Universidad de Cantabria

Facultad de Ciencias

APLICACIÓN DE TÉCNICAS DE CIENCIA DE DATOS EN EL SECTOR LOGÍSTICO

(Application of Data Science techniques in the logistics sector)

Trabajo de Fin de Grado
para acceder al

GRADO EN INGENIERÍA INFORMÁTICA

Autor: Miguel de la Cal García

Directora: Cristina Tirnauca

Co-Directora: Marta Elena Zorrilla Pantaleón

09 - 2024

Resumen

La electrónica embarcada está aportando a las empresas y usuarios la posibilidad de disponer de múltiples fuentes de información sobre el uso de sus vehículos y la gestión de los mismos por los conductores. El presente proyecto se dirige a extraer patrones descriptivos y predictivos de la actividad logística de una empresa cántabra, con objeto de ofrecer servicios optimizados y centrados en el dato que le faciliten la refactorización de los procesos actuales, así como el establecimiento de otros procesos nuevos basados en la calidad, consistencia, integridad y el conocimiento extraído de las fuentes de información disponibles.

En concreto, se trabajará con datos procedentes de la gestión y planificación de rutas para atender a los pedidos de distintos clientes de una empresa logística. El objetivo es extraer el conocimiento de las decisiones tomadas previamente por los controladores de tráfico. Estas personas se encargan de asignar ruta(s), conductor(es) y vehículo(s) una vez recibida la notificación de entregar un pedido. A partir de esta información se espera construir en el futuro un recomendador que apoye a la torre de control. El proyecto se desarrollará siguiendo las fases establecidas en la metodología CRISP-DM.

Palabras clave Ciencia de datos, Inferencia de datos, Planificación de rutas, Logística

Abstract

Embedded electronics are providing companies and users with the possibility of accessing multiple pieces of information regarding the use of their vehicles and their management by drivers. This current project aims to extract descriptive and predictive patterns from the logistics activity of a company based in Cantabria. The goal is to offer optimized services centered on data to facilitate the refactoring of current processes, as well as the establishment of new processes based on quality, consistency, integrity, and the knowledge extracted from available information sources.

Specifically, we will work with data from the management and planning of routes to serve orders from various clients. The objective is to extract knowledge from the decisions previously made by traffic controllers. These individuals are responsible for, once the notification of a delivery is given, assigning the route or routes to be taken, as well as the driver or drivers and the vehicles to be used. Once that information is known, it is possible to construct a future recommender system that supports the control tower. The project will be developed following the phases established in the CRISP-DM methodology.

Keywords Data Science, Data Inference, Route Planning, Logistics

Índice

Índice de figuras

Índice de tablas

1. Contexto y objetivo

Gracias a la electrónica embarcada, diversas empresas están aprovechando la oportunidad de obtener información variada en tiempo real sobre sus vehículos, como posición por GPS y medidas de rendimiento, y sus conductores, incluyendo, entre otras, medidas de calidad de conducción y desempeño de las tareas. La correcta gestión de estos datos y su posterior análisis resulta crucial a la hora de mejorar o automatizar la planificación de rutas [1], ya que permite trabajar con mayor eficacia y rapidez, así como un ahorro en costes, por ejemplo, por reducción de desplazamientos innecesarios.

Este trabajo de fin de grado se enmarca en el contexto del proyecto titulado "Investigación Industrial en modelado de datos e implementación de técnicas inteligentes en el sector logístico", desarrollado junto con la empresa Fagor Electrónica S. Coop, concretamente la subdivisión de Fagor Smart Data Services (SDS). Además, el trabajo está realizado en el marco de una beca de colaboración con el departamento de Ingeniería Informática y Electrónica de la Universidad de Cantabria.

Fagor Electrónica S. Coop es una cooperativa española fundada en 1966 y parte del Grupo Mondragón. Algunos de los productos que ofrecen son semiconductores, sistemas de comunicación o soluciones de automatización. En concreto, SDS fue fundada en Santander en el año 2000 y su actividad se centra en el desarrollo y venta de servicios digitales para la geolocalización en los sectores de transporte y logística. SDS desarrolló FlotasNet, un software de gestión de flotas que permite tener un control de la asignación de vehículos, conocimiento en tiempo real de la situación de estos, asignación de rutas, etc. Fagor vende este servicio a clientes (empresas de transporte) con necesidades de optimizar su apartado logístico. Recientemente se han ampliado las funcionalidades de este sistema para satisfacer las necesidades de un nuevo cliente, la empresa especializada en transporte exprés ACME. Esta cuenta con una flota de 200 tractoras y 240 semirremolques y prestan un servicio de máxima calidad, adaptado a las necesidades particulares de cada cliente. Las nuevas funcionalidades permiten a la empresa gestionar también, desde la propia aplicación, información relacionada con los pedidos. La información que se carga en la aplicación para hacer esta gestión proviene del sistema ERP(*Enterprise Resource Planning*) de la empresa, que es un sistema de información que optimiza las tareas e incluso su automatización al centralizar en una base de datos la información referente a todas las actividades, haciendo una división en módulos como recursos humanos, finanzas, gestión de relaciones con clientes, etc.

Para entender mejor cómo se comunica el software de FlotasNet con el resto de sistemas, se hace una descripción gráfica, como se puede observar en la Figura 1. Cuando la compañía de logística cliente contrata con SDS los servicios de esta plataforma, se embarcan equipos en los transportistas de la compañía de logística, se envían datos relacionados con el transporte y los vehículos desde los equipos embarcados y se almacenan en una base de datos en SDS. La compañía logística cliente accede a esta plataforma para gestionar su flota y SDS realiza las actividades de mantenimiento y soporte.

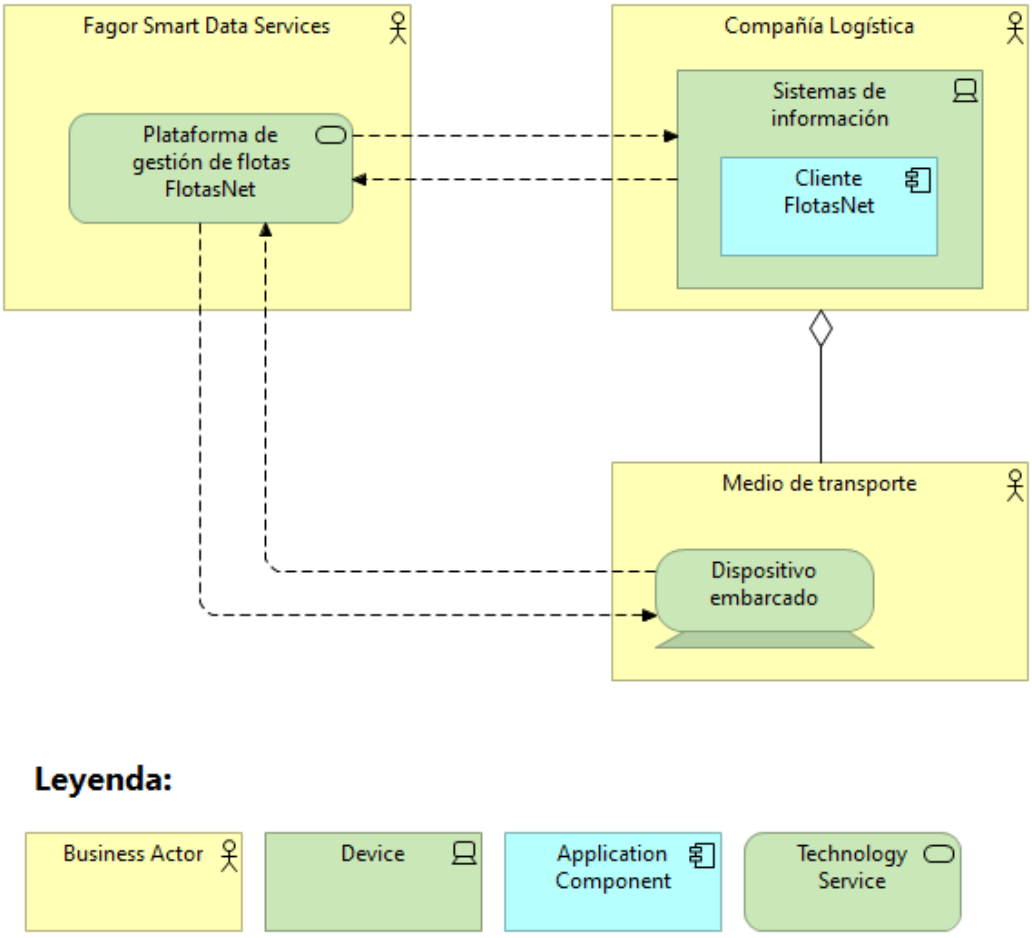


Figura 1: Modelo Archimate® de la plataforma tecnológica FlotasNet

El objetivo de este trabajo, colaboración entre la Universidad de Cantabria y SDS, es descubrir patrones en los datos que indiquen las decisiones que toman los controladores de tráfico (quienes asignan rutas, conductores y vehículos a los pedidos), y así poder determinar las variables más relevantes para construir un recomendador en el que poder apoyarse a la ahora de realizar esa tarea. Este objetivo final, como se verá más adelante, es casi utópico dado el material con el que se trabaja, por lo que los esfuerzos se han centrado en comprender el negocio y los datos, para después poder realizar un análisis que incluye la verificación de la calidad de los datos y la caracterización de elementos del negocio, en particular de las rutas que se definen para los pedidos.

Para poder lograr esta meta ha sido necesario concertar varias reuniones con SDS con el fin de comprender mejor la lógica del negocio y los datos que nos proporcionaron. Desgraciadamente, no se ha podido tener contacto directo con la propia ACME, sino solo con SDS, lo que ha hecho que algunas dudas sobre los datos que se han cumplimentado en la aplicación hayan quedado sin respuesta.

Este documento se estructura en las siguientes secciones: la Sección 2 recoge la explicación

de las técnicas de minería de datos empleadas, la Sección 3 describe los procesos por los que se analizaron los datos proporcionados por la empresa para entender lo que se registra en la base de datos y cómo funciona la lógica de negocio, en la Sección 4 se explica la selección de atributos para usar en el análisis, comprobando su calidad, y en la Sección 5 se detalla el análisis que se ha hecho sobre los datos definiendo diferentes premisas o verificando reglas de negocio y aplicando técnicas de minería de datos. Finalmente, en la Sección 6 se recogen las principales conclusiones y líneas de trabajo futuro.

1.1. Herramientas utilizadas

Como se ha mencionado previamente, toda la información con la que se ha trabajado proviene de una base de datos anonimizada de ACME, de la cual FlotasNet extrae su información para mostrarla en la interfaz gráfica de la aplicación. Para poder analizar las tablas y sus relaciones dentro de la base de datos relacional es necesario un gestor. En este caso se optó por uno de los más populares: SQL Server Management Studio de Microsoft. Además, fue conveniente usar concretamente esa herramienta, ya que es también la que se utiliza en Fagor y, por tanto, no da problemas de compatibilidad. Usando este software es posible realizar consultas sobre la base de datos y exportar los resultados a ficheros, para después analizarlos. El lenguaje que se emplea para ello es SQL (Structured Query Language).

En cuanto a la parte del análisis de los datos, se ha optado por usar el lenguaje Python, muy popular al trabajar en el ámbito de la ciencia de datos. El código se encuentra en un repositorio de GitHub[1]. A continuación se listan las principales librerías que permiten completar esta tarea:

- **pandas**: se usa en la manipulación y el análisis de datos. Gracias a ella es posible usar los datos extraídos de la base de datos y manipularlos para reorganizarlos (preprocesado), creando estructuras de datos llamadas *dataframe* sobre los que aplicar técnicas de minería de datos.
- **numPy**: necesaria al trabajar con pandas, ya que permite manipular *arrays* multidimensionales fácilmente.
- **re**: facilita la extracción de información (a través de expresiones regulares) con el objetivo de crear nuevos atributos derivados.
- **matplotlib y seaborn**: son muy útiles a la hora de imprimir gráficos.
- **plotly**: permite crear gráficos interactivos sobre los que hacer zoom cuando la cantidad de datos es demasiado grande.
- **sklearn**: proporciona los métodos necesarios para aplicar las técnicas de minerías de datos como *clustering*, reglas de asociación o árboles de decisión.
- **networkx**: permite crear y dibujar grafos.

[1]El código desarrollado se puede encontrar en: https://github.com/MiguelDeLaCal/MineriaDeDatosTFG

1.2. Metodología de trabajo

Para abordar este problema de minería de datos, se opta por seguir una metodología de trabajo estándar en este campo: *Cross Industry Standard Process for Data Mining* o CRISP-DM [2], creada inicialmente en 1996. Esta metodología define las fases del ciclo de vida de ciencia de datos. Para una mayor claridad, se cuenta con el diagrama de la Figura 2, que contiene cada una de estas fases y en la cual se indica cómo están interconectadas entre sí.

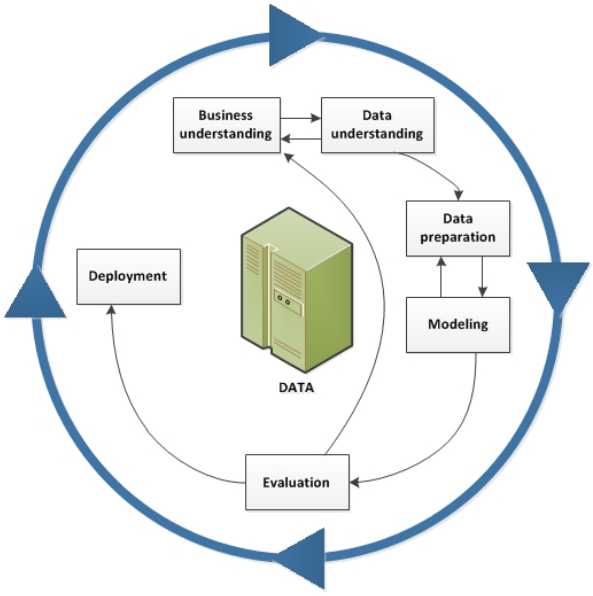


Figura 2: Diagrama de las fases del modelo CRISP-DM[1]

Antes de entrar en el detalle de cada fase, hay que remarcar que estas no se dan una única vez, ya que el proceso de minería de datos nunca termina de forma definitiva, a no ser que se agote el dinero o el tiempo que se puede o quiera dedicar al proyecto. Es un ciclo de vida en el que frecuentemente, incluso después de haber llegado a desplegar un modelo, se puede volver a una fase anterior de preparación de los datos tras haber obtenido nueva información que permita mejorar el modelo o hacer otro experimento. A continuación se listan las fases [3] que componen este proceso:

1. **Entendimiento del negocio**: incluye tareas como marcar los objetivos del proyecto, estos es, buscar lo que se quiere mejorar en el negocio, definir la meta de la minería de datos, así como el criterio de satisfacción con los resultados, o la elaboración del plan de proyecto, en el que se puede incluir el conjunto de herramientas y técnicas que se van a utilizar.

2. **Entendimiento de los datos**: en esta fase se desarrollan tareas como hacer una selección inicial de los datos de interés, describirlos y verificar su calidad.

[1]Imagen obtenida de: https://www.ibm.com/docs/es/spss-modeler/saas?topic=dm-crisp-help-overview

3. **Preparación de los datos**: en este punto se define el conjunto de datos con el que se realiza todo el trabajo. Para ello habrá que hacer una selección de los datos que formarán ese conjunto (selección definitiva tras haber pasado la fase anterior).

4. **Creación de modelos**: en esta etapa se seleccionan las técnicas que se deseen emplear y se construyen los modelos atendiendo a la correcta selección de parámetros.

5. **Evaluación**: en este paso se diseña la estrategia para evaluar el rendimiento de los modelos empleados y se ponen a prueba. Además, se definen los criterios de éxito y fracaso.

6. **Despliegue**: en esta fase se suele elaborar un plan de mantenimiento y monitorización de los modelos desplegados. Esta fase no se ha desarrollado en el proyecto.

En el presente trabajo se pueden enmarcar las tareas que se han realizado en las fases que se han descrito de la siguiente forma:

1. La fase de **entendimiento del negocio** (descrita con detalle en la Sección 3) ha consistido en reuniones con miembros de la empresa, seguidas del envío de varios documentos con consultas y documentación de la base de datos para verificar si las suposiciones que se tenían sobre el negocio eran correctas, además de consultar dudas sobre estas. Cabe mencionar que no se ha contado con un plan definido por parte de la empresa o una meta marcada sobre la que evaluar la satisfacción.

2. Dentro de la fase de **entendimiento de los datos** (Sección 3) se hizo una selección de tablas que se consideraron relevantes para la asignación de pedidos-rutas y se verificó la calidad de los datos que almacenaban. Tanto esta fase como la anterior se detallan en la misma sección, ya que se desarrollaron al mismo tiempo y un proceso contribuyó al otro. Es decir, se consultó a la empresa sobre la lógica de negocio para comprender los datos, y estos también se estudiaron y contrastaron para entender mejor otros procesos.

3. La **preparación de los datos** se refleja en la Sección 4. Se analizaron las tablas de la base de datos seleccionadas buscando aquellos campos que fuesen interesantes y de calidad para hacer un análisis.

4. En cuanto a la **creación de modelos**, los esfuerzos se centraron en el análisis de los datos (Sección 5) para encontrar patrones en la información almacenada ya que construir un modelo predictivo sin tener información suficiente (datos etiquetados) no fue viable. El análisis no se limitó a aplicar técnicas de minería de datos, sino que se investigaron una serie de aspectos de la base de datos para encontrar patrones o verificar que se sigue la lógica de negocio.

5. Para la fase de **evaluación** (Sección 6), se contrastaron los resultados obtenidos en el análisis de los datos con el personal técnico de Fagor, explicando los pasos y metodología aplicada para su obtención. La respuesta obtenida fue muy positiva: “por un lado se comprobó que las reglas de asociación se corresponden con la política real de planificación del cliente final y por otro lado se estableció una hoja de ruta para conseguir una mejora sustancial de la calidad de los datos que permita maximizar la aplicación de las tecnologías de analítica de datos.”

1.3. Cronología

A continuación se describe cómo se desarrollaron en el tiempo las fases descritas previamente en esta sección en el tiempo usando el diagrama de Gantt de la Figura 3.

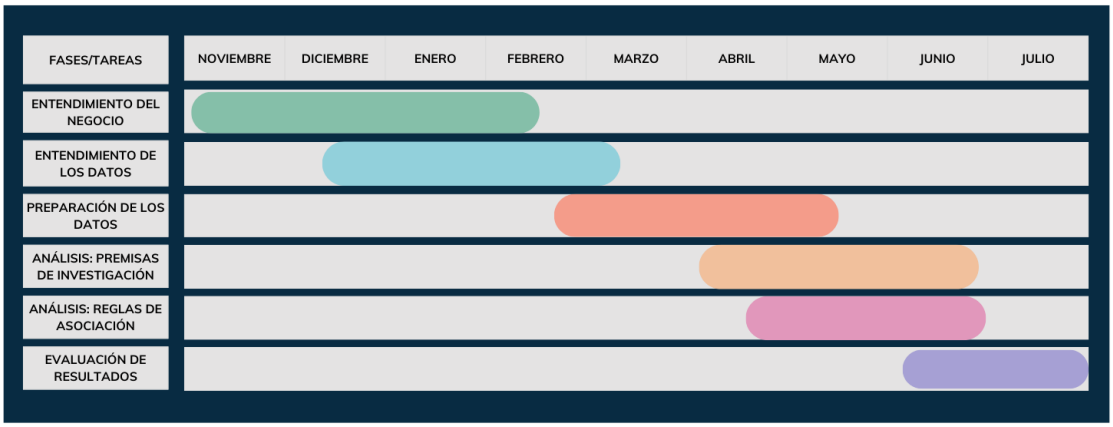


Figura 3: Diagrama de Gantt que representa el orden de desarrollo de las tareas del trabajo a lo largo del tiempo

Se comenzó por la fase de entendimiento del negocio, en la que los esfuerzos se concentraron en un primer momento en repasar la documentación disponible sobre FlotasNet y proyectos previos que se habían desarrollado con versiones similares de la base de datos con la que se trabajó, con el objetivo de tener una visión global de las funcionalidades de la aplicación y la lógica del negocio. Posteriormente, una vez se estudió la base de datos en detalle, se acotaron las tablas que eran de interés para las tareas de minería de datos. En paralelo se continuó la fase de entendimiento de negocio, al surgir dudas sobre este al ver los datos en detalle, enviando documentos a Fagor para contrastar si se comprendía correctamente.

La preparación de los datos comenzó mientras se resolvían las últimas dudas sobre los datos, ya que se tenía una idea clara de la información que se requería. Durante los meses de febrero y marzo se definió el *data set* que se describe en la Sección 4 y comenzó el análisis sobre los atributos seleccionados, descrito en la Sección 5.

Durante la fase de análisis, concretamente al plantear las premisas de investigación, se dedujo nueva información que poder incorporar al *data set*, por lo que simultáneamente continuó la fase de preparación de los datos hasta abril. A medida que se incorporó la nueva información, se fueron construyendo los conjuntos de reglas de asociación.

Finalmente, se dedicaron los meses de junio y julio a recopilar en esta memoria todo el desarrollo del trabajo, además de presentarlo a Fagor para evaluar cuán útiles han sido los resultados y conclusiones extraídas.

2. Técnicas de minería de datos empleadas

En esta sección se explican las técnicas de minería utilizadas para el procesado de datos, así como para extraer patrones. La descripción que se hará sobre estas no ahondará en todos los detalles, ya que estos métodos ya han sido estudiados en la asignatura del grado Aprendizaje Automático y Minería de Datos. Se detallan lo suficiente para comprender su uso en este trabajo, y otros temas, como aspectos internos de los algoritmos y su optimización, se remiten a la literatura.

2.1. Clustering

Hay una variedad de algoritmos que permiten aplicar *clustering* y se eligió usar *k-means*, ya que es uno de los más populares y sirve para el propósito de discretizar atributos. Dado un número entero k, el algoritmo *k-means* [4] divide el conjunto de datos que se proporcione en k grupos o *clusters*, formando una partición. Cada grupo viene representado por un centroide, que es el punto en la posición en la que se minimiza la distancia entre sí mismo y el resto de datos del *cluster* y que no necesariamente forma parte de los datos de entrada (como es el caso de otros algoritmos). Usando este algoritmo, se intenta minimizar la suma de las distancias cuadráticas de cada objeto de un *cluster* al centroide que corresponda. Esta expresión se puede representar usando la Ecuación (1).

$$E = \sum_{i=1}^{k} \sum_{x_j \in C_i} d(x_j, c_i)^2 \tag{1}$$

En esta ecuación k es el número de grupos, C_i es un grupo, x_j es un elemento del *cluster* C_i y c_i es el centroide asociado al *cluster* C_i. El término $d(x, y)$ representa la distancia euclídea entre dos vectores x e y, que es la que se emplea para medir la distancia entre cada objeto del *cluster* al centroide. Los elementos de los vectores que representan los objetos de los grupos representarían diferentes características, pero en este trabajo, al usarse *k-means* para dividir datos en rangos, estos son unidimensionales.

La documentación de la librería *sklearn* indica que se usa la implementación de este algoritmo diseñada por Lloyd [5], que a menudo se llama directamente *k-means* por ser la más común. El funcionamiento es el siguiente:

1. Se comienza con una asignación inicial de k centroides, que se definen escogiendo ejemplos de entre los n iniciales.

2. Se construyen las k particiones asignando cada punto de entre los n iniciales al centroide más cercano.

3. Una vez definidos los *clusters*, se vuelven a calcular los centroides como centro de masa de cada grupo.

4. Si los nuevos centroides son los mismos que los que se definieron previamente, se considera que el algoritmo converge y finaliza. En caso contrario, se repite el paso 2, volviendo a construir *clusters*, y se ejecutan iteraciones hasta que el algoritmo converja o se llegue al límite definido por el usuario.

A la hora de aplicarse *clustering* sobre un conjunto de datos, hay que indicar el número de *clusters* en los que se dividirá. Existen varios criterios por los que se puede decidir este valor. En el presente trabajo se opta por usar el método *elbow* o del codo, por ser simple e intuitivo. Consiste en representar gráficamente la métrica E de la Ecuación (1) en función del número de particiones que se hagan al aplicar el algoritmo sobre el conjunto de datos. Se busca el punto en el que la curva empieza a ser muy acentuada y deja de haber una gran mejora aumentando el número de *clusters* y se considera ese valor k como el mejor valor para usarse.

2.2. Árboles de decisión

Un árbol de decisión [6] es un modelo de predicción que genera un árbol y, a partir de él, un conjunto de reglas que permiten, dado un conjunto de datos, clasificarlos bajo una clase. En este caso se usará un árbol de clasificación, que predice valores discretos, pero existe otro tipo llamado árbol de regresión, que es capaz de predecir valores continuos.

Para generar este modelo se dividen los datos en dos subconjuntos: entrenamiento y test. El primero se usa para construir el clasificador, mientras que el segundo se usa para medir su precisión. Esta será determinada por el porcentaje de muestras del conjunto de test que son clasificadas correctamente.

Los atributos se dividen en dos tipos: aquellos cuyo dominio sean valores numéricos y aquellos cuyo dominio no lo sean (categóricos). Además, ha de definirse un atributo como clase objetivo. El propósito de la clasificación es construir el modelo que permita predecir la clase en los registros que no cuentan con ese dato.

Los dos principios por los que se construye un árbol de decisión son la selección de qué atributo se corresponde en cada nodo que se ramificará y el criterio por el que se dividirá (una ecuación o igualdad en el caso de atributos numéricos o el valor que toma en el caso de atributos categóricos). La manera en la que se afrontan estas decisiones es lo que diferencia a cada algoritmo que implementa el proceso. Aún así, todos los métodos de construcción tienen el mismo objetivo: lograr en los nodos hoja la máxima homogeneidad de los datos, es decir, que pertenezcan a la misma clase.

Concretamente, el algoritmo que está implementado en la librería *sklearn* es *Classification and Regression Trees* (CART). Este algoritmo se emplea tanto en clasificación como regresión, generando árboles binarios. El funcionamiento del algoritmo para la tarea de clasificar en clases es el siguiente:

1. Se comienza construyendo el árbol partiendo de un nodo que contiene todos los datos de entrenamiento.

2. Para decidir el atributo y la ramificación, CART usa como métrica el índice de Gini, que se puede expresar, de forma general, usando la Ecuación (2).

$$\text{Gini}(D) = 1 - \sum_{i=1}^{n} p_i^2 \tag{2}$$

   donde D es el conjunto de datos y p_i la frecuencia relativa de la clase i en D. Este índice mide lo homogéneo que es el conjunto de datos. En el algoritmo CART se prueba a

dividir todos los atributos en sus posibles valores (si son categóricos) o bajo diferentes puntos de corte (si son numéricos y siempre con una igualdad/desigualdad/comparación que se cumpla o no) y se busca maximizar la ganancia de Gini. Esta compara la impureza antes de la división con la que se obtiene al hacer la división en dos conjuntos de datos (haciendo una media ponderada). Se expresa usando la Ecuación (3).

$$\text{Ganancia de Gini} = \text{Gini}(D) - (\frac{|D_1|}{|D|}\text{Gini}(D_1) + \frac{|D_2|}{|D|}\text{Gini}(D_2)), \tag{3}$$

donde $|D|$ es número de ejemplos del conjunto de datos D del nodo padre y $|D_1|$ y $|D_2|$ son el número de ejemplos de los nodos hijo D_1 y D_2, respectivamente.

3. Se repite este proceso de ramificar siguiendo ese criterio de forma iterativa hasta que se llegue a una profundidad del árbol que se haya indicado o no sea posible hacer más divisiones.

Para comprender la arquitectura de un árbol de decisión, se usa el ejemplo de la Figura 4.

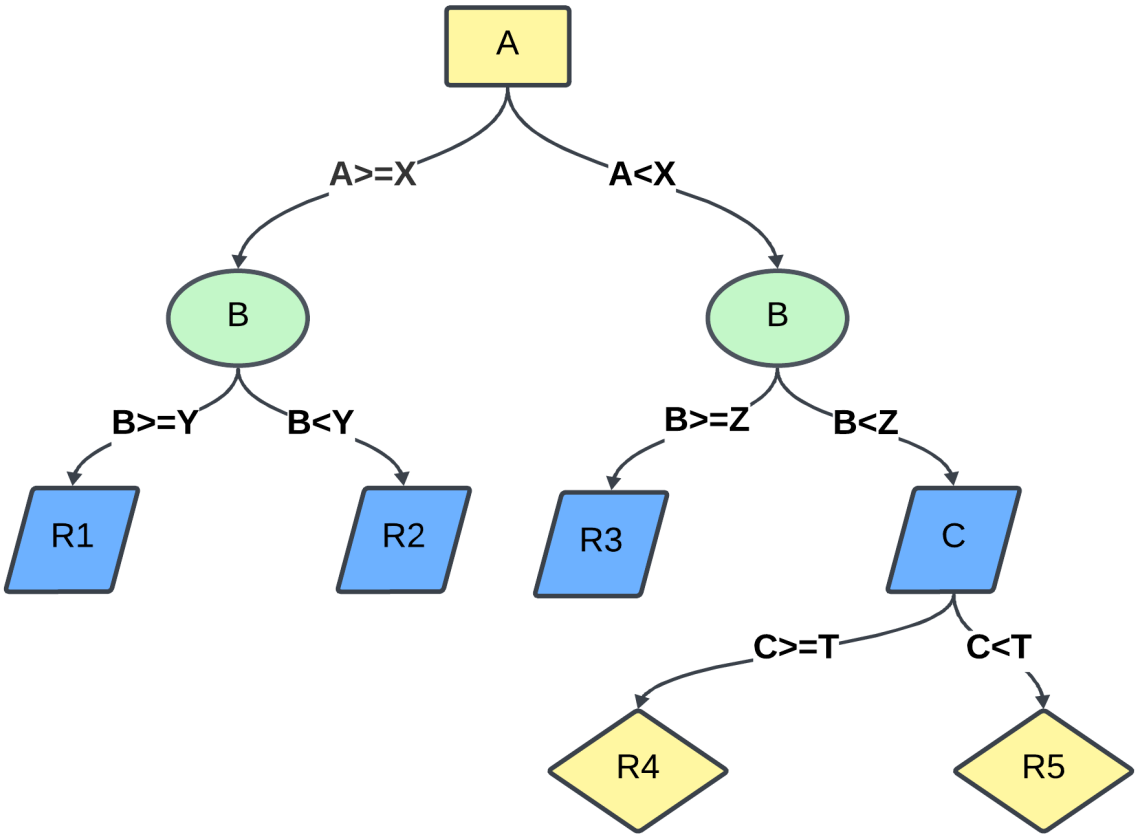


Figura 4: Arquitectura simple de un árbol de decisión binario.

En él se pueden ver los elementos clave:

- Nodo raíz y resto de nodos de decisión: estos representan la variable por la que se ramificará el árbol según su valor. En este caso, los nodos son A (raíz), B (que aparece dos veces porque en el segundo nivel el mismo atributo se usa para ramificar) y C.
- Nodos hoja: al llegar a estos nodos se definen reglas por las que se llega a un resultado final. Si el árbol fuese de clasificación, tendría definida la clase que se predice y en el caso de que fuese de regresión, sería un valor continuo. En el árbol de la Figura 4 estos nodos son $R1$, $R2$, $R3$, $R4$ y $R5$. Los caminos desde la raíz hasta cada uno de los nodos hoja permiten definir las reglas por las que se hace la clasificación.
- Ramificaciones: en cada nodo hay una condición por la que se ramifica el árbol. En el ejemplo los criterios por los que se marcan estas divisiones son los valores X, Y, Z y T.

2.3. Reglas de asociación

Las reglas de asociación [7] son una de las técnicas de minería de datos más útiles para este trabajo, ya que los resultados son intuitivos y fácilmente interpretables por personas sin conocimientos teóricos de minería de datos. Para construir estas reglas se usa una representación transaccional de los datos, es decir, cada fila del *data set* se representa como un conjunto de ítems que representan las propiedades o atributos de esa instancia. Las reglas de asociación relacionan conjuntos de atributos en los datos en forma de antecedente y consecuente. Si se tiene un conjunto de atributos X y otro Y y se logra definir una regla $X \rightarrow Y$, significa que frecuentemente cuando aparece en una transacción el conjunto X, también lo hará Y.

Hay dos términos clave a la hora de definir las reglas de asociación: soporte y confianza. Tomando la nomenclatura anterior, se define $s(X)$, el soporte de X, como la frecuencia con la que ese conjunto de atributos aparece en las instancias del conjunto de datos. El otro término es la confianza, una métrica que mide la intensidad de implicación, es decir, una forma de medir las excepciones que no cumplen la relación. Se expresa mediante la Ecuación (4).

$$\text{confianza}(X \rightarrow Y) = \frac{s(X \cup Y)}{s(X)}, \tag{4}$$

Esta métrica es relativamente natural y es fácil de explicar a un usuario no experto. Para la obtención de reglas de asociación se definen umbrales mínimos tanto para el soporte como para la confianza. Esto quiere decir que una regla $X \rightarrow Y$ solo se tendrá en cuenta si $s(X \cup Y)$ supera el umbral definido para el soporte y la confianza está por encima del mínimo definido.

Existe otra medida por la que se pueden filtrar reglas de asociación y lograr una mejor selección. Esta es el *lift*, y mide la fuerza de la asociación entre los dos conjuntos de atributos. Se expresa según la Ecuación (5).

$$\text{lift}(X \rightarrow Y) = \frac{s(X \cup Y)}{s(X) * s(Y)} \tag{5}$$

Si el *lift* es cercano a uno, significa que las probabilidades de que ocurran el antecedente y el consecuente son independientes entre sí y, por tanto, la calidad de la regla es baja. Si es mayor que uno, indica el grado en el que los dos conjuntos son dependientes, es decir, que un conjunto tiende a aparecer cuando lo hace el otro. Por otro lado, un *lift* inferior a uno indica lo contrario.

Se ha mencionado que, para construir las reglas, se obtienen conjuntos de atributos frecuentes teniendo en cuenta los umbrales definidos. Para este proceso se sigue el algoritmo *apriori* [8]. Este forma los conjuntos frecuentes haciendo un recorrido en anchura (*breadth-first*). Se comienza definiendo un conjunto por cada ítem individual y se descartan aquellos que no alcancen el umbral de soporte mínimo que se ha indicado. Esto se debe a que ningún otro conjunto que incluya esos ítems puede alcanzar el umbral (esta es la poda "*apriori*"). El siguiente paso es hacer combinaciones de los ítems y volver a filtrar solo aquellas que cumplan con el soporte mínimo. Con estas parejas se construyen tríos combinando aquellas que compartan el primer ítem y se vuelve a filtrar por el soporte. Este proceso se repite de manera iterativa en el que se combinan dos conjuntos frecuentes de tamaño k que compartan los primeros $k-1$ ítems y se comprueba si estos nuevos subconjuntos son frecuentes. Este proceso termina cuando no se pueden añadir más atributos a los conjuntos o ninguno de los candidatos de una iteración puede considerarse frecuente.

Otro paso importante a la hora de interpretar las reglas de asociación que se construyen con los datos es discriminar aquellas que sean redundantes respecto a otras, ya que al ejecutar el algoritmo pueden aparecer muchas reglas en el orden de cientos e incluso miles. Por ejemplo, tenemos dos reglas $R1 : X \rightarrow Y$ y $R2 : X' \rightarrow Y'$. Se considera que $R1$ es redundante con respecto a $R2$ si $s(X \rightarrow Y) \geq s(X' \rightarrow Y')$ y $c(X \rightarrow Y) \geq c(X' \rightarrow Y')$ en cualquier *data set*. La implementación del algoritmo de *sklearn* no filtra reglas redundantes, por lo que es trabajo del analista buscar entre las reglas que se generan y discriminar aquellas redundantes.

3. Entendimiento del negocio y de los datos

Una tarea clave a la hora de realizar este trabajo ha sido comprender la lógica del negocio, así como los datos de los que se dispone en la base de datos proporcionada por Fagor. Tras varias reuniones con la empresa, se acotó el conjunto de tablas (entre las 627 disponibles en la base de datos) que eran necesarias para representar el proceso en el cual, a partir de un pedido, se asocian rutas, conductores y vehículos. También se buscó recopilar la mayor cantidad de información posible para caracterizar las rutas tomadas. Para hacer este estudio, Fagor proporcionó una copia de la base de datos anonimizada con información de pedidos desde junio de 2023 hasta abril de 2024. Como se indicó previamente, el contexto del negocio es una empresa especializada en el transporte de pedidos en largas distancias con un mínimo número de pausas en el viaje al asignarse con bastante frecuencia dos conductores al mismo pedido. El número de conductores siempre es uno o dos.

La selección se limitó a las siguientes tablas del diagrama de la Figura 5. Los únicos atributos que se muestran son las claves primarias (*primary keys*) que identifican inequívocamente una instancia de la tabla y las claves foráneas (*foreign keys*) por las que se establecen las relaciones. Cabe mencionar que, a pesar de ser una base de datos relacional, no están definidas explícitamente las *foreign keys*, por lo que se han conocido tras consultarlo en reuniones con la empresa y verificar las relaciones y coherencia de los datos.

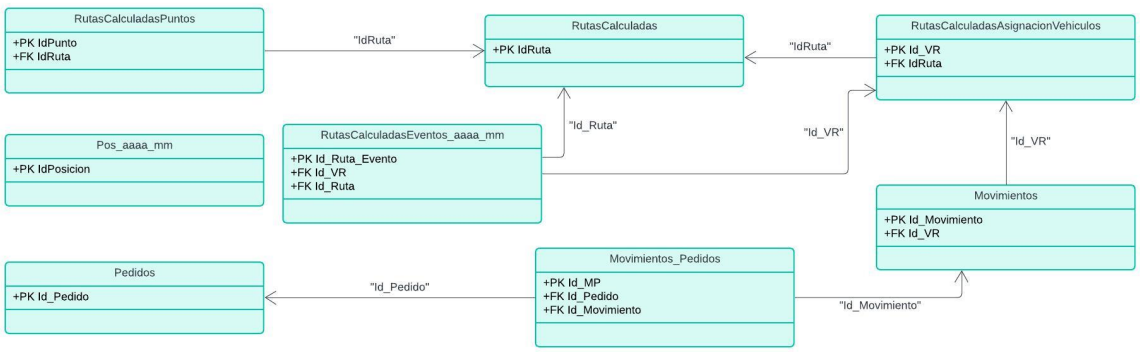


Figura 5: Diagrama de las principales tablas que recogen el proceso de gestión de pedidos-rutas

La combinación de la información de estas tablas es lo que constituye el *data set* principal con el que se trabajó. Este conjunto resume la principal información involucrada en el proceso de asignación de pedidos-rutas:

- La tabla Pedidos almacena información (volcada del sistema ERP) referente a los datos del pedido que se hace efectivo. Además de su identificación en el sistema, se guardan datos sobre las fechas en las que se pone en marcha el pedido, su estado, dimensiones y peso.
- La tabla Movimientos almacena información sobre el estado del viaje que se hace y quién lo realiza (asignación de uno o dos conductores que se turnan). Esta tabla tiene

información redundante que ya se encuentra en RutasCalculadasAsignacionVehiculos. Dado que no se ha encontrado ningún movimiento relacionado con más de una ruta (tabla RutasCalculadas), entendemos que en el caso de ACME, un movimiento es una ruta, y viceversa. A través de Movimientos se relaciona una ruta con un pedido, usando la tabla intermedia Movimientos_Pedidos.

- RutasCalculadas contiene información referente a la ruta que se ha solicitado. Tiene información sobre estimaciones de tiempo, distancia, y también indicaciones para evitar ciertos caminos. Su campo Polilyne es el que permite identificar punto a punto cada parte de la ruta y así poder dibujarse en la interfaz de Flotasnet.

- RutasCalculadasAsignaciónVehículos contiene la información referente al vehículo (su identificador) que se usa en la ruta asignada y campos que relacionan esta tabla con otras que permiten extraer información detallada sobre vehículo y caja tractora. Además, almacena las fechas estimadas (bajo el criterio de la torre de control) de inicio y fin antes de que comenzase una ruta, así como las fechas reales en las que comenzó y finalizó la ruta.

- En las tablas Pos_aaaa_mm se almacena información proporcionada por el sistema GPS del vehículo, lo que permite posicionarlo en unas coordenadas concretas y también tener acceso a señales que proporcionan diferentes sensores del vehículo. Cuenta con un gran número de instancias debido al envío periódico de información por parte del GPS, motivo por el que se definen varias tablas para las posiciones de cada mes.

- La tabla RutasCalculadasPuntos almacena los puntos que se definen desde la interfaz de Flotasnet para modificar el trayecto de la ruta que se calcula de punto a punto de forma predeterminada (proporcionada por Google Maps). Estos puntos pueden definirse como paradas para indicar si se realiza alguna tarea como cargar o descargar mercancía. Además, los puntos correspondientes a lugares recurrentes en las rutas, como puede ser un almacén, también vienen informados con un campo que lo identifica en otra tabla llamada Marcadores.

- La tabla RutasCalculadasEventos_aaaa_mm, que recopila eventos de las rutas mensualmente, almacena la misma información que la tabla de posiciones, discriminando solo aquellas en las que sucede un evento del estilo de salida/vuelta a ruta, inicio de ruta con retraso, finalización de la ruta en tiempo, etc.

Para analizar estas tablas no se contó con una documentación escrita, por lo que se realizó un informe para contrastar con la empresa si se comprendía correctamente su funcionamiento. La documentación de las tablas Movimientos, RutasCalculadas, RutasCalculadasAsignaciónVehículos, Pedidos y Pos_aaaa_mm se detalla en las Tablas 1, 2, 3, 4 y 5, respectivamente. No todas las tablas del diagrama de la Figura 5 se documentaron, por diferentes motivos. La tabla de eventos de ruta no se detalla por almacenar datos muy similares a los de posiciones, añadiendo la información del evento que registra. Las tablas restantes, RutasCalculadasPuntos y Movimientos_Pedidos, tampoco se incluyeron en este estudio porque se pudo comprender fácilmente sus campos sin necesidad de consultar a la empresa. Para cada punto de la ruta

se registra la propia ruta a la que pertenece, las coordenadas, si es una parada o no (indicando en otro campo el motivo) y una referencia al lugar con el que se corresponde (como un almacén) si está registrado en la base de datos. En cuanto a la tabla Movimientos_Pedidos, solo almacena parejas de identificadores de movimientos y pedidos para asociarlos, porque un pedido se puede entregar en varios movimientos.

A continuación se muestra la documentación más exacta a la que se ha podido llegar de las tablas mencionadas.

Tabla 1: Descripción de los campos de la tabla Movimientos

CAMPO	DESCRIPCIÓN
Id_Movimiento	Identificador de la tabla Movimientos dentro de la base de datos
CodigoViaje	Código de viaje del movimiento, coincide con el valor del campo nombre de una instancia de RutasCalculadas
Id_Vehiculo	Identificador del vehículo que realiza el movimiento
Matricula	Matrícula del vehículo asignado al movimiento
Id_Conductor	Identificador del conductor del vehículo asociado, referencia a la tabla Conductores en la base de datos
nombre_Conductor	Nombre del conductor del vehículo asociado
Apellido1_Conductor	Primer apellido del conductor del vehículo asociado
Apellido2_Conductor	Segundo apellido del conductor del vehículo asociado
Id_VR	ID que referencia al campo Id_VR en la tabla RutasCalculadasAsignacionVehiculos
Estado_Movimiento	Estado del movimiento (1-Enviado, 2-Aceptado, 3-En curso, 4-Finalizado, 5-Cancelado)
Fecha_Alta	Fecha en la que se ha creado la instancia en la tabla Movimientos
Fecha_Ult_Modif	Fecha de la última modificación en esta instancia de Movimientos
LogPedidos	No se usa en la empresa que se analiza en este trabnajo. En otro cliente este campo se usa para guardar la petición que envía para insertar un movimiento
Telefono	Teléfono de contacto del conductor del vehículo
Trunking	Se usa en otra empresa, no analizada en este trabajo
CodigoLinea	Se usa en otra empresa, no analizada en este trabajo
Transportista	Posible identificador de un transportista que participe en la entrega de algún pedido que se lleve en este movimiento
MatriculaAux	Segundo campo para introducir la matrícula del vehículo. Se usa en otra empresa, no analizada en este trabajo
DuracionTeorMinut	Duración teórica estimada para finalizar el movimiento
Id_CarrierRep	Se usa en otra empresa, no analizada en este trabajo

Continúa en la siguiente página

Tabla 1: (Continuación)

CAMPO	DESCRIPCIÓN
Id_SalesRep	Identificador del jefe de ventas
Recepcion_Ot	Cuando se manda un movimiento a la app del conductor, se actualiza este campo de fecha. Si no está actualizada, quiere decir que el conductor no recibió el movimiento en la app
Remolque	Identificador del remolque que lleve el vehículo asignado
Id_Conductor2	Identificador de un posible acompañante o copiloto que se turne con el conductor principal
nombre_Conductor2	Nombre del acompañante/copiloto
Apellido1_Conductor2	Primer apellido del acompañante/copiloto
Apellido2_Conductor2	Segundo apellido del acompañante/copiloto
Comentarios	Comentario que se puede introducir desde el aplicativo sobre el curso del movimiento.

Tabla 2: Descripción de los campos de la tabla RutasCalculadas

CAMPO	DESCRIPCIÓN
IdRuta	Identificador de la ruta
nombre	Código de viaje de la ruta
TipoCalculo	No es relevante para el entendimiento del negocio, ya que es un campo para optimizar operaciones de la aplicación web
VelocidadUrbana, VelocidadNacional, VelocidadAutopista	Estos tres campos indican los límites de velocidad a principales en esos tipos de vía dentro de la ruta definida
VelocidadFerry	Se desconoce su función
TiempoTotal	Tiempo total estimado para completar la ruta (horas)
DistanciaTotal	Distancia total de la ruta (kilómetros)
IdUser	No es relevante para el entendimiento del negocio
Fecha_Alta	Fecha en la que se crea la ruta
Fecha_Baja	Fecha en la que se dio de baja la ruta (se considera eliminada)
EvitarPeaje	(0/1) Evitar peajes en la ruta
EvitarAutopista	(0/1) Evitar ir por autopista
OptimizarPuntosRuta	(0/1) Se desconoce su función
TipoMovimiento	No se usa en la empresa que se analiza en este trabajo

Continúa en la siguiente página

Tabla 2: (Continuación)

CAMPO	DESCRIPCIÓN
Polyline	Lista de puntos que constituyen el dibujo de la ruta en el mapa
MargenInferior	Margen inferior (minutos) en el que se almacenen datos respecto a la fecha de inicio
MargenSuperior	Margen superior (minutos) en el que se almacenen datos respecto a la fecha de inicio. Esto quiere decir que desde que se llega a la fecha estimada final, se tomaría este margen de tiempo para poder seguir registrando eventos hasta que se cierre automáticamente
ArranqueParada	(0/1) Si es 1, quiere decir que si el vehículo da una posición dentro del radio de la ruta (puede ser un radio de 500 m por ejemplo), pero esa posición no está dentro del trayecto (polyine) marcado por la ruta, se manda un evento de inicio de ruta. Si es 0, quiere decir que si el vehículo da una posición dentro del radio de la ruta, y además esa posición no está dentro del trayecto marcado por la ruta, no se manda un evento de inicio de ruta. Si es 0 dando el vehículo una posición dentro del radio de la ruta, y además esa posición sí está dentro del trayecto marcado por la ruta, si se manda un evento de inicio de ruta.
ArranqueRuta	(0/1) Si es 0, quiere decir que, si el vehículo no ha dado posición en la parada de origen, no va a marcarse como iniciada la ruta, aunque dé posiciones dentro de la ruta (polyline). Si es 1, quiere decir que, si el vehículo no ha dado posición en la parada de origen, si da una posición dentro de la ruta más adelante, marcará la ruta como iniciada.
TipoCartografia	Identificador del tipo de mapa que se muestre en la aplicación al mostrar la ruta

Tabla 3: Descripción de los campos de la tabla RutasCalculadasAsignaciónVehículos

CAMPO	DESCRIPCIÓN
Id_VR	Identificador de la asignación de la ruta al conductor, vehículo, caja, etc.
IdRuta	Identificador de la ruta, referencia a la tabla RutasCalculadas
IdVehiculo	Identificador del vehículo, referencia a la tabla Vehículos
fecha_desde	Fecha definida por una persona que sirve de estimación para el inicio de la ruta
fecha_hasta	Fecha definida por una persona que sirve de estimación para la finalización de la ruta
NroSerie	Número de serie del vehículo
Id_Caja	Identificador de la caja asignada, referencia a la tabla Cajas
Matricula	Matrícula del vehículo asignado
notificacion_web, notificacion_api, notificacion_eventos, notificacion_email	Estos cuatro campos de notificación son códigos que se usan en la aplicación
observaciones	Comentarios que pueda dejar el administrador que registre la asignación
Fecha_Inicio_Real	Fecha real que introduce una persona manualmente para asignar el vehículo
Fecha_Final_Real	Fecha real hasta la que es efectiva la asignación introducida por una persona
Estado	Último estado registrado en la ruta, proveniente de la tabla de eventos de la ruta
Ult_Parada	Se actualiza con el identificador del último punto en el que se haya estado y que coincida con el valor del campo IdPunto de la tabla RutascalculadasPuntos
Fecha_Ult_Estado	Fecha en la que se cambió el valor del campo Estado por última vez
MargenInferior	Mismo significado que en la tabla de RutasCalculadas
MargenSuperior	Mismo significado que en la tabla de RutasCalculadas
Pasajeros	Número de pasajeros
RouteApproach	(0/1) Si es 1, el servicio se encarga de mandar un aviso al cliente cuando esté próximo al punto inicial
AlarmaETA	(0/1) Si es 1, el servicio se encarga de mandar un aviso si el ETA (tiempo estimado de llegada) al final de la ruta es mayor a 2 horas

Continúa en la siguiente página

Tabla 3: (Continuación)

CAMPO	DESCRIPCIÓN
AlarmaPtoPer	(0/1) Si es 1, el servicio se encarga de mandar un aviso cuando el vehículo ha hecho una parada en un punto permitido
AlarmaPtonop	(0/1) Si es 1, el servicio se encarga de mandar un aviso cuando el vehículo ha hecho una parada en un punto no permitido

Tabla 4: Descripción de los campos de la tabla Pedidos

CAMPO	DESCRIPCIÓN
Id_Pedido	Identificador del pedido
NPedido	Número/código del pedido
NPedidoExt	Número/código del pedido extra
Fecha_Alta	Fecha en la que se da de alta el pedido en Flotasnet
Fecha_Baja	Fecha en la que se da de baja el pedido en Flotasnet (solo cuando se considera eliminado)
Id_Tipo_Pedido	Identificador del tipo de pedido. En desuso
Estado	Estado del pedido (1-Asignado, 2-Aceptado, 3-No iniciado, 4-Finalizado, 5-En curso en hora, 6-En curso retrasado, 7-En curso adelantado, 8-Cancelado, 9-Rechazado)
NroPales	Número de palés que se cargan en el pedido
Peso	Peso de la carga del pedido
VolumenM3	Volumen del pedido en metros cúbicos
TipoMercancia	Este campo define el tipo de mercancía (texto)
Gastos	Gastos que estime la empresa: gasolina, peajes, salario conductor, etc. no está definido qué puede ser concretamente
Fecha_Inicio	Fecha de inicio real del pedido (introducido por una persona)
Fecha_Fin	Fecha de finalización real del pedido (introducido por una persona)
Margen_Inicio	Margen en minutos respecto a la fecha de inicio en la que se abriría automáticamente el pedido de no indicarse previamente

Continúa en la siguiente página

Tabla 4: (Continuación)

CAMPO	DESCRIPCIÓN
Margen_Fin	Margen en minutos respecto a la fecha de fin en la que se cerraría automáticamente el pedido de no indicarse previamente
Fecha_Inicio_Real	Fecha en la que el vehículo ha iniciado la ruta por posicionamiento GPS. En comparación con la Fecha_Inicio, esta última es teórica, pero la real es a la que en la realidad se ha iniciado la ruta
Fecha_Final_Real	Fecha en la que el vehículo ha finalizado la ruta por posicionamiento GPS. En comparación con la Fecha_Fin, esta última es teórica, pero la real es a la que en la realidad se ha iniciado la ruta
Fecha_Ult_Estado	Fecha del último cambio en el campo Estado
Ult_OT	Identificador de la última orden de trabajo del pedido
Via_Fin	En desuso
Id_Marca_Ini	Identificador del lugar de comienzo del pedido, referencia a tabla Marcadores
Id_Marca_Fin	Identificador del lugar de finalización del pedido, referencia a tabla Marcadores
LineaPedido	Código de la línea del pedido
TipoTransporte	Indica el tipo de transporte. Solo se ha llegado a definir en la base de datos como “COMPLETO” o “vacío”
ADR	La empresa cliente solicitó crear este campo que solo es visible por los conductores desde la aplicación que usan para trabajar
ReferenciaCliente	Código que identifica al cliente dentro del ERP
ReferenciaFactura	Código que identifica la factura del pedido dentro del ERP

Tabla 5: Descripción de los campos de la tablas de posiciones Pos_aaaa_mm

CAMPO	DESCRIPCIÓN
IdPosicion	Identificador de la posición
Fecha_Hora	Fecha y hora en la que se registró
Id_Vehiculo	Identificador del vehículo que se localiza, referencia a la tabla Vehículos
Latitud	Latitud de la posición en valor decimal

Continúa en la siguiente página

Tabla 5: (Continuación)

CAMPO	DESCRIPCIÓN
Longitud	Longitud de la posición en valor decimal
Altitud	Altitud a la que se encuentra el vehículo. No hay datos, por lo que se desconoce la medida
Velocidad	Velocidad (km/h) a la que se mueve el vehículo en ese instante
Rumbo	Grados de 0 a 360, para luego pintar el camión en el mapa de la web mirando al norte, sur, noroeste, etc
Id_Estado	Estado del vehículo, referencia a la tabla Estado_Camion
Id_Caja	Identificador de la caja que tiene acoplada el vehículo, referencia a Cajas
Id_Conductor	Identificador del conductor del vehículo, referencia a Conductores
Id_Remolque	En desuso
Distancia	Kilómetros del vehículo acumulados durante toda su vida útil
ED1-3, SD4, FA1-4, FD1-4	Varios campos para registrar eñales de sensores del vehículo. Desde el aplicativo se selecciona qué sensor asignar a qué entrada digital de este rango
Fecha_Hora_EVT	Fecha en la que tuvo lugar el evento que provocó que se registrase la posición.
Fecha_Hora_sistema	Fecha en la que se graba en la base de datos la posición
Fecha_Hora_Recepcion	Fecha y hora en la que se recibe la posición en el sistema (si no hay retrasos en el procesado debería ser casi igual al anterior campo)
Id_Cobertura_IdOT	No tiene un uso en la empresa estudiada
Tipo_Estado_OT	Tipo de estado de la orden de trabajo, referencia a otra tabla OT_Tipos_Estados

Aunque no aparecen en el diagrama de la Figura 5, otras tablas fueron analizadas por considerarse parte principal de la lógica de negocio. Estas son EventosHistorico y Tramos. EventosHistórico recoge datos sobre la posición del vehículo en un momento concreto, pero solo aquellas en las que sucede un evento relacionado con la conducción del vehículos (que puede ser del tipo de arranque de motor, parada de motor, cambio de conductor, bajada de combustible en ralentí, frenada brusca, enganche remolque, etc.). La tabla Tramos guarda información sobre la calidad de la conducción en secciones que vienen marcadas por eventos (desde que se arranca motor hasta que se para). Estas finalmente no se consideraron para analizarlas en mayor profundidad por recomendación de Fagor, ya que no proporcionan información relacionada con las decisiones que toma el controlador de tráfico.

Tras analizar los datos de las tablas, se describe el proceso que se sigue desde que se planifica una ruta hasta que se deja en el destino la mercancía y el pedido se da por finalizado.

1. ACME tiene registrado en su ERP un pedido. Se solicita en Flotasnet una ruta entre el origen y el destino de entrega. La información del pedido se registra en la tabla Pedidos y su estado sería "No iniciado" (el campo Estado de la tabla toma valor 3). Si el pedido se entregará usando diferentes vehículos/cajas/conductores en diferentes secciones del trayecto, se solicitan varias rutas, una para cada parte. La torre de control decide previamente si emplear una o varias rutas.

2. Al solicitar la ruta, se crea una instancia en la tabla RutasCalculadas, donde se recoge información como distancia total y el tiempo estimado en completarla (campos DistanciaTotal y TiempoTotal calculados desde Google Maps) o indicaciones sobre cómo evitar peajes o autopistas (campos EvitarPeaje y EvitarAutopista). A su vez, se crean instancias en la tabla RutasCalculadasPuntos, que se corresponden con puntos por los que se requiere que el conductor pase. Además del origen y destino, se pueden incluir otros puntos donde sea necesario parar (se indica si es necesario o no en el campo Parada de la tabla). Estos puntos se definen junto a la ruta y se asignarán o bien para forzar seguir una ruta que podría no ser la más corta por algún motivo (como una gran cantidad tráfico prevista) o para hacer una tarea en un lugar concreto (parar para cargar/descargar mercancía o enganchar/desenganchar remolque).

3. A continuación, se asigna un conductor, vehículo y caja a esta ruta, siguiendo el criterio del controlador de tráfico. Estos datos se almacenan en la tabla RutasCalculadasAsignacionVehiculos.

4. Una vez se haya decidido la ruta, una o varias si es el caso, además de asignarse vehículo, conductor y caja, se define un movimiento por cada ruta empleada. Estos datos se recogen en la tabla Movimientos. La asignación de qué pedidos se llevan en un movimiento se guarda en la tabla Movimientos_Pedidos. Solo hay una ruta asignada con un movimiento y viceversa.

5. Cuando se comienza el trayecto se generan periódicamente datos sobre su posición gracias al sistema GPS (por ejemplo, cada cinco minutos). Estos datos se almacenan en la tabla Pos_aaaa_mm (según el año y mes en el que se registran). Usando las posiciones que llegan al sistema, se generan los eventos de inicio, salida de ruta, parada, fin de ruta, salida/vuelta de ruta, etc. Existe un servicio externo que recibe las posiciones del GPS y se encarga de detectar esos eventos, para luego enviar esos datos a la base de datos. Estas posiciones, junto con información adicional sobre la naturaleza del evento, son la base de las instancias que se registran en la tabla RutasCalculadasEventos_aaaa_mm, relacionada a través del identificador de la ruta con la tabla RutasCalculadas.

6. En la tabla EventosHistorico se almacenan solo posiciones en las que hay un evento (arranque de motor, parada de motor, puesta en marcha, parada, etc.). En su mayoría, durante una ruta, los eventos se limitan a arranques, salidas o entradas a zona y descansos que pueda tomar el conductor. Otros que son útiles para marcar alguna ruta pueden

ser los referentes a frenadas bruscas, excesos de velocidad o colisiones. Si se dan con una frecuencia considerable, sería necesario reconsiderar la ruta usada.

7. A partir de estos datos se generarán a su vez instancias en la tabla Tramos, que almacenan información desde que se arranca el motor del vehículo hasta que se apaga, y añaden datos sobre la calidad de la conducción.

Como se ha indicado anteriormente, se prescindirá de Tramos y EventosHistorico para juntar el *data set* sobre el que se trabajará, pero para decidirlo se tuvo que entender la función que cumplen en la lógica de negocio.

A modo de ejemplo ilustrativo, se puede seleccionar un pedido e intentar reconstruir el orden en el que se introdujeron los datos en la aplicación, así como verificar que los pasos descritos anteriormente son correctos. De forma arbitraria, se elige el pedido con identificador 5894. Buscando en la tabla Movimientos_Pedidos, se puede encontrar el movimiento asignado, que en este caso es solo uno. Conociendo el movimiento (identificador 1878), se puede relacionar con la asignación de ruta al vehículo a través del campo Id_VR, que es el identificador en RutasCalculadasAsignacionVehiculos. En esta última podemos conocer el identificador de la ruta que se toma: IdRuta. Finalmente, es posible reconstruir esta ruta conociendo los puntos que se han definido para ella en RutasCalculadasPuntos, como se puede ver en la Tabla 6.

Tabla 6: Puntos definidos para la ruta de ejemplo (tabla RutasCalculadasPuntos)

IdPunto	IdRuta	nombre	Latidud	Longitud	Parada	Tiempo	Indice	Distancia
26895	7961	ENGANCHE_REM	42.728996	-1.614070	1	0	0	0
26896	7961	NaN	42,690917	-1.649310	0	NaN	340	0
26897	7961	NaN	42.502163	-1.673596	0	NaN	1010	0
26898	7961	NaN	42,461433	-1.650348	0	NaN	1167	0
26899	7961	CARGA_INI	40,500242	-3.384835	1	300	7298	351904
26900	7961	NaN	42.309144	-1.647827	0	NaN	13407	351904
26901	7961	NaN	42.634635	-1.639013	0	NaN	14441	351904
26902	7961	DESENGANCHE	42.728996	-1.614070	1	562	14732	702925

Usando la herramienta Google Maps, se puede reconstruir la ruta, como se puede ver en la Figura 6. Se trata de un viaje de ida y vuelta desde Pamplona hasta Alcalá de Henares, en el que las únicas paradas son en Pamplona (enganche y desenganche del remolque) y en Alcalá de Henares (carga).

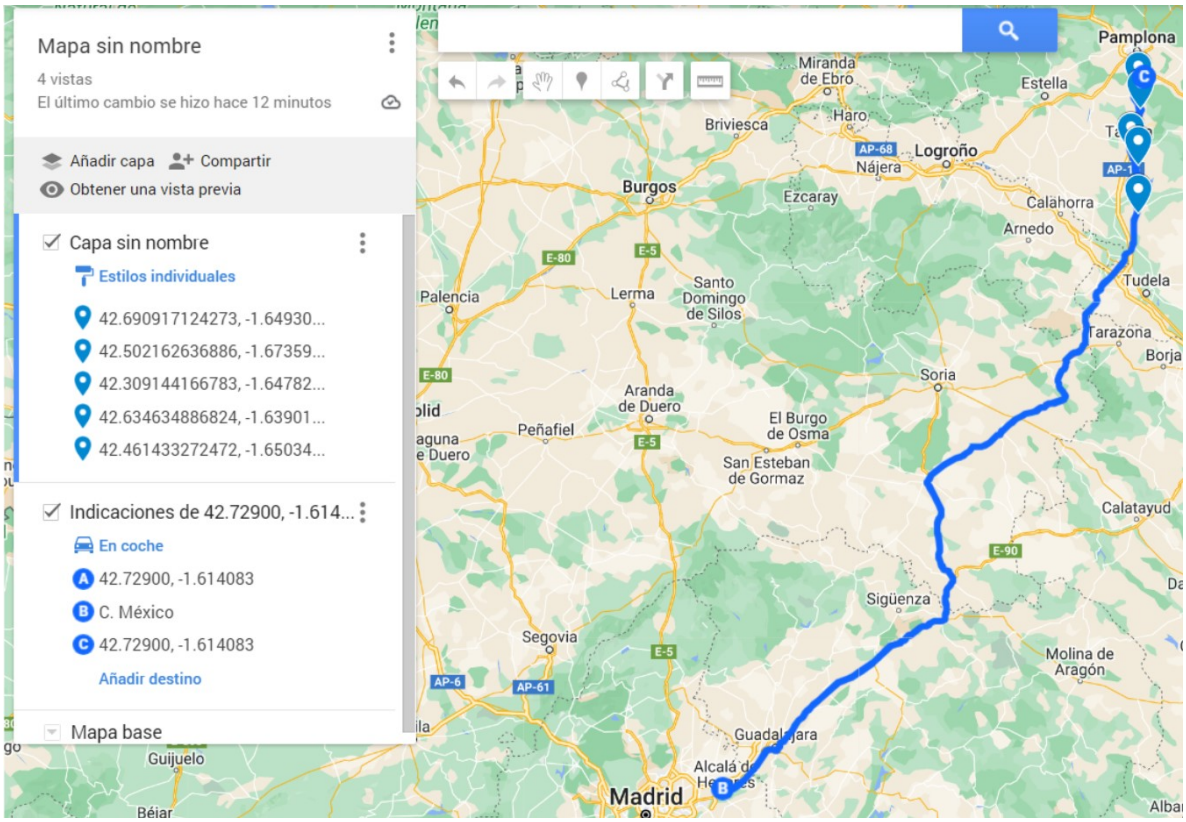


Figura 6: Reconstrucción de la ruta de ejemplo

4. Preparación de los datos

Habiendo entendido la lógica del negocio e identificado el conjunto de tablas donde se almacena la información, el siguiente paso fue definir el conjunto de datos sobre el que se hará el análisis. En algún caso ha sido necesario ampliarlo con más características, justificándose en ese caso el motivo de esa decisión.

A continuación, se relata el proceso de selección de atributos que permitieran caracterizar pedidos, rutas y conductores a partir de las cuales extraer patrones.

Con el objeto de encontrar regularidades en los datos relativos a la definición de la ruta, se necesitaban caracterizar los pedidos por los que surge. El número total de filas en la tabla es de 14337. La siguiente selección de campos se consideró interesante para poder clasificar estos pedidos:

- **Id_Tipo_Pedido**: debería referenciar de alguna forma el tipo de pedido, pero no se cuenta con esta información. ACME no cumplimentó este campo al volcar sus datos del ERP y en todas las instancias es nulo.
- **NroPales**: en este caso hay un 9.68 % de nulos, pero observando los valores reales que llega a tomar se encuentra que en un 0.13 % es 0 y en un 90.17 % es 1. Parece muy raro que se llegue a definir un campo para solo indicar si hay o no palés, además de que se esperaría en un principio que frecuentemente los camiones carguen con varios palés. Por tanto, de nuevo se puede suponer que, por parte de ACME, no se ha optado por informar con detalle este campo al usar la aplicación.
- **Peso**: se da otro caso similar, un 9.68 % de nulos, valor 0 en un 0.04 % y 1 en un 90.26 %. En este caso es incluso más incomprensible que estos datos puedan significar algo, por lo que se cree que la causa es la misma que en el anterior campo.
- **VolumenM3**: hay un 9.69 % de nulos y un 90.31 % de filas en el que se toma valor 0. Se extrae la misma conclusión.
- **TipoMercancia**: hay un 10.18 % de nulos y en el 89.82 % restante se toma valor "MERCANCIA CONCERTADA". En este caso tiene sentido el valor que se introduce, pero al ser siempre el mismo deja de ser útil para intentar usarlo para caracterizar el pedido. También se desconoce si el hecho de que el valor fuese nulo tendría algún significado, pero parece que sigue siendo un problema de falta de definición de requisitos sobre los datos pertinentes a cumplimentar en la aplicación.
- **Gastos**: hay un 9.69 % de nulos y en el 90.31 % restante solo se toma valor 0. Aquí se deberían incluir gastos como gasolina (o eso se podría esperar ya que es libre el cliente de decidir qué considera introducir en este campo), pero parece que deliberadamente desde ACME no se ha cumplimentado ese dato.

Desgraciadamente, ninguno de los atributos que podrían caracterizar al pedido son realmente útiles, ya que ninguno está correctamente informado. Por tanto, se descarta la vía de hacer análisis de datos con características de los pedidos.

Respecto a la tabla Movimientos, almacena información relacionada con el conductor o conductores que se asignan y se pueden tomar esos datos para caracterizarlos en función de las rutas en las que trabajen. La principal función de esta tabla es poder hacer de nexo entre el pedido al que pertenece y la ruta que corresponde a un movimiento. Respecto a los conductores asignados, estos pueden ser uno o dos (que se turnan durante el viaje). ACME asegura que asignar de esta manera a los conductores les permite completar envíos más rápido. Los datos en este aspecto son correctos en el sentido de que siempre se introducen identificadores de conductores válidos, pero hay casos en los que el mismo conductor ha sido registrado en la base de datos con dos identificadores diferentes en la tabla Conductores y ha aparecido como primer y segundo conductor en algún viaje (problema de consistencia). Además, en un principio llamó la atención que hubiese un considerable número de entradas a -1 en el identificador de alguno de los conductores. Tras consultarlo con Fagor, se debe a que puede haber casos en los que se cuente con un conductor contratado que no forme parte de la plantilla usual y en esos casos se le da ese identificador genérico.

En la tabla que relaciona Pedidos y Movimientos, Movimientos_Pedidos, se encontraron instancias extrañas. El diseño del aplicativo ha permitido registrar en la tabla varias parejas del mismo movimiento y pedido, por lo que sería de interés para Fagor corregirlo. Tras analizar el impacto de esta anomalía, se observó que un 31.03 % de las filas eran repetidas, por lo que se descartan esas duplicaciones y se usan las únicas para construir el conjunto de datos.

En la tabla RutasCalculadas, con 21166 filas, se seleccionaron para ser analizados los siguientes atributos:

- **VelocidadUrbana, VelocidadNacional, VelocidadAutopista, VelocidadFerry, EvitarPeaje, EvitarAutopista, OptimizarPuntosRuta**: como se indica en la documentación de la tabla, estos campos no están cumplimentados correctamente, ya que siempre toman valor nulo en el caso de las velocidades o 0 en los campos restantes. Se prescinde de usar estos campos.
- **TiempoTotal**: hay un 42.29 % de nulos, pero el resto de valores son muy variados y concuerdan con lo que se podría esperar en la realidad.
- **DistanciaTotal**: exactamente la misma proporción de nulos y valores que se pueden considerar reales que con el campo TiempoTotal.

De nuevo, estos datos tampoco pueden ser utilizados en el análisis, salvo los campos de TiempoTotal y DistanciaTotal. Aún así, más adelante se comprueba que para el conjunto de rutas que se pueden relacionar con un pedido siempre son nulos, por lo que se excluyen del análisis.

En la tabla RutasCalculadasAsignacionVehiculos, con 8957 filas, se registran datos sobre qué vehículo y caja se emplearon en la ruta, pero, dado que no se puede clasificar la mercancía que se transporta, no es interesante conocer las características del vehículo. No se dispone de información sobre los vehículos más allá de su matrícula y fecha de registro en la empresa. Por otro lado, hay información que se podría utilizar: las fechas de inicio y fin reales tras finalizar la ruta y las que se establecieron por una persona cuando se definió a modo de estimación. Las fechas estimadas están bien informadas en general y el número de entradas nulas es despreciable, siendo un 0.01 % respecto al total, tanto para la de inicio como la de

final. En cuanto a las fechas reales que se introducen, no tienen la misma calidad. En las fechas de inicio hay un 20.92 % de nulos y las de fin están cumplimentadas en bastantes más casos, con solo un 0.84 % de nulos. Por tanto, se usaron estos campos en el análisis de datos con excepción de esos casos que no lo permitan por valores nulos. También se consideraron los márgenes de tiempo en los que se da por finalizada de forma automática la ruta desde el tiempo de fin estimado, ya que así se podría estudiar si hay algún factor que provoque estos cierres por retraso de forma recurrente. En consecuencia, se añaden al *data set*.

La tabla RutasCalculadasPuntos, con 79851 registros todos ellos válidos, no da ningún problema, con la excepción de tres rutas de dos puntos en las que no se llega a actualizar el campo de distancia acumulada y permanece como nulo. Por tanto, se incluyen los campos de esta tabla en el *data set* base de trabajo.

Por último, a la hora de comprobar las tablas mensuales de eventos de ruta, se hizo un pequeño análisis sobre los datos. En concreto, se consideró interesante contabilizar y estudiar las salidas y vueltas a ruta, lo que podría ser otra fuente para encontrar patrones en rutas. Además, esta información podría servir para conocer si se toman caminos alternativos respecto a las rutas planeadas, ya que no se tiene constancia, de otra forma, de si los conductores hacen algún cambio de planes y toman rutas diferentes por algún motivo. Los datos de la tabla en sí están correctamente informados, considerando cada fila independientemente. Sin embargo, se encontraron varios casos en los que había, por ejemplo, dos salidas de ruta seguidas. Tras consultarlo con Fagor, también consideraron que esto no debería ser posible sin antes registrar una vuelta a la ruta y no se pudo dar una explicación. Aún así, tras hacer un análisis observando la media y extremos de tiempo y distancia, como se puede ver en la Figura 7, aparentemente no tienen sentido algunos datos.

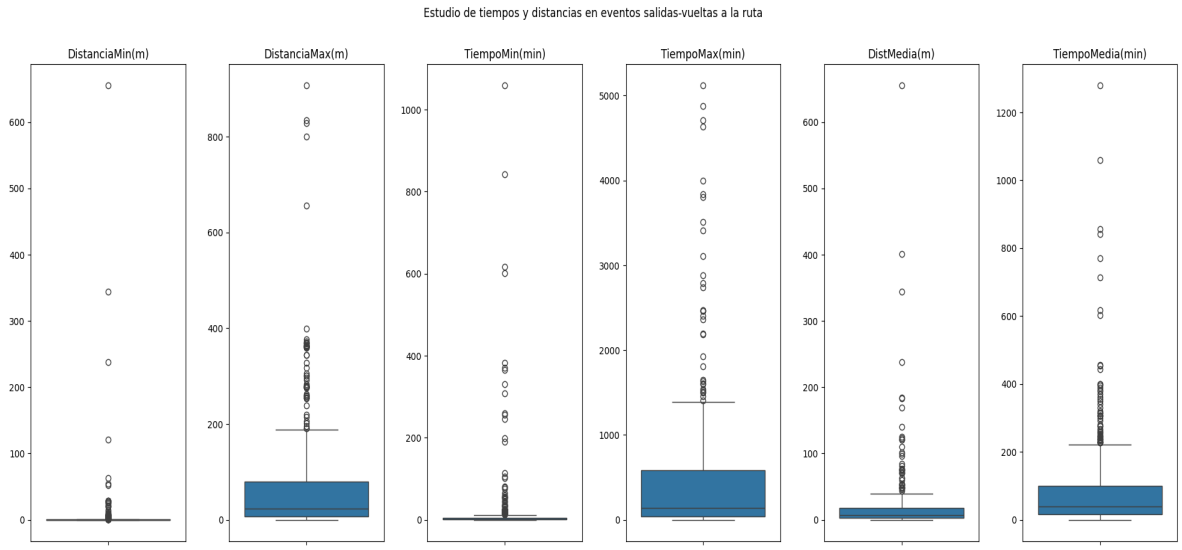


Figura 7: Estudio de tiempos y distancias que suponen bloques de salida-vuelta a una ruta (mes de diciembre). DistanciaMin representa la menor distancia de una salida-vuelta a ruta en cada una. DistanciaMax registra las de mayor distancia por ruta y DistMedia la media de todas las salidas y vueltas al camino por ruta. TiempoMin, TiempoMax y TiempoMedia miden para esos mismos casos el tiempo que ha habido entre la salida y la vuelta en minutos.

Lo que más llama la atención es que el extremo superior de tiempo entre salida y vuelta de ruta haya supuesto varias horas mientras que la máxima distancia registrada no alcance ni un kilómetro. Viendo que en los eventos hay más casos raros, como dos eventos de parada registrados estando exactamente en el mismo punto, se cree que debe haber algún fallo en el que las paradas se registren incorrectamente o ni lo hagan (lo que explicaría que pasase tanto tiempo para volver a la ruta sin haber recorrido tanta distancia). Otro motivo por el que se descarta usar las salidas y vueltas a ruta en los análisis es la sensibilidad con la que se registran. Se encontraron casos en los que en un tramo en el que no había ningún desvío ni ninguna gasolinera se registró una salida y vuelta a ruta en pocos metros. Las salidas y vueltas a ruta suelen ser de distancia corta, como se ve en la Figura 7. Tras consultarlo con Fagor, este caso extraño resultó deberse a la sensibilidad con la que se considera una salida de ruta. Al parecer, se diseñó de esa forma para poder detectar una salida a la gasolinera o pequeños desvíos de ese estilo en los que el vehículo siga localizado muy próximo al camino de la ruta establecido. Esto tiene un efecto negativo en el registro usual de eventos, ya que una pequeña fluctuación en el registro de la posición del camión puede provocar un evento erróneo de salida de ruta.

Una vez finalizado este proceso de selección de atributos interesantes, se extraen conclusiones importantes. Se observó que muchos campos no se tienen suficientemente bien informados y otros carecen de valor. Además, en este proceso de interpretación de los datos hubo dificultades, ya que solo se pudo consultar a Fagor y no a la empresa original que proporciona la información a la base de datos, lo que supuso un importante obstáculo para el avance en los objetivos iniciales del proyecto.

A continuación, se describe el *data set* de trabajo base, que se compone de filas que relacionan las instancias de las tablas Pedidos, Movimientos, RutasCalculadas y RutasCalculadasPuntos, siguiendo el diagrama que se puede ver en la Figura 5. Cada fila contendrá en los primeros campos la información del pedido extraída de su tabla. En el caso de que se necesitasen más de un movimiento para entregar el pedido y, por tanto, más de una ruta, se duplican estas filas cambiando la información acerca de movimientos y rutas. A su vez, también se registra en diferentes filas la información de cada punto de la ruta. Para describir esta estructura de forma gráfica se elaboró el esquema que se muestra en la Figura 8.

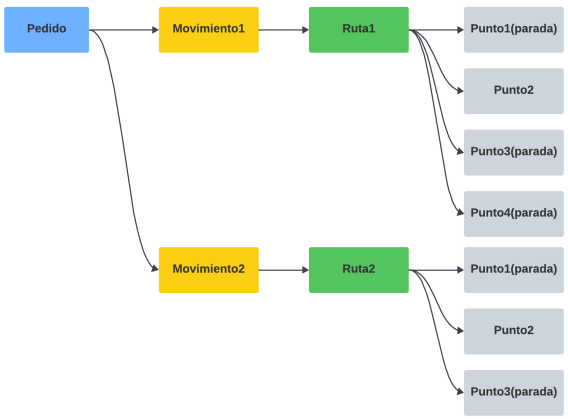


Figura 8: Esquema de la información que se almacena en el *data set* base de trabajo.

En este ejemplo un pedido requiere dos movimientos para llegar al destino. Una ruta tiene cuatro puntos definidos y la otra tres, en total se registran siete filas.

Tras conseguir el conjunto de datos, lo primero que llama la atención es la reducción en número de pedidos y rutas con las que se cuenta. La tabla de pedidos en un principio contaba con alrededor de 14000 instancias, mientras que ahora son algo más de 7649 y en la de rutas guardadas se pasa de alrededor de 21000 a 8694. En cuanto a las rutas, es entendible que pueda haber muchas más de las que se usan, ya que pueden definirse por sí mismas sin asignarse a pedidos en la aplicación. Por otro lado, se entiende que muchos pedidos no fueron efectuados o no se llevó un registro de los movimientos que supusieron, por lo que al no poder relacionarse con el resto de tablas quedarían muchos descartados. Aún con esta reducción, los datos utilizados en este estudio abarcan desde junio de 2023 hasta abril de 2024 y son suficientes para poder realizar un análisis. Este rango de tiempo se debe a que ACME comenzó a usar Flotasnet en junio de 2023, por lo que no existen registros de pedidos anteriores a esa fecha. En cuanto al resto de los datos, como el personal de conductores y la flota de vehículos, son un registro histórico de la empresa y no se limitan a ese rango temporal.

Agrupar los datos de las tablas provocó otro problema en relación con los campos de DistanciaTotal y TiempoTotal de la tabla RutasCalculadas. No se conoce la explicación, pero en todo el conjunto de rutas en el que se puede relacionar con el resto de tablas, esos campos son nulos. Se necesitaba de alguna manera conocer la longitud de las rutas, ya que medirlo de otra forma como la distancia geográfica entre origen y destino no sería lo suficientemente preciso. Se eligió recurrir al campo Distancia de la tabla RutasCalculadasPuntos, que con cada punto de parada actualiza el valor que se introduce siendo la distancia acumulada total en metros desde el inicio de la ruta. Por tanto, si de cada ruta se consulta la última parada, se podría extraer esta información. Se elige definir un nuevo atributo DistanciaReal_km para cada ruta. La definición de la distancia pasa de metros a kilómetros por conveniencia en los cálculos, ya que la inmensa mayoría de rutas están en el orden de decenas o cientos de kilómetros. En cuanto al tiempo, se buscaría de la misma manera, pero hay que tener en cuenta que este tiempo no hace referencia al que ha llevado en la realidad llegar desde el inicio de la ruta a una parada determinada, sino la estimación que daría Google Maps.

El *data set* final que se construye cuenta con 34 campos que resumen la información útil de cada tabla. Referente a Pedidos, se recogen los campos Id_Pedido, NPedido, Estado, Id_Marca_Ini y Id_Marca_Fin. De la tabla de Movimientos se seleccionan los campos Id_Movimiento, CodigoViaje, Id_Vehiculo, Matricula, Id_Conductor, Nombre_Conductor, Apellido1_Conductor, Apellido2_Conductor, Id_Conductor2, Nombre_Conductor2, Apellido1 _Conductor2, Apellido2_Conductor2 y Estado_Movimiento. En relación con la información sobre rutas, se recogen los campos IdRuta y Nombre de RutasCalculadas y los campos fecha_desde, fecha_hasta, Fecha_Inicio_Real, Fecha_Fin_Real y MargenSuperior de RutasCalculadasAsignacionVehiculos.

4.1. Discretización de atributos

En esta subsección se detalla cómo se usaron las técnicas de minería de datos *clustering* y árboles de decisión para discretizar los atributos y obtener así nuevas propiedades que usar posteriormente en la construcción y análisis de reglas de asociación.

4.1.1. Clustering

Para dividir diferentes atributos en rangos se eligió aplicar el algoritmo *k-means* en todos aquellos atributos numéricos que se usasen en las reglas de asociación.

Como ejemplo ilustrativo del proceso que se siguió en cada caso se describirá cómo se hicieron las divisiones para el número de paradas de las rutas. Como se detalló en la Sección 2, primero hay que seleccionar el número óptimo de divisiones. Para ello, se usa el método del codo. La curva que resulta al aplicar *k-means* con diferente número de *clusters* se puede ver en la Figura 9.

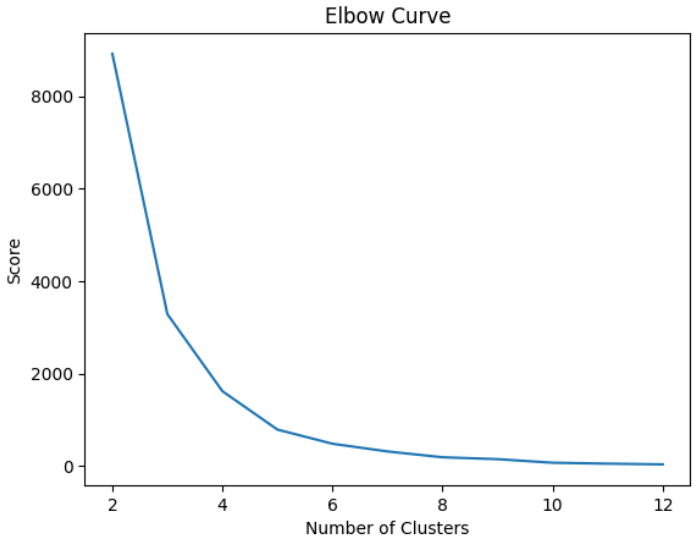


Figura 9: *Elbow curve* para hacer *clustering* con el número de paradas definidas en rutas.

En la gráfica, el eje y representa el resultado de aplicar el método score() de *sklearn* para el número de *clusters* indicado en el eje x. Se aprecia que aumentando el número de grupos se minimiza la suma de las distancias de los puntos a sus centroides al cuadrado. Se decide usar tres divisiones, ya que se puede considerar que el codo está en tres o cuatro y es preferible no usar tantos rangos para obtener conclusiones más generales en las reglas de asociación.

Para este atributo, se decidió dividir en los siguientes rangos: pocas paradas (una, dos o tres), bastantes paradas (entre cuatro y seis) y muchas paradas (al menos siete). Siguiendo este mismo proceso se derivaron nuevos atributos para el resto de propiedades que se usaron en las reglas de asociación.

4.1.2. Árboles de decisión

El uso que se le dio a la creación de un árbol de decisión clasificador fue tener una referencia visual de la división en rangos de las distancias de las rutas. Dado que no se construye con la finalidad de crear un modelo predictor funcional, las divisiones de entrenamiento y test son menos relevantes y se toma una proporción de 80 % y 20 %, respectivamente. Este árbol se creó durante la fase en la que se estudiaban las fuentes de información para conocer el tiempo de las rutas, por lo que los atributos que lo constituyen son DistanciaReal_km, TiempoReal_h,

TiempoEstimadoSis_h y TiempoEstimadoPersona_h. En resumen, estos atributos de tiempo representan la duración de las rutas según la estimación de Google, la del controlador de tráfico y la que se obtiene de las fechas de inicio y final reales una vez se ejecuta la ruta. El análisis más profundo sobre estas propiedades se detalla en la Sección 5. El árbol se definió con objeto de clasificar las rutas según si se empleaban uno o dos conductores. El resultado de la construcción de este modelo (estableciendo una profundidad máxima de cuatro niveles) se ve en la Figura 10.

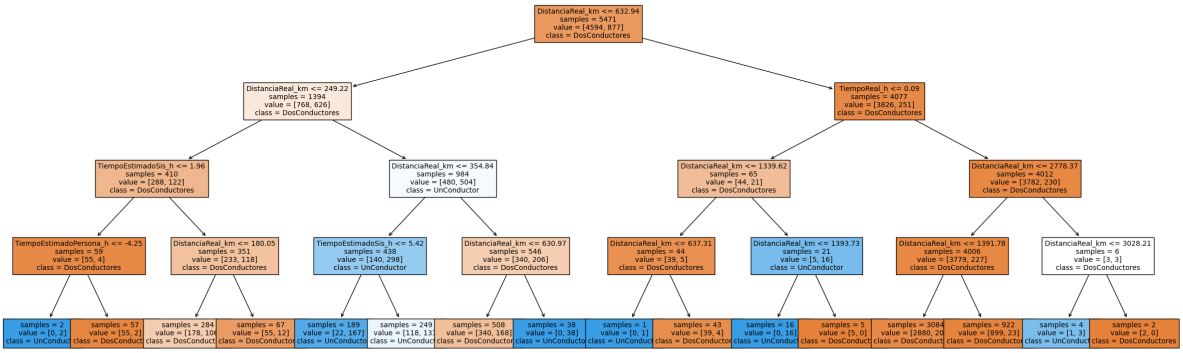


Figura 10: Árbol de decisión para estudiar el impacto del atributo DistanciaReal_km

Para el uso que se le quiere dar al árbol interesa observar los primeros niveles. Comparando las divisiones de rangos del atributo con las obtenidas mediante *clustering*, varían ligeramente en 100 kilómetros, pero esta nueva asignación resultó ser más adecuada en la construcción de reglas de asociación, contando con una mayor variedad de estas. Estos rangos finales fueron: DistanciaCorta (menor o igual a 630), DistanciaMedia (mayor que 630 y y menor o igual que 1375) y DistanciaLarga (mayor que 1375). En estos nuevos rangos se distribuyen de manera más equitativa las rutas.

5. Análisis, extracción de conocimiento y minería de datos

En esta sección se detalla todo el trabajo de análisis que se ha hecho sobre los datos, además de cómo se aplicó la minería de datos para extraer nuevo conocimiento. El objetivo es analizar los datos de forma que se puedan deducir patrones en estos y verificar que la lógica de negocio se ve reflejada correctamente en la información que se puede obtener de la base de datos. En las subsecciones siguientes se organizan las distintas propuestas analíticas que se han realizado.

5.1. Caracterización de rutas por día de la semana

Se quiso contrastar si el día de la semana en el que se lleva a cabo una ruta se debe a algún motivo particular. Se observa, para ello, la frecuencia con la que se comienzan o terminan rutas en cada día de la semana. En un principio, se toma la fecha de inicio o final de una ruta por el evento de inicio o final de esta en la tabla de eventos de ruta; sin embargo, sigue habiendo casos extraños donde, por ejemplo, no se llega ni a notificar el evento de inicio de ruta. Por tanto, por contar con el mayor número de datos, se eligió tomar como referencia las fechas de inicio y fin estimadas por la torre de control. Los resultados se pueden ver en la Figura 11. La primera conclusión que se puede extraer es que, como se podría esperar, en los fines de semana hay menos actividad. Por lo general, en el resto de días de la semana no se aprecia una diferencia lo suficientemente sustancial como para poder darle un significado. Que haya bastantes más rutas que finalizan un sábado frente a las que comienzan se debe a que alguna ruta iniciada el viernes o un día anterior termine durando hasta el sábado, ya que ACME frecuentemente hace rutas largas incluso internacionales. También explica que el lunes se finalicen menos rutas comparado con el resto de días entre semana, ya que principalmente serían rutas que han comenzado y finalizado ese mismo lunes, al ser poco frecuente tener una ruta iniciada el fin de semana.

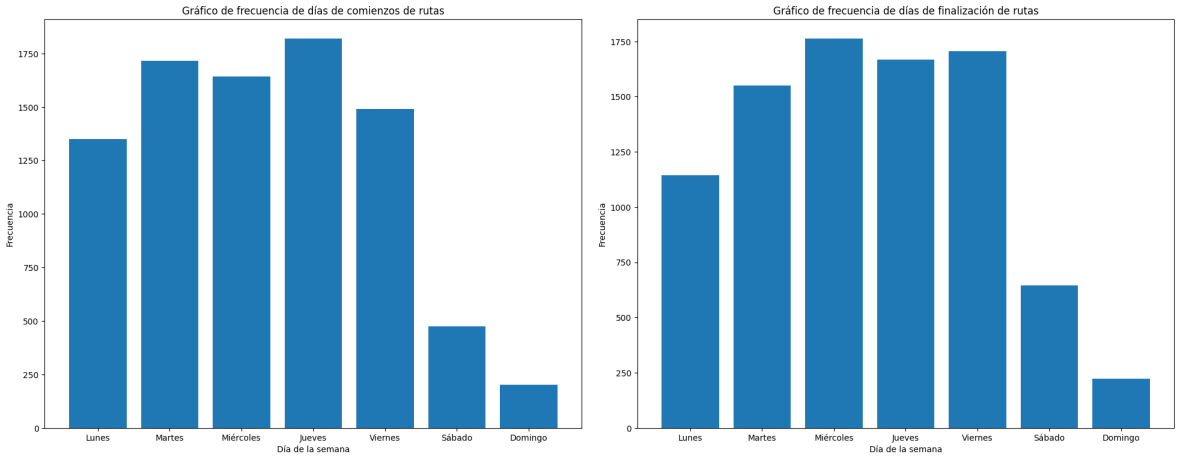


Figura 11: Frecuencias en las que se comienzan y terminan rutas por cada día de la semana

También se quiso comprobar (ver en la Figura 12) si la distancia a recorrer podría tener impacto en la decisión de cuándo asignar la ruta. Pero, en vista de los resultados, se descarta

esa hipótesis, ya que no se aprecia una diferencia significativa según el día de la semana.

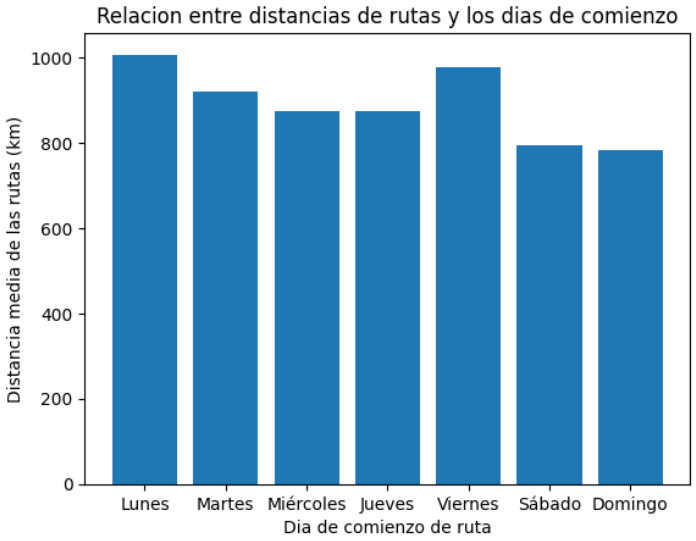


Figura 12: Distancia media de rutas que comienzan en cada día de la semana

5.2. Pedidos "vacíos" y "no vacíos"

Analizando los pedidos del conjunto de trabajo, se encontraron varios casos de pedidos que tenían el mismo nombre que otros, con la diferencia de que finalizaban con "_vacio01". Por ejemplo, dos pedidos con nombres PT23015609_10000 y PT23015609_10000_vacio1. Tras consultarlo con Fagor, solo se pudo especular el significado que podría tener esto. Una posible explicación que se le dio en un primer momento fue que estos pedidos "vacíos" fuesen viajes después de entregar un producto y que la empresa ACME necesitase que quedasen registrados por algún motivo. Para conocer la naturaleza de estos, se comparó el subconjunto de pedidos que tuviesen una versión normal y su respectiva versión "vacía". Se intentó conocer cuán similares eran estos pedidos, observando cuántos movimientos (rutas ejecutadas) fueron necesarios para completar el pedido.

En la Figura 13 se puede ver que los pedidos "vacíos" definen caminos más simples a primera vista ya que muy raramente se toma más de una ruta, con su posible cambio de conductor/vehículo/caja.

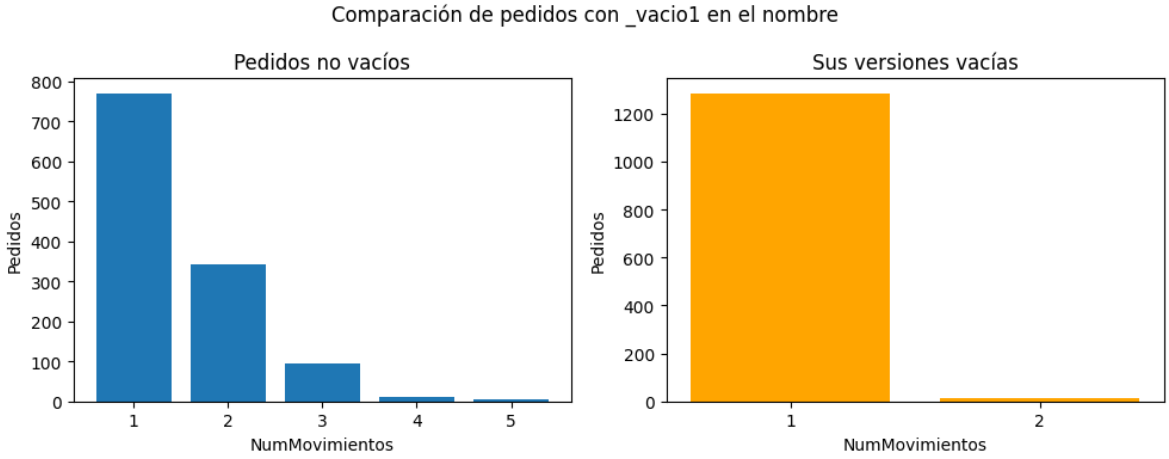


Figura 13: Comparación de número de movimientos que se emplean en hacer pedidos frente a sus versiones "vacías"

Otra observación que se hizo es que realmente esos pedidos sí que llevaban una carga, ya que si observamos el número de paradas que hacen en la Figura 14, frecuentemente son más de dos. Se puede afirmar con seguridad que esto significa llevar una carga, ya que se comprobó que todas las paradas tenían un nombre de entre los siguientes: 'DESCARGA_ADIC', 'INICIO', 'DESCARGA_TRANS', 'CARGA_TRANS', 'CARGA_ADIC', 'DESENGANCHE', 'DESCARGA_FINAL', 'ENGANCHE', 'CARGA_INI' y 'ENGANCHE_REM'. Todos estos hacen referencia al enganche inicial y final de la caja del vehículo y a cargas y descargas. En un momento se creyó que se podrían registrar paradas, por ejemplo, de descanso reglamentarias como marca la ley, pero esa información no se ve reflejada en la base de datos al definir paradas en las rutas. Solo un 0.47 % de todas las paradas registradas en la base de datos no se encontraban entre esas tareas y además su nombre tomaba valor nulo o vacío. Por tanto, esas excepciones se consideran despreciables y se toma como norma que una parada, además de la de inicio y final, significará cargar o descargar mercancía en algún lugar.

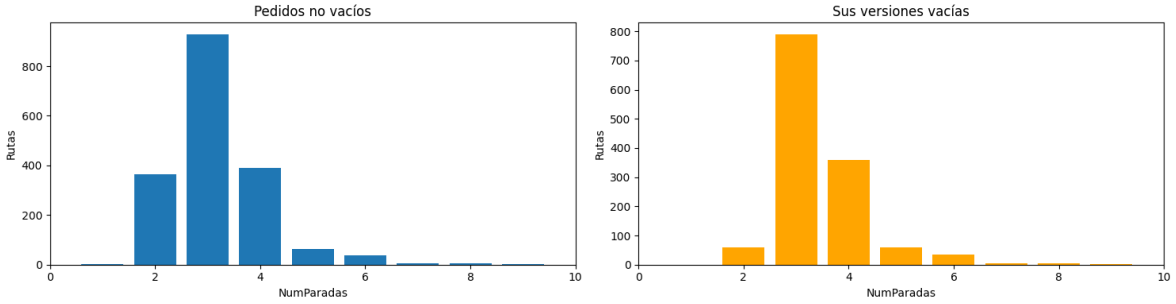


Figura 14: Relación entre el número de paradas ejecutadas y el número de rutas con ese número de paradas, para pedidos "vacíos" y "no vacíos"

De forma global no se puede extraer una conclusión sobre la relación que pueden tener estos dos tipos de pedidos, por lo que se buscaron ejemplos individuales para encontrar alguna

similitud. Previamente, se había descartado que ambos tipos representasen exactamente el mismo camino, pero observando las parejas de pedido "vacío" y "no vacío" se observó que, por lo general, los vacíos siguen una ruta o dos que forma parte del pedido normal. No solo el camino era el mismo, sino que en la propia base de datos tenían el mismo identificador de movimiento y ruta. En el 96.64 % de los casos (entre las 1180 parejas de pedidos con dos variantes) el conjunto de rutas de los pedidos "vacíos" forman parte de las rutas ejecutadas para sus contrapartes. Entre los pedidos que no cumplen esta propiedad no se ha encontrado una relación o un patrón que se repita.

En conclusión, se interpreta que estos pedidos son definidos por parte de ACME para marcar partes del viaje que se hace al hacer la entrega. El motivo por el que se hace esta práctica se desconoce. Es extraño que en los casos que tanto el pedido original como el que se marca como "vacío" se entregan en una sola ruta sea necesario hacer la distinción, ya que nada los diferencia.

5.3. Extraer información de las direcciones de lugares

Para conocer los lugares de origen y destino, así como los de paradas intermedias, de las rutas que se toman solo se dispone de la información de la tabla Marcadores. En RutasCalculadasPuntos hay un campo Marca que hace referencia a la tabla previamente mencionada. Aquí se almacenan direcciones completas, por ejemplo, de algún almacén. Se consideró interesante poder contar con información más general de estos lugares como país o localidad, de forma que se pudiese usar en el análisis. Por tanto, se aplicaron una serie de expresiones regulares para capturar la información que se deseaba. Para ello se añadió al *data set* de trabajo los datos de la tabla Marcadores en aquellas filas en las que el punto de la ruta referenciase a un lugar. El primer problema que se encontró fue que las direcciones se escribían de diferentes formas con más o menos información. Un pequeño resumen de los tipos que se encontraron fueron direcciones similares a estas:

- FRET 6, RUE DU PAVE, B6 MODULE DOORES 25 TO 31 FR-93290 Tremblay-en-France

  Las direcciones que siguen este patrón comienzan por la dirección del lugar con el nombre completo, seguido de dos caracteres que indican el código del país y lo que se entiende que es el código postal. Siempre seguido de esto, aparece el nombre de la localidad. Existen variaciones, como países donde el código se ha expresado con guiones entre dígitos.

- Rue de la chaudière, 78730 Saint-Arnoult-en-Yvelines, Francia

  Estas direcciones son similares, con la diferencia de que el país no viene indicado junto al código, sino explícitamente al final. El resto de patrones se siguen manteniendo, estando la localidad inmediatamente después del código.

- ES-25250002

  En algún caso extraño el nombre de la dirección ha terminado siendo una copia del nombre que tiene ese lugar en la base de datos. En ocasiones, viene precedido del nombre de la localidad, pero en muchos otros solo se cuenta con los caracteres del país y el código.

Este último se cree que es una fusión del código postal (primeros cinco dígitos) y otros tres dígitos que podrían tener un significado interno para ACME, como, por ejemplo, contabilizar el lugar dentro de la localidad con ese código postal. Aún así se encontraron casos donde este código no hacía referencia en ninguna parte a algún código postal, por lo que se decidió prescindir de esta información.

Una vez se definieron expresiones regulares capaces de capturar todos estos casos, se consiguió conocer la localidad y país de todos los lugares, con excepción de aquellos en los que fuese imposible conocer la localidad (como el último tipo descrito previamente) o aquellos en los que el campo que indica la dirección valiese nulo. El impacto que tuvieron las direcciones escritas como el tercer caso no fue muy grande, solo un 3.28 % de todas las direcciones analizadas fueron de ese estilo. Por tanto, se puede prescindir de ellas sin que afecte demasiado en el análisis.

Por otro lado, se observó que en algunos casos se nombran las mismas localidades de maneras diferentes, según los formatos de dirección que se han mencionado previamente. Por ejemplo, la localidad francesa Franqueses del Valle en ocasiones se escribe como Les Franqueses del Vallés. El número de inconsistencias de este estilo que se encontraron no superaba los once, por lo que para esos casos concretos se modificaron los valores del nuevo campo de localidad del *data set* para que el nombre fuese la opción que más se repite en la base de datos.

Una vez se dispone de esta información, es interesante evaluar qué localidades eran muy frecuentadas por conductores al comenzar rutas. Se muestra una gráfica general en la Figura 15.

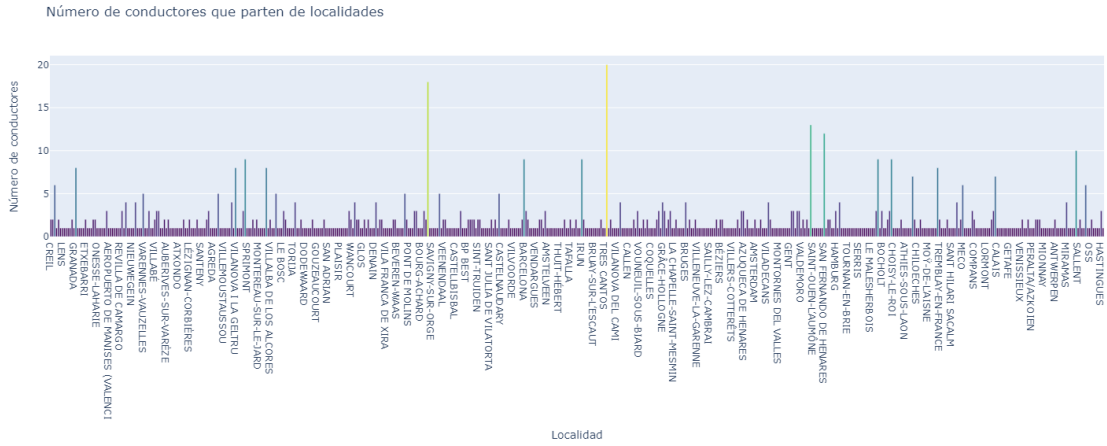


Figura 15: Gráfica de número de conductores que parten de localidades

Es difícil apreciar en detalle cada localidad y no hay espacio para que estén listadas todas en la parte inferior, pero esta gráfica da una imagen global de lo que es la norma para las localidades. La mayoría de localidades han sido origen de ruta para un solo conductor concreto y el siguiente grupo más grande no suele contar con más de cinco. Entre las pocas que pasan de este margen encontramos ciudades grandes en las que tiene sentido que haya más conductores, como Madrid, Barcelona, Alcalá de Henares o Colonia en Alemania. Los dos repuntes que se

aprecian a simple vista y destacan por encima del resto son Imarcoain en Navarra (que puede ser explicado por situarse allí las oficinas de ACME en España) con 18 conductores diferentes que en algún momento han iniciado una ruta en ese lugar y Saint-Pierre-des-Corps, una localidad en la zona central de Francia.

Si se toma otro enfoque, se puede conocer también desde cuántas localidades han llegado a iniciar sus rutas los conductores de la empresa, como se indica en la gráfica de la Figura 16. Desde esta perspectiva, es apreciable que gran parte de los conductores inician rutas desde varios lugares, estando en el orden de 30-45 localidades muchos de ellos.

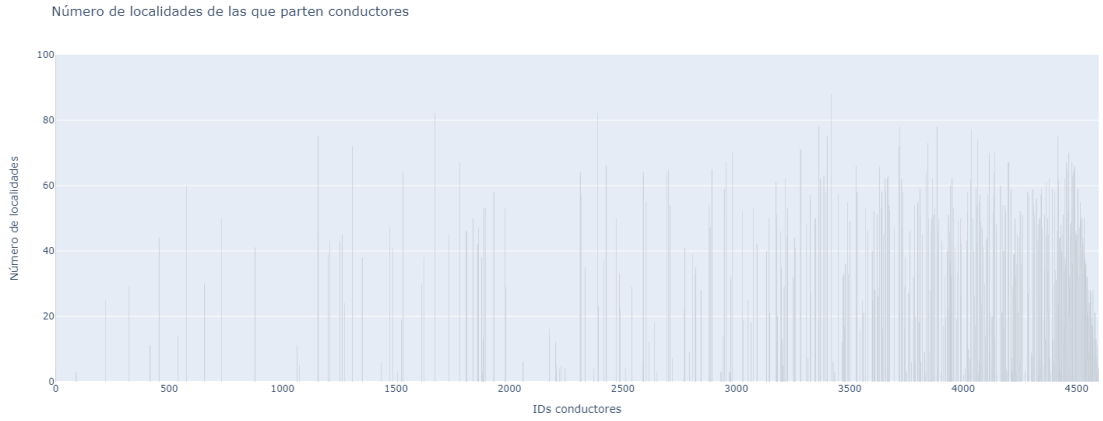


Figura 16: Gráfica de número de localidades de las que parten los conductores al iniciar rutas

Debido al gran número de conductores, al dibujar la gráfica de la Figura 16 se difuminan las barras. Para observar los datos con mayor detalle, se tomó una muestra más pequeña (haciendo uso del zoom que permite la librería plotly) y el resultado se muestra en la Figura 17.

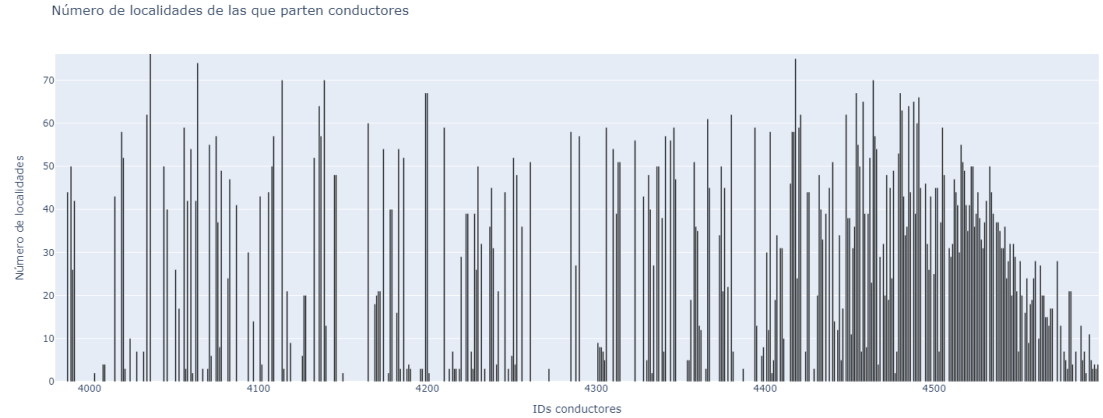


Figura 17: Gráfica de número de localidades de las que parten los conductores al iniciar rutas (muestra de 500 conductores)

Para analizar de forma global estas cantidades se calculan los percentiles, mostrados en la Tabla 7.

Tabla 7: Percentiles del número de localidades de las que parten conductores

Percentil	0.2	0.4	0.6	0.8	1
Número de localidades	9	28	42	53	140

El cálculo de los percentiles confirma que la gran mayoría de conductores han visitado entre 28 y 53 localidades para comenzar las rutas en las que han trabajado.

Otro hecho que se puede confirmar teniendo la información de los países es la cantidad de rutas que empiezan y acaban en uno diferente. Tras contabilizar los casos, se encuentra que en 5260 rutas se pasa de un país a otro diferente, mientras que en las 3434, tanto origen como destino se encuentran en el mismo país. Encaja con la noción que se tiene previamente de la actividad de ACME, es decir, transportes de larga distancia, a menudo internacionales.

5.4. Comparación de fuentes de duración de rutas

Como se ha descrito previamente, al explicar los campos de las tablas, existen tres fuentes de información para conocer la duración de una ruta. En primer lugar, está el tiempo estimado por Google, que viene dado en la tabla RutasCalculadasPuntos en el campo Tiempo. Si se toma ese tiempo actualizado en el punto final de la ruta, obtenemos ese tiempo de estimación, aunque hay que aclarar que es genérico y seguramente está pensado para coches y no grandes vehículos como camiones. La segunda fuente proviene de RutasCalculadasAsginaciónVehículos. Aquí hay dos campos fecha_desde y fecha_hasta, cuyos valores son definidos por una persona manualmente estimando a partir de la fecha de inicio esperada cuánto durará el viaje. Tras consultar con Fagor el criterio por el que se hace esta estimación, se nos informó de que vendría influenciada por una que podría dar Google sumándole un tiempo extra en función del tipo de vehículo que fuera. Finalmente, también proveniente de la tabla RutasCalculadasAsginaciónVehículos, encontramos dos campos Fecha_Inicio_Real y Fecha_Fin_Real, que han sido introducidos por una persona y deberían representar lo que más se acerca a la realidad. Aún así, Fagor indicó que el campo de MargenSuperior definido para las rutas indica un número de minutos que, sumado dos veces a la fecha de fin estimada, sirven para cerrar automáticamente la ruta (cambiando la fecha de fin real).

Primero se comenzó comparando estos tres tiempos, como se puede ver en la Figura 18. La comparación no puede ser del todo exacta, ya que no se cuenta con el mismo número de datos en cada fuente, como se indicó previamente. Por ejemplo, hay bastantes fechas de inicio reales con valor nulo que impiden calcular el tiempo de la ruta de esa forma. En la figura mencionada, lo primero que llama la atención son los tiempos negativos en las fechas estimadas por una persona. Esto se debe a que, erróneamente, se han permitido introducir fechas de inicio y fin en las que el final es anterior al inicio. En cuanto a las otras dos fuentes, a simple vista no se puede sacar una conclusión clara.

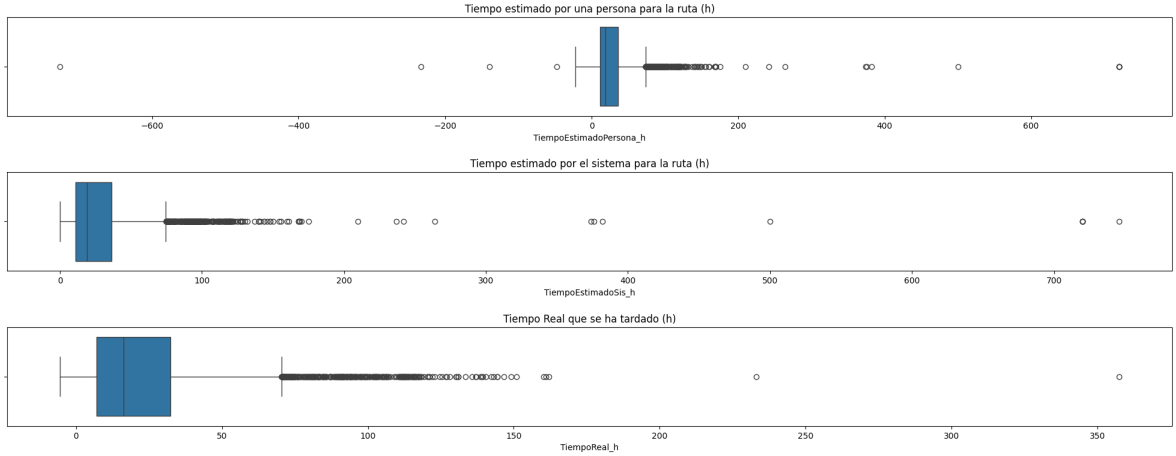


Figura 18: Diagramas de cajas para las duraciones de las rutas según las tres fuentes disponibles

Al comparar los tiempos estimados por el sistema y por la persona, se pensaría en un principio que la persona añadiese el tiempo extra antes mencionado para adaptarlo a un camión, pero resultó que esa suposición no se cumple, ya que se comprobó que en un 91 % de los casos el tiempo de duración estimado por ambas partes es exactamente el mismo. Por tanto, se puede suponer que esta es la manera frecuente en la que trabajan. Además, se observó que en todos los casos en los que se registró que Google estimaba la duración de la ruta como cero, las fechas estimadas que introdujeron las personas (campos fecha_desde y fecha_hasta) coincidieron exactamente en inicio y fin o bien se introdujo una fecha de inicio posterior a la de fin. Se desconoce la causa de estas irregularidades.

Se pensó en comparar por separado ambos tiempos estimados con el tiempo definido como real, pero en vista de que no se diferencian demasiado, solo se trabajó con el estimado por Google. Al comparar ambos se dibujó un histograma que representase la diferencia entre tiempo estimado por el sistema y real, como se ve en la Figura 19. Llama la atención que la diferencia llegue a ser tan grande. Esto, además, contradice una suposición que se tenía sobre el funcionamiento del negocio en la que se cerraba automáticamente una ruta pasado dos veces el tiempo de margen. Por tanto, se descarta la vía de análisis buscando patrones en cierres de ruta por parte del sistema.

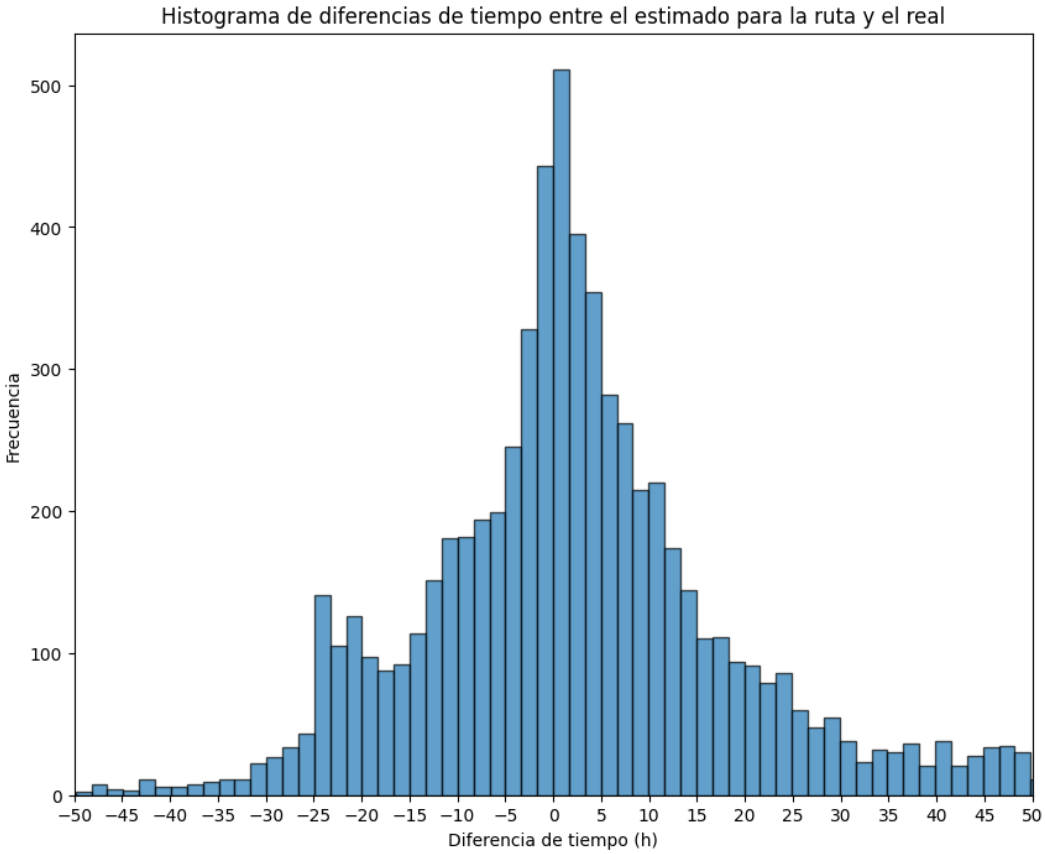


Figura 19: Histograma de diferencias de tiempo entre el estimado para la ruta por Google y el real

5.5. Análisis de clientes de los pedidos

Buscando más fuentes de información para la minería de datos, se consideró interesante analizar los clientes de los pedidos. En la base de datos estos están almacenados en la tabla Grupos_Pedidos y se relacionan con los pedidos a través de la tabla Pedidos_Grupos, por lo que se pueden añadir estos atributos al *data set* con la información de localidades extraída de las direcciones, añadiendo en cada fila de un pedido el cliente correspondiente. Se consideró interesante la posibilidad de mostrar visualmente las rutas que se toman en los pedidos de ciertos clientes. El objetivo que se propuso fue poder elegir un cliente concreto y dibujar un grafo en el que los nodos fuesen localidades y estuviesen posicionados según su localización geográfica. Las conexiones entre estos nodos representan rutas que se han tomado entre esas localidades para llevar pedidos del cliente mencionado. Además, se elige representar el tamaño de los nodos proporcionalmente al número de veces que han sido visitados. Una muestra puede ser el cliente Inditex y el grafo resultado sería el de la Figura 20. En la parte inferior izquierda, los nodos se corresponden con localidades de España. Con pesos 33 y 28 se sitúan las ciudades de Meco e Imarcoain coloreados de un verde menos oscuro y de mayor tamaño que los otros nodos. El nodo amarillo es Zaragoza y tiene un peso de 64 (se eligió que el tamaño

de los nodos creciese en escala logarítmica para que el dibujo fuese más legible). En la parte superior derecha encontramos localidades de Francia, siendo la más frecuentada la situada más al noreste, Lelystad. Los grafos seleccionando varios clientes son una fuente de información valiosa que puede ser útil a la hora de decidir asignar rutas. Por ejemplo, al comparar varios clientes a los que se quiera entregar pedidos, podrían conocerse rutas frecuentes en común y así aprovechar un mismo viaje para todos estos. El inconveniente es que, para clientes que visitan un gran número de localidades, podría ser más difícil interpretar los grafos de manera visual si estas no están demasiado distantes entre sí.

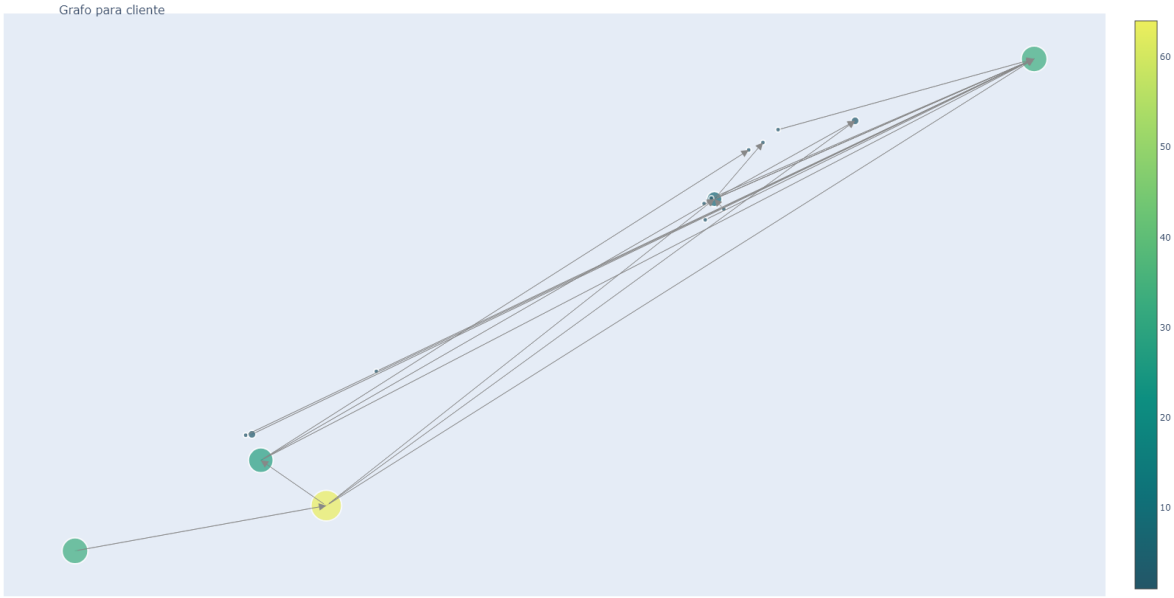


Figura 20: Grafo de rutas empleadas para llevar pedidos de la empresa Inditex

5.6. Reglas de asociación

Para sacar conclusiones interesantes y encontrar patrones en las asignaciones de conductores y rutas se definieron varios conjuntos de atributos entre los que construir reglas de asociación.

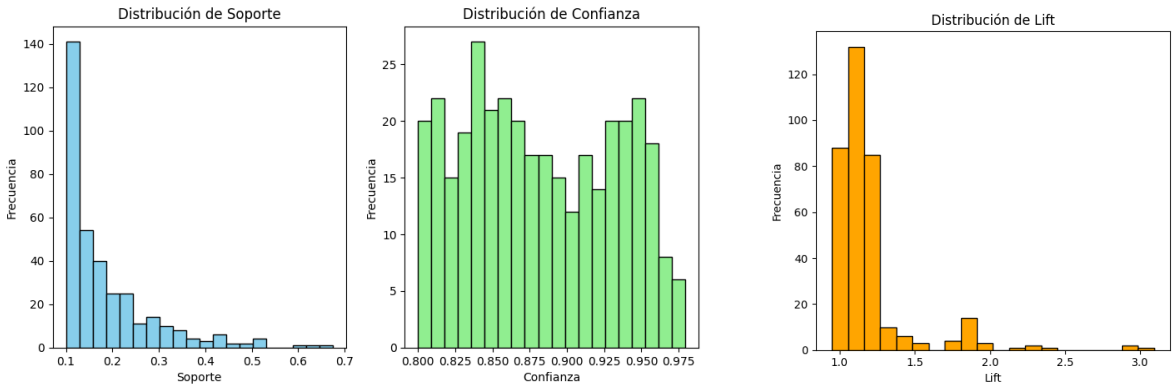
El primer conjunto de atributos que se probó hace referencia a características de las rutas que se conocen y los conductores principales que se asignan, esperando encontrar alguna regla que indique si un tipo concreto de conductor tiene una mayor tendencia a ser asignado a rutas con ciertas características. Estos atributos hacen referencia al número de paradas, cargas, descargas y puntos de las rutas, la localidad de origen y distancia total en kilómetros y, referente a los conductores si se asignan uno o dos y si el principal es uno de los más frecuentes de la zona (entendiendo esta como la localidad de origen). Tras dividir los atributos en rangos usando las técnicas de ciencia de datos explicadas en la Sección 2 y definir las etiquetas se obtienen todos los posibles ítems de las reglas, como se puede ver en la Tabla 8.

Tabla 8: División en nuevos atributos para la primera construcción de reglas de asociación.

Atributo	**División en nuevos atributos**
Distancia total de la ruta (km)	▪ DistanciaCorta: <=630 ▪ DistanciaMedia: >630 y <=1375 ▪ DistanciaLarga: >1375
Número de paradas	▪ PocasParadas: <=3 ▪ BastantesParadas: >3 y <=8 ▪ GranNumParadas: >8
Número de puntos definidos	▪ PocosPuntos: <=4 ▪ BastantesPuntos: >4 y <=9 ▪ GranNumPuntos: >9
Número de conductores	▪ UnConductor ▪ DosConductores
Número de cargas y descargas	▪ SinCargas/SinDescargas: 0 ▪ PocasCargas/PocasDescargas: >0 y <=2 ▪ BastantesCargas/BastantesDescargas: >2
Frecuencia del conductor en la zona	▪ ConductorExterno: es un contratado externo (no tiene identificador propio para contrastar) ▪ UnicoConductor: único conductor que parte de esa zona. ▪ ConductorPocofrecuente: es el conductor de esa zona que menos rutas ha iniciado desde ahí. ▪ ConductorMuyFrecuenteEntreVarios: es el conductor de esa zona que más rutas ha iniciado desde ahí.

Al construir las reglas se definieron como umbrales de soporte y confianza 0.1 y 0.8, respec-

tivamente. Se obtuvieron 352 reglas, pero antes de entrar en detalle y empezar a analizarlas se dibujaron gráficas mostrando la distribución de las tres medidas que se han mencionado anteriormente para tener una visión global de la calidad de las reglas, estas se pueden observar en la Figura 21.



(a) Distribución de métricas de soporte y confianza

(b) Distribución de métrica *lift*

Figura 21: Distribuciones de métricas para la primera construcción de reglas

Observando estas estadísticas se concluye que, por lo general, las reglas obtenidas no tienen un gran soporte, lo que puede deberse a haber muchas combinaciones diferentes de atributos. En cuanto a la confianza se cuenta con bastantes reglas que se acercan a uno, por lo que en este aspecto parecen de interés. La distribución del *lift* indica que en la gran mayoría de casos es muy próximo a uno, pero mientras no sean exactamente uno, pueden ser reglas interesantes con atributos que dependan unos de otros.

Se encontraron principalmente reglas que se podrían considerar triviales. Algunas de estas se listan en la Tabla 9:

Tabla 9: Reglas triviales obtenidas (primer conjunto de reglas)

Regla	**S**	**C**	**lift**
PocosPuntos, UnConductor ⇒ DistanciaCorta, PocasParadas	0.1087	0.8077	3.0924
DistanciaLarga ⇒ DosConductores	0.1508	0.9576	1.1967
PocasDescargas, PocasParadas, DistanciaCorta ⇒ SinCargas	0.1171	0.8079	0.8079
BastantesPuntos, SinCargas ⇒ DistanciaMedia	0.1215	0.8018	1.4527

Antes de comentar estas en concreto, hay que mencionar que muchas reglas que solo relacionaban paradas con cargas y descargas fueron descartadas del análisis. Esto se debe a que se comprobó que no existen paradas cuya función no sea otra que cargar o descargar mercancía, además de la de enganche y desenganche de la caja del camión al inicio y final de la ruta. Para discriminar entre los cientos de reglas obtenidas, se puso el foco en encontrar aquellas con confianza o *lift* altos y después se buscaron consecuentes interesantes para ver si los antecedentes podían dar información de algún patrón. La primera regla que se lista es, de entre todas, la que mayor *lift* tiene, esto es, que hay una gran dependencia en la aparición de

los conjuntos de ítems. La asociación se podría esperar, que en rutas donde se definen pocos puntos y un solo conductor estas suelen ser cortas y se realizan pocas paradas. Por otro lado, encontramos otra regla que ejemplifica el caso opuesto en el que para rutas muy largas, casi siempre (debido a una confianza muy alta) se asignan dos conductores para que se turnen. La tercera regla indica, si ignoramos la relación directa entre paradas y cargas/descargas, que en rutas cortas solo hay un origen y un destino al que llevar una mercancía, sin necesitar pasar por algún almacén a recoger algo. Por último, llama la atención en la última regla que cuando se definen muchos puntos de forma manual en la ruta, sin que estos sean paradas para cargar mercancía, la ruta es de una longitud media. Una posible explicación por la que la persona encargada de diseñar esas ruta defina muchos puntos podría ser evitar carreteras concretas o quizá forzar a tomar la autovía y no carreteras comarcales que puedan ser menos transitables para un camión. Esta explicación no pudo ser contrastada con ACME.

Desgraciadamente, parece que no tuvo un impacto claro en las reglas la frecuencia de los conductores en las zonas de las rutas. No hubo ninguna que lo tuviese como consecuente directo y cuando aparecía en los antecedentes no se pudo extraer ninguna conclusión.

Mientras se estudiaban las fuentes de información de las que extraer el tiempo de duración de las rutas se comprobó que había casos en los que los datos introducidos no eran fiables. Se quiso comprobar por qué la fecha real que supuestamente representa el inicio y final de la ruta cuando se hizo efectiva en ocasiones no estaba bien introducida. Se encontraron casos como que esas fechas fuesen exactamente iguales a las que se definieron como estimación (poco probable que se haya dado así con tanta precisión), no se introdujese la fecha de inicio real o final o que ambas fechas fuesen la misma. Por tanto, se consideró interesante categorizar las rutas añadiendo un atributo que indicase si estas fechas tenían sentido y, usando otros atributos, generar más reglas de asociación con el objetivo de encontrar algún patrón que indicara por qué no se ha seguido el plan correctamente. Los atributos que se usaron fueron: el identificador del conductor principal de la ruta (en caso de que quizá algunos concretos tuviesen mayor tendencia a cometer los fallos), la localidad de origen de la ruta, la longitud de la misma (usando los mismos rangos descritos previamente) y otro atributo que indica si las fechas introducidas cumplen con lo previsto según los criterios antes mencionados.

En la Tabla 10 se muestran reglas que se consideraron interesantes. Se definieron como umbrales de soporte y confianza 0.001 y 0.8, respectivamente.

Tabla 10: Reglas interesantes obtenidas (segundo conjunto de reglas)

Regla	**S**	**C**	**lift**
3178 ⇒ NoCumple	0.0018	0.8421	2.6865
ALGECIRAS ⇒ -1	0.0096	0.883	22.2331
BAHRAOUIYNE TANJA ⇒ -1	0.0099	1	25.1797
CASABLANCA ⇒ -1	0.0015	1	25.1797
GZENAYA ⇒ -1	0.0018	1	25.1797
TANGER ⇒ -1	0.001	1	25.1797
-1, ALGECIRAS ⇒ NoCumple	0.0083	0.8675	2.7674
BAHRAOUIYNE TANJA, -1 ⇒ NoCumple	0.0093	0.9419	3.0048
-1, CASABLANCA ⇒ NoCumple	0.0015	1	3.1902
-1, DistanciaCorta ⇒ NoCumple	0.0195	0.8942	2.8526

En este caso se obtuvieron una serie de reglas que permiten encontrar un patrón que caracteriza en qué casos estas fechas se introducen incorrectamente. Tras la primera revisión rápida de las 753 reglas obtenidas filtrando por consecuentes, confianza y *lift*, se encontraron muchos casos de muchos conductores (sus identificadores) que introducen correctamente en la gran mayoría de casos las fechas (atributo "Cumple"), por lo que se puede entender que los casos erróneos no están generalizados entre todos los conductores. También se encontraron reglas que indicaban tendencias de ciertos conductores a comenzar rutas en ciertas localidades o que fuesen de ciertas distancias, pero para el objetivo que se tiene en este análisis de reglas es irrelevante. Algunas de las reglas interesantes que se encontraron se encuentran listadas en la Tabla 10. Hay una serie de conclusiones que se extraen de estas reglas:

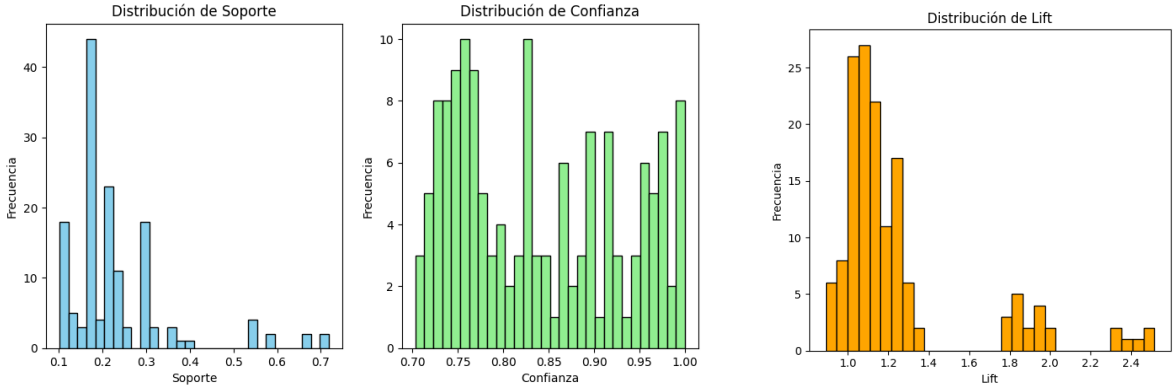
- La primera regla es un ejemplo entre algunas otras en las que conductores concretos tienden a registrar incorrectamente las fechas.
- Las siguientes cinco reglas tienen como consecuente el identificador que se le da a un conductor contratado externo (no registrado como empleado de la empresa) y todas ellas tienen como antecedente que la ruta partiese de ciudades del norte de África, concretamente Marruecos.
- Las últimas cuatro reglas indican además, que los conductores externos que parten de ciudades de esa zona y los que toman rutas cortas no suelen registrar los datos acerca del comienzo y final. Una hipótesis que podría explicar este patrón podría ser que estos conductores contratados externamente no tuviesen acceso de escritura o la obligación de registrar para ACME esos datos, pero esto no ha podido verificarse.

Otro planteamiento de análisis de reglas de asociación que se trabajó fue categorizar a los conductores y buscar algún patrón que indicase algún motivo por el que se asignan a las rutas. Siguiendo el mismo método que en el primer conjunto de reglas de asociación, se definen los atributos que se usarán. Hay que recordar que este proceso en concreto es volver a la fase anterior de preparación de datos, pero se comenta en esta sección ya que aquí es cuando toma relevancia. En la Tabla 11 se describen los atributos elegidos y cómo se discretizan.

Tabla 11: División en nuevos atributos para la tercera construcción de reglas de asociación.

Atributo	**División en nuevos atributos**
Distancia media de las rutas que toma el conductor (km)	▪ DistanciaCorta: <=630 ▪ DistanciaMedia: >630 y <=1375 ▪ DistanciaLarga: >1375
Número de localidades que ha visitado en sus rutas	▪ MenosDe20Localiades: <20 ▪ MenosDe40Localiades: >=20 y <40 ▪ MenosDe60Localiades: >=40 y <60 ▪ MuchasLocalidades: >=60
Antigüedad del conductor desde que se dio de alta en la empresa (días)	▪ MenosDe3Anhos: <1095 ▪ Entre3y5Anhos: >=1095 y <=1825 ▪ MasDe5Anhos: >1825
Número de rutas que ha completado	▪ PocasRutas: <=18 ▪ BastantesRutas: >18 y <=40 ▪ MuchasRutas: >40
Frecuencia de tiempo media entre que finaliza una ruta y le llaman para tomar otra (días)	▪ SoloHizoUnaRuta: no se puede calcular diferencia de días ▪ FrecuenciaSemanal: <=10 ▪ FrecuenteMensual: >10 y <=20 ▪ PocoFrecuente: >20 y <=70 ▪ RaramenteHaceRuta: >70

La distribución de soporte, confianza y *lift* de las 121 reglas que se obtuvieron se encuentra en la Figura 22. Se definieron como umbrales de soporte y confianza 0.1 y 0.7, respectivamente.



(a) Distribución de métricas de soporte y confianza

(b) Distribución de métrica *lift*

Figura 22: Distribuciones de métricas para la tercera construcción de reglas

En comparación con el primer conjunto de reglas, se ha obtenido un mejor soporte y una buena confianza para estas reglas. Los valores de *lift* siguen siendo próximos a uno, aunque hay más casos en los que se aleja más y, por tanto, las reglas pueden ser más interesantes.

Se extrajeron las reglas de asociación que se muestran en la Tabla 12.

Tabla 12: Reglas interesantes obtenidas (tercer conjunto de reglas)

Regla	**S**	**C**	**lift**
Entre3y5Anhos ⇒ FrecuenciaSemanal	0.1495	0.8222	1.0409
MasDe5Anhos ⇒ FrecuenciaSemanal	0.1212	0.8451	1.0698
PocasRutas ⇒ MenosDe20Localiades	0.2505	0.7425	2.5174
MenosDe3Anhos, MenosDe20Localiades ⇒ PocasRutas	0.1838	0.9192	2.7246
PocasRutas, MenosDe3Anhos, ⇒ SoloHizoUnaRuta DistanciaLarga	0.0101	0.7143	8.8393

Lamentablemente, la única información que se ha podido extraer de este análisis no tiene demasiado valor, ya que son resultados que ya se podían intuir antes de comenzar. En general, los consecuentes de las reglas han terminado siendo atributos que representan a la mayoría de conductores que toman rutas de distancia media o son llamados con una frecuencia semanal. No se pudieron construir reglas para esos casos diferentes a la media con frecuencias de trabajo diferentes o antigüedad en la empresa que pudiesen servir para caracterizarlos. En general, las reglas recogidas representan ideas obvias como que conductores que realizan pocas rutas pasan por pocas localidades.

La separación que se hace respecto a la última regla es porque forma parte de un nuevo análisis de reglas que se hizo. Para buscar patrones en los casos más extraños que se alejaban de la media, se hizo una segunda construcción de reglas con un umbral de soporte mínimo de 0.01. Sin embargo, la gran mayoría de reglas seguían siendo del mismo carácter que las mencionadas previamente, con excepción de la que aparece en la última fila de la Tabla 12. La nueva información que aporta esta es que aquellos conductores con menos tiempo en la empresa (inferior a tres años) solo hicieron una ruta y esta era una de gran longitud.

6. Conclusiones y líneas de trabajo futuro

Durante el desarrollo del trabajo se ha aprendido la importancia del ciclo de vida de la minería de datos. En un principio, se subestimaron los primeros pasos en los que se requería tener una visión global de la lógica del negocio y comprender los datos con los que se trabajaba. Este proceso terminó tomando más tiempo del esperado, requiriendo consultar dudas a través de reuniones con Fagor, además de elaborar documentos para contrastar con ellos la comprensión que se tenía sobre la información encontrada en la base de datos. La experiencia adquirida constata la importancia que tienen los metadatos y la calidad de los datos disponibles para realizar la analítica, ya que condiciona todo el proceso, conduciendo a tiempos de procesado más largos y patrones poco concluyentes.

Por otra parte, el hecho de que los procesos que registran los datos no estén alineados con el propósito del análisis hace que este sea más pobre. Resulta esencial, pero en este caso no ha sido posible por tratarse de la base de datos de un tercero, que se hubieran dado indicaciones de cómo completar los registros sobre pedidos, clientes y rutas teniendo en cuenta las reglas expertas del controlador de tráfico. Este hecho ha sido el motivo principal por el que finalmente no fue posible construir un modelo predictivo. De este estudio se extrae la importancia de incluir en el proceso información sobre si la ruta se realizó con éxito o si hubo motivos que retrasaron la entrega. Es decir, si se completó en tiempo y según lo previsto, si hubo algún tipo de incidente o si se tomó una ruta alternativa a la definida inicialmente. Otro aspecto de mejora dirigido a obtener datos de mayor calidad sería incrementar los controles al introducir información en la aplicación, dado que se han encontrado inconsistencias.

Una conclusión importante que se desprende de este trabajo es la justificación de buenas prácticas en la ingeniería de software para mitigar los problemas encontrados durante el desarrollo de las tareas. Se destacan principalmente dos: la elaboración de una documentación exhaustiva de la base de datos, que pueda ser entregada a personas externas al producto (aunque se comprende que puede ser difícil apartar tiempo del desarrollo de nuevas funciones para recopilar y explicar esta información), y el control de calidad de los datos en los sistemas de información, enfatizado anteriormente. Es comprensible que los plazos de tiempo en una empresa y el deseo de lograr la máxima eficacia en la lógica de negocio dificulten la implementación completa de estas prácticas, las cuales requieren de una dedicación adecuada de tiempo. Sin embargo, el desarrollo de este trabajo demuestra que, a largo plazo, vale la pena realizar este esfuerzo.

En resumen, el proyecto se ha realizado sin contar con toda la información necesaria para ser exitoso. No obstante, se le ofrece a Fagor un conocimiento de uso de este módulo por parte de los clientes que permitan mejorarlo, así como un conjunto de técnicas y estrategias de análisis para incorporar en su stack tecnológico, lo que permitiría en un futuro realizar proyectos de minería de datos.

La respuesta que se obtuvo de la persona responsable del proyecto por parte de Fagor, con respecto a los resultados del presente trabajo, fue: “gracias al gran trabajo realizado por Miguel en el proyecto hemos interiorizado la importancia de los metadatos y la necesidad de tener un modelo de datos completo y documentado si queremos proporcionar información procesada de alto valor añadido a nuestros clientes. Además, durante el proyecto hemos conocido tecnologías que pueden ser de gran utilidad en el futuro y que sin duda añadiremos a nuestro stack tecnológico.”

En el plano personal, el desarrollo de este trabajo me ha brindado la oportunidad de profundizar en mis conocimientos previos sobre minería de datos adquiridos durante el grado. El contraste entre estudiar las técnicas teóricamente y aplicarlas en un contexto real ha marcado una diferencia significativa en mi aprendizaje. He participado activamente en todas las etapas del ciclo de vida de la minería de datos, con excepción del despliegue y evaluación de un modelo predictivo, desde la recolección y preparación de datos hasta la extracción de conocimiento y la interpretación de resultados. A lo largo de este proceso me he enfrentado y he superado diversos desafíos que han fortalecido mi capacidad analítica y resolutiva. También colaboré estrechamente con el equipo de Fagor, consultando con ellos dudas y presentándoles los resultados obtenidos. Esta colaboración me permitió aprender a comunicar de manera efectiva hallazgos técnicos a un público no especializado. Además, ha sido mi primera experiencia en el entorno productivo y he podido contrastar las diferencias con el mundo académico.

A partir de los resultados del trabajo realizado, propondría como trabajo futuro plantear de nuevo el objetivo de minería que se pretendía, pero estableciendo previamente el proceso de recogida de información necesaria para ser exitoso. En particular, si se quiere construir un recomendador de ruta, vehículo y conductor en tiempo real, se debe disponer de un conjunto de recomendaciones históricas etiquetadas como correctas o incorrectas con las cuales entrenar al modelo.

Por último, conviene señalar que la experiencia transferida con este TFG permitirá a la empresa plantear nuevas propuestas de interés que pudiera ofrecer a sus clientes como, por ejemplo, caracterización del perfil de conducción, notificaciones de mantenimiento preventivo o clasificación de rutas a evitar.

Bibliografía

[1] Hu, W. C., Wu, H. T., Cho, H. H., & Tseng, F. H. (2020). Optimal route planning system for logistics vehicles based on artificial intelligence. Journal of Internet Technology, 21(3), 757-764.

[2] Shearer, C. (2000). The CRISP-DM model: the new blueprint for data mining. Journal of data warehousing, 5(4), 13-22.

[3] Chapman, P., Clinton, J., Kerber, R., Khabaza, T., Reinartz, T., Shearer, C., & Wirth, R. (1999, March). The CRISP-DM user guide. In 4th CRISP-DM SIG Workshop in Brussels in March (Vol. 1999). sn.

[4] Charu C. Aggarwal & Chandan K. Reddy. (2014). Data Clustering Algorithms and Applications. CRC Press.

[5] Lloyd, S. (1982). Least squares quantization in PCM. IEEE Transactions on Information Theory, 28(2), 129-137.

[6] Pujari, A. K. (2001). Data mining techniques. Universities press.

[7] Hand, D. J., Mannila, H., & Smyth, P. (2001). Principles of Data Mining. MIT Press.

[8] Jiawei, H., & Micheline, K. (2006). Data mining: concepts and techniques. Morgan kaufmann.