



FACULTAD DE CIENCIAS

Memoria de Prácticas Externas

External Internship Report

Memoria de prácticas externas realizadas en

Evenbytes S.L

Alumno: Alberto Campuzano Solórzano
Grado en Matemáticas

Periodo: 16/06/2025 – 29/08/2025

Tipo de prácticas: Curriculares + Extracurriculares

Número de horas: 300 + 50

Tutor externo: Jacinto Pelayo, CEO
Tutor académico: Alicia Nieto Reyes

Santander, 1 de agosto de 2025

Resumen

Durante mis prácticas en Evenbytes S.L., una consultora tecnológica especializada en soluciones cloud y transformación digital, he desarrollado competencias en el diseño de aplicaciones web y, de manera principal, en ciencia de datos. En la fase inicial, centrada en la formación acerca del desempeño principal de la empresa, trabajé con el *framework* Angular, el entorno de *Google Cloud Platform* (GCP) y el sistema de control de versiones *GitHub*. Posteriormente, participé en diversos proyectos, destacando especialmente uno orientado a la implementación de un modelo de predicción del precio de la electricidad y a la optimización de los procesos de fundición de unos hornos para la reducción del consumo energético. Esta experiencia me ha permitido integrar los conocimientos adquiridos en los grados de Matemáticas y Física, además de proporcionarme una visión más clara y aplicada del entorno profesional tecnológico.

Palabras clave: Ciencia de datos, modelo de predicción, machine learning, Google Cloud Platform

Abstract

During my internship at Evenbytes S.L., a technology consultancy specializing in cloud solutions and digital transformation, I developed skills in web application design and, primarily, in data science. In the initial phase, which focused on training in the company's core operations, I worked with the Angular *framework*, the *Google Cloud Platform* (GCP) environment, and the *GitHub* version control system. Subsequently, I participated in several projects, most notably one aimed at implementing a prediction model for electricity prices and optimizing furnace casting processes to reduce energy consumption. This experience has allowed me to integrate the knowledge acquired from my Mathematics and Physics degrees, in addition to providing me with a clearer and more applied vision of the professional technological environment.

Keywords: Data science, predictive model, machine learning, Google Cloud Platform

Índice general

Siglas y acrónimos	4
Agradecimientos	5
1 Introducción	6
1.1 Elección	6
1.2 Objetivos	6
1.3 Centro	7
1.4 Práctica Externa	7
2 Descripción de las prácticas	8
2.1 Actividades desempeñadas	8
2.2 Formación	8
2.2.1 Angular	8
2.2.2 GCP	12
2.2.3 GitHub	13
2.3 Proyecto principal	14
2.3.1 Predicción de precios de la luz	14
2.3.1.1 Estudio y comprensión del mercado	14
2.3.1.2 Obtención y preparación de los datos	15
2.3.1.3 Análisis exploratorio	15
2.3.1.4 SARIMAX	15
2.3.1.5 XGBoost	17
2.3.1.6 TFT	18
2.3.1.7 Resultados	19
2.3.2 Modelo físico	19
2.3.2.1 Situación simplificada: Primera aproximación	19
2.3.2.2 Situación general	20
2.3.2.3 Traducción matemática	20
2.3.3 Optimización de la superficie de enfriamiento	21
2.4 Relación entre formación recibida y actividades	21
2.5 Atención y asesoramiento recibido	21
3 Conclusiones	23
Apéndice	24
3.1 Códigos formación	24
3.2 Código SARIMAX	27
3.3 Código XGBoost	27

3.4	Código TFT	28
-----	----------------------	----

Siglas y acrónimos

Como es habitual en documentos técnicos, a menudo se emplean sigla y acrónimos para abreviar términos específicos del área. Más aun, en en ámbito científico, el inglés predomina como lengua de trabajo por lo que muchos términos se encuentran en este idioma y la traducción al español no es siempre directa. A continuación se presenta una lista de los acrónimos más relevantes junto con su significado:

UC Universidad de Cantabria

IA Inteligencia artificial

TFT Temporal Fusion Transformers (Transformadores de fusión temporal)

XGBoost Extreme Gradient Boosting (Aumento extremo de gradiente)

SARIMAX (Seasonal AutoRegressive Integrated Moving Average with eXogenous regressors) (Medias móviles integradas autorregresivas estacionales con regresores exógenos)

GCP Google Cloud Platform (Plataforma Google Cloud)

TSF Time series forecasting (Predicción de series temporales)

MAE Mean absolute error (error absoluto promedio)

MAE Root mean squared error (raíz del error cuadrático promedio)

ACF Autocorrelation function (función de autocorrelación)

PACF Partial autocorrelation function (función de autocorrelación parcial)

API Application Programming Interface (Interfaz de Programación de Aplicaciones)

PVPC Precio de venta al pequeño consumidor

ESIOS Estructura de Intercambio de Información del Operador del Sistema (Operador del Sistema Eléctrico Español)

Agradecimientos

Me gustaría agradecer a todas las personas de Evenbytes su amabilidad e integración, la plantilla de trabajo es como una gran familia y han hecho que este tiempo haya sido muy agradable. En particular, a Miguel y Pedro, quienes más cerca de mí han estado en ese inmejorable *equipo Data* y que me han enseñado día a día. Con ellos me he divertido, he aprendido y he cumplido uno de mis objetivos con estas prácticas, que era conocer de primera mano el mundo empresarial. Naturalmente, agradecer también a Jacinto Pelayo, CEO de la empresa, su confianza desde el primer momento para la incorporación a la empresa. Finalmente, a mi tutora académica, Alicia Nieto, por haberme ayudado en la organización y elaboración de esta memoria.

En último lugar, no puedo dejar pasar la oportunidad (ni me dejarían) de agradecer a mi familia su apoyo. Como es comprensible, el inicio en el mundo laboral no es sencillo y gracias a ellos me ha costado menos.

En resumen, estas prácticas no solo me han permitido aplicar conocimientos adquiridos a lo largo de estos cuatro años de formación, sino que también han sobrepasado las expectativas de conocer el mundo empresarial y me han proporcionado aquella visión que necesitaba para decidir mi futuro profesional.

Muchas gracias

Capítulo 1

Introducción

1.1. Elección

El proceso que llevó a la decisión de incorporarme a Evenbytes estuvo influenciado por diversos factores. En primer lugar, tuve la fortuna de que amigos míos, ya graduados de esta misma facultad, realizasen las prácticas en esta empresa, teniendo estos además diferentes perfiles, de manera que contaba de inicio con la ventaja de conocer la empresa de manera previa. Además, durante la feria de empleo realizada el 30 de abril de este mismo año pude asistir a los diferentes puestos que situaban las empresas asistentes para promocionarse y captar posibles futuros trabajadores. Dentro de estos grupos, Evenbytes fue la empresa que más llamó mi atención gracias a un proyecto que involucraba la implementación de predicciones estadística, modelado físico y optimización matemática. Fue por tanto la conjunción de un proyecto que aunase las dos carreras en las que me estoy formando, con el conocimiento previo del buen ambiente de trabajo, la que hizo que me decantase por Evenbytes.

1.2. Objetivos

La decisión acerca del futuro laboral es una tarea complicada, y más aún cuando tu formación te otorga una plenitud de opciones. Además, me encontraba ante la indecisión de si continuar por el mundo académico o desligarme de lo *teórico* e insertarme en la aplicación de los conocimientos adquiridos. Ante esta situación, veía más necesario que tener un contacto directo con el mundo profesional para poder dilucidar cuál quería que fuera mi futuro. Entre las ramas de mi interés se encontraban principalmente el sector financiero y el tecnológico, ya que considero que ambas áreas ofrecen un campo de desarrollo muy amplio para los perfiles con formación en Física y Matemáticas.

Por tanto, buscar una empresa que encajase con estas características era indispensable. La mayor parte de las ofertas actuales, especialmente en ámbitos relacionados con el análisis de datos, la modelización estadística y el desarrollo de soluciones tecnológicas, se alinean con esta línea de interés. A la hora de seleccionar un centro que me aceptase, una de mis prioridades fue también el ambiente laboral, puesto que, en mi opinión, uno aprende más cuando se encuentra cómodo.

En este aspecto, Evenbytes me acogió de manera total, siendo un claro ejemplo de un entorno de trabajo saludable en el que se valora la iniciativa, la capacidad de adaptación y la aportación personal al proyecto común. Considero que realizar las prácticas en una empresa con estas características me permitía no sólo adquirir experiencia práctica sino también contrastar en primera persona cómo se aplican competencias transversales como el pensamiento crítico, toma de decisiones en proyec-

tos reales y la proactividad. Asimismo, esta oportunidad suponía un primer paso importante para explorar con mayor claridad qué caminos profesionales me gustaría recorrer en el futuro.

1.3. Centro

Evenbytes es una consultora tecnológica que impulsa la transformación digital de empresas mediante soluciones tecnológicas avanzadas. Con un recorrido de más de 10 años de experiencia, su objetivo es ayudar a las organizaciones a ser más eficientes, competitivas e innovadoras, adaptándose siempre a sus necesidades específicas.

Especializados en el desarrollo de software, infraestructuras en la nube y consultoría tecnológica, en Evenbytes se posiciona como un aliado estratégico para empresas de todos los tamaños, desde startups hasta grandes corporaciones. Son Google Cloud Partners y cuentan con experiencia consolidada en áreas como IA, Big Data e IoT, poniendo estas tecnologías al alcance de sus clientes para que puedan afrontar los retos de un entorno empresarial en constante evolución. Sus principales áreas de especialización incluyen el desarrollo de software personalizado, gestión de infraestructuras en la nube, automatización y optimización de procesos, Inteligencia artificial y Big Data e integración de sistemas y plataformas.

La empresa actualmente cuenta con 16 trabajadores y 4 estudiantes de prácticas. Entre los perfiles de los compañeros se encontraban principalmente el Grado universitario en Ingeniería Informática, Ingeniería de Telecomunicaciones, Grados superiores relacionados con la informática y y Grado en Matemáticas. La actividad está dividida en áreas de gerencia, ventas, consultoría, desarrollo y data science. Los 3 socios y principales responsables ocupan los cargos de CEO, CTO y COO

1.4. Práctica Externa

Las labores realizadas durante el tiempo de prácticas han sido diversas. Como bien es sabido, la formación tanto en matemáticas como en física dotan de una gran capacidad de polivalencia a la persona que la posee, por lo que los trabajos que he realizado han sido diversos. Previamente se ha mencionado que Evenbytes es una empresa que se encuentra en plena expansión dentro del sector de la transformación digital. Por ello, en las semanas iniciales de las prácticas llevé a cabo labores de formación relacionadas principalmente con esta rama: Programación en Angular, diseño de páginas web, entorno GCP, GitHub, etc. Tras haber realizado dichas tareas me incorporé en el equipo de *Data Science*. En este departamento fui incluido en varios proyectos que se describirán a continuación, en los que he aprendido diversos programas relacionados con la ciencia de datos como *BigQuery*, *Jupyter*, *Python* ... además de técnicas para el tratado de datos y predicciones como *SARIMAX* o *redes neuronales*. Adicionalmente, debido a las características de uno de los proyectos me vi involucrado en el modelado físico de un proceso termodinámico, por lo que también he puesto en práctica los conocimientos adquiridos en el grado en Física.

Capítulo 2

Descripción de las prácticas

2.1. Actividades desempeñadas

Durante mi periodo de prácticas me he visto involucrado en diversos proyectos y dado que las actividades desempeñadas han sido variadas, no es del todo posible crear un *log frame* que las recoja todas de manera organizada en el sentido temporal. Además, dichos proyectos no se han iniciado y finalizado durante mi estancia por lo que no se puede definir de una manera feaciente las fases en las que se encontraban. Por tanto, se diseccionará la descripción de las prácticas en dos partes. En primer lugar, una asociada a la formación, en la que se explicarán los conceptos básicos aprendidos en las primeras semanas de trabajo. A continuación, se expondrá el proyecto al que mayor tiempo dediqué por motivos de extensión, si bien incluirá una sección te menciones a estos otros trabajos.

2.2. Formación

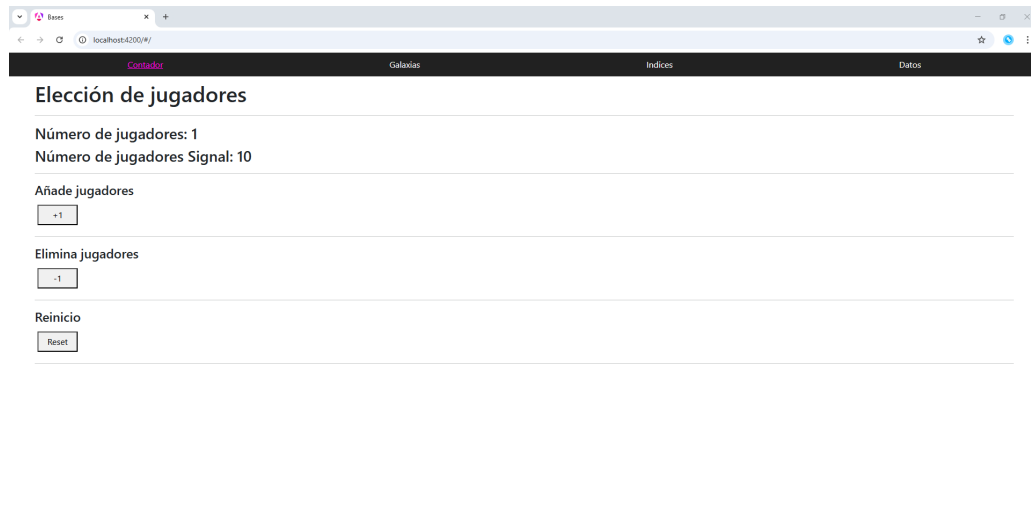
Mis primeros pasos en el aprendizaje de las herramientas de trabajo de Evenbytes está dominado por: *Angular*, *entorno GCP* y *GitHub*. Estas tres herramientas forman el pilar básico de la empresa. Angular es el framework principal utilizado para el desarrollo de aplicaciones web. Como *partners* de Google, el entorno GCP proporciona la infraestructura en la nube preferida por la empresa y GitHub que es un sistema de control de versiones ampliamente extendido.

2.2.1. Angular

Angular es un framework de desarrollo basado en *JavaScript* y *HTML*¹ para la creación de aplicaciones web dinámicas, que permite estructurar el código de manera modular y facilita la generación de interfaces interactivas. Naturalmente, una empresa dedicada en parte a la transformación digital requiere de personal capaz de implementar páginas web o aplicaciones para que los clientes hagan uso de ellas. Como ejemplo, uno de los productos de Evenbytes son los dispositivos de fichaje horario, por lo que los datos que se registran en el momento en el que un trabajador entra a su jornada laboral, además de ser tratados, se mostrarán para que la empresa empleadora pueda manejarlos, filtrarlos, etc.

El periodo dedicado al aprendizaje de este ítem fue inferior a una semana pero aun así el poder conocer el trabajo que supone la creación e implementación de páginas y aplicaciones fue satisfactorio y pude desarrollar ciertos ejemplos, como los formularios que se muestran a continuación:

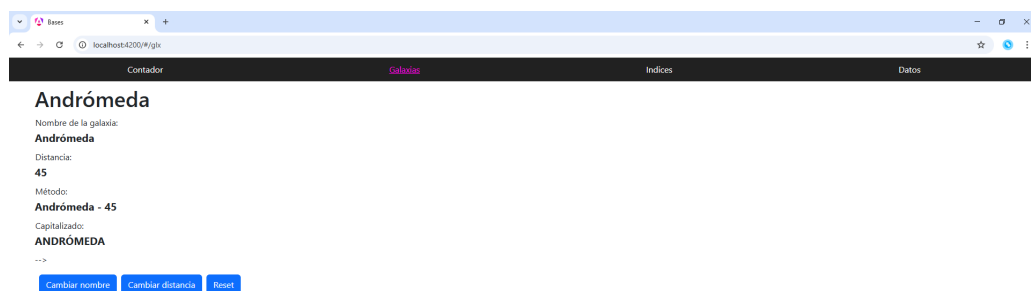
¹Técnicamente *TypeScript* pero esencialmente es lo mismo.



The screenshot shows a web browser window with the address bar displaying 'localhost:4200/#/'. The page has a dark navigation bar with four tabs: 'Contador' (active), 'Galaxias', 'Indice', and 'Datos'. The main content area is titled 'Elección de jugadores'. It displays the following information and controls:

- Número de jugadores: 1
- Número de jugadores Signal: 10
- Añade jugadores: A button with a '+' icon and the number '1'.
- Elimina jugadores: A button with a '-' icon and the number '1'.
- Reinicio: A button labeled 'Reset'.

Figura 2.1: En este se pueden agregar y disminuir *jugadores* del entorno.



The screenshot shows a web browser window with the address bar displaying 'localhost:4200/#/gix'. The page has a dark navigation bar with four tabs: 'Contador', 'Galaxias' (active), 'Indice', and 'Datos'. The main content area is titled 'Andrómeda'. It displays the following information and controls:

- Nombre de la galaxia: **Andrómeda**
- Distancia: **45**
- Método: **Andrómeda - 45**
- Capitalizado: **ANDRÓMEDA**
- Below the text, there is a small right-pointing arrow icon.
- At the bottom, there are three buttons: 'Cambiar nombre', 'Cambiar distancia', and 'Reset'.

Figura 2.2: Formulario que permite variar los datos de una galaxia restaurarlos por los iniciales.

Figura 2.3: En este entorno se permite agregar elementos a la lista de manera general o condicionada.

Figura 2.4: Formulario similar a la anterior pero en este caso permitía operar con los datos mostrándolos transformados.

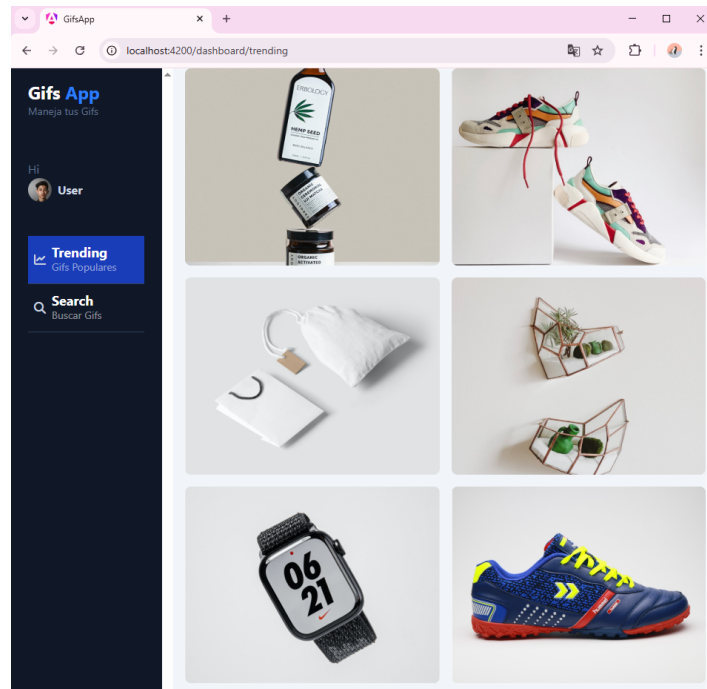
En las figuras² se muestran capturas de la primera página desarrollada, en la que se tenían diversos formularios, con sus *URLs* específicas. Los datos, inventados naturalmente, hacían referencia al nombre de galaxias, distancia, origen e identificador. La funcionalidad de la página era la creación de formularios cuyos datos se agregarán a listas de manera general, condicionada (en este caso particular por el origen o la distancia). Además, se añadió la posibilidad de exponer los datos, cambiarlos o restaurarlos.

En segundo lugar, se procedió a la creación de una aplicación más elaborada, que tuviera un menú interactivo como capa principal y que registrara en él ciertos datos como el usuario o una foto

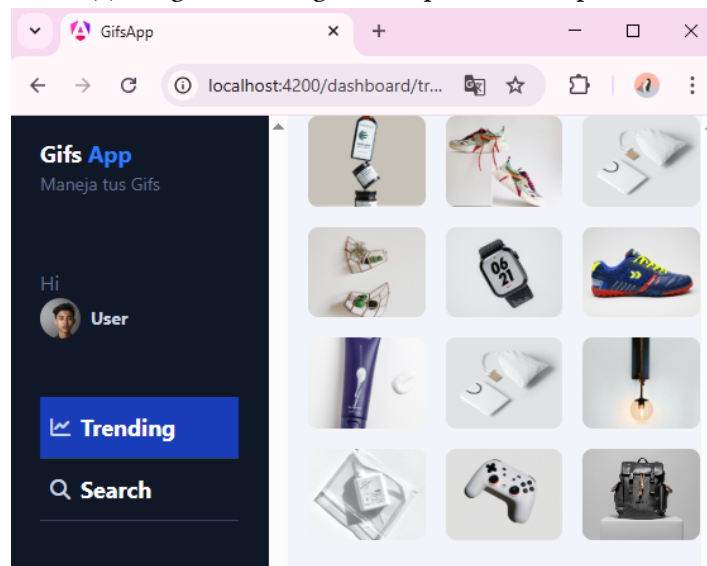
²La primera de ellas se corresponde con una prueba en la que se analizaba el comportamiento de distintas formas de definir variables en Angular y no tiene relación con las siguientes.

de perfil.

El navegador se comportaba de manera reactiva, adaptándose al tamaño de la pantalla, tal y como se muestra en la Figura 2.5. A la izquierda de las imágenes puede apreciarse el menú lateral, en el que aparecen los ítems (en este caso, dos) que permiten acceder al resto de modalidades de la página.



(a) Imagen de navegador en pantalla completa.



(b) Imagen de navegador en pantalla reducida.

Figura 2.5: Segunda aplicación web con menú interactivo.

El repositorio de *gifs* incorporado en la aplicación se organizaba mediante una pestaña denominada *trending*, que permitía ordenarlos. En este caso, como es evidente, no se podía disponer de un registro de las consultas de los usuarios, por lo que las entradas se organizaban de manera alfabética.

2.2.2. GCP

Este entorno proporciona la infraestructura en la nube necesaria para desplegar aplicaciones, almacenar datos y ejecutar servicios de manera flexible y segura. La empresa Evenbytes es, como ya se ha mencionado, *partner* de Google, por lo que brinda la posibilidad de trabajar con la mayor parte de funcionalidades de esta plataforma. Trabajar en la nube facilita el acceso compartido, la escalabilidad de los recursos y la automatización de procesos, lo que resulta especialmente valioso en proyectos donde intervienen varios miembros del equipo o que requieren alta disponibilidad.

En mi caso particular, implementé una *Cloud Function* encargada de realizar el proceso de extracción de información de una página web (llamado *web scraping*) y la generación de un fichero con los resultados. La *Cloud Function* permite ejecutar fragmentos de código en respuesta a eventos o invocaciones programadas, sin necesidad accionar explícitamente el programa. Desde Python realicé el código para conectarse a la página de Wikipedia sobre cocientes intelectuales por país, extraer las tablas relevantes y convertir los datos en un archivo CSV. El código de la función se desplegó en GCP mediante las herramientas de línea de comandos y quedó configurado para ejecutarse de forma automática a intervalos regulares.

Para esta última tarea hice uso de Cloud Scheduler, un servicio que permite programar tareas basadas en cron. La configuración del Scheduler se gestionó a través de Terraform. Gracias a esta combinación, cada ejecución de la función generaba automáticamente un nuevo bucket³ de almacenamiento en Cloud Storage, en el que se guardaba el fichero CSV con los datos actualizados.

A continuación se muestran fragmentos de código de la programación del *web scraping*, la planificación en Terraform y el resultado del CSV.

```
13 def web_scraping():
14     # Busco en cada país
15     filas = driver.find_elements(By.XPATH, '//*[@class="wikitable sortable jquery-tablesorter"]//tbody/tr')
16     print(f'Número de filas encontradas: {len(filas)}')
17     data = []
18
19     print("Comenzamos a buscar la información de las filas...")
20     # Las matrices
21     for i, fila in enumerate(filas):
22         try:
23             # Busco el elemento de la imagen
24             elemento_imagen = fila.find_element(By.XPATH, '//*[@img]')
25             # Voy a coger ahora el nombre del país
26             texto_completo = elemento_imagen.get_attribute('alt')
27             nombre = texto_completo.replace("Bandera de ", "")
28             # Quiero extraer los td ahora
29             td_completos = fila.find_elements(By.TAG_NAME, 'td')
30             # IQ en tercera columna por lo que lo imprimo directamente ahí
31             if len(td_completos) > 2:
32                 iq = td_completos[2].text
33                 position = td_completos[0].text
34                 data.append({
35                     'Position': position,
36                     'NombrePaís': nombre,
37                     'IQ': iq,
38                 })
39             else:
40                 print(f"La fila {i+1} no tiene suficientes columnas")
41         except Exception as error:
42             print(f"Error al procesar la fila {i+1}: {error}")
43
44     if driver:
45         driver.quit()
46         driver = None
47     print("Webdriver cerrado tras el scraping.")
48
49     # Creo el dataframe en pandas
50     df = pd.DataFrame(data)
51
52     if df.empty:
53         print("No se encontraron datos para exportar.")
54         # Aseguro: Respondo explícito para este caso
55         return "No se encontraron datos para exportar. El scraping pudo fallar.", 200
56
57     # Lo exporto a csv
58     csv_filename = f'iq_paises.csv'
59     csv_path = os.path.join('/tmp', csv_filename)
60     df.to_csv(csv_path, index=False, encoding='utf-8')
61     print(f"DataFrame guardado localmente en: {csv_path}")
```

(a) Fragmento del código de *web scraping* en Python [Listado 3.1].

```
1 provider "google" {
2   project = var.gcp_project_id
3   region  = var.gcp_region
4 }
5
6 # Obtener información del servicio de Cloud Run existente
7 data "google_cloud_run_service" "my_cloud_run_service" {
8   name     = var.cloud_run_service_name
9   location = var.gcp_region
10 }
11
12 # Crear el trabajo de Cloud Scheduler
13 resource "google_cloud_scheduler_job" "my_scheduler_job" {
14   name           = "invoke-iq-paises-alberto-daily"
15   description    = "Invoca el servicio de Cloud Run IQ Países diariamente"
16   schedule       = "0 9 * * 2,4"
17
18   http_target {
19     uri      = "https://iq-paises-998937693176.europe-west1.run.app"
20     http_method = "POST"
21
22     headers = {
23       "Content-Type" = "application/json"
24     }
25
26     oidc_token {
27       service_account_email = var.scheduler_service_account_email
28     }
29   }
30
31   time_zone = "Europe/Madrid"
32 }
```

(b) Ejemplo de configuración de Terraform para el *scheduler* [Listado 3.2].

Figura 2.6: Fragmentos código entorno GPC.

³Los buckets son contenedores que permiten almacenar objetos de manera segura y escalable. En este proyecto, cada bucket contenía los resultados de la extracción en formato CSV, de forma que quedaba un histórico de ejecuciones accesible a través de la consola de GCP o mediante las APIs correspondientes.

A modo de breve explicación de los códigos mostrados, para el caso del *web scraping* la idea principal es generar un navegador virtual al que se le manda a la página de la que queremos extraer los datos. De manera previa, se investiga el código HTML para saber la estructura de la página, de modo que creamos un bucle que busque los elementos que queremos y extraiga la información de ellos, es decir, si tenemos una estructura en la que todos los objetos tienen una propiedad llamada *nombre*, hacemos que nuestro código acceda a él para extraer todos los nombres de la página y guardarlos en un fichero.

Por otro lado, el scheduler es más sencillo, le introducimos los datos del proyecto, función a llamar... y definimos las órdenes bajo las que se ejecuta. En el código aparece en la línea 16 el fragmento `0 9 * * 2,4` que significa que se ejecute a las 9:00, *cualquier día* (numérico), *cualquier mes*, los *martes* y *jueves*.

2.2.3. GitHub

En último lugar estuvo el aprendizaje de GitHub, que actúa como sistema de control de versiones, permitiendo gestionar el historial de cambios en el código, coordinar el trabajo y asegurar que todas las contribuciones queden registradas de forma ordenada. Su uso es fundamental en un entorno profesional, ya que permite mantener la trazabilidad de los desarrollos, revisar modificaciones y garantizar la integridad de los proyectos a medida que evolucionan. Su estructura es muy sencilla y puede entenderse rápidamente pensando en grafos. Se crea una rama principal del proyecto y de ellas, cuando se quiere trabajar en una parte específica del mismo, salen hijas. De esta manera los cambios realizados se hacen fuera del proyecto principal, permitiendo guardar los cambios, avanzar únicamente en esa línea o abrir nuevas. Cuando una parte se da por revisada y acabada se unen ramas a la principal (padre), de manera que se va completando el trabajo. Los comandos básicos de Git son *commit*, que guarda los cambios realizados en el repositorio local; *pull*, que actualiza la copia local con las modificaciones del repositorio remoto; *push*, que sube los cambios locales al repositorio compartido; y *merge*, que combina diferentes ramas en una sola versión final.

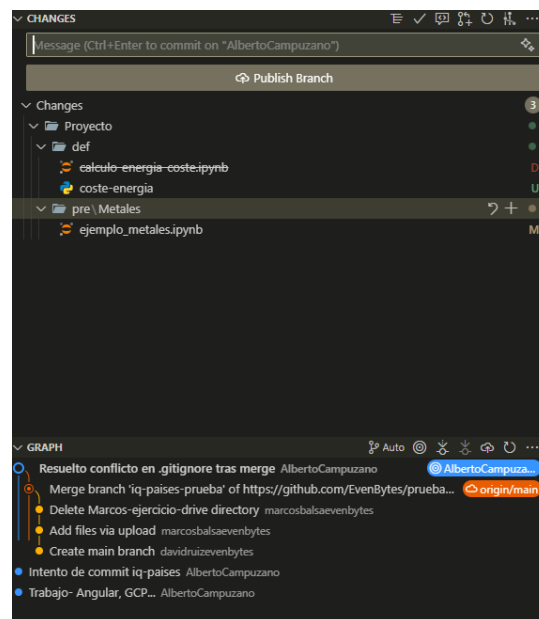


Figura 2.7: Imagen GitHub

Nota: Por motivos de confidencialidad, ciertos contenidos no serán mostrados. Dado que el objetivo de estas prácticas es el manifiesto del conocimiento aprendido y técnicas implementadas no causará un gran perjuicio en la comprensión de la misma.

2.3. Proyecto principal

Este trabajo trataba de la optimización del gasto energético de una empresa que contaba con hornos de fundición. Para ello, se tuvieron que realizar predicciones del precio de la electricidad mediante modelos estadísticos. Posteriormente, se modelizó el proceso de fundición llevado a cabo en la fábrica y por último se trató de optimizar la selección de los materiales a fundir, dependiendo del espacio disponible en fábrica y el precio de la luz en cada tramo horario, estimado previamente.

Por motivos de extensión, y porque es la parte a la que más tiempo he dedicado, la sección asociada a la predicción de precios de la luz será la más extensa y detallada. Respecto del modelado y proceso de optimización se explicará más brevemente, además de que pueden incluir información privada del cliente por lo que no se incidirá demasiado.

2.3.1. Predicción de precios de la luz

Para llevar a cabo un modelo predictivo de los precios de la luz para una empresa hemos de realizar una serie de pasos. En primer lugar, como es evidente, debemos conocer y entender qué es lo que queremos predecir, bajo qué condiciones y qué variables afectan al precio de la luz, tanto naturales, como puede ser la demanda, la generación de ciertos tipos de energía, como aquellas relacionadas con la legislación vigente. Posteriormente debemos decidir cómo se extraeran los datos necesarios. Puesto que estamos hablando de un proyecto real, la empresa espera un proceso automatizado por lo que se debe tener en cuenta una extracción periódica de datos.

Una vez planteado el contexto en el que vamos a movernos, debemos decidir qué modelos probar, conociendo sus ventajas y desventajas. Mi elección, basada en búsquedas de desarrollos similares, fue optar por tres modelos:

SARIMAX Modelo estadístico clásico

XGBoost Modelo de *machine learning*

TFT Modelo de *deep learning*

Por último, por requisitos del proyecto, la predicción se realizaba a un mes vista, lo cual como es lógico, no es óptimo. Sin embargo, esto era algo fijo por lo que los resultados esperados son mejorables, recortando el horizonte de predicción y reentrenado los modelos con mayor asiduidad.

2.3.1.1. Estudio y comprensión del mercado

Dado que la empresa es una electrointensiva no se rige por el régimen general del pequeño consumidor (PVPC). Para la toma de datos de la variable objetivo, el precio de la luz, se consultó el precio del mercado SPOT diario, que es el que regula la venta de luz al por mayor.

Para calcular exactamente, dentro de lo que podemos entender como *exacto* en un modelo de predicción, el precio de la luz debemos tener en cuenta tanto el precio de venta como los peajes.

Estos últimos son valor añadido a la factura de la luz que depende del tipo de empresa y de la hora de consumo. De cualquier forma, en lo que a nuestro modelo se refiere, únicamente nos interesa el precio del mercado SPOT diario, que viene dado de manera horaria.

2.3.1.2. Obtención y preparación de los datos

Los datos se obtuvieron a través de la API pública proporcionada por ESIOS (Sistema de Información del Operador del Sistema Eléctrico), que permite acceder a los precios horarios históricos, así como a otros indicadores del sistema eléctrico. De manera adicional al precio de la luz, y como se ha comentado previamente, se tomaron variables explicativas, es decir, que se cree que tienen relación y afectan al precio de la que queremos. En este caso, se tomó: La demanda real de consumo, la generación de energía eólica y solar.

En lo relativo a la preparación de los datos, salvando las diferencias en función de la implementación del modelo, el procedimiento fue similar: Una vez descargados los datos se formatean de manera que tengamos aquellos valores que queremos, es decir, el precio, demanda, etc. y la fecha en la que se dio. Además, de cara a refinar nuestro modelo, se introdujeron variables temporales como el día de la semana, día del mes o mes del año. Un estudio mucho más exhaustivo puede hacerse incluyendo más variables o con más combinaciones. En [1] puede verse como incluyen el precio del gas o la emisión de CO₂ entre otro, llevándoles a mejores resultados, aunque con mayor trabajo.

2.3.1.3. Análisis exploratorio

2.3.1.4. SARIMAX

Este es un modelo estadístico clásico para la predicción de series temporales. Su fortaleza radica en su capacidad para modelar explícitamente tres componentes clave de la serie:

- **Tendencia (I - Integrado):** Se logra a través de la diferenciación para hacer que la serie sea estacionaria.
- **Estacionalidad (S - Seasonal):** Captura patrones que se repiten en intervalos fijos (por ejemplo, diarios o semanales).
- **Variables (X):** Permite incluir predictores externos, como la demanda o la generación de energía.

El proceso consiste en:

- **Identificación:** El modelo SARIMAX viene caracterizado por los parámetros $(p, d, q)x(P, D, Q, s)$ donde:
 - p : Orden del componente autorregresivo (AR) no estacional. Indica el número de rezagos de la variable endógena que se incluyen en el modelo.
 - d : Orden de diferenciación no estacional. Representa la cantidad de veces que se deben diferenciar los datos para hacer la serie estacionaria.
 - q : Orden del componente de media móvil (MA) no estacional. Indica el número de rezagos del error del pronóstico que se incluyen en el modelo.
 - P : Orden del componente autorregresivo (AR) estacional.

D : Orden de diferenciación estacional.

Q : Orden del componente de media móvil (MA) estacional.

s : Período de estacionalidad, en este caso vemos que $s = 24$ de manera natural al tratarse de precios horarios.

- **Estimación:** Una vez definidos los órdenes, se ajusta el modelo al conjunto de datos de entrenamiento. El modelo aprende los coeficientes que ponderan la importancia de los valores pasados, los errores de pronóstico pasados y las variables exógenas.

La predicción con SARIMAX es un proceso iterativo o recursivo:

- **Entrenamiento:** Al igual que con otros modelos, se reentrena un modelo SARIMAX final con su orden óptimo utilizando todos los datos históricos disponibles para maximizar la información aprendida.
- **Predicción:** Para predecir el horizonte futuro (ej. las próximas 24 horas), el modelo:
 - Predice el primer paso ($t+1$) utilizando todos los datos históricos reales.
 - Para predecir el segundo paso ($t+2$), utiliza los datos históricos y la predicción que acaba de hacer para $t+1$ como si fuera un dato real.

Este proceso se repite para cada punto en el horizonte de predicción.

En nuestro caso, para determinar los órdenes de los parámetros del modelo, se analizan las funciones de autocorrelación (ACF) y autocorrelación parcial (PACF). La ACF mide la correlación de la serie consigo misma en diferentes rezagos, mientras que la PACF mide esta misma correlación eliminando la influencia de los rezagos intermedios.

En las siguientes imágenes podemos ver los gráficos de ACF y PACF para la serie de precios de la luz, lo que nos permite identificar los patrones de estacionalidad y los órdenes de los modelos.

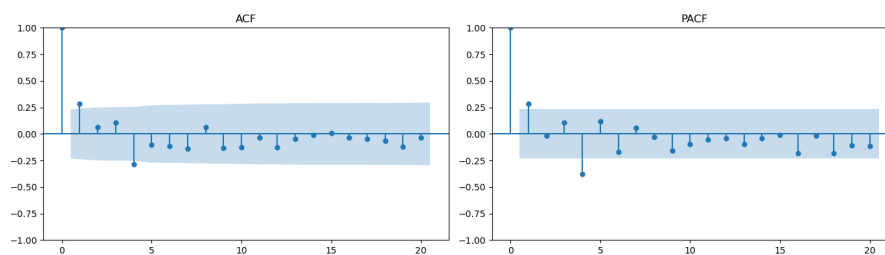


Figura 2.8: ACF y PACF de la serie de precios de la luz sin diferenciar. Se observa una clara estacionalidad y una autocorrelación que decae lentamente, lo que sugiere la necesidad de diferenciación.

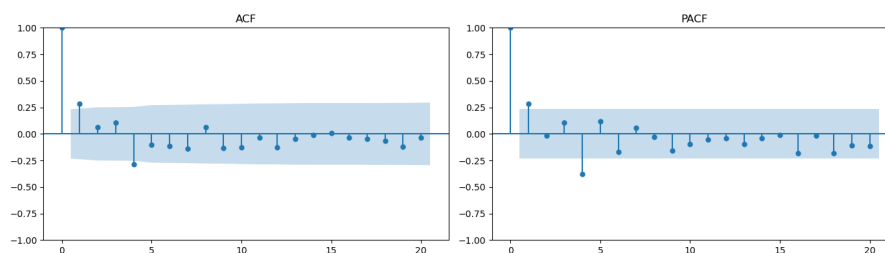


Figura 2.9: ACF y PACF de la serie de precios con diferenciación estacional. Aunque la estacionalidad ha sido eliminada, la autocorrelación no estacional aún persiste.

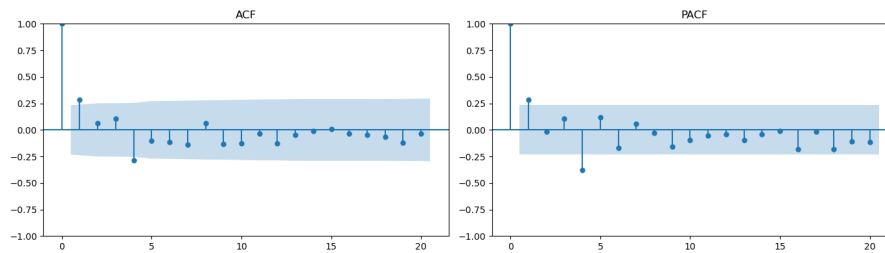


Figura 2.10: ACF y PACF de la serie con diferenciación doble. Ahora la serie es estacionaria y los rezagos significativos nos indican los posibles órdenes p, q, P, Q .

Basado en el análisis de las funciones de autocorrelación, se han preseleccionado los siguientes tres modelos con sus respectivos parámetros:

- **Modelo 1: SARIMAX(1,1,1)x(1,1,1,24):** Se elige este modelo porque los picos en los gráficos de ACF y PACF de la serie doblemente diferenciada son significativos en el primer rezago no estacional y estacional. Esto sugiere un orden de 1 para los componentes AR, MA, y sus contrapartes estacionales.
- **Modelo 2: SARIMAX(0,1,1)x(0,1,1,24):** Este modelo se considera debido a que, en el gráfico PACF, los picos son más pronunciados en los primeros rezagos, lo que podría indicar la preponderancia de los términos de media móvil.
- **Modelo 3: SARIMAX(2,1,0)x(2,1,0,24):** Se opta por este modelo como alternativa, ya que los gráficos de ACF y PACF también muestran una posible influencia de los segundos rezagos, tanto estacionales como no estacionales.

Estos tres modelos serán evaluados y comparados utilizando métricas como el MAE y el RMSE para determinar cuál de ellos ofrece un mejor ajuste a los datos.

2.3.1.5. XGBoost

Este modelo, basado en árboles de decisión, también es relativamente sencillo de implementar aunque, como es habitual en este tipo de modelos, presenta menor interpretabilidad respecto a lo que ocurre dentro del algoritmo. Su robustez radica en la capacidad de aprender patrones complejos mediante un enfoque de ensamblado (*ensemble*) y mejora por aproximación secuencial de errores. Además, se ve reforzado por técnicas de regularización y control del *overfitting*, lo que permite obtener modelos generalizables y eficientes en predicción. Entre las características más destacadas de XGBoost:

- **Regularización integrada:** A diferencia de otros modelos basados en árboles, XGBoost incluye regularización L_1 y L_2 como parte de la función de coste, permitiendo controlar la complejidad del modelo y reducir el riesgo de *overfitting*.
- **Manejo de valores nulos:** Tiene un tratamiento nativo para valores faltantes, aprendiendo automáticamente la mejor dirección en los nodos de decisión.
- **Velocidad y eficiencia:** Permite la construcción de árboles de manera paralela. Además, utiliza una aproximación de segundo orden (Taylor) sobre la función de pérdida, incorporando tanto el gradiente como el hessiano. Esto no solo acelera la convergencia del modelo, sino que también permite trabajar con funciones de pérdida más generales, si bien lo más común es trabajar con el error cuadrático medio.

- **Importancia de variables:** Aunque la interpretabilidad del modelo global es limitada, permite extraer métricas de importancia relativa de cada predictor.

El entrenamiento se realiza mediante un enfoque de *boosting*:

- **Inicialización:** Se parte de una predicción base (por ejemplo, la media de la variable a predecir).
- **Aprendizaje secuencial:** Cada árbol posterior intenta corregir los errores residuales cometidos por el conjunto acumulado hasta el momento.
- **Optimización de la función de pérdida:** Se utiliza una aproximación de segundo orden (gradiente y hessiano) para mejorar la precisión de la optimización.
- **Regularización:** Durante el proceso se penaliza la complejidad del modelo (profundidad de los árboles, número de nodos) para evitar *overfitting*.

La predicción con XGBoost se realiza paso a paso, como la SARIMAX.

2.3.1.6. TFT

El Temporal Fusion Transformer (TFT) es una arquitectura de deep learning diseñada específicamente para el forecasting de series temporales multi-horizonte. Combina mecanismos de atención (propios de los Transformers) con redes neuronales recurrentes y capas de gating. Sus características clave son:

- **Manejo de Múltiples Tipos de Variables:** Integra de forma nativa variables estáticas, entradas conocidas en el futuro y variables observadas en el pasado (ej. precio, demanda).
- **Aprendizaje de Patrones a Múltiples Escalas:** Puede identificar patrones temporales tanto a corto como a largo plazo simultáneamente.

El entrenamiento consiste en:

- **Preparación de los Datos:** Los datos se estructuran en "ventanas" de entrada de una longitud fija para predecir una "ventana" de salida (ej. las próximas 24 horas).
- **Ajuste del Modelo:** Se entrena una única red neuronal que aprende las interacciones entre todas las variables a lo largo del tiempo.

La predicción con TFT se realiza en un solo paso:

- **Reentrenamiento:** Se ajusta el modelo final con todos los datos históricos disponibles.
- **Predicción Multi-Horizonte Directa:** Para generar la predicción, se le proporciona al modelo la última ventana de datos históricos disponibles. En una única pasada hacia adelante (forward pass), el modelo genera directamente todas las predicciones para el horizonte completo.

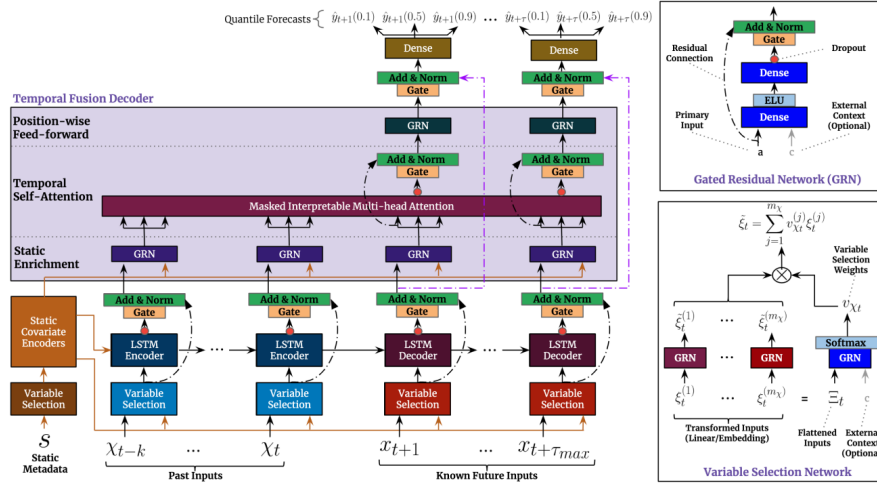


Figura 2.11: Arquitectura del TFT [2]

Respecto a la decisión de hiperparámetros de nuestro modelo TFT es más complicada. Este tipo de modelos son más exigentes en el formato de entrada de los datos y más sensible a las configuraciones, si bien lo compensa con un gran rendimiento. Por ello no me adentraré mucho en ello:

- **n estimators**
- **max depth**
- **learning rate**

2.3.1.7. Resultados

2.3.2. Modelo físico

2.3.2.1. Situación simplificada: Primera aproximación

Dado un metal conocido (X) de temperatura de fusión T_f , calor específico en los estados sólido y líquido, c_{sol} y c_{liq} respectivamente, entonces a la hora de conocer cuánta energía será necesaria tendremos que tener en cuenta:

- I El metal X a una temperatura inicial T_0 debe ser calentado hasta la temperatura de fusión T_f , esto dependerá del *calor específico* del material en el estado *sólido*, que viene a ser el *coste* energético para aumentar la temperatura del mismo.
- II Una vez alcanzado dicho objetivo el material cambia de estado de la materia requiriendo una nueva cantidad de calor dependiente del *calor latente* del material, denominado como L_{s-l} .
- III Finalmente supondremos que seguiremos elevando la temperatura por lo que, de manera análoga al paso I, tendremos que tener en cuenta esta energía. Llamaremos a este límite T_c

Para realizar dichos cálculos, haremos uso de las fórmulas de calor para calor específico y transiciones de fase.

$$Q_{pf} = m \cdot c_{sol} \cdot (T_f - T_0). \quad (2.1)$$

$$Q_f = m \cdot L. \quad (2.2)$$

$$Q_e = m \cdot c_{liq} \cdot (T_c - T_f). \quad (2.3)$$

De este modo, $Q = Q_{pf} + Q_f + Q_e$. Finalmente, deberíamos tener en cuenta que los hornos no son ideales y tenemos una cierta pérdida de energía. Este hecho, cuantificado por el rendimiento $\eta \in [0, 1]$, provoca un aumento en el calor total necesario: $Q^{\text{tot}} = Q/\eta$.

2.3.2.2. Situación general

Nos situamos ahora ante el problema general, en el que partimos de temperaturas iniciales cercanas a la ambiente ($T_0 \simeq 25^\circ\text{C}$) mientras que llegamos por lo menos a $T = 1450^\circ\text{C}$. En este caso tendremos que aplicar la formulación general:

$$Q = m \cdot \int_{T_a}^{T_b} c(T) dT, \quad (2.4)$$

donde m es la masa del compuesto X y $c(T)$ es la función del calor específico con la temperatura. Para poder operar con esta expresión sería necesario consultar datos tabulados sobre el material y una vez obtenidos los datos de c para distintas temperaturas interpolamos obtener una función con la que calcular, de manera aproximada, el calor en la Ecuación (2.4). De este modo llegaríamos a una situación similar a la descrita en la Apartado 2.3.2.1, obteniendo Q^{tot} . En ambos casos, como se va a tener desarrollo polinómico de $c(T)$, podemos calcular

$$Q = m \cdot \int_{T_a}^{T_b} c(T) dT = m \cdot \int_{T_a}^{T_b} [a_0 + a_1 T + \dots + a_n T^n] dT = \quad (2.5)$$

$$= m \cdot \left[a_0 T + \frac{a_1}{2} T^2 + \dots + \frac{a_n}{n+1} T^{n+1} \right] \Big|_{T_a}^{T_b} = m \cdot p(T_a, T_b), \quad (2.6)$$

donde sustituyendo obtenemos el valor deseado. Para el primer caso, tenemos que T_a coincide con la temperatura *ambiente* de la pieza mientras que T_b con la de fusión del material X. Por otro lado, para la etapa final, la temperatura de fusión coincide con la inicial y la final, supondremos que es 1450°C para todas las piezas.

2.3.2.3. Traducción matemática

La situación anterior debe ser convenientemente planteada como restricciones, de cara a la resolución del problema de optimización con el que nos encontramos. Aunando lo descrito en la Apartado 2.3.2.2 podemos escribir que el calor Q necesario para la pieza X es:

$$Q = Q_{pf} + Q_f + Q_e + Q' = m \cdot (p_I + L + p_{III}) + Q', \quad (2.7)$$

Donde el término Q' es la energía debida a un requerimiento del fabricante, tanto de precalentamiento como de pos-fundición.

Finalmente, debería considerarse el rendimiento η de los hornos. Supondremos, salvo que se indique lo contrario, que ambos hornos presentan el mismo rendimiento. Adicionalmente, dado que este valor es una constante, tenemos que $\min \eta f(x) = \eta \min f(x)$ por lo que prescindiremos de este valor puesto que no influye en la función objetivo y únicamente debería ser considerado en última instancia, a la hora de predecir el gasto estimado. Ya por último, se ha obtenido el calor Q con unidad (J) *julios*, mientras que habitualmente el coste eléctrico se da en €/kWh por lo que deberemos realizar la debida conversión. Esta, al igual que pasa con el rendimiento, influye en el valor numérico final pero no en el problema a optimizar.

2.3.3. Optimización de la superficie de enfriamiento

El presente problema aborda la distribución de las piezas metálicas recién fundidas sobre la superficie de enfriamiento disponible en la nave. Claramente, el objetivo es maximizar el número total de piezas que pueden colocarse simultáneamente. Esta superficie cuenta además con restricciones físicas relacionadas con la seguridad y operatividad del proceso, como márgenes laterales y pasillos interiores obligatorios de al menos 1.5 metros de ancho.

Cada pieza fundida es clasificado según su diametro y para su manipulación ha sido previamente colocada en una caja cuya dimensión se determina en función del tipo correspondiente. Estas cajas se disponen en planta sobre la superficie, sin posibilidad de apilarse. Una vez colocadas, las cajas no pueden solaparse ni invadir los pasillos o márgenes de seguridad. Es decir, deben respetarse las distancias mínimas entre ellas y con respecto a los bordes del área de enfriamiento. Esto impone restricciones geométricas estrictas sobre su posicionamiento.

El objetivo final es obtener una disposición óptima que maximice la cantidad total de piezas colocadas, permitiendo una planificación eficiente del enfriado. Dado el carácter discreto y combinatorio del problema, este puede abordarse mediante algoritmos de optimización específicos como métodos de empaquetado 2D, algoritmos genéticos o formulaciones de programación entera si se requiere una solución exacta.

2.4. Relación entre formación recibida y actividades

Con salvedad de la etapa de formación, creo que ha quedado de manifiesto la íntima relación entre los conocimientos adquiridos durante el grado en matemática, y con menor relevancia en física, y las tareas desempeñadas durante este tiempo. Intentando particularizar en asignaturas: Programación, impartida durante el segundo cuatrimestre del primer curso, las asignaturas de estadística, principalmente Inferencia estadística, del segundo cuatrimestre de tercero (cuarto en mi caso debido a la distribución del Doble Grado) y en cuanto a física, claramente, Termodinámica.

Aunque a priori pueda parecer que no hay mucha relación entre modelos de redes neuronales, machine learning y deep learning, una vez investigas y te adentras en los mecanismos que hacen que estas tecnologías (bases de la IA) funcionen, rápidamente observas que la estadística son clave en estas áreas. Esta característica, que quizás queda algo más alejada del público general, es uno de los mayores atractivos para personas con formación en matemáticas que se sienten atraídas por una de la que probablemente sea la revolución del siglo: La IA.

2.5. Atención y asesoramiento recibido

Durante mi estancia en la empresa el trato y atención han sido inmejorables. En este tiempo me he encontrado bajo la supervisión de mis dos compañeros del equipo de Data, ambos analistas de datos, si bien cabe remarcar que el apoyo global de la plantilla ha estado presente.

En todo el periodo he estado presente y participado activamente en reuniones de proyectos, tomas de decisión junto a los responsables (CEO, CTO y COO), de manera que además del aprendizaje

técnico he sido capaz de conocer el funcionamiento de situaciones habituales en el entorno laboral.

Capítulo 3

Conclusiones

El programa de prácticas llevado a cabo estos dos meses ha sido excelente, más que satisfactorio. Como se ha comentado en la introducción y varias veces a lo largo de esta memoria, mi principal interés estaba en el acercamiento al mundo laboral, de manera que pudiera conocer más de cerca el sector. En estos dos meses y medio me he adentrado dentro de un mundo que desconocía completamente más allá del interés superficial que suscita la IA y sus aplicaciones a situaciones de mercado.

Todas las habilidades, tanto técnicas como personales, adquiridas en este tiempo serán de una enorme utilidad para toda la vida. Además, el aprendizaje llevado a cabo en tareas como el trabajo en equipo, los plazos de entrega, las decisiones en proyectos y el trato con clientes han permitido que tenga una primera toma de contacto con el mundo profesional.

A modo de resumen y para finalizar: Las primeras semanas, en las que mi formación estuvo enfocada en un ámbito que desconocía totalmente y mi posterior inclusión en un grupo de trabajo ajeno tanto a la formación como a mi conocimiento, supusieron unas semanas de dureza ya que todo era nuevo. Progresivamente fui aprendiendo y adaptandome al flujo de trabajo, viendome incluido en diversos proyectos, entre ellos el principal expuesto aquí, de gran dificultad tanto en lo técnico como en su implementación para el cliente. Sin embargo, esta situación me ha enseñado a trabajar de manera autónoma, documentarme y asumir el liderazgo para la toma de decisiones, confiando en mi trabajo y conocimientos, de manera que he podido sacar adelante uel trabajo.

Apéndice

Incluyo a continuación informacion que pudiera ser relevante, completando a lo expuesto previamente que, por extensión, se ha omitido.

3.1. Codigos formación

```
1 from flask import Flask, request
2 from selenium import webdriver
3 from selenium.webdriver.common.by import By
4 import json
5 import os
6 import pandas as pd
7 from google.cloud import storage
8 from google.oauth2 import service_account
9
10 app = Flask(__name__)
11
12 @app.post("/")
13 def web_scraping():
14     """
15     Funcion que accede a la pagina de la wikipedia que contiene el
16     listado por paises de promedios de iq
17     extrae de ellas el pais e iq y crea un bucket con un csv con estos
18     datos
19     """
20     driver = None
21
22     try:
23         print("Cargamos las configuraciones del webdriver")
24         chrome_options = webdriver.ChromeOptions()
25         chrome_options.add_argument("--headless")
26         chrome_options.add_argument("--disable-gpu")
27         chrome_options.add_argument("window-size=1024,768")
28         chrome_options.add_argument("--no-sandbox")
29         chrome_options.add_argument("--disable-dev-shm-usage")
30
31         driver = webdriver.Chrome(options=chrome_options)
32         print("WebDriver inicializado con exito.")
33
34         # Cargo la pagina en la que quiero scrappear
```

```

33 driver.get('https://es.wikipedia.org/wiki/
    CI_y_la_riqueza_de_las_naciones')
34 print("Pagina cargada con exito.")
35
36 # Busco en cada fila
37 filas = driver.find_elements(By.XPATH, '//*[@class = "
    wikitable sortable jquery-tablesorter"]//tbody//tr')
38 print(f"Numero de filas encontradas: {len(filas)}")
39 data = []
40
41 print("Comenzamos a buscar la informacion de las filas...")
42 # Las recorro
43 for i, fila in enumerate(filas):
44     try:
45         # Tomo el elemento de la imagen
46         elemento_imagen = fila.find_element(By.XPATH, '//*[@img')
47         # Voy a coger ahora el nombre del pais
48         texto_completo = elemento_imagen.get_attribute('alt')
49         nombre = texto_completo.replace("Bandera de ", "")
50         # Quiero extraer los iq ahora
51         td_completos = fila.find_elements(By.TAG_NAME, 'td')
52         # IQ en tercera columna por lo que lo printeo
           directamente ahi
53         if len(td_completos) > 2:
54             iq = td_completos[2].text
55             posicion = td_completos[0].text
56             data.append({
57                 'Posicion': posicion,
58                 'NombrePais': nombre,
59                 'IQ': iq,
60             })
61         else:
62             print(f"La fila {i+1} no tiene suficientes columnas")
63     except Exception as error:
64         print(f"Error al procesar la fila {i+1}: {error}")
65
66 if driver:
67     driver.quit()
68     driver = None
69 print("WebDriver cerrado tras el scraping.")
70
71 # Creo el dataframe en pandas
72 df = pd.DataFrame(data)
73
74 if df.empty:
75     print("No se encontraron datos para exportar.")
76     # Asegurado: Retorno explicito para este caso
77     return "No se encontraron datos para exportar. El scraping
           pudo fallar.", 200
78
79 # Lo exporto a csv

```

```

80     csv_filename = 'iq_paises.csv'
81     csv_path = os.path.join('/tmp', csv_filename)
82     df.to_csv(csv_path, index=False, encoding='utf-8')
83     print(f"DataFrame guardado localmente en: {csv_path}")
84
85     # Compruebo el tipo de dato
86     print(df.dtypes)
87
88     # Cloud storage
89     # Nombre del JSON
90     archivo_key = 'sandbox_estudiantes_key.json'
91     # Nombre del bucket
92     nombre_bucket = 'test_tablas_alberto'
93     # Nombre del archivo CSV en GCS (usamos el mismo nombre que el
94     local)
95     iq_paises_cgs = csv_filename
96     # Creo las credenciales
97     credenciales = service_account.Credentials.
98         from_service_account_file(archivo_key)
99     # Inicializo con las credenciales
100    storage_cliente = storage.Client(credentials=credenciales)
101    # Tomamos la referencia del bucket
102    bucket = storage_cliente.bucket(nombre_bucket)
103    # Sube el archivo al bucket de GCS
104    blob = bucket.blob(iq_paises_cgs)
105    blob.upload_from_filename(csv_path)
106    print(f"Archivo '{iq_paises_cgs}' subido exitosamente a 'gs://{nombre_bucket}/{iq_paises_cgs}'")
107    # Elimino la ruta al csv
108    os.remove(csv_path)
109    print(f"Archivo local '{csv_path}' eliminado.")
110    return "Web scraping y subida a GCS completados exitosamente.", 200
111
112    except FileNotFoundError:
113        error_msg = f"Error: No se encontro el archivo de clave JSON '{archivo_key}'. Asegurate de que esta en la raiz del proyecto Docker."
114        print(error_msg)
115        return error_msg, 500
116    except Exception as error:
117        error_msg = f"Error inesperado durante la ejecucion: {error}"
118        print(error_msg)
119        return error_msg, 500
120    finally:
121        # Si driver != None lo cerramos, por si ha habido algun fallo y no se cierra
122        if driver:
123            driver.quit()
124            print("WebDriver cerrado en el bloque finally final.")

```

```

124
125 if __name__ == "__main__":
126     app.run(host="0.0.0.0", port=8080, debug=True)

```

Listing 3.1: Código de web scrapping completo.

```

1 provider "google" {
2     project = var.gcp_project_id
3     region  = var.gcp_region
4 }
5
6 # Obtener informacion del servicio de Cloud Run existente
7 data "google_cloud_run_service" "my_cloud_run_service" {
8     name      = var.cloud_run_service_name
9     location  = var.gcp_region
10 }
11
12 # Crear el trabajo de Cloud Scheduler
13 resource "google_cloud_scheduler_job" "my_scheduler_job" {
14     name          = "invoke-iq-paises-alberto-daily"
15     description   = "Invoca el servicio de Cloud Run IQ paises diariamente"
16     schedule      = "0 9 * * 2,4"
17
18     http_target {
19         uri          = "https://iq-paises-998937693176.europe-west1.run.app"
20         http_method  = "POST"
21
22         headers = {
23             "Content-Type" = "application/json"
24         }
25
26         oidc_token {
27             service_account_email = var.scheduler_service_account_email
28         }
29     }
30
31     time_zone = "Europe/Madrid"
32 }

```

Listing 3.2: Código de Terraform para el Google Scheduler.

3.2. Código SARIMAX

hola

3.3. Código XGBoost

hola

3.4. Código TFT

Bibliografía

- [1] N. V. Muñoz, *Previsión del Precio SPOT en el Mercado Eléctrico Español con Redes Neuronales*, 2024. dirección: https://oa.upm.es/82722/3/TFG_NICOLAS_VEGA_M.pdf.
- [2] B. Lim, S. O. Arik, N. Loeff y T. Pfister, *Temporal Fusion Transformers for Interpretable Multi-horizon Time Series Forecasting*, 2020. arXiv: 1912.09363 [stat.ML]. dirección: <https://arxiv.org/abs/1912.09363>.