Universidad de Valladolid

Escuela de Ingeniería Informática de Valladolid

TRABAJO FIN DE Máster

Máster en Ingeniería Informática
Big Data

Aplicación de técnicas Big Data a sistemas de climatización

Autor:
An Chen
Tutor:
Quiliano Isaac Moro Sancho

Agradecimientos

En primer lugar, quiero agradecer a toda mi familia y sobre todo a mis padres por la confianza que han depositado en mí y el apoyo constante que ha hecho posible la culminación de este proyecto.

A mis amigos por haberme acompañado en los buenos momentos y haberme ayudado a esquivar los malos momentos.

Quiero mostrar mi gratitud a mis tutores por el tiempo dedicado para guiarme a largo del proyecto. Siempre están allí para resolver cualquier duda que tuviera. Sin duda alguna sus orientaciones y sus ayudas han sido claves para la realización de este proyecto.

También me gustaría agradecer la Fundación CIDAUT por la oportunidad de trabajar y formar profesionalmente en la empresa y a mis compañeros del trabajo por sus ayudas incondicionales y por los buenos y malos chistes que han ayudado en los momentos de estrés.

Por último, gracias a todas las personas que han ayudado de forma directa o indirecta.

Contenido

Figuras

Gráficas

Esquema de contenidos

Para poner en práctica mis conocimientos adquiridos durante el curso de master, en este trabajo fin de Máster analizaré y trataré con técnicas Big Data el conjunto de datos recopilados de los sistemas de climatización[1] proporcionado por la Fundación CIDAUT.

Este TFM consta de dos partes: una primera parte focalizado en el análisis de los datos, y una segunda parte centrado en el entrenamiento de modelos predictivos para estimar el consumo de las calderas.

La primera parte se inicia con la carga de datos en los servidores de almacenamiento de Amazon (S3) y continua con la extracción de informaciones a partir de los datos brutos. Posteriormente se muestran en graficas los resultados del análisis mediante la plataforma de visualización Tableau. Esto implica la creación de 3 paneles de mandos que desempeñan 3 funciones diferentes:

- **Indicadores de producción.**
- **Indicadores de mantenimiento predictivo.**
- **Indicadores de caracterización demanda.**

En la segunda parte se prosigue con el entrenamiento de modelos predictivos. Para este fin se ha seleccionado 3 algoritmos (regresión lineal, regresión con bosque aleatorio y regresión con árbol de decisión reforzado con gradiente), los cuales requieren que los datos sean procesados y que los modelos sean probados mediante validación cruzada para hallar la mejor configuración de los parámetros. Esta parte del trabajo finaliza con la generación de 3 modelos predictivos capaces de estimar el consumo de las instalaciones.

[1] Los sistemas climatización consisten un conjunto de calderas y sus componentes informáticos de control y gestión.

1. INTRODUCCIÓN Y OBJETIVOS

El presente Trabajo Fin de Máster tiene como objetivo la aplicación de mis conocimientos adquiridos de Big Data durante el Máster en un entorno real que es proporcionado por la Fundación CIDAUT, El trabajo consiste en el desarrollo de una aplicación global para gestionar y controlar la cadena de valor de la biomasa en su uso energético, en concreto para la utilización de biocombustibles sólidos en combustión directa para la generación térmica. Mediante la investigación en la hibridación de sensores y tecnologías conectadas e interrelacionadas entre sí con la computación en la nube (cloud computing), así como con el desarrollo de protocolos de intercambio y manipulación de datos sobre cuadros de mando, se pretende lograr una estrategia que maximiza el rendimiento de los equipos.

La Fundación CIDAUT es un centro de investigación centrado en el sector del transporte y la energía. La monitorización de los sistemas de climatización es uno de sus proyectos de investigación donde recoge datos de la cadena de valor: de logística y suministro de biomasa, de los sistemas generadores térmicos y de los sistemas de climatización; a partir de servicios de internet creados desde los que enviar y recibir información mediante paneles de control y aplicaciones móviles, junto con un sistema de telegestión de los equipos generadores térmicos, y una adquisición de información de la demanda térmica.

Todos estos datos son enviados a la nube Ciderplus cloud (servidores privados dedicados al almacenamiento de datos y monitorización del sistema de climatización) donde son procesados para detectar, registrar y enviar alertas. Posteriormente son utilizados para generar de modelos de clasificación de los niveles con los que se puede gestionar sistemas mediante informes y cuadros de mando.

El desarrollo de estos modelos de clasificación y algoritmos de aprendizaje automático, supone una ventaja importante para el proceso de la toma de decisiones. Por ejemplo, en el presente Trabajo Fin de Máster se utilizarán estos modelos para estimar las demandas de combustibles.

1.1. Descripción general del proyecto

El uso de biomasa, en forma de pellets o astilla, es cada vez más extendido en la generación térmica, con las ventajas de balance neutro de CO_2 e independencia de los combustibles fósiles,

además de unos precios de combustible cada vez más competitivos respecto a los combustibles fósiles.

Para emplear esta energía renovable, sin sufrir pérdida de prestaciones ni de eficiencia, es necesario instalar sistemas de combustión de tecnología avanzada que además empleen herramientas informáticas para su buen control y seguimiento de operación. No hay que olvidar que el empleo de biomasa para generar energía térmica conlleva una serie de acciones de mantenimiento particulares como: limpieza de cenizas, mayor número de accionamientos para el trasiego de la biomasa, control de la calidad del combustible, así como una logística de suministro de la biomasa donde su control y eficiencia son determinantes para obtener buenos resultados en la explotación de este tipo de instalaciones.

Por tanto, el uso de tecnología TIC (Tecnologías de la Información y la Comunicación) en todos los procesos que se incluyen en la cadena de valor de la biomasa en energía térmica para este tipo de instalaciones, hace que se mejore la eficiencia del uso de biomasa, se simplifica las gestiones de mantenimiento y logística de aprovisionamiento y se genera conocimiento de gran valor a partir de los datos recopilados.

1.2. Objetivos del proyecto

Con el fin de crear un panel de mandos con el que se puede visualizar la evolución de los indicadores de la eficiencia de las calderas y extraer conocimientos de los datos recopilados se ha establecido los siguientes objetivos:

- Creación de un Panel de mandos que muestran la evolución eventual de los indicadores:
    - Rendimiento en modo bajo demanda: Indicador de la eficiencia del sistema de combustión.
    - Rendimiento promedio: Indicador de eficiencia, tanto del sistema generador como de la estrategia de operación. La evolución de este parámetro permite detectar el deterioro del equipo.
    - Consumo diario de cada caldera.
    - Tiempo dedicado en cada modo de funcionamiento. (Apagado, Arranque, Bajo demanda, Calentamiento, Stand-By)
    - Relación entre potencia media diaria y potencia máxima: Indicador de porcentaje de exigencia de los equipos para cada instalación.
    - Tiempo de promedio de arranque y de apagada: A través de este parámetro permite detectar la presencia de escoria en el hogar o ensuciamiento del orificio de encendido.

- Número de arranques y paradas diarias.
- Diferencia máxima diaria entre la temperatura de salida de la caldera y temperatura de inercia máxima en modo bajo demanda.
- Diferencia máxima diaria entre temperatura de retorno de la caldera y temperatura mínima en modo bajo demanda.

- Basándose en los datos registrados que recogen las características de cada equipo y la situación meteorológica de la población, se generan modelos predictivos con el fin de estimar el consumo provisional de cada instalación.

1.3. Infraestructura del sistema

En todas las instalaciones se disponen de un conjunto de sensores que recogen datos con una frecuencia de un registro por cada 60 segundos y los envían a los servidores privados de CIDAUT (Ciderplus cloud) mediante unos dispositivos Raspberry.

Desde los servidores privados de CIDAUT se realiza la ingestión de los datos al sistema de almacenamiento persistente de Amazon (Amazon S3), desde donde un clúster de Apache Spark extrae los datos para procesarlos y convertirlos posteriormente en informaciones legibles para la plataforma de visualización (Tableau), estos datos generados son almacenados de nuevo en el S3 con el fin que lograr la independencia con respecto al clúster de Apache Spark.

El clúster de Apache Spark está formado por 3 instancias de las cuales una instancia es maestra y otras dos instancias son esclavas. Todas estas instancias son de tipo m4.large que está formado por 2 CPU’s virtuales y cuenta con 8 GB de memoria RAM.

Con los datos suministrados por el clúster de Apache Spark y la plataforma de visualización, Tableau, se generan un panel de mandos con el que los usuarios pueden monitorizar la evolución eventual de cada instalación, de esta manera se pretende que el sistema sirva como apoyo para los usuarios en sus tomas de decisiones.

2. ANÁLISIS

En el capítulo anterior, grosso modo, se ha descrito los componentes y el funcionamiento del sistema informático que están siendo utilizado actualmente por la Fundación CIDAUT. En este capítulo se analizará el software utilizado para la monitorización y se establecerá el alcance del proyecto actual, así como las líneas generales que seguirá el desarrollo.

2.1. Situación actual

Actualmente la Fundación CIDAUT recogen informaciones a través de los sensores instalados en las calderas y transmiten estas informaciones mediante dispositivos Raspberry a los servidores remotos de Fundación CIDAUT. En estos servidores se procesan los datos para dotarles de un formato que permite guardarlos en base de datos. Más tarde, los clientes pueden conectarse a la plataforma web y visualizar los datos almacenados en graficas. Cada una de las gráficas contiene una serie de parámetros de configuración que los clientes pueden modificarlos para visualizar informaciones más concretas.

Figura 1: Ciderplus

La herramienta actual ayuda a los clientes y operadores a monitorizar y a encontrar una forma eficiente de controlar el sistema de climatización. No obstante, a medida que el volumen de datos crece, surgen nuevas formas de tratar los datos y extraer informaciones adicionales.

2.2. Usuarios

Basándonos en la situación actual del sistema podemos identificar 3 tipos usuarios:

- **Administradores**: Los administradores son los empleados de CIDAUT que tendrán acceso a la herramienta y con la posibilidad de configurar el entorno de la plataforma web o paneles de mandos.

- **Operadores**: Los operadores son técnicos de la propia instalación donde está situado el sistema de climatización. Poden visualizar informaciones a través de la plataforma web y en base a ellas pueden realizar modificación sobre el sistema de climatización.

- **Clientes**: Los clientes son propietarios o el encargado de la instalación que utilizarán la herramienta principalmente para consultar informaciones relativas al sistema.

2.3. Alcance y líneas de desarrollo

Partiendo de la idea de tratar los datos y extraer informaciones adicionales se han generado los dos objetivos del proyecto:

- **Graficar los indicadores:** se realizarán las mismas graficas del Ciderplus solo que esta vez con un volumen de datos mucho mayor. Por ejemplo, comparación de consumo entre diferentes equipos durante múltiplos años y bajo diferentes modos de funcionamiento.
- **Estimar consumos:** mediante algoritmos de aprendizaje automático se pretenderá crear modelos predictivos para estimar consumos aproximados de cada caldera. De esta forma poder planificar la logística de reabastecimiento de los combustibles de cada instalación.

Para poder tratar el gran volumen de datos registrados por la Fundación CIDAUT y de forma eficiente, primero, se necesitará ingerir el conjunto de datos almacenados en el base de datos relacional en datos no relacionales.

Una vez que los datos son adoptados como datos no relacionales se procesarán estos datos para extraer los indicadores objetivos. Las informaciones extraídas de los indicadores serán almacenadas y mostradas a través de unos paneles de mandos. Para los paneles de mandos se aprovecharán el diseño antiguo de las gráficas dándolas una distribución alterna. Siguiendo la necesitad de los clientes se distribuirán los indicadores en función de la información que ofrecen, principalmente se pueden clasificar los indicadores en tres grandes grupos: Indicadores de producción, Indicadores de mantenimiento predictivo e Indicadores de caracterización demanda.

Para conseguir el segundo objetivo del proyecto - Estimar consumos, se creará modelos predictivos. Al igual que el proceso anterior se utilizará los datos no relacionales. Primero se analizará el conjunto de los datos para determinar las propiedades de cada variable y posteriormente se tratará el conjunto de datos para dotarle el formato adecuado para el

entrenamiento del modelo. Para obtener un mejor resultado se realizará comparaciones entre varios modelos diferentes. Por último, al igual que los indicadores, se representarán los resultados de la estimación mediante graficas.

3. GESTIÓN DEL PROYECTO

3.1. Metodología del proyecto (Proceso unificado)

El proceso unificado es una metodología de desarrollo software enfocado a un desarrollo iterativo e incremental. El avance del proyecto está dirigido por los casos de uso, los cuales pueden modificarse a lo largo del proyecto.

El proceso unificado es ideal para un entorno de desarrollo complejo donde los objetivos y los requisitos del proyecto sufre constante cambios. Para proporciona a este tipo proyecto la flexibilidad necesaria que le permita asimilar los eventos imprevistos.

La idea subyacente al proceso unificado es la de evaluar constantemente las condiciones actuales del proyecto, y sobre ella realizar las planificaciones para el desarrollo futuro, en lugar de realizar sobre las condiciones previstas. De esta manera permite al equipo de desarrollo abordar los problemas de las metodologías tradicionales, adoptando las nuevas condiciones y asegurando de esta forma que la entrega final se ajuste a la necesidad del cliente.

El proceso unificado está compuesto por cuatro fases: Inicio, Elaboración, Construcción y Transición, que a su vez cada una de ellas esta divida en iteraciones de duraciones variadas. A final de cada iteración se necesita entregar una nueva solución del proyecto que consta de un incremento en la funcionalidad o una mejora de los ya existentes. [1] [2]

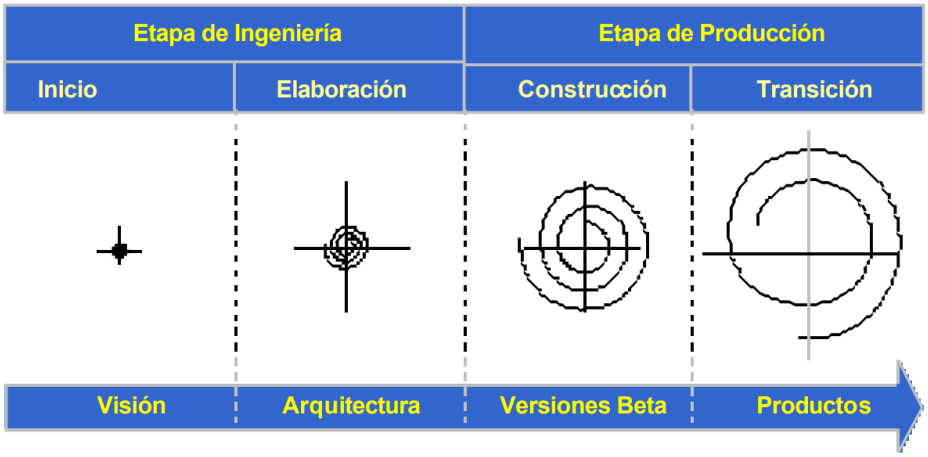

Figura 2: Ciclo de vida de Proceso Unificado - [1]

Cada una de las iteraciones es un ciclo de vida a menor escala que se asemeja a un ciclo de vida en cascada. Comienza con el análisis de requisitos y pasan por las etapas de diseño e implementación. Finalmente concluye con la prueba.

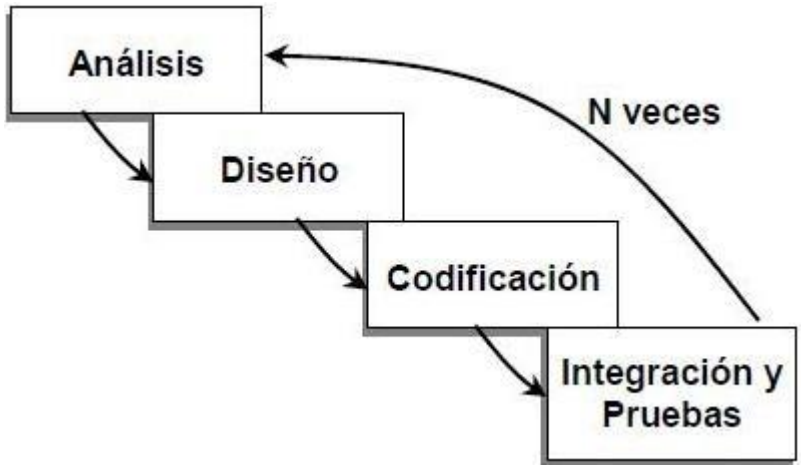

Figura 3: Estructura de Iteración - http://programaenlinea.net/proceso-unificado-rational-rup/

Un proyecto de desarrollo basado en el proceso unificado se rige por los casos de uso. La idea es que al inicio de cada iteración se selecciona un conjunto de casos de uso más relevantes y de ellos se extrae los objetivos de la iteración actual, de modo que incrementa de forma iterada las funcionalidades del proyecto.

Se puede ver el proceso unificado como una forma de diversifica el proyecto, partiendo de una visión global del sistema se detalla en subsistemas con diferentes funcionalidades que forma parte del sistema global.

Por último, y no por ellos menos importante, el proceso unificado se centra en identificar los riesgos potenciales del sistema. Avanza el desarrollo conforme con los riesgos registrados para disminuir la probabilidad de la ocurrencia y reduce el impacto que puede suponer el evento.

Una vez que se ha detallado las características del proceso unificado a continuación analizamos las diferentes fases del desarrollo:

- **Inicio:** durante esta fase se lleva a cabo la descripción resumida del producto final y del entorno, de las cuales se identifica las funcionalidades primarias del sistema y realizar una evaluación inicial de los riesgos del sistema.

- **Elaboración:** en esta fase se intenta identificar la mayoría de los requisitos u objetivos del sistema y, partiendo de ellos, se esboza una solución básica que se refinará a lo largo del proyecto. En cuanto los riesgos detectados deben ser tratados para reducir sus impactos sobre el proyecto.

- **Construcción:** esta es la fase principal de proyecto donde se realizan la implementación de los objetivos establecidos en la fase anterior. En cada iteración de esta fase se incorporan nuevas funcionalidades del sistema. Se da por finalizado esta fase cuando todos los objetivos del proyecto hayan sido completados.

- **Transición:** en esta última fase de proyecto se realizar prueba de producto final y la corrección de los fallos en el producto.

3.2. Planificación del proyecto

De acuerdo con los principios definidos por la metodología de proceso unificado, se concreta la planificación del proyecto de la siguiente forma [3]:

- **Inicio:** durante esta primera fase con una sola iteración se intentará comprender el entorno de proyecto y los factores que intervienen, estableciendo el alcance y los objetivos principales del proyecto. También se intentará identificar los riesgos potenciales del proyecto.

- **Elaboración, iteración 1:** una vez que haya adquirido conocimientos básicos sobre el proyecto, en esta iteración se intentará elaborar un plan inicial con las funcionalidades principales del proyecto y trazar una estructura del sistema preliminar.

- **Elaboración, iteración 2:** basándose en las informaciones aportadas por la iteración anterior se centrará ahora en extraer la mayor cantidad de los requisitos posibles

(identificar los parámetros relevantes de los paneles de mandos y establecer el objetivo de los modelos predictivos).

- **Construcción, iteración 1:** siguiendo la planificación programada en esta iteración se llevará a cabo la instalación del sistema y la ingestión de los datos.

- **Construcción, iteración 2:** con los datos cargado en los servidores se proseguirá el proyecto con la extracción de los parámetros definidos.

- **Construcción, iteración 3:** el objetivo de esta iteración es diseñar e implementar los paneles de mandos para visualizar los parámetros extraídos de la iteración previa.

- **Construcción, iteración 4:** en esta iteración se centrará en la creación de los modelos predictivos.

- **Transición:** en esta última fase e iteración del proyecto se verificará los resultados obtenidos del proyecto, desde los paneles de mando, hasta los modelos predictivos. En el caso de detectar algún tipo de fallo se deberá actualizar el proyecto y repetir este proceso hasta obtener un resultado consistente.

3.3. Temporización del proyecto

En este proyecto cada día laboral equivale a 4 horas laborables.

Tarea	Duración	Fecha Inicio	Fecha Fin
Proyecto Completo	75 días	15/03/2018	29/06/2018
Fase de Inicio	11 días	15/03/2018	03/04/2018
Comparación entre las diferentes tecnologías y determinar la plataforma a utilizar.	4 días	15/03/2018	20/03/2018
Aprendizaje sobre la plataforma	7 días	21/03/2018	03/04/2018
Fase de elaboración	5 días	04/04/2018	10/04/2018
Comprensión de entorno del proyecto – iteración 1	3 días	04/04/2018	06/04/2018
Planificación de proyecto – iteración 2	2 días	09/04/2018	10/04/2018
Fase de construcción	53 días	11/04/2018	27/06/2018
Configuración de sistema – Iteración 1	2 días	11/04/2018	12/04/2018
Extracción y transferencia de datos – iteración 1	5 días	13/04/2018	19/04/2018

Tratamiento de datos en brutos – Iteración 2	5 días	20/04/2018	26/04/2018
Almacenamiento y extracción de datos útiles – Iteración 2	3días	27/04/2018	02/05/2018
Elaboración de Dashboard – Iteración 3	5 días	03/05/2018	09/05/2018
Obtención de los datos – Iteración 4	2 días	10/05/2018	11/05/2018
Exploración y visualización de datos – Iteración 4	5 días	14/05/2018	21/05/2018
Tratamiento y transformación de datos – Iteración 4	5 días	22/05/2018	29/05/2018
Generación de modelos – Iteración 4	15 días	30/05/2018	19/06/2018
Prueba y ajuste de modelos – Iteración 4	6 días	20/06/2018	27/06/2018
Fase de transición	6 días	28/06/2018	05/07/2018
Prueba de paneles de mando y modelos predictivos	6 días	28/06/2018	05/07/2018

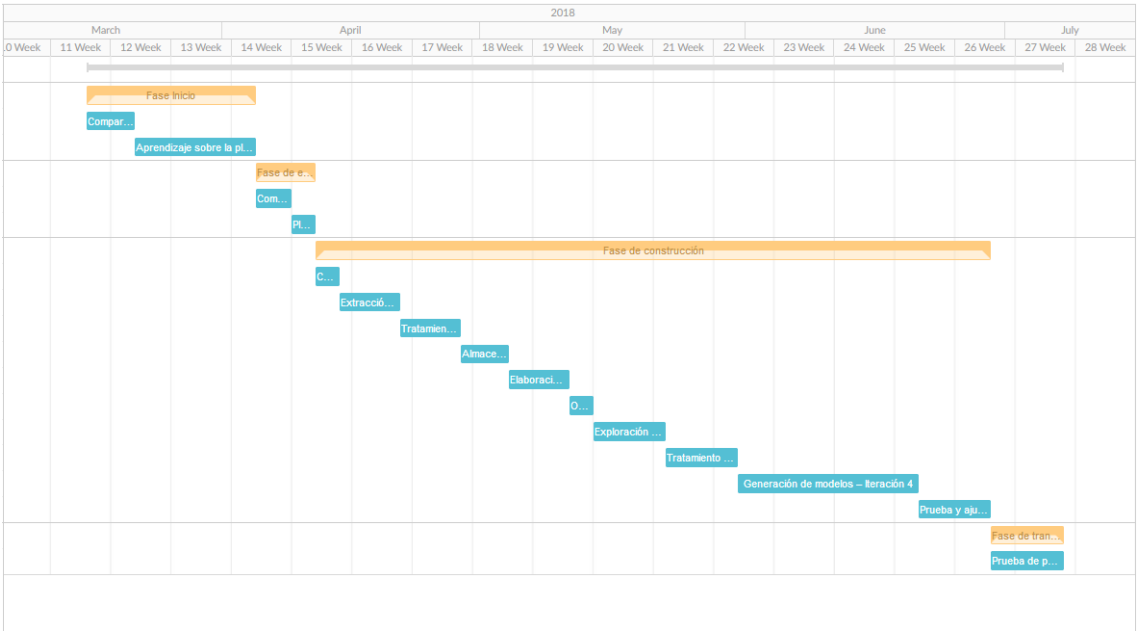

Figura 4: Diagrama Gantt

3.4. Gestión de riesgos

La gestión de riesgos se encarga de identificar, mitigar y monitorizar las ocurrencias imprevistas del proyecto que pueden tener impactos positivos o negativos. La idea principal de la gestión de riesgos es identificar y trazar un plan de contingencia para los riesgos en el inicio de proyecto, de modo que puede evitar o reducir el efecto del evento. Durante el proyecto, cuando un riesgo surge. se activará inmediatamente el plan de contingencia que permitirá tratarlo de forma sistemática y, a su vez, monitorizará toda la evolución del riesgo con el objetivo de mejora los proyectos futuros.

Una de las acciones que se llevar a cabo durante la gestión de riesgos es el proceso de cuantificación, que consiste en clasificar y ordenar los riesgos en función del impacto que pueden tener sobre el proyecto. Para determinar de forma cuantitativo el impacto de un riesgo sobre el proyecto es necesario identificar su probabilidad y consecuencia. Sin embargo, teniendo las dos medidas independientes sigue siendo difícil priorizarlos, por lo tanto, en su lugar se recurre a la métrica combinadas de ambas, la exposición al riesgo:

Exposición al riesgo = Probabilidad x Consecuencia

Modificación de requisitos		
Número: **01**	Fecha: 04/2018	Fase: Inicio
Probabilidad: Muy baja	Imparto: Medio	Exposición: 2
Descripción del riesgo: Debido las malas interpretaciones o los aspectos cambiantes del entorno. Los requisitos iniciales pueden sufrir modificación a lo largo del proyecto.		
Consecuencia		
Las modificaciones de los requisitos suelen implica mayor coste y tiempo para completar el proyecto.		
Planificación de riesgo		
Estrategias: • Reducción de riesgo • Reserva del riesgo	**Plan de acción:** Dedica mayor recurso en la fase inicial para minimizar la probabilidad de suceso. Reserva amplio margen de holguras para no alterar la planificación inicial en el caso de que suceda.	

Falta de datos		
Número: **02**	Fecha: 04/2018	Fase: Construcción
Probabilidad: Media	Imparto: Muy Alto	Exposición: 6
Descripción del riesgo: Durante el desarrollo de los modelos predictivos es posible que no se encuentran suficientes datos con los que se puede tratar.		
Consecuencia		
La carencia de datos para el entrenamiento de los modelos predictivos significaría cancelación del proyecto.		
Planificación de riesgo		
Estrategias: • Evitar el riesgo	**Plan de acción:** Comprender las características del proyecto y localizar suficientes datos antes de empezar con el desarrollo de modelos predictivos.	

Indisposición del personal		
Número: **03**	Fecha: 04/2018	Fase: Proyecto
Probabilidad: Muy baja	Imparto: Muy Alto	Exposición: 4
Descripción del riesgo: A lo largo de proyecto los personales implicados en el proyecto pueden enfermar o tener un accidente. Por lo que no estarán disponible para completar las tareas asignadas.		
Consecuencia		
Aplazamiento del labor asignado al personal durante su periodo de recuperación.		
Planificación de riesgo		
Estrategias: • Reserva del riesgo	**Plan de acción:** Redistribuye trabajos asignados entre el tiempo restante del proyecto o entre los personales disponibles.	

Fallos en los equipos de desarrollo		
Número: **04**	Fecha: 04/2018	Fase: Proyecto
Probabilidad: Muy baja	Imparto: Bajo	Exposición: 1
Descripción del riesgo: Durante el proyecto puede estropear equipos de desarrollo.		
Consecuencia		
La indisponibilidad de recursos de desarrollo conlleva el retraso en el proyecto.		
Planificación de riesgo		
Estrategias: • Protección del riesgo • Reducción del riesgo	**Plan de acción:** Realizar revisiones periódicas con frecuencia para comprobar el estado de los equipos y reservar equipos adicionales para el caso de que ocurra el evento	

Estimación incorrecta del proyecto		
Número: **05**	Fecha: 04/2018	Fase: Proyecto
Probabilidad: Alta	Imparto: Medio	Exposición: 5
Descripción del riesgo: La estimación del coste realizado en el inicio del proyecto puede diferir del avance real del proyecto.		
Consecuencia		
La mala estimación del coste de proyecto significa modificación en el coste y tiempo de plan inicial.		
Planificación de riesgo		
Estrategias: • Protección del riesgo	**Plan de acción:** Mediante constante monitorización del proyecto para detectar posibles fallos en la planificación lo antes posible y redistribuir de modo que reduzca el impacto del riesgo.	

	Likelihood of incident scenario	Very Low (Very Unlikely)	Low (Unlikely)	Medium (Possible)	High (Likely)	Very High (Frequent)
Business Impact	Very Low	0	1	2	3	4
	Low	1	2	3	4	5
	Medium	2	3	4	5	6
	High	3	4	5	6	7
	Very High	4	5	6	7	8

We have based the estimation of risk levels on ISO/IEC 27005:2008 (10).

Figura 5: Matriz de evaluación de riesgo - [3]

4. PLATAFORMAS DE DESARROLLO

Para poder realizar estudios con las tecnologías de Big Data requiere un sistema con suficientes capacidades de cómputo y sobre todo teniendo en cuenta la velocidad de crecimiento de los datos, el sistema deberá ser altamente escalable. La creación y mantenimiento de un sistema de computación propia requiere mucho esfuerzo y es inadecuado, por lo tanto, para este proyecto se ha optado por contratar los proveedores externos.

En el mercado actualmente se puede encontrar muchos proveedores que ofrecen sistemas enfocados a la tecnología de Big Data. Entre ellos se debe descartar Amazon EMR y Azure HDinsight. Se optado por Amazon EMR en este proyecto debido a la cantidad de documentaciones disponibles que supondrían una gran ayuda para el desarrollo del proyecto.

4.1. Amazon Elastic MapReduce (EMR)

Amazon EMR es un servicio web que proporciona un marco de trabajo destinada al Big Data que es previamente configurada con el objetivo de optimizar y simplificar los procesos de tratamiento de grandes volúmenes de datos.

Los clústeres de EMR están compuestos por una serie de instancias de Amazon Elastic Compute Cloud (EC2). Una de las ventajas de este sistema es su escalabilidad, que permite ampliar el número de las instancias en tiempo de ejecución, incluso se puede configurar el sistema para que el proceso de escalado sea automático. Otra de las ventajas es la posibilidad de almacenar los datos de forma persistente en los servicios de almacenamiento de Amazon para evitar el mantenimiento de los clústeres de computación innecesaria.

Amazon EMR permite ejecutar marcos de trabajo distribuidos más populares como Hadoop, Apache Spark, Presto … Estos marcos de trabajos son dedicados para realizar análisis financieros, la simulación científica, la bioinformática y el aprendizaje automático. [4]

4.2. Apache Spark

Apache Spark es un proyecto de código libre destinado para el procesamiento de datos. Apache Spark a diferencia de su predecesor, Hadoop que carga los datos desde el disco para realizar los análisis, en su lugar almacena los datos en la memoria de los equipos y los carga directamente de allí. De esta forma reduce drásticamente el tiempo de acceso a los datos y resulta ideal para los proyectos que utilizan algoritmos de aprendizaje automático. Este tipo de algoritmos utilizan de forma iterada los mismos datos, por lo tanto, al disponer de los datos en memoria o en cache, la velocidad de acceso se ve significativamente incrementada.

Apache Spark está formado por el núcleo de Spark y una serie de bibliotecas. El núcleo de Spark funciona como el motor de ejecución distribuida, mientras los APIs de Java, Scala y Python como la plataforma para realizar el desarrollo de las aplicaciones distribuida. [5] [6] [7] [8]

Bibliotecas de Spark:

- **MLlib:** MLlib es una biblioteca de Spark enfocada para facilitar y optimizar el proceso de aprendizaje automático que ofrece las siguientes herramientas:
    - Algoritmos de aprendizaje automático: incluye algoritmos para realizar clasificación, regresión y agrupamiento.
    - Caracterización: conjunto de herramientas para tratar y preparar los datos.
    - Pipelines: herramienta de construcción y evaluación de los modelos.
    - Persistencia: permite cargar y almacenar los modelos generados de forma persistente.
    - Utilidades: conjunto de herramientas para realizar en general operaciones estadísticas.

- **Spark Streaming:** Spark Streaming es una biblioteca que permite la ingestión de datos desde fuentes externas de forma dinámica mediante herramientas como Kafka, Flume, Kinesis …

  Spark Streaming ofrece una abstracción en alto nivel llamado DStream que representa el flujo continuo de datos. Ese flujo de datos, internamente se representa como un conjunto de datos con formato RDD, lo cual puede ser tratado como si fuera un conjunto normal de datos una vez que es cargado en el entorno de Spark.

- **GraphX:** GraphX es una extensión que proporciona la capacidad de analizar grafos a Spark. Esta biblioteca incluye operadores en alto nivel optimizados para realizar cálculos de grafos complejos.

5. PROCESO ETL Y CREACIÓN DE DATA LAKE

5.1. Extracción de datos

Como todos los proyectos de Big Data, antes de poder tratar y analizar los datos se necesita recopilarlos. Para el desarrollo de este Trabajo Fin de Máster se recogerá los datos de dos fuentes diferentes. Por una parte, los datos proporcionados por la Fundación CIDAUT, los cuales describen el estado y las características de cada instalación de caldera. Por otra parte, los datos proporcionados por la Junta de Castilla y León, que contienen informaciones meteorológicas de casi todas las poblaciones de castilla y león.

Los datos proporcionados por la Fundación CIDAUT son recopilados mediante unos sensores instalados en las calderas y son enviados a través de unos Raspberry's al servidor privado de CIDAUT, donde son tratados para ser almacenados en unos ficheros con formato CSV.

5.2. Cargar datos

Con los datos que serán generados en el anterior proceso, se procederá la operación de cargar datos en el servidor remoto de Amazon. Para ellos, se subirá los datos al sistema de almacenamiento S3 de Amazon a través de una interfaz web. Las ventajas de disponer los datos en el S3 son [9]:

- Aislamientos de datos persistentes con la capacidad de computo. Esto quiere decir que se podrá eliminar los clústeres de computación sin afectar los datos y de esta manera ahorrar el coste de mantenimiento de los clústeres de computación.

- Todos los datos almacenados en el sistema son cifrados y solo se puede acceder a estos datos mediante el ACL (Listas de Control de Acceso).

- El sistema de almacenamiento tipo S3 cuenta con una durabilidad de **99,999999999%** y con copias distribuidas en varias zonas.

- Baja latencia en la lectura y escritura de datos en el sistema.

- Altamente escalable, capacidad de almacenamiento prácticamente infinita y pago solo por la capacidad utilizada.

5.3. Procesar datos

Una vez que los datos están disponibles en los servidores de S3 son cargados a los clústeres de computación para ser tratados y analizados. Concretamente se extrae de los datos originales los siguientes parámetros para que sean monitorizados en la siguiente etapa de visualización:

- **Tiempo promedio de arranque y apagado.**
- **Tiempo dedicado a cada modo de funcionamiento.**
- **Consumo diario de cada caldera.**
- **Número de arranques y apagados.**
- **Rendimiento promedio.**
- **Rendimiento promedio en modo bajo demanda.**
- **Diferencia máxima diaria entre temperatura de salida y temperatura de inercia superior en modo bajo demanda.**
- **Diferencia máxima diaria entre temperatura de retorno y temperatura de inercia inferior en modo bajo demanda.**

5.4. Parámetros de los datos recogidos

En este punto se detallará los datos implicados para generar los modelos predictivos y los paneles de mandos. A continuación, se describirá todos los parámetros implicados:

Parámetros ambientales	
Temperatura	Temperatura local en Grados Celsius.
Radiación	Indicador de radiación local percibida cuya unidad de medida es vatios por metro cuadrado ($\frac{w}{m^2}$)
Precipitación	Precipitación local medido en milímetros (o litros por metro cuadrado).
Humedad	Humedad relativa (%)
Vel.Viento	Velocidad de viento en metros/segundo.
Dir. Viento	Dirección de viento medido en grados, siendo grado 0 el norte.
Día festivo	Puede tomar valores: falso y verdadero.
Día de la semana	Puede tomar valores de 1 a 7 siendo el 1 lunes y el 7 el domingo.
Día del mes	Puede tomar valores de 1 a 31 indicando el día del mes que se encuentra.
Mes	Puede tomar valores de 1 a 12 siendo el 1 enero y 12 diciembre.
Año	Valor relativo respecto al año de instalación, es decir el primer año tiene como valor 0 y el segundo 1 ...

Parámetro objetivo:

Consumo (D_SEG_SINFIN)	El parámetro que se pretender predecir con el modelo generado.

6. DISEÑO DE PANELES DE MANDOS

Existe una amplia gama de plataformas que proporcionar soportes para generar graficas con el fin de que los usuarios puedan analizar los datos de forma visual, tales como Qlik Sense, FusionCharts, Highcharts, Carriot, Tableau ... [15]

6.1. Elección de la plataforma de visualización

Entre todas las plataformas de visualizaciones disponibles en el mercado se ha optado por Tableau en este proyecto basándose en los siguientes criterios [16]:

- La interfaz para crear gráficas y paneles de Tableau en comparación con otras herramientas, resulta especialmente simple e intuitiva para que los usuarios puedan diseñar paneles prácticamente desde el primer momento.

- Tableau permite disponer de los paneles y graficas de forma remota para mantener los datos de los clientes siempre sincronizados.

- Los elementos analíticos creados por Tableau son optimizados para que sean capaces de procesar y mostrar de forma instantánea a los usuarios.

- Para poder proporcionar mejor servicio a aquellos clientes que necesitan realizar análisis más avanzadas, Tableau ofrece soportes de desarrollo para flujos de trabajo complejos que aumenta significativamente la reutilización de los datos procesados en los diferentes elementos visuales.

- A partir de la versión 10, Tableau incorpora nuevas características que permite una comunicación directa con los servidores donde residen los datos originales y donde realizan el procesamiento de datos. Por lo ejemplo con los servidores de Amazon.

6.2. Diseño de paneles

Respecto al diseño de los paneles para que resulten lo más intuitivo posible a los usuarios, se ha decidido dividir los parámetros en tres paneles diferentes basándose en el tipo de información que aporta cada uno.

- **Indicadores de producción:**

  Los parámetros incluidos en este panel permiten vigilar el buen funcionamiento de los equipos. Este panel tiene como objetivo la de comparar entre diferentes equipos para detectar eventos que mejoran la producción de las instalaciones.

- **Indicadores de mantenimiento predictivo:**

  Permiten detectar posibles fallos en la máquina en base a los valores atípicos y a la evolución anormal de los indicadores. Son empleados para ver evolución de cada equipo, no para comparar entre equipos.

- **Indicadores de caracterización demanda:**

  Permiten detectar situaciones anómalas, caracterizar el tipo de demanda de la instalación e identificar el mejor momento para realizar un cambio de estrategia operacional para que ser más eficiente la instalación.

Una vez que están definido los propósitos de cada panel de mando, a continuación, se detallará las decisiones visuales que se van a tomado para cada uno de ellos.

6.2.1. Indicadores de producción:

Como ya se ha explicado en los párrafos anteriores los indicadores de producción sirve para realizar comparación entre diferentes equipos entonces implica que en las gráficas y las informaciones numéricas del panel debería permitir mostrar simultáneamente uno o más equipos. Basándose en los requisitos especificados del usuario, en total hay 4 gráficas y 4 informaciones numéricas que se necesitan mostrar en el panel.

Tablas:

- Rendimiento
- Rendimiento bajo demanda
- Tiempo bajo demanda

- Potencia media/Potencia máxima

Información numérica:

- Tiempo bajo demanda
- Potencia
- Rendimiento bajo demanda
- Rendimiento

Las gráficas e informaciones exceptos las obtenidas bajo modo demanda. El resto de las gráficas e informaciones ya están diseñados en el antiguo sistema por lo tanto se aprovechará sus diseños con los que los usuarios ya están familiarizadas. Y para seguir la misma línea de diseño las otras graficas e informaciones se utilizarán los mismos diseños. Aparte de los elementos visuales e informativos también habrá dos elementos configurables que los usuarios pueden configurar que son:

- **Intervalo de tiempo:** A través de esta variable los usuarios pueden seleccionar el periodo de las informaciones que quieren consultar.
- **Lista de instalaciones:** A través de esta variable los usuarios pueden seleccionar las instalaciones entre las que quieren realizar la comparación.

Teniendo en cuenta que todas graficas tienen la misma importancia en este caso se colocarán los 4 graficas de forma equitativo ocupando cada una de ellas el mismo espacio formando una matriz de 2 x 2 y se colocará las informaciones numéricas en el lado derecho del panel y debajo de las informaciones las variables configurables.

En cuanto a la selección de los colores se ha utilizado el color blanco para el fondo que nos permite tener una mayor claridad en el panel de mando y relajar los ojos. [15] Se utilizará la paleta de colores por defecto de Tableau para representar cada una de los equipos seleccionado, de esta forma nos garantizamos un mayor contraste posible entre los equipos que nos permite diferenciarlos fácilmente.

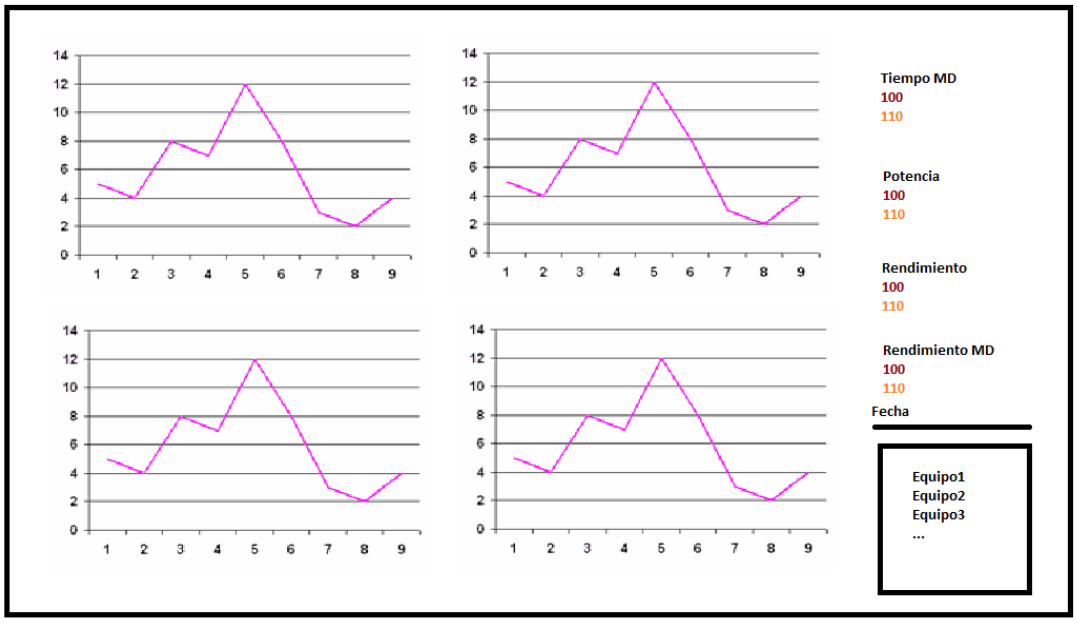

Figura 6: Boceto 1 - Indicadores de producción e Indicadores de caracterización demanda

6.2.2. Indicadores de mantenimiento predictivo

Existen tres gráficas y tres informaciones numéricas en el panel que son las siguientes:

Tablas:

- Tiempo arranque
- Tiempo apagado
- Rendimiento

Informaciones numéricas:

- Tiempo en modo arranque (En minutos)
- Tiempo en modo apagado (En minutos)
- Rendimiento

Está vez el panel contiene 3 graficas por lo tanto no se puede distribuir de forma equitativa las gráficas, además entre las 3 graficas la gráfica de “Rendimiento” tiene una importante mucho mayor que las otras dos graficas. Por lo tanto, en este panel la gráfica de “Rendimiento” ocupará un espacio mucho mayor que las otras dos con la intensión de remarcar la importancia de la gráfica “Rendimiento” respecto a las otras.

A igual que el panel anterior los usuarios dispondrán de dos parámetros (Intervalo de tiempo, Equipo) de configuración, aunque esta vez con una ligera modificación en el segundo parámetro que solo se permite seleccionar un equipo a la vez ya que este panel no es para realizar comparaciones entre los equipos sino para detectar defectos en el propio sistema.

En cuanto a la selección de colores esta vez se ha selecciona de nuevo el color blanco para el fondo y el color azul como el color principal ya que es uno de los colores que tiene mayor contraste con el color blanco.

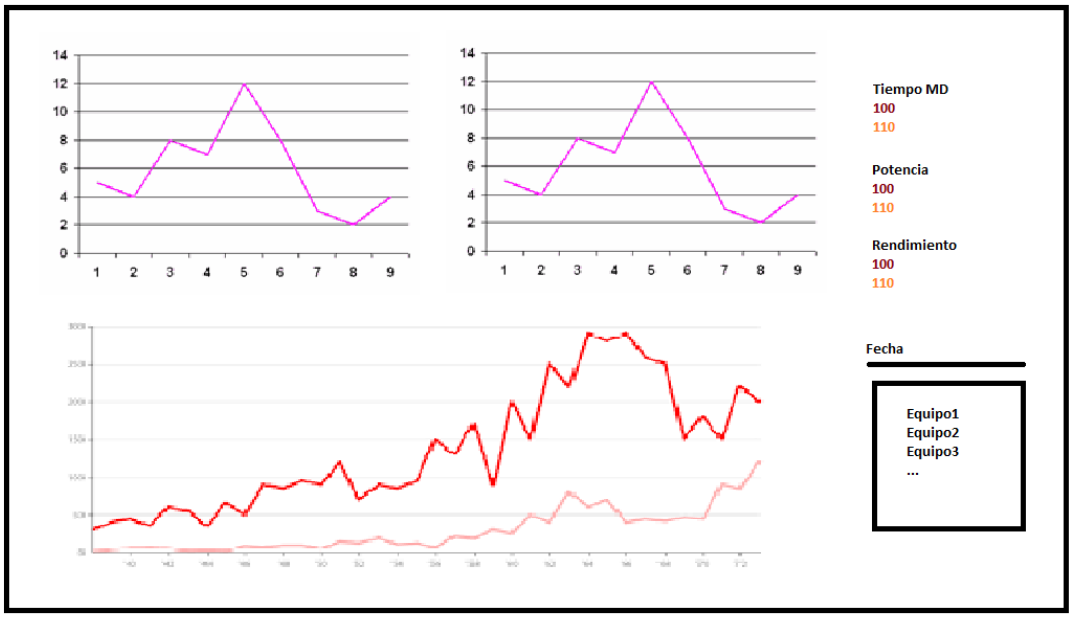

Figura 7: Boceto 2 - Indicadores de mantenimiento predictivo

6.2.3. Indicadores de caracterización demanda

En este panel volvemos a tener 4 gráficas y 4 informaciones numéricas y todas ellas con una importancia muy similares entre ellos. Por lo tanto, se utilizar la misma lógica de distribución utilizada en el primer panel para distribuir los elementos visuales de este panel.

Tablas:

- Nº de arranques
- Nº de paradas
- Diferencia máxima entre Tª salida y Tª inercia superior
- Diferencia máxima entre Tª retorno y Tª inercia inferior

Informaciones numéricas:

- Promedio de diferencia máxima entre Tª salida y Tª inercia superior
- Promedio de diferencia máxima entre Tª retorno y Tª inercia inferior
- Nº total de arranques
- Nº total de paradas

Para la paleta de colores se utilizará el color blanco para fondo y el color azul como el color principal.

** Este panel de mando tiene un diseño muy similar al panel de mando de los indicadores de producción. Figura 6*

7. IMPLEMENTACIÓN

En este capítulo se detallará el proceso de implementación del proyecto que se ha planificado en el capítulo anterior. Se dividirá el proceso en dos fases. Una primera fase en la que se llevará a cabo el proceso de extracción, procesado y visualización de datos. Y una segunda fase que se centrará en la implementación de modelos de predicciones. [16] [17]

7.1. Extraer, transformar y cargar de datos

En esta fase se recolectará y tratará los datos para mostrarlos posteriormente en los cuadros de mandos. Durante el proceso adaptarán los datos para que sean adecuados para generar los modelos predictivos.

7.1.1. Extracción de datos

Tal como se ha explicado en el capítulo de diseños, el proyecto presente tiene dos fuentes de datos. Una de ellas es proporcionada por la Fundación CIDAUT y la otra es proporcionada por la Junta de Castilla y León. En esta fase solo se analizará la implementación de sistema de recogida de datos por parte de la Fundación CIDAUT, ya que de la segunda no se conoce en detalles los procesos aplicados para la obtención de los datos.

Figura 8: estructura ciderplus

Los datos proporcionados por la Función CIDAUT son de 3 tipos diferentes:

1. Datos de las variables recogidas por el equipo Ciderplus (Sistema informático instalado en las instalaciones para controlar y monitorizar las calderas.)
2. Datos de los cambios de estado que sufre el equipo Ciderplus
3. Datos de las alertas que se producen en el equipo Ciderplus.

El equipo Ciderplus recoge el valor de las variables, de los cambios de estado y de las alertas cada minuto, escribiéndolas en dos archivos de texto que se envía a un servidor en la nube y el envío se realiza cada 15 minutos.

El primer archivo es el de las variables y tiene el siguiente aspecto:

Hora	Variable	Valor
0:1:0	TT105	40.4
0:1:0	PT203	-0.50
0:1:0	TT205	1372.0

El segundo archivo está destinado a recoger las alertas y los cambios de estado y tiene el siguiente aspecto:

Hora	Alarma	Estado
0:6:19	Modo MA	OFF
0:6:19	Modo MC	ON
1:59:0	Alr_G_19	OFF

Cada archivo es procesado en el servidor para dar como resultado tres nuevos archivos. El primero registran las variables por cada minuto y tiene el siguiente aspecto:

D_FILE_CALDERA	F_FECHA	D_TEMP_LINEA	D_PRESION	D_TEMP_GAS_ENTRADA
EQUIPO1	01/01/2016 0:01:00	32.7	-54.15	182.9
EQUIPO1	01/01/2016 0:02:00	32.8	-59.69	182.0

D_TEMP_GAS_ESCAPE	D_OXI_GAS_ESCAPE	D_TEMP_AGUA_CASCADA	D_TEMP_AGUA_HOGAR	D_TEMP_AGUA_CENICERO
71.3	1725		64.7	58.8
69.8	1792		61.2	59.1

D_TEMP_AGUA_SALIDA	D_TEMP_AGUA_RETORNO	D_TEMP_INERCIA_SUP	D_TEMP_INERCIA_INF	D_DEMANDA
67.5	58.8	67.1	61.4	TRUE
63.2	59.1	67.5	61.6	TRUE

D_SEG_SINFIN	D_POTENCIA	ID_ESTADO	D_TEMP_INIERCIA1	D_TEMP_INIERCIA2	D_CONTADOR_ENERGIA	D_TEMP_EXT
0.0	189.1	46			0.0	850.0
20.6	111.0	46			0.0	850.0

El segundo recoge todos los cambios de estado:

D_FILE_CALDERA	D_ACRONIMO	D_NOMBRE_COMPLETO	F_INICIO	F_FIN
EQUIPO2	Modo MF	Apagado	09/09/2013 18:16	09/09/2013 19:30

EQUIPO2	Modo SB	Stand-By	09/09/2013 19:30	10/09/2013 8:55
EQUIPO2	Modo SB	Stand-By	10/09/2013 8:55	10/09/2013 8:56

El tercero recoge todas las alertas:

D_FILE_CALDERA	D_ACRONIMO	D_NOMBRE_COMPLETO	TIPO	F_FECHA	D_VALOR
EQUIPO2	bAlr_MA_01	Tiempo Encendido máx.	nivel 1	06/09/2013 10:02	ON
EQUIPO2	bAlr_MD_09	Temp. Gas errónea	nivel 1	09/09/2013 14:06	ON
EQUIPO2	bAlr_MD_09	Temp. Gas errónea	nivel 1	09/09/2013 14:09	OFF

7.1.2. Cargar Datos

Con el fin de poder analizar los datos recogidos en el anterior paso, se carga los datos en el sistema de almacenamiento persistente de Amazon (Amazon S3). Una vez que los datos son cargados en el S3, se procede con el procesamiento de datos.

Llegados a este punto se necesita transformar los datos de los sensores de calderas almacenados en el S3 para que sean legibles para la plataforma de visualización y aptos para la creación de los modelos de predicciones.

Como el primer paso de la transformación se carga todos los datos en el clúster de Spark y se elimina las columnas de índices que se han creado de forma automático y se agrupan los ficheros pequeños en ficheros más grandes, reduciendo así la cantidad de accesos necesarios a diferentes ficheros. Cuando es eliminada la columna de índices y son agrupados en ficheros grandes, estos datos son almacenados de nuevo en el S3.

7.1.3. Procesar Datos

En cuanto al extracción de los datos para el panel de mandos se divide el proceso en dos subtareas. En la primera subtarea se incluye los cálculos de los parámetros que se pueden extraer directamente a través de los datos originales. Y en la segunda subtarea se incluye un paso intermedio para filtrar los datos originales y quedar únicamente con datos pertenecientes al modo bajo demanda.

Para llevar a cabo la filtración de datos de modo bajo demanda, se cargan dos conjuntos de datos en el clúster de Spark: el conjunto de datos de sensores de calderas que cuenta con más de 10 millones de entradas, y el conjunto de datos de sensores de cambios de estados que cuenta con más de 100000 entradas. El conjunto de datos de cambios de estados contiene los intervalos en el que las calderas están en el modo bajo demanda que sirve para decidir si los datos del conjunto de sensores de calderas pertenecen o no al intervalo de modo bajo demanda. Por último, los datos filtrados son almacenados nuevamente el S3 para poder continuar con la segunda subtarea.

7.1.3.1.Tiempo promedio de arranque y apagado

Con el objetivo de calcular el tiempo promedio de arranque y apagado se carga el conjunto de datos de cambios de estados donde contiene los intervalos de tiempo dedicado al arranque y apagado de cada instalación. En base a los datos anteriores se calcula la duración de cada intervalo y se almacena en una columna adicional. Para terminar, se agrupa y se calcula el promedio de la nueva columna por día y por instalación.

7.1.3.2. Tiempo dedicado a cada modo de funcionamiento

A igual que el atributo anterior, para el cálculo de este atributo se carga el conjunto de datos de cambios de estados. Pero esta vez se filtra para quedar con los cinco modos de funcionamiento: apagado, arranque, bajo demanda, calentamiento y stand-by, y se calcula la duración de cada intervalo. En último lugar se agrupan los datos por día y por instalación.

7.1.3.3. Consumo diario de cada caldera

El cálculo de consumo diario requiere en el primer lugar cargar el conjunto de datos de calderas. A continuación, se calcula el consumo de cada entrada (registrado por minuto) mediante la siguiente fórmula:

Consumo = D_SEG_SINFÍN * Constante De Caldera

Una vez calculado el consumo de cada instancia se agrupan todos ellos por día y se guardan en una columna adicional.

7.1.3.4. Número de arranques y apagados

Para calcular el número de arranques y apagados se carga el conjunto de datos de cambios de estados y se filtran los datos cargados en el sistema por los modos de arranque y apagado. Una vez que los datos hayan sido cargados y filtrados, se procede a agrupar los datos por día y por instalación. Por último, se cuenta la frecuencia de aparición de cada modo y se los guardan en una nueva columna.

7.1.3.5. Rendimiento promedio

Respecto el cálculo del parámetro rendimiento promedio, en primer lugar, se necesita cargar los datos de calderas y luego se aplican las siguientes fórmulas para obtener el rendimiento de las calderas:

$$Rendimiento = \frac{Energía\ Generada}{Pellet\ consumido}$$

$$Energía\ Generada = \frac{D\_Potencia}{60}$$

$$Consumo = D\_SEG\_SINFIN\ * Constante\ de\ Caldera$$

Por último, para completar el proceso se agrupan los datos de la columna adicional por día y se calculan el rendimiento promedio.

7.1.3.6.Rendimiento en modo bajo demanda

El cálculo de este parámetro utiliza el mismo proceso que se ha realizado en el cálculo de rendimiento para los datos totales, solo que esta vez se emplea el conjunto de datos filtrados en el modo bajo demanda en lugar del conjunto de datos totales.

7.1.3.7. Diferencia máxima diaria entre temperatura de salida y temperatura de inercia en modo bajo demanda

A fin de obtener la diferencia máxima diaria entre la temperatura de salida y la temperatura de inercia, se calcula en el primer lugar la diferencia entre la temperatura de salida y temperatura de inercia para cada entrada y finalmente se agrupan todas las entradas por día e instalación para hallar el valor máximo de la diferencia.

7.1.3.8. Diferencia máxima diaria entre temperatura de retorno y temperatura de inercia en modo bajo demanda

Se aplica el mismo proceso que se ha utilizado para el cálculo de atributo anterior, a diferencia de que esta vez se utiliza la temperatura de retorno en lugar de temperatura de salida.

7.1.4. Visualizar los datos

Una vez el conjunto de datos de los parámetros objetivo es extraído, se lo ingiere en la plataforma de visualización Tableau. Siguiendo lo planificado en el capítulo se disponen de los parámetros en 3 paneles de mandos reutilizando el diseño la aplicación antigua con el que los usuarios están familiarizados, solo que esta vez con un conjunto de datos mucho mayor.

- **Indicadores de producción:**

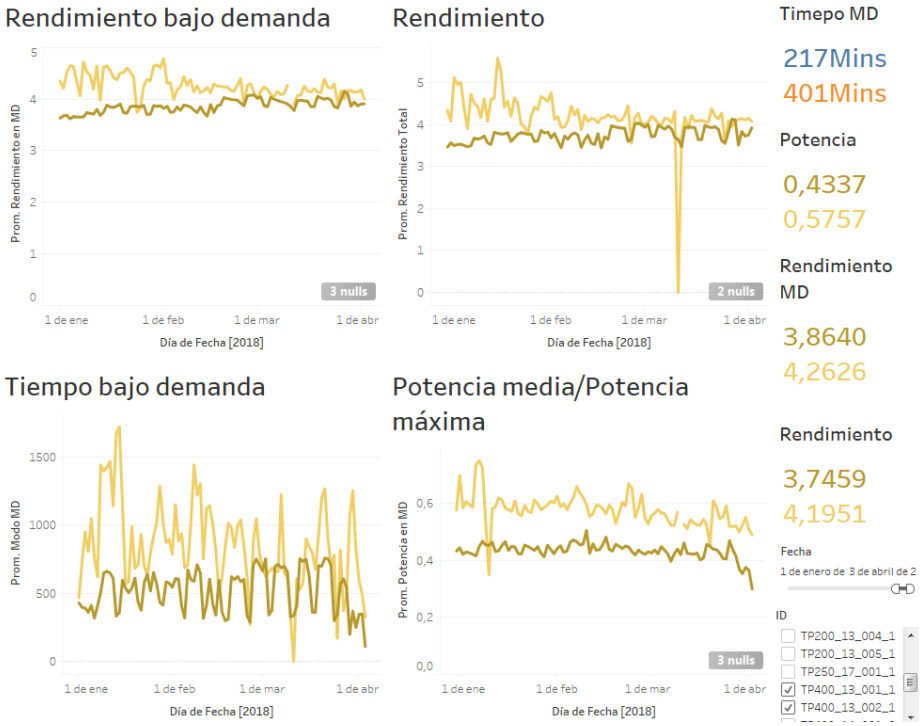

Figura 9: Panel de mando - producción

- **Indicadores de mantenimiento predictivo:**

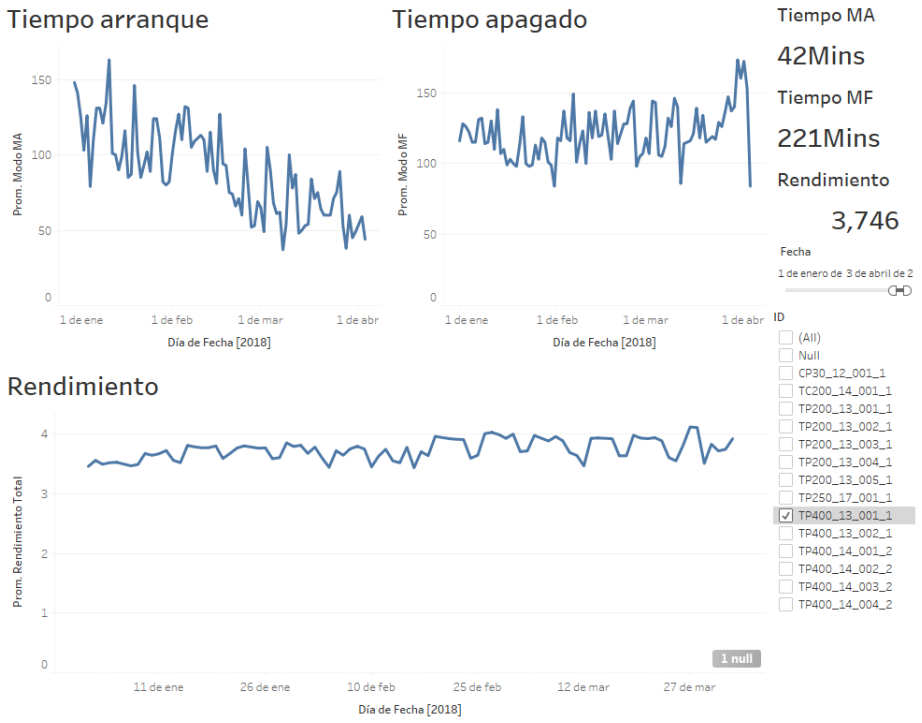

Figura 10: Panel de mando – Mantenimiento predictivo

- **Indicadores de caracterización demanda:**

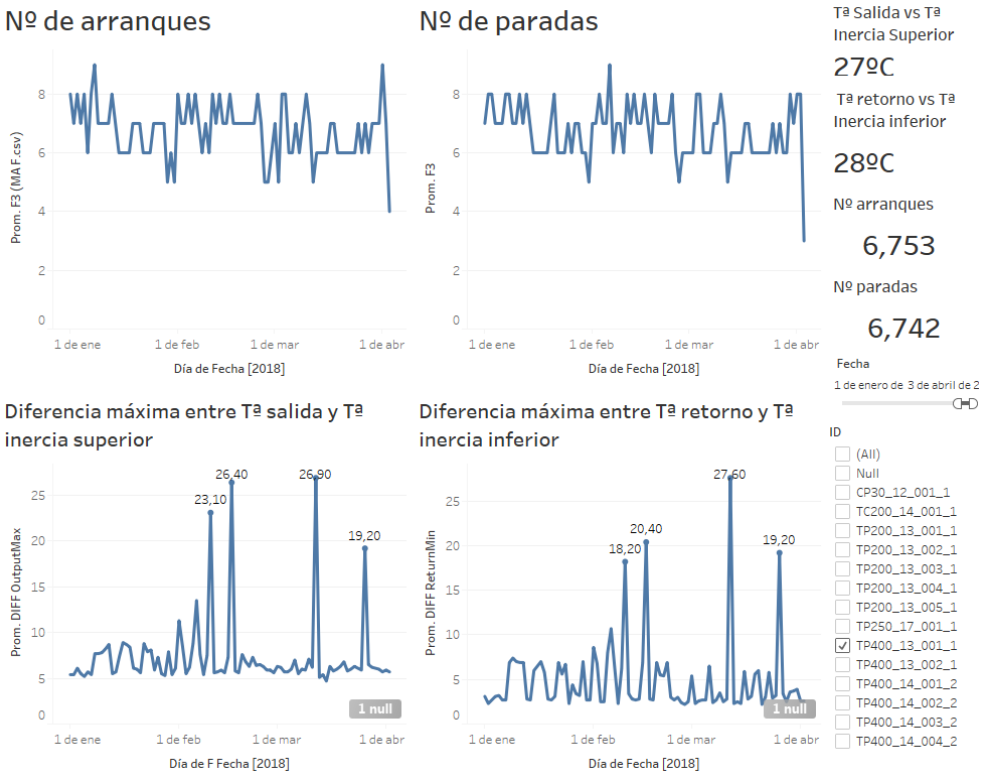

Figura 11:Panel de mando – Caracterización demanda

7.2. Algoritmos predictivos

En este apartado se detallará los algoritmos que serán utilizados para realizar la previsión de demandas de consumos en cada instalación. Con fin de hallar el mejor modelo de predicción para el sistema, se realizará comparaciones entre 3 algoritmos de predicciones regresivos disponibles en la biblioteca de ML: Regresión lineal, regresión con bosque aleatorio, regresión con árbol de decisión reforzado con gradiente (GBTR).

Se utilizará la validación cruzada para la validación de los modelos generados por cada algoritmo y obtener el conjunto de parámetros que los optimizan. Por consiguiente, se repartirá los datos en múltiples particiones y se realizará tantas iteraciones como las particiones que se han generado. Cuando haya completado todas las iteraciones se calculará el valor promedio de la métrica de evaluación. Por último, en base a los resultados obtenidos en la validación cruzada se decidirán la configuración óptima de cada algoritmo.

Con los modelos de predicción definitivos se calcularán los valores de previsión de demandas de cada instalación y se incorporan estos resultados en el panel de mandos de los usuarios. Los

usuarios decidirán el modelo que vayan a utilizar en función de sus necesidades, si quieren ajustar la cantidad de carga o maximizar la reserva de combustibles.

Teniendo en cuenta que el volumen de datos recopilados está en constante incremento y el estado de las instalaciones sufren constante modificaciones, los modelos predictivos se deben actualizar constantemente para adaptarse a estas evoluciones y mejorarlos basándose en los nuevos datos adquiridos.

7.2.1. Regresión lineal

El algoritmo de regresión lineal intenta hallar la función lineal que mejor define la relación de dependencia que existe entre las variables X, Y, de forma que el modelo generado es capaz de predecir el valor aproximado del Y dado el valor de X.

El modelo se puede describir de la siguiente forma: **y = c + b*x**. Donde Y es el variable dependiente, c es un constante, b es el coeficiente de regresión y X es el variable independiente.

El proceso de aprendizaje de regresión lineal tiene como objetivo minimizar la función de pérdida especificado, que a su vez es regularizado por funciones de penalización. En el caso de Spark proporciona soportes para dos tipos de funciones de pérdidas [10] [11]:

Error Cuadrático:

$$\min_{\omega} \frac{1}{2n} \sum_{i=1}^{n} (X_i \omega - y_i)^2 + \lambda \left[\frac{1-\alpha}{2} ||\omega||_2^2 + \alpha ||\omega||_1 \right]$$

La función de pérdida Error Cuadrado es la función de pérdida más utilizada en el campo de la estadística. Mediante la cual se puede calcular la diferencia de una instancia concreta respecto el valor generado con el modelo de predicción.

Huber:

$$\min_{\omega,\sigma} \frac{1}{2n} \sum_{i=1}^{n} \left(\sigma + H_m \left(\frac{X_i \omega - y_i}{\sigma} \right) \sigma \right) + \frac{1}{2\lambda} ||\omega||_2^2$$

La función de pérdida Huber en comparación con la función de pérdida Error Cuadrado es más tolerante con los valores atípicos. Sin embargo, para su utilización ha de tenerse en cuenta que solo admite el método Rigde como función de regularización.

Los parámetros que se ajusta en el algoritmo son [11] [12]:

- **Elastic Net:** Este parámetro se utiliza para regularizar un modelo de regresión lineal empleando la combinación lineal de las penalizaciones de los métodos Lasso y Ridge. Cuando el parámetro toma el valor 0 el modelo se regularizará exclusivamente por la función de pérdida Ridge y cuando toma el valor 1 se regularizará exclusivamente por la función Lasso.

- **Función de pérdida:** Spark permite ajustar el modelo de regresión lineal mediante las dos funciones de pérdidas definidas anteriormente **Error Cuadrado** y **Huber**.

- **Iteración máxima:** Este parámetro define el número máximo de iteraciones que está permitido utilizar en el entrenamiento de modelos.

- **Parámetro de regularización (Lambda):** Este parámetro se utilizar para actualizar los parámetros Theta (vector de variables independientes) mediante el algoritmo de descenso de gradiente. De esta forma evita el sobreajuste del modelo.

7.2.2. Regresión con bosque aleatorio

Los bosques aleatorios es uno de los algoritmos más populares y más efectivos del campo de aprendizaje automática. El modelo generado por el algoritmo se compone de múltiples modelos de árboles que se han generado de forma independiente con diferentes subconjuntos del conjunto de datos. A la hora de tomar la decisión, en este tipo de modelos se combinan las decisiones

tomadas por los diferentes modelos de árboles, de tal forma que se puede expresar el modelo de la siguiente forma [13]:

$$g(x) = f_0(x) + f_1(x) + f_2(x) + \cdots$$

Los parámetros que se ajusta en el algoritmo:

- **Número de árboles:** Este parámetro sirve para determinar la cantidad de árboles que se generarán durante el entrenamiento que posteriormente se combinarán para calcular la predicción.

- **Profundidad máxima:** Este parámetro delimita los niveles de decisiones que hay en los árboles. Al incrementar la profundidad de los árboles el resultado obtenido del modelo suele mejorar. Sin embargo, tiene el riesgo de sobreajuste del modelo.

7.2.3. Regresión con árbol de decisión reforzado con gradiente (GBTR)

Regresión con árbol de decisión reforzado con gradiente es una técnica ampliamente utilizada para llevar a cabo el proceso de aprendizaje automático. Es la aplicación combinada de la potenciación del gradiente con el modelo de predicción con árbol de decisión.

Los árboles reforzados con gradiente consisten en conjuntos de árboles de decisión que entrenan de forma iterada para minimizar el valor de la función. Esta técnica es capaz de manejar datos categóricos y multiclases, además no requiere parámetros escalado para el proceso.

El modelo de predicción generado tiene una estructura escalonada, de forma que las predicciones de las instancias se calculan basándose en las diferentes decisiones que se toman en cada nivel del modelo.

Los parámetros que se regula en el GBTR [14]:

- **Iteración máxima:** A igual que en el algoritmo de regresión lineal este parámetro define el número máximo de iteraciones permitidos.

- **Profundidad máxima:** Este parámetro tiene la misma función en el algoritmo de GBTR que en el algoritmo de regresión con bosques aleatorios. Sirve para definir los niveles de los árboles.

- **Tamaño de paso:** Este parámetro define el tamaño de paso que se utilizará en el algoritmo de descenso de gradientes en su búsqueda de minimiza la función.

7.3. Implementación de modelo predictivo

Con el fin de ilustrar de forma más detallada el procedimiento utilizado en el proyecto para la creación del modelo predictivo se utilizará como ejemplo el proceso de desarrollo de una de las instalaciones (A partir de aquí vamos a llamar la instalación ejemplo como instalación 1 que es un instituto). El mismo proceso se aplicará para todas las instalaciones.

7.3.1. Exploración y visualización de datos

Para poder realizar la exploración y visualización de datos, antes, se debe unificar los tres conjuntos datos disponibles: Datos de consumos, Datos de meteorología y Datos de calendario laboral/escolar.

Como primer paso, se unifica los datos de meteorología y datos de consumos. Para ellos se debe transformar los formatos de fechas de cada conjunto para que tengan el mismo formato. Una vez que las fechas están en el mismo formato se combinan los dos conjuntos de datos y se obtiene que el conjunto de datos combinados, que dispone de 29961 entradas, mientras el conjunto de datos del consumo original tiene 71145 entradas.

Esto significa que al conjunto de datos de meteorología obtenido a través del portal de la Junta de Castilla y León [18] le faltan 39253 registros para la localidad de **Población1 (La población más cercana a la instalación 1)** desde la fecha de 20/01/2014 hasta 02/04/2018. Teniendo en cuenta que los 39253 registros se suponen un 55% de datos totales, se deberá tratar de obtener los datos ausentes a partir de los registros de los datos de las estaciones de meteorología de localidades más próximas que son las estaciones de **Población2** y **Población3**. El resultado final obtenido de la operación es un conjunto de datos con 69214 instancias, es decir sigue habiendo unas 1931 instancias sin datos, pero teniendo en cuenta que solo supone un 2.7 % de datos totales se omitirá estos datos.

Como resultado final del proceso anterior se obtiene los siguientes parámetros:

- **D_SEG_SINFIN:** Variable directamente ligado con el consumo de cada instalación que se ha creado con una frecuencia de cada minuto y para que sea compatible con los datos de meteorología, es agrupado con una frecuencia de cada media hora. Como resultado el conjunto de datos cuenta con 69214 instancias, de los cuales solo 20165 instancias tienen valores distintos de 0.

$$pellets\ consumido\ (kgs) = D\_SEG\_SINFIN\ * Constante\ de\ Caldera$$

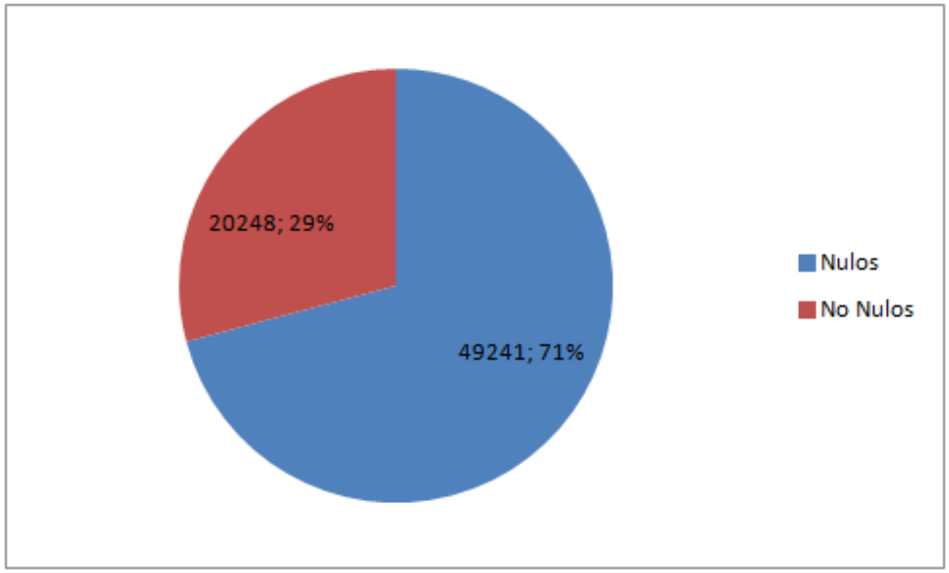

Gráfico 1: Gráfico de tarta D_SEG_SINFIN

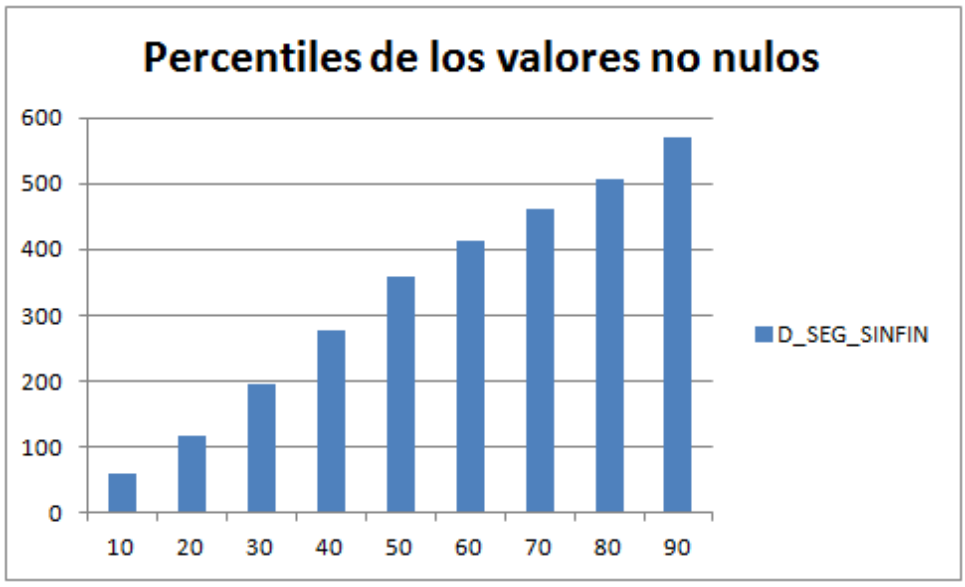

Gráfico 2: Gráfica de barras – percentiles

- **Temperatura:** Este parámetro toma valores desde –12.35 ºC hasta 36.07 ºC, aunque las instancias de las temperaturas extremas son muy escasas.

En el segundo diagrama *[Gráfico 4*: Gráfica de barras – D_SEG_SINFIN promedio por Temperatura*]*se puede observar una clara dependencia entre las variables de temperatura y D_SEG_SINFÍN. Cuando menor sea la temperatura mayor es el valor de D_SEG_SINFÍN, es decir más materiales consume. Sin embargo, esta dependencia tiene un comportamiento anormal para las temperaturas extremas. La causa de esta irregularidad se debe a la escasez de las instancias registradas para estos rangos de temperaturas, lo que les hacen extremadamente susceptibles antes los valores atípicos.

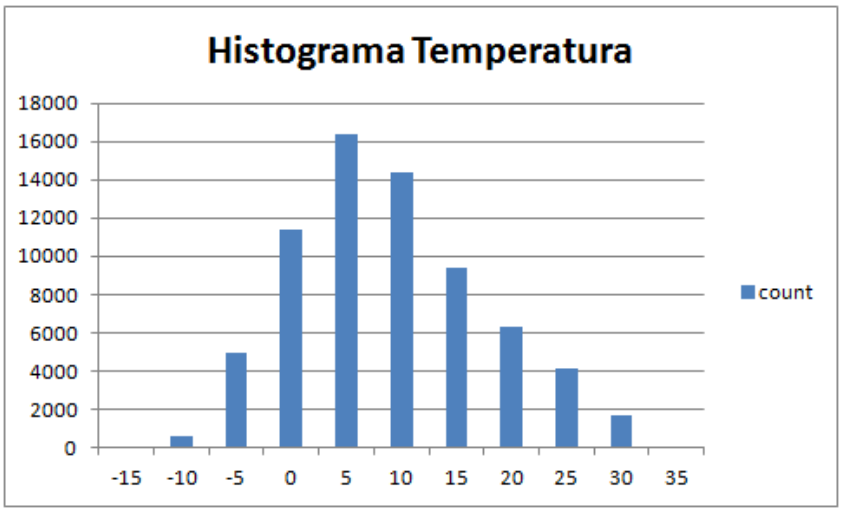

Gráfico 3: Histograma Temperatura

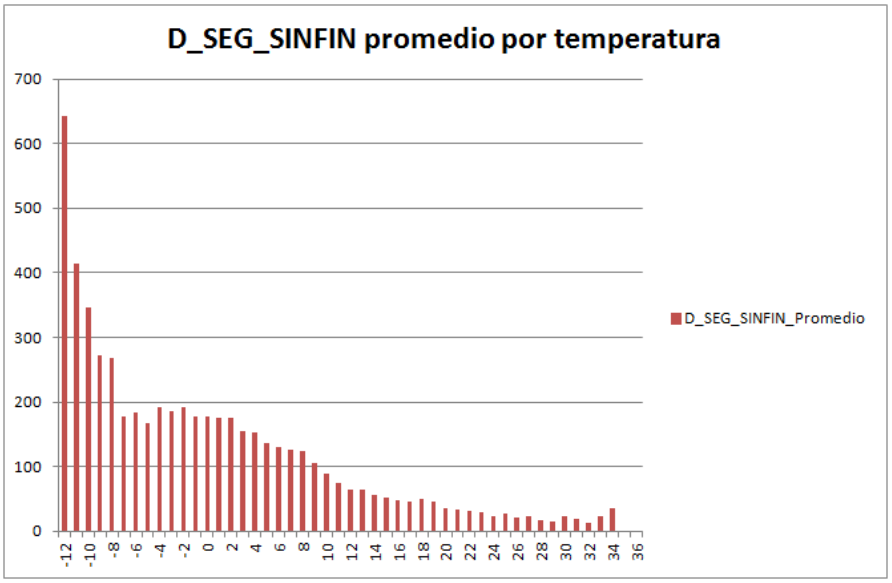

Gráfico 4: Gráfica de barras – D_SEG_SINFIN promedio por Temperatura

- **NoLectivo:** Este parámetro puede tomar como valor "True" y "False". Siendo, el valor "True" indica que es un día no lectivo y el valor "false" indica que es un día lectivo. Un 32% de las instancias registradas corresponder a los días no lectivos y 68% corresponder a los días lectivos. El valor promedio del D_SEG_SINFÍN para las instancias de los días lectivos es 2 veces más que las instancias de los días no lectivos, un indicador claro de la influencia del parámetro sobre el valor de D_SEG_SINFÍN.

Gráfico 5: Gráfico de tarta - NOLECTIVO

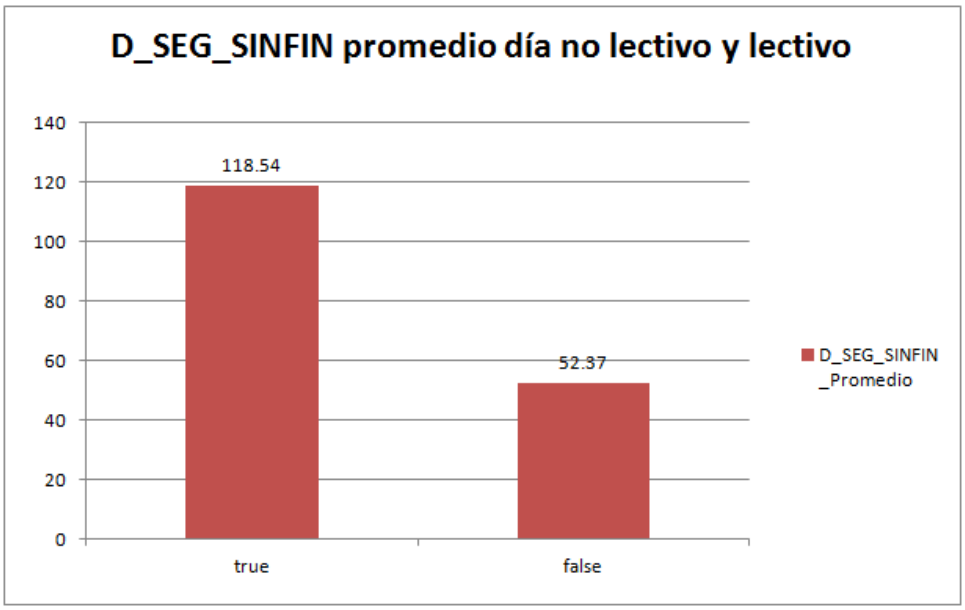

Gráfico 6: Gráfica de barras – D_SEG_SINFIN promedio por NOLECTIVO

- **ANIO:** Como se puede observar en la gráfica de valor promedio *[Gráfico 7]* el primer valor y el ultimo valor son muy distinto de los valores intermedios, esto se debe a que el primer año de la instalación empezó en los meses de verano por lo tanto tiene un promedio de consumo mucho menor. Mientras el registro de ultimo año finalizó en el inicio de primavera los días con menor temperatura del año por lo tanto tiene un valor mucho mayor.

  Para los 3 años intermedios se puede observar una ligera subida de consumo probablemente debido al deterioro de la instalación conforme al avance del tiempo.

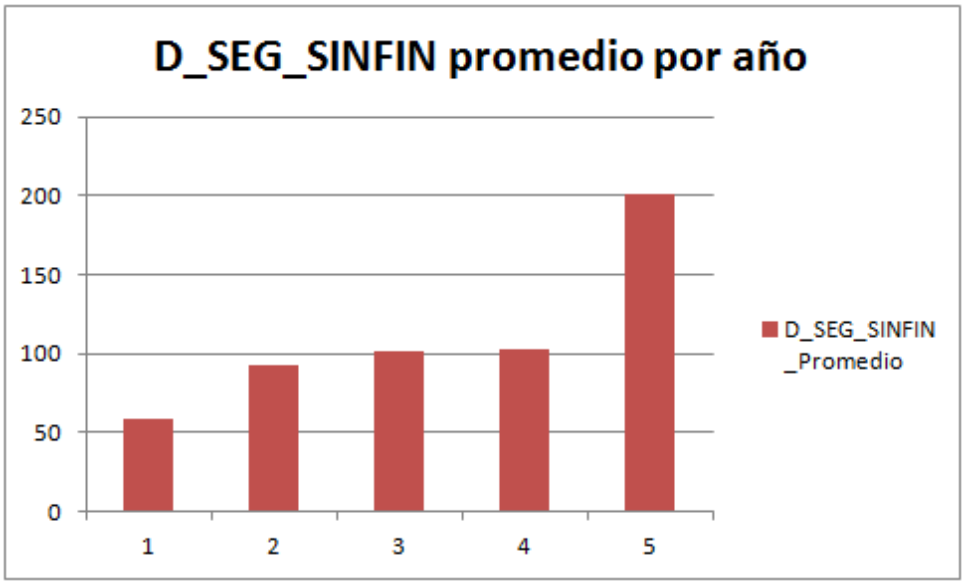

Gráfico 7: Gráfica de barras – D_SEG_SINFIN promedio por ANIO

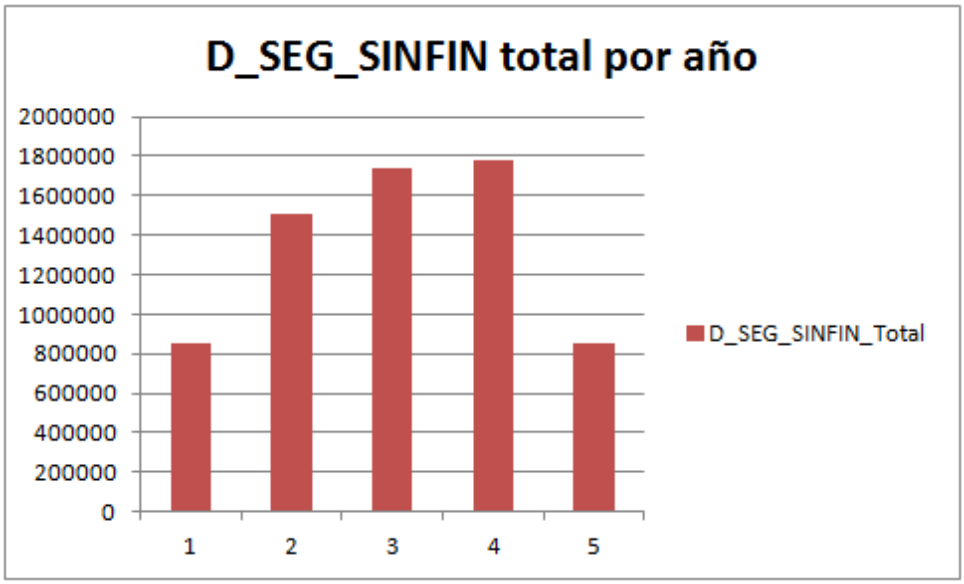

Gráfico 8: Gráfica de barras – D_SEG_SINFIN total por ANIO

- **Mes:** este parámetro puede tomar valores de 1 a 12 siendo 1 enero y 12 diciembre. Mediante el diagrama *[Gráfico 9]* se puede identificar una clara dependencia de la variable respecto al valor de D_SEG_SINFÍN.

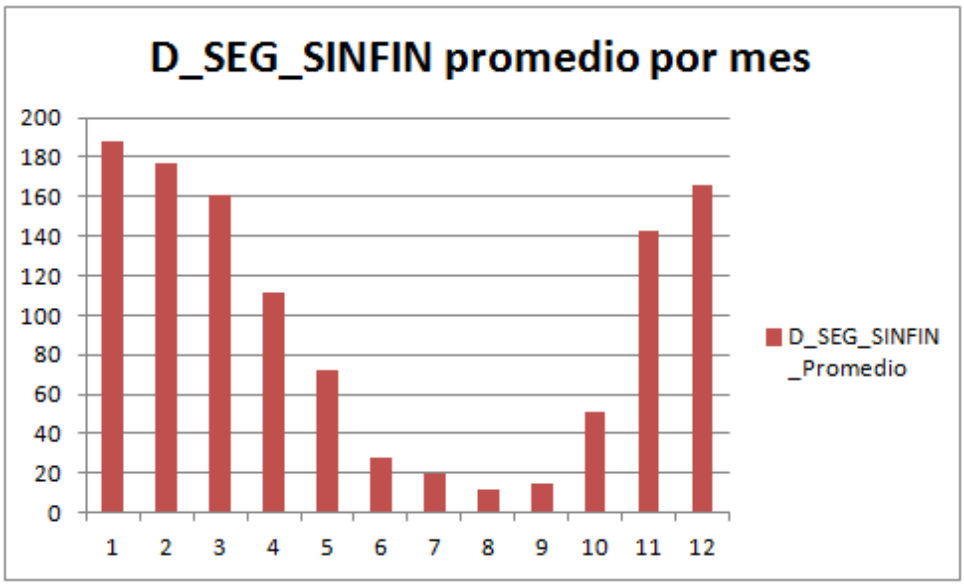

Gráfico 9: Gráfica de barras – D_SEG_SINFIN promedio por Mes

- **Dia_Mes:** La grafica de distribución promedio de G_SED_SINFIN obtenido para este parámetro indica que el parámetro en cuestión no guarda una relación de dependencia clara con el valor a predecir.

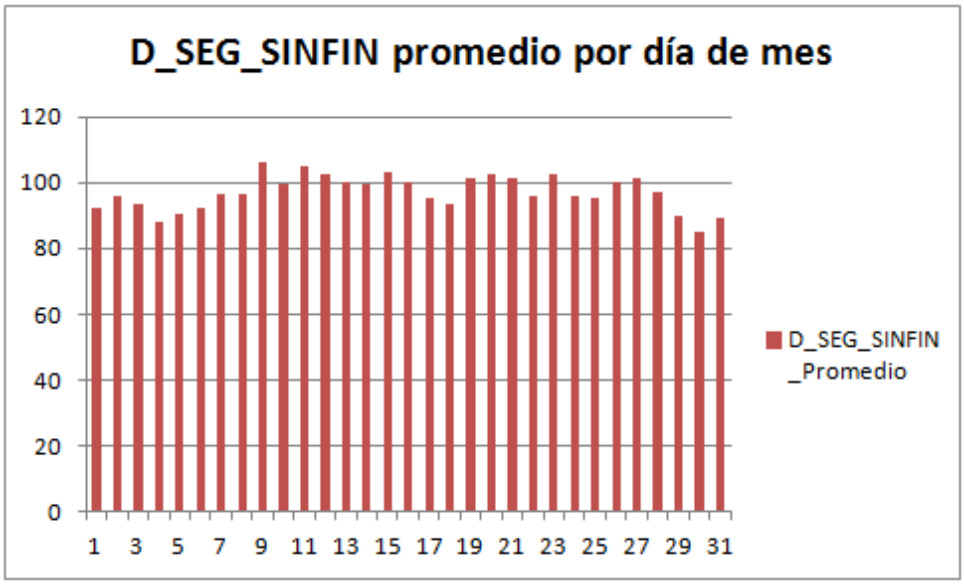

Gráfico 10: Gráfica de barras – D_SEG_SINFIN promedio por Dia_Mes

- **Dia_Semana:** Como se puede apreciar en la gráfica inferior *[Gráfico 11]* los días laborales de la semana tienen un promedio de consumo significativamente superior a los días de fines de semana.

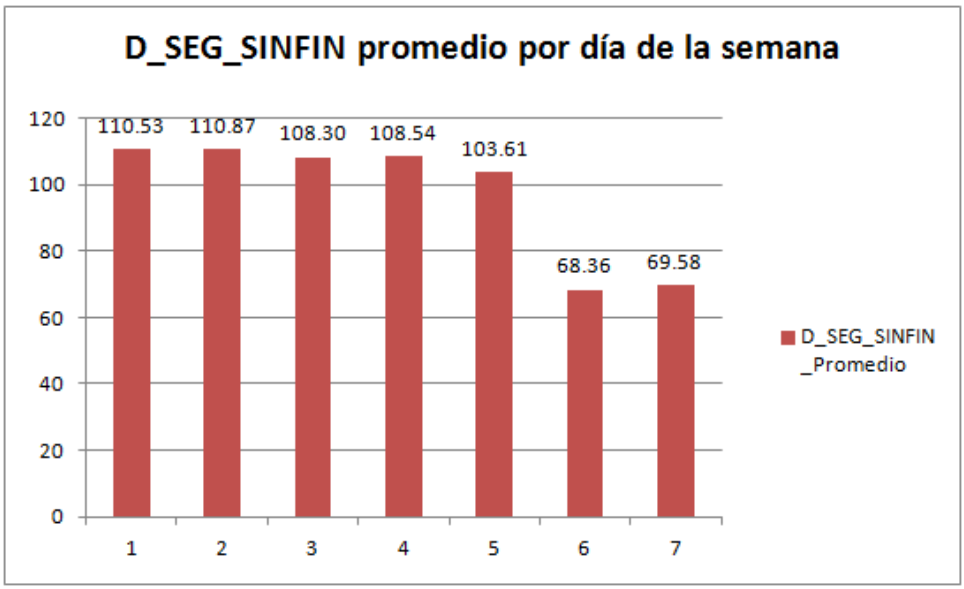

Gráfico 11: Gráfica de barras – D_SEG_SINFIN promedio por Dia_Semana

- **Hora:** Este parámetro puede tomar valores desde 0 hasta 47, es decir que cada unidad de este parámetro equivale a media hora real. La gráfica inferior de consumo promedio *[Gráfico 12]* indica que la instalación se encuentra en su periodo de mayor consumo de 14 (7:00) a 28 (14:00) que coincidente precisamente con el horario de las clases.

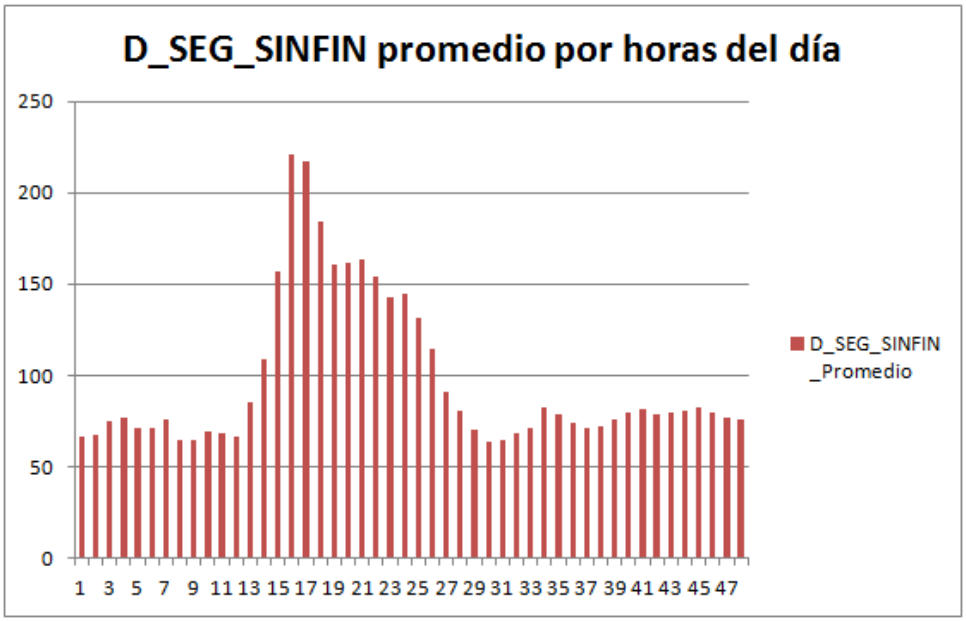

Gráfico 12: Gráfica de barras – D_SEG_SINFIN promedio por Hora

- **Precipitacion:** Parámetro tomado con una precisión de una décima de milímetro siendo el valor mínimo registrado 0.0 y el valor máximo registrado 22.9. Basándose en la gráfica inferior se puede deducir que casi la totalidad de conjunto de datos tiene como valor 0.0, mientras para el resto de los valores no dispone de suficientes entradas.

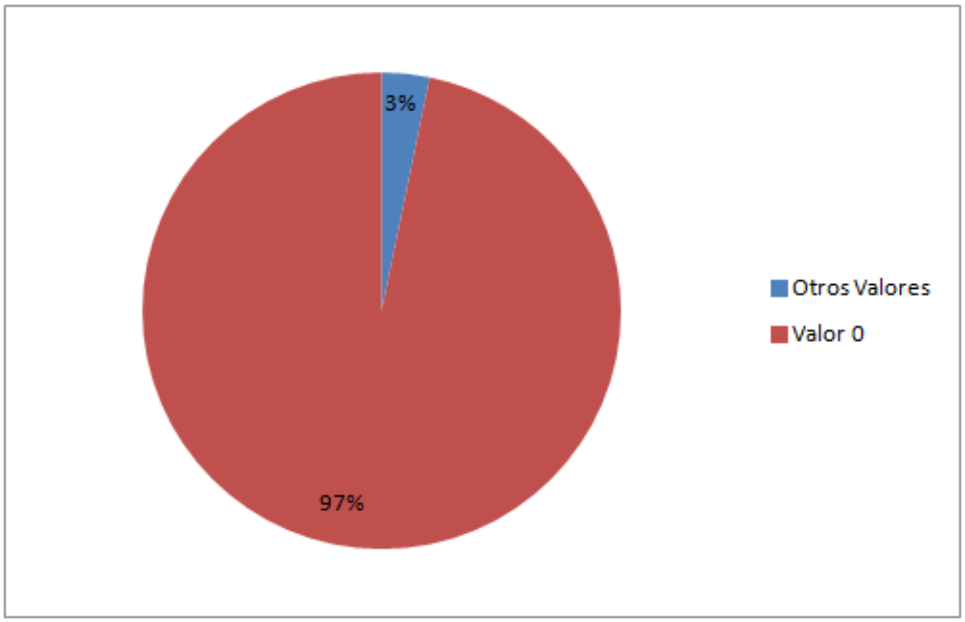

Gráfico 13: Gráfico de tarta – Precipitación

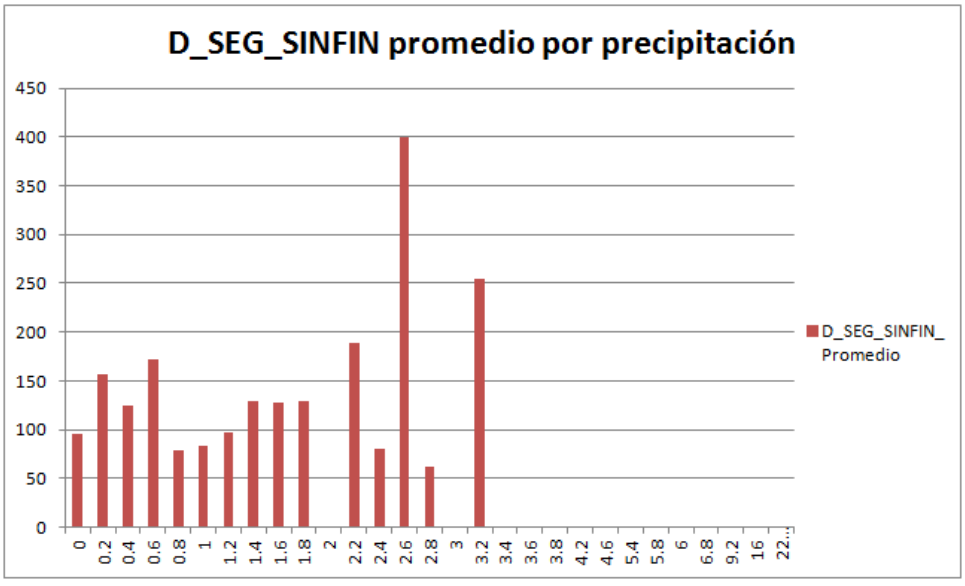

Gráfico 14: Gráfica de barras – D_SEG_SINFIN promedio por Precipitacion

- **Humedad:** Como puede observar en la gráfica *[Gráfico 15]* este parámetro guarda una relación de dependencia clara con el parámetro D_SEG_SINFÍN. Cuando mayor es el valor de humedad mayor es el valor de D_SEG_SINFÍN. Excepto para los valores cercanos a 0% que cuentan muy pocas instancias y que lo hace muy susceptible ante los valores atópicos.

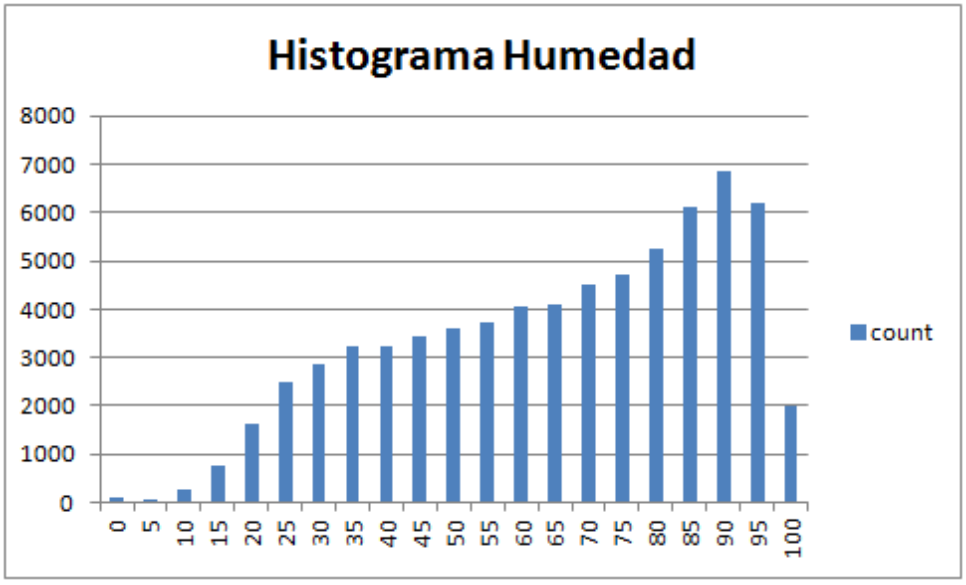

Gráfico 15: Histograma Humedad

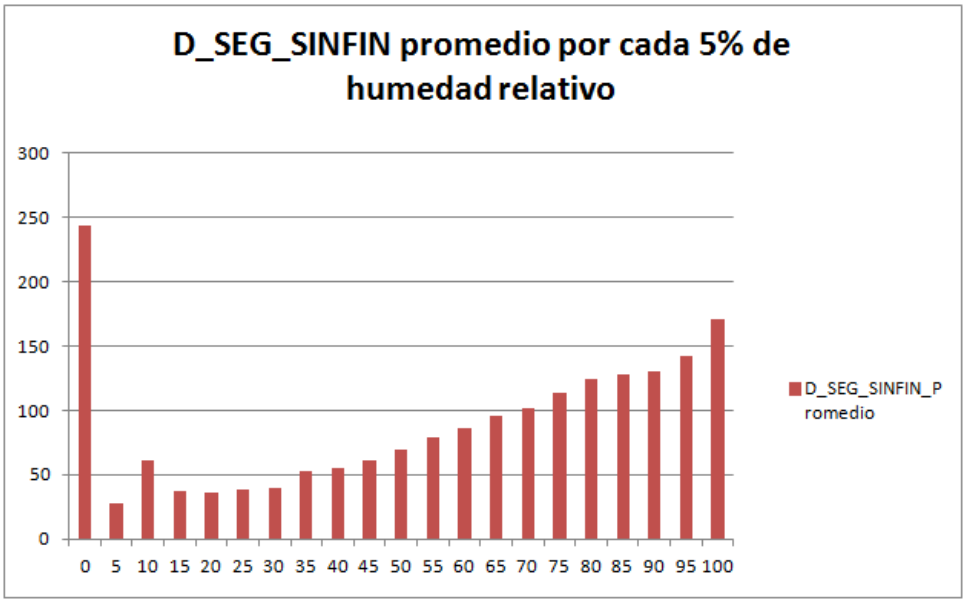

Gráfico 16: Gráfica de barras – D_SEG_SINFIN promedio por Humedad

- **Radiacion:** este parámetro indica la radiación solar percibido en la localidad, se ha registrado valores desde 0.0 hasta 1237.0. Aunque más de un 50% de las instancias registradas tienen como valor 0.0. En la *[Gráfico 18]* se puede apreciar, a medida que aumenta el valor de la radiación el valor de D_SEG_SINFÍN se ve disminuido ligeramente.

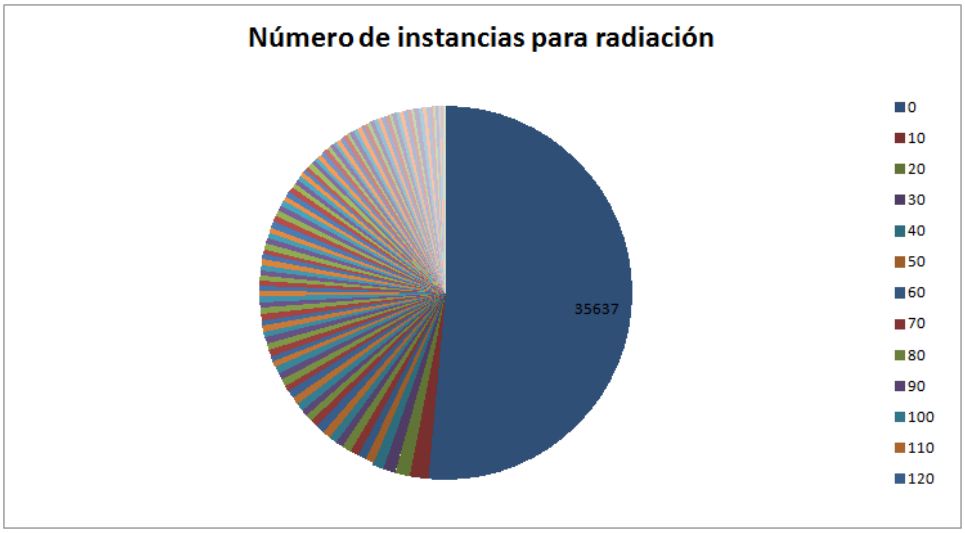

Gráfico 17: Gráfico de tarta – Radiacion

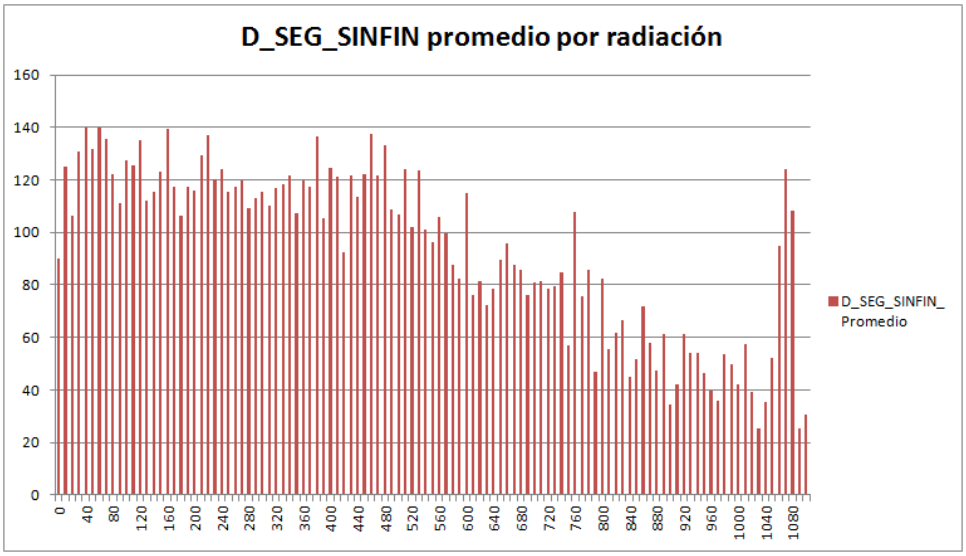

Gráfico 18: Gráfica de barras – D_SEG_SINFIN promedio por Radiacion

- **Vel.Viento:** Este parámetro se ha registrado valores desde –12.47 hasta 17.05, aunque la mayor parte de los datos se concentran entre el rango de 0 a 7. Mediante la gráfica *[Gráfico 20]* se puede observar que la velocidad de viento no tiene un efecto apreciable sobre el valor de D_SEG_SINFÍN. Excepto para los valores muy altos de viento que se ver muy incrementado los valores de D_SEG_SINFÍN, pero para estos rangos hay muy pocas entradas.

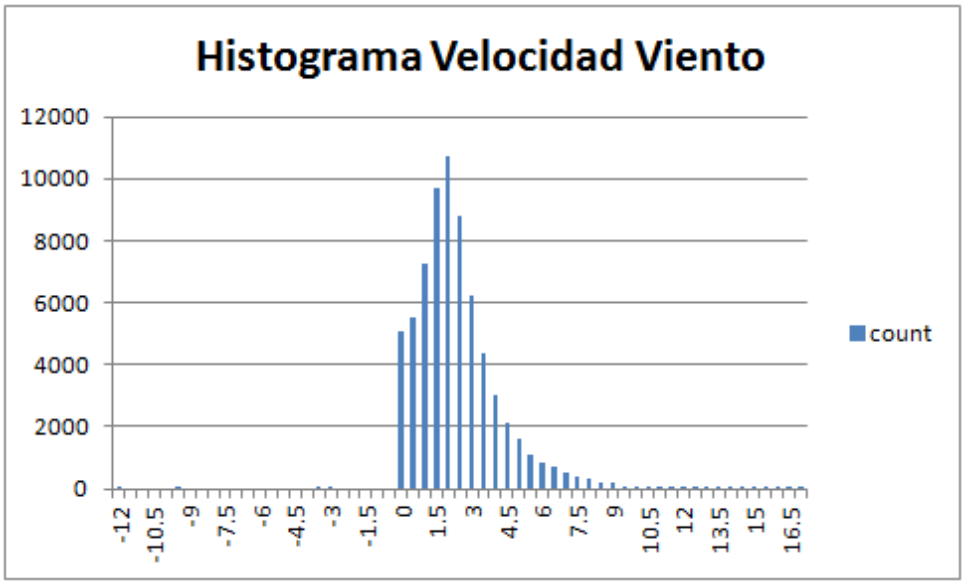

Gráfico 19: Histograma Vel.Viento

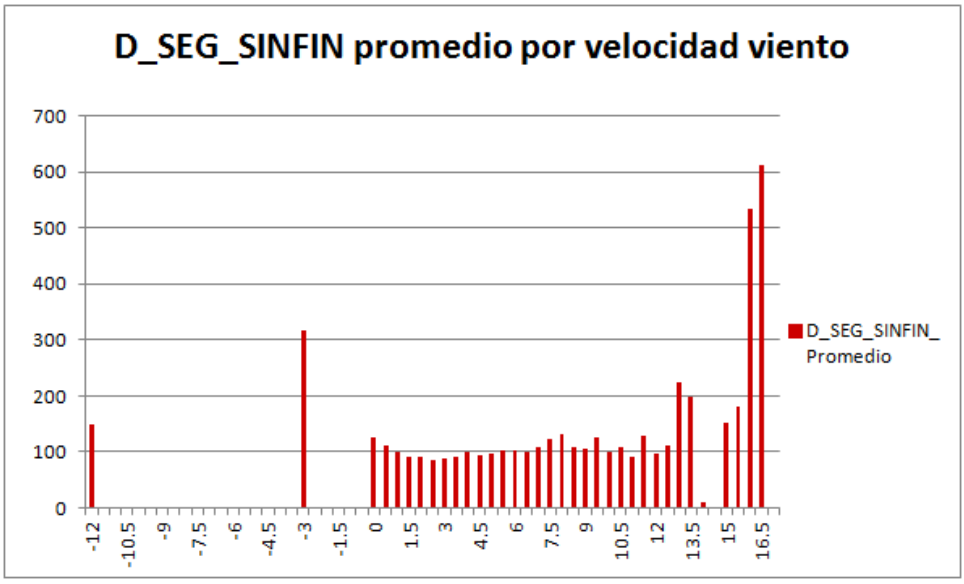

Gráfico 20: Gráfica de barras – D_SEG_SINFIN promedio por Vel.Viento

- **Dir.Viento:** parámetro medido en grados que puede tomar valores desde 0 a 360. Se ha divido los datos en rango de 0.5 grados para que disponga de suficiente entradas en cada intervalo.

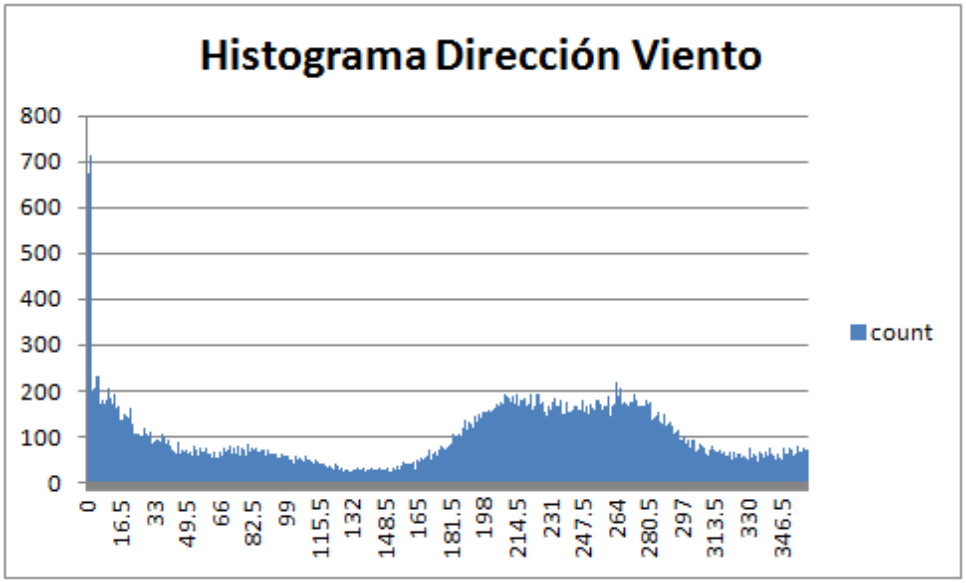

Gráfico 21: Histograma Dir.Viento

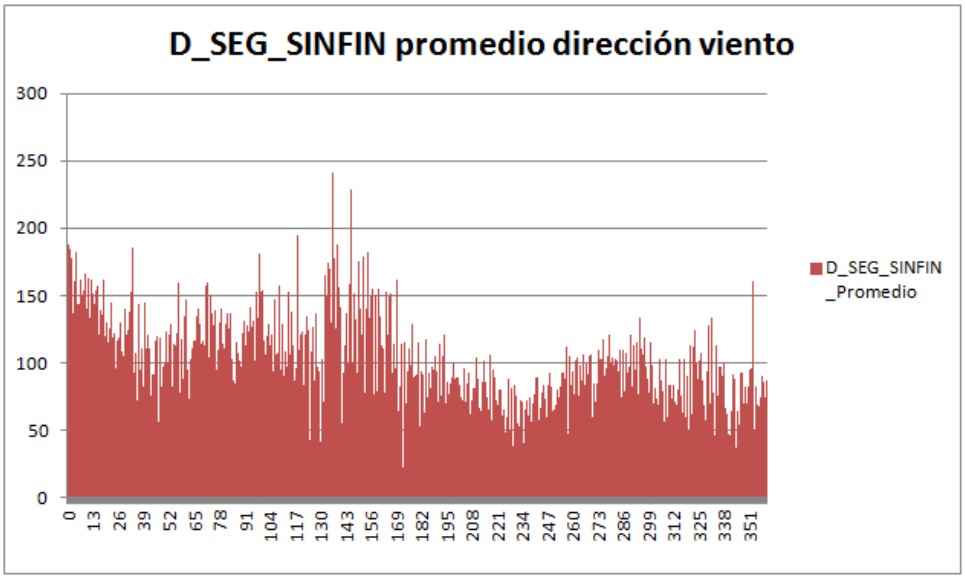

Gráfico 22: Gráfica de barras – D_SEG_SINFIN promedio por Dir.Viento

- **Conclusión:**

Como conclusión para la fase de exploración y visualización de datos se puede concluir que entre el parámetro Dia_Mes y D_SEG_SINFÍN no existe una relación de dependencia. El parámetro Precipitacion para casi la totalidad de las instancias registradas tiene como valor 0. En cuando a parámetro Vel.Viento no tiene una relación de dependencia clara con respecto al

parámetro de D_SEG_SINFÍN y para los casos de los valores extremos no dispone de las instancias suficientes como para poder tenerlo en cuenta en el modelo de predicción. Razones por los cuales los parámetros Dia_Mes, Precipitacion y Vel.Viento serán excluidos del modelo.

7.3.2. Tratamiento de valores atípicos

Los datos obtenidos del mundo real a menudo contienen datos erróneos, datos nulos y valores atípicos. Tanto los datos erróneos como los datos nulos ya son tratados en la sección anterior. En esta fase del desarrollo se aplicará técnicas de detección de valores atípicos para identificar estas instancias y aplicar transformaciones convenientes para suprimir o reducir el impacto de estos datos sobre el modelo predictivo. [10]

7.3.2.1.MAD

Desviación Absoluta de Mediana (MAD) es una técnica estadística que se utiliza para detectar los valores atípicos del conjunto de datos de una sola dimensión. La idea central de esta técnica es establecer intervalo de aceptación alrededor de la mediana. Se puede calcular el intervalo de aceptación con la siguiente formula:

$$MAD = median(|X_i - median(X)|)$$

Siendo *median(x)* mediana de conjunto de datos y *Xi* las instancias del conjunto de datos y |*Xi* – *median(x)*| el valor absoluto de la desviación de la instancia respecto la mediana. Es una variación del promedio de la desviación absoluta que se ve menos afectado por los datos extremos, puesto que se calcular a partir de la mediana en lugar de la media que es altamente sensible ante los datos extremos. [19]

7.3.2.2.Criterio de Chauvenet

El criterio de Chauvenet es otra alternativa para realizar la detección de valores atípicos y es ampliamente utilizado en sectores como astronomía, tecnología nuclear, epidemiología y en otros muchos campos de la ciencia.

La idea subyacente en el criterio de Chauvenet consiste en establecer dos bandas simétricas de aceptación o limites alrededor de la media y todos aquellos valores que se encuentran fuera de esas bandas son considerados como valores atípicos. Debido a que las bandas de aceptación se

establecen alrededor de la media resulta que esta técnica es muy sensible ante los valores extremos a diferencia de MAD. [20] [21]

Con el fin de poder determinar los valores atípicos se necesita calcular los valores de Z-Score de cada instancia que posteriormente servirá para comparar con las bandas de aceptación. Z-Score es una medida que indica la diferencia existente entre la instancia y el promedio de la población en función de la desviación estándar y se puede calcular con la siguiente fórmula:

$$Z = \frac{X - \mu}{\sigma}$$

Siendo $\boldsymbol{x}$ la instancia, $\boldsymbol{\mu}$ la media *y* $\boldsymbol{\sigma}$ la desviación estándar del conjunto.

En cuanto, el cálculo del valor límite para el Z-Score se puede realizar de dos formas diferentes:

- Mediante la tabla de criterio de Chauvenet:

N	T
3	1.383
4	1.534
5	1.645
6	1.732
7	1.803
8	1.863
9	1.915
10	1.960
11	2.000
12	2.037
13	2.070
14	2.100
15	2.128
16	2.154
17	2.178
18	2.200
19	2.222
20	2.241
21	2.260
22	2.278
23	2.295
24	2.311
25	2.326

26	2.341
27	2.355
28	2.369
29	2.382
30	2.394
31	2.406
32	2.418
33	2.429
34	2.440
35	2.450
36	2.460
37	2.470
38	2.479
39	2.489
40	2.498
50	2.576
100	2.807
500	3.291
1000	3.481

**** T: Desviación estándar máxima permitida***

**** N: número de entradas.***

Desde la tabla anterior se selecciona el valor T con el valor de N más aproximo al número de las instancias que hay en el conjunto de datos como el valor límite de Z-Score. Todas aquellas instancias con Z-Score que exceden de ese valor son consideradas como valores atípicos.

No obstante, esta forma de calcular valor límite de Z-Score está muy limitada y es poco precisa.

- Mediante la función de probabilidad acumulativa inversa de la distribución normal:

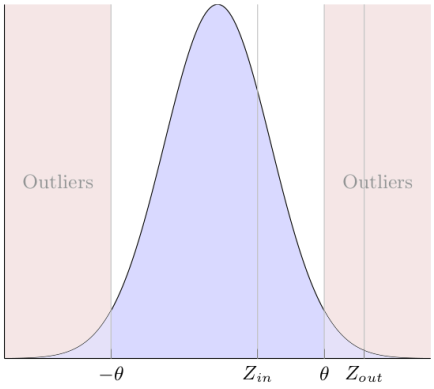

Figura 12: Distribución normal [21]

```
def calculateZScoreLim(size: Long): Double = {
    val pv = 1-(1/(2.0*size))
    val standardNormal = new NormalDistribution(0, 1);
        val infNorm = standardNormal.inverseCumulativeProbability(pv);
    return infNorm;
}
```

En primer lugar, se calcular el p-valor del conjunto y mediante el p-valor y la función de probabilidad acumulativa inversa se obtiene el valor de θ que se establecerá como el valor límite de Z-Score del conjunto.

7.3.2.3.D_SEG_SINFIN

La técnica de detección de valores atípicos con MAD es una técnica más robusta en comparación con la técnica de detección de valores atípicos con Criterio de Chauvenet, ya que se ve menos afectado por los valores extremos. Sin embargo, no se puede aplicar la técnica de detección de valores atípicos mediante MAD para el presente trabajo fin de máster, dado que más de un 50% de los datos del conjunto tienen el mismo valor. Por lo tanto, en este caso se aplicará la técnica de detección de valores atípicos con Criterio de Chauvenet. Para ellos primero ha de hallar la media y la desviación estándar del conjunto.

```
val mean = data.agg(avg(column)).first().getDouble(0);
```

val stddev = data.agg(stddev_pop(column)).first().getDouble(0);

Media: 97.19

Desviación estándar: 186.27

A continuación, se calcula los valores de Z-Score para cada instancia con la formula $Z = \frac{X-\mu}{\sigma}$, con el valor de Z-Score calculado se procede con el cálculo del valor límite de Z-Score mediante la función de probabilidad acumulativa inversa.

El valor límite para un conjunto de 69214 instancias es: 4.33694.

Como resultado de la operación, **67** entradas son clasificadas como valores atípicos que serán eliminadas del conjunto.

7.3.2.4.Temperatura

A igual que el parámetro D_SEG_SINFIN, el parámetro temperatura es numérica y se necesita tratarlo con el criterio de Chauvenet para eliminar los valores atípicos. Siguiendo el mismo proceso aplicado para el tratamiento del parámetro D_SEG_SINFIN, en primer lugar, se calcula el valor promedio y la desviación estándar del conjunto:

Media: 11.382

Desviación estándar: 8.638

Finalmente se calcula el valor Z-Score de cada instancia y se compara el valor obtenido con el valor de Z-Score limite.

El valor límite para un conjunto de 69147 instancias es: 4.33672.

De donde resulta que ninguna instancia del conjunto es marcada como valor atípico.

7.3.2.5.Humedad

El parámetro numérico Humedad está previamente tratado para prescindir de los valores atípicos y para normalizar el conjunto de datos. De manera que el parámetro Humedad no requiere ningún tipo de tratamiento adicional antes del entrenamiento del modelo.

7.3.2.6.Radiacion

El conjunto de datos del parámetro Radiacion se debe someter los mismos procesos aplicados a los parámetros de D_SEG_SINFIN y Temperatura. En primer lugar, se calcula el valor promedio y la desviación estándar del conjunto:

Media: 201.401

Desviación estándar: 291.653

Para acabar se calcula el Z-Score de cada instancia y se filtra los valores calculados con el límite establecido para el Z-Score.

El valor límite para un conjunto de 69147 instancias es: 4.33672.

Para concluir no se ha detectado ninguna instancia como valor atípico.

7.3.2.7.ANIO

Parámetro numérico que solo puede tomar valores enteros positivos incluyendo el número cero. Donde el número 0 corresponde al año en el que se completó la instalación y 1 el segundo año después de la instalación y así sucesivamente.

7.3.3. Normalización

La normalización de los parámetros es algo esencial para muchos de los algoritmos de aprendizaje automático tales como SVM (Máquinas de Vectores de Soporte), K-NN (K Vecinos más próximos), Regresión Logística, Redes Neuronales...

El proceso de normalización de los datos consiste en re-escalar los parámetros para mejorar el comportamiento durante el entrenamiento de manera que el modelo predictivo pueda aprovechar el mayor rendimiento posible de los datos.

Existen muchas formas de normalizar los datos [10] [22] [23] [24] [25]:

- **Escala Característica (min-max scaling):** La técnica de normalización Escala Característica también conocido como "normalización". Es una técnica que se suelen usar para asegurar que los datos estén dentro de un rango determinado habitualmente de (0,1). La desventaja de utilizar esta técnica es que puede amplificar el ruido de los datos al limitar el rango.

$$X_{normalized} = \frac{X - X_{min}}{X_{max} - X_{min}}$$

- **Estandarización:** Estandarización o normalización Z-Score, la aplicación de esta técnica sobre el conjunto de datos da lugar a un nuevo conjunto de datos que sigue la distribución normal donde la media igual a 0 y la desviación estándar vale 1. Razón por el cual esta técnica suele ser utilizado para normalizar los datos con fin de permitirles realizar comparaciones entre conjuntos de datos que tengan diferentes unidades.

  La estandarización se suele utilizar para optimizar algoritmos como regresión logística, SVM, K-NN etc.

  A igual que muchas otras técnicas basadas en la media y en la desviación estándar, uno de los inconvenientes de esta técnica es que resulta ser extremadamente sensible ante los valores atípicos. Por lo tanto, antes de aplicarla sobre el conjunto de datos se requiere un tratamiento de los valores atípicos.

$$X_{normalized} = \frac{X - X_{mean}}{X_{stddev}}$$

- **Escalador máximo absoluto:** Esta técnica funciona de manera muy similar la Normalización, aunque la salida de esta técnica está limitada en un rango de [-1, 1] en lugar de [0, 1]. Está diseñado para ser utilizado en conjunto de datos que esta centralizado en el cero.

$$X_{normalized} = \frac{X}{X_{max}}$$

- **Escalador robusto:** Esta técnica de normalización re-escalan los datos de acuerdo con el rango intercuartílico (IQR). De manera que el proceso es robusto frente a los valores atípicos.

Llegado a este punto hay que mencionar la importancia del Estandarización para el análisis de los componentes principales (PCA), el algoritmo que se va a utilizar para reducir la dimensión de los datos. En PCA el objetivo principal es encontrar la dirección de los componentes que maximizan la varianza que hay entre los mismos. Si dos variables tienen diferentes unidades (por ejemplo, metros y kilos) entonces no se puede realizar una comparación directa entre las varianzas de las variables ya que no tienen la misma unidad. Por lo tanto, es preciso aplicar la Estandarización a los datos antes de realizar el análisis de los componentes principales para que todas las variables estén en una misma escala y pueden realizar comparaciones entre sí.

Volviendo ahora la vista al trabajo presente, para los algoritmos que serán utilizados en el proyecto tanto regresión lineal, regresión con Árboles aleatorios como regresión GBT no resulta imprescindible la normalización de las características. Sin embargo, es muy conveniente su utilización ya que mejoraría significativamente el rendimiento de los datos. En conclusión, en esta fase de normalización se aplica las dos técnicas anteriores (Estandarización, Escala Característica) sobre los parámetros Temperatura y Radiacion para así poder contrarrestar posteriormente la efectividad de cada una de ellas sobre el conjunto de datos.

Normalización:

```
val normalize_result = data.withColumn("Temperatura", (col("Temperatura")-min_Temperature)/ (max_Temperature - min_Temperature)).withColumn("Radiacion",(col("Radiacion")-min_Radiation)/(max_Radiation-min_Radiation));
```

Estandarización:

```
val standardize_result = data.withColumn("Temperatura", (col("Temperatura")-mean_Temperature)/(stddev_Temperature)).withColumn("Radiacion",(col("Radiacion")-mean_Radiation)/(stddev_Radiation))
```

7.3.4. Categorización

Las características categóricas son características que pueden tomar un valor o una lista de posibles valores limitados. Esto significa que no se puede incluirlos directamente como entradas para el entrenamiento del modelo, se necesita transformarlos de tal forma que el Spark pueda interpretarlos como categórica.

Para dotarles de las propiedades de una característica categórica se suele codificarlos con la técnica conocido como **1-of-k**. Esta técnica transforma las características en vectores binarios con K columnas, donde cada columna puede tomar valores de uno y cero. Todas las columnas del vector tienen valor 0 excepto la columna de índice que corresponde al estado de la variable. Por ejemplo, para representar una característica que puede tomar como valor estudiante y programador. Se representará de la siguiente forma:

Estudiante → [1, 0]

Programador → [0, 1]

Aclarado lo anterior, a continuación, se detallará las transformaciones que se aplicarán sobre las características del proyecto para categorizarlos. [10]

7.3.4.1.DIA_SEMANA

Parámetro categórico originalmente puede tomar 7 valores diferentes desde 1 hasta 7. Ahora bien, mediante la visualizar del diagrama de barras con este parámetro como dimensión y valor promedio del consumo como medida *[Gráfico 11]* y teniendo en cuenta las características del establecimiento donde está localizado la instalación que se trata de un instituto. Por estas razones se puede reestructurar este parámetro en dos categorías muy dispares una categoría que incluye los días lectivos de la semana (1 - 5) y otra categoría que incluye los días no lectivos de la semana (6, 7).

7.3.4.2.NOLECTIVO

Este parámetro puede tomar dos posibles valores “true” si no es un día lectivo en la población donde se localiza la instalación y “false” si es un día lectivo.

7.3.4.3.MES

Mediante la visualización del diagrama de consumo promedio por mes *[Gráfico 9]* no se puede establecer una clara clasificación de los datos ya que los valores de cada mes varían de forma progresiva, en consecuencia, se reservará las 12 categorías originales del parámetro.

7.3.4.4.Hora

Se puede apreciar en el diagrama *[Gráfico 12]* dos áreas claramente distintas entre los valores correspondientes al rango de 14 (7:00 AM) a 28 (2:00 PM) que tienen una variación progresiva y el resto de los valores que tienen valores muy semejantes. Razón por el cual se puede inducir a una agrupación de los valores restantes en una sola categoría. Mientras los valores comprendidos entre 14 a 28 se conservan sus respectivas categorías, dando lugar así a 16 categorías en total.

Por otro lado, si se tiene en cuenta que el establecimiento de la instalación es un instituto se puede hacer valer el análisis anterior, ya que el horario de apertura del instituto esta comprendidos entre las 8:00 a 14:00 que corresponde aproximadamente con el rango comprendido entre 14 a 28.

7.3.4.5.DirViento

Los valores de este parámetro pueden variar desde 0 hasta 360, medidos en grados. Para diversificar este parámetro en categorías más reducidas se puede recurrir a la figura la brújula *[Figura 13]*. A partir del cual se puede establecer 8 rangos de los valores cada uno de ellos con una amplitud de 45 grados. Siendo el primer rango el área comprendido entre 0 y 45 o dicho de otra manera norte y noreste.

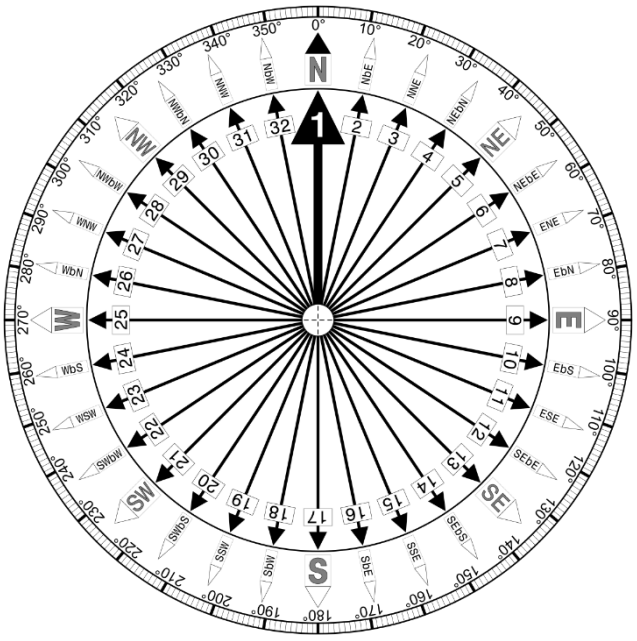

Figura 13: Brújula [26]

Ahora teniendo ya listo los parámetros categóricos se proceder con la categorización. Para ello se utiliza la herramienta proporcionada por la biblioteca ml de Spark concretamente la clase org.apache.spark.ml.feature.OneHotEncoderEstimator. Esta clase se encarga de transformar todas las columnas introducidas en vectores binarios de forma similar a la técnica de 1-of-k mencionado anteriormente.

```
val encoderInputFeatures = Array("DIA_SEMANA", "DirViento","Hora","MES","NOLECTIVO");

val encoderOutputFeatures = Array("DIA_SEMANA_C", "DirViento_C","Hora_C","MES_C","NOLECTIVO_C");

val encoder = new OneHotEncoderEstimator().setInputCols(encoderInputFeatures).setOutputCols(encoderOutputFeatures);

val categorizer= encoder.fit(data);
```

7.3.5. Reducción de dimensión

Es muy habitual que los datos brutos recogidos contengan características irrelevantes o contengan alguna forma de estructura subyacente o características latentes que se puede utilizar como sustituto para entrenar el modelo predictivo en lugar de los datos originales que probablemente contienen ruidos. Por lo tanto, resulta imprescindible extraer de los datos brutos una representación que evite las características irrelevantes y aprenda de las características latentes.

La idea subyacente a la reducción de dimensión es tomar un conjunto de datos con una dimensión D. Luego extrae del conjunto una representación del mismo con una dimensión K que suele ser mucho menor que el de D, pero sigue representando la mayor parte de la varianza posible de los datos originales. De manera que se consigue reducir el coste de la computación y suprimir parte de los ruidos provocados por las características redundantes. [10]

Existe varias técnicas para conseguir la reducción de dimensiones:

7.3.5.1. Análisis de componentes principales (PCA)

PCA fue desarrollado a finales de siglo XIX por Pearson, pero no fue popularizada su uso hasta la aparición de las computadoras, sobre todo con la aparición de aprendizaje automático.

PCA intenta extraer de los datos brutos un conjunto de componentes principales K donde los componentes no son correladas entre sí. Los componentes son computados de forma secuencial, seleccionando siempre el componente con mayor varianza posible de conjunto, con la condición de que el nuevo componente sea independiente de los componentes ya incluidos en el conjunto. [10] [25] [27] [28]

El proceso para calcular los componentes principales es la siguiente:

1. Partiendo de un vector aleatorio X:

$$X = \begin{pmatrix} X_1 \\ X_2 \\ \vdots \\ X_p \end{pmatrix}$$

2. Se calcular la matriz de varianza-covarianza del vector original:

$$var(X) = \Sigma = \begin{pmatrix} \sigma_1^2 & \sigma_{12} & \cdots & \sigma_{1p} \\ \sigma_{21} & \sigma_2^2 & \cdots & \sigma_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_{p1} & \sigma_{p2} & \cdots & \sigma_p^2 \end{pmatrix}$$

3. Se obtiene un vector con las variables $y_1, y_1, ..., y_p$ donde cada variable es una combinación lineal de las variables originales del vector X y se puede considerar $e_{i1}, e_{i2}, ..., e_{ip}$ como los coeficientes de regresión.

$$\begin{gathered} Y_1 = e_{11}X_1 + e_{12}X_2 + \cdots + e_{1p}X_p \\ Y_2 = e_{21}X_1 + e_{22}X_2 + \cdots + e_{2p}X_p \\ \vdots \\ Y_p = e_{p1}X_1 + e_{p2}X_2 + \cdots + e_{pp}X_p \end{gathered}$$

Se tiene la varianza poblacional y covarianza poblacional:

Varianza

$$var(Y_i) = \sum_{k=1}^{p}\sum_{l=1}^{p} e_{ik}e_{il}\sigma_{kl} = e_i'\Sigma\mathrm{e_i}$$

Covarianza

$$cov(Y_i, Y_j) = \sum_{k=1}^{p}\sum_{l=1}^{p} e_{ik}e_{jl}\sigma_{kl} = e_i'\Sigma\mathrm{e_j}$$

4. Se recoge los coeficientes e_{ij} en el vector:

$$e_i = \begin{pmatrix} e_{i1} \\ e_{i2} \\ \vdots \\ e_{ip} \end{pmatrix}$$

5. Una vez completado los pasos anteriores se procede con el cálculo del primer componente principal. El primer componente principal es el componente que se tiene la máxima varianza del conjunto. Es decir, el componente que maximiza el resultado de la formula siguiente:

$$var(Y_1) = \sum_{k=1}^{p}\sum_{l=1}^{p} e_{1k}e_{1l}\sigma_{kl} = e_1'\Sigma e_1$$

Y opera bajo la restricción de:

$$e_1'e_1 = \sum_{j=1}^{p} e_{1j}^2 = 1$$

6. Con el primer componente calculado se continua con el cálculo de segundo componente. Al igual que el cálculo de primer componente se buscar entre los componentes restantes el componente que maximiza el resultado de la siguiente formula:

$$var(Y_2) = \sum_{k=1}^{p}\sum_{l=1}^{p} e_{2k}e_{2l}\sigma_{kl} = e_2'\Sigma e_2$$

El segundo componente computado además de la restricción impuesto al primer componente se debe tener en cuenta que sea incorrelada respecto al primer componente:

$$cov(Y_1, Y_2) = \sum_{k=1}^{p}\sum_{l=1}^{p} e_{1k}e_{2l}\sigma_{kl} = e_1'\Sigma e_2$$

7. Se repite el paso anterior para todas las variables del vector.

8. Por último, el porcentaje de la variabilidad de la reducción de dimensión se puede obtener de la siguiente forma:

$$\frac{\lambda_1 + \lambda_2 + \cdots + \lambda_k}{\lambda_1 + \lambda_2 + \cdots + \lambda_p}$$

Siendo $\lambda_i = var(Y_i)$

7.3.5.2.Descomposición de valores singulares (SVD)

La técnica de reducción de dimensión con SVD consiste en descomponer una matriz **X** de dimensión **m x n** en tres componentes: **$X = U * S * V^T$,** siendo **U** una matriz de dimensión **m x m**, **S** una matriz diagonal de dimensión **r x r (siendo r el valor mínimo de m y n)** donde los valores del diagonal son conocidos como valores singulares de **X** y **V^T** la matriz de dimensión **n x n**. [10]

Para conseguir el objetivo de la reducción de la dimensión, en lugar de quedarse con todos los valores singulares, se queda solo con los **K** valores más grandes, el cual representa la mayor variabilidad posible de la matriz original. Una vez seleccionado los **K** valores más relevantes se puede proceder con la reconstrucción de la matriz reducida mediante la siguiente formula:

$$X \sim U_k * S_k * V_k{}^T$$

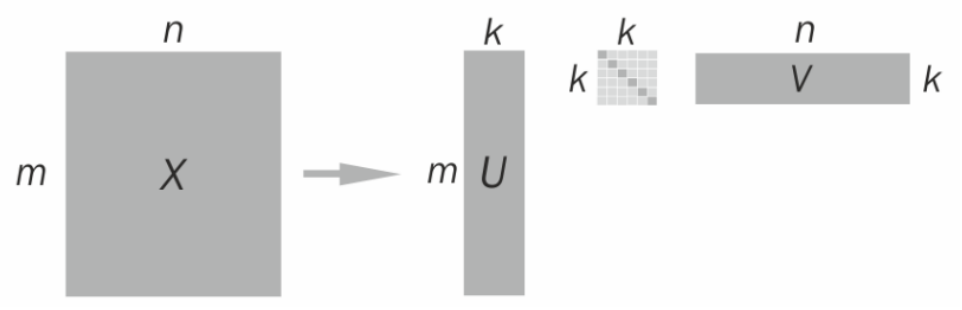

Figura 14:Diagrama de descomposición de valores singulares (SVD)- [10]

Volviendo ahora con el proyecto presente, en la biblioteca **MLlib** de **Spark** se puede encontrar las dos técnicas de reducción de dimensión: **análisis de componentes principales (PCA)** y **descomposición de valores singulares (SVD)**. No obstante, no es necesario incluir los dos modelos en el proyecto para la validación cruzada si se tiene en cuenta que el resultado obtenido de la reducción de dimensión mediante cualquiera de las técnicas indicadas anteriormente es idéntico en mayoría de los casos. Por cuál solo se incluirá PCA en el proyecto:

val pca = new PCA().setInputCol("features"). setOutputCol("pcaFeatures");

7.3.6. Validación cruzada

Existen diversas técnicas para realizar la validación de los modelos generados. Tales como Re-sustitución, Holdout, validación cruzada de K particiones, validación cruzada deja uno fuera etc. Todas ellas tienen como objetivos probar la efectividad del modelo generado y hallar los parámetros que maximizan el rendimiento de los algoritmos de predicción. [29]

- **Holdout:** La técnica de validación Holdout es la más sencilla de todas, simplemente divide el conjunto de datos en dos partes: una parte de datos para el entrenamiento del modelo y otra parte de datos para comprobar la efectividad del modelo generado. La utilización de esta técnica tiene la inconveniencia de que el resultado de la evaluación depende en gran medida de la división de los datos. Es decir, el resultado de la evaluación puede variar bruscamente dependiendo de la división realizada. Además, durante el entrenamiento no se puede aprovechar la totalidad de los datos disponibles para la generación del modelo.

- **Validación cruzada de K particiones (*k-fold cross-validation*):** Esta técnica se divide el conjunto de datos en K particiones y se repite las mismas operaciones de entrenamiento y comprobación K veces. En cada iteración se retiene una partición para realizar la comprobación y K-1 particiones para realizar el entrenamiento. Una vez terminada se calcular la media aritmética de la medida de la evaluación de todas iteraciones. La ventaja de esta técnica respecto al **Holdout** es que resulta menos relevante la división de los datos. Sin embargo, tiene la desventaja de que se necesita ejecutar K veces el algoritmo de entrenamiento. En otras palabras, el coste de la computación es K veces más elevado.

- **Validación cruzada dejan uno fuera (*Leave-one-out cross-validation*):** Esta técnica es equivalente a la validación cruzada de k iteraciones llevado a extremo. Consiste en realizar **n** iteraciones sobre un conjunto de datos con **n** instancias, dejando fuera una instancia fuera del conjunto de entrenamiento en cada iteración. Este tipo de evaluación es una muy buena opción para conjuntos de datos muy reducidos ya que el error generado es muy bajo. Sin embargo, el elevado coste de la computación lo hacer prácticamente inviable para conjuntos de datos de grandes volúmenes.

Considerando que la técnica Holdout es muy inestable en cuanto al resultado de la evaluación y la técnica de validación cruzada dejan uno fuera resulta demasiado costoso para el tamaño del conjunto de datos del proyecto; razón por la cual se ha decidido utilizar la validación cruzada de K iteración en el proyecto.

Una vez decido la técnica de la validación, ahora se debe determinar el número de iteraciones que se realizará. En mayoría de los casos se aplicar 10 iteraciones para cada validación. No obstante, para un conjunto de datos de gran tamaño la variación en el resultado del entrenamiento es mucho menor. De modo que no es necesario realizar demasiadas iteraciones para garantizar la estabilidad del resultado obtenido. Además, el tiempo de entrenamiento para un conjunto datos de gran tamaño es muy prolongado. En resumen, considerando el tamaño del conjunto de datos del proyecto se realizará solo 3 iteraciones.

Por otra parte, los parámetros que se incluyen en cada algoritmo para la validación son:

- **Parámetros generales para todos los algoritmos:**
    - **Tipos de normalización:** Para este proyecto se compara la efectividad de las técnicas de normalización **Escala Característica** y **Estandarización.**

- **PCA – K:** El número de dimensiones que se pretende reducir con el algoritmo PCA (39 - 42), excepto el caso de algoritmo GBT (40).

- **Regresión Lineal**
  - **Elastic Net:** Para hallar el valor de este parámetro que maximizar el rendimiento, se entrena el modelo para los valores de Elastic Net 0.0, 0.1, 0.5 y 1.0.
  - **Función de pérdida:** Se compara el rendimiento obtenido mediante la función de pérdida basada en el error cuadrático y la función de pérdida de Huber.
  - **Máxima iteración:** Se comprueba el resultado del modelo para los números máximos de iteraciones 10, 100 y 500.
  - **Parámetro de regularización:** Comprueba la efectividad del modelo para el parámetro de regularización con los valores 0.01, 0.001 y 0.0001.

```
val pca = new PCA().setInputCol("features").setOutputCol("pcaFeatures");

val lr = new LinearRegression().setFeaturesCol("pcaFeatures").setLabelCol("label").setLoss(lossFunction);

val pipeline = new Pipeline();

val pipeline_stages = Array[PipelineStage](categorizer, vectorAssembler, pca, lr);

val pipeline_grid = new ParamGridBuilder().baseOn(pipeline.stages -> pipeline_stages).addGrid(pca.k, pca_K).addGrid(lr.regParam, lr_regParam).addGrid(lr.elasticNetParam, lr_elasticNetParam).addGrid(lr.maxIter, lr_maxIter).build();

val regressionEvaluator = new RegressionEvaluator().setMetricName(evaluationMetric);
```

val cv = new CrossValidator().setEstimator(pipeline).setEvaluator(regressionEvaluator).setEstimatorParamMaps(pipeline_grid).setNumFolds(3);

val cvModel = cv.fit(data);

- **Regresión GBT**
  - **Máxima iteración:** Se fija el número de máxima iteración 10 a este algoritmo.
  - **Máxima profundidad:** Se valida la efectividad del algoritmo para las profundidades máximas 3, 5, 7 y 8.
  - **Tamaño de paso:** Se compara los resultados de modelos para los tamaños de paso 0.01 y 0.001.

val pca = new PCA().setInputCol("features").setOutputCol("pcaFeatures");

val gbt = new GBTRegressor().setFeaturesCol("pcaFeatures").setLabelCol("label");

val pipeline = new Pipeline();

val pipeline_stages = Array[PipelineStage](categorizer, vectorAssembler, pca, gbt);

val pipeline_grid = new ParamGridBuilder().baseOn(pipeline.stages -> pipeline_stages).addGrid(pca.k, pca_K).addGrid(gbt.maxIter, maxIter).addGrid(gbt.maxDepth, maxDepth).addGrid(gbt.stepSize, stepSize).build();

val regressionEvaluator = new RegressionEvaluator().setMetricName(evaluationMetric);

val cv = new CrossValidator().setEstimator(pipeline).setEvaluator(regressionEvaluator).setEstimatorParamMaps(pipeline_grid).setNumFolds(3);

- **Regresión con árboles aleatorios**
  - **Máxima profundidad:** Se comprueba el rendimiento de los modelos para las máximas profundidades 3, 4, 5, 6 y 8.
  - **Número de árboles:** Se compara la efectividad de los modelos para los números de árboles 20, 50 y 100.

val pca = new PCA().setInputCol("features").setOutputCol("pcaFeatures");

val rf = new RandomForestRegressor().setFeaturesCol("pcaFeatures").setLabelCol("label");

val pipeline = new Pipeline();

val pipeline_stages = Array[PipelineStage](categorizer, vectorAssembler, pca, rf);

val pipeline_grid = new ParamGridBuilder().baseOn(pipeline.stages -> pipeline_stages).addGrid(pca.k, pca_K).addGrid(rf.numTrees, numTrees).addGrid(rf.maxDepth, maxDepth).build();

val regressionEvaluator = new RegressionEvaluator().setMetricName(evaluationMetric);

val cv = new CrossValidator().setEstimator(pipeline).setEvaluator(regressionEvaluator).setEstimatorParamMaps(pipeline_grid).setNumFolds(3);

7.3.7. Evaluación de validación cruzada

Para poder evaluar la efectividad de los modelos predictivos obtenidos es necesario obtener la retroalimentación de las métricas de evaluación y a partir de ellas optimiza el algoritmo hasta hallar los parámetros que maximiza el rendimiento del mismo.

Existen muchas métricas para evaluar los modelos de predicción de los cuales se han incluido 4 métricas en la biblioteca de ML para evaluar los modelos de regresión.

- **Raíz de error cuadrático medio (RMSE)** [30]**:** o también conocido como raíz de desviación cuadrática media (RMSD), es la métrica más usada para realizar evaluaciones de modelos de predicción regresivo. RMSE se calcula el cuadrado del residuo o la diferencia que hay entre los valores previstos y los valores observados. En general se busca minimiza el valor RMSE. No obstante, en la práctica RMSE nunca podrá alcanzar un valor 0 ya que significaría un modelo de predicción perfecto.

$$RMSD = \sqrt{\frac{\sum_{t=1}^{T}(\hat{y} - y_t)^2}{T}}$$

- **Error cuadrático medio (MSE)** [31]**:** es un estimador que mide el promedio de los errores al cuadrado. En otras palabras, el cuadrado de las diferencias entre los valores previstos y los valores observados. Por lo tanto, MSE a igual que RMSE nunca puede tomar valores negativos y tampoco puede ser nulo.

$$MSE = \frac{1}{n}\sum_{i=1}^{n}\left(Y_i - \hat{Y}_i\right)^2$$

- **Coeficiente de determinación(R^2)** [32] [33]**:** el coeficiente de determinación indica la relación lineal que existen entre dos variables. En el caso de modelo predictivo concretamente se trata de calcular la relación lineal que hay entre la predicción y la observación. Esta métrica puede tomar valores comprendidos entre 0 y 1. El valor 0 indica que no existe ninguna relación entre la predicción y observación. Mientras, 1 indica un ajuste perfecto del modelo. No obstante, a pesar de que R^2 es una métrica no negativa R^2 puede tomar valores negativos y cuando esto sucede se considera R^2 como 0.

$$r = \frac{n(\sum xy) - (\sum x)(\sum y)}{\sqrt{[n\sum x^2 - (\sum x)^2][n\sum y^2 - (\sum y)^2]}}$$

Siendo r coeficiente de correlación y el coeficiente de determinación es el cuadrado de r.

- **Error medio absoluto (MAE):** el error medio absoluto consiste en calcular el promedio de la diferencia absoluta entre la predicción y la observación. Y se define con la siguiente formula:

$$MAE = \frac{\sum_{i=1}^{n} |y_i - x_i|}{n} = \frac{\sum_{i=1}^{n} |e_i|}{n}$$

R^2 tiene la ventaja de que sus valores están comprendidos entre 0 y 1, por lo tanto, su evaluación es muy intuitiva, mientras RMSE tiene la ventaja de que está expresado en las mismas unidades que la predicción. Además, de R^2 es una medida relativa, RMSE es una medida absoluta que aporta un valor más indicativo que R^2.

Por los motivos anteriores, en el proyecto actual se utilizará RMSE como la métrica de evaluación para la validación cruzada.

Por último, se detallará los resultados obtenidos de la validación cruzada:

- **Regresión Lineal**

Elastic Net	PCA-K	Regularización	Max Iter	Pérdida	Normalización	rmse
1	42	0.01	100	SE	Estandarización	164.90
0.5	42	0.01	10	SE	Estandarización	164.90
1	42	0.01	500	SE	Estandarización	164.90
1	41	0.01	10	SE	Estandarización	164.90

1	41	0.01	100	SE	Estandarización	164.90
1	41	0.01	500	SE	Estandarización	164.90
0.5	41	0.01	10	SE	Estandarización	164.90
0.5	41	0.01	100	SE	Estandarización	164.90
0.5	41	0.01	500	SE	Estandarización	164.90
0.5	42	0.01	500	SE	Estandarización	164.90
1	42	0.001	500	SE	Estandarización	164.90
0.1	41	0.01	10	SE	Estandarización	164.90
0.1	41	0.01	100	SE	Estandarización	164.90
0.1	41	0.01	500	SE	Estandarización	164.90
0	42	0.01	100	SE	Estandarización	164.90
0	42	0.01	500	SE	Estandarización	164.90
0	42	0.01	10	SE	Estandarización	164.90
0	42	0.01	500	SE	Escala Característica	164.90
0	42	0.01	100	SE	Escala Característica	164.90
0	41	0.01	10	SE	Escala Característica	164.90
0	41	0.01	100	SE	Escala Característica	164.90

0	41	0.01	500	SE	Escala Característica	164.90
0	41	0.01	10	SE	Estandarización	164.90
0	41	0.01	100	SE	Estandarización	164.90
0	41	0.01	500	SE	Estandarización	164.90
0.5	42	0.001	10	SE	Escala Característica	164.90
0	42	0.01	10	SE	Escala Característica	164.90
0.1	42	0.01	500	SE	Estandarización	164.90
0.5	42	0.001	100	SE	Estandarización	164.90
1	42	1.00E-04	10	SE	Escala Característica	164.90
0.5	42	0.01	100	SE	Estandarización	164.90
1	41	0.001	10	SE	Estandarización	164.90
1	41	0.001	100	SE	Estandarización	164.90
1	41	0.001	500	SE	Estandarización	164.90
0.5	42	0.001	500	SE	Estandarización	164.90
0.1	42	0.001	10	SE	Estandarización	164.90
0.1	42	0.01	10	SE	Escala Característica	164.90
0.5	41	0.001	10	SE	Estandarización	164.90

0.5	41	0.001	100	SE	Estandarización	164.90
0.5	41	0.001	500	SE	Estandarización	164.90
0.1	41	0.01	10	SE	Escala Característica	164.90
0.1	41	0.01	100	SE	Escala Característica	164.90
0.1	41	0.01	500	SE	Escala Característica	164.90
0.1	42	0.001	500	SE	Estandarización	164.90
1	42	1.00E-04	100	SE	Estandarización	164.90
0.1	42	0.01	500	SE	Escala Característica	164.90
0.1	42	0.001	10	SE	Escala Característica	164.90
0.1	42	0.001	100	SE	Estandarización	164.90
0.1	41	0.001	10	SE	Estandarización	164.90
0.1	41	0.001	100	SE	Estandarización	164.90
0.1	41	0.001	500	SE	Estandarización	164.90
1	42	1.00E-04	500	SE	Escala Característica	164.90
1	42	1.00E-04	10	SE	Estandarización	164.90
0	42	0.001	10	SE	Escala Característica	164.90
0	41	0.001	100	SE	Escala Característica	164.90

0	41	0.001	500	SE	Escala Característica	164.90
0	41	0.001	10	SE	Estandarización	164.90
0	41	0.001	100	SE	Estandarización	164.90
0	41	0.001	10	SE	Escala Característica	164.90
0	42	1.00E-04	100	Huber	Estandarización	199.17066
0	42	1.00E-04	100	Huber	Escala Característica	199.31200
0	42	1.00E-04	100	Huber	Escala Característica	199.31200
0	41	1.00E-04	100	Huber	Escala Característica	199.57300
0	41	1.00E-04	100	Huber	Estandarización	199.88691
0	41	1.00E-04	500	Huber	Estandarización	200.04291
0	42	1.00E-04	500	Huber	Escala Característica	200.05200
0	40	1.00E-04	100	Huber	Escala Característica	200.08900
0	41	1.00E-04	500	Huber	Escala Característica	200.08900
0	40	1.00E-04	500	Huber	Escala Característica	200.13400
0	40	1.00E-04	500	Huber	Estandarización	200.14611
0	40	1.00E-04	100	Huber	Estandarización	200.14645
0	42	1.00E-04	500	Huber	Estandarización	200.15807

0	39	1.00E-04	500	Huber	Escala Característica	200.39100
0	39	1.00E-04	500	Huber	Estandarización	200.40267
0	39	1.00E-04	100	Huber	Estandarización	200.40399
0	42	0.001	100	Huber	Escala Característica	200.54600
0	42	0.001	100	Huber	Estandarización	200.54632
0	41	0.001	100	Huber	Escala Característica	200.67500
0	39	1.00E-04	100	Huber	Escala Característica	200.69900

* Debido al tamaño de los datos solo se ha incluido los mejores resultados obtenidos a variar cada parámetro.

Se puede observar en los resultados anteriores que el parámetro que más influye en el modelo de la predicción es la función de pérdida que tiene casi un 20% más de RMSE utilizando la función de pérdida Huber en lugar de Error Cuadrado. Basándose en los resultados obtenidos de la validación cruzada se puede concluir que los parámetros que maximizan el rendimiento de algoritmo de regresión lineal son:

Elastic-Net → 1

PCA-K → 42

Regularización → 0.01

Máxima iteración → 100

Función de pérdida → Error Cuadrado

Tipo de normalización → RMSE

- **Regresión GBT**

PCA-K	Profundidad	Tamaño de Paso	Max Iter	Normalización	rmse
40	6	0.01	10	Escala Característica	162.66
40	8	0.01	10	Escala Característica	162.72
40	8	0.001	10	Escala Característica	163.06
40	6	0.001	10	Escala Característica	163.23
40	6	0.01	10	Estandarización	163.63
40	8	0.01	10	Estandarización	163.73
40	5	0.01	10	Escala Característica	164.07
40	8	0.001	10	Estandarización	164.09
40	6	0.001	10	Estandarización	164.19
40	5	0.01	10	Estandarización	164.66
40	5	0.001	10	Escala Característica	164.71
40	5	0.001	10	Estandarización	165.21
40	3	0.01	10	Escala Característica	169.72
40	3	0.01	10	Estandarización	169.73
40	3	0.001	10	Escala Característica	170.63

40	3	0.001	10	Estandarización	170.73

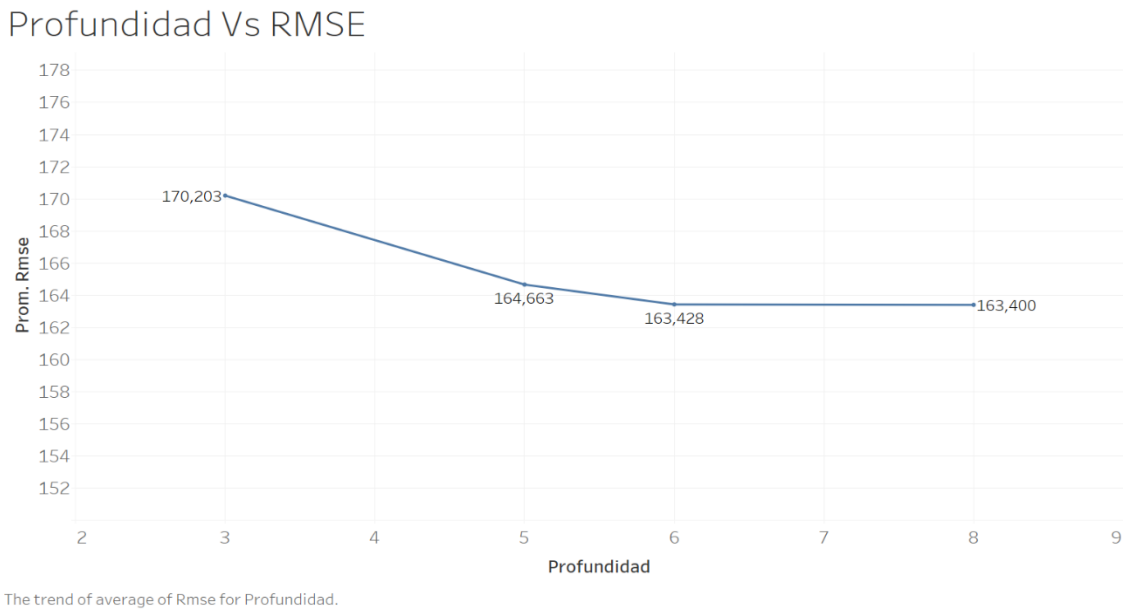

Figura 15: Profundidad vs RMSE - GBTR

En vista del resultado obtenido para el algoritmo regresión GBT, cabe resaltar que el parámetro que tiene mayor influencia sobre el modelo es la profundidad de árbol. En cuanto la configuración optima de los parámetros es la siguiente:

PCA-K → 40

Máxima profundidad → 6

Tamaño de paso → 0.01

Máxima iteración → 10

- **Regresión RF**

PCA -K	Nº Árboles	Profundidad	Normalización	rmse
41	100	8	Escala Característica	158.39
41	50	8	Escala Característica	158.40
40	100	8	Escala Característica	158.40

42	100	8	Escala Característica	158.40
40	50	8	Escala Característica	158.47
42	50	8	Escala Característica	158.50
41	100	8	Estandarización	158.51
41	50	8	Estandarización	158.52
40	100	8	Estandarización	158.53
42	100	8	Estandarización	158.53
40	50	8	Estandarización	158.56
39	100	8	Estandarización	158.57
42	50	8	Estandarización	158.57
39	50	8	Escala Característica	158.61
39	100	8	Escala Característica	158.62
39	50	8	Estandarización	158.72
42	20	8	Escala Característica	158.89
41	20	8	Escala Característica	158.90
40	20	8	Escala Característica	158.92
39	20	8	Escala Característica	158.95
39	20	8	Estandarización	159.05
42	20	8	Estandarización	159.07
40	20	8	Estandarización	159.10
41	20	8	Estandarización	159.12
40	100	6	Estandarización	160.84
41	100	6	Estandarización	160.85
41	50	6	Estandarización	160.85
40	50	6	Estandarización	160.85
42	100	6	Estandarización	160.86
42	50	6	Estandarización	160.97
39	100	6	Estandarización	160.98
40	100	6	Escala Característica	161.05
40	50	6	Escala Característica	161.08
39	50	6	Estandarización	161.09
41	100	6	Escala Característica	161.11
42	100	6	Escala Característica	161.11
41	50	6	Escala Característica	161.16
42	50	6	Escala Característica	161.23

41	20	6	Estandarización	161.24
39	100	6	Escala Característica	161.26
42	20	6	Estandarización	161.28
39	20	6	Estandarización	161.28
39	50	6	Escala Característica	161.30
40	20	6	Estandarización	161.31
41	20	6	Escala Característica	161.31
42	20	6	Escala Característica	161.33
40	20	6	Escala Característica	161.38
39	20	6	Escala Característica	161.57
40	100	5	Estandarización	162.59
40	50	5	Estandarización	162.59
41	100	5	Estandarización	162.63
41	50	5	Estandarización	162.63
42	100	5	Estandarización	162.72
42	50	5	Estandarización	162.76
39	100	5	Estandarización	162.77
39	50	5	Estandarización	162.83
41	20	5	Estandarización	162.89
39	20	5	Estandarización	162.91
40	50	5	Escala Característica	163.04
40	100	5	Escala Característica	163.06
42	20	5	Estandarización	163.07
40	20	5	Estandarización	163.09
41	100	5	Escala Característica	163.10
41	50	5	Escala Característica	163.13
42	100	5	Escala Característica	163.13
42	50	5	Escala Característica	163.17
40	20	5	Escala Característica	163.21
41	20	5	Escala Característica	163.22
42	20	5	Escala Característica	163.24
39	100	5	Escala Característica	163.26
39	50	5	Escala Característica	163.29
39	20	5	Escala Característica	163.62
40	100	4	Estandarización	164.99

42	50	4	Estandarización	165.02
40	50	4	Estandarización	165.03
41	50	4	Estandarización	165.06
41	100	4	Estandarización	165.07
42	100	4	Estandarización	165.12
39	50	4	Estandarización	165.21
39	100	4	Estandarización	165.23
39	20	4	Estandarización	165.38
41	20	4	Estandarización	165.43
40	20	4	Estandarización	165.53
42	20	4	Escala Característica	165.57
42	20	4	Estandarización	165.61
41	20	4	Escala Característica	165.61
40	100	4	Escala Característica	165.67
40	20	4	Escala Característica	165.68
41	100	4	Escala Característica	165.70
40	50	4	Escala Característica	165.70
42	100	4	Escala Característica	165.76
41	50	4	Escala Característica	165.83
39	100	4	Escala Característica	165.85
42	50	4	Escala Característica	165.89
39	20	4	Escala Característica	165.93
39	50	4	Escala Característica	165.99
40	50	3	Estandarización	168.18
41	50	3	Estandarización	168.24
40	100	3	Estandarización	168.34
39	50	3	Estandarización	168.35
41	100	3	Estandarización	168.39
42	100	3	Estandarización	168.41
42	50	3	Estandarización	168.49
42	20	3	Estandarización	168.50
41	20	3	Estandarización	168.50
39	100	3	Estandarización	168.52
40	20	3	Estandarización	168.62
39	20	3	Estandarización	168.68

40	20	3	Escala Característica	168.78
41	20	3	Escala Característica	168.79
42	20	3	Escala Característica	168.83
40	50	3	Escala Característica	169.00
40	100	3	Escala Característica	169.02
39	20	3	Escala Característica	169.02
41	100	3	Escala Característica	169.04
42	100	3	Escala Característica	169.05
41	50	3	Escala Característica	169.15
39	100	3	Escala Característica	169.16
39	50	3	Escala Característica	169.20
42	50	3	Escala Característica	169.20

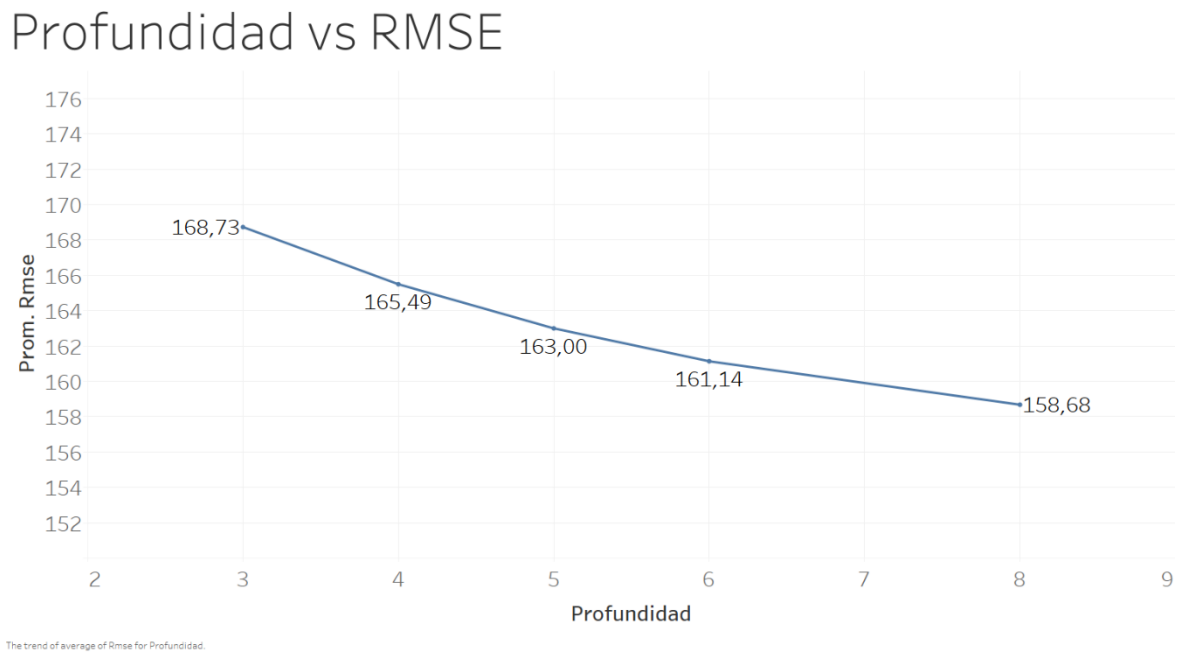

Figura 16: Profundidad vs RMSE (RF)

En el algoritmo de regresión RF, al igual que el algoritmo de regresión GBT, el parámetro que más influye en la efectividad del modelo es la profundidad del árbol. Como conclusión del resultado obtenido, la mejor configuración de los parámetros para el algoritmo de regresión RF es:

PCA-K → 41

Nº Árbol → 100

Profundidad → 8

Tipo de normalización → Escala Característica

7.4. PRUEBAS REALIZADAS

En el apartado anterior se ha podido identificar los parámetros que maximizan el rendimiento de los algoritmos predictivos. En este apartado se usarán estos modelos generados para calcular los valores futuros del D_SEG_SINFÍN.

Para poder comprobar la efectividad de las predicciones de calculadas se usarán los siguientes parámetros:

- **Raíz de error cuadrático medio (RMSE):** Como ya se ha explicado en el capítulo anterior, este parámetro sirve para calcular la diferencia relativa entre los valores previsto y los valores observados.

- **Coeficiente de determinación (R^2):** Este parámetro describe la relación lineal que hay entre la predicción y la observación.

LRSE metrics:

RMSE: 177.001

R2: 0.1

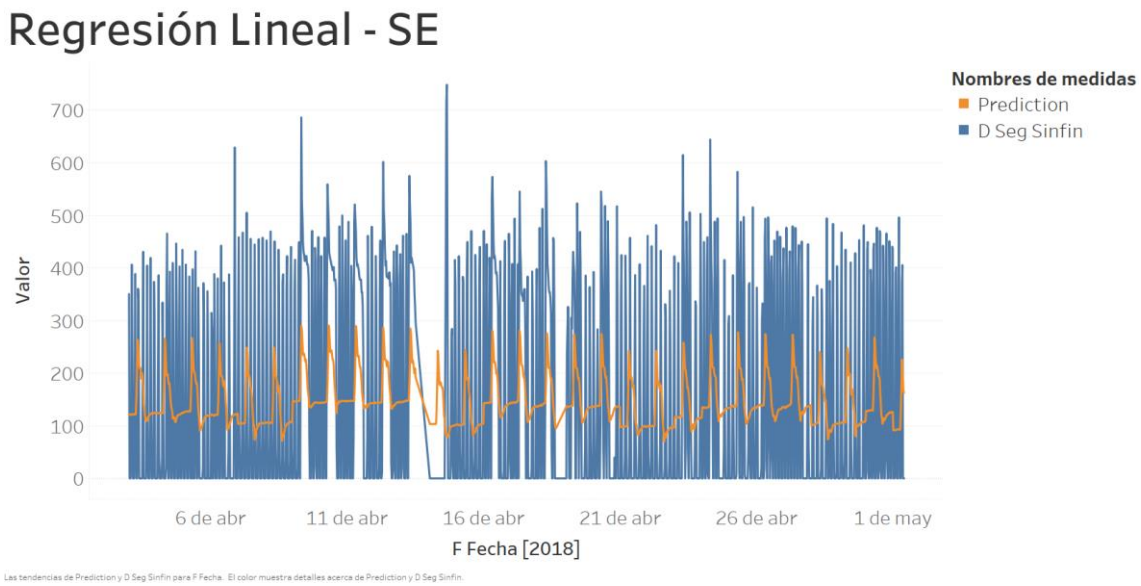

Gráfico 23: Regresión Lineal – Se (Predicciones en intervalos de media hora)

GBTR metrics:

RMSE: 175.63

R2: 0.114

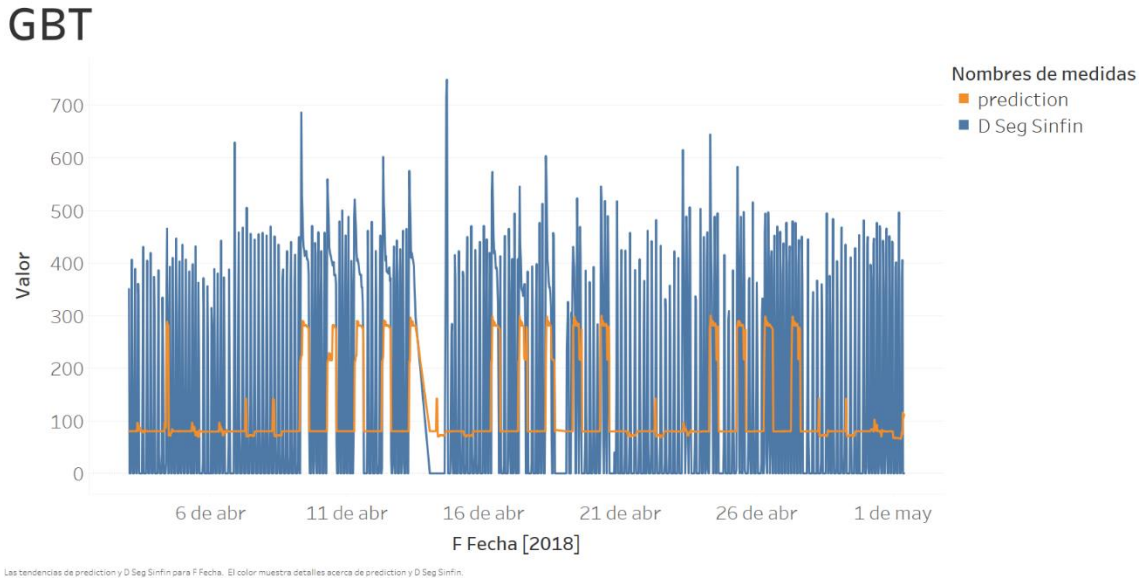

Gráfico 24: GBT (Predicciones en intervalos de media hora)

LRHuber metrics:

RMSE: 230.725

R2: -0.529

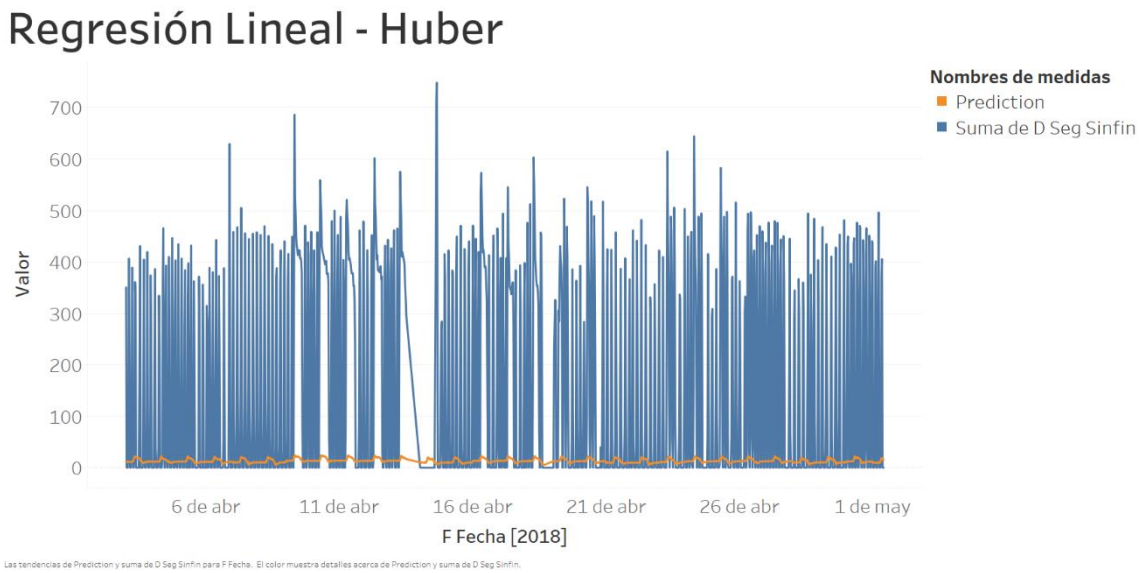

Gráfico 25: Regresión Lineal – Huber (Predicciones en intervalos de media hora)

RFR metrics:

RMSE: 174.654

R2: 0.124

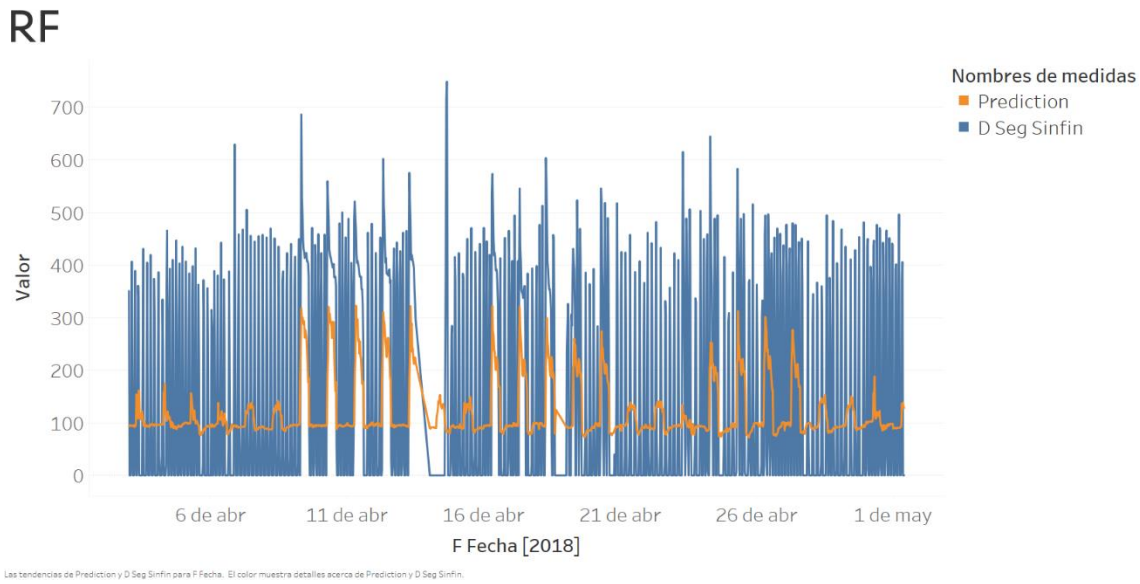

Gráfico 26: Regresión con bosque aleatorio (Predicciones en intervalos de media hora)

Se puede apreciar, tanto a través de las métricas, como a través de la gráfica comparativa, que existe una diferencia excesiva entre la predicción y el valor observado para todos los algoritmos.

El algoritmo de regresión con bosque aleatorio (RFR) es el algoritmo que ha obtenido el mejor resultado. Mientras el algoritmo de regresión lineal con función de pérdida Huber (LRHuber) es con diferencia es el que ha obtenido el peor resultado.

Basándose en los resultados anteriores, se puede llegar a la conclusión de que el modelo generado no se puede utilizar para predecir los valores futuros del D_SEG_SINFIN (consumo) con precisión a intervalos de cada media hora. Afortunadamente tampoco requiere esa precisión para estimular el momento para reponer los combustibles.

LRSE metrics:

RMSE: 1695.741

R2: 0.476

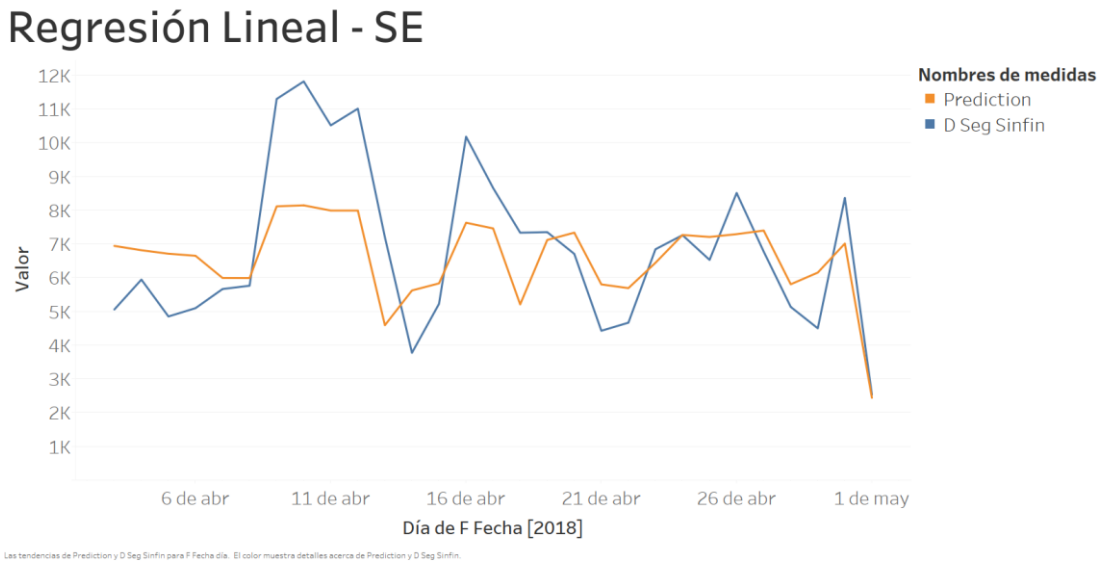

Gráfico 27: Regresión Lineal – SE (Predicciones diarias)

GBTR metrics:

RMSE: 2323.227

R2: 0.016

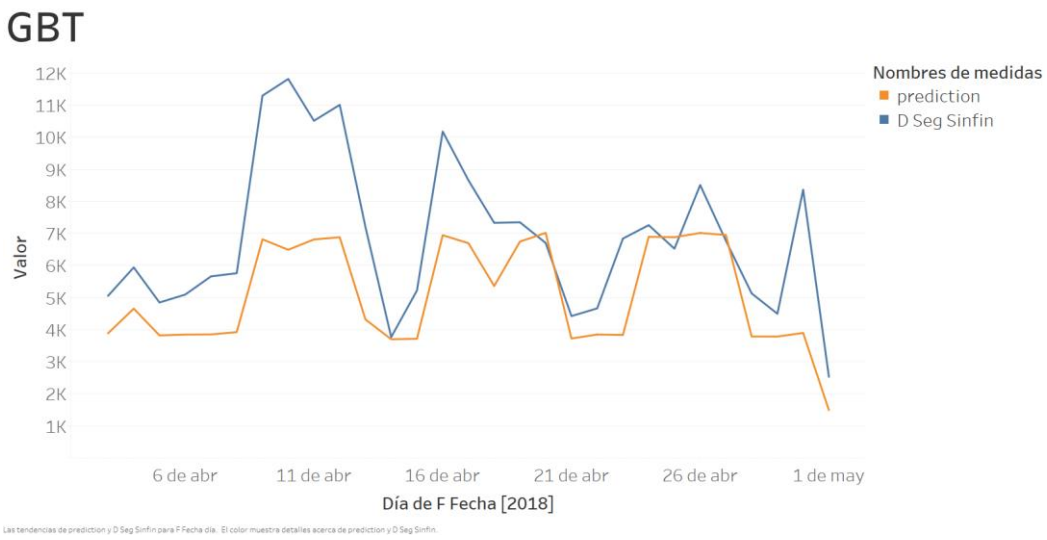

Gráfico 28: GBTR (Predicciones diarias)

LRHuber metrics:

RMSE: 6658.453

R2: -7.086

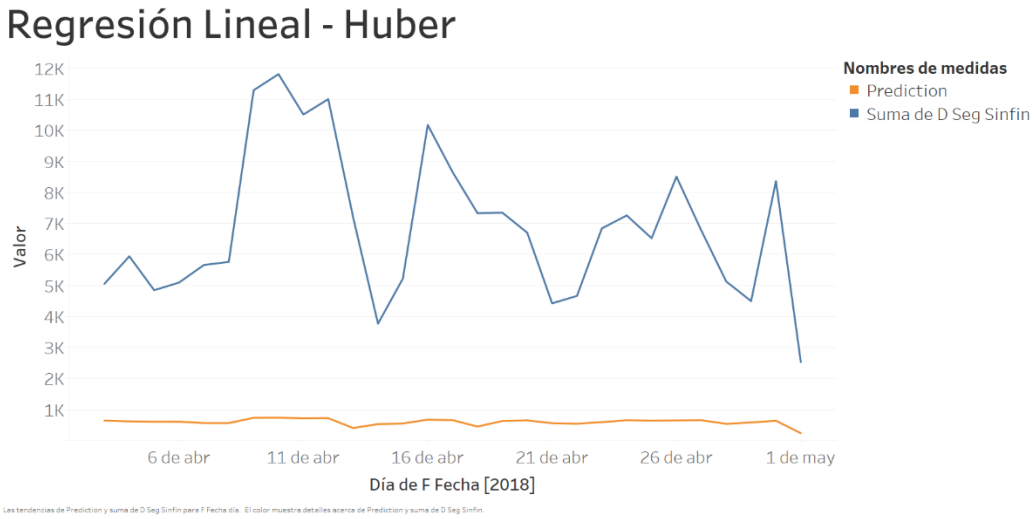

Gráfico 29: Regresión Lineal – Huber (Predicciones diarias)

RFR metrics:

RMSE: 2133.675

R2: 0.17

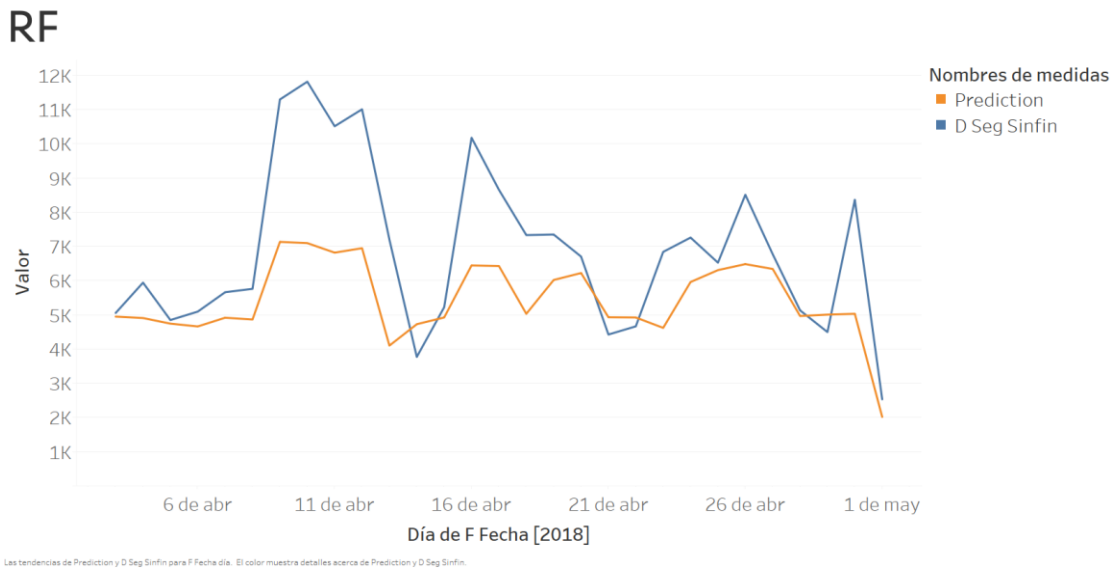

Gráfico 30: Regresión con bosque aleatorio (Predicciones diarias)

Como se puede observar en este caso, las predicciones se ajustan bastante mejor a los valores observados de forma diaria. A excepción de GBTR y LRHuber que se ven empeorados. No obstante, si consideramos que el objetivo de estas predicciones de consumo tiene como objetivo determinar el tiempo de recargar de combustible que suelen realizar con la frecuencia de 2 veces al mes. Por lo tanto, incluso se puede ampliar el rango de la predicción para que el resultado obtenido sea ajustado mejor al valor observado. En este caso se comprobará los resultados obtenidos en intervalos de semanas.

LRSE metrics:

RMSE: 6608.116

R2: 0.85

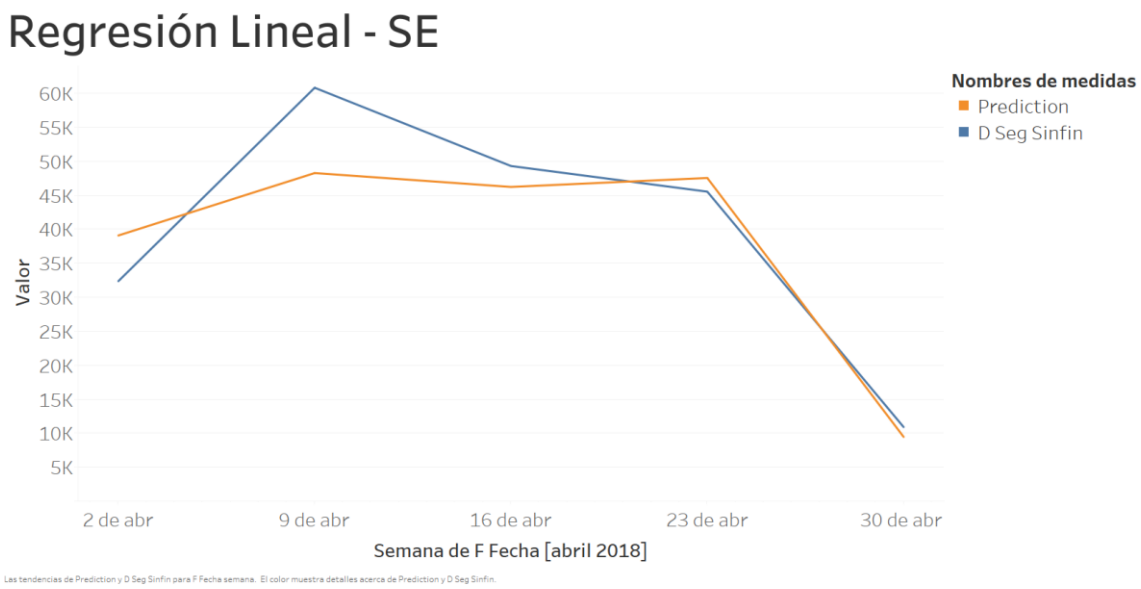

Gráfico 31: Regresión Lineal – SE (Predicciones semanales)

GBTR metrics:

RMSE: 11904.486

R2: 0.513

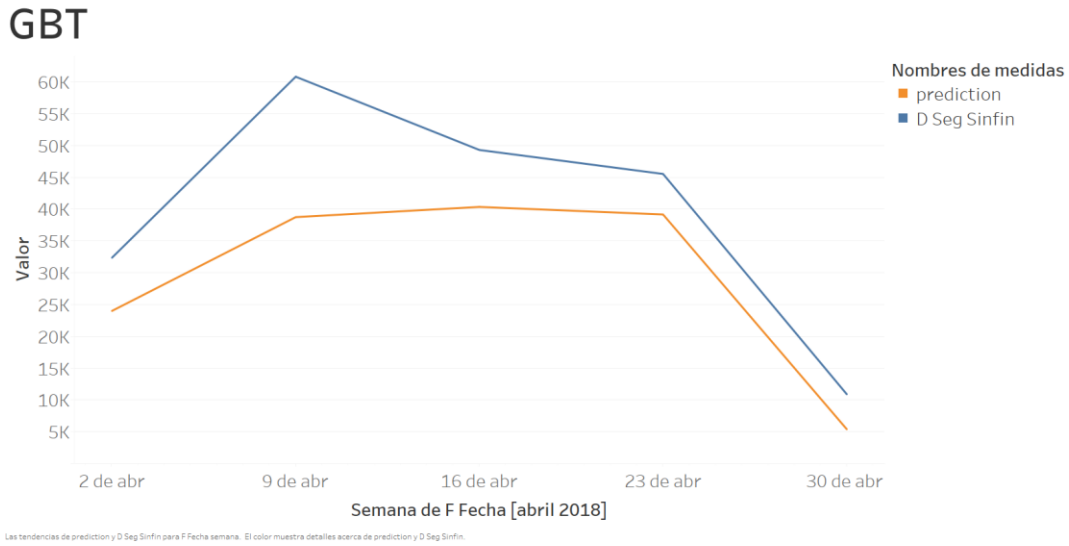

Gráfico 32: GBTR (Predicciones semanales)

LRHuber metrics:

RMSE: 39604.87

R2: -4.387

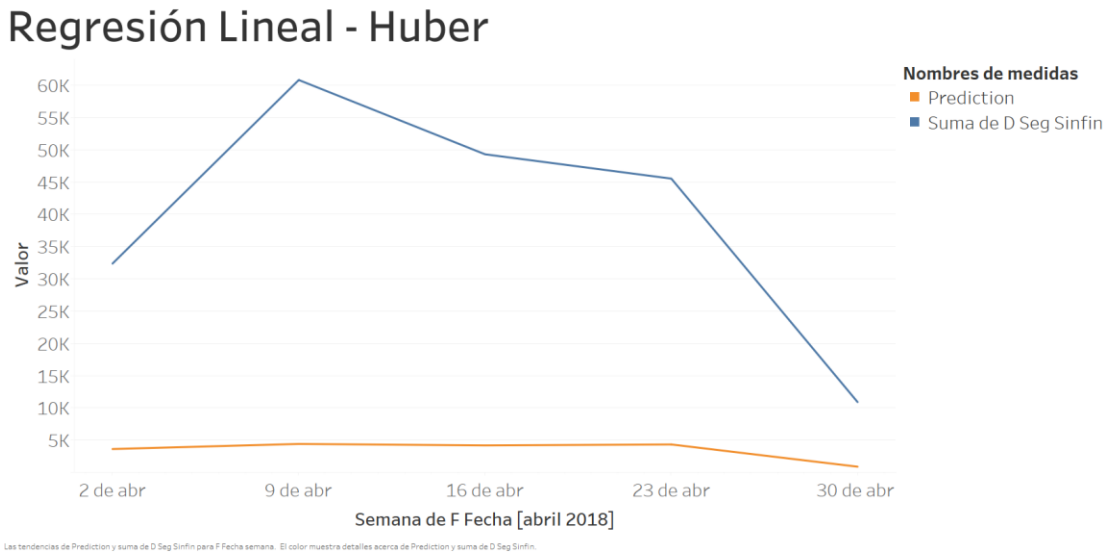

Gráfico 33: Regresión Lineal – Huber (Predicciones semanales)

RFR metrics:

RMSE: 10102.49

R2: 0.649

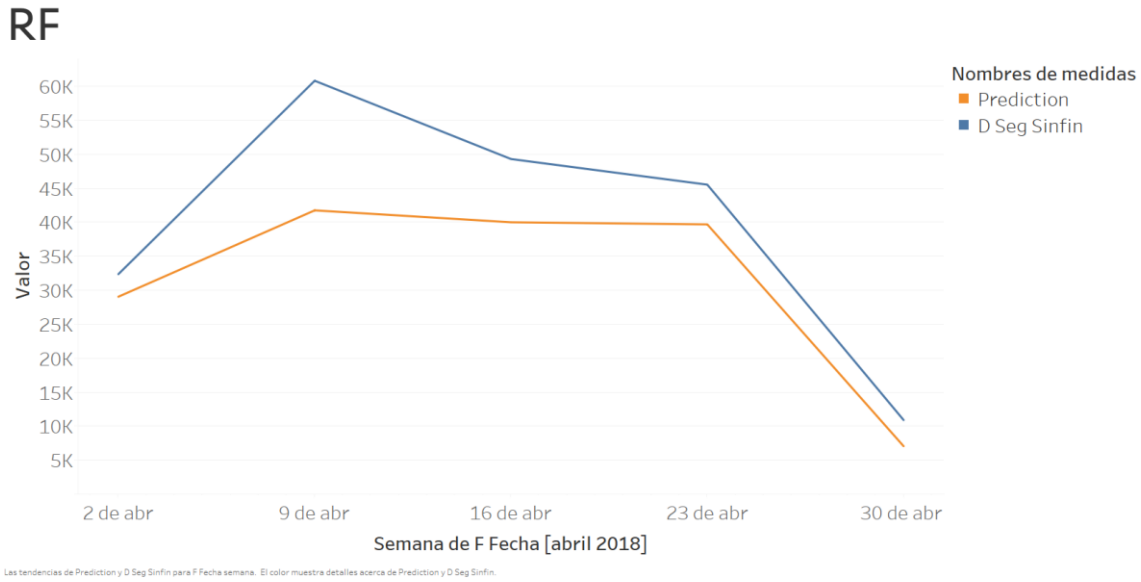

Gráfico 34: Regresión con bosque aleatorio (Predicciones semanales)

Respecto a las predicciones semanales, se puede observar una mejora en todos los algoritmos, aunque el algoritmo de regresión lineal con Huber sigue ofreciendo las peores predicciones y sin ninguna correlación lineal entre las predicciones y los valores observados. Mientras el algoritmo LRSE se ha convertido en el algoritmo que proporciona las mejores predicciones, teniendo casi solo la mitad de RMSE respecto a los otros dos algoritmos restante y un valor de R^2 de 0.85.

En resumen, los modelos generados no son aptos para realizar predicciones con una precisión de cada media hora. Sin embargo, se puede utilizar para estimular el tiempo aproximado de recargas de combustibles en cada instalación, así como la cantidad requerida.

8. CONCLUSIONES

El presente Trabajo Fin de Máster tiene como objetivo extracción de informaciones relevantes para optimizar el consumo y el control preventivo de los sistemas de climatización, a partir del análisis y tratamiento de los datos registrados.

El proyecto completo se puede dividir en dos partes, donde la primera parte se ha concluido con la finalización de implementación y diseño de los paneles de mando. Para ellos en primer lugar se ha cargado el conjunto de datos en el servidor de almacenamiento de Amazon donde es tratado para extraer los valores de los indicadores de cada panel. Después los valores generados de los indicadores son cargados al Tableau para ser visualizados de forma gráfico.

Respecto el desarrollo de la segunda parte de proyecto, al igual que la primera parte comienza con la carga de los datos en los servidores de Amazon y desde allí son analizados para determinar las propiedades de cada parámetro involucrado. A continuación, son procesados con fin de dotarles la estructura requerida para el entrenamiento de los modelos predictivos. Durante el proceso de entrenamiento los modelos son sometidos a la validación cruzada para hallar la mejor configuración del algoritmo. Por último, para da por finalizado esta parte del proyecto los modelos son verificados mediante una serie de pruebas donde se detallan la efectividad de cada uno.

Una vez resumido los objetivos conseguidos en el proyecto, examinaremos brevemente ahora las líneas de trabajo futuros que consta de cuatros metas:

- **Streaming:** con miras a la automatización completa de sistema, es una propuesta interesante incluir el streaming en el proyecto para que de esta forma los nuevos datos generados sean ingerido de forma automática y los modelos predictivos son entrenados y mejorados constantemente con la inclusión de los nuevos datos.

- **Modelo predictivo genérico:** el modelo genérico aprovechará todos los datos registrados para generar el modelo predictivo, que se supone será apto para realizar las predicciones de cualquier instalación. Obviamente para que esto sea viable se deberá incluir las características de cada instalación en el entrenamiento del modelo. Con este modelo se pretenderá realizar predicciones para aquellas instalaciones nuevas que no dispondrán de registros históricos.

- **Prevención de fallos:** en lugar de realizar predicciones enfocados al consumo de combustibles y facilitar la logística de suministro, se puede crear modelos predictivos para detectar los problemas potenciales de la caldera y de esta forma anticipar y evitar a los eventos indeseables.

- **Determinar el conjunto de parámetros que optimiza el rendimiento del sistema:** el mismo proceso que se ha utilizado para crear el sistema predictivo de consumo se puede utilizar para otros parámetros de sistema para identificar los errores potenciales e incluso halla el patrón que optimiza el funcionamiento.

Bibliografía

[1] M. Á. Laguna, «Universidad de Valladolid - El método de desarrollo, El proceso unificado,» [En línea]. Available: https://www.infor.uva.es/~mlaguna/cd/cd6. [Último acceso: 05 05 2018].

[2] Wikipedia, «Wikipedia - Proceso Unificado,» 29 05 2016. [En línea]. Available: https://es.wikipedia.org/wiki/Proceso_unificado. [Último acceso: 05 05 2018].

[3] D. L. F. R. Pablo, «PGP Gestión de riesgos,» 2016.

[4] Amazon, «Amazon - EMR,» 2018. [En línea]. Available: https://aws.amazon.com/es/emr/. [Último acceso: 05 07 2018].

[5] Apache Spark, «Apache Spark - Graphx,» 08 06 2018. [En línea]. Available: https://spark.apache.org/docs/2.3.1/graphx-programming-guide.html. [Último acceso: 05 07 2018].

[6] Apache Spark, «Apache Spark - Streaming,» 08 06 2018. [En línea]. Available: https://spark.apache.org/docs/2.3.1/streaming-programming-guide.html. [Último acceso: 05 07 2018].

[7] Hortonworks, «Hortonworks - Apache Spark,» [En línea]. Available: https://es.hortonworks.com/apache/spark/. [Último acceso: 05 07 2018].

[8] Apache Spark, «Apache Spark - MLlib,» 28 02 2018. [En línea]. Available: https://people.apache.org/~pwendell/spark-nightly/spark-master-doc/lastest/ml-guide.html. [Último acceso: 05 07 2018].

[9] Amazon, «Amazon - S3,» 2018. [En línea]. Available: https://aws.amazon.com/es/s3/faqs/#. [Último acceso: 16 07 2018].

[10] R. Dua, M. S. Ghotra y N. Pentreath, «Machine Learning with Spark [Seconde Edition],» 2017.

[11] Apache Spark, «Apache Spark - LinearRegression,» 08 02 2018. [En línea]. Available: https://spark.apache.org/docs/2.3.0/api/java/index.html?org/apache/spark/ml/regression/LinearRegression.html. [Último acceso: 29 06 2018].

[12] Wikipedia, «Wikipedia - Elastic net regularization,» 15 06 2018. [En línea]. Available: https://en.wikipedia.org/wiki/Elastic_net_regularization. [Último acceso: 01 07 2018].

[13] Apache Spark, «Apache Spark - RandomForestRegresor,» 11 07 2017. [En línea]. Available:

https://spark.apache.org/docs/2.2.0/api/scala/index.html#org.apache.spark.ml.regression.RandomForestRegressor. [Último acceso: 05 07 2018].

[14] M. Nedrich, «Atomic Object - An introduction to gradient descent and linear regression,» 24 06 2014. [En línea]. Available: https://spin.atomicobject.com/2014/06/24/gradient-descent-linear-regression/. [Último acceso: 16 07 2018].

[15] L. Columbus, «SelectHub - Tableau vs QlikView vs Microsoft Power BI,» 2016. [En línea]. Available: https://selecthub.com/business-intelligence/tableau-vs-qlikview-vs-microsoftpower-bi/. [Último acceso: 29 05 2018].

[16] A. G. Carlos y P. J. Belarmino, Técnicas escalables de análisis de datos - Tema 2: Metodologías análisis big data, 2017-18.

[17] A. G. Carlos y P. J. Belarmino, Técnicas escalables de análisis de datos - Tema 3: Aprendizaje automático, 2017-18.

[18] Junta de Castilla y León, «JCYL Datos Abiertos,» 2018. [En línea]. Available: https://datosabiertos.jcyl.es/web/jcyl/RISP/es/Plantilla66y33/1284162055979/_/ _/_. [Último acceso: 02 04 2018].

[19] NIST/SEMATECH , «e-Handbook of Statistical Methods,» 30 10 2013. [En línea]. Available: https://www.itl.nist.gov/div898/handbook/eda/section3/eda35h.htm. [Último acceso: 25 06 2018].

[20] S. Glen, «Statistics How To - Chauvenet's Criterion,» 12 10 2017. [En línea]. Available: www.statistichowto.com/chauvenets-criterion. [Último acceso: 16 07 2018].

[21] B. Reese, «Outliers and Structural Change,» 2016. [En línea]. Available: http://breese.github.io/2016/08/07/outliers-and-structural-change.html. [Último acceso: 16 07 2018].

[22] S. Raschka, «About Feature Scaling and Normalization,» 11 07 2014. [En línea]. Available: http://sebastianraschka.com/Articles/2014_about_feature_scaling.html. [Último acceso: 29 06 2018].

[23] Y. LeCun, L. Botton, G. B. Orr y K.-R. Müller, «Efficient BackProp,» 1998. [En línea]. Available: http://yann.lecun.com/exdb/publis/pdf/lecun-98b.pdf. [Último acceso: 28 06 2018].

[24] S. Morante, «Synergic Partners - Warnings about normaling data,» 17 09 2017. [En línea]. Available: http://www.synergicpartners.com/en/espanol-precauciones-a-la-hora-denormalizar-datos-en-data-science/. [Último acceso: 29 06 2018].

[25] S. Raschka, «Implementing a Principal Component Analysis (PCA),» 13 04 2014. [En línea]. Available: http://sebastianraschka.com/Articles/2014_pca_step_by_step.html. [Último acceso: 16 07 2018].

[26] Wikipedia, «Wikpedia - Imagen Brújula,» 13 06 2016. [En línea]. Available: https://fr.wikipedia.org/wiki/Graduation_de_la_rose_des_vents. [Último acceso: 26 05 2018].

[27] The Pennsylvania State University, «Penn State SCIENCE,» 2018. [En línea]. Available: https://onlinecourses.science.psu.edu/stat505/node/49/. [Último acceso: 16 07 2018].

[28] J. M. M. Diazaraque, «Análisis de Componentes Principales,» [En línea]. Available: http://halweb.uc3m.es/esp/Personal/personas/jmmarin/esp/AMult/tema3am.pdf. [Último acceso: 16 07 2018].

[29] J. Schneider, «Carnegie Mellon University School of Computer Science - Cross validation,» 07 02 1997. [En línea]. Available: https://www.cs.cmu.edu/~schneide/tut5/node42.html. [Último acceso: 16 07 2018].

[30] Wikipedia, «Wikipedia - Root Mean Square Deviation,» 10 07 2018. [En línea]. Available: https://en.wikipedia.org/wiki/Root-mean-square_deviation. [Último acceso: 12 07 2018].

[31] Wikipedia, «Wikipedia - Mean Squared error,» 15 06 2018. [En línea]. Available: https://en.wikipedia.org/wiki/Mean_squared_error. [Último acceso: 12 07 2018].

[32] Wikipedia, «Wikipedia - Coefficient of determination,» 10 07 2018. [En línea]. Available: https://en.wikipedia.org/wiki/Coefficient_of_determination. [Último acceso: 12 07 2018].

[33] S. Glen, «Statistic How To - Coefficient of determination r squared,» 21 05 2018. [En línea]. Available: http://www.statisticshowto.com/probability-and-statistics/coefficient-ofdetermination-r-squared/. [Último acceso: 12 07 2018].

[34] Chapman y Hall/CRC, de Statistical Regression and Classification: From Lineal Models to Machine Learning , 2017.

[35] F. Cheung, «GitHub-SparkPercentile,» 22 02 2015. [En línea]. Available: https://gist.github.com/felixcheung/92ae74bc349ea83a9e29. [Último acceso: 15 06 2018].

[36] N. M. Razali y Y. B. Wah, «WayBack Machine - Power Comparisons of Shapiro-Wilk, Kolmogorov-Smirnov, Lilliefors and Anderson-Darling tests,» 2011. [En línea]. Available: https://web.archive.org/web/20150630110326/http://instatmy.org.my/downloads/ejurnal%202/3.pdf. [Último acceso: 29 06 2018].

[37] Apache Spark, «Apache Spark - ML Classification Regression,» 11 07 2017. [En línea]. Available: https://spark.apache.org/docs/2.2.0/ml-classification-regression.html. [Último acceso: 22 06 2018].

[38] Apache Spark, «Apache Spark - ML Turning,» 11 07 2017. [En línea]. Available: https://spark.apache.org/docs/2.2.0/ml-tuning.html. [Último acceso: 03 07 2018].

[39] B. Cutler, «Spark Cross Validation with Multiple Pipelines,» 18 08 2017. [En línea]. Available: https://bryancutler.github.io/cv-pipelines/. [Último acceso: 02 07 2018].

[40] D. Becker, «Kaggle - Cross Validation,» 01 2018. [En línea]. Available: https://www.kaggle.com/dansbecker/cross-validation. [Último acceso: 05 07 2018].

[41] Wikipedia, «Wikipedia - Categorical variable,» 26 06 2018. [En línea]. Available: https://en.wikipedia.org/wiki/Categorical_variable. [Último acceso: 05 07 2018].

[42] P. Gupta, «Towards Data Science,» 05 06 2017. [En línea]. Available: https://towardsdatascience.com/cross-validation-in-machine-learning72924a69872f. [Último acceso: 16 07 2018].

[43] Wikipedia, «Wikipedia - Mean Absolute Error,» 04 05 2018. [En línea]. Available: https://en.wikipedia.org/wiki/Mean_absolute_error. [Último acceso: 12 07 2018].

[44] Turi, «Turi - Random Forest Regression,» [En línea]. Available: https://turi.com/learn/userguide/supervisedlearning/random_forest_regression.html . [Último acceso: 02 07 2018].

[45] M. S. Q. Isaac y V. P. Carlos, BIG DATA: INTELIGENCIA DE NEGOCIOS, 2017-18.