ESCUELA TÉCNICA SUPERIOR DE INGENIERÍA (ICAI)

MÁSTER EN BIG DATA: TECNOLOGÍA Y ANALÍTICA AVANZADA

ANOMALY DETECTION IN RETAIL SALES.

Autor: Ignacio Jiménez Serrano

Director: Alejandro Ortega

Madrid

Junio 2018

Declaro, bajo mi responsabilidad, que el Proyecto presentado con el título

ANOMALY DETECTION IN RETAIL SALES.

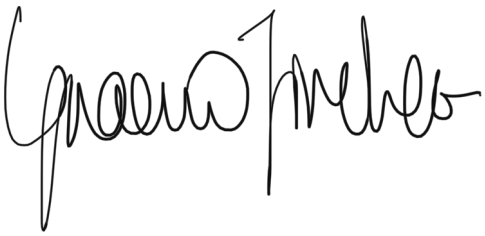
en la ETS de Ingeniería - ICAI de la Universidad Pontificia Comillas en el

curso académico 2018/19 es de mi autoría, original e inédito y

no ha sido presentado con anterioridad a otros efectos.

El Proyecto no es plagio de otro, ni total ni parcialmente y la información que ha sido

tomada de otros documentos está debidamente referenciada.



Fdo.: Ignacio Jiménez Serrano Fecha: 2 / Jul / 2021

Autorizada la entrega del proyecto

ALEJANDRO ORTEGA

Fdo.: Alejandro Ortega Fecha: 2 / Jul / 2021

Vº Bº del Coordinador de Proyectos

Fdo.: Carlos Morrás Ruiz-Falcó Fecha: 2 / Jul / 2021

AUTORIZACIÓN PARA LA DIGITALIZACIÓN, DEPÓSITO Y DIVULGACIÓN EN RED DE PROYECTOS FIN DE GRADO, FIN DE MÁSTER, TESINAS O MEMORIAS DE BACHILLERATO

1º. Declaración de la autoría y acreditación de la misma.

El autor D. IGNACIO JIMÉNEZ SERRANO

DECLARA ser el titular de los derechos de propiedad intelectual de la obra: ANOMALY DETECTION IN RETAIL SALES, que ésta es una obra original, y que ostenta la condición de autor en el sentido que otorga la Ley de Propiedad Intelectual.

2º. Objeto y fines de la cesión.

Con el fin de dar la máxima difusión a la obra citada a través del Repositorio institucional de la Universidad, el autor **CEDE** a la Universidad Pontificia Comillas, de forma gratuita y no exclusiva, por el máximo plazo legal y con ámbito universal, los derechos de digitalización, de archivo, de reproducción, de distribución y de comunicación pública, incluido el derecho de puesta a disposición electrónica, tal y como se describen en la Ley de Propiedad Intelectual. El derecho de transformación se cede a los únicos efectos de lo dispuesto en la letra a) del apartado siguiente.

3º. Condiciones de la cesión y acceso

Sin perjuicio de la titularidad de la obra, que sigue correspondiendo a su autor, la cesión de derechos contemplada en esta licencia habilita para:

a) Transformarla con el fin de adaptarla a cualquier tecnología que permita incorporarla a internet y hacerla accesible; incorporar metadatos para realizar el registro de la obra e incorporar "marcas de agua" o cualquier otro sistema de seguridad o de protección.
b) Reproducirla en un soporte digital para su incorporación a una base de datos electrónica, incluyendo el derecho de reproducir y almacenar la obra en servidores, a los efectos de garantizar su seguridad, conservación y preservar el formato.
c) Comunicarla, por defecto, a través de un archivo institucional abierto, accesible de modo libre y gratuito a través de internet.
d) Cualquier otra forma de acceso (restringido, embargado, cerrado) deberá solicitarse expresamente y obedecer a causas justificadas.
e) Asignar por defecto a estos trabajos una licencia Creative Commons.
f) Asignar por defecto a estos trabajos un HANDLE (URL *persistente)*.

4º. Derechos del autor.

El autor, en tanto que titular de una obra tiene derecho a:

a) Que la Universidad identifique claramente su nombre como autor de la misma
b) Comunicar y dar publicidad a la obra en la versión que ceda y en otras posteriores a través de cualquier medio.
c) Solicitar la retirada de la obra del repositorio por causa justificada.
d) Recibir notificación fehaciente de cualquier reclamación que puedan formular terceras personas en relación con la obra y, en particular, de reclamaciones relativas a los derechos de propiedad intelectual sobre ella.

5º. Deberes del autor.

- El autor se compromete a:

a) Garantizar que el compromiso que adquiere mediante el presente escrito no infringe ningún derecho de terceros, ya sean de propiedad industrial, intelectual o cualquier otro.
b) Garantizar que el contenido de las obras no atenta contra los derechos al honor, a la intimidad y a la imagen de terceros.
c) Asumir toda reclamación o responsabilidad, incluyendo las indemnizaciones por daños, que pudieran ejercitarse contra la Universidad por terceros que vieran infringidos sus derechos e intereses a causa de la cesión.
d) Asumir la responsabilidad en el caso de que las instituciones fueran condenadas por infracción

de derechos derivada de las obras objeto de la cesión.

6º. Fines y funcionamiento del Repositorio Institucional.

La obra se pondrá a disposición de los usuarios para que hagan de ella un uso justo y respetuoso con los derechos del autor, según lo permitido por la legislación aplicable, y con fines de estudio, investigación, o cualquier otro fin lícito. Con dicha finalidad, la Universidad asume los siguientes deberes y se reserva las siguientes facultades:

- La Universidad informará a los usuarios del archivo sobre los usos permitidos, y no garantiza ni asume responsabilidad alguna por otras formas en que los usuarios hagan un uso posterior de las obras no conforme con la legislación vigente. El uso posterior, más allá de la copia privada, requerirá que se cite la fuente y se reconozca la autoría, que no se obtenga beneficio comercial, y que no se realicen obras derivadas.
- La Universidad no revisará el contenido de las obras, que en todo caso permanecerá bajo la responsabilidad exclusive del autor y no estará obligada a ejercitar acciones legales en nombre del autor en el supuesto de infracciones a derechos de propiedad intelectual derivados del depósito y archivo de las obras. El autor renuncia a cualquier reclamación frente a la Universidad por las formas no ajustadas a la legislación vigente en que los usuarios hagan uso de las obras.
- La Universidad adoptará las medidas necesarias para la preservación de la obra en un futuro.
- La Universidad se reserva la facultad de retirar la obra, previa notificación al autor, en supuestos suficientemente justificados, o en caso de reclamaciones de terceros.

Madrid, a 2 de Julio de 2021

ACEPTA

Fdo. Ignacio Jiménez Serrano

Motivos para solicitar el acceso restringido, cerrado o embargado del trabajo en el Repositorio Institucional:

ESCUELA TÉCNICA SUPERIOR DE INGENIERÍA (ICAI)

MÁSTER EN BIG DATA: TECNOLOGÍA Y ANALÍTICA AVANZADA

ANOMALY DETECTION IN RETAIL SALES.

Autor: Ignacio Jiménez Serrano

Director: Alejandro Ortega

Madrid

Junio 2018

ANOMALY DETECTION IN RETAIL SALES.

Autor: Ignacio Jiménez Serrano
Director: Alejandro Ortega
Entidad Colaboradora: COFARES

RESUMEN DEL PROYECTO

Palabras clave: Anomalías, prophet, autoarima, series temporales, medicamentos.

1. Introducción

En el proceso de venta de productos al por menor hay numerosas etapas que no están automatizadas y por tanto entra en juego el error humano causando en las empresas gastos que se podrían evitar mediante la detección de anomalías. Esta técnica consiste en identificar aquellos pedidos que se desvían del comportamiento habitual mediante inteligencia artificial. Una detección temprana de las anomalías en el proceso de venta y distribución es muy interesante a nivel de negocio ya que puede prevenir y no 'curar' errores evitables.

2. Definición del proyecto

El proyecto se ha divido en 4 etapas: adquisición de datos, tratamiento de datos, selección del modelo, visualización.

En la primera etapa, se estudia las diferentes tablas que la empresa pone a la disposición de este proyecto para seleccionar las variables más apropiadas para el caso de uso. Para la unión y extracción de datos se utiliza la herramienta de Bigquery.

En la etapa del tratamiento datos hay dos objetivos, la selección de los productos y la elección de la dimensión a la que se quiere realizar el proyecto. Mediante un análisis preliminar de los diferentes productos con los que trabaja Cofares se elige aquellos que se consideren relevantes para el caso de uso. También, se decide sobre que dimensión agrupar estos productos, si se quieren agrupar a nivel: cliente, código postal … o si se quieren agrupar por dimensiones mas amplias como grupo terapéuticos, comunidades autónomas…

En la tercera etapa se realiza un estudio de la viabilidad de diferentes modelos de predicción. Se le da importancia a la precisión del modelo, pero sobre todo, al tiempo de ejecución. En función de la dimensión elegida en la etapa anterior el proyecto trabaja con miles o millones de series temporales. Al trabajar con tantas series temporales, asignar una única lógica para generar el umbral que delimita la región de normalidad provoca mucho error en la detección de anomalías, por tanto, en esta etapa también se desarrolla un modelo de clasificación de las series temporales para asignar a cada grupo una lógica diferente.

Y por último en la etapa de visualización se desarrolla la herramienta con la que trabajarán los departamentos de auditoría interna y dirección del medicamento en Data Studio.

3. Diseño de los Modelos

Tras un estudio de diferentes agrupaciones de series temporales se ha decidido agrupar por Grupo Terapéutico – Comunidad Autónoma. Al trabajar con tantas series temporales, asignar una única lógica para generar el umbral que delimita la región de normalidad provoca mucho error en la detección de anomalías, por eso, se presenta un modelo de clasificación de las series temporales. El cálculo del umbral se basa en la predicción de ventas que estima un modelo de predicción. Para la selección del modelo de predicción se realiza un estudio de viabilidad de la función autoarima y prophet, se comparan y se concluye cuál de las dos funciones se ajusta mejor a nuestro caso de uso

4. Modelo clasificación

El objetivo de la clasificación es separar por grupos las st más estables de lss meneos estables, para ellos, se ha elegido las dos siguientes variables: el número de ceros en la serie temporal (st) y la media de la cantidad confirmada de los pedidos. Teniendo estas dos variables el número óptimo de clústeres es 3.

5. Modelo de predicción

Para la selección del modelo se va a tener dos condiciones en cuenta, el tiempo de ejecución y la precisión del modelo.

Tabla 1: comparativa resultados modelos de predicción

AUTOARIMA

	rmse	media	ratio	tiempo
0	2.595305	0.904762	2.868495	33.153006
1	3.846676	5.738095	0.670375	28.690603
2	0.158906	0.023810	6.674060	6.785869
3	140.172223	661.452381	0.211916	26.733364
4	45.342302	197.452381	0.229637	47.491807
5	1960.541743	8484.190476	0.231082	40.255469
6	234.502923	852.714286	0.275008	66.560135
7	7726.020359	15974.333333	0.483652	42.707572
8	216.295921	396.023810	0.546169	57.967325

PROPHET

	rmse	media	ratio	tiempo
0	3.197915	0.904762	3.534537	2.361084
1	4.006063	5.738095	0.698152	2.454201
2	0.224981	0.023810	9.449191	2.712605
3	75.825243	661.452381	0.114634	2.499854
4	36.365709	197.452381	0.184175	2.528173
5	1239.977315	8484.190476	0.146152	2.676439
6	291.444407	852.714286	0.341784	2.561660
7	2228.988938	15974.333333	0.139536	2.601567
8	63.685094	396.023810	0.160811	2.574198

Analizando la Tabla 1 se puede ver que, para la primera condición, está claro que prophet supera con diferencia a la función autoarima. La media de prophet está cercana a 2 segundos, mientras que la de autoarima casi llega a los 40 segundos, es decir, tarda casi 20 veces más de lo que tarda prophet.

Para la segunda condición, la precisión de las st se ha calculado la variable ratio que refleja la proporción entre el rmse de cada serie con su media. Utilizando la columna de ratio destaca que la función prophet tiene mejores resultados que la función autoarima.

En conclusión, viendo que la función prophet tiene una gran diferencia de tiempo de ejecución respecto a autoarima y también tiene mejores resultados en precisión, se concluye que para este caso de uso se va a utilizar la función prophet.

6. Detección de anomalías

Una de las razones por la que se realizó la clasificación de las st fue para poder separar las st más estables de las menos y de esta forma generar diferentes lógicas para calcular el umbral.

En los grupos donde se encuentran las series temporales menos estables se seguirá la lógica de la Ilustración 2. Para las st estables se definirá la región de normalidad como *3*predicción* del modelo, es decir, si las ventas superan tres veces la predicción del modelo se clasificará como pedido anómalo. Un ejemplo de las anomalías detectadas sería la Ilustración 1, en esta figura se tiene en azul el límite de la región de normalidad, en rojo la predicción del modelo y en negro las ventas reales. En este caso solo se han detectado dos anomalías (estrellas verdes) sobre el mes 11 del año 2020, se ve claramente como hay un claro pico muy exagerado en la st negra.

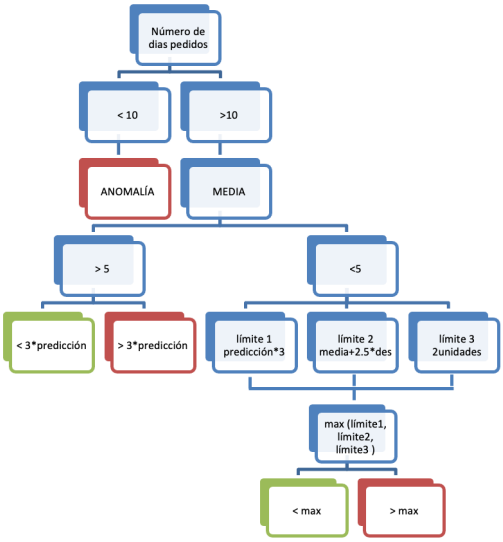


Ilustración 2: lógica seguida para cálculo de anomalías

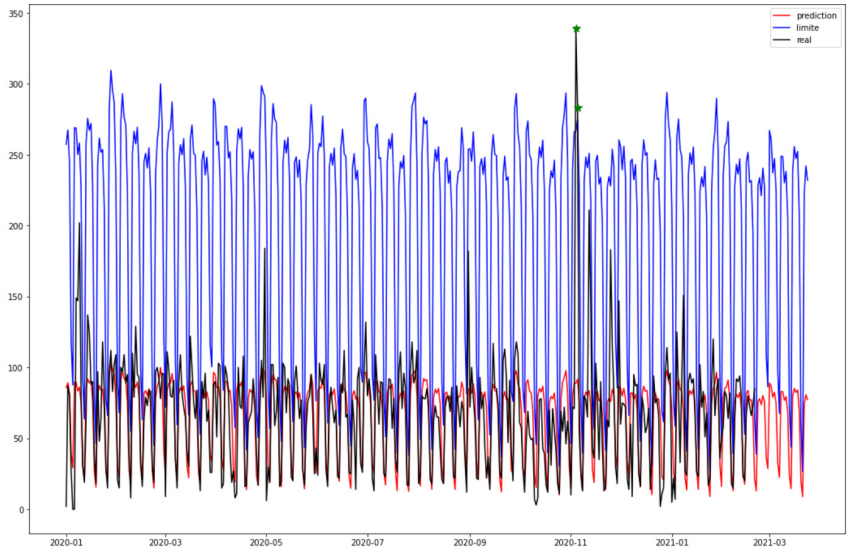


Ilustración 1: ejemplo de anomalías detectadas

7. conclusiones

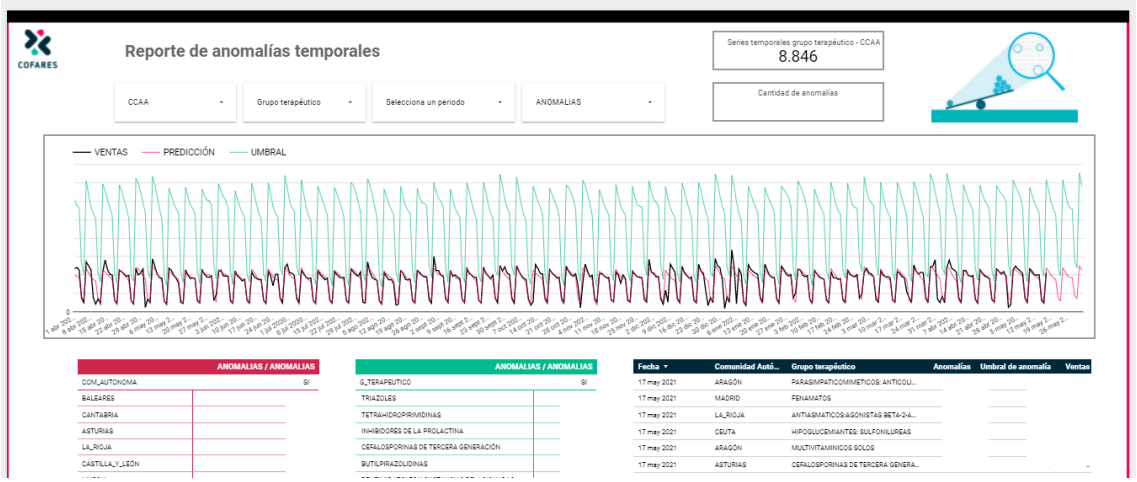


Ilustración 3: herramienta diseñada

En este proyecto se ha conseguido desarrollar una herramienta interactiva para que el departamento de auditoría interna pueda identificar, investigar y seguir de cerca los pedidos anómalos. Esta herramienta representa gráficamente el modelo de clasificación de pedidos de forma sencilla y directa mejorando la anterior herramienta.

Gracias a la adecuada selección de variables, la herramienta proporciona a los auditores información reciente. De esta forma se ha conseguido reducir el tiempo de detección de anomalías disminuyendo el periodo de actuación.

ANOMALY DETECTION IN RETAIL SALES.

Author: Ignacio Jiménez Serrano
Supervisor: Alejandro Ortega
Collaborating Entity: COFARES

ABSTRACT

Keywords: Anomalies, prophet, autoarima, time series, medications

1. Introduction

In the process of retailing products there are numerous stages that are not automated and therefore human error comes into play, causing losses in companies that could be avoided by detecting anomalies. This technique consists of identifying those orders that deviate from the usual behavior using artificial intelligence. An early detection of anomalies in the sales and distribution process is very interesting at the business level since it can prevent and not 'cure' avoidable errors.

2. Project definition

The project has been divided into 4 stages: data acquisition, data processing, model selection, visualization.

In the first stage, the different tables that the company makes available for this project are studied to select the most appropriate variables for the use case. The Bigquery tool is used for data union and extraction.

In the data processing stage there are two objectives, the selection of the products and the choice of the dimension to which the project is to be carried out. Through a preliminary analysis of the different products with which Cofares works, those that are considered relevant for the use case are chosen. Also, it is decided on which dimension to group these products, if they want to be grouped at the level: customer, postal code ... or if they want to be grouped by bigger dimensions such as therapeutic group, CCAA ...

In the third stage, a feasibility study of different prediction models is carried out. Importance is given to the precision of the model, but above all, the execution time. Depending on the dimension chosen in the previous stage, the project works with thousands or millions of time series. When working with so many time series, assigning a single logic to generate the threshold that delimits the normality region causes a lot of error in the detection of anomalies, therefore, at this stage a classification model of the time series is also developed to assign to each group a different logic.

And finally, in the visualization stage, the tool with which the internal audit will work in Data Studio is developed.

3. Design of the Models

After a study of different groupings of time series, it was decided to group by Therapeutic Group - Autonomous Community. When working with so many time series, assigning a single logic to generate the threshold that delimits the normality region causes a lot of error in the detection of anomalies, therefore, a classification model of the time series is presented. The threshold calculation is based on the sales prediction estimated by a prediction model. For the selection of the prediction model, a feasibility study of the autoarima and prophet function is carried out, they are compared and it is concluded which of the two functions best fits our use case.

4. Classification model

The objective of the classification is to separate the most stable st from the least stable, for them, the following two variables have been chosen: the number of zeros in the time series (st) and the mean of the order quantity. Having these two variables, the optimal number of clusters is 3.

5. Prediction model

For the selection of the model, two conditions will be taken into account, the execution time and the precision of the model.

Tabla 2:comparative results prediction models

AUTOARIMA

	rmse	media	ratio	tiempo
0	2.595305	0.904762	2.868495	33.153006
1	3.846676	5.738095	0.670375	28.690603
2	0.158906	0.023810	6.674060	6.785869
3	140.172223	661.452381	0.211916	26.733364
4	45.342302	197.452381	0.229637	47.491807
5	1960.541743	8484.190476	0.231082	40.255469
6	234.502923	852.714286	0.275008	66.560135
7	7726.020359	15974.333333	0.483652	42.707572
8	216.295921	396.023810	0.546169	57.967325

PROPHET

	rmse	media	ratio	tiempo
0	3.197915	0.904762	3.534537	2.361084
1	4.006063	5.738095	0.698152	2.454201
2	0.224981	0.023810	9.449191	2.712605
3	75.825243	661.452381	0.114634	2.499854
4	36.365709	197.452381	0.184175	2.528173
5	1239.977315	8484.190476	0.146152	2.676439
6	291.444407	852.714286	0.341784	2.561660
7	2228.988938	15974.333333	0.139536	2.601567
8	63.685094	396.023810	0.160811	2.574198

Analyzing Tabla 2 it can be seen that, for the first condition, it is clear that prophet far exceeds the autoarima function. The average of prophet is close to 2 seconds, while autoarima almost reaches 40 seconds, it takes almost 20 times as long as it takes prophet.

For the second condition, the precision of the st it has been calculated the ratio variable that reflects the proportion between the rmse of each series with its mean. Using the ratio column, the prophet function has better results than the autoarima function.

In conclusion, seeing that the prophet function has a great difference in execution time respect to autoarima and also has better results in precision, it is concluded that the prophet function will be used for this use case.

6. Anomaly detection

One of the reasons why the st classification was carried out was to be able to separate the most stable st from the least and thus generate different logics to calculate the threshold.

In the st is not stable the logic of Illustration 4 will be followed. For stable st, the region of normality will be defined as *3*prediction* of the model. If sales exceed three times the prediction of the model it will be classified as a failed order. An example of the anomalies detected would be Illustration 3, in this figure the limit of the normality region is shown in blue, the prediction of the model in red and actual sales in black. In this case, only two anomalies (green stars) have been detected on the 11th month of the year 2020, it is clearly seen a very exaggerated peak in the black st.

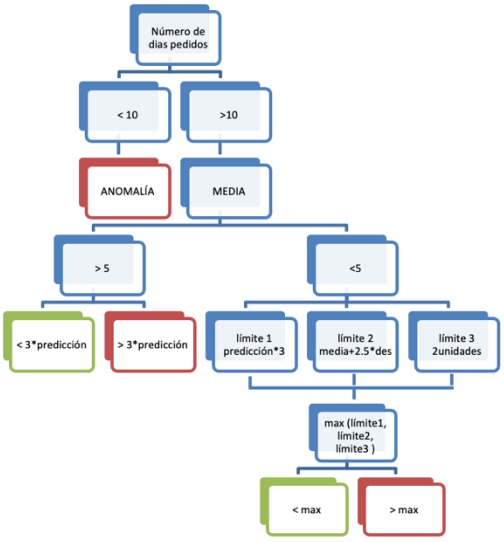


Ilustración 4: logic followed to detection anomalies

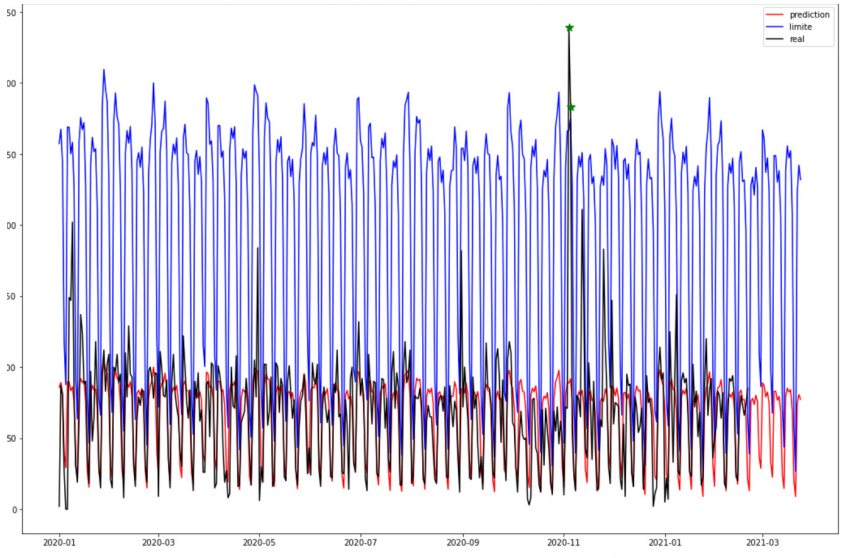


Ilustración 3: example of detected anomalies

8. conclusions

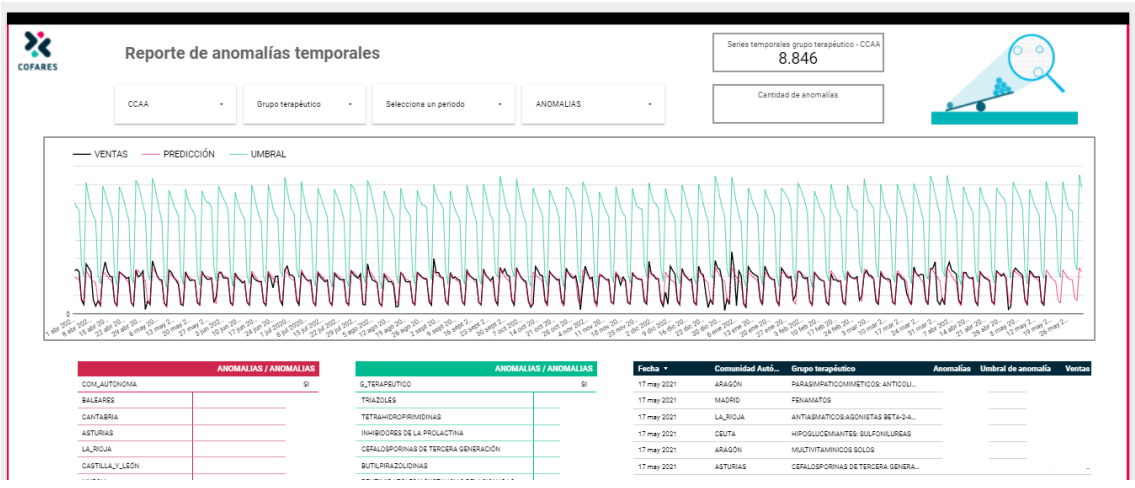


Ilustración 5: designed tool

In this project, an interactive tool has been developed so that the internal audit department can identify, investigate and closely monitor anomalous orders. This tool graphically represents the order classification model in a simple and direct way, improving the previous tool.

Thanks to the appropriate selection of variables, the tool provides the auditors with recent information. In this way, the time for detecting anomalies has been reduced by reducing the period of action.

Índice de la memoria

Índice de figuras

Índice de tablas

Capítulo 1. INTRODUCCIÓN

En el proceso de venta de productos al por menor hay numerosas etapas que no están automatizadas y por tanto entra en juego el error humano causando en las empresas gastos que se podrían evitar mediante la detección de anomalías. Esta técnica consiste en identificar aquellos pedidos que se desvían del comportamiento habitual mediante inteligencia artificial. Una detección temprana de las anomalías en el proceso de venta y distribución es muy interesante a nivel de negocio ya que puede prevenir y no 'curar' errores evitables.

Dentro del sector farmacéutico, Cofares es la empresa líder en distribución de medicamentos en España, una cooperativa de distribución de medicamentos y productos sanitarios, de capital 100% farmacéutico. Se crea como una cooperativa de capital íntegramente farmacéutico cuya finalidad principal es rentabilizar las compras de sus socios, apoyarles en la gestión de sus oficinas de farmacia y ser garante del modelo farmacéutico español. Su visión es proveer a los socios de Cofares de las herramientas necesarias para llevar a cabo su actividad, ayudándoles a afrontar juntos los retos, teniendo siempre como referencia que, para la cooperativa, la fortaleza de la Oficina de Farmacia es y será su principal razón de ser.[1]

Este proyecto se va a desarrollar dentro de Cofares y tiene como objetivo la detección de pedidos anómalos o fraudulentos para asegurar una distribución que garantice un impacto positivo en todos sus socios a la vez que se maximiza los beneficios de la empresa.

1.1 MOTIVACIÓN DEL PROYECTO

La pandemia que estamos viviendo ha sido todo un reto en la planificación y distribución de muchos bienes esenciales como los medicamentos. Cofares ha demostrado un compromiso con la sociedad poniendo en valor la excelencia y la capacidad de innovación impulsando el ecosistema de la salud en momentos tan necesarios, por tanto, generar modelos que ayuden a detectar pedidos que pongan en riego la logística de la empresa es generar modelos que, no solo maximizan los beneficios de Cofares, sino que también impulsan el ecosistema de la salud.

1.2 ESTADO DE LA CUESTIÓN

Tras una búsqueda de soluciones a problemas parecidos al de proyecto se ha encontrado un par de artículos relacionados [2][3]. En estos artículos se expone la dificultad de definir la región de normalidad y diferentes formas de detectar anomalías.

Entre los dos artículos destaca [2] por su similitud con el trabajo realizado. En este proyecto se utiliza técnicas de machine learning para el mantenimiento predictivo industrial. El objetivo del modelo es detectar picos y puntos de cambios en los datos de mantenimiento. Para detectar estos picos en las series temporales utiliza, al igual que en este proyecto, el paquete de Facebook llamado Porphet.

Se puede ver demostrada en estos artículos la utilidad del uso de técnicas de machine learning para la detección de anomalía en aplicaciones como control o fraude.

Capítulo 2. DEFINICIÓN DEL TRABAJO

2.1 OBJETIVOS

- Detección de pedidos anómalos o fraudulentos.

Desarrollar un modelo de clasificación que detecte con éxito los pedidos anómalos. Es importante destacar que una anomalía en los datos no tiene por qué corresponderse con una anomalía a nivel de negocio, por lo que es importante analizar los pedidos para generar variables que se ajusten al caso de uso.

- Reducir el tiempo de detección.

Una detección temprana de las anomalías puede ser decisivo en la toma de decisiones, por tanto, es importante generar un modelo que aporte al departamento de auditoría interna y dirección del medicamento información actualizada de los pedidos. Se generará un fichero que se lanzará periódicamente para aportar información reciente a los departamentos.

- Mejorar la herramienta de los analistas.

Es importante generar gráficas que representen el potencial de estos modelos y que sean fácilmente interpretables. Se buscarán formas de visualización interactivas que representen con sencillez los datos obtenidos mejorando los defectos de la herramienta anterior y conservando los positivos.

2.2 METODOLOGÍA

El proyecto se ha divido en 4 etapas: adquisición de datos, tratamiento de datos, selección del modelo, visualización.

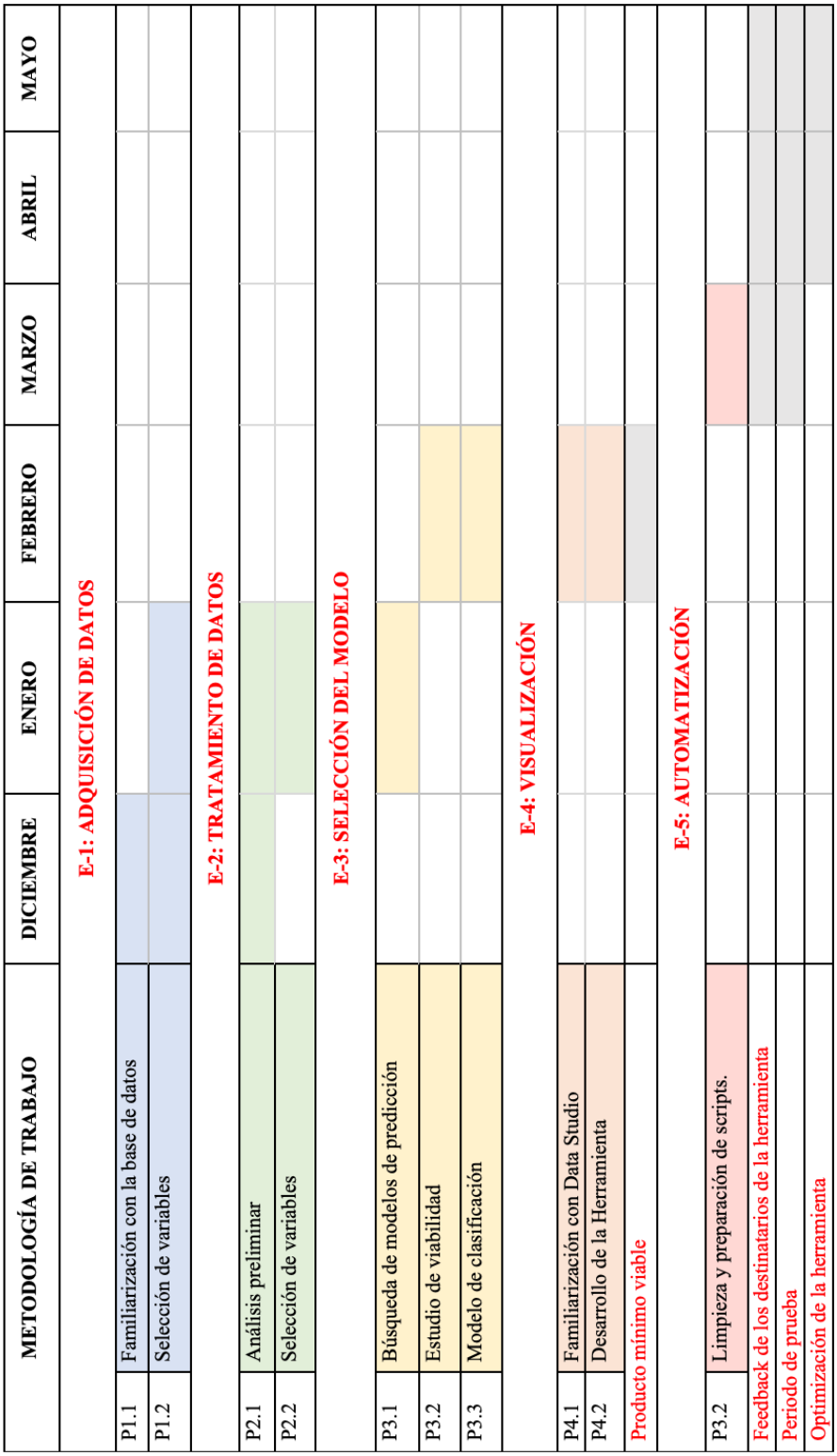
En la primera etapa, se estudiarán las diferentes tablas que la empresa pone a la disposición de este proyecto para seleccionar las variables más apropiadas para el caso de uso. Para la unión y extracción de datos se utilizará la herramienta de Bigquery.

En la etapa del tratamiento datos hay dos objetivos, la selección de los productos y la elección de la dimensión a la que se quiere realizar el proyecto. Mediante un análisis preliminar de los diferentes productos con los que trabaja Cofares se elegirán aquellos que se consideren relevantes para el caso de uso. También, se decidirá sobre que dimensión agrupar estos productos, si se quieren agrupar a nivel: cliente, código postal … o si se quieren agrupar por dimensiones más amplias como grupo terapéuticos, comunidades autónomas…

En la tercera etapa se realizará un estudio de la viabilidad de diferentes modelos de predicción. Se le dará importancia a la precisión del modelo, pero sobre todo, al tiempo de ejecución. En función de la dimensión elegida en la etapa anterior el proyecto trabajará con miles o millones de series temporales. Al trabajar con tantas series temporales, asignar una única lógica para generar el umbral que delimita la región de normalidad provoca mucho error en la detección de anomalías, por tanto, en esta etapa también se desarrollará un modelo de clasificación de las series temporales para asignar a cada grupo una lógica diferente.

Y por último en la etapa de visualización se desarrollará la herramienta con la que trabajarán los departamentos de auditoría interna y dirección del medicamento en Data Studio.

2.3 PLANIFICACIÓN



Capítulo 3. MODELO DESARROLLADO

Como ya se ha comentado antes, este proyecto esta compuesto por 4 etapas: adquisición de datos, tratamiento de datos, selección del modelo, visualización. En este apartado se explica con detalle los pasos seguidos en cada una de las etapas, así como las aproximaciones realizadas para poder implementarlas automáticamente.

3.1 ADQUISICIÓN DE DATOS

En esta etapa se ha estudiado las diferentes tablas que la empresa pone a la disposición de este proyecto para seleccionar las variables más apropiadas para el caso de uso.

Uno de los objetivos de este proyecto es aportar al departamento de auditoría interna y dirección del medicamento información reciente de los pedidos, por tanto, un factor importante para la selección de las variables ha sido la fase del proceso de ventas de la que se quiere informar. Como este proyecto se va a basar en el historial de ventas, las variables más relevantes son las cantidades vendidas de cada producto. En función de la fase del proceso de venta en la que se encuentre la variable cantidad, aportará una información u otra.

Esquema 1: Evolución de la variable cantidad en el proceso de ventas al por menor

Cantidad pedida: La cantidad que la farmacia solicita, esta cantidad es la que más probabilidades tiene de contener errores humanos y de las más interesantes a nivel de negocio ya que una anomalía en esta fase puede ocasionar grandes problemas en la empresa.

Cantidad confirmada: La cantidad que Cofares indica que va a entregar al cliente. Como la *cantidad pedida* puede venir con error a la hora de realizar el pedido (por ejemplo, añadir un cero de más por parte del farmacéutico) la empresa ya tiene un proceso de filtrado para detectar algunos pedidos anómalos. Por tanto, puede haber variaciones importantes entre esta cantidad confirmada y la cantidad pedida.

Cantidad envida: Es la cantidad que se introduce en la furgoneta y finalmente se suministra al cliente. Por posibles errores en la información del stock del almacén, esta cantidad puede variar levemente respecto a la cantidad confirmada.

Todas las cantidades son interesantes de analizar y cada una tiene casos de usos diferentes, pero cuanto más temprano se detecte la anomalía, más errores se pueden prevenir y no 'curar'. Teniendo esto en mente para la selección de variables, también es oportuno aprovechar los filtros que ya existen en la empresa para identificar pedidos anómalos.

En conclusión, por la primera condición de cuanto más temprano mejor, se seleccionaría la cantidad pedida, pero para aprovechar el filtro ya existente se ha elegido la cantidad confirmada como variable principal del proyecto.

3.2 PREPARACIÓN DE DATOS

En la etapa del tratamiento datos hay dos objetivos, la selección de los productos y la elección de la dimensión a la que se quiere realizar el proyecto. Se entiende como dimensión a la agrupación de los productos para reducir la complejidad del proyecto.

3.2.1 SELECCIÓN DE PEDIDOS

A continuación, se va a explicar el proceso seguido para la selección de los productos, este proceso analítico consiste en: representación → aislamiento →análisis.

El primer paso del proceso analítico es captar gráficamente aquellos pedidos que se desvían del comportamiento normal. Para esta captación gráfica se ha utilizado el gráfico de boxplot.

Por ejemplo, en la Figura 1 cada punto refleja una cantidad pedida de un producto específico por un cliente. En el eje X tenemos la cantidad pedida y en el eje de las Y la Comunidad Autónoma del cliente.

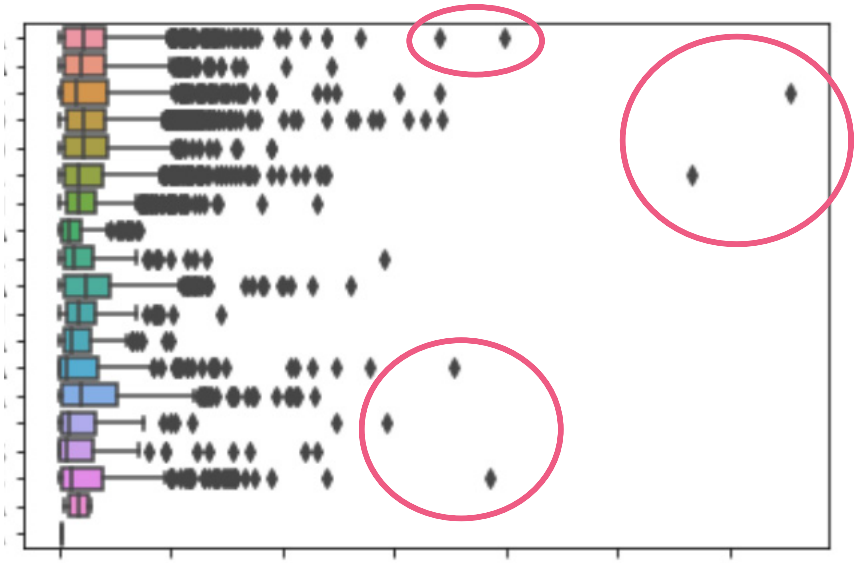


Figura 1: detección de anomalías utilizando boxplot

Como se puede ver Figura 1 gracias al boxplot se pueden ver rápidamente puntos muy alejados del resto de pedidos, estos puntos son anomalías. Una vez encontradas estas anomalías se procede a aislarlas para estudiar su naturaleza y buscar posibles patrones entre los pedidos, de esta forma se puede ver si tienen interés de negocio.

Una anomalía en los datos no tiene por qué ser una anomalía a nivel de negocio, por ejemplo, uno de esos puntos puede ser una farmacia de gran tamaño en alguna ciudad importante que debido a la diferencia de clientes respecto a otras puede destacar, pero, si se ve el historial de ventas de esa farmacia su comportamiento no es anómalo ya que suele pedir esa cantidad de ese producto con frecuencia. Esto no sería una anomalía a nivel de negocio porque no supone ninguna sorpresa para la logística de la empresa.

Esta es una de las razones por las que las series temporales son interesantes, no solo hay que analizar los pedidos entre sí, sino que también hay que comparar el pedido con el historial del cliente que lo realiza. Para simplificar un poco, se considerará anomalía aquel pedido que sea una "sorpresa" para la empresa pudiendo ocasionar problemas.

Por otro lado, sí sería una anomalía a nivel de negocio un producto que se ha pedido en una cantidad elevada por una farmacia no muy grande. Esto si que puede ser una "sorpresa" para la empresa y es necesario investigarlo.

Por ejemplo, en la

Figura *2* imagine que el eje Y son farmacias y en el eje X cantidad confirmada y cada punto es un producto pedido. Se puede ver como para la farmacia B un pedido de cantidad X_1 puede ser una anomalía ya que se aleja del comportamiento normal del resto de los pedidos que ha realizado, pero en cambio un pedido de cantidad X_1 para la farmacia A no sería tan extraño. Con esto se ilustra un ejemplo de la importancia de no solo comparar los pedidos entre sí, sino que también hay que estudiar el historial de ventas del cliente para confirmar que estamos antes una anomalía de negocio.

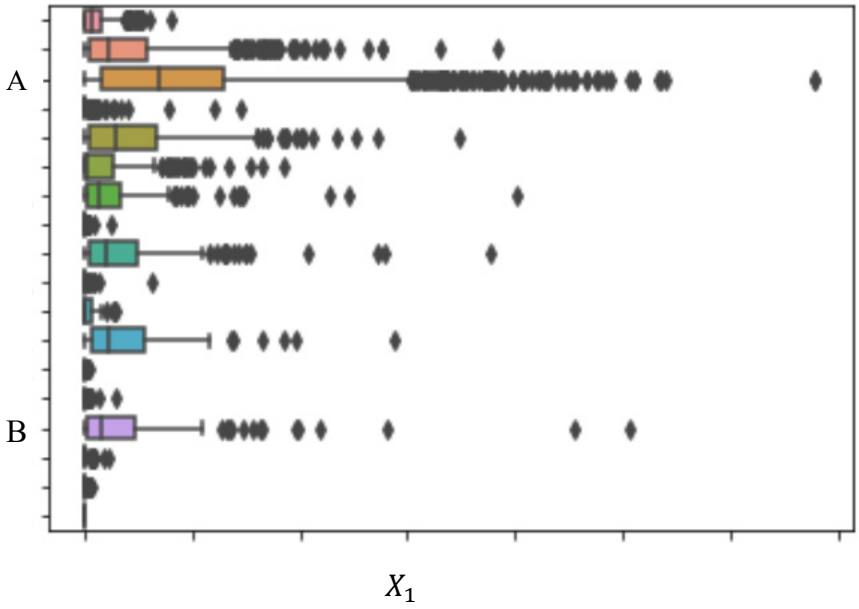


Figura 2: representación gráfica de una anomalía de negocio

Tras estudiar los productos para diferentes agrupaciones, se han visto patrones en los puntos anormales y se ha decidido centrar el análisis en solo los pedidos de medicamentos, descartando otros productos sanitarios con parafarmacia, y solo a clientes que sean farmacias, excluyendo hospitales o clínicas.

3.2.2 SELECCIÓN DE LA DIMENSIÓN

Tras haber elegido los productos que son útiles para el caso de uso, ahora, hay que decidir por qué características hay que agruparlos. Para esta decisión hay que evaluar dos factores: falta de datos y el tiempo de ejecución.

Una buena agrupación debe proporcionar para cada grupo un mínimo de datos para que el modelo se pueda ajustar correctamente al historial de ventas, es decir, hay que evitar todas aquellas agrupaciones que genere grupos cuyo historial de ventas sea en su mayoría 0

pedidos diarios. A continuación, el la Figura 3 se muestra un ejemplo de lo que hay que evitar mientras que en la Figura 4 se muestra un ejemplo de lo que se busca.

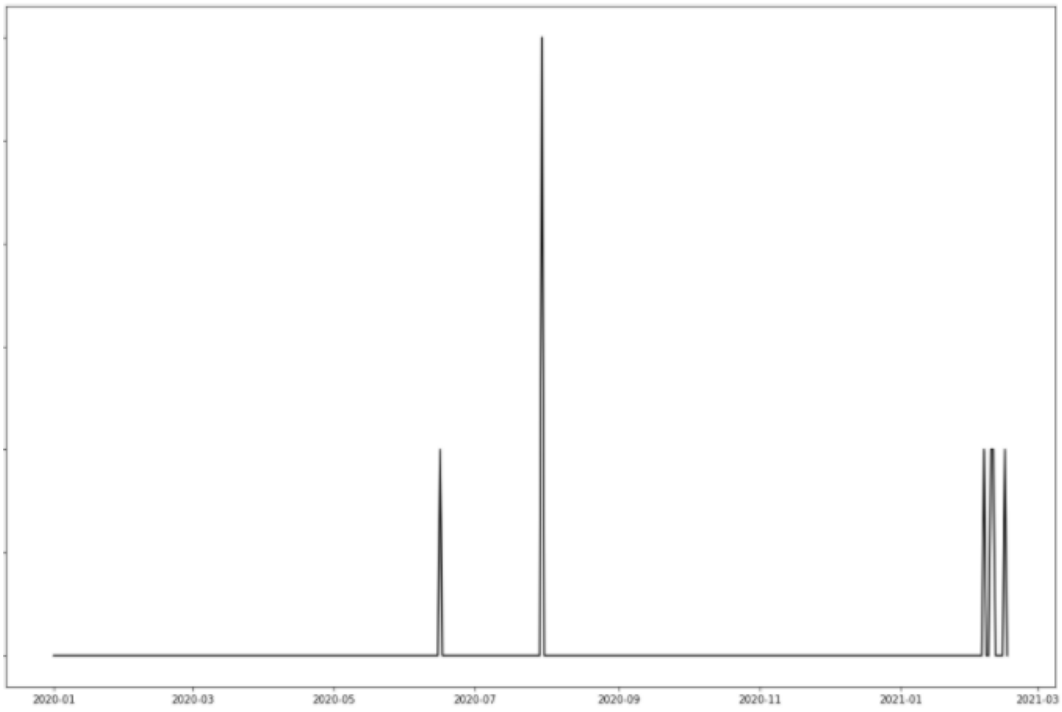


Figura 3: agrupación que se debe evitar

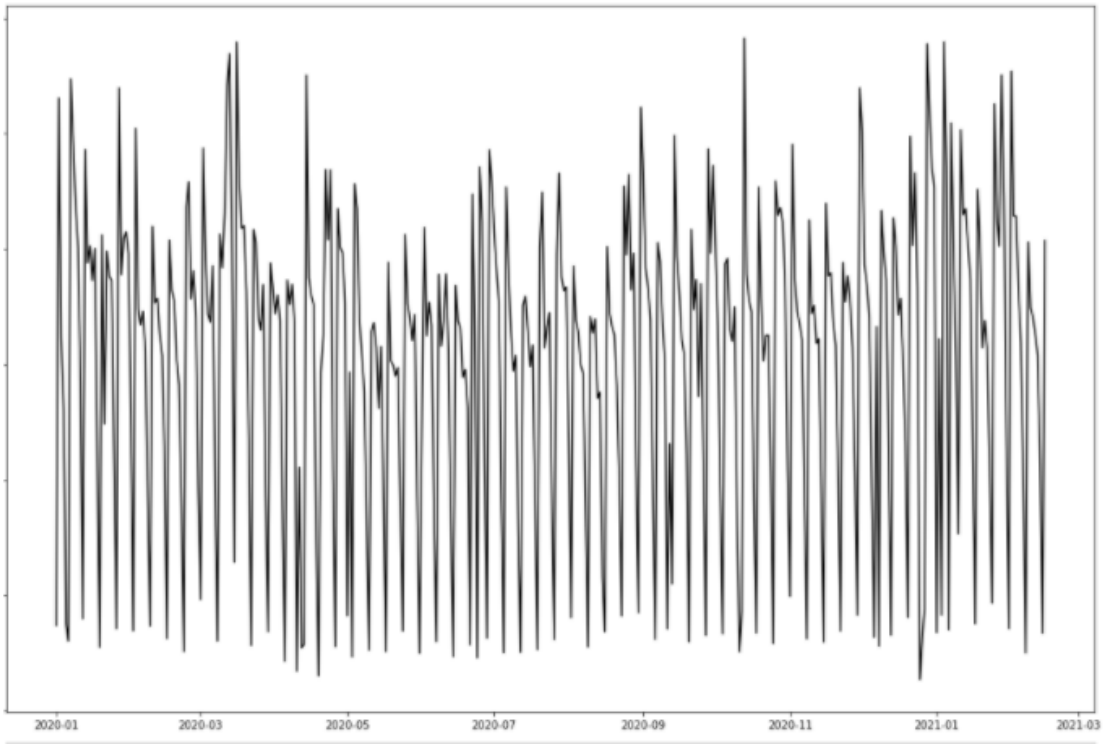


Figura 4: agrupación correcta

El segundo factor para tener en cuenta es el tiempo de ejecución. Si no hubiera muchos productos para analizar se podría ajustar una serie temporal para cada historial de cliente-producto, pero esto no es viable. Hay que buscar una agrupación que a la hora de ejecutarse tarde menos de 12h.

Tras probar diferentes agrupaciones se ha decidido que este proyecto va a detectar anomalías para los grupos Comunidad Autónoma – Grupo Terapéutico (8846 grupos). Esta agrupación proporciona información útil como la ubicación y principio activo del medicamento, también, la mayoría de los grupos tienen información suficiente como para que el modelo se pueda ajustar correctamente y se ejecuta en menos de 7h.

3.3 SELECCIÓN DEL MODELO

El objetivo de este apartado es seleccionar el modelo que se va a utilizar para la predicción de las ventas para cada agrupación Grupo Terapéutico – Comunidad Autónoma. Para ellos se realiza un estudio de viabilidad de la función autoarima y prophet, se comparan y se concluye cuál de las dos funciones se ajusta mejor a nuestro caso de uso.

Al trabajar con tantas series temporales, asignar una única lógica para generar el umbral que delimita la región de normalidad provoca mucho error en la detección de anomalías, por eso, en esta etapa también se presenta un modelo de clasificación de las series temporales para asignar a cada grupo una lógica diferente.

3.3.1 DISEÑO DE LOS MODELOS

Se parte de una tabla que tiene por filas los productos pedidos por cada cliente y por columnas información de cada pedido, dentro de esa información algunas de las variables más relevantes son:

- Fecha: fecha en la que el cliente realiza el pedido
- CantConf: cantidad que se confirma al cliente que se le va a suministrar.
- GT: grupo terapéutico del producto pedido
- CCAA: comunidad autónoma del cliente

A partir de esta tabla se crearán todos los modelos del proyecto. A continuación, se explica en primer lugar el modelo de clasificación seguido de los modelos de predicción.

3.3.1.1 Clasificación series temporales

Los dos principales problemas para la creación del modelo es el tiempo de ejecución y la falta de datos. Con el modelo de clasificación se quiere reducir el efecto de la falta de datos, por eso, se han elegido las dos siguientes variables: el número de ceros en la serie temporal (st) y la media de la cantidad confirmada de los pedidos.

- El número de ceros en la serie temporal (st): Esta variable identifica cuantos días del año no se han realizado ningún pedido. Esta información nos ayudará a identificar en qué series temporales el modelo no se va a poder ajustar correctamente y por lo tanto no van a ser muy fiables las predicciones.
- La media de la cantidad confirmada de los pedidos: Esta variable será útil para especificar el umbral que delimite la región de anomalías.

La distribución de los grupos para estas dos variables queda:

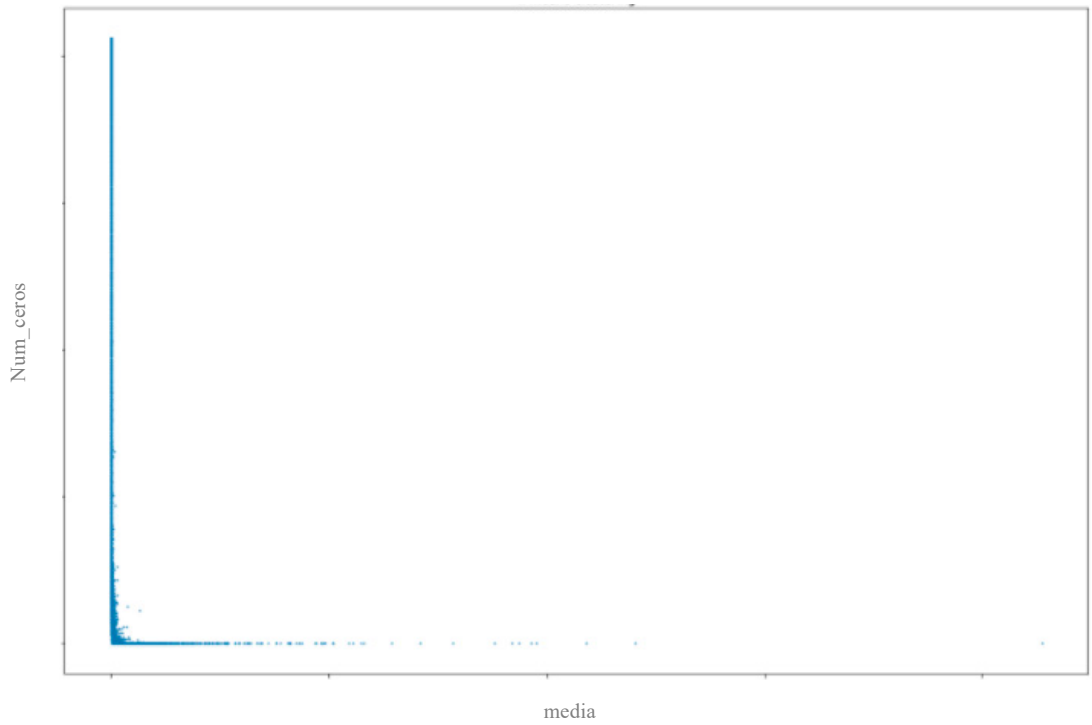


Figura 5: Distribución de las st en función de la media y num_ceros

Una vez se tiene la información de estas dos variables para todos los grupos CCAA- Grupo Terapéutico, se procede a la clasificación de las series temporales utilizando el modelo Kmeans.

Uno de los atributos que hay que definir en este modelo es el número de clústeres que se quiere generar. La selección del número óptimo de clústeres es una tarea complicada ya que

no existe una única solución óptima, sino que existen varias soluciones correctas. En este caso se ha elegido la gráfica de codo para estudiar qué número de clústeres es el más adecuado para nuestro proyecto.

Para la selección del número de clústeres se tiene la *Figura 6*. En esta figura se puede ver como el número óptimos de clústeres está entre 3 y 4. Para ver con más claridad cuál es la solución más adecuada para nuestro caso se representa los soluciones en la Figura 7 y la Figura 8.

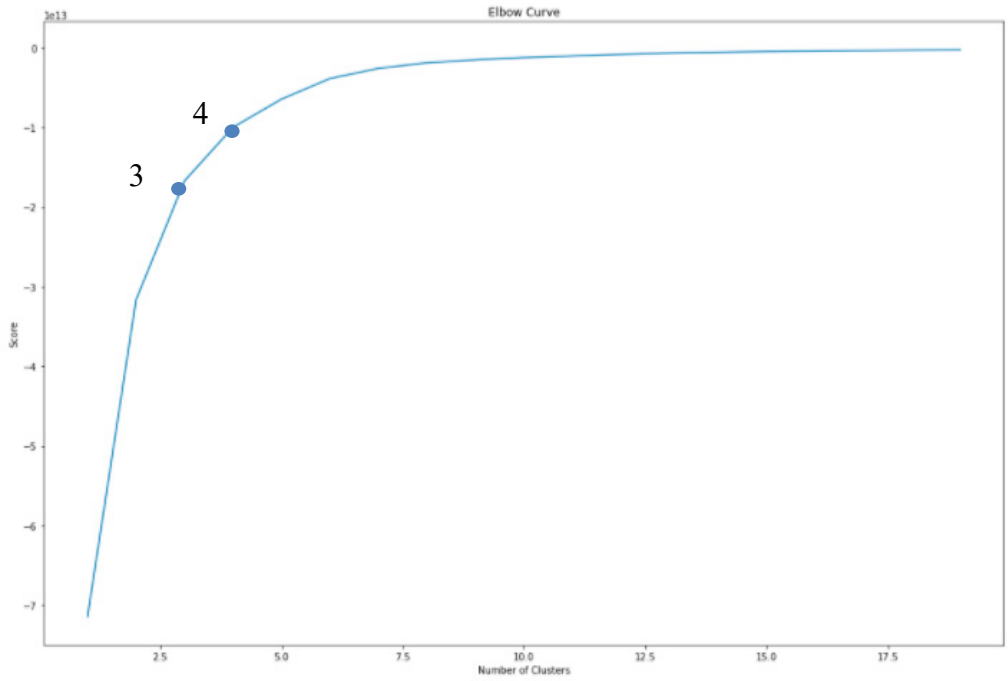


Figura 6: Gráfica del codo

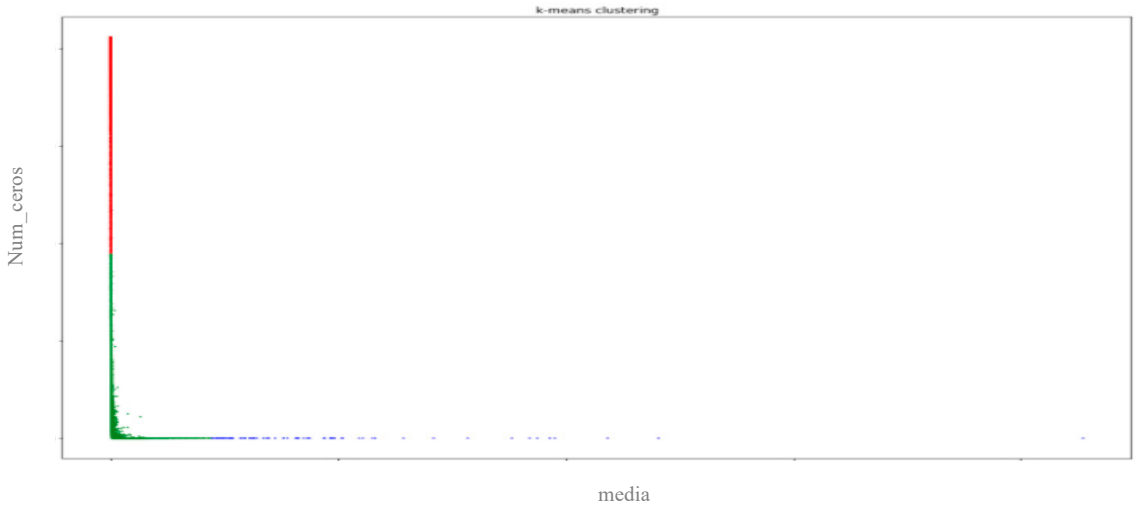


Figura 7: Resultados de agrupamiento por 3 clústeres

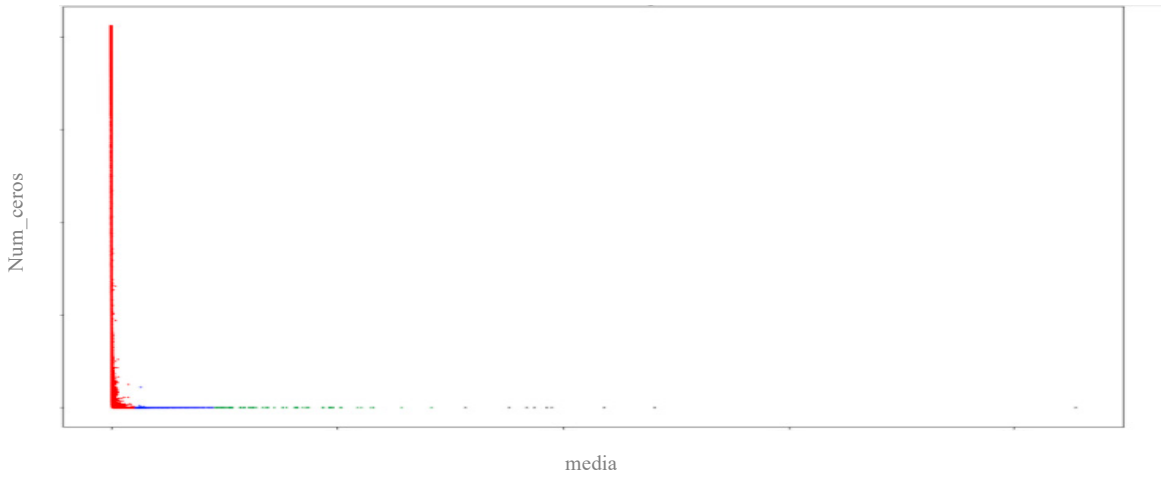


Figura 8: Resultado agrupamiento por 4 clústeres

Por un lado, se puede apreciar que en la Figura 8, los grupos no depende de la variable num_ceros, la diferenciación entre los clústeres solo depende del valor de la media. Como

uno de los objetivos de la clasificación es identificar que series no disponen de mucha información, es decir cuáles tiene muchos ceros en sus series temporales, descartamos la opción de 4 clústeres.

Por otro lado, en la Figura 7 los grupos si que dependen de ambas variables. Se puede apreciar una diferenciación según el número de ceros dividendo aquellos medicamentos que se piden más frecuentemente de las que son más puntuales y una diferenciación según la media separando aquellos medicamentos que se consumen en mayores cantidades de los que se consumen en pequeñas cantidades. De esta forma los 8846 grupos quedan clasificados de la siguiente forma:

Tabla 3: número de series temporales por cada clúster.

	Nº series temporales
Grupo 1	3521
Grupo 2	80
Grupo 3	5245

A continuación, un ejemplo de cada clúster:

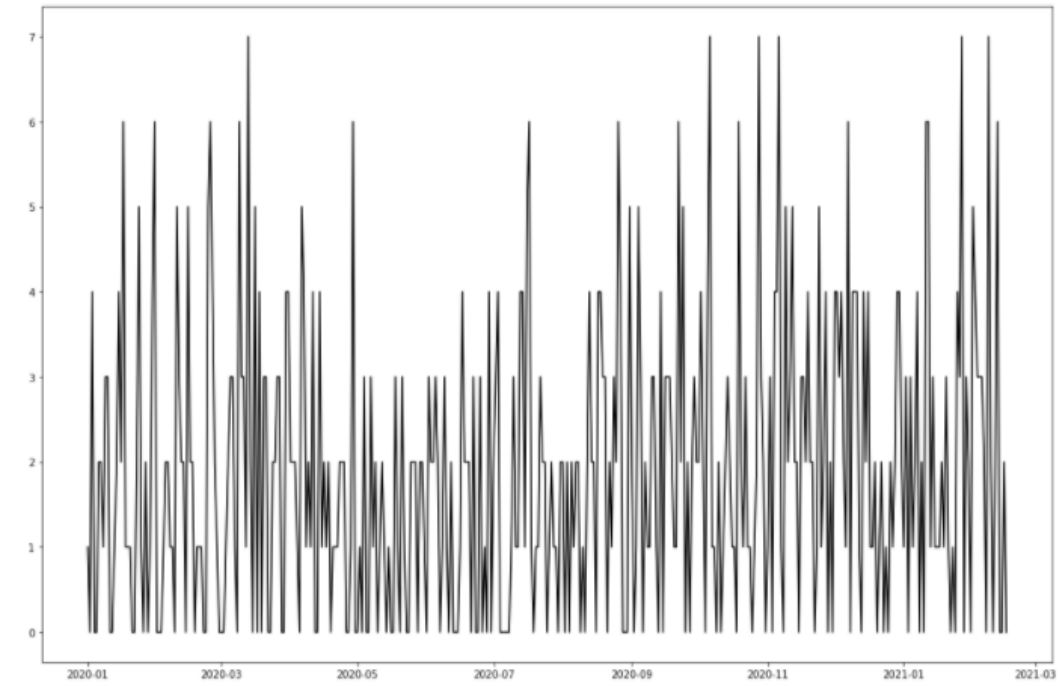


Figura 9: serie temporal del clúster 1

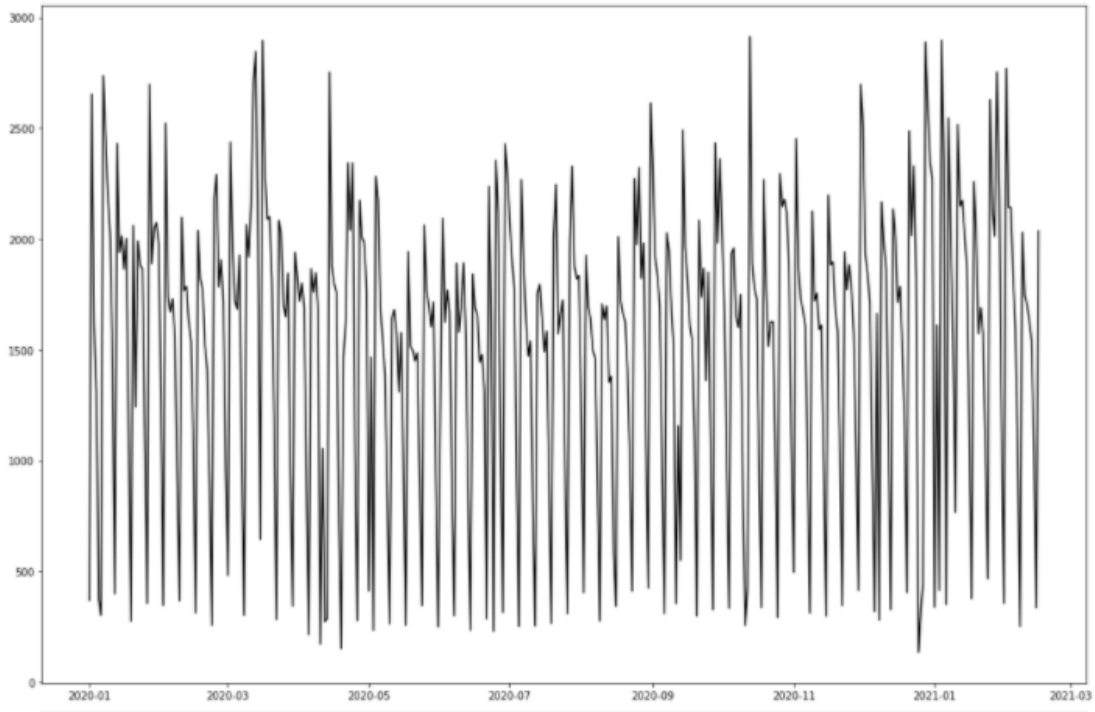


Figura 10: serie temporal clúster 2

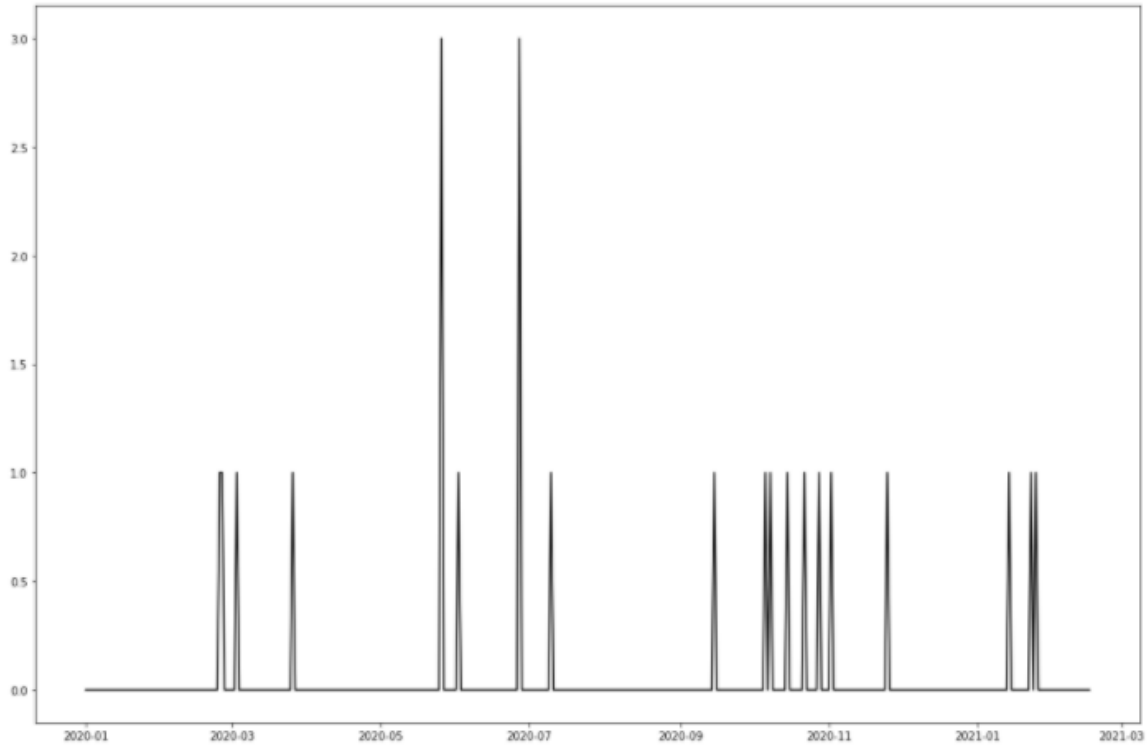


Figura 11: serie temporal clúster 3

Se puede apreciar en las anteriores figuras cómo el clúster 1 representan las series temporales de los productos que se piden casi todos los días pero en pocas cantidades, el clúster 2 identifica aquellos productos que se piden todos los días y en cantidades grandes y por último el clúster 3 que son las series temporales de aquellos productos que son muy puntuales.

3.3.1.2 Modelo de predicción

Para escoger el modelo que mejor se ajusta al historial de ventas de los grupos, se ha realizado una comparativa entre dos: autoarima y Prophet. A continuación, se estudia cuál de los dos modelos se ajusta mejor a las condiciones del problema, se evalúa las ventajas e inconvenientes y se selecciona la mejor opción.

3.3.1.2.1 Autoarima

La función auto_arima ajusta el mejor ARIMA model a una serie de tiempo univariante según AIC , AICc , BIC o HQIC . La función realiza una búsqueda (por pasos o en paralelo) sobre posibles órdenes de modelo dentro de las restricciones proporcionadas.[4]

El procedimiento seguido es el siguiente, primero se han separado los datos de las series temporales en dos grupos, training y test. El grupo de training son los datos de entrada a la función autoarima. Una vez tenemos el modelo ARIMA que mejor se ajusta a la st se hace una predicción de los datos test. Teniendo ya los datos de la predicción se comparan con los datos test y se analizan los resultados.

Para estudiar el funcionamiento de esta función se realiza una pequeña prueba con las siguientes características:

```
model = auto_arima(df_gt['y'], trace=True,m =7)
```

- `df_gt['y']`: Cantidad Confirmada para cada día del año. Es importante destacar que hay que rellenar los huecos de aquellos días en los que no se hayan realizado ningún pedido, esos huecos hay que rellenarlos con ceros.
- `trace=True`: Para imprimir por pantallas los detalles del ajuste.
- `m =7`: El periodo de diferenciación estacional, con m = 7 se indica que es mensual.

Un ejemplo de los detalles que devuelve la función para el ajuste de una serie temporal se representa en la *Figura 12*:

```
Performing stepwise search to minimize aic
 ARIMA(2,1,2)(1,0,1)[7] intercept   : AIC=inf, Time=2.03 sec
 ARIMA(0,1,0)(0,0,0)[7] intercept   : AIC=8034.077, Time=0.02 sec
 ARIMA(1,1,0)(1,0,0)[7] intercept   : AIC=7683.832, Time=0.86 sec
 ARIMA(0,1,1)(0,0,1)[7] intercept   : AIC=inf, Time=1.01 sec
 ARIMA(0,1,0)(0,0,0)[7]             : AIC=8032.079, Time=0.02 sec
 ARIMA(1,1,0)(0,0,0)[7] intercept   : AIC=8028.611, Time=0.05 sec
 ARIMA(1,1,0)(2,0,0)[7] intercept   : AIC=inf, Time=1.39 sec
 ARIMA(1,1,0)(1,0,1)[7] intercept   : AIC=7707.981, Time=1.02 sec
 ARIMA(1,1,0)(0,0,1)[7] intercept   : AIC=7879.241, Time=0.18 sec
 ARIMA(1,1,0)(2,0,1)[7] intercept   : AIC=7681.046, Time=1.97 sec
 ARIMA(1,1,0)(2,0,2)[7] intercept   : AIC=inf, Time=2.00 sec
 ARIMA(1,1,0)(1,0,2)[7] intercept   : AIC=7696.437, Time=1.61 sec
 ARIMA(0,1,0)(2,0,1)[7] intercept   : AIC=7677.863, Time=1.91 sec
 ARIMA(0,1,0)(1,0,1)[7] intercept   : AIC=7619.246, Time=0.99 sec
 ARIMA(0,1,0)(0,0,1)[7] intercept   : AIC=7892.910, Time=0.11 sec
 ARIMA(0,1,0)(1,0,0)[7] intercept   : AIC=7714.369, Time=0.44 sec
 ARIMA(0,1,0)(1,0,2)[7] intercept   : AIC=7677.859, Time=1.60 sec
 ARIMA(0,1,0)(0,0,2)[7] intercept   : AIC=7818.046, Time=0.68 sec
 ARIMA(0,1,0)(2,0,0)[7] intercept   : AIC=7674.653, Time=1.43 sec
 ARIMA(0,1,0)(2,0,2)[7] intercept   : AIC=inf, Time=1.31 sec
 ARIMA(0,1,1)(1,0,1)[7] intercept   : AIC=inf, Time=1.20 sec
 ARIMA(1,1,1)(1,0,1)[7] intercept   : AIC=7713.780, Time=1.37 sec
 ARIMA(0,1,0)(1,0,1)[7]             : AIC=inf, Time=0.35 sec

Best model:  ARIMA(0,1,0)(1,0,1)[7] intercept
Total fit time: 23.623 seconds
```

Figura 12: resultado autoarima

Como se puede ver en la *Figura 12* el modelo va probando diferentes hiperparámetros para el modelo ARIMA y se queda con el que mejor AIC obtenga, en este caso para esta serie temporal la función autorima ha estimado que el modelo que mejor se ajusta es un ARIMA(0,1,0)(1,0,1).

Se hace la prueba para 10 series temporales y el resultado de precisión y tiempo de ejecución son los siguientes:

Tabla 4: resultados prueba autoarima 10 st

	rmse	media	ratio	tiempo
0	2.595305	0.904762	2.868495	33.153006
1	3.846676	5.738095	0.670375	28.690603
2	0.158906	0.023810	6.674060	6.785869
3	140.172223	661.452381	0.211916	26.733364
4	45.342302	197.452381	0.229637	47.491807
5	1960.541743	8484.190476	0.231082	40.255469
6	234.502923	852.714286	0.275008	66.560135
7	7726.020359	15974.333333	0.483652	42.707572
8	216.295921	396.023810	0.546169	57.967325

En la *Tabla 4* se tiene por columnas la siguiente información:

- **rmse**: error cuadrático medio de los datos test con la predicción.
- **media**: media de la cantidad pedidas de la parte test.
- **ratio**: la proporción del valor rmse respecto a la media.
- **tiempo**: tiempo que se ha tardado en ajustar el modelo.

Por un lado, se puede analizar que el tiempo de ejecución tiene de media 38 segundos. Si extrapolamos esta media a los 8846 grupos nos da un tiempo de ejecución de 94 h de ejecución. Esto se pasa de los límites esperados, en ese proyecto se busca tardar menos de 12h de ejecución.

Por otro lado, analizando la precisión del ajuste del modelo de la *Tabla 4* se puede ver que solo 4 de las 9 series temporales tienen un ratio inferior al 30%, esto no es muy buen indicativo de un buen modelo. Para examinar con mayor claridad la precisión se va a representar los resultados de una serie temporal para cada grupo de la clasificación explicada en el apartado 3.3.1.1, estas series temporales son de la *Tabla 4*, las filas 0, 3 y 2 respectivamente.

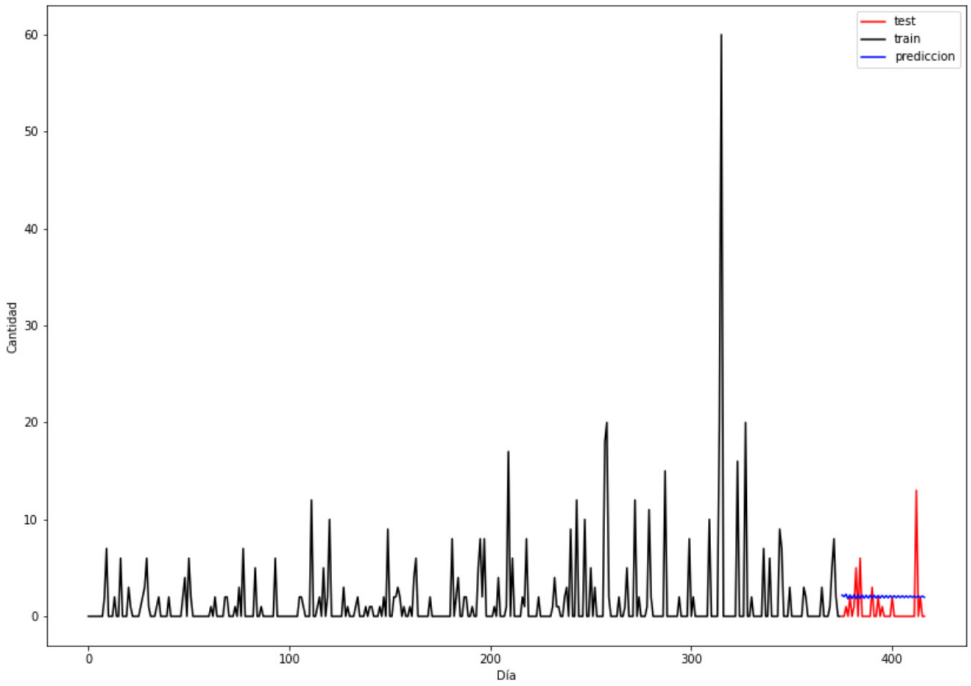


Figura 13: resultados ajuste autoarima sobre st del clúster 1

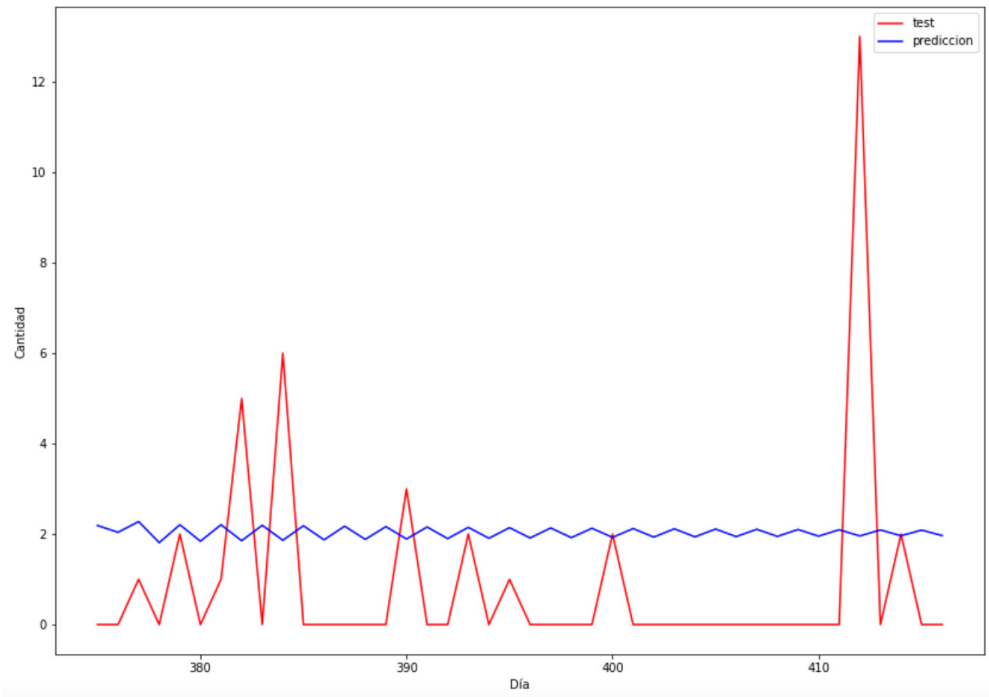


Figura 14: zoom sobre la parte test de la Figura 13

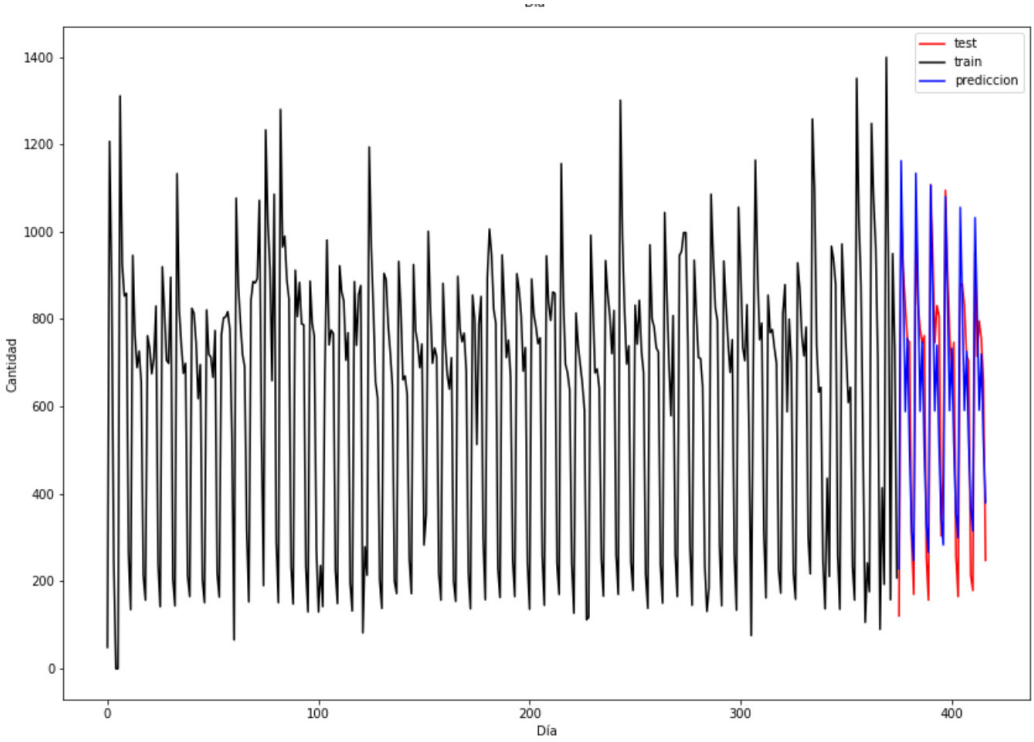


Figura 15: resultados ajuste autoarima sobre st del clúster 2

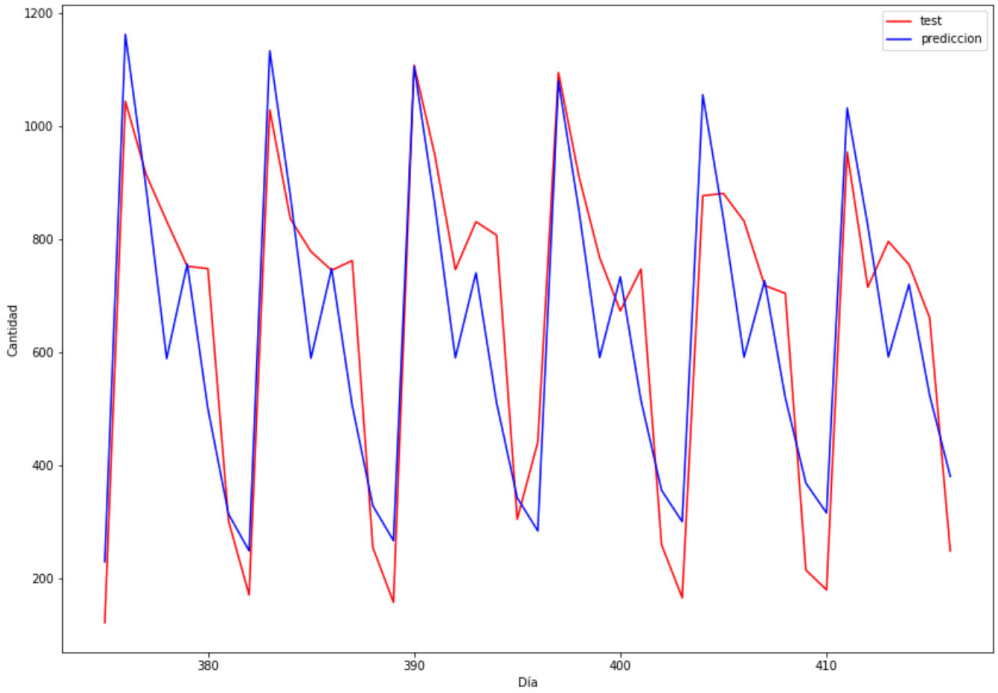


Figura 16: zoom sobre la parte test de la Figura 15

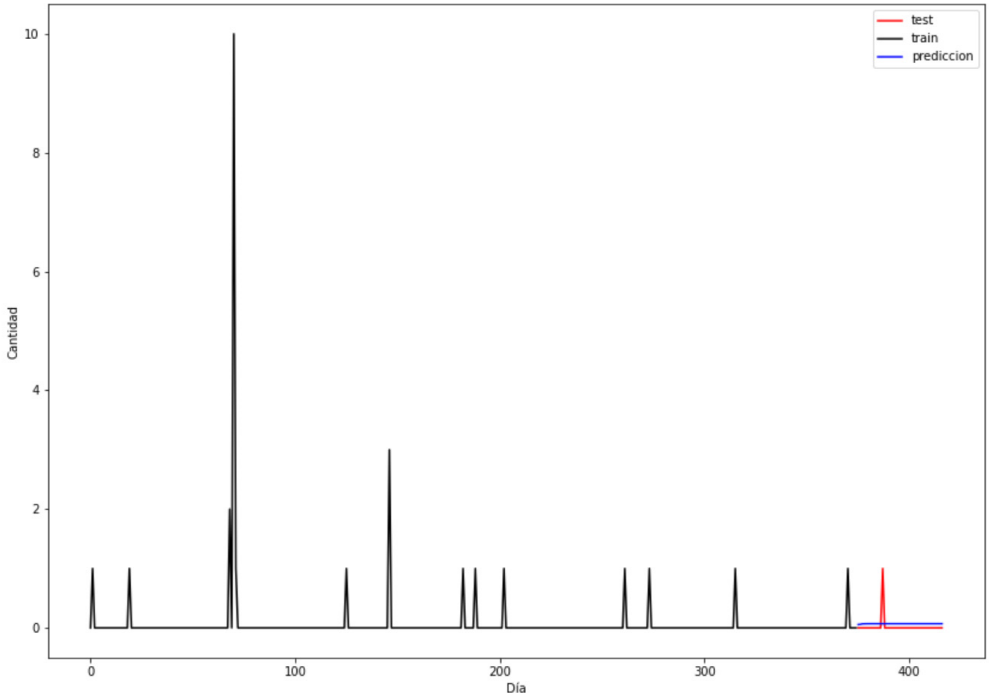


Figura 17: resultados ajuste autoarima sobre st del clúster 3

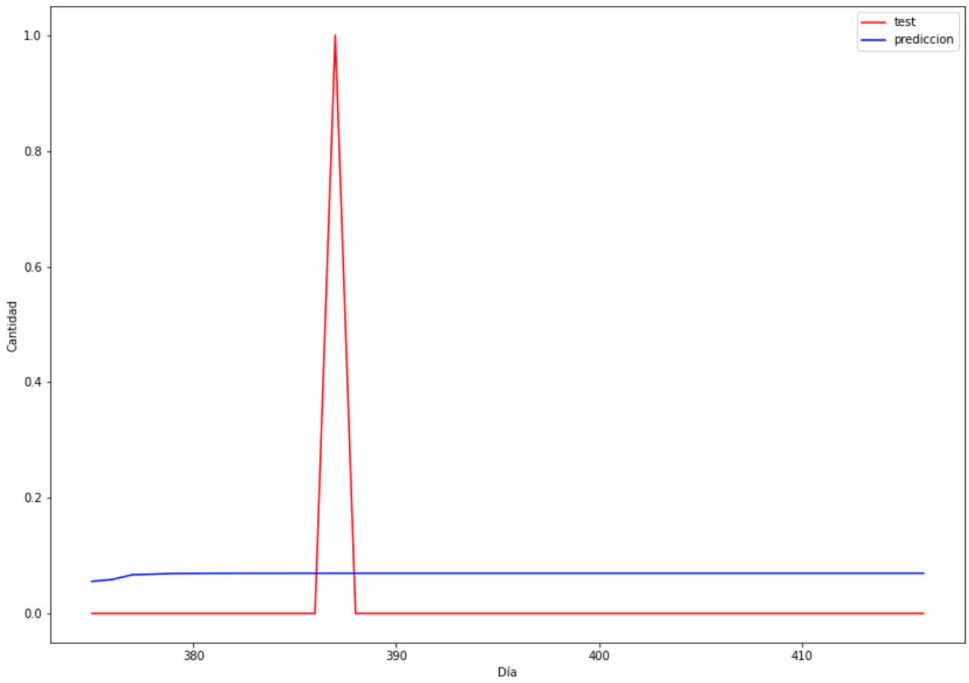


Figura 18: zoom sobre la parte test de la Figura 17

Examinando las figuras anteriores se puede ver como esta función predice bastante mal las series temporales que son poco estables, es decir a la mayoría de las series temporales del cluster 1 y 3. Por otro lado, como primera prueba para la predicción de las st del cluster 2 está bastante bien.

A continuación, se realiza el mismo procedimiento seguido con la función autoarima pero ahora con la función Prophet.

3.3.1.2.2 Prophet

Prophet es una función automática que busca el mejor conjunto de hiperparámetros para realizar predicciones de series de tiempo basado en un modelo aditivo donde las tendencias no lineales se ajustan a la estacionalidad anual, semanal y diaria, más los efectos de las vacaciones.[5]

Al igual que con autoarima el procedimiento seguido ha sido el siguiente. Se han separado los datos de las series temporales en training y test, los datos training se han utilizado para ajustar el modelo que posteriormente se ha utilizado para predecir los datos test. Una vez se tienen las predicciones se han comparado los datos test y se han analizado los resultados.

Como primera prueba se ha decido utilizar los valores que vienen en por defecto en la función, pero especificando una estacionalidad mensual:

```
model = Prophet().add_seasonality(name = 'monthly', period = 30.5, fourier_order = 5)
```

- `period = 30.5`: número de días en un período.
- `fourier_order = 5`: número de componentes de Fourier a utilizar.

Se hace la prueba para las mismas 10 series temporales con las que se evaluó la función autoarima y los resultados son lo siguientes:

Tabla 5: Resultados ajuste función Prophet

	rmse	media	ratio	tiempo
0	3.197915	0.904762	3.534537	2.361084
1	4.006063	5.738095	0.698152	2.454201
2	0.224981	0.023810	9.449191	2.712605
3	75.825243	661.452381	0.114634	2.499854
4	36.365709	197.452381	0.184175	2.528173
5	1239.977315	8484.190476	0.146152	2.676439
6	291.444407	852.714286	0.341784	2.561660
7	2228.988938	15974.333333	0.139536	2.601567
8	63.685094	396.023810	0.160811	2.574198

En la *Tabla 5* se tiene por columnas la siguiente información:

- **rmse**: error cuadrático medio de los datos test con la predicción.
- **media**: media de la cantidad pedidas de la parte test.
- **ratio**: la proporción del valor rmse respecto a la media.
- **tiempo**: tiempo que se ha tardado en ajustar el modelo.

En cuanto al tiempo de ejecución se puede ver que de media es de 2.5 segundos, que si extrapolamos a las 8846 series se estima que tardará un poco más de 6 horas. Cumple la condición impuesta en el proyecto de duración menor a 12h de ejecución.

Por otro lado, comentando el error, destaca que más del 50% de las st tienen un error menor al 20 %, esto sí que es un buen indicativo de un modelo aceptable. Para analizar la precisión, se representa una serie temporal por cada clúster. Estudiando las siguientes figuras se puede ver que al igual que ocurría con autoarima, prophet se ajusta correctamente al clúster 2 pero muy mal al clúster 1 y 3.

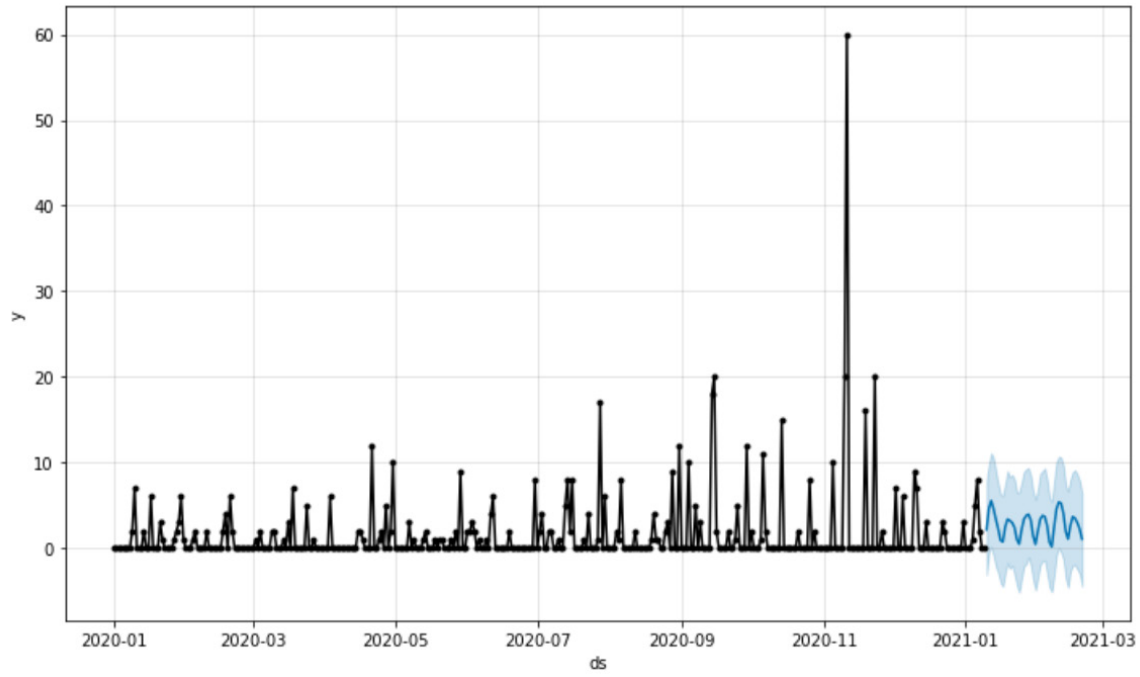


Figura 19: resultados ajuste prophet sobre st del clúster 1

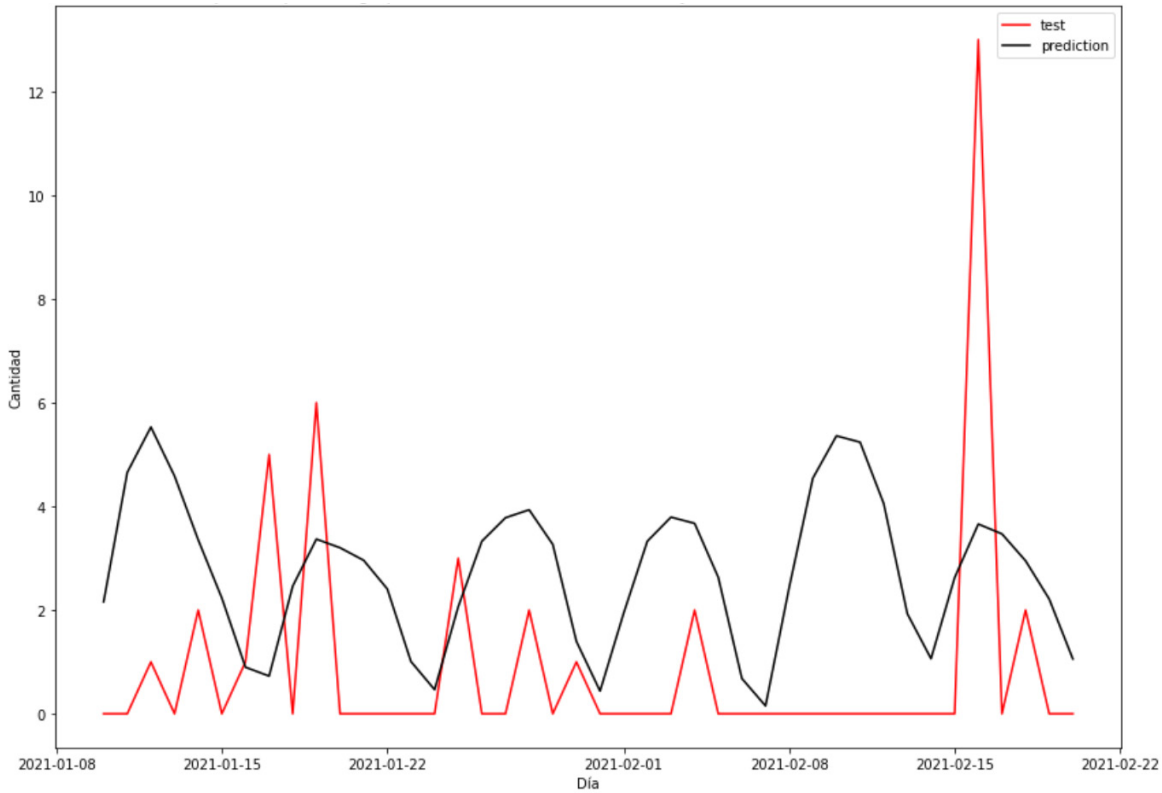


Figura 20: zoom sobre la parte test de la Figura 19

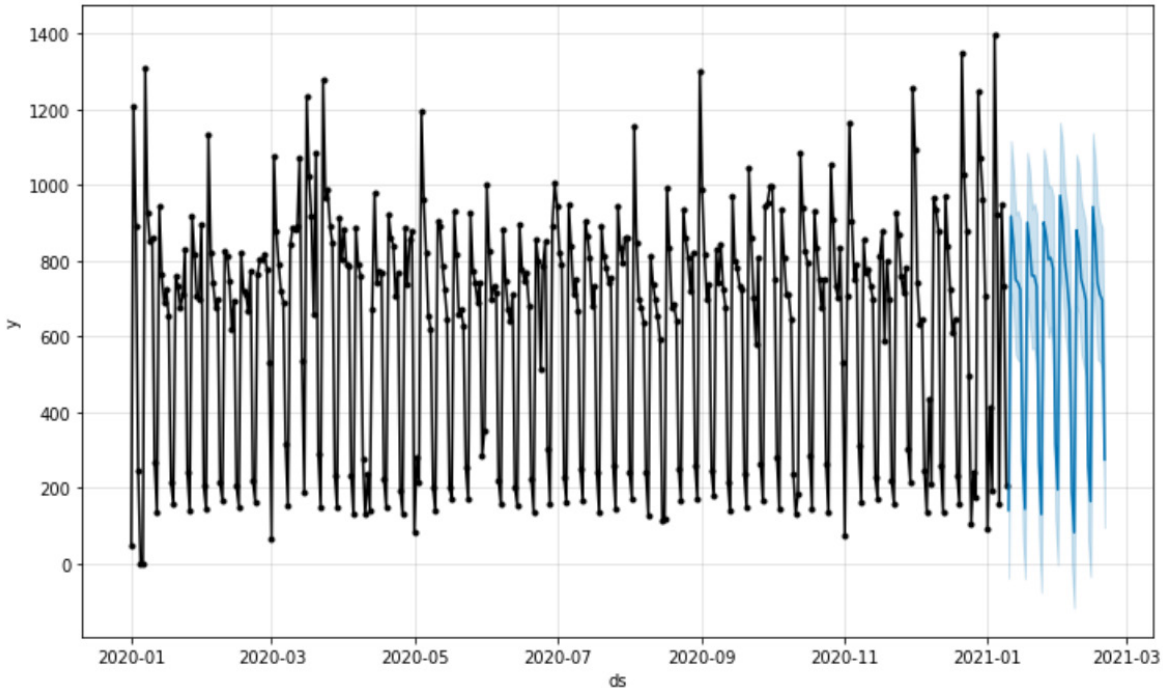


Figura 21: resultados ajuste prophet sobre st del clúster 2

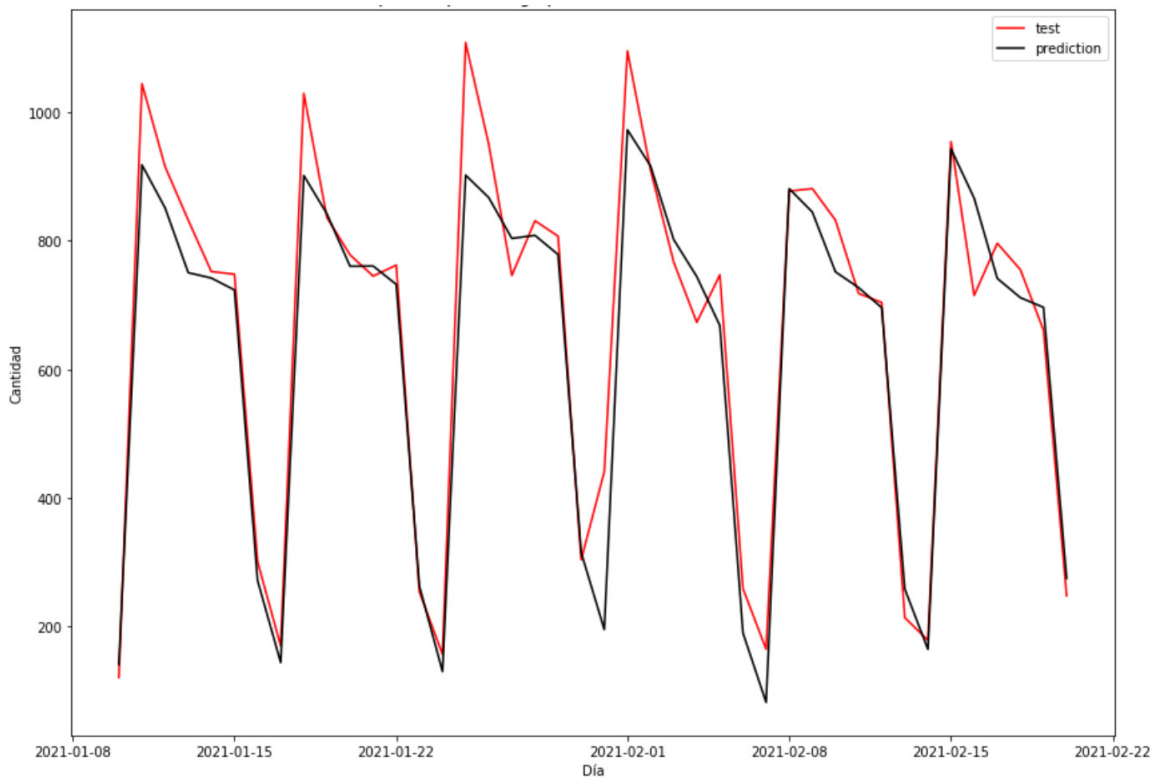


Figura 22: zoom sobre la parte test de la Figura 21

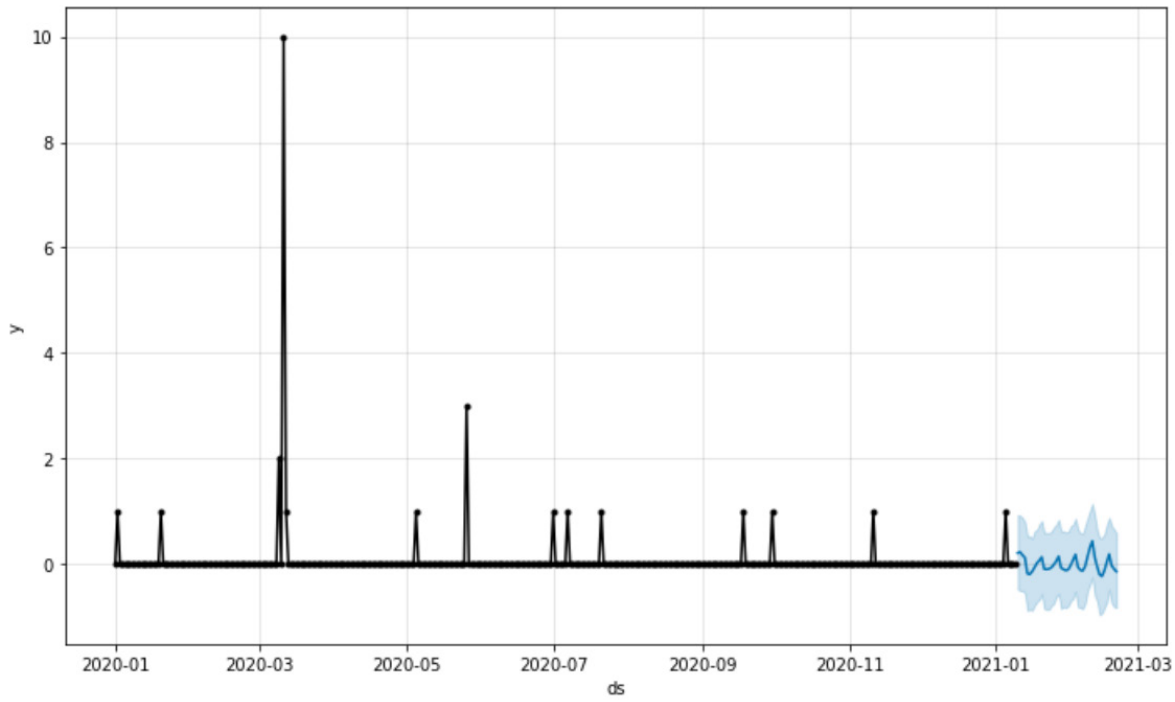


Figura 23: resultados ajuste prophet sobre st del clúster 3

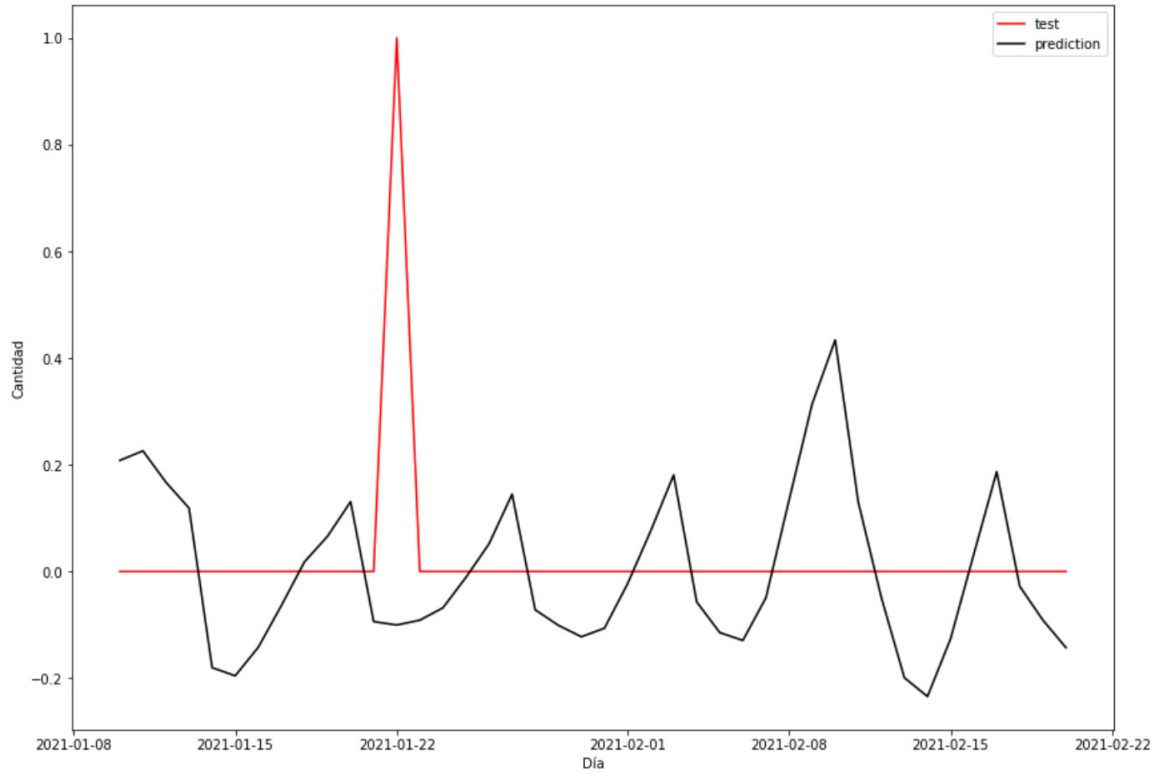


Figura 24: zoom sobre la parte test de la Figura 23

3.3.2 COMPARATIVA

A continuación, se comparan los resultados de ambas funciones y de concluye cuál es más adecuada para nuestro caso de uso.

AUTOARIMA

	rmse	media	ratio	tiempo
0	2.595305	0.904762	2.868495	33.153006
1	3.846676	5.738095	0.670375	28.690603
2	0.158906	0.023810	6.674060	6.785869
3	140.172223	661.452381	0.211916	26.733364
4	45.342302	197.452381	0.229637	47.491807
5	1960.541743	8484.190476	0.231082	40.255469
6	234.502923	852.714286	0.275008	66.560135
7	7726.020359	15974.333333	0.483652	42.707572
8	216.295921	396.023810	0.546169	57.967325

PROPHET

	rmse	media	ratio	tiempo
0	3.197915	0.904762	3.534537	2.361084
1	4.006063	5.738095	0.698152	2.454201
2	0.224981	0.023810	9.449191	2.712605
3	75.825243	661.452381	0.114634	2.499854
4	36.365709	197.452381	0.184175	2.528173
5	1239.977315	8484.190476	0.146152	2.676439
6	291.444407	852.714286	0.341784	2.561660
7	2228.988938	15974.333333	0.139536	2.601567
8	63.685094	396.023810	0.160811	2.574198

Para la selección del modelo hay que tener en cuenta dos factores: el tiempo de ejecución y la precisión del modelo, en este caso las columnas tiempo y ratio.

Un tiempo de ejecución aceptable tiene que ser menor a 4,5 segundo por serie temporal para que la ejecución total dure menos de 12 h y una precisión aceptable tiene que ser cuanto más baja mejor.

Comparando ambas tablas se puede ver que, para la primera condición, está claro que prophet supera con diferencia a la función autoarima. La media de prophet está cercana a 2, mientras que la de autoarima casi llega a los 40 segundos, es decir, tarda casi 20 veces más de lo que tarda prophet. Por tanto, si tenemos solo en cuenta esta condición, la función prophet es las más adecuada.

Ahora se compara la segunda condición, precisión. Para poder comparar los errores de las st se ha calculado el ratio entre el rmse de cada serie con su media, no es lo mismo tener un rmse de 2 cuando el pedido es de 100 unidades que cuando el pedido es de 5. Utilizando la columna de ratio destaca que la función prophet tiene mejores resultados que la función

autoarima para las series temporales con medias altas, en cambio cuando las medias son bajas, la función autoarima se ajusta mejor. Hay que tener en cuanta que la cantidad de series con medias bajas es más del 80 % del total. Hay que analizar con más detalle la precisión de estas funciones en los clústeres 1 y 3. Para ello se va a realizar otra prueba con ambas funciones, pero esta vez con 100 series temporales.

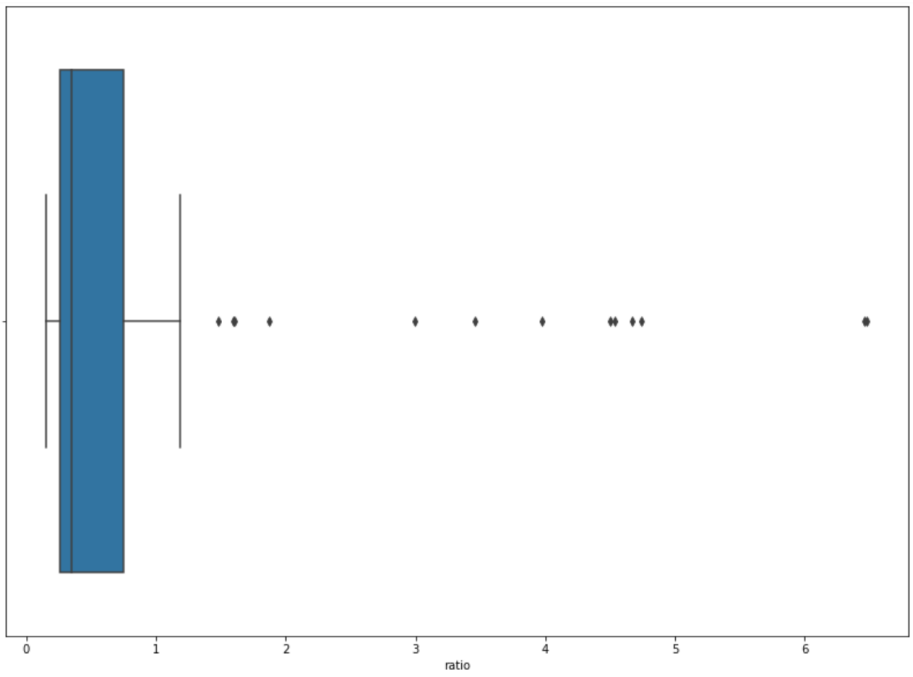


Figura 25: resultados de la función autoarima con 100 st

Para las 100 series temporales tenemos que la media del ratio es de 0.89 y la mediana de 0.34.

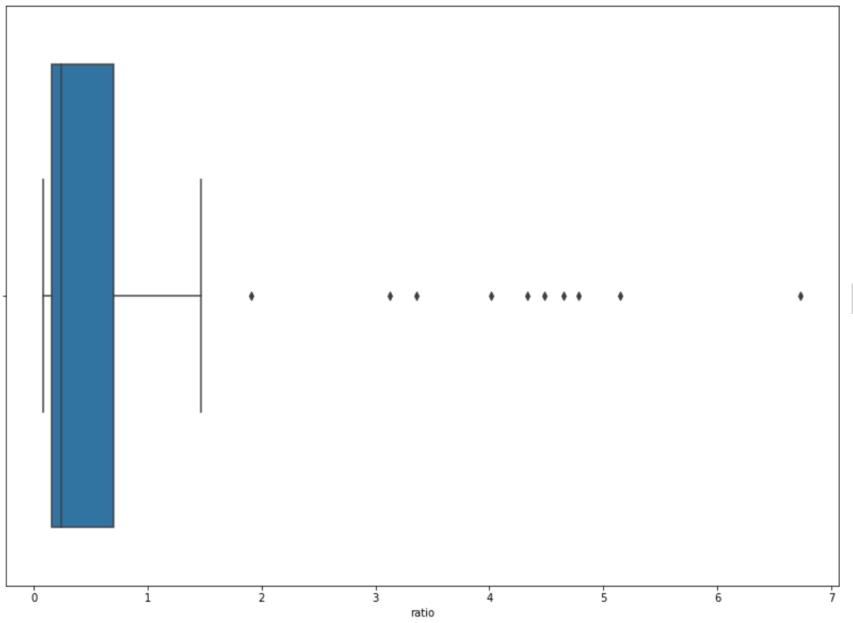


Figura 26: resultados de la función prophet con 100 st

Para las 100 series temporales tenemos que la media del ratio es de 0.80 y la mediana de 0.24.

Juntando estos resultados en una tabla comparativa:

Tabla 6: comparativa de resultados de 100 st

	AUTOARIMA	**PROPHET**
MEDIA	0.89	0.8
MEDIANA	0.34	0.24

Analizando las *Figura 25* y *Figura 26* a simple vista no se ve grandes diferencias, pero en cambio si comparamos los resultados de la tabla *Tabla 6* se puede apreciar que la mediana de prohet esta 0.1 por debajo de la de autoarima. Esto significa que el 50% de las series temporales de prophet se encuentran en un ratio por debajo de 0.24.

En conclusión, viendo que la función prophet tiene una gran diferencia de tiempo de ejecución respecto a autoarima y que a pesar de tener menos exactitud en las series temporales del cluster 1 y cluster 3, las series temporales del modelo tiene un ratio de precisión 0.1 puntos por debajo de la función autoarima. Se concluye que para este caso de uso se va a utilizar la función prophet.

Se ha ajustado las 8846 st y la media del tiempo de ejecución han sido de 2 segundos, la media del ratio es 1.52 y la mediana 0.69. La distribución del ratio queda de la siguiente manera:

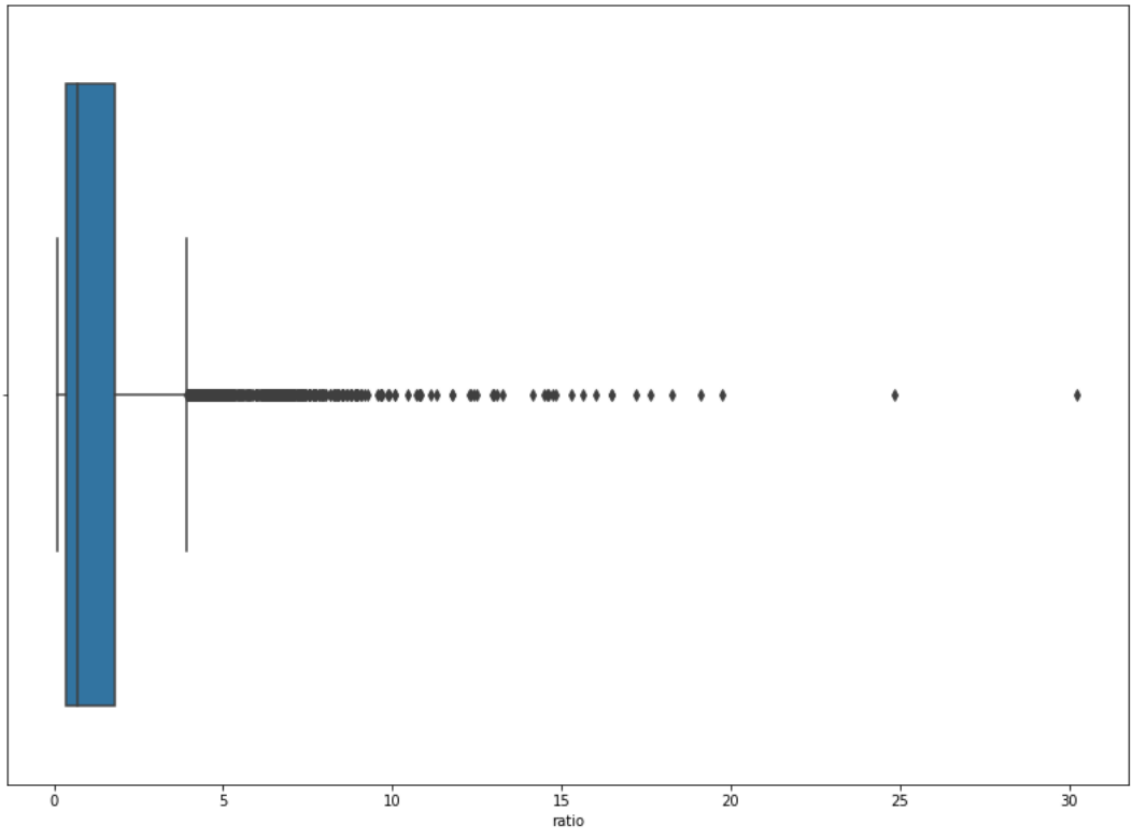


Figura 27: resultados de la función prophet con todas las st

Los resultados finales han quedado un poco peor de lo esperado, ya que con la prueba de 100 st la mediana estaba en 0.24 y ahora en 0.69.

3.4 DETECCIÓN DE ANOMALÍAS

El objetivo de este apartado es definir la zona de 'normalidad'. Para ello, se utiliza el modelo de predicción diseñado en el apartado anterior, se estima las ventas a futuro y con esta predicción se calcula un umbral que delimite la región de 'normalidad'. Todas aquellas ventas diarias que superen este umbral se clasificarán como anomalías.

3.4.1 DEFINICIÓN DEL UMBRAL

Como se ha explicado en la introducción, este proyecto va a servir como herramienta para el departamento de auditoría interna pueda identificar, investigar y seguir de cerca los pedidos anómalos, por tanto, esta decisión del umbral va a estar ligada a las características de trabajo de este departamento.

Este departamento utiliza los recursos que la empresa pone a su disposición para encontrar y analizar los pedidos más anómalos. Por temas de limitaciones de personal y tiempo no pueden analizar todos los pedidos que les gustaría, entonces, han solicitado que cada día se detecte un número reducido de anomalías. De esta forma, por un lado, no se sobrecarga de información a los auditores permitiendo que analicen las anomalías y por otro se puede ir comprobando si el modelo las detecta correctamente e ir trabajando con el *feedback*.

Como primera prueba se va a seleccionar el umbral como 3 veces la predicción que devuelva el modelo. Es decir, si un día la suma de todos los pedidos de un grupo CCAA-GT es mayor a 3 veces el valor que predice el modelo, se detectará como anomalía.

Para tener una idea más clara de lo que se está buscando se representan algunos ejemplos de las anomalías detectadas en algunas de las series temporales. Como se puede ver en la leyenda de las figuras, en color rojo se tiene la predicción del modelo, en negro el valor real y en azul el límite impuesto, las anomalías son las estrellas verdes.

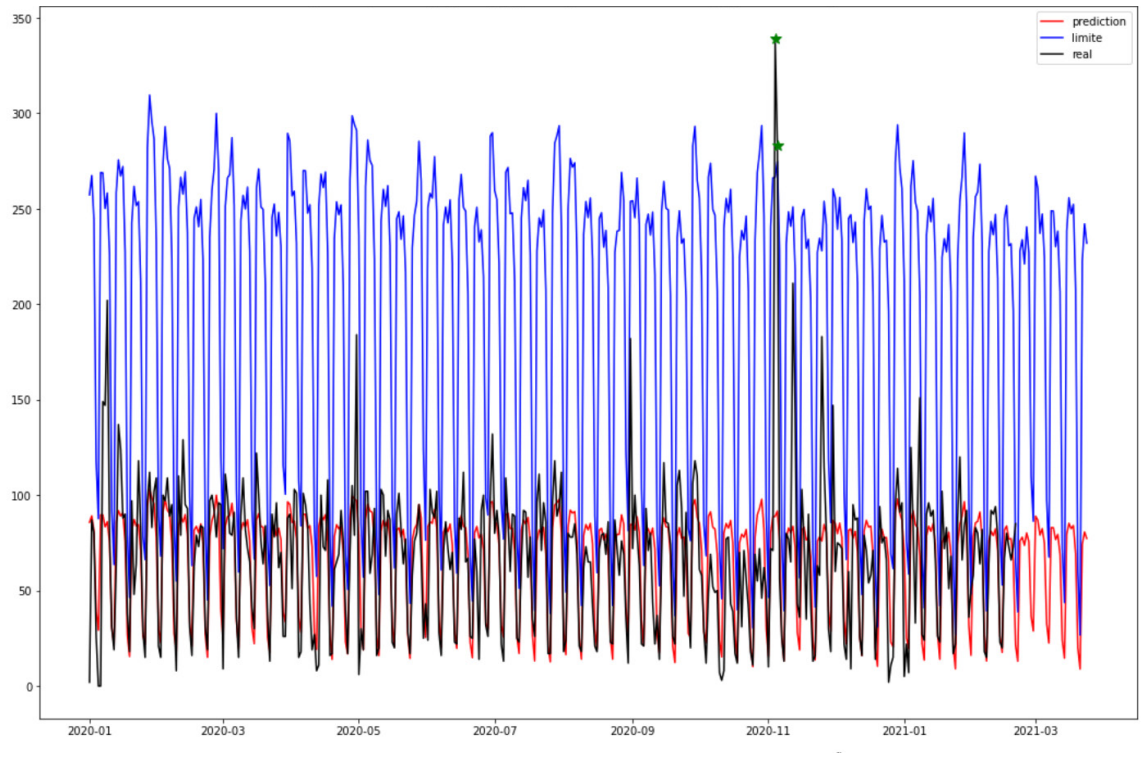


Figura 28: detección de anomalía con límite superior x3

En la *Figura 28* las anomalías detectadas son picos muy marcados en la serie temporal, en este caso se encuentran en el mes 11 del 2020. Se pueden apreciar más picos en el historial de ventas, pero al no haber tanta diferencia con respecto a su serie temporal, no tienen interés de negocio, por ese se ha puesto un índice de 3, para que detecte solo lo picos muy exagerados.

Esta lógica funciona muy bien para las series temporales que son muy estables, pero no se puede decir lo mismo de las otras. A continuación, se muestran algunos de los fallos encontrados.

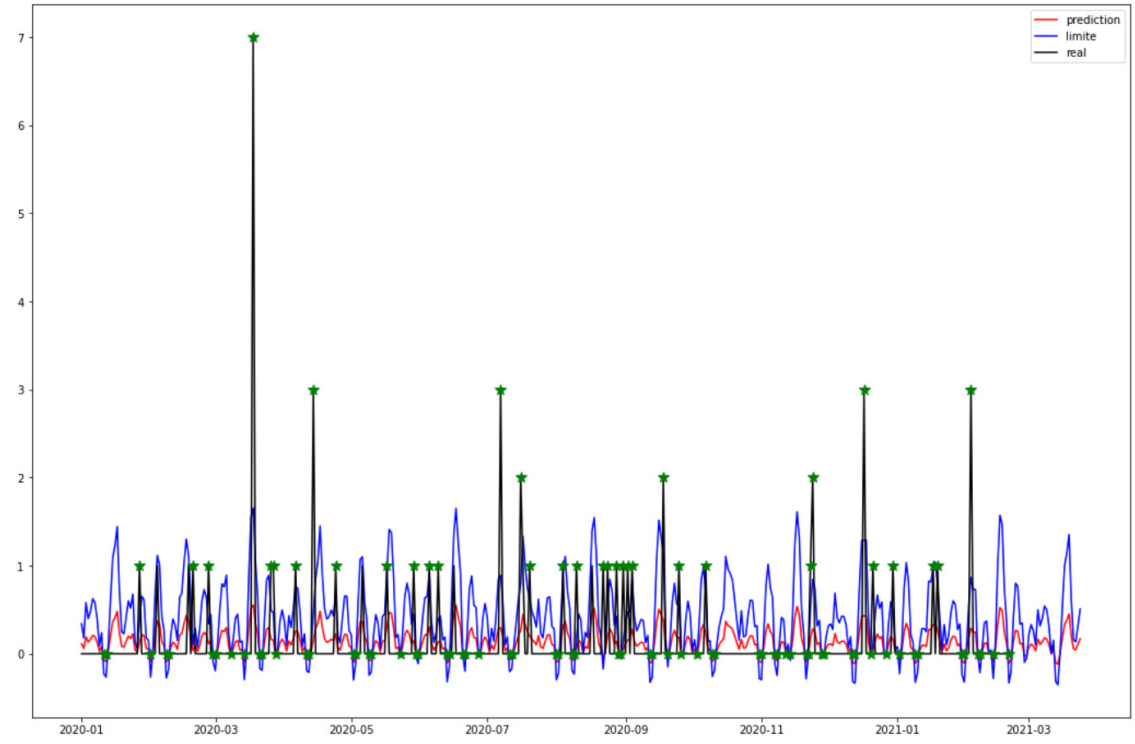


Figura 29: mal funcionamiento de la detección de anomalías en st poco estable

En la *Figura 29* hay varios fallos. Primero el modelo no se ajusta bien a la st y predice con un error muy alto, incluso hay días que estima una venta negativa provocando que cualquiera que sea la venta real va a salir mayor a la predicción y por tanto anomalía, por eso el gran número de estrellas verdes en el valor 0. También se ve que prácticamente todos los días que se ha vendido, incluso una unidad, se ha detectado como anomalía. Este es un claro ejemplo del mal funcionamiento de la detección de anomalías en series temporales poco estables.

Como ya se explico anteriormente, una de las razones por la que se realizó la clasificación de las st fue para poder separar las st más estables de las menos y de esta forma generar diferentes lógicas para calcular el umbral.

A continuación, se va a explicar por separado el cálculo del umbral para cada uno de los cústers.

3.4.1.1 Cluster 1 & cluster 3

Las características de estos clusters son series temporales de pedidos de pocas unidades y muchos de ellos poco frecuentes, por tanto, el modelo que se ha ajustado para la predicción no va a ser un gran indicativo para calcular las anomalías. La lógica que se va a seguir es la siguiente:

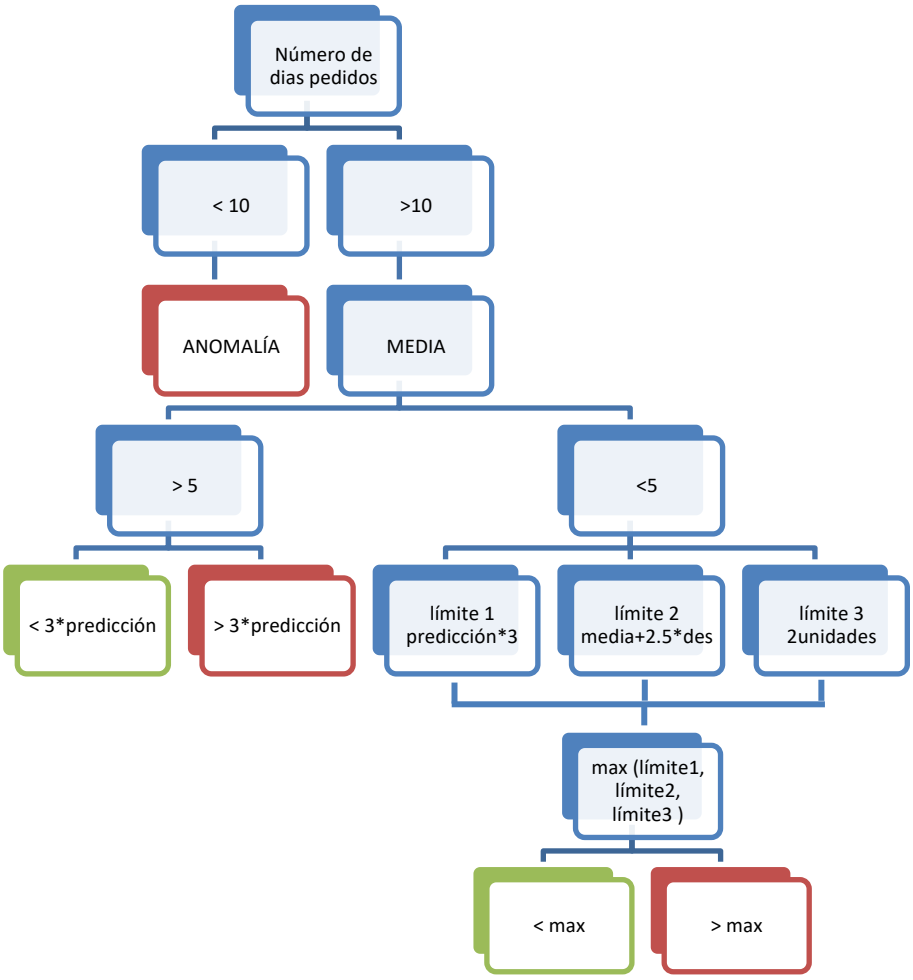


Figura 30: esquema detección de anomalías clúster 1

Lo primero que se mide para ver si el pedido diario es una anomalía es contar cuantos días del año se han realizado pedidos de esta st, si el número de días es menor a 10, se considera que pertenece a un grupo de medicamentos atípico y por tanto se indicará como anomalía.

De esta forma se puede realizar un seguimiento del pedido y controlar que no se acabe el stock del almacén.

Si el número de veces que se ha realizado un pedido de este grupo es mayor a 10 días al año lo siguiente que se mide es la media de las cantidades pedidas. Si la media es mayor a 5 se considera que la serie puede ser estable y se calcula el límite como 3 veces la predicción del modelo. En caso de que la venta sea mayor a este límite se considera anomalía.

Si la media es menor a 5 unidades, se considera que la series sigue siendo poco estable y se utilizarán tres diferentes formas de calcular el límite. El máximo de estos tres límites será el umbral que defina la región de normalidad. Si la cantidad del pedido está por encima de este umbral, se considera pedido anómalo. Estos tres límites son:

- **3*predicción**: el límite será 3 veces la predicción del modelo.
- **Media+2.5*des**: se considera que todas las cantidades que se encuentren en el rango de [0, media+2.5*des] es un pedido normal. La media y la desviación estándar se calculan utilizando las cantidades de todos los días de la serie temporal, no solo de los días que se realizan un pedido.
- **2unidades**: hay series temporales que tanto el límite 1 como el límite 2 dan un valor por debajo de 1 (esto pasa en las series temporales cuyo valor habitual de pedido es cantidad 1) provocando que se detecte como anomalía cualquier pedido que se haga. Para que esto no pase, se pone el límite a 2 unidades, es decir, que se identifique como anomalías todos aquellos que compren 3 o más unidades (que sería tres veces la cantidad normal pedida).
-

A continuación, algunos ejemplos de la detección de anomalías con los límites que se han comentado anteriormente.

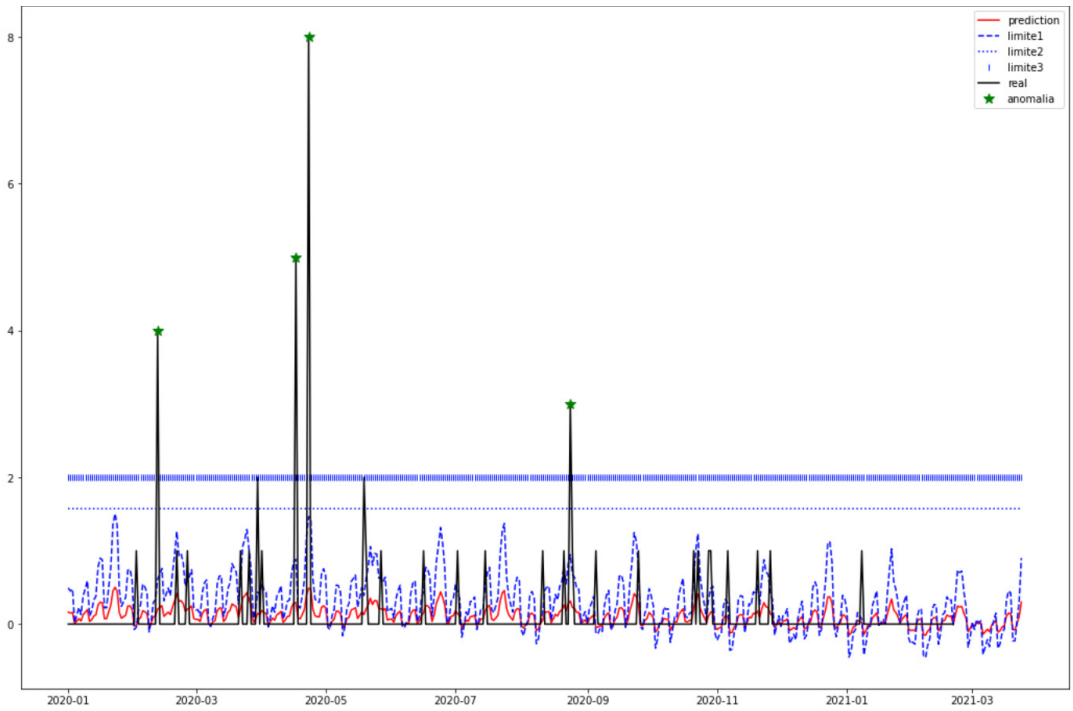


Figura 31: cálculo del umbral para el clúster 3

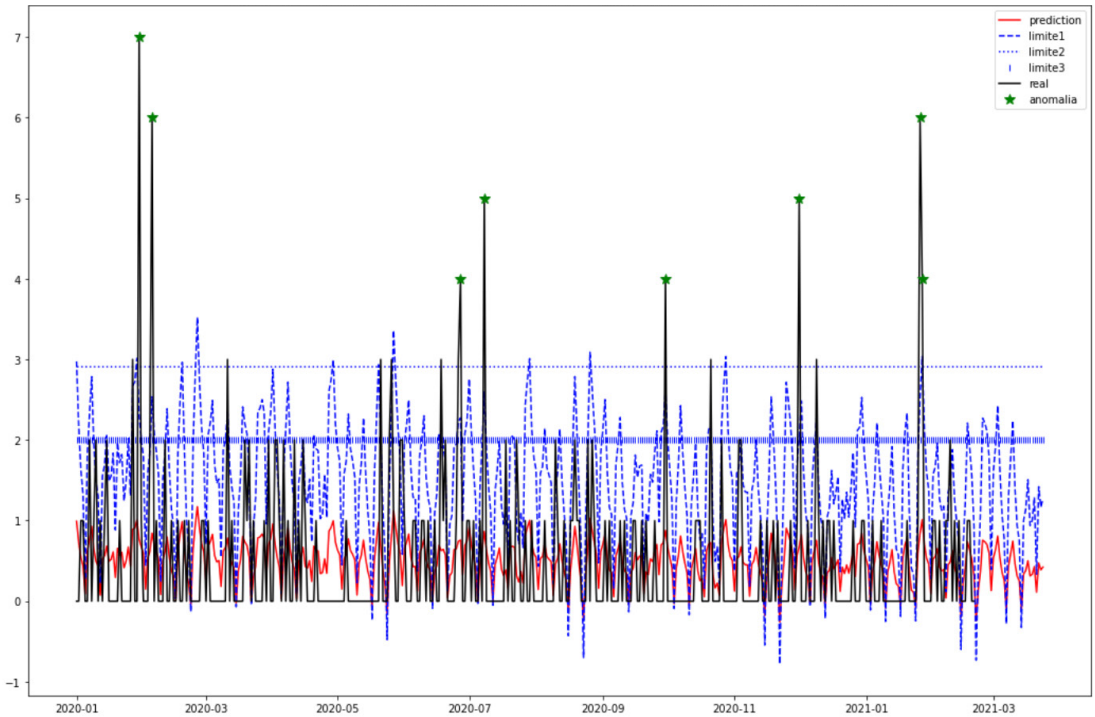


Figura 32: del cálculo del umbral para el clúster 1

3.4.1.2 Cluster 2

Las series temporales del clúster 2 son las st más estables y en la que mejor funciona el cálculo del umbral mediante la predicción del modelo, el único fallo son los fines des de semana, hay que crear una regla que regule estos días.

Como se pude ver en la *Figura 33* los "valles" de la serie temporal representan los fines de semanas y se puede apreciar como se detectan demasiadas anomalías que no reflejan un problema de negocio. El modelo predice muy por debajo del valor real y esto hace que salten muchas anomalías al calcular el umbral como tres veces la predicción.

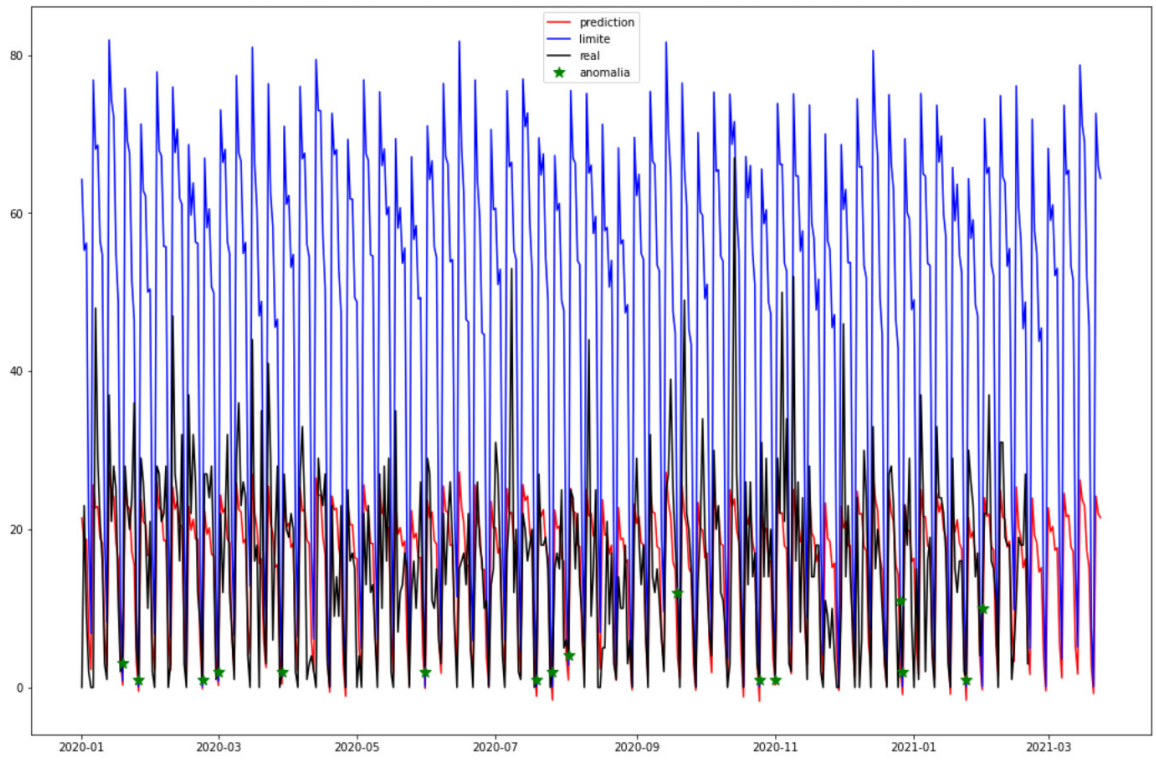


Figura 33: ejemplo detección errónea de anomalías los fin de semanas

Para que se aprecie con más claridad, se hace zoom de la *Figura 33*.

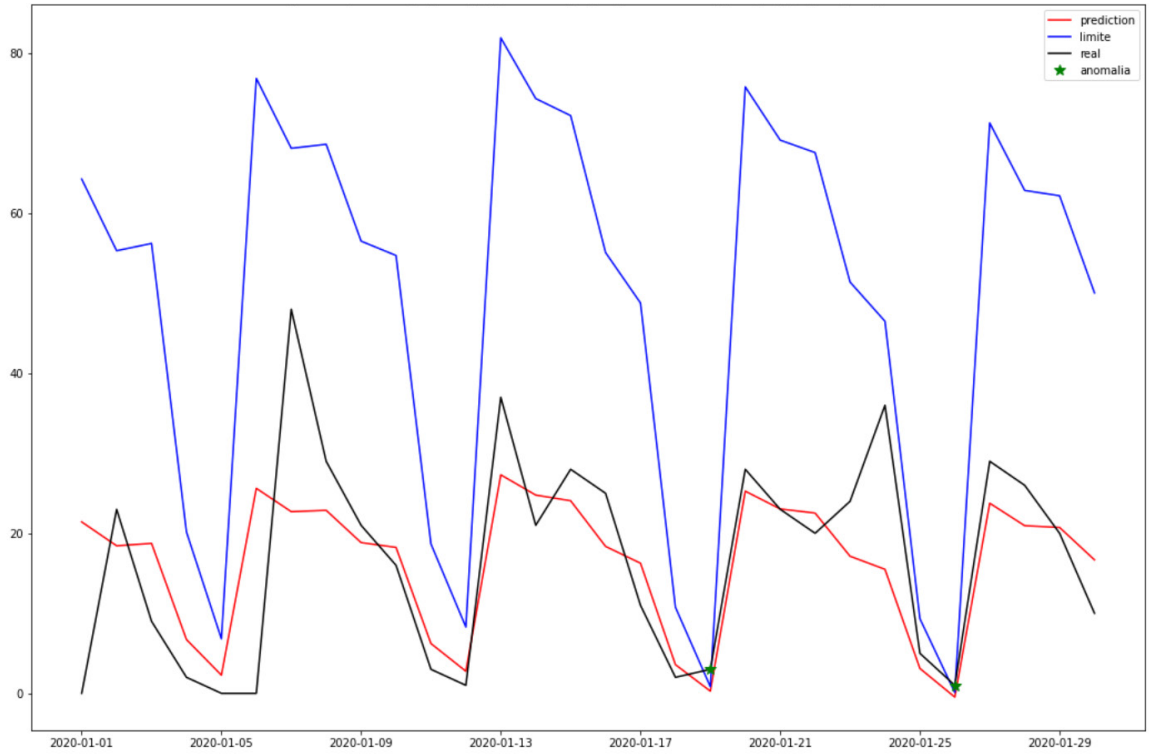


Figura 34: zoom de la Figura 33

Para resolver esto, se ha definido dos límites diferentes y se elegirá el límite más alto.

- **3*predicción**: el límite será 3 veces la predicción del modelo.
- **Media + 2.5 des**: se considera que todas las cantidades que se encuentren en el rango de [0, media+2.5*des] es un pedido normal.

Para que se vea el cambio, se representa la misma st de la *Figura 33* pero introduciendo esta nueva lógica. Como se puede ver han desaparecido todas las anomalías.

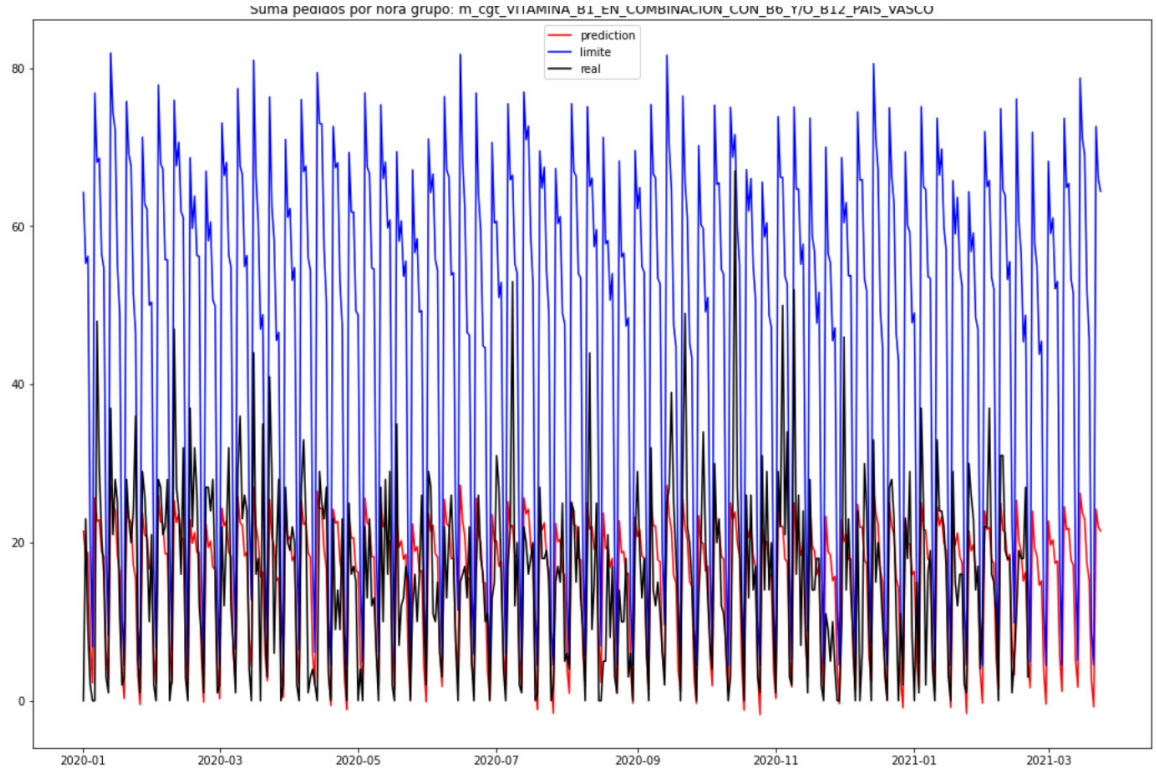


Figura 35: errores corregidos de la Figura 33

Capítulo 4. ANÁLISIS DE RESULTADOS

En este capítulo por un lado se va a poner un ejemplo de como hubiera funcionado un día del año y por otro se va a enseñar la herramienta diseñada para que el departamento de auditoría interna analice las anomalías.

4.1 ANÁLISIS DE ANOMALÍAS DEL MARTES 15/2

Se han detectado un total de 76 anomalías el día 15/2. Este número es un poco elevado para que se pueda hacer un seguimiento de todas ella. Una posible solución sería aumentar el umbra y sustituir, por ejemplo, las ecuaciones *3*predicción* por *3.5*predicción* o *4*predicción*. También se podría aumentar el rango de normalidad de *[0, media+2.5*des]* a *[0, media+3*des]*.

Algunas de las anomalías que se han detectado que sí son anomalías de negocio son la Figura 36 y Figura 37. Pero por otro lado están las Figura 38 y Figura 39, estas anomalías no están muy claro si son anomalías de negocio, ¿bajo que circunstancias sí serían anomalías de negocio? Una posible respuesta sería cuando provocaran una escasez de stock. Para ello, una mejora sería enriquecer nuestros datos con la información de los almacenes y crear una nueva variable que nos indique el porcentaje de stock que se ve reducido con cada pedido, de esta forma solo se detectaría como anomalía cuando "arrasara" el stock.

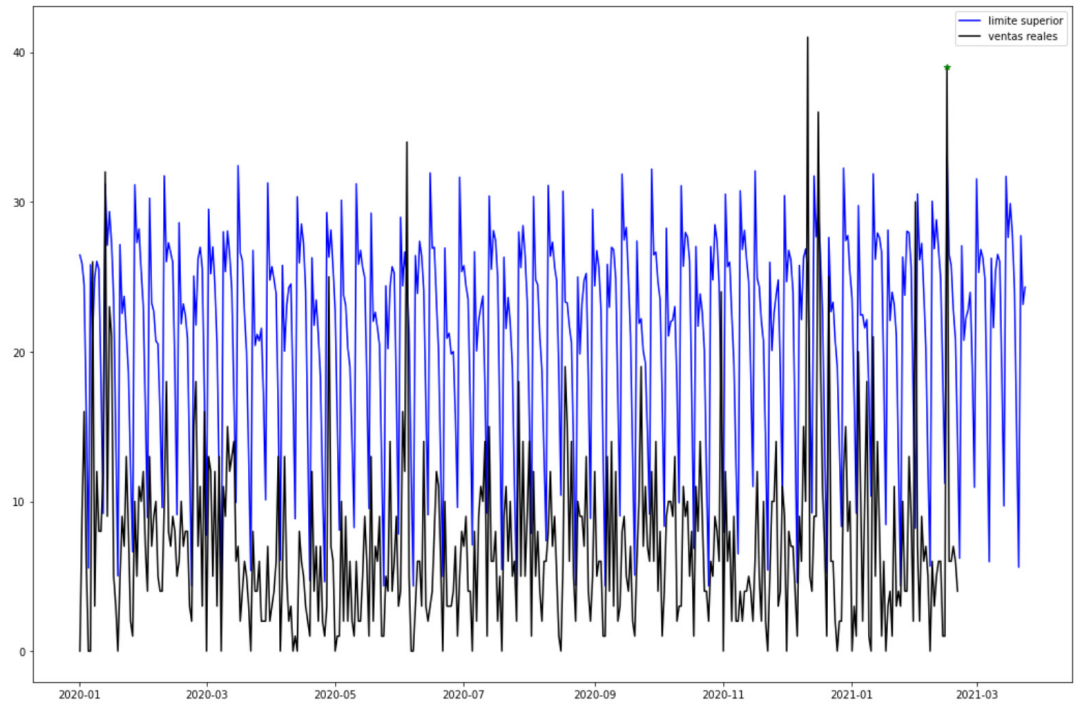


Figura 36: anomalía de negocio detectada

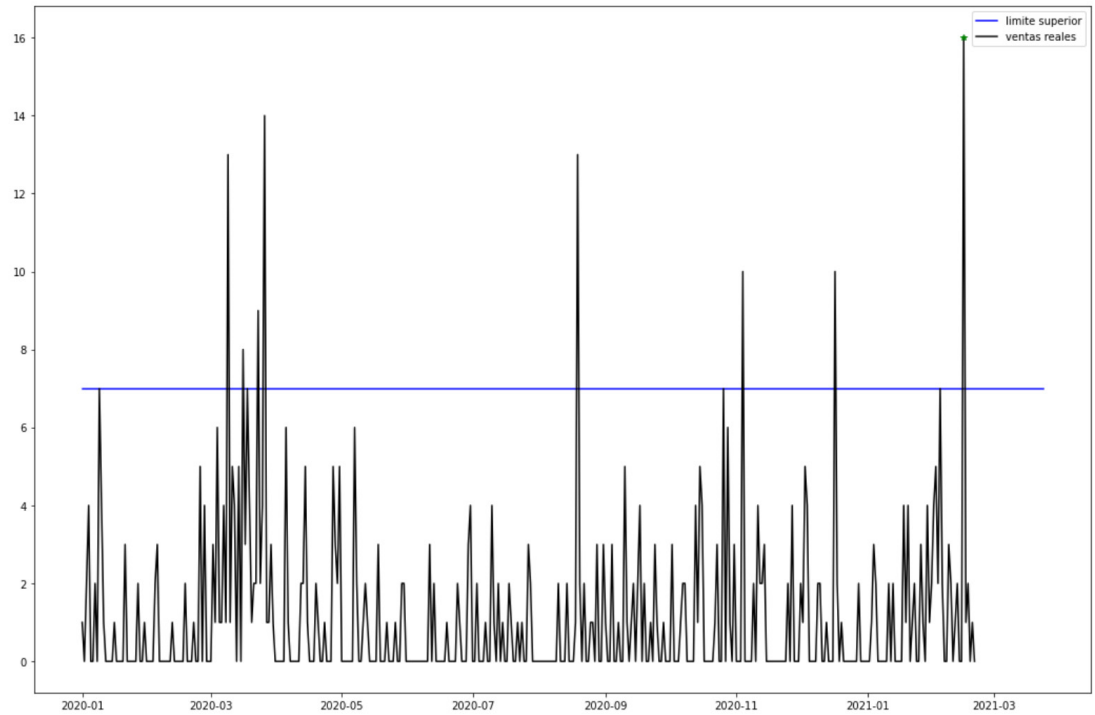


Figura 37: anomalía de negocio detectada

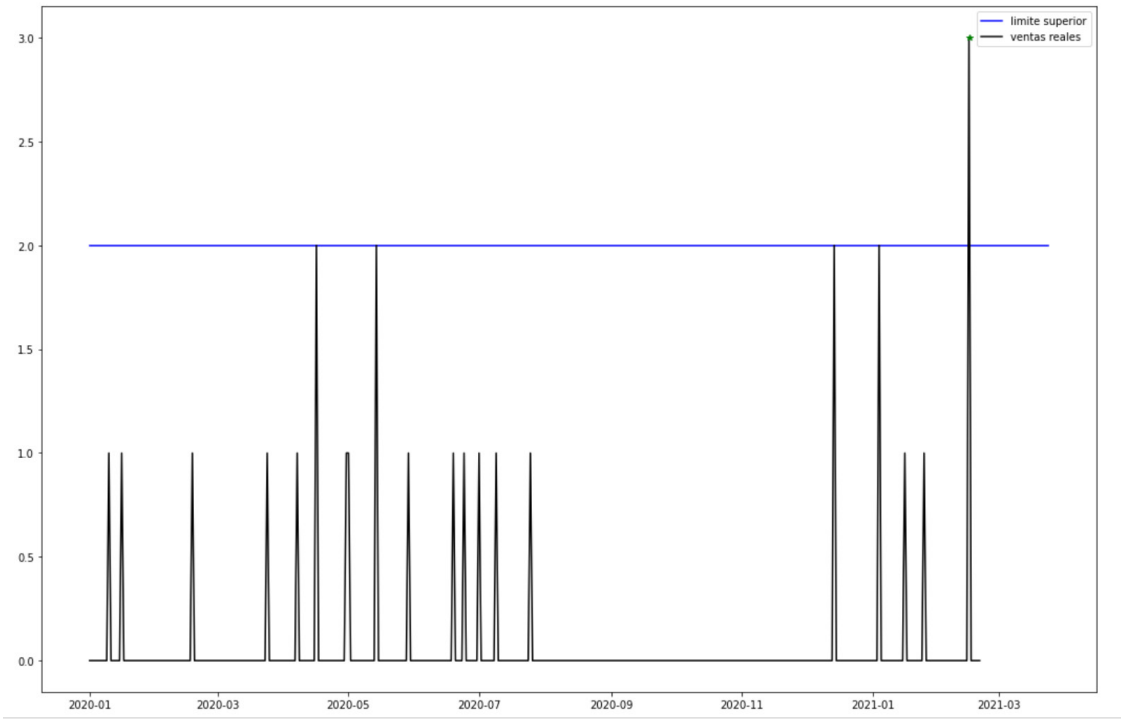


Figura 38: ¿anomalía de negocio?

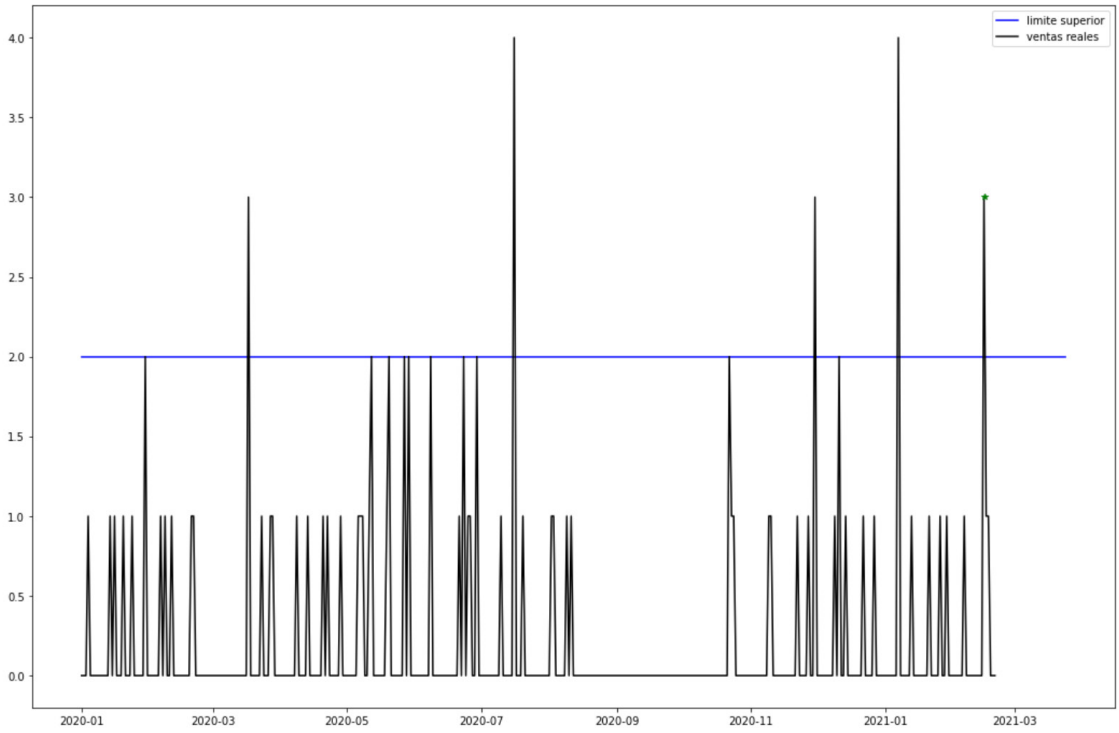


Figura 39: ¿anomalía de negocio?

4.2 HERRAMIENTA DISEÑADA

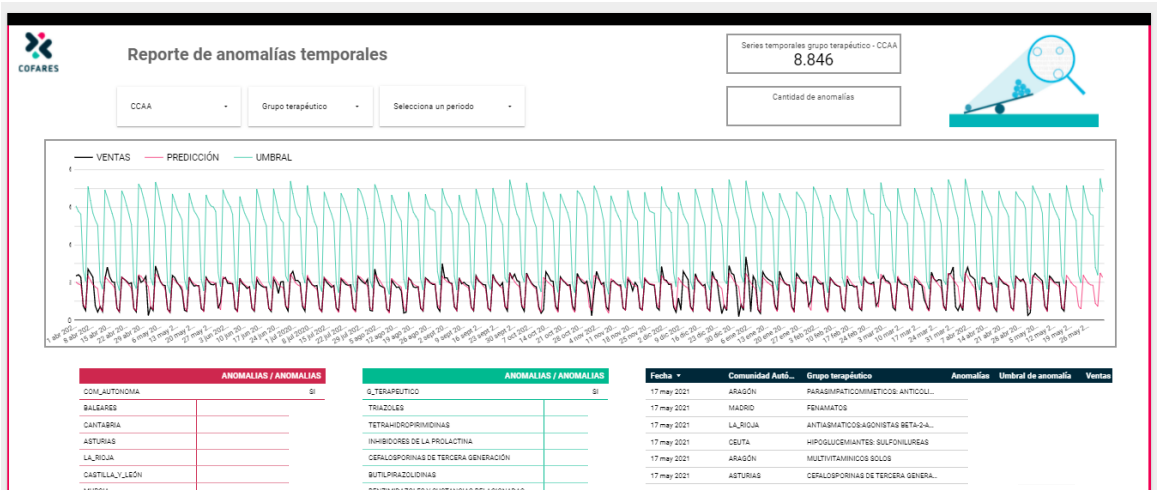


Figura 40: herramienta diseñada

La herramienta diseñada se ha pensado de tal forma que sea lo más visible posible y con la que se pueda interactuar para adaptarse a las necesidades de los auditores.

Empezando por arriba a la izquierda se tiene 3 filtros:

- **CCAA**: para especificar que Comunidad Autónoma se quiere analizar.
- **Grupo Terapéutico**: para especificar que GT se quiere analizar.
- **Selecciona un periodo**: concreta un periodo de tiempo concreto.

Continuando arriba a la derecha se tienen dos contadores. Uno que indica el número total de series temporales y otro que cuenta el número total de anomalías encontradas en todas las st. Estos contadores cambian con respecto a los filtros que se han comentado antes.

Un poco más abajo se tiene en horizontal una gráfica del tiempo. En esta gráfica se tienen 3 series temporales:

- **VENTAS**: en color negro se representan las ventas reales.
- **PREDICCIÓN**: en color rosa se representa las ventas que predice el modelo.

- **UMBRAL**: en color verde se representa el límite máximo que se ha calculado para especificar la región de normalidad.

Hay que tener en cuenta que en esta gráfica si no se especifica una CCAA y un GT concreto lo que se va a ver es la suma de muchas series temporales y por tanto va a perder sentido analítico. Para poder hacer un análisis concreto hay que utilizar los filtros que encontramos arriba.

Por último, se tiene las tres tablas en la parte inferior de la herramienta. Estas tablas están pensadas para poder guiar al auditor en su búsqueda de anomalías. La idea es que el auditor lo primero que haga sea mirar estas tablas para identificar que anomalías han aparecido, después vaya a los filtros de la parte superior y por último que vea la serie temporal y concluya si merece la pena realizar un seguimiento. Estas tablas suman el número de anomalías por CCAA y por GT (tablas rosa y verde respectivamente) y se tiene una tercera tabla que da la información mas específica de las anomalías como la fecha en la que sucede, la CCAA a la que pertenece, su GT, el umbral que no debería pasar y las ventas reales.

Capítulo 5. CONCLUSIONES Y TRABAJOS FUTUROS

En este proyecto se ha conseguido desarrollar una herramienta interactiva para que el departamento de auditoría interna pueda identificar, investigar y seguir de cerca los pedidos anómalos. Esta herramienta representa gráficamente el modelo de clasificación de anomalías de forma sencilla y directa mejorando la anterior herramienta.

Gracias a la adecuada selección de variables, la herramienta proporciona a los auditores información reciente. De esta forma se ha conseguido reducir el tiempo de detección de anomalías disminuyendo el periodo de actuación.

Esta herramienta se basa en un modelo de clasificación de anomalías que se divide en tres partes: clasificación de las series temporales, predicción de ventas y cálculo de la región de normalidad.

La clasificación de series temporales se ha realizado con éxito ya que divide los tres grupos deseados utilizando las variables *número de días pedidos* y *media de la st.*

El modelo de predicción, por un lado, tarda aproximadamente 5h cumpliendo la condición impuesta en el proyecto de una duración menor a 12h. Pero, por otro lado, la precisión del modelo se ha visto limitada por el gran número de series temporales poco estables. Esta limitación se ha tenido que compensar en la siguiente etapa mediante diferentes lógicas para calcular la región de normalidad en función de los puntos débiles de cada clúster.

Al final, la región de normalidad calculada ha conseguido detectar un pequeño número de anomalías diarias, aunque un poco más alto de lo esperado. No sobrecarga exageradamente de información a los auditores, pero siguen sin dar tiempo a revisar todas ellas.

Para trabajos futuro, habría que pensar formas de reducir el número de anomalía detectadas y confirmar que sean anomalías de negocio. Para reducir el número de anomalías detectadas se podría endurecer los cálculos del umbral pensando nuevas ecuaciones o lógicas. También

estaría bien como mejora aprovechar el feedback de los auditores e ir clasificando que anomalías de las detectadas en verdad no son anomalías de negocio y por tanto carecen de interés, de esta forma, se reduciría también el número diario sin endurecer mucho las condiciones.

Otro punto para mejorar sería cambiar la agrupación de las series temporales. Ahora mismo los auditores pueden concretar la anomalía a una región determinada y un grupo terapéutico, pero no saben exactamente que cliente ha provocado las anomalías. Para dar más información útil a los auditores estaría bien probar a agrupar por grupo terapéutico y almacén, así es sencillo introducir la variable del stock que ayudaría a determinar con facilidad que anomalías tiene interés de negocio.

Capítulo 6. BIBLIOGRAFÍA

[1] "Cofares." https://www.cofares.es/nuestra-cooperativa.

[2] Rincón Quintero, Brayan Stiven. Aplicación de machine learning en el mantenimiento predictivo industrial con herramientas de código abierto, 2020. http://expeditiorepositorio.utadeo.edu.co/bitstream/handle/20.500.12010/10108/Trabajo%20de%20grado.pdf?sequence=1&isAllowed=y

[3] Leyanis López-Avila,Niusvel Acosta-Mendoza,Andrés gago-Alonso. Detección de anomalías basada en aprendizaje profundo: Revisión. http://scielo.sld.cu/scielo.php?pid=S2227-18992019000300107&script=sci_arttext&tlng=en

[4] Autoarima. http://alkaline-ml.com/pmdarima/index.html

[5] Prophet. https://facebook.github.io/prophet/