilizado en este proyecto, utiliza conjuntos de datos históricos etiquetados, donde cada muestra está asociada con una salida conocida, como la temperatura o la precipitación futura; en este caso usamos la presión atmosférica para una clasificación binaria de CT y ciclón no tropical. Los algoritmos supervisados, como las redes neuronales y los árboles de decisión [23][24][25], se entrenan para aprender las relaciones entre las entradas (variables meteorológicas pasadas) y las salidas (variables meteorológicas futuras).

Por otro lado, el aprendizaje no supervisado se utiliza para descubrir patrones subyacentes en los datos sin la necesidad de etiquetas explícitas [23][12][25]. Estas técnicas, como el *clustering* y la reducción de la dimensionalidad, pueden ayudar a identificar patrones climáticos emergentes, clasificar regiones similares en función de su comportamiento climático y detectar anomalías en los datos observados.

Redes Neuronales

En el contexto que nos ocupa, la capacidad que tenemos los humanos de emitir un diagnóstico con la observación de nuestro entorno, por ejemplo, mirando el cielo, sería de mucha utilidad en el ámbito tecnológico.

Por ello, intentamos transmitir estas observaciones a través de imágenes que se apoyan en datos que aportan valor informativo, como la presión atmosférica en ese momento, la altitud, la humedad... a los ordenadores con el fin de que puedan aprender a extraer y reconocer patrones que le permitan llegar a los mismos procesos de comprensión y toma de decisiones que los seres humanos.

Las redes neuronales [24] son modelos computacionales que intentan imitar el funcionamiento y la estructura de nuestro cerebro. Se caracterizan por:

1. **Neuronas:** Son unidades básicas de procesamiento en una red neuronal. Cada neurona toma una o más entradas, las procesa mediante una función de activación y produce una salida. Estas neuronas se denominan nodos e igual que las neuronas biológicas, se comunican con el resto de nodos creando una “red”.



Imagen 3: Comparación neurona biológica y un nodo de una red neuronal

2. **Capas:** Las neuronas en una red neuronal se organizan en capas. Las capas se pueden dividir en tres tipos principales:
 - **Capa de entrada (input):** Recibe los datos de entrada y los transmite a la capa siguiente.
 - **Capas ocultas (hidden):** Realizan el procesamiento intermedio entre la capa de entrada y la capa de salida.
 - **Capa de salida (output):** Produce los resultados finales de la red neuronal.

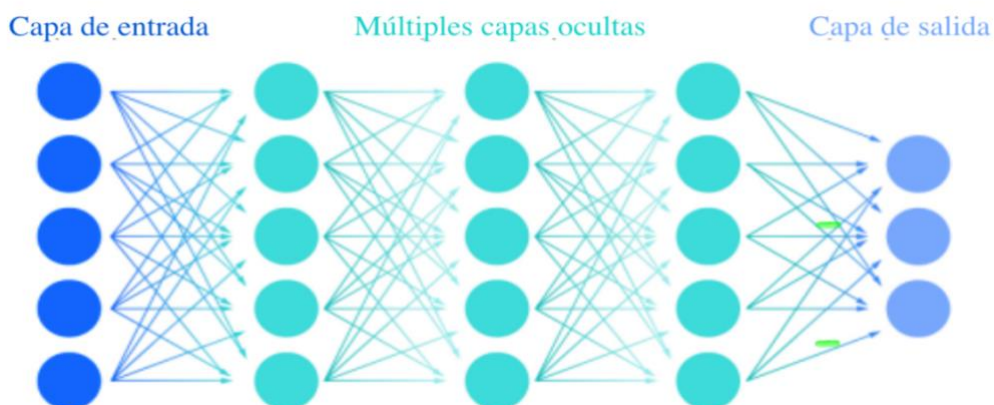


Imagen 4: Capas de una red neuronal

3. **Conexiones:** Las neuronas agrupadas en capas se conectadas entre sí a través de conexiones donde cada nodo tiene un peso y un umbral asociado que determinan su importancia en la propagación de la información a través de la red.

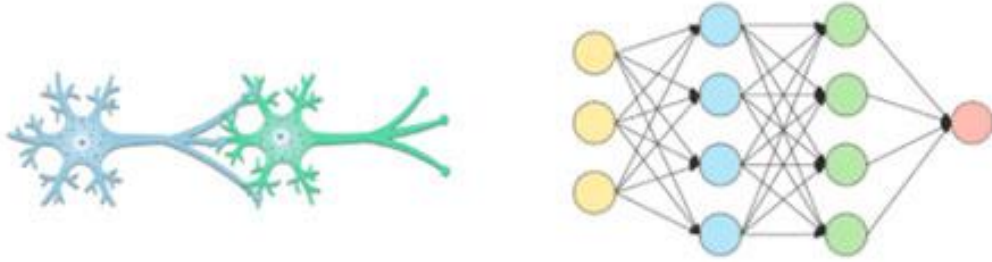


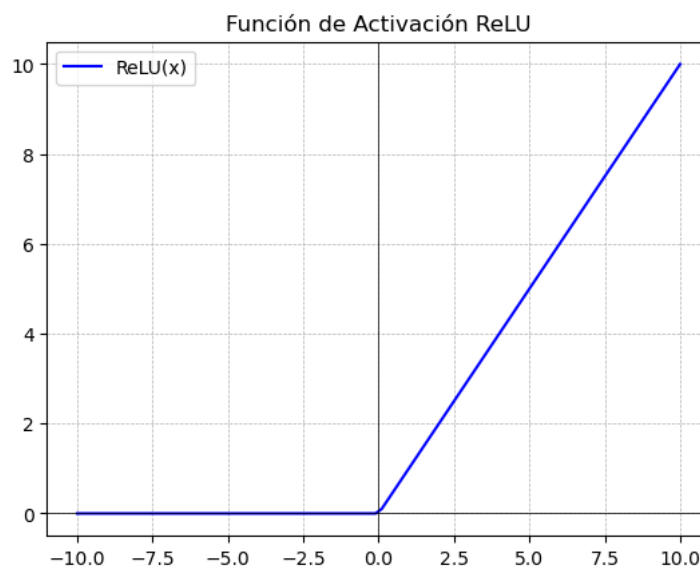
Imagen 5: Comparativa de conexiones entre una red neuronal biológica y el concepto de red neuronal en tecnología

4. **Funciones de activación:** Cada neurona utiliza una función de activación para determinar su salida en función de las entradas recibidas y los pesos de las conexiones. Algunas funciones de activación comunes incluyen la función sigmoide, la función softmax, la función ReLU (Rectified Linear Unit) y la función de activación lineal. En este caso vamos a contar con una capa con una función de activación ReLU y otra capa con función de activación softmax.

- **Función ReLU:** es una de las más utilizadas en redes neuronales profundas porque mejora la convergencia y evita problemas como el **vanishing gradient**, ocurre cuando los gradientes en una red neuronal se hacen **muy pequeños** durante la retropropagación, dificultando la actualización de los pesos de las capas más profundas. Su función es $ReLU(x) = \max(0, x)$.

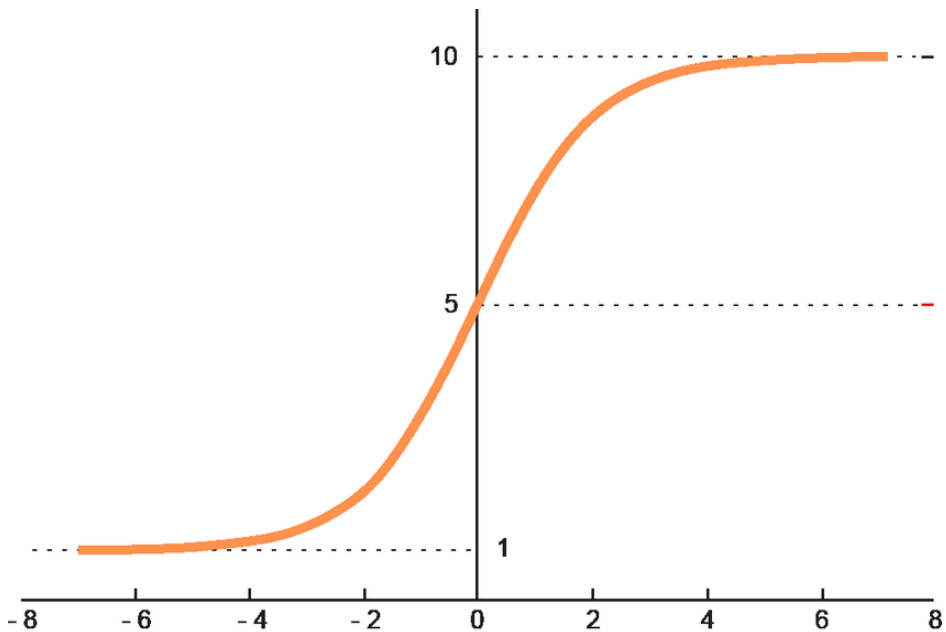
Esto significa:

- Si $x > 0$, entonces $ReLU(x) = x$. (el valor se mantiene igual).
- Si $x \leq 0$, entonces $ReLU(x) = 0$. (se convierte en cero).



Función Softmax: Toma un vector de entrada de valores reales y produce un vector de salida de la misma longitud, donde cada elemento está en el rango de 0 a 1 y la suma de todos los elementos es igual a 1. Cada valor en el vector de salida representa la probabilidad de que la entrada pertenezca a una clase específica.

Su función es $\text{softmax}(x_i) = \frac{e^{x_i}}{\sum_{j=1}^N e^{x_j}}$



5. **Entrenamiento:** El proceso de entrenamiento de una red neuronal implica ajustar los pesos de las conexiones para que la red pueda producir las salidas deseadas para un conjunto dado de entradas. Esto generalmente se hace utilizando algoritmos de optimización como el descenso de gradiente estocástico (SGD) o algoritmos más avanzados como Adam o RMSprop.
6. **Aprendizaje profundo:** Las redes neuronales profundas (también conocidas como redes neuronales profundas) son redes neuronales con múltiples capas ocultas. Estas redes han demostrado ser muy efectivas en una variedad de tareas de aprendizaje automático, como reconocimiento de imágenes, procesamiento de lenguaje natural, visión por computadora y más.

Redes Neuronales Convolucionales (CNN), Redes Neuronales Recurrentes (RNN) y Redes Neuronales Densamente conectadas

Las Redes Neuronales Convolucionales (CNN) son especialmente útiles para el procesamiento de imágenes satelitales y mapas meteorológicos [27]. Estas redes, diseñadas para procesar datos que tienen una topología de rejilla, son capaces de extraer características espaciales y temporales de los datos, como patrones de nubes, vientos y temperaturas, lo que las hace ideales para la predicción climática a corto plazo.

Por otro lado, las Redes Neuronales Recurrentes (RNN) son adecuadas para modelar la dependencia temporal en los datos climáticos, como series temporales de temperatura y

precipitación. Las RNN pueden capturar la evolución de los patrones climáticos a lo largo del tiempo y generar pronósticos precisos para diferentes horizontes de tiempo.

Las redes neuronales densamente conectadas (también conocidas como fully connected networks) están compuestas por capas de neuronas donde cada neurona en una capa está conectada a todas las neuronas de la capa anterior y posterior, lo que las hace "densamente" conectadas. Son útiles en escenarios donde las relaciones complejas entre características individuales necesitan capturarse para realizar clasificaciones. Aunque no manejan la localidad espacial como las CNN ni las secuencias temporales como las RNN, son muy eficientes para encontrar patrones en un conjunto de datos bien definido y normalizado. Este tipo de redes son las adecuadas para el fin de este proyecto, por eso vamos a indagar un poco más en su composición y funcionamiento:

1. **Arquitectura Layered:** Estas redes tienen una estructura en capas, que típicamente incluyen una capa de entrada, varias capas ocultas y una capa de salida. Cada capa está compuesta por una serie de neuronas o nodos conectados a los nodos de la capa anterior y posterior.
2. **Funciones de Activación:** Utilizan funciones de activación para introducir no linealidades en el modelo, permitiéndoles aprender relaciones complejas en los datos. Las funciones más comunes incluyen ReLU (Rectified Linear Unit), sigmoid y tanh. La elección de la función de activación puede depender del tipo específico de problema y de las características de los datos.
3. **Propagación Hacia Adelante (Forward Propagation):** Durante el entrenamiento y la inferencia, los datos de entrada pasan a través de las capas de la red de manera secuencial, comenzando en la capa de entrada, pasando por las capas ocultas y terminando en la capa de salida donde se hace una predicción.
4. **Backpropagation y Optimización:** Utiliza el algoritmo de backpropagation para actualizar los pesos de la red. Este proceso implica calcular el gradiente de la función de pérdida respecto a cada peso utilizando la regla de la cadena, y luego ajustar los pesos para minimizar la pérdida. Los optimizadores comunes incluyen SGD (Stochastic Gradient Descent), Adam y RMSprop.

3.3. Tendencias actuales en el uso de ML para la predicción climática

El campo de las redes neuronales y deep learning es un área vasta y en constante evolución, especialmente en aplicaciones específicas como la meteorología y la predicción de ciclones. Algunas de las propuestas en las que se ha integrado el uso de redes neuronales para mejorar la precisión de los resultados son:

Descripción de trabajos relacionados

- **Predicción de la Formación de Ciclones Tropicales:** Un estudio reciente utilizó redes neuronales convolucionales, específicamente ResNet y UNet, para predecir la formación de CTs a partir de datos ambientales a gran escala en el Océano Pacífico. Este estudio demostró que el deep learning puede alcanzar una habilidad de pronóstico óptima a 12–18 horas de antelación, lo

que sugiere un enfoque alternativo a los métodos tradicionales de seguimiento de vórtices [13].

- **Estimación del Tamaño de Ciclones Tropicales:** Otra investigación aplicó deep learning para estimar el tamaño de CTs usando datos de satélite infrarrojos y microondas. El modelo fue diseñado para aprender relaciones intrínsecas entre los datos de imagen de los ciclones y sus características, ofreciendo una nueva manera de estimar parámetros críticos de los ciclones [14]
- **Predicción de Trayectorias de Ciclones Tropicales:** Un enfoque innovador para predecir las trayectorias de los CTs involucró el uso de autoencoders y redes recurrentes con unidades de compuerta (GRU). Este modelo combinó capas de preprocesamiento de datos, autoencoders y GRUs para predecir lugares de aterrizaje de CTs basándose en trayectorias históricas y atributos meteorológicos. Los resultados indicaron que este modelo superaba significativamente a los métodos de predicción numérica del tiempo en términos de precisión [15].

Colaboraciones y Actividades Relacionadas con el Aprendizaje Automático en ECMWF

ECMWF ha nombrado un coordinador de Inteligencia Artificial (IA). y aprendizaje automático para coordinar los esfuerzos paninstitucionales y ha comenzado a adquirir hardware de GPU adecuado para los proyectos de aprendizaje automático, para el nuevo HPC y para la European Weather Cloud, desarrollada en colaboración con EUMETSAT.

Se ha publicado el primer conjunto de datos de referencia para aplicaciones de aprendizaje automático en modelado del tiempo y el clima, con contribuciones del ECMWF (Weatherbench; Rasp et al., 2020), y se están realizando más esfuerzos con las contribuciones del ECMWF.

Además de las iniciativas actuales de uso de Python en ECMWF (como la capacitación y las API de datos), se ha introducido un nuevo proyecto llamado CliMetLab con el objetivo expreso de ayudar a las aplicaciones de aprendizaje automático a simplificar el acceso a la información climática y meteorológica. CliMetLab permite a los usuarios concentrarse en la investigación en lugar de en problemas técnicos como el acceso a datos y formatos, integrando la importación de datos desde el ¹Copernicus Climate Change Service Climate Data Store (CDS) y el ECMWF Meteorological Archival and Retrieval System (MARS) en entornos de Python.

Lo que hace ClimetLab es actuar como interfaz entre el lado izquierdo y derecho de la imagen:

¹ Copernicus es el programa insignia de la Unión Europea para la observación de la Tierra. Proporciona datos y servicios gratuitos y abiertos en apoyo de la protección del medio ambiente, la comprensión del cambio climático y la gestión de emergencias.

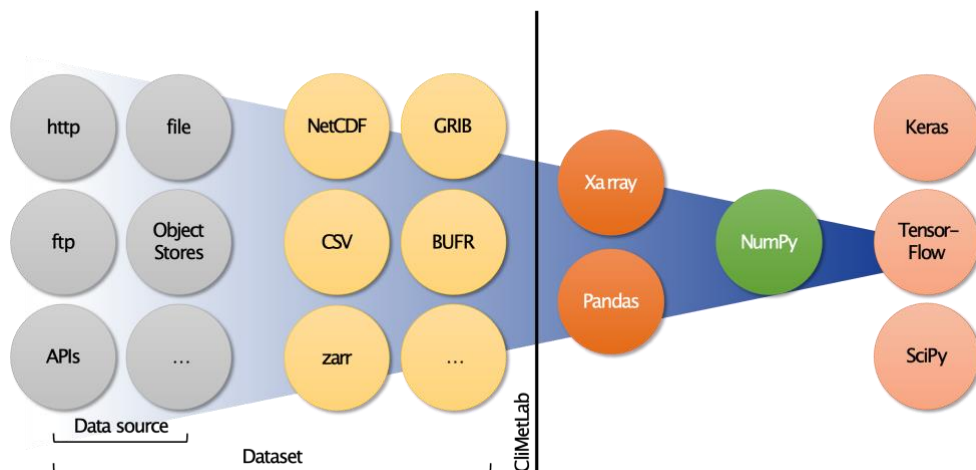


Imagen 6: Como CliMetlab simplifica en tratamiento de datos

CliMetLab presenta dos conceptos: Fuente de datos y Conjunto de datos, para hacer esto. Consiste en una variedad de técnicas de acceso, que incluyen la lectura de archivos, la descarga desde sitios web y la utilización de APIs, y están representadas por las fuentes de datos.

- **Fuente de datos:** Una fuente de datos es un objeto que nos da acceso a los datos y que se crea al llamar `cml.load_source(name, *args, **kwargs)` Siendo `cml` la referencia a la librería de `climetlab`. Este objeto nos proporciona métodos útiles para manejar los datos como `to_xarray()`, `to_pandas()` o `to_numpy()`.

Tipos de fuentes de datos disponibles:

- 'file', archivo. La fuente de datos es un archivo.
 - 'url' o 'url-pattern'. La fuente obtiene los datos de una url o lista de url.
 - 'cds'. La fuente de datos es Copernicus Data Store.
 - 'mars'. La fuente de datos es el Meteorological Archival and Retrieval System at ECMWF.
 - 'multi'. Múltiples fuentes simultáneamente.
 - 'zenodo'. Se cargan los datos de Zenodo.
 - 'indexed_urls'. Aún experimental, carga los datos de una url que contiene datos de tipo GRIB haciendo descargas parciales.
- **Conjuntos de datos:** Un Conjunto de datos es un objeto creado utilizando `cml.load_dataset(nombre, arg1, arg2=..., ...)` con el nombre y los argumentos apropiados, que proporciona acceso a un conjunto de datos bien definido que proporciona código junto con los datos. También proporciona metadatos y funcionalidades adicionales. El nombre identifica de manera única el conjunto de datos y los argumentos se pueden utilizar para especificar un subconjunto del conjunto de datos. Estos son los datos se pueden acceder utilizando métodos como `to_xarray()`, `to_pandas()`... Los metadatos relevantes se adjuntan

directamente al conjunto de datos para proporcionar información adicional.

El objeto Conjunto de datos es una instancia de una clase de Python en la que los usuarios del complemento pueden compartir código adicional.

- **Representación gráfica:** Además, es importante destacar que CliMetLab utiliza Magics como parte de su funcionalidad. Magics es el software de representación meteorológica desarrollado por ECMWF, el cual ofrece diversas capacidades para la visualización de datos meteorológicos. CliMetLab integra esta herramienta como parte de su infraestructura, lo que permite a los usuarios generar visualizaciones avanzadas y personalizadas de los datos climáticos y meteorológicos que están analizando.

Magics ofrece una amplia gama de opciones de representación, incluyendo la creación de contornos, campos de viento, observaciones, imágenes de satélite, símbolos, texto, ejes y gráficos. Permite trabajar con diferentes formatos de datos, como GRIB, NetCDF, BUFR y datos en formato de base de datos ODB, lo que brinda flexibilidad en la manipulación y visualización de la información meteorológica.

Al integrar Magics en CliMetLab, los usuarios tienen acceso a una potente herramienta de visualización que les permite explorar y entender mejor los datos climáticos y meteorológicos. Esto facilita el análisis de tendencias, la identificación de patrones y la toma de decisiones informadas en una amplia gama de aplicaciones relacionadas con el clima y el tiempo.

Desafíos y limitaciones en el uso de ML

Este enfoque no está exento de desafíos y limitaciones que deben abordarse para maximizar su efectividad y utilidad en el campo meteorológico. En este sentido, es crucial comprender y enfrentar los obstáculos que surgen al aplicar técnicas de ML a datos meteorológicos.

1. Complejidad y diversidad de los datos:

Los datos meteorológicos pueden ser complejos debido a la gran cantidad de variables que se deben considerar, como la temperatura, la humedad, la presión atmosférica, la velocidad y dirección del viento, entre otros.

Además, estos datos pueden ser heterogéneos en términos de formatos, resoluciones espaciales y temporales, lo que dificulta su integración y procesamiento uniforme.

2. Escasez de datos de calidad:

Aunque se han recopilado datos meteorológicos durante décadas, la disponibilidad de datos de calidad puede variar según la región y la fuente de datos.

Otra posible limitación viene del tamaño y la calidad, con valores indefinidos, errores de medición o inconsistencias que pueden afectar la capacidad de los modelos de ML para aprender patrones precisos.

3. Sensibilidad a condiciones extremas y eventos raros:

Los eventos meteorológicos extremos, como son los CTs, son relativamente raros en comparación con condiciones meteorológicas más típicas.

Debido a su baja frecuencia en los datos de entrenamiento, los modelos de ML pueden tener dificultades para capturar y generalizar adecuadamente los patrones asociados con estos eventos extremos.

4. **Interpretabilidad:**

Algunos modelos de ML, especialmente las redes neuronales profundas y otros modelos de aprendizaje profundo, pueden ser difíciles de interpretar y entender. La falta de transparencia y capacidad de explicación en estos modelos puede plantear desafíos en términos de confianza en los resultados y comprensión de cómo se toman las decisiones.

Capítulo 4 Planificación

4.1. Metodología

La metodología que se ha seguido durante este proyecto es una metodología ágil, un enfoque iterativo e incremental para el desarrollo del proyecto que se centra en la colaboración, la adaptabilidad y la entrega continua de valor.

1. Planificación inicial:

- Se realiza una reunión inicial para definir el alcance general del proyecto y establecer los objetivos y requisitos iniciales.
- Se crea un backlog funcionalidades que se desean desarrollar durante el proyecto, priorizando los elementos más importantes y los que se deben abordar primero.

2. Iteraciones o sprints:

- El proyecto se divide en iteraciones o sprints (2-4 semanas), durante los cuales se desarrollan y entregan los avances conseguidos hasta ese momento.
- En el inicio de cada sprint, se lleva a cabo una reunión de planificación donde se seleccionan las tareas del backlog que se abordarán en ese sprint teniendo en cuenta si en el sprint anterior se han quedado alguna tarea sin finalizar.
- Durante el sprint, el equipo trabaja en el desarrollo de las tareas seleccionadas, con intercambio de correos o mensajes por el chat del grupo de trabajo para mantenerse al tanto del progreso y abordar cualquier problema.
- Al final de cada sprint, se lleva a cabo una revisión del trabajo realizado y una identificación de las áreas de mejora.

3. Colaboración y comunicación:

- Se crea un grupo de trabajo en Teams de manera que hay un canal de comunicación abierto y constante mediante el chat y correos cuando las cuestiones a abordar son más extensas, también se utilizan las reuniones mediante videollamada de Teams cuando sea necesario.
- Como herramienta de seguimiento de tareas, para mantener un seguimiento claro del progreso y las prioridades, se ha utilizado Trello.

4. Flexibilidad y adaptabilidad:

- Como se trata de un trabajo de investigación, este punto es crucial a la hora de organizarse porque no siempre se consigue la total compleción de las tareas en el tiempo que se estimó inicialmente.
- El feedback regular entre alumno y tutores se incorporan continuamente al proceso de investigación y desarrollo con el fin de que el resultado final sea lo más fiable posible.

5. Entrega continua de valor:

- Se entregan informes con los resultados obtenidos con el fin de que los tutores evalúen si el rumbo que se ha tomado es el correcto o si hay que corregir o cambiar ciertos parámetros de la investigación. Los canales usados serán vía correo electrónico si son entregas no previstas y el directorio creado en Teams para las entregas programadas.

4.2. Proceso de desarrollo

En la parte práctica de la creación del modelo se ha seguido un modelo en cascada, los resultados obtenidos en el Sprint anterior se han ido usando para enriquecer al siguiente a medida que se iba avanzando en el proyecto y se iban sacando conclusiones. Cada Sprint se compondrá de diferentes historias de usuario que a su vez se segmentan en tareas

4.2.1. Sprint 1

- **Kick-off del proyecto**
 - Reunión de inicio
 - Reunión sobre los tipos de datos con Carlos
- **Estado del arte**
 - Análisis modelos actuales
 - Análisis predicciones actuales
- **Primera aproximación al modelo**
 - Preparar entorno de trabajo
 - Recrear modelo de ejemplo de ClimeLab
- **Inicio memoria**
 - Redacción introducción de memoria
- **Sprint review**
 - Sprint Review (El alumno revisa que tareas quedan pendientes y analiza los resultados para exponer en la retrospectiva)
 - Reunión retrospectiva
- **Refinamiento siguiente Sprint**
 - Análisis de los resultados
- **Planificación siguiente Sprint**
 - Reunión planning

4.2.2. Sprint 2

- **Estado del arte**
 - Reunión sobre meteorología con Daniel
- **Creación cuaderno**
 - Plantear caso de estudio inicial
 - Inicio de cuaderno
- **Preparación de dataset**
 - Descargar los datos de ERA
- **Redacción memoria**
 - Modificar introducción
 - Redacción Introducción a los CTs
- **Sprint review**

- Sprint Review (El alumno revisa que tareas quedan pendientes y analiza los resultados para exponer en la retrospectiva)
- Reunión retrospectiva
- **Refinamiento siguiente Sprint**
 - Análisis de los resultados
- **Planificación siguiente Sprint**
 - Reunión planning

4.2.3. Sprint 3

- **Implementación cuaderno**
 - Notebook de clasificación
- **Refinamiento de dataset**
 - Criba de datos (3 meses de 2020)
- **Redacción memoria**
 - Ampliar Introducción a los CTs
 - Inicio de redacción apartado Planificación
 - Apartado Bibliografía
- **Sprint review**
 - Sprint Review (El alumno revisa que tareas quedan pendientes y analiza los resultados para exponer en la retrospectiva)
 - Modificar Sprint 2
 - Reunión retrospectiva
- **Refinamiento siguiente Sprint**
 - Análisis de los resultados
- **Planificación siguiente Sprint**
 - Reunión planning

4.2.4. Sprint 4

- **Implementación cuaderno**
 - Notebook de clasificación (tomando presión a nivel 500)
- **Redacción memoria**
 - Modificar Introducción a los CTs
 - Completar apartado Planificación
 - Actualizar Bibliografía
- **Sprint review**
 - Sprint Review (El alumno revisa que tareas quedan pendientes y analiza los resultados para exponer en la retrospectiva)
 - Modificar Sprint 2 y 3
 - Reunión retrospectiva
- **Refinamiento siguiente Sprint**
 - Análisis de los resultados
- **Planificación siguiente Sprint**
 - Reunión planning

4.2.5. Sprint 5

- **Implementación cuaderno**
 - Notebook de clasificación (tomando presión a nivel 925)
- **Redacción memoria**
 - Modificar apartado Planificación
 - Actualizar Bibliografía
- **Sprint review**
 - Sprint Review (El alumno revisa que tareas quedan pendientes y analiza los resultados para exponer en la retrospectiva)
 - Modificar Sprint 4
 - Reunión retrospectiva
- **Refinamiento siguiente Sprint**
 - Análisis de los resultados
- **Planificación siguiente Sprint**
 - Reunión planning

4.2.6. Sprint 6

- **Implementación cuaderno**
 - Notebook de clasificación (diferencia geopotencial)
 - Entrenamiento modelo
- **Sprint review**
 - Sprint Review (El alumno revisa que tareas quedan pendientes y analiza los resultados para exponer en la retrospectiva)
 - Modificar Sprint 5
 - Reunión retrospectiva
- **Refinamiento siguiente Sprint**
 - Análisis de los resultados
- **Planificación siguiente Sprint**
 - Reunión planning

4.2.5. Sprint 7

- **Ampliación dataset**
 - Descargar datos HURDAT (centros de ciclones)
 - Crear dataset final con datos de ERA5
- **Redacción memoria**
 - Redactar Cuadernos de Aprendizaje
 - Ampliar Bibliografía
- **Sprint review**
 - Sprint Review (El alumno revisa que tareas quedan pendientes y analiza los resultados para exponer en la retrospectiva)
 - Reunión retrospectiva
- **Refinamiento siguiente Sprint**

- Análisis de los resultados
- **Planificación siguiente Sprint**
 - Reunión planning

4.2.5. Sprint 8

- **Implementación cuaderno caso final**
 - Notebook caso final
 - Entrenamiento modelo
- **Redacción memoria**
 - Modificar Cuadernos de Aprendizaje
 - Redacción Contexto del trabajo
 - Ampliar Bibliografía
- **Sprint review**
 - Sprint Review (El alumno revisa que tareas quedan pendientes y analiza los resultados para exponer en la retrospectiva)
 - Reunión retrospectiva
- **Refinamiento siguiente Sprint**
 - Análisis de los resultados
- **Planificación siguiente Sprint**
 - Reunión planning

4.2.6. Sprint 9

- **Cuaderno Caso final**
 - Modificación Notebook caso final
- **Redacción memoria**
 - Corregir Sprint 5
 - Redacción Caso de estudio
 - Actualizar Bibliografía
- **Sprint review**
 - Sprint Review (El alumno analiza los resultados para exponer en la retrospectiva)
 - Modificar Sprint 9
 - Reunión retrospectiva
- **Refinamiento siguiente Sprint**
 - Análisis de los resultados

4.2.6. Sprint 10

- **Redacción memoria**
 - Empezar redacción Experimentación y evaluación
 - Actualizar Bibliografía
 - Correcciones en capítulos anteriores
- **Sprint review**

- Sprint Review (El alumno analiza los resultados para exponer en la retrospectiva)
- Reunión retrospectiva
- **Refinamiento siguiente Sprint**
 - Reunión planning

4.2.6. Sprint 11

- **Redacción memoria**
 - Continuar redacción Experimentación y evaluación
 - Actualizar Bibliografía
 - Correcciones en capítulos anteriores
- **Sprint review**
 - Sprint Review (El alumno analiza los resultados para exponer en la retrospectiva)
 - Reunión retrospectiva

4.3. Diagrama de Gantt

A continuación, se mostrará el diagrama de Gantt (ver Figura). Las semanas de trabajo el primer año consistían en jornadas de lunes a viernes y a partir de octubre de 2023 abarcan de viernes a domingo y los miércoles, por disponibilidad horaria de trabajo. Además, no se han tenido en cuenta aquellas semanas en las que no se ha trabajado por exámenes, vacaciones y otros eventos que se citan en el apartado de obstáculos.

4.4. Obstáculos

A lo largo del proyecto han ido surgiendo diferentes inconvenientes que han interferido significativamente en el curso planificado del proyecto. Aquí vamos a nombrar aquellos que han supuesto un mayor retraso.

La transición de los servicios de datos de Copernicus desde Reading, Reino Unido, a Bolonia, Italia, implicó varias fases con periodos temporales de inactividad. Las fechas clave en las que los servicios estuvieron no disponibles incluyen:

- **8 de septiembre al 11 de noviembre de 2022:** Durante el periodo principal de transición, hubo inactividades intermitentes mientras se movían y reconfiguraban los sistemas. Específicamente, la mitad del archivo estuvo accesible el 28 de octubre, el 75% el 3 de noviembre y el archivo completo el 11 de noviembre de 2022.

Esto supuso que no se pudiese trabajar en la parte de desarrollo ya que los datos de los datasets se obtenían y almacenaban en cache, de modo que cada vez que se necesitaban descargar los datos, no estaban disponibles y se perdían los anteriores.

En los periodos de exámenes, el trabajo efectivo también se vio afectado debido a una mayor dedicación al resto de carga académica del alumno.

En octubre de 2022 el alumno comenzó a trabajar por lo que la disponibilidad del alumno disminuyó considerablemente.

En marzo de 2023 el ordenador donde se elaboraba el desarrollo del proyecto se estropea, en abril se repone y hay que volver a configurar el entorno de trabajo y migrar todo lo desarrollado hasta ese momento de la nube donde estaban las copias de seguridad.

4.5. Tiempo total dedicado

El proyecto comenzó el 5 de septiembre de 2022 y finalizó en julio de 2024. A lo largo del desarrollo del TFG, se presentaron varios periodos de inactividad que afectaron el curso del proyecto. A continuación, se detallan estos periodos de inactividad y cómo afectaron el tiempo total dedicado.

Fechas de inicio y fin:

- **Inicio del proyecto:** 5 de septiembre de 2022
- **Fin del proyecto:** julio de 2024

Periodos de inactividad:

1. **8 de septiembre al 11 de noviembre de 2022:** Inactividades intermitentes durante 64 días. Asumiremos una inactividad efectiva de 32 días, considerando la mitad del periodo.
2. **Periodos de exámenes:** Asumiremos 2 semanas de inactividad total por cada semestre. Desde septiembre de 2022 hasta julio de 2024 hay 4 semestres, resultando en 8 semanas (56 días) de inactividad.

3. **Octubre de 2022:** Inicio de trabajo del alumno, afectando su disponibilidad. Asumiremos una reducción de disponibilidad equivalente a 2 días por semana durante 20 meses, resultando en 160 días.
4. **Marzo de 2023:** Ordenador estropeado, afectando el trabajo por 1 mes (30 días).

Tiempo total efectivo dedicado:

- **Duración total:** 21 meses
- **Inactividad:** 9 meses
- **Tiempo efectivo:** 21 meses – 9 meses = 12 meses

Por lo tanto, el tiempo total dedicado para completar el TFG, considerando las semanas efectivas de trabajo y los periodos de inactividad mencionados, ha sido de aproximadamente 12 meses.

4.6. Herramientas utilizadas

En esta sección se entra más en detalle en las tecnologías específicas que se han empleado para desarrollar la aplicación: lenguajes de programación empleados, gestor de bases de datos, herramientas usadas en la planificación y en la generación de documentación, sistema operativo, etc. Asimismo, se exponen los criterios por los que se han seleccionado las herramientas elegidas, de entre otras posibles, haciendo referencia a posibles ventajas e inconvenientes.

Las herramientas que se han planteado para este proyecto han sido:

- Microsoft Teams, plataforma de colaboración y comunicación desarrollada por Microsoft, ofrece varias funcionalidades diseñadas para mejorar la productividad y la colaboración en equipos de trabajo. Se va a usar como modo de comunicación y también como repositorio para los entregables y documentación compartida.
 - Espacio de trabajo compartido, compuesto por varias carpetas:
 - Directorio general, contiene todos los demás directorios del proyecto y los entregables que se van a ir elaborando al final de los sprints.
 - Directorio de la memoria, se almacena las versiones de la memoria del proyecto.
 - Directorio de recursos, se almacenan los recursos compartidos durante el desarrollo del proyecto como pueden ser publicaciones

de interés, enlaces...

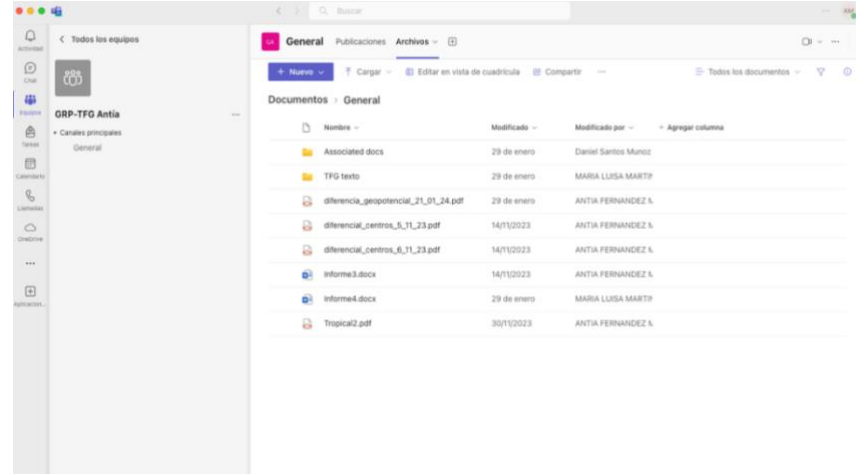


Imagen 8: Grupo de Teams

- Canal de comunicación, este espacio cuenta con un chat donde pueden participar todos los miembros del equipo y donde se realizarán las reuniones mediante videollamadas.

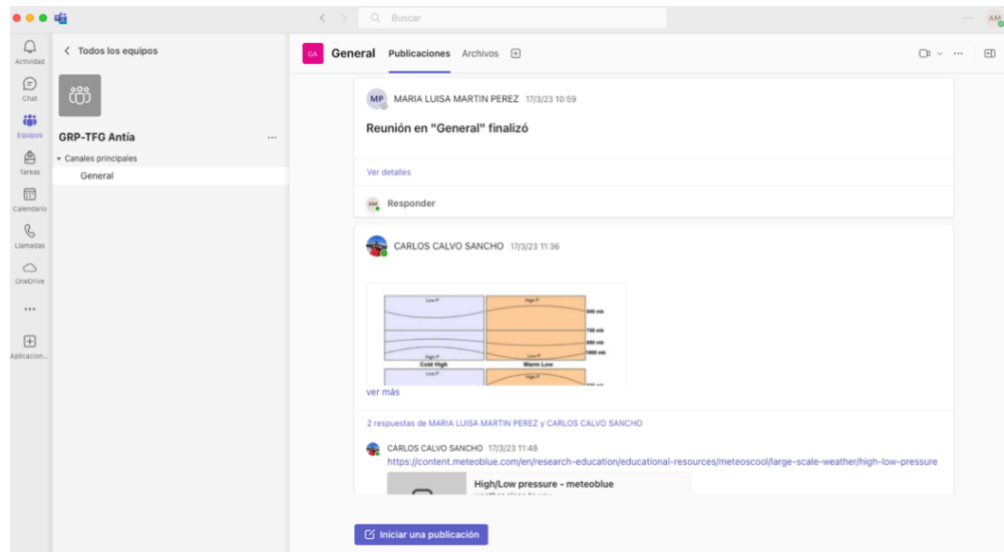


Imagen 9: Canal de Teams

- Uso de Microsoft Word para la redacción por sugerencia de los tutores ya que facilita la corrección continua y la posibilidad de dejar comentarios en las secciones deseadas.
- Trello como herramienta de organización, se compone de un tablero donde se crean tarjetas con objetivos y tareas. Dentro del tablero hay una disposición por columnas:
 - **Backlog**, tareas y objetivos que aún no han sido planificados en ningún sprint. Bien por falta de definición o por falta de tiempo en el sprint actual.
 - **Sprint backlog**, son los objetivos del sprint actual.

- **To do**, tareas que se han planificado para el sprint actual que aún no han sido abordadas.
- **In progress**, tareas que están en progreso en el momento.
- **Complete**, tareas que han sido completadas.

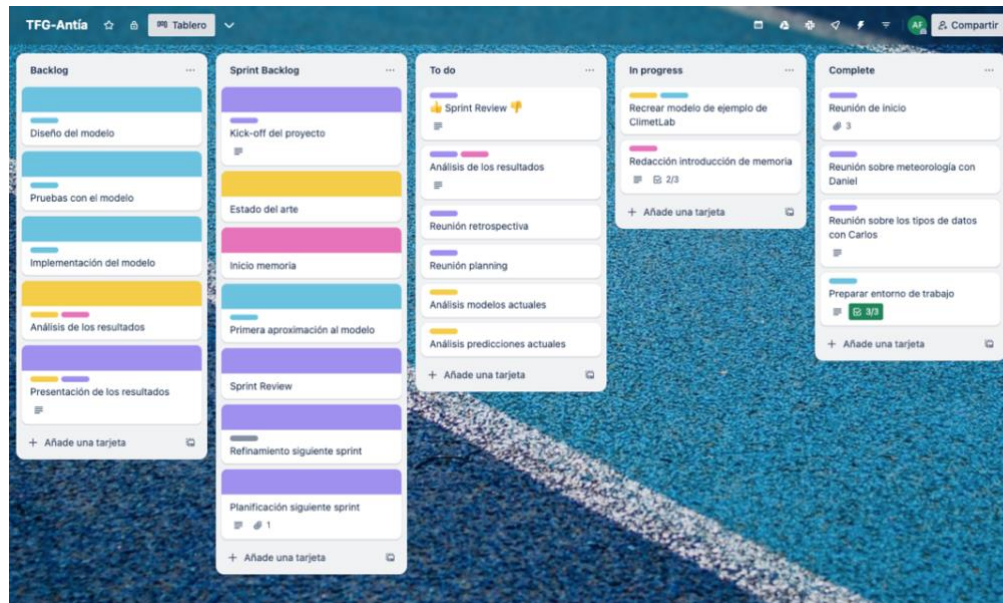


Imagen 10: Tablero de Trello

En cada tarea se puede añadir título, descripción, fechas, checklist... Se usan dependiendo de la finalidad de dicha tarea, como ejemplo:

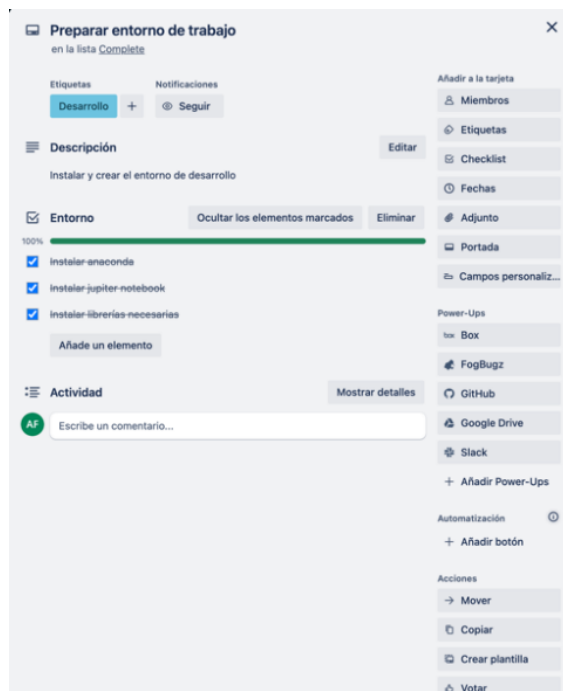


Imagen 11: Detalle de tarea en Trello

- Anaconda-navigator con Jupiter Notebook como entornos de desarrollo de los modelos y las pruebas prácticas, está totalmente integrado con python y puedes ir ejecutando y viendo los resultados de forma progresiva.

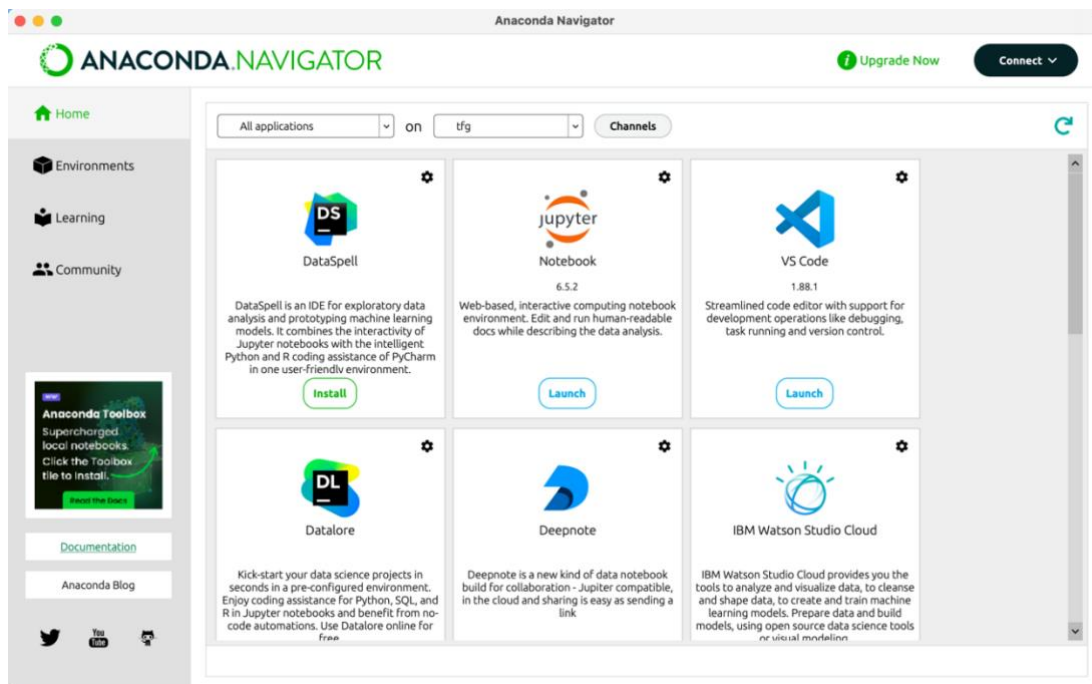


Imagen 12: Anaconda navigator

- Python como lenguaje de desarrollo, destacando ClimeLab, librería anteriormente presentada. Actualmente aún sigue en desarrollo, pero ha sido de gran utilidad para el tratamiento de los datos en formato GRIB.
- TensorFlow, para desarrollar y entrenar el modelo, librería compatible con python.
- Keras, para construir el modelo, compatible con Tensorflow y python.
- Numpy, para manejar los datos para el programa, librería compatible con python.
- Atom como entorno de programación; es un entorno intuitivo con muchas librerías útiles y es hasta ahora es el entorno en el que más cómoda me encuentro trabajando en ios.

4.7. Presupuestos

En este apartado incluiremos los recursos técnicos, que abarca los materiales y equipos tanto de hardware como de software necesarios para llevar a cabo el proyecto y también los gastos en recursos humanos, es decir, los costos asociados a los empleados.

Recursos técnicos

Empezamos por los recursos gratuitos. Estos son herramientas de software con licencia gratuita, como Word, Trello, Microsoft Teams o Jupyter Notebook.

En cuanto a hardware, el proyecto comenzó con el uso de un MacBook Air de 13" del año 2014. Este dispositivo, con una vida útil estimada de 7 años, tuvo un costo aproximado de 800€. A lo largo del resto del proyecto, se empleó un MacBook Pro Max M1 con 1TB de almacenamiento, cuyo costo es de aproximadamente 2500€ y con una vida útil estimada de 5 años. Posteriormente, calcularemos el costo de amortización asociado al período de trabajo del proyecto, que es de 12 meses, considerando el tiempo que se usó cada dispositivo.

Además, se requiere una conexión a Internet de alta calidad. Su uso principal será para realizar reuniones a través de videollamadas usando Microsoft Teams y para descargar los conjuntos de datos que se utilizarán, recordemos que vamos a trabajar cacheando los datasets por lo es muy importante la conexión. También será necesario para consultar artículos, revistas científicas e información necesaria. En total serán 12 meses de acceso a Internet con una tarifa de 600Mb de fibra simétrica por 39€ al mes.

- **MacBook Air** durante 4 meses: $800\text{€ de coste} \times 4,76\% \text{ de uso} = 38,08\text{€}$
- **MacBook Pro** durante 8 meses: $2500\text{€ de coste} \times 13,33\% \text{ de uso} = 333,36\text{€}$
- **Internet:** $39\text{€ al mes} \times 1200\% \text{ de uso} = 468\text{€}$

Sumando estos costos, el total destinado a recursos técnicos es: $38,08\text{€} + 333,36\text{€} + 468\text{€} = 839,44\text{€}$.

Recursos humanos

Para nuestro proyecto, estimamos los salarios de diferentes roles: analista, desarrollador y un climatólogo. Asumimos que cada trabajador realiza un total de 40 horas a la semana.

El salario promedio de un analista es de 30000€ anuales, lo que son 2500€ al mes y 15,62€ por hora.

Un climatólogo tiene un salario anual promedio de 50000€, que corresponde a 4,167€ al mes y 24,04€ por hora.

Por último, un desarrollador gana aproximadamente 25000€ al año, es decir, 2080€ al mes y 13€ por hora.

Sumamos también los gastos a la Seguridad Social, que representan un 30% de su salario bruto.

El Trabajo Fin de Grado está planeado para desarrollarse en aproximadamente 300 horas y la duración ha sido de 12 meses. Calculando el costo por hora de cada rol, tenemos:

- **Analista:** 15,62€ por hora
- **Climatólogo:** 24,04€ por hora
- **Desarrollador:** 13€ por hora

Para un total de 300 horas, los costos son:

- **Analista:** $170 \text{ horas} \times 15,62\text{€} = 2655,4\text{€}$

- **Climatólogo:** 50 horas * 24,04€ = 1202€
- **Desarrollador:** 80 horas * 13€ = 1040€

Sumando estos costos, el total de salarios antes de los gastos de Seguridad Social es: 2655,4€ + 1202€ + 1040€ = 4897,4€.

Los gastos de Seguridad Social, que son un 30% del salario bruto, serían:

- **Analista:** 170 horas * 15,62€ * 0.30 = 796,62€
- **Climatólogo:** 50 horas * 24,04€ * 0.30 = 360,6€
- **Desarrollador:** 80 horas * 13€ * 0.30 = 320€

Gastos de Seguridad Social totales: 796,62€ + 360,6€ + 320€ = 1477,22€

Por lo tanto, el presupuesto total de recursos humanos es:

	Salario por hora	Horas totales	Salario total sin SS	Coste SS	Total
Analista	15,62€	170	2655,4€	796,62€	3452,02€
Climatólogo	24,04€	50	1202€	360,6€	1562,6€
Desarrollador	13€	80	1040€	320€	1360€
TOTAL					6374,62€

Tabla 1: Presupuesto recursos humanos

Parte II Desarrollo de la propuesta y resultados

Capítulo 5 Cuadernos de aprendizaje

En esta sección trataremos el caso de estudio objeto de este proyecto. Presentaremos también unos pequeños informes que se fueron redactando a medida que se avanzaba donde se reflejan los avances, obstáculos y conclusiones.

5.1. Caso de estudio elegido

Como ya mencionamos en el contexto, la introducción de modelos basados en IA en la predicción del tiempo supone un avance significativo en la meteorología. Estos modelos no solo complementan a los tradicionales, sino que también ofrecen mejoras en términos de velocidad, precisión y capacidad de adaptación. La integración de ambas tecnologías puede llevar a predicciones meteorológicas más fiables y oportunas, beneficiando a sectores como la agricultura, la aviación, y la gestión de desastres naturales...

5.2. Conjuntos de datos utilizados

5.2.1 Reanálisis ERA5

En esta sección, abordaremos el uso del conjunto de datos (*dataset*) **Reanalysis ERA5**; En particular, se utiliza la variable presión a nivel del mar **Pressure Levels** (ERA5-PL). Este dataset fue desarrollado por el ECMWF y está disponible desde 2017. Forma parte de una iniciativa mundial para mejorar la comprensión de las condiciones atmosféricas, convirtiéndose en una herramienta fundamental para los investigadores y científicos en meteorología y estudios climáticos.

Los datos de ERA5-PL se derivan de la asimilación de observaciones provenientes de diversas fuentes, como satélites, estaciones terrestres, aviones y boyas. Estas observaciones se integran en un modelo de predicción numérica que abarca la atmósfera global, permitiendo la recopilación y procesamiento de un registro detallado y coherente de variables atmosféricas como temperatura, presión, viento y humedad a distintos niveles de presión. Este conjunto de datos cubre un amplio periodo desde 1950 hasta la actualidad, ofreciendo resolución temporal horaria y espacial de 30 km [5].

Para una manipulación y análisis eficientes, los datos se organizan en archivos NetCDF. Estos archivos contienen no solo las variables atmosféricas de interés, sino también metadatos que describen la calidad de los datos, su fuente y la metodología de recopilación. La inclusión de anotaciones detalladas facilita el uso del dataset en

investigaciones específicas, como el análisis de fenómenos climáticos extremos o la validación de modelos de predicción climática [7].

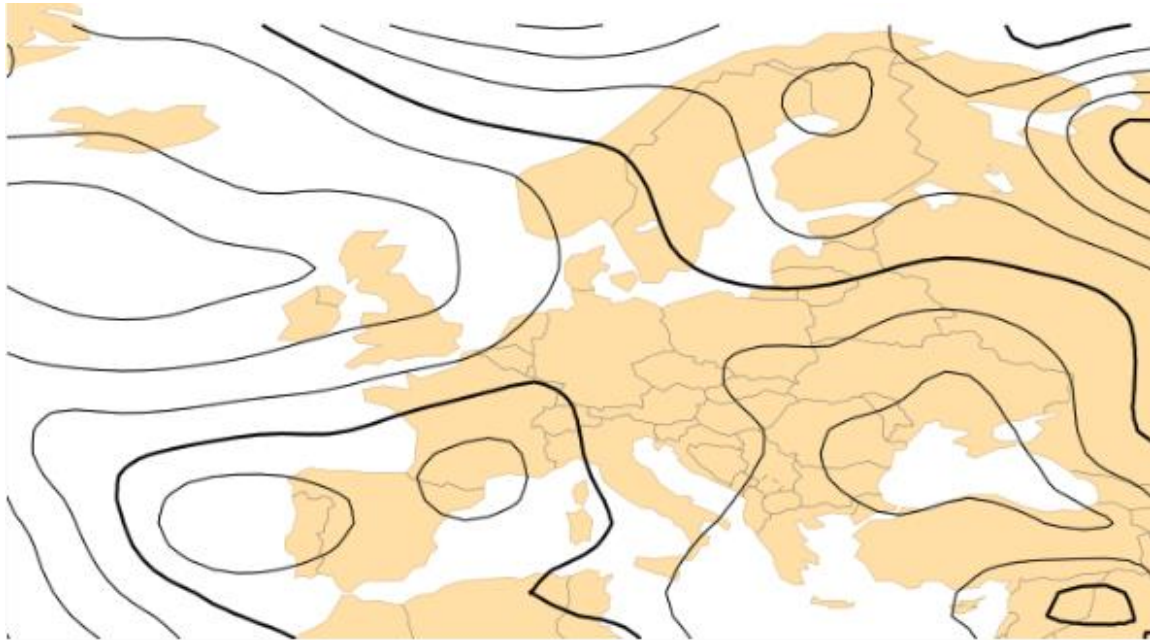


Imagen 13: Ilustración de una distribución de presión a nivel del mar

5.2.2 Base de Datos de Huracanes del Atlántico HURDAT

A continuación, exploraremos el conjunto de datos HURDAT (Atlantic Hurricane Database) [6] [8] [10], que recopila información detallada sobre CTs y subtropicales que han afectado la cuenca del Atlántico Norte, incluyendo el Golfo de México y el Mar Caribe. Esta base de datos, mantenida por el National Hurricane Center (NHC) y la National Oceanic and Atmospheric Administration (NOAA).

HURDAT proporciona un registro continuo y detallado de la trayectoria y características de los CTs desde 1851 hasta la actualidad, con datos recopilados a intervalos de 6h. La base de datos incluye información sobre la posición geográfica (latitud y longitud), la velocidad del viento, la presión mínima en el centro del sistema y la categoría de intensidad de cada ciclón en distintos momentos de su ciclo de vida [1].

La recopilación de datos para cada ciclón proviene de una variedad de fuentes, como observaciones directas, datos satelitales, informes de aeronaves y estaciones meteorológicas terrestres. Esta información se procesa y valida para garantizar su coherencia y precisión a lo largo del tiempo. Inicialmente, la recolección de datos se basaba principalmente en informes de barcos y estaciones costeras, pero con el avance de la tecnología, la precisión y cantidad de datos han mejorado significativamente gracias al uso de satélites y aviones de reconocimiento [2].

Los datos se organizan en un formato accesible que permite su análisis y visualización mediante software de procesamiento de datos meteorológicos. Cada entrada en la base de datos incluye no solo la trayectoria y la intensidad del ciclón, sino también metadatos que describen las condiciones de recolección y las metodologías utilizadas. Las anotaciones

detalladas acompañan al conjunto de datos, facilitando su uso en investigaciones científicas, estudios climáticos y análisis de riesgos [3].



Imagen 14: Trayectoria o track del huracán Katrina

5.3. Datasets previos

Antes de llegar al caso de estudio final se han realizado distintas pruebas con distintos niveles de altitud. Antes de presentar estos datasets y los notebooks con los resultados vamos a presentar la obtención de los datos ya que es algo que tienen todos en común.

Se delimita la zona geográfica de aparición. Por proximidad, delimitamos la zona más próxima o aquella área de origen que puede llegar a afectar a España. El cuadrante elegido inicialmente es:

- $0 < \text{latitud} < 60$
- $-90 < \text{longitud} < 0$

Sobre esta zona, creamos el primer conjunto de datos. Partimos de un Excel, con un tamaño de 100 filas, donde cada fila corresponde a un fenómeno meteorológico que puede ser un CT o no. Este Excel tiene el siguiente aspecto:

Date	Lat	Lon	TC
2018103112	35.5	-52.5	1
2018103118	38	-50.75	0

Tabla 2: Formato del Excel con los datos de los centros

donde...

La columna "Date" es la fecha y hora

La columna "Lat" es la latitud del centro del ciclón

La columna "Lon" es la longitud del centro del ciclón

La columna "TC" es una clasificación binaria donde 0 quiere decir no CT y 1 quiere decir CT.

A partir de este Excel elaboramos una constante con los datos que necesitamos solicitar y que va a contener el dataset

```
SAMPLES = [  
    ("2018-10-31", "12", (35.5, -52.5), (1.0, 0.0)),  
    ("2018-10-31", "18", (38, -50.75), (0.0, 1.0)),  
]
```

donde de izquierda a derecha tenemos:

1. Fecha
2. Hora (Cada 6 horas: 0-6-12-18)
3. Centro del ciclón
4. Etiqueta (CiclónTropical, NoCiclónTropical) marcando con un 1.0 la opción que es y 0.0 la que no.

Proseguimos con la petición de datos a Copernicus:

```
source = load_source(  
    "cds",  
    "reanalysis-era5-pressure-levels",  
    variable="z",  
    level=level,
```

```

product_type="reanalysis",
area=[area[0]+10, area[1]-10, area[0] - 10, area[1] + 10],
date=date,
grid=[0.25, 0.25],
time=time,
)

```

Parámetros

1. **"cds"**: Indica que se está utilizando el servicio de datos de Copernicus (CDS).
2. **"reanalysis-era5-pressure-levels"**: Especifica el conjunto de datos de reanálisis de niveles de presión ERA5.
3. **variable="z"**: Define la variable a obtener, que en este caso es la altura geopotencial ("z"). Esta variable es comúnmente utilizada para representar la altura de una superficie de presión constante en la atmósfera, como 500 hPa.
4. **level=level**: Indica el nivel de presión en hectopascales (hPa) para el cual se están solicitando los datos, aquí no especificamos el valor de "level" ya que va a variar en los diferentes datasets que vamos a crear.
5. **product_type="reanalysis"**: Especifica que el tipo de producto solicitado es "reanálisis", que implica datos procesados y asimilados que proporcionan una visión coherente de la atmósfera a lo largo del tiempo.
6. **area=[area[0] + 10, area[1] - 10, area[0] - 10, area[1] + 10]**: Define el área geográfica para la cual se solicitan los datos. Los valores están en grados de latitud y longitud, especificando un rectángulo. Aquí, area contiene coordenadas de latitud y longitud de los centros de los fenómenos meteorológicos, y se está expandiendo en 10 grados alrededor de los puntos centrales, es decir estamos creado la ventana de estudio.
7. **date=date**: Especifica la fecha para la cuales se solicitan los datos. "date" es la variable previamente definida en SAMPLE.
8. **grid=[0.25, 0.25]**: Define la resolución espacial de la cuadrícula para los datos solicitados. Aquí, cada celda de la cuadrícula tiene un tamaño de 0.25 grados por 0.25 grados.

Equivalencia en Kilómetros:

- **Latitud**: Un grado de latitud es aproximadamente constante a 111 km en cualquier lugar de la Tierra.
- **Longitud**: La distancia correspondiente a un grado de longitud varía con la latitud, siendo máxima en el ecuador y disminuyendo a medida que se avanza hacia los polos.

Ejemplo de Cálculo en el Ecuador

En el ecuador, cada celda de 0.25 grados de latitud por 0.25 grados de longitud representa un área aproximadamente:

- Latitud: $0.25 * 111 \text{ km} = 27.75 \text{ km}$.
- Longitud: $0.25 * 111 \text{ km} = 27.75 \text{ km}$.

Por lo tanto, el área de cada celda en el ecuador es aproximadamente:

$$27.75 \text{ km} \times 27.75 \text{ km} \approx 770 \text{ km}^2$$

9. **time=time:** Indica la hora específica del día para la cual se desean los datos. “time” es una variable que se obtiene de SAMPLE que contiene la información horaria requerida.

5.3.1 Informe 1

En este primer caso de estudio trabajamos con level=500 hPa, que es aproximadamente a media troposfera (5000 m) y es un nivel clave para el análisis meteorológico.

```
[1] from tensorflow.keras.layers import Input, Dense, Flatten
    from tensorflow.keras.models import Sequential

Python

... 2023-05-15 22:07:43.896674: I tensorflow/core/platform/cpu_feature_guard.cc:193] This TensorFlow binary is optimized with oneAPI Deep Neural Network Library (oneDNN) to use the follow
To enable them in other operations, rebuild TensorFlow with the appropriate compiler flags.

[2] import climetlab as cml
    import keras
    import xarray

Python

tropical = cml.load_dataset("tropical") ...
Las salidas están contraídas ...

[4] tropical.to_xarray()

Python

... xarray.Dataset
> Dimensions: (number: 1, time: 1, step: 1, isobaricInhPa: 1, latitude: 121, longitude: 121)
> Coordinates:
  number          (number)          int64 0
  time            (time)            datetime64[ns] 2018-11-01T...
  step            (step)            timedelta64[ns] 00:00:00
  isobaricInhPa   (isobaricInhPa)   float64 500.0
  latitude        (latitude)        float64 58.5 58.25 5...
  longitude       (longitude)       float64 -62.0 -61.75...
  valid_time      (time, step)      datetime64[ns] ...
> Data variables:
  z               (number, time, step, isobaricInhPa, latitude, longitude) float32 ...
> Indexes: (6)
> Attributes:
```

Imagen 14: Muestra Notebook 1 - Carga de librerías y petición de dataset



Imagen 15: Muestra Notebook 2 - Muestra gráfica

```
[9] (x_train, y_train, f_train), (x_test, y_test, f_test) = tropical.load_data2(tropical2test_size=0.3, fields=True)
Python

[ ] model = Sequential()
model.add(Input(shape=x_train[0].shape))
model.add(Flatten())
model.add(Dense(64, activation="sigmoid"))
model.add(Dense(2, activation="softmax"))
Python

[11] model.compile(optimizer="adam", loss="categorical_crossentropy", metrics=["accuracy"])
print(model.summary())
Python

... Model: "sequential"
Layer (type) Output Shape Param #
-----
flatten (Flatten) (None, 14641) 0
dense (Dense) (None, 64) 937088
dense_1 (Dense) (None, 2) 130
-----
Total params: 937,218
Trainable params: 937,218
Non-trainable params: 0
None
```

Imagen 16: Muestra Notebook 3 - División conjunto muestra/test y creación del modelo

```
[12] h = model.fit(x_train, y_train, epochs=20, verbose=1)
Python

Las salidas están contróidas ...

[13] model.evaluate(x_test, y_test, verbose=1)
Python

... 2/2 [=====] - 0s 2ms/step - loss: 0.6930 - accuracy: 0.5085
... [0.6930063366889954, 0.508474588394165]

predicted = model.predict(x_test)

for p, f in zip(predicted, f_test):
    print(p)
    print(f)
    cm.plot_map(f, title=tropical.title(p))
Python

... 2/2 [=====] - 0s 2ms/step
... [0.5073007 0.4926993]
GridField(z, 500, 20180926, 1200, 0, 0)

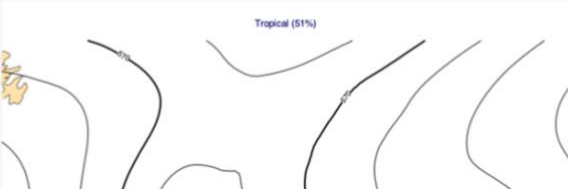
... 
```

Imagen 17: Muestra Notebook 4 - Entrenamiento del modelo, evaluación, predicción y muestra de predicción

De los 59 casos del test clasifica mal 12, es decir, acierta en un 79,66% de los casos en este estudio particular.

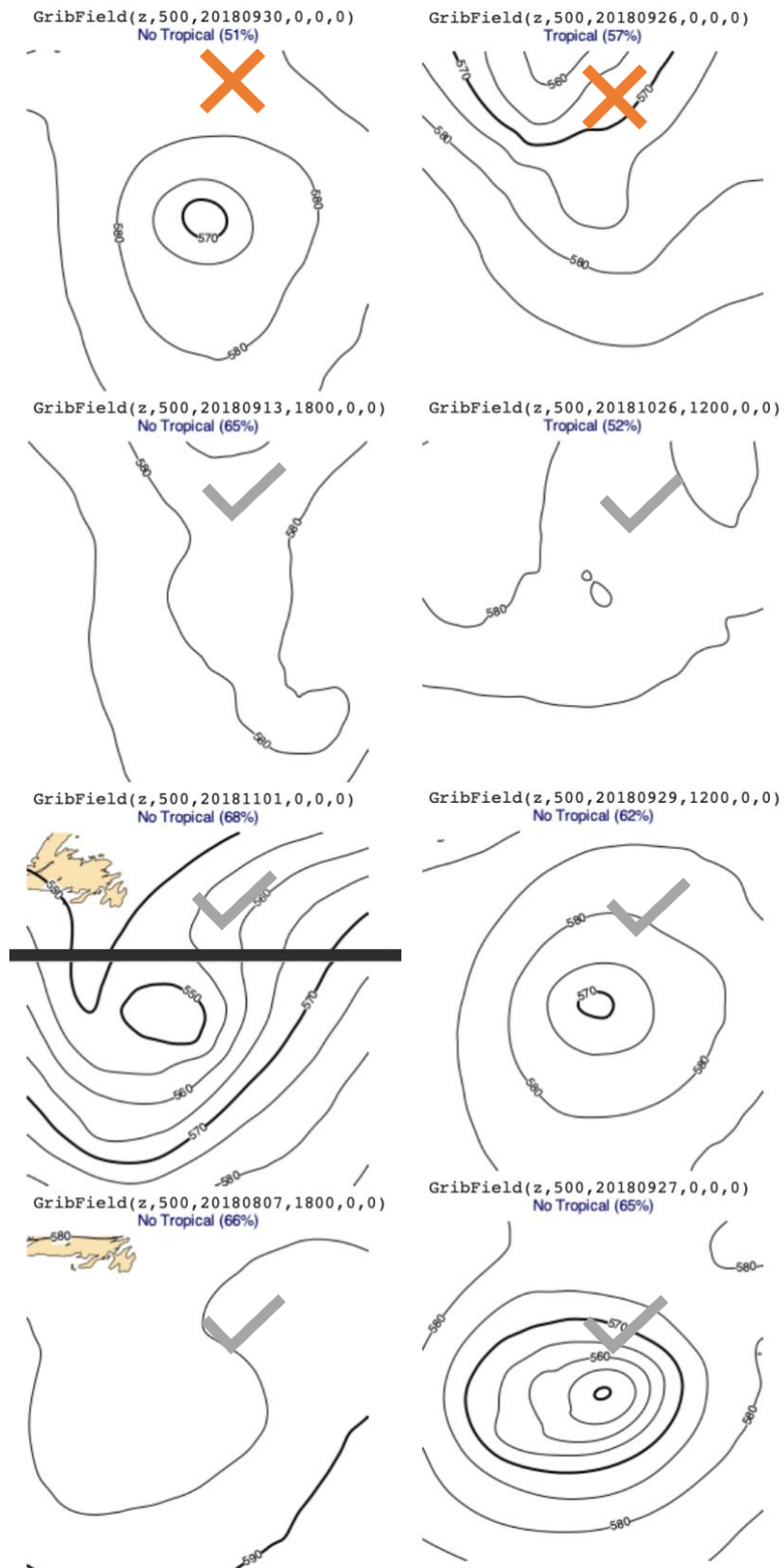


Imagen 18: Ejemplo de clasificación

Como los resultados no son los esperados, se prueba a entrenar el modelo con diferentes configuraciones, cambiando las épocas, el tamaño de las muestras... Pero esto sigue sin satisfacer las expectativas por lo que se decide probar otro enfoque.

5.3.2 Informe 2

En este contexto, la variable atmosférica, espesor geopotencial, emerge como una variable clave en la detección y el análisis de CTs. El espesor geopotencial, que se define como la diferencia en la altura geopotencial entre dos niveles de presión específicos en la atmósfera, proporciona una medida integral de la temperatura media de la capa de aire comprendida entre esos niveles [22].

El cálculo del espesor geopotencial se fundamenta en la **ecuación hipsométrica**, que establece la relación entre la diferencia de altura geopotencial (ΔZ) entre dos niveles de presión (p_1 y p_2) y la temperatura media de la capa de aire en cuestión:

$$\Delta Z = \frac{R_d * T_m}{g} \ln \frac{p_1}{p_2}$$

donde:

- ΔZ : **Espesor geopotencial** (m), es decir, la diferencia en la altura geopotencial entre dos niveles de presión.
- R_d : **Constante de los gases para el aire seco** ($\approx 287 \text{ J/kg} \cdot \text{K}$).
- T_m : **Temperatura media de la capa atmosférica** (K), determinada como el promedio ponderado entre los dos niveles de presión.
- g : **Aceleración debida a la gravedad** ($\approx 9.81 \text{ m/s}^2$).
- p_1 y p_2 : **Presiones en los dos niveles de interés** (Pa), con $p_1 > p_2$, es decir, p_1 corresponde al nivel inferior y p_2 al nivel superior.

Esta variable es especialmente útil en meteorología para identificar y caracterizar estructuras atmosféricas significativas, como los sistemas de baja presión y los núcleos cálidos característicos de los CTs.

Gráfico 1: Espesor Geopotencial en un Ciclón Tropical

Descripción: Este gráfico muestra un perfil vertical de la atmósfera, destacando los niveles de 500 hPa y 925 hPa. Se visualiza cómo el espesor geopotencial es mayor en el núcleo cálido de un CT.

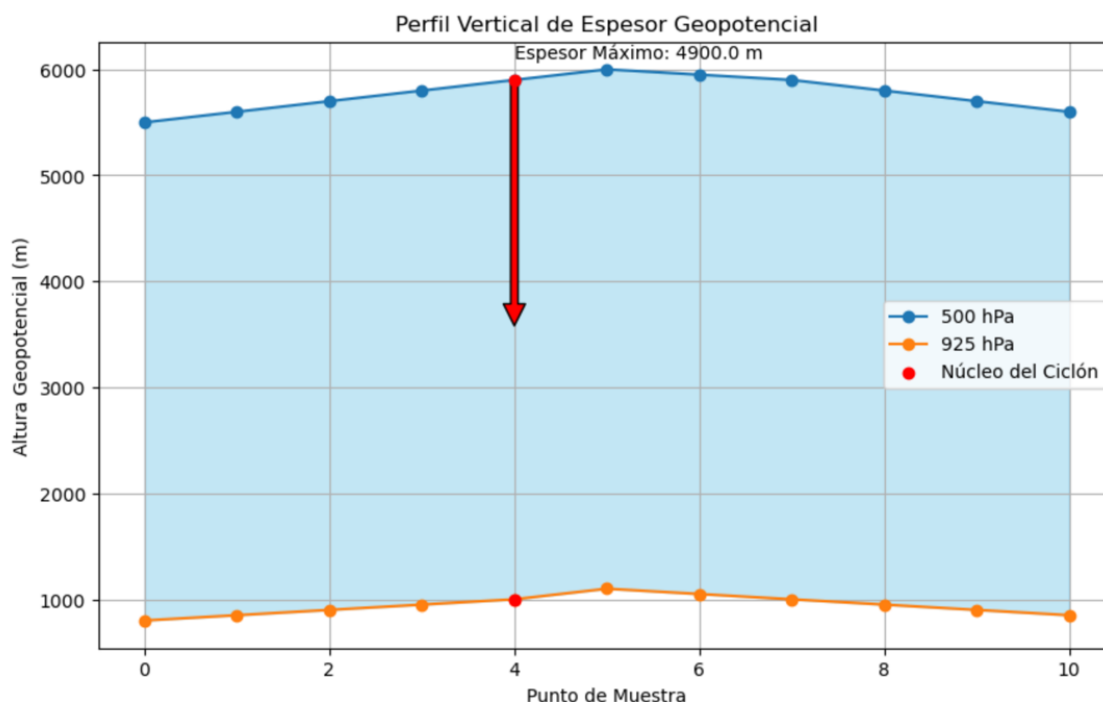


Imagen 19: Perfil vertical de la atmósfera

- **A:** Perfil atmosférico con el espesor geopotencial entre 500 hPa y 925 hPa marcado.
- **B:** Núcleo cálido en un CT donde el espesor geopotencial es mayor debido a las altas temperaturas.

La elección del espesor geopotencial entre los niveles de 500 hPa y 925 hPa no es arbitraria. Este rango de presión abarca una parte significativa de la troposfera, donde se desarrollan la mayoría de los fenómenos meteorológicos que afectan la superficie terrestre. Los niveles de 500 hPa y 925 hPa son particularmente relevantes porque:

1. Representación de la Temperatura Media:

- El espesor geopotencial entre estos dos niveles proporciona una indicación directa de la temperatura media en la capa de aire intermedia. Dado que los CTs son sistemas de baja presión con núcleos cálidos, un mayor espesor geopotencial es un indicador de la presencia de un núcleo cálido y, por lo tanto, de un posible CT.

Gráfico 2: Relación entre Espesor Geopotencial y Temperatura

Descripción: Un gráfico que muestra cómo el espesor geopotencial aumenta con la temperatura media entre dos niveles de presión. Esto ayuda a ilustrar por qué el espesor geopotencial es mayor en regiones con núcleos cálidos, como en los CTs.

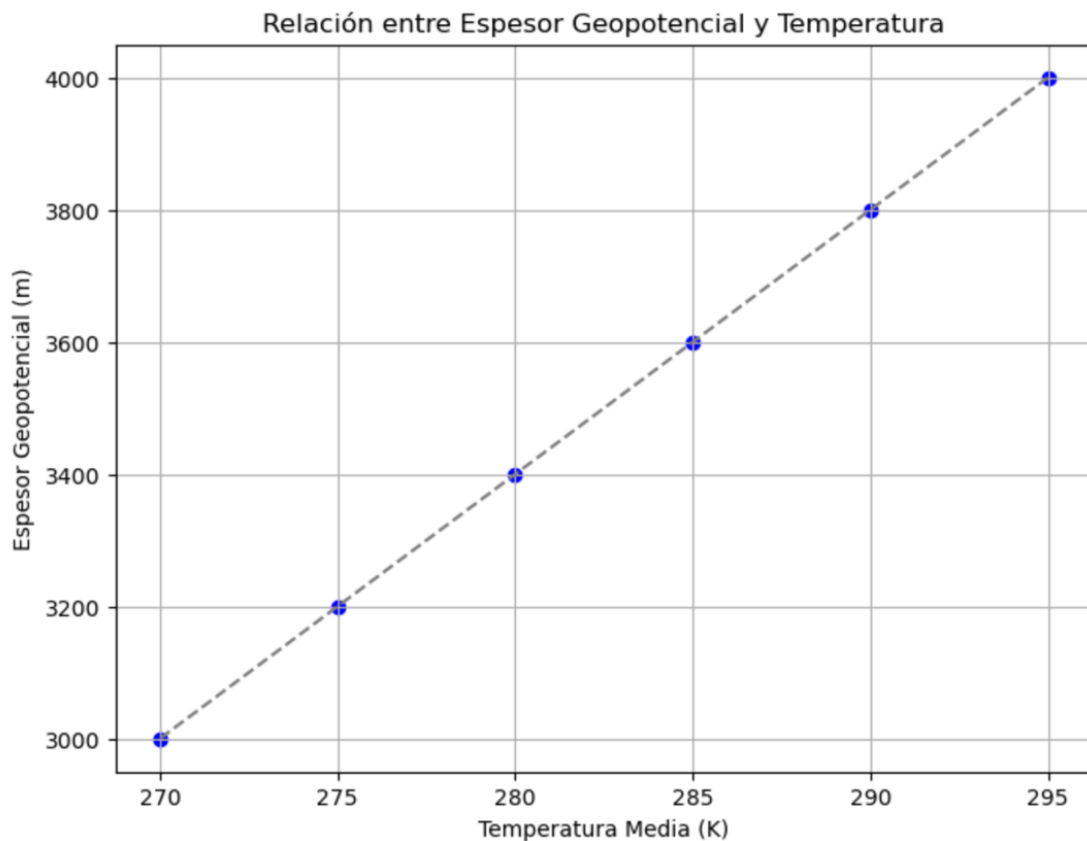


Imagen 20: Relación entre espesor geopotencial y temperatura

2. Detección de Anomalías Térmicas:

- La variabilidad en el espesor geopotencial permite detectar anomalías térmicas, las cuales son fundamentales para identificar la formación y la intensificación de CTs. Estas anomalías a menudo preceden el desarrollo de estos sistemas, lo que ofrece una herramienta valiosa para la predicción temprana.

Gráfico 3: Anomalías del Espesor Geopotencial

Descripción: Mapa de anomalías del espesor geopotencial que muestra áreas donde el espesor es significativamente mayor o menor de lo normal. Las áreas con mayor espesor pueden indicar la presencia de núcleos cálidos, mientras que una reducción del espesor puede indicar inestabilidad atmosférica.

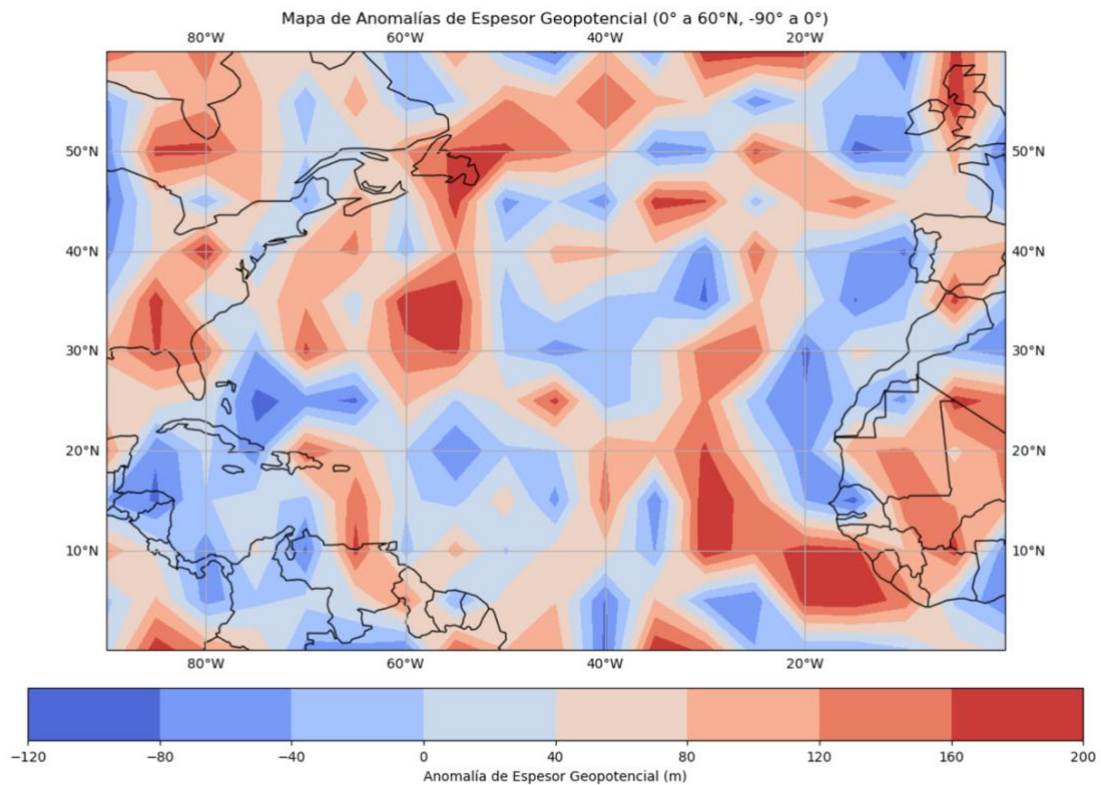


Imagen 21: Anomalías en el espesor geopotencial

3. Análisis de la Estabilidad Atmosférica:

- El espesor geopotencial entre 500 hPa y 925 hPa también es un buen indicador de la estabilidad atmosférica. Una disminución en este espesor puede indicar la presencia de una atmósfera inestable, propicia para el desarrollo de fenómenos convectivos y la formación de ciclones.

4. Identificación de Patrones Atmosféricos:

- La configuración del espesor geopotencial ayuda a identificar patrones atmosféricos como depresiones y crestas, que son fundamentales en la dinámica de los CTs. Estos patrones son esenciales para predecir la trayectoria y la intensidad de los ciclones.

Gráfico 4: Espesor Geopotencial y Patrones Atmosféricos

Descripción: Un mapa que muestra el espesor geopotencial con isobaras que indican áreas de alta y baja presión. Estas características ayudan a identificar los sistemas meteorológicos asociados a los CTs.

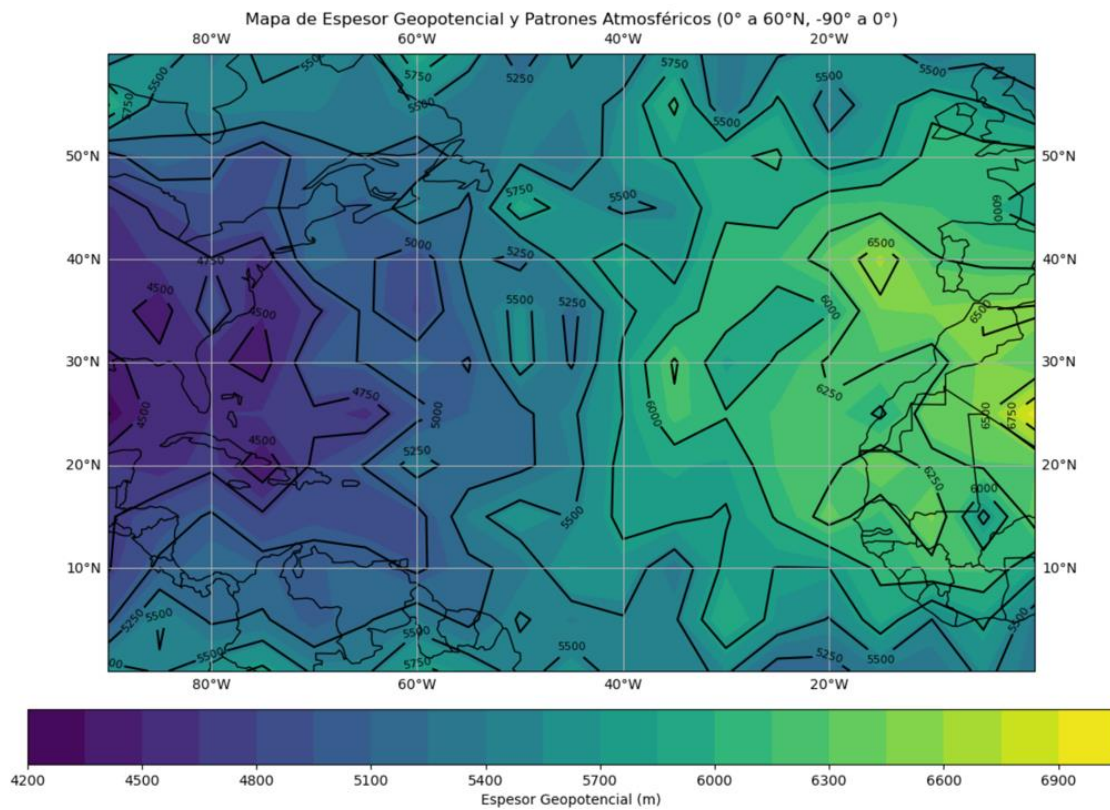


Imagen 22: Espesor geopotencial

Por estos motivo, en esta ocasión vamos a crear 2 Datasets uno correspondiente al geopotencial al nivel 500 hPa y otro al nivel 925 hPa, para posteriormente, calcular el espesor geopotencial como la diferencia entre las alturas geopotenciales de estos niveles. Para ello tenemos creamos un método, propio del dataset, que se encargue de restar los datos de ambos datasets:

```

def _load_data2(self, data2, normalise=True):
    data = []
    for field, label in self._fields:
        for field2, label2 in data2._fields:
            if normalise:
                array1 = normalise_01(field.to_numpy())
                array2 = normalise_01(field2.to_numpy())
                array = array1 - array2
            else:
                array1 = field.to_numpy()
                array2 = field2.to_numpy()
                array = array1 - array2
            data.append((array, label, field))
    return data

```

Imagen 23: Función que resta los dos datasets para obtener el espesor geopotencial

Y lo mismo con la función que divide los datos en conjunto de entrenamiento y conjunto test:

```

def load_data2(self, data2, normalise=True, test_size=0.5, shuffle=True, fields=False):
    data = self._load_data2(data2, normalise)

    if shuffle:
        random.shuffle(data)
    half = int(len(data) * (1.0 - test_size))

    x_train, y_train, f_train = (
        np.array([x[0] for x in data[:half]]),
        np.array([x[1] for x in data[:half]]),
        [x[2] for x in data[:half]],
    )

    x_test, y_test, f_test = (
        np.array([x[0] for x in data[half:]]),
        np.array([x[1] for x in data[half:]]),
        [x[2] for x in data[half:]],
    )

    if fields:
        return (x_train, y_train, f_train), (x_test, y_test, f_test)

    return (x_train, y_train), (x_test, y_test)

```

Imagen 24: Función que divide el dataset

Dividimos nuestra base de datos en conjunto de entrenamiento y de prueba haciendo uso de la función `load_data2()` presentada anteriormente.

Después de seguir los mismo pasos que, en notebook anterior, se llega a la evaluación

```
model.evaluate(x_test, y_test, verbose=1)
```

```
2/2 [=====] - 0s 2ms/step - loss: 0.1673 - accuracy: 0.9322  
[0.16730493307113647, 0.9322034120559692]
```

En este caso este modelo tiene una exactitud de 0.9322 y una función de pérdida que en el conjunto de entrenamiento decrece (ver Muestra Notebook 4) lo cual es favorable y nos indica que el entrenamiento mejora con cada “época”.

```
In [15]: h = model.fit(x_train, y_train, epochs=100, verbose=1)
```

```
Epoch 1/100  
5/5 [=====] - 0s 2ms/step - loss: 0.0630 - accuracy: 0.9926  
Epoch 2/100  
5/5 [=====] - 0s 2ms/step - loss: 0.0597 - accuracy: 1.0000  
Epoch 3/100  
5/5 [=====] - 0s 2ms/step - loss: 0.0602 - accuracy: 1.0000  
Epoch 4/100  
5/5 [=====] - 0s 2ms/step - loss: 0.0575 - accuracy: 1.0000  
Epoch 5/100  
5/5 [=====] - 0s 2ms/step - loss: 0.0607 - accuracy: 1.0000  
Epoch 6/100  
5/5 [=====] - 0s 2ms/step - loss: 0.0564 - accuracy: 1.0000  
Epoch 7/100  
5/5 [=====] - 0s 2ms/step - loss: 0.0548 - accuracy: 0.9926  
Epoch 8/100  
5/5 [=====] - 0s 2ms/step - loss: 0.0558 - accuracy: 1.0000  
Epoch 9/100  
5/5 [=====] - 0s 2ms/step - loss: 0.0580 - accuracy: 1.0000  
Epoch 10/100  
5/5 [=====] - 0s 2ms/step - loss: 0.0566 - accuracy: 1.0000  
Epoch 11/100  
5/5 [=====] - 0s 2ms/step - loss: 0.0627 - accuracy: 0.9853  
Epoch 12/100  
5/5 [=====] - 0s 2ms/step - loss: 0.0550 - accuracy: 1.0000  
Epoch 13/100  
5/5 [=====] - 0s 2ms/step - loss: 0.0549 - accuracy: 1.0000  
Epoch 14/100  
5/5 [=====] - 0s 2ms/step - loss: 0.0497 - accuracy: 1.0000  
Epoch 15/100  
5/5 [=====] - 0s 2ms/step - loss: 0.0518 - accuracy: 1.0000  
Epoch 16/100  
5/5 [=====] - 0s 2ms/step - loss: 0.0472 - accuracy: 1.0000  
Epoch 17/100  
5/5 [=====] - 0s 2ms/step - loss: 0.0464 - accuracy: 1.0000  
Epoch 18/100  
5/5 [=====] - 0s 2ms/step - loss: 0.0448 - accuracy: 1.0000  
Epoch 19/100  
5/5 [=====] - 0s 2ms/step - loss: 0.0464 - accuracy: 1.0000  
Epoch 20/100  
5/5 [=====] - 0s 2ms/step - loss: 0.0451 - accuracy: 1.0000  
Epoch 21/100  
5/5 [=====] - 0s 2ms/step - loss: 0.0476 - accuracy: 0.9926  
Epoch 22/100  
5/5 [=====] - 0s 2ms/step - loss: 0.0428 - accuracy: 1.0000  
Epoch 23/100  
5/5 [=====] - 0s 2ms/step - loss: 0.0436 - accuracy: 1.0000  
Epoch 24/100  
5/5 [=====] - 0s 2ms/step - loss: 0.0428 - accuracy: 1.0000  
Epoch 25/100  
5/5 [=====] - 0s 2ms/step - loss: 0.0428 - accuracy: 1.0000  
Epoch 26/100  
5/5 [=====] - 0s 2ms/step - loss: 0.0424 - accuracy: 1.0000
```

Imagen 25: Muestra Notebook 5 - Entrenamiento diferencia geopotencial

Probamos nuestro modelo utilizando la función predict, le pasamos el conjunto test para que realice predicciones:

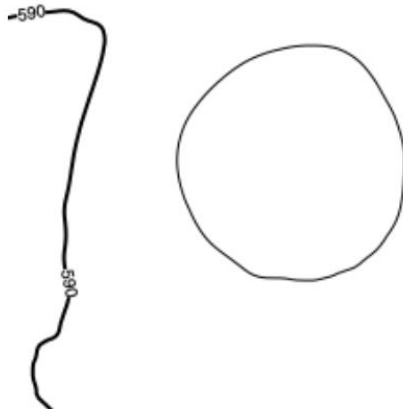
```
predicted = model.predict(x_test)
```

A continuación, se muestran las predicciones realizadas por nuestro modelo:

[4.4375239e-04 9.9955624e-01]

GribField(z,500,20180805,1800,0,0)

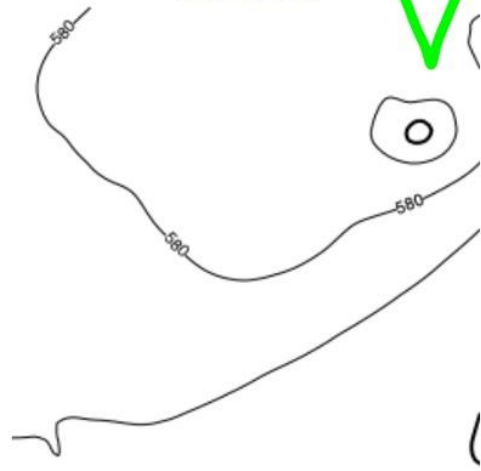
No Tropical (100%)



[9.9917346e-01 8.2653482e-04]

GribField(z,500,20180915,0,0,0)

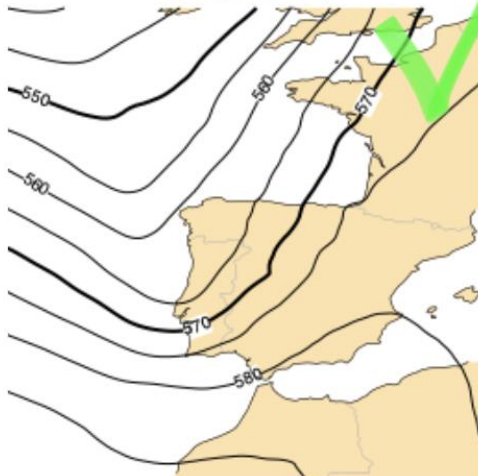
Tropical (100%)



[0.07378105 0.926219]

GribField(z,500,20181014,0,0,0)

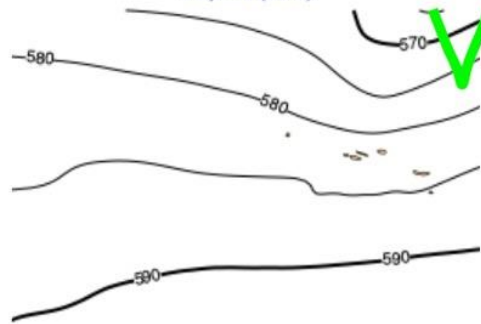
No Tropical (93%)



[0.9876056 0.01239439]

GribField(z,500,20180916,1800,0,0)

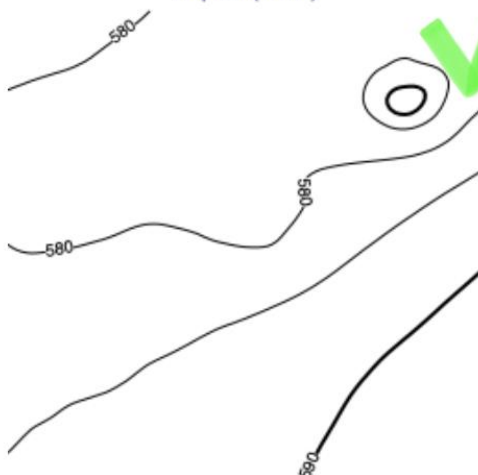
Tropical (99%)



[0.9970475 0.00295259]

GribField(z,500,20180915,1200,0,0)

Tropical (100%)



[0.18770944 0.8122906]

GribField(z,500,20180928,1800,0,0)

No Tropical (81%)

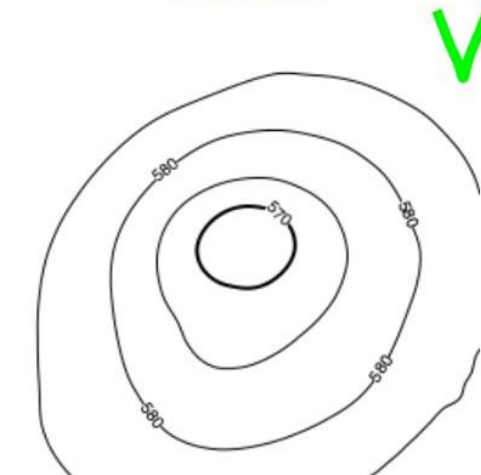


Imagen 26: Predicciones exitosas

Vemos que al calcular el espesor geopotencial, las predicciones mejoran considerablemente, se obtienen solo 4 predicciones fallidas de las 59.

Entonces, guardamos este modelo, para aumentar el tamaño de los datasets y ver si sigue siendo lo suficientemente bueno.

```
model.save("./modeloDifGeopotencial.h5")
```

Capítulo 6 Caso de estudio

6.1. Dataset final

Después de realizar diversas pruebas en los informes 1 y 2, hemos preparado un dataset más grande y desafiante para evaluar la eficacia del modelo guardado. Este dataset se obtiene de HURDAT y contiene datos de fenómenos meteorológicos bajo las siguientes condiciones:

- Presión < 1000 hPa
- Latitud entre 0° y 60° Norte
- Longitud entre -60° y 0°
- Fenómenos ocurridos a partir del año 2000

	bassin	number	year	name	time	type	status	lat	lon	knots	category	pressure
41765	AL	5	2001	dean	#####		TS	40,6	-59,4	60	1	994
41766	AL	5	2001	dean	#####		TS	42,1	-57,5	60	1	995
41767	AL	5	2001	dean	#####		TS	43,5	-56	55	1	996
41768	AL	5	2001	dean	#####		TS	44	-53,4	50	1	997
41769	AL	5	2001	dean	#####		EX	45,5	-50,5	45	1	998
41770	AL	5	2001	dean	#####		EX	47	-48,5	45	1	999
41771	AL	5	2001	dean	#####		EX	49	-45	45	1	999
41799	AL	6	2001	erin	#####		TS	26,5	-58,8	40	1	999
41800	AL	6	2001	erin	#####		TS	27,2	-59,2	45	1	999
41801	AL	6	2001	erin	#####		TS	28,4	-59,8	60	1	994
41822	AL	6	2001	erin	#####		HU	40,6	-59,3	70	1	982
41823	AL	6	2001	erin	#####		HU	42	-58,1	65	1	987
41824	AL	6	2001	erin	#####		HU	43,3	-56,7	65	1	987
41825	AL	6	2001	erin	#####		HU	44,7	-55,2	65	1	984
41826	AL	6	2001	erin	#####		TS	46,7	-52,7	60	1	981
41827	AL	6	2001	erin	#####		EX	49	-51	60	1	981
41828	AL	6	2001	erin	#####		EX	52	-49	55	1	978

Imagen 27: Muestra de HURDAT

La muestra inicial tiene 1977 entradas y abarca un rango de fechas desde 2001 hasta 2018. Para optimizar los cálculos, decidimos reducir el periodo de estudio a 2001-2010, resultando en 997 entradas.

Esta muestra contiene información sobre los centros de fenómenos meteorológicos, la fecha y la hora en que ocurrieron, y lo utilizamos para hacer peticiones de nuevos datos a ERA5 y realizar predicciones adicionales sobre este conjunto de datos significativamente mayor.

6.1.1. Preparación de los Datos

Para facilitar el procesamiento de los datos y su integración con el sistema de petición de ERA5, transformamos la tabla original a un formato más manejable mediante Python.

```

import pandas as pd

# Cargar solo la columna de interés desde el archivo de Excel
df = pd.read_excel('datos_de_centros.xlsx', usecols=['status', 'category'])

# Obtener los valores de la columna de interés como una lista
status = df['status'].tolist()[0:996]
category = df['category'].tolist()[0:996]

# Verificar que tengas suficientes valores en la lista
if len(status) == len(SAMPLES):
    # Agregar los nuevos valores como cuarto elemento a cada tupla
    SAMPLES = [(date, time, coords, status, category) for (date, time, coords), status, category in zip(SAMPLES, status, category)]
    print(SAMPLES)
else:
    print(len(status), len(SAMPLES))
    print("La cantidad de valores en la columna de Excel no coincide con la cantidad de tuplas en SAMPLES.")

```

A continuación, se presenta un ejemplo del formato utilizado:

```

SAMPLES = [
    ("2001-08-28", "00", (-57.5, 42.1)),
    ("2001-08-28", "06", (-56.0, 43.5)),
    ("2001-08-28", "12", (-53.4, 44.0)),
    ("2001-08-28", "18", (-50.5, 45.5)),
    ...
]

```

6.2. Aplicación del Modelo

Utilizamos el modelo de red neuronal que habíamos guardado previamente, desarrollado con la biblioteca Keras. La arquitectura del modelo incluye capas de entrada y salida, así como capas ocultas. El modelo fue entrenado con un conjunto de datos normalizados y etiquetas codificadas.

El modelo entrenado se evaluó utilizando una muestra de calibración, es decir, un conjunto de prueba independiente, obteniendo un rendimiento de 93.22% y una función de pérdida que decrecía consistentemente durante el entrenamiento. Se realizaron predicciones y se compararon con las etiquetas reales para calcular las métricas de rendimiento.

```

modelo = keras.models.load_model('./modeloDifGeopotencial.h5')

```

6.3. Evaluación sobre la Muestra de ERA5

Para evaluar la eficacia del modelo, utilizamos una muestra de verificación, es decir, datos adicionales obtenidos de ERA5, el reanálisis de datos climáticos del ECMWF que proporciona un registro detallado de variables atmosféricas. La evaluación se centró en analizar la precisión del modelo al predecir fenómenos meteorológicos en esta nueva muestra de datos.

1. Obtención de Datos de ERA5:

- Se realizó una petición de datos atmosféricos específicos a ERA5, incluyendo variables como la temperatura, presión y espesor geopotencial, para las ubicaciones y tiempos correspondientes a la muestra inicial de HURDAT.
2. **Cálculo del Espesor Geopotencial:**
 - Se calcula el espesor geopotencial entre ambas muestras igual que en los informes anteriores, utilizando datos de altitud a diferentes niveles de presión (500 hPa y 925 hPa). Este cálculo nos permite obtener una medida integral de la temperatura media de la capa de aire comprendida entre estos niveles, que es ya vimos que es bastante útil para la evaluación de los CTs.
 3. **Preparación de los Datos de ERA5:**
 - Los datos obtenidos se estructuraron de manera adecuada para integrarse con el modelo previamente entrenado. Esto implica la normalización de los datos y la adaptación a los formatos requeridos por el modelo.
 4. **Clasificación y Uso del Conjunto de Prueba:**
 - Se utiliza el 50% de la muestra para realizar la clasificación inicial. El modelo sólo es capaz de identificar fenómenos meteorológicos específicos en un 44% de los casos.

```
: print('TOTAL: ', len(predicted2), ' ACIERTOS: ', aux, 'PORCENTAJE DE ACIERTOS: ', aux/len(predicted2))
TOTAL:  477  ACIERTOS:  213 PORCENTAJE DE ACIERTOS:  0.44654088050314467
```

5. **Entrenamiento Adicional con Nuevos Datos:**
 - Con base en los resultados obtenidos, decidimos alimentar el modelo con nuevos datos de la muestra ampliada de ERA5. Dividimos los datos en un 30% para entrenamiento y un 70% para prueba para evaluar si el modelo pudiera mejorar su precisión.
 - **Proceso de Codificación de Etiquetas:** Dado que la nueva muestra no contenía etiquetas de tipo one-hot, traducimos las etiquetas existentes (LO, SD, SS, TD, TS, HU, EX) a la clasificación binaria utilizada inicialmente, donde 1 indica CT (CT) y 0 no CT (no CT).

```
import numpy as np
from tensorflow.keras.utils import to_categorical
y2_train_encoded = [np.array([1.0, 0.0]) if label in ['TS', 'TD', 'HU']
                     else np.array([0.0, 1.0]) for label in y2_train]
y2_train_encoded = np.array(y2_train_encoded)
y2_test_encoded = [np.array([1.0, 0.0]) if label in ['TS', 'TD', 'HU']
                   else np.array([0.0, 1.0]) for label in y2_test]
y2_test_encoded = np.array(y2_test_encoded)
```

- Se utilizó el conjunto de entrenamiento (x2_train, y2_train_encoded) para entrenar el modelo mejorado, configurando 10 épocas para el proceso de entrenamiento.
- Durante cada época, se realizó una optimización iterativa para minimizar la función de pérdida y mejorar la precisión del modelo. Los resultados del entrenamiento se muestran a continuación:

```
modelo.fit(x2_train, y2_train_encoded, epochs=10)
```

```
Epoch 1/10
22/22 [=====] - 0s 2ms/step - loss: 0.2790 - accuracy: 0.8852
Epoch 2/10
22/22 [=====] - 0s 2ms/step - loss: 0.2778 - accuracy: 0.8953
Epoch 3/10
22/22 [=====] - 0s 2ms/step - loss: 0.2635 - accuracy: 0.8996
Epoch 4/10
22/22 [=====] - 0s 2ms/step - loss: 0.2688 - accuracy: 0.9010
Epoch 5/10
22/22 [=====] - 0s 2ms/step - loss: 0.2745 - accuracy: 0.8924
Epoch 6/10
22/22 [=====] - 0s 2ms/step - loss: 0.2695 - accuracy: 0.8881
Epoch 7/10
22/22 [=====] - 0s 2ms/step - loss: 0.2569 - accuracy: 0.9039
Epoch 8/10
22/22 [=====] - 0s 2ms/step - loss: 0.2508 - accuracy: 0.9096
Epoch 9/10
22/22 [=====] - 0s 2ms/step - loss: 0.2467 - accuracy: 0.9024
Epoch 10/10
22/22 [=====] - 0s 2ms/step - loss: 0.2455 - accuracy: 0.9039

<keras.callbacks.History at 0x7fac28de8d00>
```

```
modelo.evaluate(x2_test, y2_test_encoded, verbose=1)
```

```
10/10 [=====] - 0s 938us/step - loss: 0.6839 - accuracy: 0.7324
```

6. Resultados de la Evaluación:

- El nuevo entrenamiento resultó en una mejora significativa, alcanzando un 73% de exactitud en la evaluación y una mejora progresiva en la precisión del modelo a medida que avanzaban las épocas de entrenamiento. Este resultado indica que el modelo tiene una buena capacidad para generalizar y predecir fenómenos meteorológicos utilizando datos de ERA5.

6.4. Resultados y Mejora del Modelo

Guardamos este modelo mejorado y realizamos nuevas predicciones, tomando como conjunto de prueba el 90% de la muestra.

```
: print('TOTAL: ', len(predicted2), ' ACIERTOS: ', aux, 'PORCENTAJE DE ACIERTOS: ', (aux/len(predicted2))*100)
TOTAL: 897 ACIERTOS: 755 PORCENTAJE DE ACIERTOS: 84.16945373467112
```

Los resultados mejoraron notablemente, pasando de un 44% a un 84% de aciertos.

Ejemplos de Clasificación Exitosa:

A modo de ilustración, se presentan algunos ejemplos en los que el modelo fue capaz de realizar una clasificación correcta en situaciones complejas, donde la detección manual podría resultar difícil.



Imagen 28: Ejemplo de predicción exitosa en casos dudosos

El modelo identificó correctamente ciclones en áreas donde las condiciones atmosféricas eran complejas y confusas, destacando su capacidad para manejar grandes volúmenes de datos y realizar predicciones precisas en poco tiempo.

6.5. Conclusión

El modelo mejorado y su evaluación demuestran que la integración de datos adicionales y el reentrenamiento pueden llevar a una mejora significativa en la precisión de las predicciones. El uso de datos de ERA5 y el procesamiento adecuado de las etiquetas han permitido que el modelo se adapte mejor a la tarea de clasificación de CTs, proporcionando una herramienta útil para la predicción de estos fenómenos climáticos.

Capítulo 7 Conclusiones y trabajo futuro

En este último capítulo tratamos las conclusiones a las que se llegan al final del proyecto, así como posibles mejoras para líneas de trabajos futuros.

7.1. Conclusiones

La implementación de este modelo puede tener un impacto significativo en la preparación y respuesta ante CTs, mejorando la seguridad pública y reduciendo las pérdidas económicas. Las instituciones meteorológicas pueden beneficiarse directamente al integrar estas tecnologías en sus sistemas de monitoreo y predicción.

Los objetivos que nos han llevado hasta aquí pueden explicarse como:

Evaluación de las Predicciones Meteorológicas Actuales (OBJ-1): En el **Capítulo 3**, se realizó un análisis detallado de las técnicas actuales de predicción meteorológica, identificando las limitaciones de los enfoques tradicionales y explorando cómo los modelos de Machine Learning (ML) pueden superarlas. Las Redes Neuronales Convolucionales (CNN) y las Redes Neuronales Recurrentes (RNN) demostraron ser herramientas efectivas para mejorar la precisión y eficiencia de las predicciones meteorológicas, cumpliendo así con el primer objetivo.

Desarrollo y Entrenamiento del Modelo Predictivo (OBJ-2): En el **Capítulo 5**, se diseñó y entrenó un modelo predictivo utilizando conjuntos de datos previamente clasificados. Este modelo mostró gran habilidad en la clasificación de CTs, validando la eficacia de las redes neuronales densamente conectadas en la identificación de estos fenómenos.

Adquisición de Datos Climáticos (OBJ-3): En el **Capítulo 4**, se utilizaron datos climáticos del reanálisis ERA5, específicamente de los niveles de presión de 500 y 925 hPa. La utilización de ClimeLab facilitó significativamente la adquisición y gestión eficiente de estos datos, cumpliendo plenamente con el objetivo de emplear herramientas modernas en el desarrollo del proyecto.

Preprocesamiento de los Datos (OBJ-4): En el **Capítulo 4**, se llevó a cabo un exhaustivo preprocesamiento de los datos, que incluyó la creación de muestras detalladas. Esto fue crucial para preparar los datos para el análisis y entrenamiento del modelo, garantizando la calidad de los datos utilizados.

Diseño y Entrenamiento del Modelo (OBJ-5): En el **Capítulo 5**, se implementó un modelo de red neuronal utilizando la biblioteca Keras. Este modelo logró capturar y aprender patrones complejos presentes en los datos climáticos, demostrando su eficacia en la identificación de CTs.

Evaluación del Modelo (OBJ-6): En el **Capítulo 6**, el modelo se evaluó exhaustivamente utilizando una muestra independiente. Las predicciones mostraron una precisión significativa, cumpliendo así el objetivo de desarrollar un modelo predictivo eficaz.

La metodología ágil utilizada, con iteraciones y revisiones periódicas, facilitó la organización y continuidad del trabajo. Además, la integración de ClimeLab fue esencial para simplificar el acceso y procesamiento de datos meteorológicos complejos, permitiendo centrar los esfuerzos en la optimización del modelo. También hay que destacar la comunicación fluida y efectiva a través de Teams, que ayudó a resolver rápidamente cualquier duda o problema que surgiera, asegurando que el proyecto se mantuviera en el buen camino.

7.2. Conclusiones personales

En el ámbito personal, el desarrollo de este proyecto me ha permitido profundizar en el campo de la predicción meteorológica mediante el uso de Machine Learning. Durante este proceso, he llegado a comprender lo complicado que es trabajar con estos sistemas, pero también su gran potencial para mejorar cómo entendemos y predecimos fenómenos meteorológicos. Además de aprender mucho sobre la parte técnica, este proyecto me ha ayudado a mejorar en la gestión de proyectos, la resolución de problemas y el trabajo en equipo. Estas habilidades serán muy valiosas en mi futuro profesional, ya que he aprendido a cómo enfrentar desafíos difíciles de manera organizada y a colaborar efectivamente con otros para alcanzar metas comunes.

7.3 Trabajo futuro

Aunque el proyecto logró sus objetivos, hay varias áreas que podrían beneficiarse de un trabajo adicional para mejorar los resultados y ampliar el alcance del modelo:

Ampliación del Dataset:

- Aumentar el dataset con más variables atmosféricas y expandir la cobertura geográfica para mejorar la precisión del modelo.

Exploración de Nuevas Arquitecturas:

- Explorar la implementación de modelos híbridos que combinen ML con métodos tradicionales para abordar mejor las incertidumbres en la predicción de CTs.

Procesamiento en Tiempo Real:

- Desarrollar capacidades de procesamiento en tiempo real para la implementación en sistemas de monitoreo y respuesta rápida ante desastres meteorológicos.

Evaluación Continua y Validación Cruzada:

- Implementar técnicas de validación cruzada más robustas y evaluar el modelo con datos recientes para asegurar su relevancia y precisión a lo largo del tiempo.

Estas líneas de trabajo futuro podrían potenciar significativamente la eficacia del modelo y su aplicabilidad en escenarios reales, proporcionando una herramienta más robusta y útil para la predicción y gestión de fenómenos meteorológicos extremos.

Parte III Apéndices

Apéndice A Manual de Instalación

1. Descarga e Instalación de Anaconda

1. Descargar Anaconda:

- Abre tu navegador web y visita la página oficial de Anaconda: Anaconda Downloads.

2. Instalar Anaconda:

- Abre el archivo `.pkg` descargado y sigue las instrucciones en pantalla para completar la instalación.
- Una vez finalizada la instalación, abre la Terminal y verifica la instalación ejecutando el siguiente comando:

```
conda --version
```

- ### 3. Abrir la Terminal:
- Una vez que Anaconda esté instalado, abre la Terminal en el nuevo ordenador.

- ### 4. Importar el entorno desde el archivo YAML:
- Navega al directorio donde copiaste el archivo `tfg_env.yaml` y ejecuta el siguiente comando para crear el entorno desde el archivo YAML:

```
conda env create -f tfg_env.yaml
```

- ### 5. Activar el nuevo entorno:
- Después de que el entorno se haya creado, actívalo con el siguiente comando:

```
conda activate tfg_env
```

- ### 6. Verificación y ejecución:
- Una vez que el entorno esté activado, verificar la instalación de Python y las dependencias, y luego ejecutar tu proyecto en Jupyter Notebook.

- **Verificar la versión de Python:**

```
python --version
```

- **Iniciar Jupyter Notebook:**

```
jupyter notebook
```


Apéndice B Contenido adjunto

A continuación, se detallan los archivos que se adjuntan para complementar la memoria. Son archivos .ipynb, archivos python para obtener los datasets y modelo utilizados.

- **Notebooks de prueba:** Durante la fase de investigación se realizan una serie de informes donde se documentan los distintos resultados en función de los parámetros escogidos para formar los dataset y entrenar el modelo.
 - Notebook inicial. Antes de iniciar el proyecto se experimenta con este notebook para una primera toma de contacto.
 - Cuaderno: pruebaCimetLab.ipynb
 - Modelo: modeloEntrenado.h5
 - Archivo Python para obtener el dataset: high_low.py
 - Notebooks informes. Una vez iniciado el proyecto se hacen pruebas con distintos parámetros hasta conseguir un modelo satisfactorio.
 - Cuadernos:
 - Tropical_1.ipynb
 - Tropical_2.ipynb
 - Tropical_3.ipynb
 - diferencial_centros.ipynb
 - Materiales para ampliar el tamaño del dataset:
 - Cuaderno: Centros.ipynb
 - datos_de_centros.xlsx
 - Modelos:
 - modeloEntrenado.h5
 - modeloDifGeopotencial.h5
 - Archivos Python para obtener los datasets:
 - tropical.py
 - tropical2.py
 - centros.py
 - centros2.py
- **Notebook caso final:** Una vez los informes nos marcan el camino a seguir se obtiene el caso final de estudio.
 - Cuaderno:
 - diferencial_geopotencial.ipynb
 - Modelo:
 - modeloEntrenado.h5
 - Archivos Python para obtener los datasets:
 - centros.py
 - centros2.py

Definición de siglas y abreviaturas

Es interesante mostrar una tabla o lista de abreviaturas y siglas (acrónimos) de uso extendido en la Memoria, pero de los que es posible que el lector de la misma no tenga conocimiento o sean ambiguos.

ML: Machine Learning

IA: Inteligencia Artificial

CT: Ciclón Tropical

NOAA: National Oceanic and Atmospheric Administration

GFS: Global Forecast Service

NHC: National Hurricane Center

ECMWF: European Centre for Medium-Range Weather Forecasts

Bibliografía y Webgrafía

- [1] Dvorak, V. F. (1975). Tropical Cyclone Intensity Analysis Using Satellite Data. NOAA Technical Report, NESDIS 11.
- [2] Dvorak, V. F., & Velden, C. S. (1990). The Dvorak Technique in the Satellite Era. Monthly Weather Review, 118(5), 859-865.
- [3] National Aeronautics and Space Administration (NASA). What is a hurricane, typhoon, or tropical cyclone? Recuperado de: <https://www.nasa.gov/audience/forstudents/k-4/stories/nasa-knows/what-are-hurricanes-k4.html>
- [4] Fundación Aon. Coste de las Catástrofes - Análisis Anual 2020. Recuperado de: <https://www.aon.com/data-analytics/impact-forecasting/reports/2020-annual-global-climate-and-catastrophe-report.pdf>
- [5] European Centre for Medium-Range Weather Forecasts (ECMWF). Reanalysis ERA5. Recuperado de: <https://www.ecmwf.int/en/forecasts/datasets/reanalysis-datasets/era5>
- [6] National Oceanic and Atmospheric Administration (NOAA). Global Forecast System (GFS). Recuperado de: <https://www.ncdc.noaa.gov/data-access/model-data/model-datasets/global-forecast-system-gfs>
- [7] National Oceanic and Atmospheric Administration (NOAA). "Tropical Cyclone Climatology." NOAA National Hurricane Center, 2021. Recuperado de: <https://www.nhc.noaa.gov/climo/>
- [8] European Centre for Medium-Range Weather Forecasts (ECMWF). Integrated Forecasting System (IFS). Recuperado de: <https://www.ecmwf.int/en/forecasts/datasets/integrated-forecast-system-ifs>
- [9] National Hurricane Center (NHC). Tropical Cyclone Report: Hurricane Katrina. Recuperado de: https://www.nhc.noaa.gov/data/tcr/AL122005_Katrina.pdf
- [10] National Hurricane Center (NHC). Tropical Cyclone Report: Hurricane Charley. Recuperado de: https://www.nhc.noaa.gov/data/tcr/AL032004_Charley.pdf
- [11] National Hurricane Center. "About Tropical Cyclones." National Oceanic and Atmospheric Administration, 2022. URL: <https://www.nhc.noaa.gov/aboutcyclones.php>
- [12] ECMWF. ML at ECMWF: A roadmap for the next 10 years. Recuperado de: <https://www.ecmwf.int/sites/default/files/elibrary/2021/19877-machine-learning-ecmwf-roadmap-next-10-years.pdf>

- [13] Meteoradares. (2012, agosto 26). Y por qué necesitábamos un radar Doppler. Recuperado de: <https://meteoradares.wordpress.com/2012/08/26/y-por-que-necesitabamos-un-radar-doppler/>
- [14] Chanh Kieu, et al. "Predicting Tropical Cyclone Formation with Deep Learning". *Weather and Forecasting*, vol. 39, no. 1, 2024. Recuperado de: [AMETSOC](#)
- [15] "Tropical cyclone size estimation based on deep learning using infrared and microwave satellite data". *Frontiers in Environmental Science*, 2020. Recuperado de: [Frontiers](#)
- [16] Yuping Zhang, Jianguo Pan. "A Novel Deep Learning Approach for Tropical Cyclone Track Prediction Based on Auto-Encoder and Gated Recurrent Unit Networks". *Applied Sciences*, vol. 10, no. 11, 2020. Recuperado de: [MDPI](#)
- [17] Kerry Emanuel. "The Hurricane-Climate Connection." *Bulletin of the American Meteorological Society*, vol. 89, no. 5, 2008, pp. ES10–ES20. Recuperado de: [AMS Journals](#)
- [18] Emanuel, K. A. "The Maximum Intensity of Hurricanes." *Journal of the Atmospheric Sciences*, vol. 45, no. 7, 1988, pp. 1143-1155. Recuperado de: [Journals](#)
- [19] World Meteorological Organization (WMO). "Tropical Cyclones: A Global View." WMO, 2017. Recuperado de: <https://public.wmo.int/en/our-mandate/focus-areas/natural-hazards-and-disaster-risk-reduction/tropical-cyclones>
- [20] Emanuel, K. A. "An Overview of Tropical Cyclones: Their Nature and Impacts." *Annual Review of Earth and Planetary Sciences*, vol. 31, 2003, pp. 75-104. Recuperado de: <https://www.annualreviews.org/doi/abs/10.1146/annurev.earth.31.100901.141259>
- [21] National Oceanic and Atmospheric Administration (NOAA). "El Niño & La Niña (El Niño-Southern Oscillation)." NOAA, 2022. Recuperado de: <https://www.noaa.gov/education/resource-collections/weather-atmosphere/el-nino>
- [22] Emanuel, K. A. (2005). *Divine Wind: The History and Science of Hurricanes*. Oxford University Press.
- [23] National Aeronautics and Space Administration (NASA). "Machine Learning in Earth Science." NASA Earth Science Division, 2021. Recuperado de: <https://earthdata.nasa.gov/learn/articles/machine-learning-in-earth-science>
- [24] Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep Learning*. MIT Press. Recuperado de: <https://www.deeplearningbook.org/>
- [25] Aggarwal, C. C. (2018). *Neural Networks and Deep Learning: A Textbook*. Springer. Recuperado de: <https://link.springer.com/book/10.1007/978-3-319-94463-0>
- [26] NASA Science. "A Force of Nature: Hurricanes in a Changing Climate." NASA, 2021. Recuperado de: <https://science.nasa.gov/earth/climate-change/a-force-of-nature-hurricanes-in-a-changing-climate/>

[27] Gao, Z.; Tan, C.; Wu, L.; Li, S.Z. SimVP: Simpler Yet Better Video Prediction. In Proceedings of the 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 19–20 June 2022; pp. 3170–3180. Recuperado de: [Simpler Yet Better Video Prediction](#)

