

**Pronóstico de Venta para una Empresa Farmacéutica:
Análisis Comparativo Entre Modelos Tradicionales y Algoritmos de Aprendizaje
Automático**

CARLOS HERNAN CARDOSO MEDINA

Universidad del Valle
Facultad de Ingeniería
Programa Académico de Estadística
Santiago de Cali
2022

**Pronóstico de Venta para una Empresa Farmacéutica:
Análisis Comparativo Entre Modelos Tradicionales y Algoritmos de Aprendizaje
Automático**

CARLOS HERNAN CARDOSO MEDINA

Trabajo de grado presentado como requisito para optar al título de:
Estadístico

Director

Ms. Jhon Pablo Guzman

Co-director

Ph.D. Jaime Mosquera Restrepo

Universidad del Valle

Facultad de Ingeniería

Programa Académico de Estadística

Santiago de Cali

2022

Agradecimientos

Gracias a Dios por permitir la alineación del universo para poder culminar este proceso.

A BigMartin, madre, padre, Maria T., Martha, Stepha, Cesitar, Alejo y Maria por ser inspiración. Su acompañamiento fue soporte fundamental.

A Juan David, Camilo, Yafa, Nilsa, Alicia, Toñita, y Monica por su continuo aporte, por ser, amor y aliento en los momentos de flaqueza.

A mis amigos quienes con su apoyo moral e incondicional fueron siempre voz de ánimo. Julian, Hector, Maria Isabel y Cornelio.

A mis directores de trabajo de grado, Pablo y Jaime, por su disposición a colaborar y poner su conocimiento y experiencia en pro de cerrar este importante proceso académico y de vida.

A mis profesores de Univalle, mil gracias por compartir sus conocimientos y brindar los instrumentos para ser un profesional.

Y a todas aquellas personas que hicieron parte de este proceso, aportando su buena energía y mucho más para cerrar este proceso.

Declaración

Me permito afirmar que he realizado el presente trabajo de grado de manera autónoma y con la única ayuda de los medios permitidos y no diferentes a los mencionados en la propia tesis. Todos los pasajes que se han tomado de manera textual o figurativa de textos publicados los he reconocido en el presente trabajo. Ninguna parte del presente trabajo se ha empleado en ningún otro tipo de tesis. Santiago de Cali, Valle del Cauca, Colombia, 15 de agosto de 2022.

Igualmente declaro que los datos utilizados en este trabajo están protegidos por las correspondientes cláusulas de confidencialidad.

Carlos Hernan Cardoso Medina

Resumen

Las empresas se enfrentan a un entorno volátil, incierto, complejo y ambiguo que dificulta la toma de decisiones para la consecución del logro de corto plazo y la adaptación del largo plazo. Para competir en este entorno el pronóstico de demanda es fundamental para la adecuada planeación de capacidades operacionales, logísticas y financieras para responder al mercado. Se reconocen diversos métodos de pronósticos como el criterio de experto, promedio móvil, métodos estocásticos, y actualmente emergen con fuerza los algoritmos de aprendizaje automático. Este trabajo, desarrolla un modelo de pronóstico para dos productos del portafolio comercial de una compañía farmacéutica a partir de un análisis comparativo de desempeño entre modelos ARIMA y algoritmos de aprendizaje automático teniendo en cuenta la serie histórica de ventas. Los resultados muestran que los modelos ARIMA y de aprendizaje tienen un mejor desempeño que la metodología actual de promedio móviles y que el modelo Prophet, muestra para este caso, mejores métricas de desempeño.

Palabras clave: pronóstico de venta, ARIMA, aprendizaje automático, Prophet y XGBoost.

Tabla de Contenido

1	Introducción	1
2	Formulación del problema	2
2.1	Definición Del Problema	3
2.2	Justificación	4
2.3	Objetivos	5
2.3.1	<i>Objetivo general.....</i>	<i>5</i>
2.3.2	<i>Objetivos específicos.....</i>	<i>5</i>
3	Antecedentes.....	6
3.1	Las competencias makridakis.	7
3.2	El algoritmo PROPHET.....	8
3.3	Pronósticos en la industria farmacéutica.....	9
4	Marco teórico	11
4.1	Marco conceptual.....	11
4.1.1	<i>¿Qué es un buen pronóstico?.....</i>	<i>12</i>
4.2	Técnicas cuantitativas para pronósticos.....	15
4.2.1	<i>Promedio móvil.....</i>	<i>15</i>
4.2.2	<i>Modelo autorregresivo integrado de media móvil (ARIMA).....</i>	<i>16</i>
4.3	Algoritmos de aprendizaje automático: de los árboles de decisión al XGBoost	24
4.3.1	<i>De los árboles de decisión al XGBoost.....</i>	<i>25</i>
4.3.2	<i>Algoritmo Phophet.....</i>	<i>33</i>
5	Metodología	36
6	Resultados.....	40
6.1	Fase 1 - Comprensión del negocio o problema.....	40
6.2	Fase 2 - Comprensión de los datos	41
6.3	Fase 3 – Preparación de los datos	43
6.4	Fase 4 – Modelado	48
6.5	Fase 5 – Evaluación	54
7	Conclusiones	58
8	Bibliografía	59
	Apéndice A. Aspectos de calidad de un pronóstico.....	65
	Apéndice B. Modelos de desarrollo aditivo	66
	Apéndice C. Diagrama de proceso y SIPOC	68
	Apéndice D. Líneas de comando R.....	69

Lista de tablas

Tabla 1 <i>Algoritmo random forest</i>	28
Tabla 2 <i>Implementación del Gradient Boosting en regresión</i>	31
Tabla 3 <i>Secuencia para la modelación</i>	38
Tabla 4 <i>Resumen de resultados de la estimación</i>	49
Tabla 5 <i>Supuestos sobre el error</i>	50
Tabla 6 <i>Creación de variables Componente115y Componente141</i>	53
Tabla 7 <i>Componente115 – Medidas de desempeño</i>	55
Tabla 8 <i>Componente141 – Medidas de desempeño</i>	56

Lista de Figuras

Figura 1 <i>Ejemplo de las gráficas FAC y FACP de un modelo MA(1)</i>	18
Figura 2 <i>Evolución de los árboles de decisión</i>	25
Figura 3 <i>Fases de la metodología CRISP-DM</i>	36
Figura 4 <i>Esquema del SIPOC</i>	37
Figura 5 <i>Ventas históricas 2010-2020</i>	42
Figura 6 <i>Descomposición de la serie histórica C115</i>	44
Figura 7 <i>Análisis de la estacionalidad de la serie histórica C115</i>	44
Figura 8 <i>Serie imputada en julio de 2019</i>	45
Figura 9 <i>Análisis de los rezagos de la serie históricas C115</i>	45
Figura 10 <i>Descomposición de la serie histórica C141</i>	46
Figura 11 <i>Análisis de la estacionalidad de la serie histórica C141</i>	47
Figura 12 <i>Análisis de los rezagos de la serie histórica C141</i>	47
Figura 13 <i>Serie residuos Vs. el tiempo</i>	50
Figura 14 <i>Selección del mejor modelo Prophet</i>	53
Figura 15 <i>Modelos de pronóstico XGBoost</i>	54
Figura 16 <i>Pronóstico Componente115</i>	55
Figura 17 <i>Pronóstico Componente141</i>	57
Figura 18 <i>Flujo del proceso actual de pronóstico de demanda</i>	68

1 Introducción

Tener una planeación que facilite las decisiones óptimas empresariales requiere de una robusta gestión de demanda soportada por la generación de pronósticos confiables y oportunos. Las desviaciones a nivel de pronóstico podrían implicar ventas perdidas, afectación del nivel de servicio, sobrecostos logísticos, obsoletos, problemas de inventario, faltantes/excesos de materias primas y un complejo ambiente laboral.

La empresa farmacéutica para la cual se desarrolla este trabajo se enfrenta, para el logro de sus objetivos, a un entorno complejo y dinámico dada su presencia en 14 países con más de 600 referencias a nivel de producción y comercialización a través de las distintas unidades de negocio. Por ello, se ha propuesto construir un proceso de planeación sincronizado que logre el óptimo balance entre oferta productiva y la demanda; soportado (entre otros) en la mejora de precisión del pronóstico actual (MAPE del 75%) que ha sido identificado como causa-raíz del bajo nivel de servicio, excesos de inventarios y ventas perdidas entre otros.

Este trabajo de investigación busca proponer un modelo de pronóstico de ventas para dos productos CORE según desempeño entre el método actual de pronóstico, ARIMA y algoritmos de aprendizaje automático. Para ello se presentan formalmente la definición, la justificación y los objetivos del problema (Cap. 1), la evolución de los métodos de pronóstico y estudios comparativos entre métodos tradicionales y los algoritmos de aprendizaje automático (Cap. 2), los fundamentos conceptuales y teóricos (Cap. 3), la metodología seleccionada (Cap. 4), los resultados obtenidos (Cap. 5) y conclusiones, bibliografía y anexos (Cap. 6 al 9).

La información generada en el presente estudio será de gran utilidad para la implementación del proceso de demanda colaborativa que soportará el proceso de planeación integrada del negocio por parte de la compañía.

2 Formulación del problema

La gerencia de organizaciones en entornos VICA (volátiles, inciertos, complejos y, ambiguos) ha derivado en cambios de gestión y liderazgo de negocios (Blackburn R, 2014). Es decir, cambio que ocurren continuamente los cuales son difíciles de predecir dado los múltiples factores que intervienen, dificultando con esto direccionar acciones para gestionarlos. En particular estos entornos dificultan la gestión eficiente de la cadena de abastecimiento (Christopher & Holweg, 2011). Motivo por el cual, se requieren de variedad de herramientas que permitan conocer el mercado y al consumidor, para mantener un mejor nivel de servicio sin arriesgar rentabilidad e inversión de capital (Van der Vorst & Beulens, 2002).

Entre las herramientas adoptadas se encuentra el pronóstico ventas dentro de la gestión de demanda por su importancia para soportar la toma decisiones racionales (Apichatibutarapong, 2015). Además, no solo clave para el cumplimiento de objetivos organizacionales sino para la competitividad; dado que no tomar decisiones correctas, puede ir en detrimento de la organización (Vidal Holguin, 2010). Actualmente, hay diversos métodos de pronósticos, pero su utilización depende del contexto del negocio, mercado y los productos/servicios.

La Industria Farmacéutica no es la excepción al rol crítico de la predicción de ventas y su impacto en el cumplimiento de objetivos (Anshuman, Costas, & Conor, 2000), dada la intervención de múltiples factores en su cadena de suministro globalizada. Se torna fundamental contar con un pronóstico confiable para lograr el balance demanda – capacidad y lograr el propósito de garantizar abastecimiento a productos relacionados con la salud humana (Doganis, Alexandridis, Patrinos, & Sarimveis, 2006).

La empresa farmacéutica para la que se desarrolla este trabajo está implementando el IBP (Integrated Business Planning) como método de planeación y garantizar la alineación comercial, operacional y, financiera con visual de 12 meses. Para ello es importante implementar mejores

prácticas de gestión de demanda a nivel de pronóstico de ventas, dado que su precisión global actual con un MAPE del 75%. está afectando los resultados del IBP.

El proceso de pronóstico actual no se enmarca en un proceso colaborativo y no tiene definida una metodología estadística rigurosa que genere el primer resultado para el posterior consenso. Actualmente, cada mes se genera un pronóstico de ventas para cada producto con visual de 12 meses soportados en promedios móviles y criterios expertos, esto implica que solo se está capturando la tendencia (primer momento condicional de la serie de tiempo) y se pierde oportunidad al no capturar el efecto de estacionalidades. Se trabaja con las ventas, como proxy de la demanda, dado que el módulo de facturación del sistema de planificación de recursos (ERP) no cuenta con información de las ventas perdidas por disponibilidad de producto entre otras causas.

El área de planeación de demanda, mejora continua y tecnología diseñaron un proyecto para cerrar la brecha existente en el proceso de pronóstico y establecieron como piloto mínimo viable (PMV o MVP por sus siglas en inglés) identificar un modelo de pronóstico estadístico soportado en el análisis comparativo entre modelos de promedios móviles, ARIMA y algoritmos de aprendizaje automático para dos referencias del portafolio. Posteriormente se desarrollará una API de integración para lograr consenso por parte del equipo de tecnología de la empresa y no es parte del alcance de este trabajo. Teniendo en cuenta esta problemática, la pregunta de investigación por abordar se describe a continuación.

2.1 Definición Del Problema

¿Permite el análisis histórico mensual de las ventas de los productos componente115 y componente141, identificar un modelo de proyección de ventas mensuales a 12 meses que mejore el desempeño de los pronósticos actuales de la compañía soportándose en un análisis comparativo entre modelos ARIMA y los algoritmos de aprendizaje automático?

2.2 Justificación

La compañía farmacéutica para la que se realiza este trabajo estableció como meta para el 2022 incrementar el nivel de servicio del 83% al 100% en los productos estratégicos, aquellos que concentran el 60% de los ingresos. Para ello debe aumentar la precisión del pronóstico comercial e institucional en dichos productos del 40% al 70%.

El actual proceso de pronóstico, frente al modelo teórico de Gestión de Pronóstico de Clase Mundial¹, no cuenta con un modelo estadístico de pronóstico para el consenso de demanda. Un benchmarking realizado, concluyó que la organización no está madura a nivel de proceso para realizar inversión inicial y que los softwares comerciales desarrollan pronósticos a partir de suavización exponencial, metodología ARIMA y máquinas de aprendizaje automático.

Ante este escenario se planteó un proyecto para desarrollar internamente un piloto de pronóstico de ventas a 12 meses para los productos **componente115** y **componente141**. Estos productos son referente comercial de la marca, cuentan con presencia en todas las geografías en las líneas comercial e institucional, hacen parte del Pareto de generación de ingresos y su MAPE histórico promedio es del 70%.

Identificar, a partir del análisis histórico mensual del comportamiento de ventas de los productos **componente115** y **componente141**, un modelo de proyección de ventas mensual a 12 meses que mejore el desempeño de los pronósticos actuales de la compañía soportándose en un análisis comparativo entre ARIMA y modelos de aprendizaje automático se convierte en una victoria temprana para apalancar el cumplimiento de la meta crucialmente importante y lograr eficiencias operacionales, financiero y de entorno laboral.

¹ El cuestionario mide cuatro dimensiones: integración funcional, enfoque, sistemas e indicadores de desempeño de los procesos de pronóstico de las empresas. Cada dimensión la conforman preguntas tipo Liker con valores que van desde 1 hasta 5, donde 1 indica “Nunca se cumple” y 5 “Siempre se cumple”.

2.3 Objetivos

2.3.1 *Objetivo general*

Desarrollar un modelo de proyección de ventas mensual a 12 meses para el **componente115** y el **componente141** a partir de las series históricas mensual de ventas que mejore el desempeño de los pronósticos actuales de la compañía soportándose en un análisis comparativo entre modelos ARIMA y los algoritmos XGBoost y PROPHET de aprendizaje automático.

2.3.2 *Objetivos específicos*

- Identificar los modelos de pronóstico de ventas alineado a la necesidad de la empresa tanto en método de series de tiempo como en aprendizaje automático.
- Ajustar para cada uno de los productos modelos de pronóstico de venta, con el método de series de tiempo y los algoritmos de aprendizaje automático.
- Seleccionar el modelo idóneo para cada producto acorde a los análisis de desempeño y capacidad de pronóstico.

3 Antecedentes

Se han desarrollado acercamientos cualitativos o cuantitativos para realizar pronósticos. Desde lo cuantitativo, a partir de 1970, las metodologías que se han aplicado en los pronósticos de ventas normalmente han sido métodos de series de tiempo lineales o no lineales (Doganis, Alexandridis, Patrino, & Sarimveis, 2006), a través de métodos como promedios móviles, suavizaciones exponenciales, análisis de tendencias, modelos ARIMA y modelo de Winter entre otros (Mercuryeva, Valverga y Smirnov 2019).

Es una buena práctica la proyección de series tiempo soportada por modelos ARIMA o SARIMA (Adhikari & Agrawal, 2013). Ejemplo de esto son los buenos resultados logrados para el pronóstico de precios de criptomonedas en función del impacto de las redes sociales (Tandon, Revankar, Parihar, & Singh, 2021) o los logrados por un modelo SARIMA en el pronóstico de la demanda turística (Goh & Law, 2002) y la predicción del flujo de tráfico vehicular (Williams & Hoel, 2003). Los ARIMA y SARIMA tiene su capacidad de predicción restringida por los supuestos de linealidad en sus parámetros (Doganis, Alexandridis, Patrino, & Sarimveis, 2006) y su incapacidad para detectar patrones altamente volátiles (Hamzaçebi, 2008). Como derivación, Zhang (2003) no los considera valido para todos los casos, especialmente para series temporales no lineales.

Desde la década de 1980, avances computacionales e infraestructura de datos han dado lugar a algoritmos complejos y la evolución de las técnicas de aprendizaje automático (Diez y Fernandez, 2005). De hecho, a través de las redes neuronales artificiales (RNA o ANN por sus siglas en inglés) se ha venido cubriendo la eficiencia ante no linealidad, y han tomado un papel protagónico en el dominio de pronóstico (Doganis, Alexandridis, Patrino, & Sarimveis, 2006). Kuo y Xue (1998) demostraron que sus modelos de inteligencia artificial desarrollados podrían encontrar relaciones no lineales mejores que las series de tiempo convencionales y Adhikari y Agrawal (2013) realizaron análisis de series estacionales a través de las redes neuronales

artificiales estacionales (SANN), redes neuronales retardadas en el tiempo (TLNN) y, máquinas vectoriales (SVM) presentando resultados satisfactorios.

Wang, Zoy, Su, Li y Chaudhry (2013) subrayan que los estudios comparativos entre modelos tradicionales y ANN no son concluyentes respecto a la eficiencia de un método sobre el otro. Es decir, se debe conocer el problema dado que no existe una técnica mejor para realizar pronósticos para series de tiempo (Zhang & Kline, 2007). Ejemplo de esto es el estudio de Alón, Qi y Sadiwski (2001) donde la predicción ANN superó el suavizado exponencial de Winters y las técnicas de Box Jenkins para la predicción de ventas en el sector retail. Por su parte Doganis, Alexandridis, Patrinos y Sarimveis (2006) encontraron resultados a nivel de eficiencia equivalentes entre las ANN y ARIMA para predecir las ventas de productos alimenticios perecederos. Para series de memoria larga tanto las ANN como la metodología ARFIMA tienen el mismo desempeño (Tang, Almeida, & Fishwick, 1991). A nivel multivariado, las ANN mostraron mejor desempeño que las series multivariadas para predecir los precios de la harina en tres ciudades de USA; (Chakraborty, Mehorotra, Mohan, & Ranka, 1990)

Makridakis y Hibon (1979) concluyeron que los métodos simples funcionan bien en comparación con los más complejos y estadísticamente sofisticados. Así se crearon las competencias Makridakis (Competencias M) con el objetivo de evaluar y comparar empíricamente la precisión de diferentes métodos de pronóstico (Hyndman R., 2020).

3.1 Las competencias makridakis.

La primera Competencia Makridakis (M), celebrada en 1982, utilizó 1001 series de tiempo y 15 métodos de predicción. Se encontró que métodos estadísticamente sofisticados o complejos no proporcionan necesariamente pronósticos más precisos que los más simples, la eficiencia de los diversos métodos varía según la medida de precisión que se utilice y la precisión depende de la longitud del horizonte de pronóstico (Makridakis, y otros, 1982).

Los resultados de la segunda competencia (M2) no cambiaron respecto a M aunque se permitió combinar métodos de pronóstico con juicio personal y hacer preguntas adicionales sobre

las series de datos (Makridakis, y otros, 1993). Los hallazgos de M3 no cambiaron respecto a M y M2, se incluyeron investigadores en el área de redes neurales y series de tiempo y se utilizó error porcentual absoluto medio simétrico (SMAPE), clasificación promedio, error porcentual absoluto simétrico medio para evaluar la precisión (Makridakis & Hibon, 2000).

La competencia M4 en 2018 reunió un total de 100.000 series de tiempo con diferente temporalidad e industria, incluyendo el requerimiento de intervalos de predicción (PI) (Makridakis, Spiliotis, & Assimakopoulos, 2020). Se observó que la combinación de métodos mejora la precisión del pronóstico, de los 17 métodos más precisos, 12 eran combinaciones con enfoque principalmente estadístico, los modelos híbridos (modelos estadísticos y aprendizaje automático) produjeron pronósticos e intervalos más precisos (+10% en SMAPE), el segundo método más preciso combinó siete métodos estadísticos y uno aprendizaje automático y los seis métodos puros de aprendizaje automático presentados en el M4 presentaron bajo desempeño.

M5 en 2020, utilizó datos de Walmart de 100,000 series de tiempo diarias jerárquicas, desde SKU hasta la demanda total en área geográfica, información de precios, actividad publicitaria/promocional, niveles de inventario y el día de la semana al que se refieren los datos. En M5 se evidenció superioridad de los algoritmos de aprendizaje automático, específicamente LightGBM y XGBoost. Además, aparecen los algoritmos DeepAR y N-BEATS de aprendizaje profundo, el algoritmo ganador es una combinación de modelos, los algoritmos en el Top 50 utilizaron aprendizaje cruzado. (Makridakis, Spiliotis, & Assimakopoulos, 2020)

3.2 El algoritmo PROPHET.

En el orden de identificar tendencia, estacionalidad e impacto de días festivos, Facebook creó el Prophet; un algoritmo interpretable en el que se modela la tendencia y sus puntos de cambio, la estacionalidad (anual, semanal y diaria) y los efectos de las vacaciones para lograr pronósticos (Papacharalampous & Tyralis, 2020). Prophet sobresale en el procesamiento de datos de periodicidad diaria con valores atípicos, cambios estructurales y múltiples períodos de estacionalidad (Zhao, Liu, Vanos, & Cao, 2018).

Algunos casos de éxito de este algoritmo son los de Erinjeri et al. (2020) para reducir horas no planificadas en la programación de procedimientos de pacientes hospitalizados. Park et al. (2019) para el pronóstico de visitas diarias al departamento de emergencias y Zhao et al. (2018) estudió patrones de día de la semana y estacionales de concentraciones de PM 2.5 sobre los Estados Unidos. A nivel comparativo, en el estudio sobre pronóstico de la temperatura mensual por Papacharalampous, Tyrallis y Koutsoyiannis (2018), los resultados del Prophet estuvieron cerca de los resultados de los métodos de suavizado exponencial y modelos ARIMA. Mientras que el Prophet logró mejores resultados, respecto a suavizado exponencial y modelos ARIMA, para predecir el tráfico de un sitio web (Subashini, Sandhiya, Saranya & Harsha, 2019) y pronosticar la enfermedad de COVID (Satrio, Darmawan, Nadia & Hanafiah, 2021).

3.3 Pronósticos en la industria farmacéutica

La industria farmacéutica busca soluciones que mejoren el desempeño de los pronósticos por fuera de las técnicas tradicionales. Carbonneau, Laframboise y Vahidov (2008) investigaron la aplicabilidad de técnicas avanzadas de aprendizaje automático, incluidas redes neuronales, redes neuronales recurrentes y máquinas de vectores de soporte, para pronosticar una demanda distorsionada al final de una cadena de suministro (efecto látigo) encontrando que las técnicas de aprendizaje automático no fueron significativamente mejor a nivel que la del modelo de regresión. Sin embargo, Candan et al. (2014) encontró mejor desempeño de las técnicas de análisis difuso (ANFIS) para pronosticar la demanda respecto a modelos tradicionales. Se consideró factores humanos y estacionales, enfermedades epidémicas, cuotas de mercado de competidores, productos y condiciones de comercialización que no suelen existir en los conjuntos de datos de la empresa, complejizando la réplica del ejercicio.

Un algoritmo híbrido presentó mejor desempeño que el ARIMA para pronosticar ventas de las compañías de distribución farmacéuticas. El modelo híbrido sería la combinación de un ARIMA para la parte lineal y una ANN a partir de los residuos (Zadeh, Sepehri & Farvaresh, 2014). Algunos estudios son realizados, donde el foco no era comparar contra otras técnicas, fue

el estudio de la predicción de demanda de la cadena de abastecimiento de unidades farmacéuticas utilizando ANN multicapa a partir de 519 datos históricos de 43 unidades de productos (Amalnick, Habibifar, Hamid y Bastan, 2019). Nowadly y Jung (2020) realizaron un modelo para pronóstico de demanda de largo plazo soportado en varias técnicas de máquinas de aprendizaje automático (random forest, ANN y máquina de soporte vectorial).

Según la literatura revisada, cada caso de pronóstico ser revisado y tratado acorde a su contexto. Los modelos ARIMA (SARIMA) y regresiones lineales han sido consideradas técnicas válidas para lograr pronósticos. Sin embargo, los supuestos de linealidad y autocorrelación son una restricción. No hay elementos para decir que existen diferencias significativas entre los pronósticos realizados por los modelos ARIMA (SARIMA) versus las ANN como si los hay respecto a algoritmos híbridos; estos últimos con mayor precisión. Así mismo M5 confirmó que los algoritmos de aprendizaje automático, según el desempeño del SMAPE, son métodos eficientes para generar pronósticos. Enfoques basados en árboles como LightGBM o XGBoost, ocupan los primeros lugares en M5 así como métodos que modelan tendencia y estacionalidad, como el algoritmo PROPHET.

A partir de la revisión de la literatura se procederá entonces a identificar un modelo de pronóstico de ventas de los productos componente115 y componente141 a partir de los análisis comparativos entre los modelos ARIMA (SARIMA y los algoritmos XGBoost y PROPHET.

4 Marco teórico

La primera parte de este marco teórico, el marco conceptual, proporciona definiciones sobre el proceso de planeación de demanda y el rol del pronóstico de ventas. La segunda parte está orientada a presentar conceptos y definiciones de los modelos de promedios móviles y ARIMA y la última parte hará lo propio con los algoritmos XGBoost y Prophet.

4.1 Marco conceptual

Lograr un modelo de pronóstico de demanda implica claridad conceptual e integral del requerimiento y dificultad dado los diversos factores que intervienen en esta tarea fundamental para la gestión de demanda de una organización (Lunenburg, 2012). Es importante definir y comprender el uso de los pronósticos, establecer cualidades de los inputs y outputs, caracterizar el tipo de información que se tiene, y cómo es su integración a la organización. Relevante precisar el nivel de granularidad, por ejemplo, se hace a nivel agregado o por cada referencia.

Entender el requerimiento implica comprensión del Macroproceso de gestión de demanda definido en tres fases. La fase inicial de crear un pronóstico de demanda preciso y válido para periodos futuros (pronóstico inicial), seguido por la fase de análisis de gap entre el pronóstico inicial y el plan comercial (análisis de brechas) para posteriormente, acordar un consenso entre los actores claves de gestión de demanda (revisión de demanda). Con el consenso, se procede balancear demanda y capacidades para abastecer el mercado.

Fundamental definir que entiende la organización por demanda. A menudo se logra el pronóstico inicial a partir de las ventas; entendidas como el hecho contable de un ingreso (precio x cantidad). Esto debido a que con frecuencia las organizaciones no conocen su demanda dado que no registran las ventas perdidas y no tienen un balance entre despachos entre otros.

La temporalidad es clave para dimensionar el requerimiento y se especifica en tres formas. El primero, el período del pronóstico que es la unidad básica de tiempo para la cual se realiza el pronóstico y depende del proceso en estudio, y el registro de las transacciones (diario, semanal, mensual o anual). El segundo, el horizonte de planeación es el número de períodos en el futuro cubiertos por el pronóstico. El último, el intervalo del pronóstico es la frecuencia con la que se efectúan los nuevos pronósticos.

Desde la mirada integral, Moon (2013) propone revisar cuatro dimensiones. Dimensión funcional para garantizar flujos de información soportada en comunicación, coordinación y colaboración. Dimensión tecnológica que involucra estrategia digital y política de datos que soportan el proceso. Integración de enfoque que implica alineamiento en los conceptos fundamentales (entendimiento de demanda, jerarquías, y técnica de pronóstico entre otros). Por último, dimensión de seguimiento y control para medir la efectividad del pronóstico.

4.1.1 *¿Qué es un buen pronóstico?*

Un pronóstico debe de estar alineado con el juicio de expertos (consistencia), tener similitud con la realidad (calidad) y generar beneficios de su uso (valor) (Murphy, 1993). La calidad se define en términos del sesgo, asociación, eficiencia, confiabilidad y capacidad (ver Apéndice A). Una medida que permite determinar el desempeño de un pronóstico es la precisión y se mide como la diferencia entre el valor real observado y su pronóstico.

$$e_t = y_t - \hat{y}_t \quad (1)$$

Donde e_t , y_t y \hat{y}_t son el error del pronóstico, el valor real y el pronóstico en el periodo t , respectivamente. Esta medida tiene la limitación de que tiene la misma escala de los datos, por lo cual no puede utilizarse para realizar comparativos entre series definidas en diferentes escalas. Otros indicadores de precisión dependientes de la misma escala utilizados son:

$$MAE = \frac{1}{t} \sum_{t=1}^T |e_t| \text{ y } RMSE = \sqrt{\frac{1}{t} \sum_{t=1}^T (e_t^2)} \quad (2)$$

Donde MAE = Error absoluto medio y RMSE = Raíz del Error cuadrático medio. Otra medida de error de pronóstico frecuentemente utilizada en la industria, por su no dependencia de la escala del conjunto de datos, es el error absoluto porcentual (Absolute Percentage Error):

$$APE = 100\% \cdot \left| \frac{y_t - \hat{y}_t}{y_t} \right| \text{ y } APE' = 100\% \cdot \left| \frac{y_t - \hat{y}_t}{\hat{y}_t} \right| \quad (3)$$

Estas medidas están dadas para un solo período y esto no es muy útil. Para disponer de indicadores para t períodos se utiliza la desviación absoluta media (MAD), el error cuadrático medio (ECM) y desviación absoluta porcentual media (MAPE).

$$MAD = \frac{\sum_{t=1}^T |y_t - \hat{y}_t|}{t} \text{ y } ECM = \frac{\sum_{t=1}^T (y_t - \hat{y}_t)^2}{t} \quad (4)$$

$$MAPE = \frac{1}{t} \sum_{t=1}^T \left| \frac{y_t - \hat{y}_t}{y_t} \right| \cdot 100\% \text{ o } MAPE' = \frac{1}{t} \sum_{t=1}^T \left| \frac{y_t - \hat{y}_t}{\hat{y}_t} \right| \cdot 100\% \quad (5)$$

Las medidas basadas en errores porcentuales tienen el inconveniente de volverse indefinida si $y_t = 0$ o $\hat{y}_t = 0$ para cualquier período t , tendrá valores extremos cuando cualquier y_t o \hat{y}_t sea cercana a cero. Esta y otras observaciones condujeron a la "simétrica" MAPE (SMAPE) (Armstrong j. s., 1985), el cual se define por:

$$SMAPE = \frac{1}{t} \sum_{t=1}^T \left(200 \cdot \frac{|y_t - \hat{y}_t|}{(y_t + \hat{y}_t)} \right) \quad (6)$$

Sin embargo, si y_t es cercano a cero, \hat{y}_t es también probable que sea cercano a cero. Por lo tanto, este indicador todavía implica que se pueda dar una división por un número cercano a cero, por lo que el cálculo resultaría inestable. Además, el valor de SMAPE puede ser negativo, por lo que no es realmente una medida de "errores porcentuales absolutos" en absoluto. Dados estos inconvenientes, se recomienda y se propone la utilización de los Errores Escalados como una alternativa (Hyndman & Koehler, 2006):

$$q_t = \frac{e_t}{\frac{1}{t-1} \sum_{t=2}^T |y_t - y_{t-1}|} \quad (7)$$

Un error de escala es menor que uno si se deriva de un pronóstico mejor que el pronóstico promedio calculado sobre los datos de entrenamiento. Por el contrario, es mayor que uno si el pronóstico es peor que el pronóstico promedio calculado sobre los datos de entrenamiento. El error medio escalado es simplemente:

$$MASE = \frac{1}{t} \sum_{t=1}^T |q_t| \quad (8)$$

Parte del ejercicio no es solo medir el error sino identificar la causa-raíz del error que no permite la generación de un buen pronóstico. Se consideran que un sistema de pronósticos no será eficiente si se utilizan datos poco confiables, desactualizados o insuficientes, cantidad de historia de ventas, sesgos profesionales, no lectura de cambios en el mercado, manejo no adecuado de datos atípicos y la no inclusión de la demanda no servida. Esto último, por incapacidad de los procesos de no generar datos sobre demanda no atendida, manejo de promociones y despachos fragmentados que imposibilitan tener la demanda agregada que se termina aproximando por facturación o despacho.

4.2 Técnicas cuantitativas para pronósticos

Una serie temporal es la sucesión de observaciones correspondientes a una variable a lo largo de intervalos regulares de tiempo. Se puede hablar de la serie en función de la periodicidad que se coleccionen los datos (diario, mensual, trimestral, anual, etc.). Entendiendo por proceso estocástico como un conjunto de variables X_t con $t \in T$ y $T \subseteq R$, donde las variables X_t toman valor en un espacio probabilístico definido, se define serie temporal como una realización de un proceso estocástico en tiempo discreto, donde los elementos T están ordenados y corresponden a instantes equiespaciados de tiempo.

El objetivo del análisis de las series temporales es el conocimiento de un patrón o secuencia de comportamiento que permita prever la evolución futura de la variable a estudio, suponiendo que las condiciones del entorno futuras no cambiarán con respecto a las condiciones que enmarcan las observaciones pasadas. Desde un punto de vista teórico y bajo el objetivo de comprender la evolución temporal de una determinada variable, el enfoque clásico de series temporales considera que el comportamiento de una variable en el tiempo es el resultado de la integración de movimientos regulares en el tiempo (tendencia), fluctuaciones repetidas no periódicas (ciclo), movimientos regulares y repetitivos en el tiempo (estacionalidad) y elementos no regulares que no siguen comportamientos sistemáticos (residual).

Existen métodos cualitativos y cuantitativos para conseguir pronósticos. Los cualitativos se utilizan ante ausencia de datos históricos y prevalece el criterio de expertos (ítems nuevos). Entre los cuantitativos están los basados en datos históricos, los que buscan causalidad o analogía y la simulación por sensibilización de parámetros. Esta sección abordará la teoría para realizar pronósticos a partir de series temporales específicamente, promedios móviles y los modelos ARIMA.

4.2.1 Promedio móvil

Esta técnica es una de las más simples, pero no menos útil. El promedio móvil es adecuado para patrones estables o perpetuos, con poca o ninguna tendencia. El modelo subyacente para este

tipo de procesos es $y_t = b + e_t$; Donde y_t = valor real observado en el periodo t , b = constante que representa el proceso uniforme y e_t = variable aleatoria normal con media cero y varianza desconocida (la parte aleatoria del proceso). El valor de b varia en diferentes momentos de la secuencia de tiempo, pero para un momento específico b es una constante. Si se tiene todas las T observaciones en el tiempo $y_1 + y_2 + \dots + y_T$ y asumiendo b como una constante, se puede calcular a partir de los mínimos cuadrados;

$$\sum_{t=1}^T e_t = \sum_{t=1}^T (y_t - b)^2 \quad (9)$$

Derivando respecto a b

$$\frac{\partial \sum_{t=1}^T e_t^2}{\partial b} = -2 \sum_{t=1}^T (y_t - b) = -2 \left(\sum_{t=1}^T y_t + Tb \right) \quad (10)$$

Igualando a cero Derivando respecto a b

$$\hat{y}_{t+1} = b = \frac{1}{T} \sum_{t=1}^T y_t \quad (11)$$

4.2.2 Modelo autorregresivo integrado de media móvil (ARIMA)

Los modelos Autorregresivos Integrados de Media Móvil (ARIMA) tienen su enfoque en describir las autocorrelaciones en los datos. En estos modelos se asume que la serie temporal, si es estacionaria, puede representarse por un modelo autorregresivo de media móvil ARMA (p, q) , siendo p el número de los términos autorregresivos y q el número de términos móviles:

$$Z_t^d = c + \phi_1(Z_{t-1}^d) + \dots + \phi_p(Z_{t-p}^d) + a_t - \theta_1 a_{t-1} - \dots - \theta_q a_{t-q} \quad (12)$$

Donde Z_t^d es la observación en el tiempo t de la serie diferenciada de orden d y a_t es una colección de variables aleatorias independientes e idénticamente distribuidas con $E(a_t) = 0$ y varianza $V(a_t) = \sigma_a^2$ conocida como ruido blanco. Estos modelos pueden ser representados con el operador de rezago del tiempo B , así $BZ_t = Z_{t-1}$ y $B^n Z_t = Z_{t-n}$. Bajo esta transformación la serie de tiempo estacionaria se puede escribir como $\phi(B)(Z_t) = \theta(B)a_t$, donde

$$\phi(B) = 1 - \phi_1 B - \dots - \phi_p B^p \text{ y } \theta(B) = 1 - \theta_1 B - \dots - \theta_q B^q \quad (13)$$

En ausencia de estacionariedad, es decir, que el nivel de la serie y variabilidad dependen del tiempo t ; se puede utilizar el operador diferencia ∇ para operar $\nabla Z_t = (1 - B)Z_t$ y $\nabla^d Z_t = (1 - B)^d Z_t$. Así, la serie no estacionaria Z_t , podrá ser representada por una serie ARIMA (p, d, q), de la siguiente forma:

$$\phi(B)\nabla^d(Z_t) = c + \theta(B)a_t \quad (14)$$

$$\varphi(B)(Z_t) = c + \theta(B)a_t \quad (15)$$

donde $\varphi(B) = \phi(B)\nabla^d$ que es un polinomio de orden $p + d$. La serie de tiempo Z_t también se puede representar como una función de transferencia lineal de la serie de ruido blanco $Z_t = \mu + \psi(B)a_t$, donde $\psi(B) = 1 + \psi_1 B + \psi_2 B^2 + \dots$ y puede ser calculado como $\psi(B) = \theta(B)/\phi(B)$.

Para el proceso de modelación es clave del análisis descriptivo de una serie de tiempo, esto es análisis de tendencia, ciclo, estacionalidad y aleatoriedad. Lo es también medir el grado de dependencia que muestran los datos del tiempo t con los datos de hace k períodos anteriores, para esto se utiliza la Función de Autocorrelación Simple (FAC) que adicionalmente ayuda a identificar si la serie es estacionaria cuando los valores de p_k tienden rápidamente a cero y el valor q del

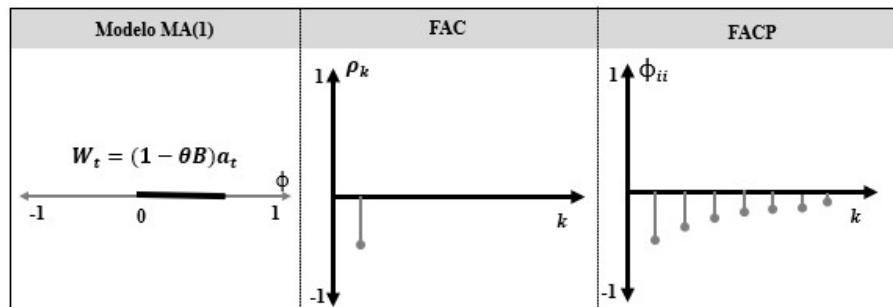
proceso MA cuando sus primeras p_i son distintas a cero para $i = 1, 2, \dots, q$. Se define el coeficiente de autocorrelación:

$$p_k = \frac{E[(Z_t - \mu)(Z_{t+k} - \mu)]}{E(Z_t - \mu)^2} \text{ para } k = 0, \pm 1, \pm 2, \dots \quad (16)$$

La FACP, que mide la autocorrelación entre Z_t y Z_{t+k} sin tener en cuenta la dependencia lineal de Z_{t+1} hasta Z_{t+k-1} , entrega información del componente autorregresivo. Si el proceso es $AR(p)$, las autocorrelaciones parciales $\widehat{\Phi}_{ii}$ se distribuyen de manera independiente, con media $E(\widehat{\Phi}_{ii}) = \phi_{ii}$ y con $Var(\widehat{\Phi}_{ii}) = 1/(T - d)$ para $i > p$. La Figura 1 muestra una FAC y FACP teóricas de un proceso MA(1).

Figura 1

Ejemplo de las gráficas FAC y FACP de un modelo MA(1)



Fuente: Guerrero, 2003.

De Box y Jenkins (1970) se pueden destacar cuatro etapas para la construcción de un modelo de serie temporal: Identificación, Estimación, Verificación y Predicción y elección.

Identificación. Se determinar el orden de los polinomios autorregresivos (p), de los promedios móviles (q) y las veces que debe aplicarse el operador diferencia (d). El primer paso es

la elección de una potencia λ de tal manera que $\sigma_t/\mu_t^{1-\lambda} = \text{constante}$ donde σ_t y μ_t representan la desviación estándar y la media de la variable Z_t y T es el número de observaciones de la serie $\{Z_t\}$ para $t = 1, 2, 3, \dots, T$. La transformación H resultante será:

$$H(Z_t) = \begin{cases} Z_t^\lambda & \text{si } \lambda \neq 0 \\ \log(Z_t) & \text{si } \lambda = 0 \end{cases} \quad (17)$$

La grafica de la FAC muestral de $\{H(Z_t)\}$, $\{\nabla H(Z_t)\}$ y $\{\nabla^2 H(Z_t)\}$ y su tendencia a cero rápidamente, indican que no se requiere diferenciación. La prueba de raíz unitaria Dickey Fuller aumentado, realiza el contraste formal tomando tres modelos con ϕ como parámetro de interés; si $\phi = 0$ el proceso posee una raíz unitaria.:

$$Z_t = \phi Z_{t-1} + a_t, \quad Z_t = \mu + \phi Z_{t-1} + a_t \quad \text{y} \quad Z_t = \mu\phi Z_{t-1} + \phi_2 Z_{t-2} + a_t \quad (18)$$

Si r_j es la estimación de p_j , el número de autocorrelaciones r_k distintas de cero de la FAC muestral da el orden q del modelo. Se dice que es significativamente distinta de cero si:

$$|r_k| > 2 \sqrt{(1/(T-d)) \left(1 + 2 \sum_{j=1}^q r_j^2 \right)} \quad \text{para } k > q \quad (19)$$

Determinar el orden del proceso $AR(p)$ se hace a partir del comportamiento de la FACP. Se establece que (ϕ_{ii}) es distinto de 0 si el valor calculado de $\widehat{\phi}_{ii}$ se encuentra fuera del intervalo definido por

$$\pm 2\sqrt{\text{Var}(\phi_{ii})} = \pm 2/\sqrt{T-d} \quad \text{para } i > p \quad (20)$$

Estimación. Identificado el proceso ARIMA (p,d,q) de la serie $H(Z_t)$, se estima el vector $\beta = (\phi_1, \dots, \phi_p, \theta_0, \theta_1, \dots, \theta_q)$ vía Mínimos Cuadrados o Máxima Verosimilitud, ambos métodos parten de $\{a_t\}$ y los supuestos $E(a_t) = 0$ y $V(a_t) = \sigma_a^2$. En el caso de Mínimos Cuadrados se busca $\min \sum_t a_t^2$ y en el caso de la función de verosimilitud, asumiendo que $a_t \sim N(0, \sigma_a^2)$, esta se deriva de la función de densidad conjunta de los a_t .

$$f(a_{d+p+1}, a_{d+p+2}, \dots, a_T) = (2\pi)^{-(T-d-p)/2} (\sigma)^{-(T-d-p)/2} e^{-\sum_{t=d+p+1}^T a_t^2 / 2\sigma_a^2} \quad (21)$$

Si $W_t = W^d H(Z_t)$; $a_t = W_t - \phi_1 W_{t-1} - \dots - \phi_p W_{t-p} - \theta_0 + \theta_1 a_{t-1} + \dots + \theta_q a_{t-q}$. Dado el potencial de estimaciones para W_t pero no para ϕ, θ y σ_a^2 , se hace de a_t una función $S(\phi, \theta_0, \theta)$

$$f(\phi, \theta_0, \theta, \sigma_a^2 | W) = (2\pi)^{-(T-d-p)/2} (\sigma_a)^{-(T-d-p)/2} e^{S(\phi, \theta_0, \theta) / 2\sigma_a^2} \quad (22)$$

De la Ecuación 22 y suponiendo que se tiene estimación para $S(\phi, \theta_0, \theta)$. se maximiza la función de verosimilitud $\ell(\sigma_a^2 | W; \hat{\phi}, \hat{\theta}_0, \hat{\theta}) = \log[L(\sigma_a^2 | W; \hat{\phi}, \hat{\theta}_0, \hat{\theta})]$:

$$\frac{\partial \ell}{\partial \sigma_a^2} = \frac{-T + d + p}{2\sigma_a^2} + \frac{S(\hat{\phi}, \hat{\theta}_0, \hat{\theta})}{2\sigma_a^4} = 0 \rightarrow \sigma_a^2 = \frac{S(\hat{\phi}, \hat{\theta}_0, \hat{\theta})}{T - d - p} \quad (23)$$

Así maximizar el $L(\phi, \theta_0, \theta, \sigma_a^2 | W)$ se reduce a minimizar $S(\phi, \theta_0, \theta)$. Guerrero (2003), citando a Box y Jenkins (1970), sugiere el algoritmo de Marquardt para dicho calculo.

Verificación. Esto implica análisis para $\hat{a}_t = W_t - \widehat{W}_t$. Para los supuestos $E(a_t) = 0$ y $V(a_t) = \sigma_a^2$ se parte de la media y la varianza de \hat{a}_t :

$$m(\hat{a}) = \frac{1}{T-d-p} \sum_{t=t'}^T \hat{a}_t \quad y \quad \hat{\sigma}_a = \sqrt{\sum_{t=t'}^T [\hat{a}_t - m(\hat{a})]^2 / (T-d-p)} \quad (24)$$

Si se cumple $|\sqrt{T-d-p} m(\hat{a})/\hat{\sigma}_a| \geq 2$, entonces la media de los residuos es distinta de cero. Para validar varianza constante de $\{a_t\}$ se grafican los residuos contra el tiempo y se observa si la varianza es constante o presenta algún tipo de tendencia en el tiempo. Además, se debe validar que las $\{a_t\}$ son mutuamente independientes, y dado que independencia implica no autocorrelación se requiere que $\rho_k(a) = 0$ para toda $k \neq 0$. De la FAC muestral de los residuos:

$$r_k(\hat{a}) = \frac{\sum_{t=t'}^{N-k} \hat{a}_t \hat{a}_{t+k}}{\sum_{t=t'}^N \hat{a}_t^2}, \quad k = 1, 2, \dots \quad y \quad t' = d + p + a \quad (25)$$

Si la media 0, la desviación estándar de $r_k(\hat{a}) = \sqrt{Var[r_k(\hat{a})]} = 1/\sqrt{N-d-p}$, entonces si $|r_k(\hat{a})| \geq 2/\sqrt{N-d-p}$ se dirá que la autocorrelación k-ésima es distinta de cero. Formalmente, el estadístico Q' que sigue una distribución ji-cuadrado con $k-p-q$ grados de libertad es un test para validar autocorrelación de los residuos.

$$Q' = (T-d-p)(T-d-p+2) \sum_{k=1}^K r_k^2(\hat{a}) / (T-d-p-k) \quad (26)$$

Verificar normalidad de a_t se puede hacer vía histograma de residuos o identificando las observaciones por fuera del intervalo $(-2\hat{\sigma}_a, 2\hat{\sigma}_a)$; estas no deben superar $(N-d-p)/20$. Otros supuestos a validar es identificar observaciones fuera de $(-3\hat{\sigma}_a, 3\hat{\sigma}_a)$, que no se puedan reducir el número de parámetros (parsimonia) verificando que el intervalo de confianza para cada parámetro no debe pasar por 0 y, por último, estabilidad en parámetros (no correlacionados).

Predicción y elección del modelo. Los predictores óptimos son aquellos que minimizan, en promedio, los errores de predicción al cuadrado y se obtienen calculando los valores esperados de observaciones futuras, condicionadas a datos observados. De $W_t = \nabla^d T(Z_t)$, si desde t se quiere pronosticar la observación W_{t+h} , este será la combinación lineal entre los valores de W_t , y por ende de los errores a_t y será denotado como $\tilde{W}_t(h)$; el pronóstico óptimo será denotado por $\hat{W}_t(h)$. El pronóstico óptimo minimiza el error cuadrático medio:

$$E_t[W_{t+h} - \hat{W}_t(h)]^2 = \min_{\tilde{W}_t(h)} E_t[W_{t+h} - \tilde{W}_t(h)]^2 \quad (27)$$

Donde E es la esperanza condicional, dada toda la información hasta el momento t :

$$E_t[W_{t+h} - \tilde{W}_t(h)]^2 = E\{[W_{t+h} - \tilde{W}_t(h)]^2 | Z_t, Z_{t-1}, \dots\} \quad (28)$$

Dado que $W_{t+h} = -\sum_{j=0}^{\infty} \psi_j a_{t+h-j}$; $\psi_0 = 1$ la predicción óptima será las esperanzas condicionadas de las primeras T observaciones $\hat{W}_t(h) = \sum_{j=0}^{\infty} \psi_j a_{t+h-j}$ y el error de pronóstico mínimo viene dado por:

$$e_t(h) = W_{t+h} - \hat{W}_t(h) = \sum_{j=0}^{h-1} \psi_j a_{t+h-j} \quad (29)$$

Con $E_t[e_t(h)] = 0$ y $Var_t[e_t(h)] = \sum_{j=0}^{h-1} \psi_j^2 \sigma_a^2$. El intervalo de confianza vendría dado:

$$\hat{W}_t(h) \pm Z_{\frac{\alpha}{2}} * \sqrt{Var_t[a_t(h)]} \quad (30)$$

La elección del modelo se da por el Criterio de Información de Akaike (AIC), AIC Corregido (AICC) y el Criterio de Información Bayesiana (BIC). El criterio AIC parte de la función de verosimilitud de un modelo ARIMA:

$$L(\beta) = -\frac{T}{2} \ln \sigma^2 - \frac{1}{2} \sum_{t=1}^T \ln v_{t|t-1} - \frac{1}{2\sigma^2} \sum_{t=1}^T \frac{a_t^2}{v_{t|t-1}} \quad (31)$$

Multiplicando por -2, estimando los parámetros se estiman con los datos y se calcula la esperanza de esta verosimilitud respecto a futuras observaciones se tiene:

$$E(-2L(\beta)) = T \ln \sigma_{MV}^2 + 2k \quad (32)$$

T es el tamaño muestral utilizado para estimar en modelo, σ_{MV}^2 el estimador MV de la varianza de los errores y k el número de parámetros estimados para calcular las predicciones. Este criterio podría sobrestimar el número de parámetros en el modelo y surge el AIC corregido:

$$AICC = T \ln \sigma_{MV}^2 + T \frac{1 + k/T}{1 - (k + 2)/T} \quad (33)$$

BIC es un criterio Bayesiano alternativo propuesto por Schwarz (1978) basado en maximizar la probabilidad a posteriori del modelo dada la serie y tiende a elegir modelos más parsimoniosos.

$$BIC = T \ln \sigma_{MV}^2 + k \ln T \quad (34)$$

Una variación de los modelos ARIMA, es el modelo SARIMA (Seasonal ARIMA) e incorpora factores estacionales y no estacionales (modelo multiplicativo). Definido como $SARIMA(p, d, q)(P, D, Q)_E$ con p , q y d como el orden AR, MA y la diferenciación del componente no estacional y P , Q y D como el orden AR, MA y la diferenciación del componente estacional. Este modelo se puede escribir

$$(B^E)\nabla_E^D(Z_t - \mu) = (B^E)\alpha_t \quad (35)$$

$$\phi(B)\phi(B^E)\nabla^d\nabla_E^D(Z_t - \mu) = \theta(B)\theta(B^E)\alpha_t \quad (36)$$

Donde $\phi(B) = (1 - \phi_1 L, \dots, \phi_p B^p)$ es el polinomio AR regular de orden p , $\phi(B^E) = (1 - \phi_1 L^E, \dots, \phi_p B^{Ep})$, es el polinomio AR estacional de orden P , $\theta(B) = (1 - \phi_1 L, \dots, \phi_q B^q)$ es el polinomio MA regular de orden q , $\theta(B^E) = (1 - \phi_1 L^E, \dots, \phi_q B^{Eq})$, es el polinomio MA estacional de orden Q , ∇_E^D y ∇^d representan las diferencias estacionales y regulares respectivamente, y α_t es un proceso ruido blanco. Un supuesto de este modelo es que la relación de dependencia estacional es la misma para todos los períodos. En la práctica, se deben seguir los mismos pasos descritos para modelos ARIMA, esto es identificación, estimación, validación y pronóstico).

4.3 Algoritmos de aprendizaje automático: de los árboles de decisión al XGBoost

Algoritmos de aprendizaje automático o Machine Learning (ML) es una rama de la Inteligencia Artificial (IA) que ha tomado relevancia dado el volumen de datos disponibles, la capacidad computacional y versatilidad para resolver problemas a partir de algoritmos matemáticos. Se clasifican en aprendizaje supervisado si se utilizan entradas para predecir una salida, no supervisado si se hace inferencias directamente de una población y de refuerzo por la mejora continua a partir de premios. A continuación, se presentan los algoritmos de aprendizaje supervisado XGBoost y Prophet.

4.3.1 De los árboles de decisión al XGBoost

El algoritmo XG Boost (Extreme Gradient Boosting) es una técnica de aprendizaje supervisado (Chen y Guestrin, 2016) que es evolución de los árboles de decisión (ver Figura 2).

Figura 2

Evolución de los árboles de decisión



Nota. Adaptado de Espinosa, J, 2020.

Árboles de decisión. Dado p inputs predictores y N observaciones: (x_i, y_i) para $i = 1, 2, \dots, N$, con $x_i = (x_{i1}, x_{i2}, \dots, x_{ip})$; el algoritmo busca dividir el espacio de las variables predictoras X_p en M regiones disjuntas R_1, \dots, R_M . La respuesta a la predicción viene dada por

$$f(x) = \sum_{m=1}^M c_m I(x \in R_m) \quad (37)$$

Donde la constante c_m es definida para cada región. La división binaria recursiva, se logra identificando las regiones $R_1(j, s) = \{X | X_j \geq s\}$ y $R_2(j, s) = \{X | X_j < s\}$ que reduzca lo máximo posible la suma de cuadrados $RRS = \sum (y_i - f(x_i))^2$. Es decir,

$$\min_{j,s} \left[\min_{c_1} \sum_{i \in R_1(j,s)} (y_i - c_1)^2 + \min_{c_2} \sum_{i \in R_2(j,s)} (y_i - c_2)^2 \right] \quad (38)$$

La solución para $\hat{c}_1 = \frac{1}{n} \sum_{i \in R1(j,s)} (y_i)$ y $\hat{c}_2 = p \frac{1}{n} \sum_{i \in R2(j,s)} (y_i)$ para cada selección de j y s . Se debe repetir el proceso hasta que se alcanza un criterio de detención. Debido al tamaño del árbol se puede producir sobreajuste. Tener menos regiones R_m puede conducir a menor varianza y mejorar la interpretación, para lo cual se debe definir un criterio coste-complejidad para seleccionar un árbol $T \subset T_0$, donde T_0 es el árbol inicial.

$$C_\alpha(T) = \sum_{m=1}^{|T|} \sum_{x_i \in R_m} (y_i - \hat{c}_m)^2 + \alpha |T| \quad (39)$$

Buscando que $C_\alpha(T)$ sea lo más pequeño posible. $|T|$ indica el número de nodos terminales del árbol T , R_m es la región correspondiente al m -ésimo nodo terminal de T , y \hat{c}_m es el valor de predicción en la región R_m . El parámetro de ajuste α controla una compensación entre la complejidad o tamaño del subárbol y su ajuste a los datos de entrenamiento. Cuando $\alpha = 0$, entonces el subárbol T será igual a T_0 . Se puede seleccionar un valor de α usando validación cruzada.

Los árboles de decisión presentan problema de equilibrio entre sesgo y varianza, esto es un árbol de alta presión con los datos de entrenamiento puede cambiar los resultados al utilizar los datos de validación. Para dar solución a este problema surgen los métodos de ensamble, que combinan múltiples modelos en uno nuevo, con el objetivo de generar balance entre sesgo y varianza. Dos de los tipos de los métodos de ensamble más utilizados son Bagging y Boosting.

. Si se tiene un conjunto n de observaciones independientes Z_i cada una con varianza σ^2 ; la varianza de $\bar{Z} = (\sum_{i=1}^n Z_i/n)$ viene dada por σ^2/n ; esto es, promediar el conjunto de observaciones implica reducir la varianza y, así, incrementar el poder de predicción. Específicamente, calcular $\hat{f}^1(x), \hat{f}^2(x), \dots, \hat{f}^B(x)$ en el caso de B conjuntos de entrenamiento distintos y realizar la media de todos ellos:

$$\hat{f}_{bag}(x) = \frac{1}{B} \sum_{b=1}^B \hat{f}^{*b}(x) \quad (40)$$

Usar un número elevado de árboles no conduce al sobreajuste y se utiliza el valor B que ha maximizado la reducción del error. Dado que el Bagging implica la construcción de muchos árboles, su interpretación es compleja. Esto se resuelve identificando la importancia de cada predictor a partir de su aporte a la reducción del RSS (regresión).

Un predictor muy influyente llevaría a un escenario donde casi todos los árboles creados estarán dominados por el mismo predictor y serán muy parecidos entre ellos. Como consecuencia el proceso apenas conseguirá disminuir la varianza y, por lo tanto, tampoco mejorará el modelo. Random Forest es una técnica que mitiga este problema.

Random Forest. El algoritmo Random Forest o bosques aleatorios (Breiman, 2001) es una técnica que genera múltiples árboles de decisión sobre un conjunto de datos y sus resultados son combinados a fin de obtener un modelo único y más robusto en comparación con los resultados de cada árbol por separado (Lizares, 2017). Cada árbol es construido a partir de un subconjunto aleatorio de variables m (predictores) de forma que $m < P$ (P total predictores); se sugiere $m \approx p/3$ para problemas de regresión. Una función general para la regresión sería:

$$f_{rf}^B(x) = \frac{1}{B} \sum_{b=1}^B T_b(x, \theta_b) \quad (41)$$

θ_b corresponde a los hiperparámetros que caracterizan el número y los valores de los nodos terminales de cada árbol. La secuencia del algoritmo se muestra en la Tabla 1.

Dado que los árboles se ajustan empleando muestras generadas por Bootstrapping conlleva que, en promedio, cada ajuste utilice solo dos tercios de las observaciones originales. Al tercio restante se le llama out of bag (OOB). El OOB se utiliza para la validación cruzada registrando las observaciones empleadas y prediciendo la respuesta de la observación i haciendo uso de aquellos árboles en los que esa observación ha sido excluida y promediándolos.

Tabla 1*Algoritmo random forest*

-
- Para $b = 1$ hasta B
 - Obtener una muestra por método Bootstrap Z^* de tamaño N del conjunto de datos de entrenamiento.
 - Generar un árbol del bosque aleatorio T_b con los datos de la muestra Z^* .
 - Seleccione m variables al azar de las p variables.
 - Elija la mejor variable/punto de división entre los m al minimizar el RSS.
 - Divida el nodo en dos nodos secundarios.
 - Se repite la secuencia anterior de manera recursiva hasta alcanzar un criterio de parada. Puede ser el tamaño mínimo de nodo n_{min} , que el árbol tenga un máximo de nodos terminales o que la incorporación del nodo reduzca el error en al menos un porcentaje dado.
 - Las salidas de todos los árboles se combinan en una salida final $\{T_b\}_1^B$
-

Este algoritmo es simple de entrenar comparado con técnicas más complejas logrando rendimiento similar, es certero en bases de datos grandes, estima los predictores más importantes y, mantiene su precisión con proporciones grandes de datos perdidos. Sin embargo, hay dificultad para interpretar resultados desde lo gráfico, se tiene poco control sobre lo que hace el modelo y las predicciones se hacen en el rango de valores del conjunto de entrenamiento.

Modelos Boosting basados en árboles. El Boosting, introducido por Schapire (1990), mejora la capacidad predictiva de métodos de aprendizaje de alta variabilidad. La idea es generar predicciones a partir de una variable de salida Y dado un vector de variables a partir un conjunto de predictores $G(X)$ débiles. Entendiendo débil como aquel con baja tasa de acierto en el

pronóstico. El Boosting para árboles de regresión puede iniciar expresando:

$$T(x, \theta) = \sum_{m=1}^M \gamma_m I(x \in R_j) \quad (42)$$

Donde $\theta = \{R_m, \gamma_m\}$ con $m = 1, 2, \dots, M$. Se obtiene $\hat{\theta} = \min_{\theta} \sum_{j=1}^J \sum_{x_i \in R_j} L(y_i, \gamma_m)$, esto es minimizar la función de pérdida. A su vez, identificar R_m implica:

$$\tilde{\theta} = \min_{\theta} \sum_{j=1}^J \sum_{x_i \in R_j} \tilde{L}(y_i, T(x_i, \theta)) \quad (43)$$

Se obtiene a partir de modelos de desarrollo aditivo (Apéndice B):

$$\hat{\theta}_m = \min_{\theta_m} \sum_{j=1}^J L(y_i, f_{m-1}(x_i) + T(x_i, \theta_m)) \quad (44)$$

Donde dadas las regiones R_{jm} estimar el valor γ_{jm} se realiza a partir de minimizar:

$$\hat{\gamma}_{jm} = \min_{\gamma_{jm}} \sum_{j=1}^J L(y_i, f_{m-1}(x_i) + \gamma_{jm}) \quad (45)$$

En casos de regresión es encontrar el $\hat{\theta}_m$ que obtenga mejor predicción para los residuos actuales: $y_i - f_{m-1}(x_i)$; donde $\hat{\gamma}_{jm}$ viene dado por la media de las observaciones en cada región. Para clasificación se debe buscar la solución $\hat{\theta}_m$ (utilizando pérdida exponencial) que minimiza la

tasa de error ponderado $\sum_{i=1}^N w_i^m I(y_i \neq T(x_i, \theta_m))$ con pesos $w_i^{m1} = e^{-y_i f_{m-1}(x_i)}$. Así se tienen las expresiones

$$\hat{\theta}_m = \min_{\theta_m} \sum_{j=1}^J w_i^m e^{-y_i T(x_i, \theta_m)} \quad (46)$$

$$\hat{y}_{jm} = \log \frac{\sum_{x_i \in R_{jm}} w_i^m I(y_i = 1)}{\sum_{x_i \in R_{jm}} w_i^m I(y_i = -1)} \quad (47)$$

Resolver estas optimizaciones requiere algoritmo especializados como el Descenso Rápido o el Gradient Boosting. Partiendo de que cualquier función de pérdida diferenciable se puede resolver vía optimización numérica y dada la pérdida total:

$$L(f) = \sum_{i=1}^N L(y_i, f(x_i)) \quad (48)$$

El objetivo principal es minimizar $L(f)$ con respecto a f tanto como sea posible, donde $f(x)$ es la suma de árboles $f_M(x) = \sum_{m=1}^M T(x, \theta_m)$. Se puede entonces ver como una optimización numérica $\hat{f} = \min_f L(f)$, donde los parámetros de f e R^N son los valores de la función $f(x_i)$ de cada uno de los N puntos de datos x_i : $f = (f(x_1), f(x_2), \dots, f(x_N))$; lo que se puede expresar como una suma de componentes vectoriales $f = \sum_{i=1}^M h_m$ con $h_m \in R^N$ donde $f_0 = h_0$ y luego f_m se calcula teniendo en cuenta el valor de f_{m-1} , el cual es la suma de las actualizaciones inducidas previamente.

El método de descenso rápido elige $h_m = \rho_m g_m$ donde $\rho_m = \min_{\rho} L(f_{m-1} - \rho g_m)$ es un escalar y $g_m \in R^N$ es el gradiente $L(f)$ evaluado en $f = f_{m-1}$. Esto es,

$$g_{im} = \left[\frac{\partial L(y_i, f(x_i))}{\partial f(x_i)} \right]_{f(x_i)=f_{m-1}(x_i)} \quad (49)$$

El Gradient Boosting busca en cada paso el árbol que dado un f_{m-1} , más reduce $\hat{\theta}_m = \min_{\theta_m} \sum_{j=1}^J L(y_i, f_{m-1}(x_i) + T(x_i, \theta_m))$. Es decir, cada predicción dada por $T(x_i, \theta_m)$ es análoga al gradiente de descenso. Aquí los componentes procedentes de los árboles $t_m = T(x_1, \theta_m), \dots, T(x_N, \theta_m)$ no son independientes y están obligados a ser predicciones de un árbol de nodo J_m . En el caso del gradiente no hay restricción de este tipo; solo la dirección de máximo descenso. La solución para \hat{y}_{jm} y $\rho_m = \min_{\rho} L(f_{m-1} - \rho_m g_m)$ es similar que en el método de máximo descenso. El paso a paso para la implementación en regresión se puede ver en la tabla 2.

Tabla 2*Implementación del Gradient Boosting en regresión*

-
- Iniciar con $f_0(x) = \sum_{i=1}^N L(y_i, \gamma)$
 - Desde $M = 1$ hasta $m = M$ calcular
 - Para $i = 1, 2, \dots, N$, $r_{im} = \left[\frac{\partial L(y_i, f(x_i))}{\partial f(x_i)} \right]_{f=f_{m-1}}$
 - Ajustar un árbol de regresión a los objetivos r_{im} especificando las regiones terminales $R_{jm}, j = 1, 2, \dots, J_N$
 - Para cada región calcular: $\gamma_{jm} = \min_{\gamma} \sum_{x_i \in R_{jm}} L(y_i, f_{m-1}(x_i) + \gamma)$
 - Actualizar $f_m(x) = f_{m-1}(x) + \sum_{j=1}^{J_m} \gamma_{jm} I(x \in R_{jm})$
 - Como salida $\hat{f}(x) = f_M(x)$
-

Algoritmo XGBoost. Desarrollado por Chen y Guestrin (2016), este algoritmo viene dominando algunas de las competencias más importante a nivel de aprendizaje automático con la idea de mejorar el desempeño de un aprendiz débil (el árbol de decisión) adicionando arboles secuencialmente para que al fusionarse se logre un modelo más preciso, con mayor eficacia, velocidad computacional y el rendimiento del modelo:

$$f_M(x) = \sum_t f_t(x_i) \quad (50)$$

Donde f_t es el árbol independiente $T(x_i, \theta_t)$ para el tiempo t . Dado

$$f_t(x) = f_{t-1}(x_i) + T(x_i, \theta_t) \quad (51)$$

Se obtiene la función de perdida a minimizar:

$$E_t = \sum_{i=1}^N E(y_i, f_{t-1}(x_i) + T(x_i, \theta_t)) + \Omega(f_t) \quad (52)$$

El termino $\Omega(f_t)$ es de regularización. De una aproximación de segundo grado:

$$E_t = \sum_{i=1}^N [g_i T(x_i, \theta_t) + \frac{1}{2} h_i T(x_i, \theta_t)^2] + \Omega(f_t) \quad (53)$$

Donde $g_i = [\partial L(y_i, f(x_i))/\partial f(x_i)]_{y=y_{t-1}}$ y $h_i = [\partial^2 L(y_i, f(x_i))/\partial f(x_i)^2]_{y=y_{t-1}}$ son los gradientes de primer y segundo orden de la función de pérdida definida. Este proceso se sigue de forma iterativa hasta que el error es minimizado con el árbol final f_m . Entre las ventajas del algoritmo están la de manejar grandes bases de datos con múltiples variables, se puede trabajar con valores perdidos y hace procesamiento en paralelo de todos los parámetros ajustables que se controlan en el entrenamiento como tamaño de árbol y del nodo (hiperparametros). Entre las desventajas está que solo trabaja con vectores numéricos.

4.3.2 Algoritmo Phophet

Prophet es un modelo regresivo aditivo $y(t) = g(t) + s(t) + h(t) + \varepsilon(t)$ donde $g(t)$ describe la tendencia, $s(t)$ define la estacionalidad, $h(t)$ concreta eventos (vacaciones, feriados, paros, etc.) y $\varepsilon(t)$: son los efectos no explicados por el modelo. El modelo de tendencia $g(t)$ está basado en un modelo de saturación de crecimiento y un modelo lineal por partes para explicar el valor y en el momento t , definido por:

$$g(t) = \frac{C}{1 + e^{-k(t-m)}} \quad (54)$$

Con C como capacidad máxima de crecimiento, k es la tasa de crecimiento, y m un parámetro de compensación. Facebook propone remplazar el valor fijo de capacidad de crecimiento C fijo por una capacidad móvil $C(t)$ y no asumir tasa de crecimiento contante k .

Para puntos de cambio de tendencia S para los tiempos $s(j)$ con $j = 1, 2, \dots, S$, se define el vector de tasas de ajuste a $\delta \in R^S$, donde δ_j es delta de cambio de tasa de crecimiento en el momento $s(t)$. Se define la tasa de crecimiento hasta t como $k + \sum_{j:t>s} \delta_j$. Definiendo un vector $a(t) \in \{0,1\}^S$ de la forma:

$$a_j(t) = \begin{cases} 1, & \text{si } t \geq s_j \\ 0, & \text{otro caso} \end{cases} \quad (55)$$

Definiendo entonces la tasa de crecimiento en el momento t como $k + a(t)^T \delta$. Con el parámetro k ajustado, Es necesario estabilizar el parámetro de corrección m para cada cambio j .

$$\gamma_j = \left(s_j - m - \sum_{l < j} \gamma_l \right) + \left(1 - \frac{k + \sum_{l < j} \delta_l}{k + \sum_{l \leq j} \delta_l} \right) \quad (56)$$

Se define entonces, el modelo logístico:

$$g(t) = \frac{C(t)}{1 + e^{-(k+a(t)^T \delta)[t - (m+a(t)^T \gamma)]}} \quad (57)$$

En caso de que el modelo no muestre necesidad de tener una tasa de saturación $C(t)$ por tener capacidad ociosa, se utiliza la tendencia lineal $(k+a(t)^T \delta)[t - (m+a(t)^T \gamma)]$.

El punto de cambio de tendencia s_j puede ser especificado por un especialista o ser seleccionado automáticamente. Asumiendo que $\delta_j \sim \text{Laplace}(0, \tau)$ se pueden remplazar τ por la varianza inferida desde los datos. A partir del marco bayesiano y posterior uso de la máxima verosimilitud se tiene que $\lambda = \frac{1}{S} \sum_{j=1}^S |\delta_j|$ es el parámetro de escala:

$$\forall j > T, \begin{cases} \delta_j = 0, & \text{con probabilidad } \frac{T-S}{T} \\ \delta_j \sim \text{Laplace}(0, \lambda). & \text{con probabilidad } \frac{S}{T} \end{cases} \quad (58)$$

Con esta medida se asegura que en el pronóstico se tendrán puntos de cambio con la misma frecuencia y magnitud según lo visto en la historia.

El modelo de ciclo $s(t)$ recurre a la serie de Fourier para modelar ciclos definida como la sumatoria infinita que converge a una función periódica, continua y puede ser expresada:

$$s(t) = \sum_{n=1}^N \left(a_n \cos\left(\frac{2\pi nt}{P}\right) + b_n \sin\left(\frac{2\pi nt}{P}\right) \right) \quad (59)$$

Donde P es el período regular ($P = 7$ semanal y $P = 365.25$ anual). Se requiere estimar $2N$ parámetros $\beta = [a_1, b_1, \dots, a_N, b_N]^T$ mediante la construcción de un vector de estacionalidad por cada valor de t del historial y futuro de los datos. Para una estacionalidad anual y $N = 10$:

$$X(t) = \left[\cos\left(\frac{2\pi 1t}{365.25}\right), \dots, \sin\left(\frac{2\pi 10t}{365.25}\right) \right] \quad (60)$$

Con esto el componente estacional será $s(t) = X(t)\beta$ con $\beta \sim N(0, \sigma^2)$. Para el ingreso de días feriados $h(t)$, definiendo los eventos i (que afecta el normal comportamiento de la serie) sea D_i el conjunto de pasados y fechas futuras para ese evento, se agrega una función de indicador que representa si el tiempo t es durante el evento i , y asigna a cada feriado un parámetro k_i que es el correspondiente cambio en el pronóstico.

$$Z(t) = [1(t \in D_1), \dots, 1(t \in D_L)] \quad (61)$$

$$h(t) = Z(t)k \text{ con } k \sim N(0, v^2) \quad (62)$$

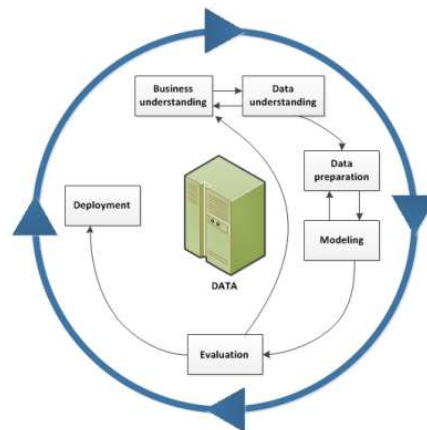
Se ha descrito los fundamentales de la gestión de demanda y el rol del pronóstico de ventas, así como la teoría sobre promedio móviles (método actual de proyección), modelos ARIMA y los algoritmos de aprendizaje automático XGBoost y Prophet como herramientas para cumplir los objetivos planteados. A continuación, se despliega la estrategia metodológica.

5 Metodología

Para dar respuesta a los objetivos planteados en este proyecto, se revisó las metodologías CRISP-DM, SEMMA y Catalyst. CRISP-DM, que incluye la comprensión del negocio para generar eficiencias de modelación desde el inicio, fue la adoptarla como referencia para el desarrollo de este proyecto de grado. A continuación, se describen las fases de CRISP-DM.

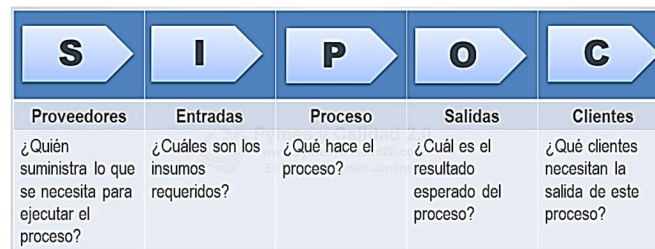
Figura 3

Fases de la metodología CRISP-DM



Fuente: IBM

Fase 1 - Comprensión del negocio. Esta parte de la metodología queda cubierta en el primer capítulo del presente trabajo con los objetivos, planteamiento y justificación del problema. Sin embargo, para entender de cerca el proceso de realizar un pronóstico estadístico de ventas realizado por la compañía farmacéutica, se utilizará la metodología SIPOC (Proveedores, Entradas, Proceso, Salidas y Clientes) para identificar a nivel cualitativo los actores claves y sus relaciones, los insumos, entregables, los acuerdos de servicio y el flujo de información del proceso de pronóstico.

Figura 4*Esquema del SIPOC*

Fuente: Hernández, J. 2019.

Fase 2 - Comprensión de los datos. La construcción de las series históricas de ventas de los componente115 y componente141 la hace el equipo de planeación de demanda. Los equipos de mercadeo y ventas harán un resumen cualitativo de eventos temporales que han impactado las ventas de los componentes. Se mencionó en la justificación, que la selección de estos dos componentes se da por ser referencia de marca, su presencia en todas las geografías en las líneas comercial e institucional, son parte del Pareto del ingreso, están en el top de productos con mayor margen de rentabilidad y dado su MAPE histórico promedio del 70%, corregirlo implica disminuir el MAPE global que es del 75%.

La serie componente115 cuenta con datos mensuales de ventas desde noviembre de 2014 y el componente141 desde agosto de 2010. No se cuenta con información de covariables y se realizaran pronósticos basados en la autocorrelación temporal. La inclusión del criterio de experto no es del alcance de este modelo.

Fase 3 - Preparación de los datos. Para dar cumplimiento al objetivo planteado se realizará un análisis exploratorio de las series de datos para identificar presencia de tendencia y estacionalidad, puntos atípicos, datos faltantes, continuidad histórica y se graficará las funciones de ACF, PACF y demás que permitan guiar las modelaciones. Según lo conversado con planeación de demanda, las dos series de tiempo cuentan con datos ininterrumpidos en el tiempo. Estos productos no registran participación en promociones por lo que no se observará comportamientos

crecientes de ventas por efectos de canibalismo de marca y/o producto. En caso de detectar datos atípicos se revisará el origen de este y se remplazará por el valor del componente tendencial para ese momento.

Fase 4 – Modelado. En esta fase, se realizará el modelado de datos siguiendo la metodología ARIMA, XGBoost y Prophet; técnicas elegidas en función del objetivo del problema. Se procederá de la siguiente manera:

Tabla 3

Secuencia para la modelación

-
- Separar los datos en dos grupos, entrenamiento y prueba. Se dejarán las últimas doce observaciones mensuales como grupo de pruebas. Esto alineado con la necesidad de lograr pronósticos con una ventana temporal de doce meses.
 - Crear una línea base de comparación. En este caso sería el promedio de los pronósticos obtenidos por los promedios móviles de 3, 6 y 12 meses. Las métricas de desempeño de estos modelos serán entregadas por el área de planeación de demanda.
 - Realizar la estimación del modelo ARIMA (SARIMA). Realizar las proyecciones y calcular las pruebas de eficiencia. Estos modelos tendrán las respectivas pruebas sobre los supuestos.
 - Realizar la estimación del modelo XGBoost. Realizar las proyecciones, optimización de hiperparámetros y calcular las pruebas de eficiencia.
 - Realizar la estimación del modelo PROPHET. Realizar las proyecciones, optimización de hiperparámetros y calcular las pruebas de eficiencia.
-

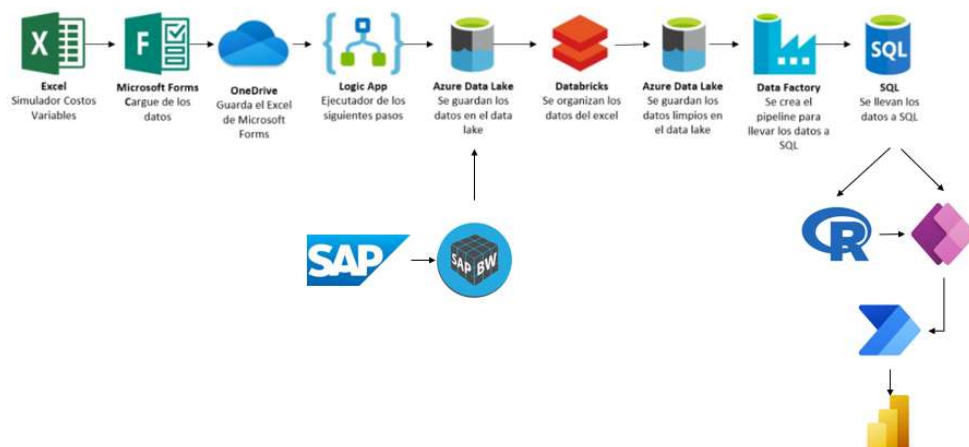
Fase 5 – Evaluación. Se evalúa el modelo teniendo en cuenta el cumplimiento de los criterios de éxito del problema y considerando que la fiabilidad para el modelo se aplica solamente para los datos sobre los que se realizó el análisis. Para la selección del modelo ARIMA (SARIMA) se utilizará los criterios BIC y AIC. Posteriormente el mejor de los modelos ARIMA (SARIMA) se comparará con el Prophet y XGBoost según su eficiencia teniendo en cuenta las medidas de desempeño MAPE, RMSE, MSE y SMAPE.

Fase 6- Implementación. No la cubre el presente trabajo. Sin embargo, la fase de comprensión del negocio y entendimiento del problema entrega a través del taller SIPOC y el levantamiento del flujo del proceso, información relevante para gestionar el conocimiento interno para una implementación óptima. Desde la fase de modelación saldrán algunas recomendaciones relevantes para esta fase.

La figura 5 muestra la arquitectura definida para implementar el proceso de pronóstico colaborativo una vez se valide el PMV y se escale a la totalidad de los productos. Los datos del presupuesto ajustado (pronóstico comercial) llegan desde Excel y se integran junto con los datos históricos de ventas y el presupuesto, que vienen de SAP, a Azure Data Lake. Con Databricks se hace ingeniería de datos y pasan al Data Factory donde se extraen con SQL. Los datos van a R para realizar el pronóstico estadístico y este posteriormente se integra a Power App para realizar el consenso. El pronóstico estadístico, el presupuesto, el presupuesto ajustado y el consenso van a Power BI donde se construirán los tableros de seguimiento y control al pronóstico y se estimara los ingresos para las reuniones con el comité directivo.

Figura 5

Arquitectura de la solución



6 Resultados

Una vez definido el problema, así como los métodos y metodología para dar cumplimiento a los objetivos, se dispone a plasmar los resultados y análisis de estos. Se seguirá un hilo conductor dado por las fases de la metodología CRISP-DM y se analizarán en paralelo los dos componentes.

6.1 Fase 1 - Comprensión del negocio o problema

Se han dedicado la primera parte del trabajo a definir y justificar el problema, así como plantear los objetivos de investigación. Para entender el papel del pronóstico de ventas, se realizó el taller SIPOC los días 22, 23 y 24 de marzo del 2022 en sesiones de dos horas diarias con la participación de las áreas de planeación de demanda, cadena de abastecimiento, finanzas, mercadeo, comercial, operaciones y talento humano. El resultado del taller fue el levantamiento del proceso macro de gestión de demanda dividido en sus tres fases: pronóstico inicial, pronóstico de consenso y planeación de la demanda.

Tres son los inputs para el pronóstico inicial de ventas del componente¹¹⁵ y componente¹⁴¹. El primero, el plan de negocio entregado por las áreas de mercadeo y ventas con las expectativas a doce meses. El segundo, es el listado de precios proyectados a nivel de SKU por área geográfica, tipo de cliente y convenio que entrega el área financiera. Por último, el área de planeación de demanda actualiza el histórico de ventas desde el ERP. Al no tener información de ventas perdidas y las entregas asociadas a ordenes distintas, se asume las ventas (pedido facturado) como proxy de la demanda.

Actualmente, con el histórico de ventas se calcula el promedio de la proyección soportada en promedio móvil de doce (12), seis (6), tres (3) meses, y se asume como pronóstico estadístico. Se procede a preparar la reunión de consenso con la evaluación de desempeño del mes anterior y año corrido (MAPE), el pronóstico inicial estadístico y las expectativas comerciales y sus

respectivas desviaciones versus presupuesto. Cada escenario incluye la estimación del ingreso operacional (precio x cantidad)

En la reunión de consenso se llega a un acuerdo sobre venta (demanda) final e ingresos proyectados globales. Se hace análisis específico a productos Pareto según, país y canal de comercialización. Para llevarlo a un nivel mayor de granularidad se adjudican las ventas a cada nivel proporcionalmente según comportamiento de los últimos tres meses. El rol del área de operaciones es lograr el balance entre el escenario de ventas y las capacidades productivas. El resultado es la proyección operacional y financieras a doce meses para el comité directivo.

De ser aprobado el plan, se inician los cargues de datos del plan de ventas (demanda) a SAP para obtener el plan de producción y de despachos. Las proyecciones de seis meses son utilizadas por el área de compras de materias primas para gestionar el abastecimiento. En caso de no aprobar las proyecciones el comité directivo se debe hacer un ejercicio de sensibilización de variables para lograr un escenario alineado a las expectativas organizacionales.

El comité de consenso lo conforman las áreas de Finanzas, Planeación (Planeación de demanda), Mercadeo, Comercial, Producción y Logística. La gerencia general, Finanzas, Planeación, Producción, Legal y Gestión Humana conforman el comité directivo. Se utiliza Excel para cálculos, Tableau con visualizador, SAP – DP para consolidar la información y Outlook como canal de comunicación. Un inhibidor detectado en el SIPOC fue la apreciación subjetiva del 71% de actividades manuales. El apéndice C muestra el flujo del proceso y el resumen del taller SIPOC concertados con la compañía.

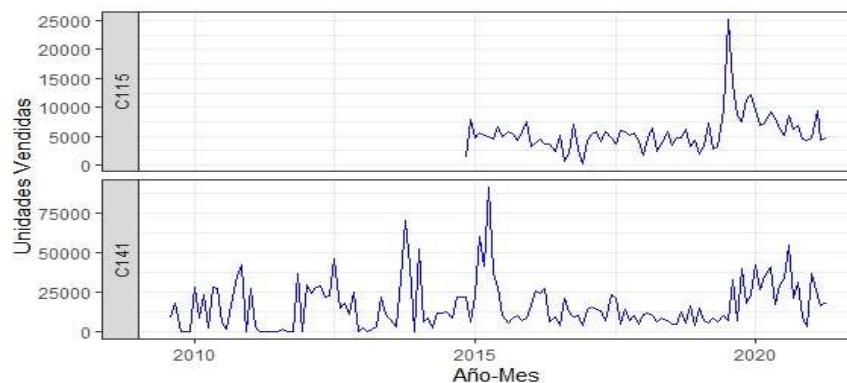
6.2 Fase 2 - Comprensión de los datos

La industria farmacéutica es una industria oligopolista en el que un número pequeño de empresas (pertenecientes a un grupo reducido de países) dominan casi totalidad de la producción, investigación y comercialización de los fármacos en el mundo. Entregando esto poder comercial

y de mercado, lo que les permite obtener beneficios mayores que los que obtendrían en una situación competitiva (Sosa, 2002). La empresa foco de este trabajo no es líder de mercado, es seguidora de precios y el Pareto de sus ingresos lo conforman por la comercialización fármacos genéricos y la línea cuidado y belleza.

Figura 6

Ventas históricas 2010-2020



El componente 115 (C115) es un producto de la línea de fármacos genéricos, cuenta con datos desde noviembre de 2014 hasta abril 2021, registra ventas mensuales de 5.545 unidades en promedio. Las ventas anuales han experimentado crecimiento, cerrando el 2020 en 82.538 unidades que equivale una tasa de crecimiento compuesto anual (CAGR) de 4.73% respecto a 2015. Sin embargo, respecto al 2019 se vio una caída del 2.3%. Precisamente el año 2019 registro el año de mayor venta con 107.054 unidades entregada al mercado con un segundo semestre con ventas promedio mayores a 13.000 unidades al mes (incluye el histórico mensual de 25.144 unidades en Julio). El área comercial explica el comportamiento de 2019 por el ingreso a un mercado en Centro América un competidor presentaba problemas de importación en el mismo mercado más el aumento de demanda de vitaminas y antigripales

Comercialmente el año 2020 representa la estabilidad de los mercados actuales para este componente. C115 es un producto maduro y su crecimiento depende del mercado inorgánico dado el poder de las firmas con que se compite en los actuales mercados actuales. Las variaciones

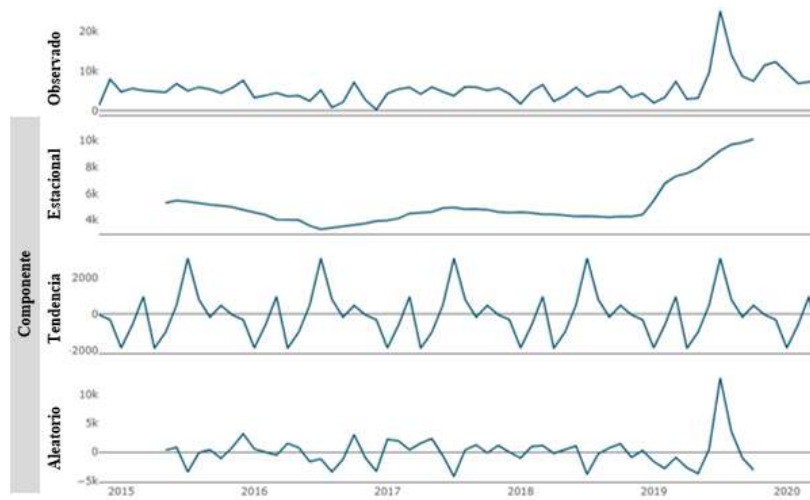
mensuales se dan por incremento de precio de las firmas líderes (no genéricas). Para el 2022 y 2023 no se espera el ingreso a una nueva región y se espera potenciar el producto a nivel de marca para sostener el terreno ganado.

El componente C141 (C141) de la línea de cuidado y belleza tiene datos disponibles desde agosto de 2010 hasta abril 2021, registra ventas mensuales de 16.400 unidades en promedio. Las ventas anuales han experimentado un crecimiento, cerrando el 2020 en 344.935 unidades; que equivale a una CAGR de 4.34% respecto a 2012 y del 9% respecto al año 2019. El 2020 mantuvo ventas mensuales promedio de 28.700 unidades al mes. El mes histórico es abril de 2015 con 91.332 unidades entregadas al mercado en un año que significativo ventas de 331.793 unidades.

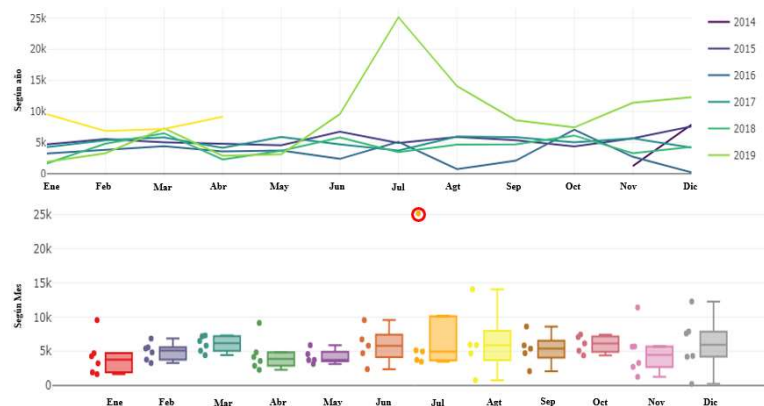
El área comercial define a C141 como maduro y su crecimiento depende de mercados inorgánicos dado que en los mercados actuales compite con firmas con poder de mercado. Se viene experimentando un crecimiento desde el año 2019 soportado en nivel de servicio y cambios en la política de despachos. El COVID no impacto el consumo de este producto, pero si el post COVID ha traído problemas a la cadena logística. Para el 2022 y 2023 no se espera el ingreso a una nueva región que implique crecimientos atípicos en las ventas de este componente.

6.3 Fase 3 – Preparación de los datos

El análisis exploratorio de la serie de ventas del C115, en la Figura 6, muestra que a nivel de tendencia antes de 2019 la serie era estable entorno a la media de 5.545 unidades y a partir de 2019 la serie experimenta tendencia de crecimiento lineal soportado por esfuerzos comerciales y efecto COVID. Aparece un posible factor estacional compuesto por la secuencia de un ciclo corto que inicia en enero, tiene su auge en febrero y finaliza en marzo; seguido de un ciclo largo que inicia expansión en abril para llegar al auge en junio/julio y finaliza en diciembre. Aunque en la parte final aparece una observación atípica (alejada de media) en el componente aleatorio, se puede decir que este es estacionario en media.

Figura 7*Descomposición de la serie histórica C115*

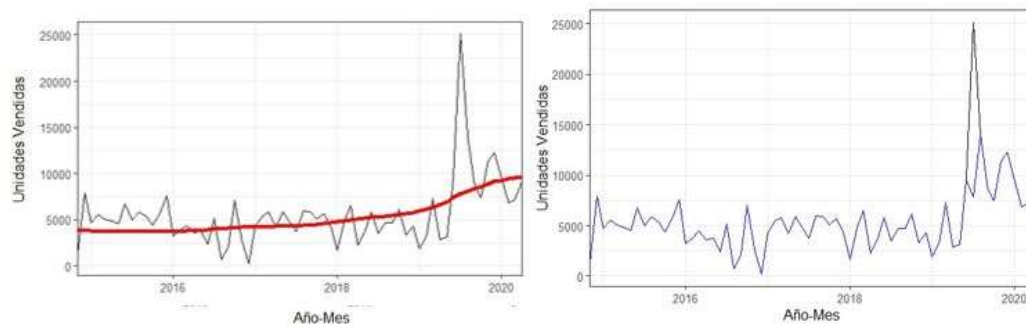
La Figura 7 (superior) presenta un análisis mes-año de las ventas del C115 evidenciando el incremento en ventas descritos anteriormente para los años 2019 y 2020. Se identifica adicionalmente que este incremento ocurre después de junio de 2019. La Figura 8 (inferior) expone el diagrama de cajas y alambres para cada mes durante los años 2014-2020 evidenciando el ciclo del primer trimestre descrito anteriormente. El ciclo abril y diciembre descrito de la Figura 6, resulta ser el efecto del incremento en ventas de los años 2019 y 2020 para estos meses.

Figura 8*Análisis de la estacionalidad de la serie histórica C115*

Se identifica, en la Figura 7, ventas atípicas en el mes de julio de 2019 explicadas como la materialización de esfuerzos comerciales y eventos del entorno. Por no ser representativo de la realidad, este dato es imputado por el valor en ese mes según tendencia (ver Figura 8).

Figura 9

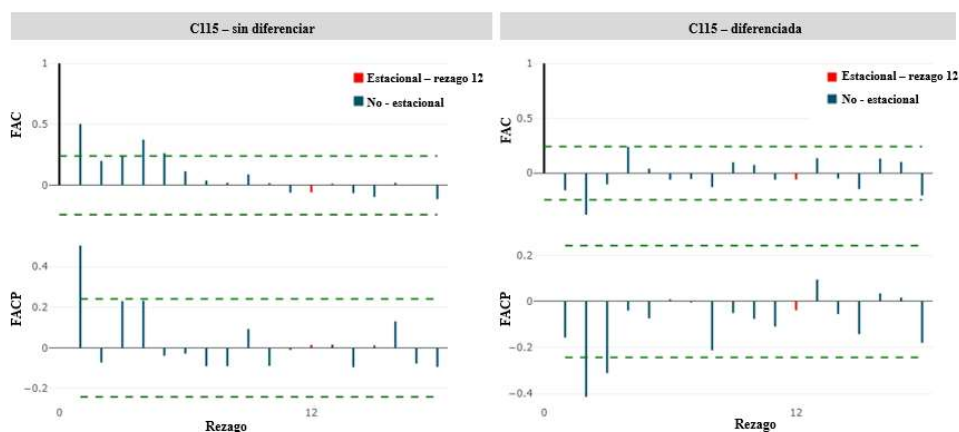
Serie imputada en julio de 2019



A partir de la Figura 9, se deduce que la serie C115 tiene relación con sus rezagos especialmente hasta el quinto. Su FAC y FACP sugieren un modelo AR(1). Sin embargo, la prueba de DFA de raíz unitaria ($p\text{-value} = 0.3967$) sugiere que la serie no es estacionaria y se debe diferenciar la serie C115. Al diferenciar se identifica un proceso IMA (1,1).

Figura 10

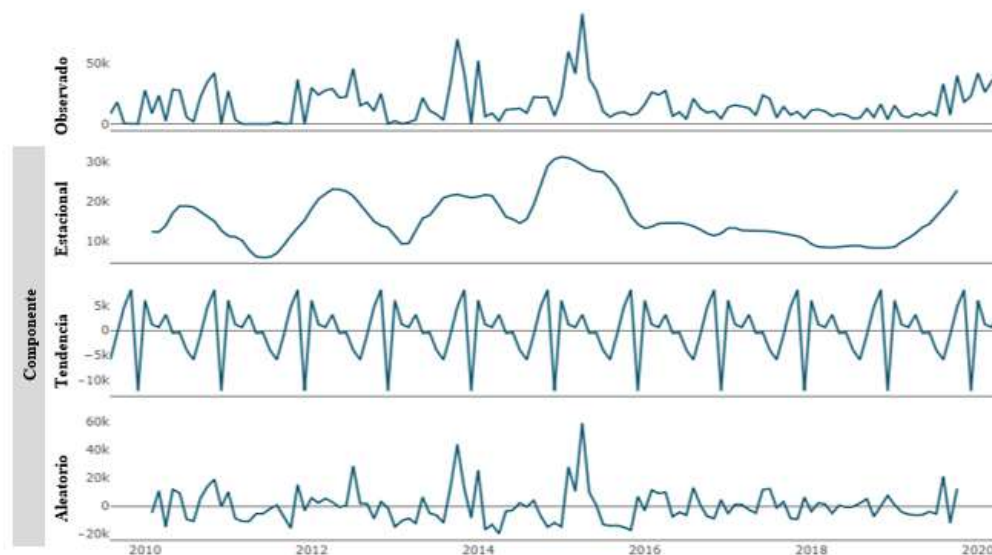
Análisis de los rezagos de la serie históricas C115



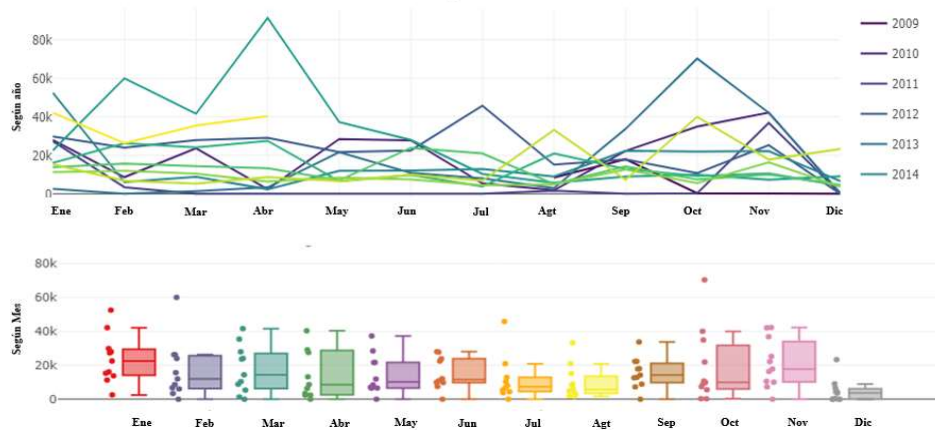
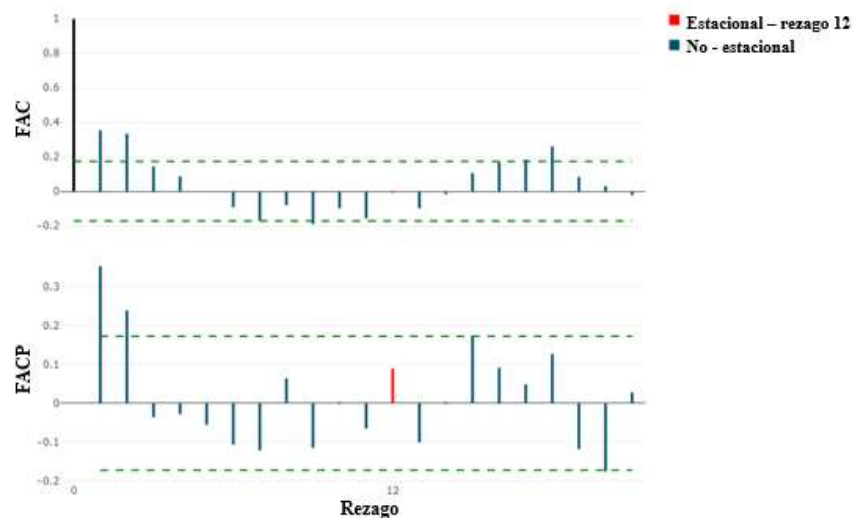
El análisis exploratorio de la serie de ventas C141 (Figura 10) muestra que a nivel de tendencia se observa que antes de 2016 presenta un comportamiento oscilatorio (ciclo de largo plazo) para después llegar un comportamiento estable hasta 2019, allí inicia una tendencia lineal creciente. Esto concuerda temporalmente con lo explicado por el área comercial de ingreso a nuevos mercados, maduración y cambios en políticas de despacho. Igualmente aparece lo que es un posible factor estacional iniciando en el mes de enero y finalizando en diciembre. Se identifican como auge el mes de febrero. Se observa al componente aleatorio con estructura estacionaria. Dos atípicos en los meses octubre de 2013 y abril de 2015 se detectan.

Figura 11

Descomposición de la serie histórica C141



La Figura 11 (superior) muestra que la serie anual de ventas del C141 tiene incremento en ventas descritos anteriormente para los años 2019 y 2020. Las ventas promedio mes del año 2012 fueron las más altas registradas con 22.475 unidades vendidas promedio mes. Los años 2013 y 2015 son representativas a nivel de unidades vendidas; el 2013 con un segundo semestre registro 26.000 unidades promedio mes y el 2015, ventas promedio mes en el primer semestre de 46.780 unidades. La Figura 11 (inferior) muestra posible comportamiento estacional, por disminución de ventas a mitad de año (junio a agosto) y diciembre.

Figura 12*Análisis de la estacionalidad de la serie histórica C141***Figura 13***Análisis de los rezagos de la serie histórica C141*

Se identifican que la serie de ventas del C141 presenta correlación con sus dos primeros rezagos (Figura 12). La prueba DFA ($p\text{-value} = 0.01$) rechaza la hipótesis nula de no estacionariedad, por tanto no requiere de diferenciación y teóricamente se identifica un modelo MA(2).

6.4 Fase 4 – Modelado

Como se enuncio en la Tabla 3 del capítulo 5, se dividen los datos en entrenamiento y prueba (los doce últimos datos). Inicialmente se realiza la estimación del modelo ARIMA (SARIMA) para las series de ventas C115 y C141. Se realizará primero un ajuste manual según el análisis descriptivo y posteriormente se ejecutara la función `auto.arima()` de R como mecanismo de validación de la estimación.

Alineado con el análisis descriptivo se propone para la serie C115 un modelo AR(1) y diferenciar la serie. El modelo AR(1) entrega un AIC de 1215, mientras la FAC de la serie diferenciada sugiere un modelo IMA(1,q). Se sensibiliza el parámetro q para generar los modelos IMA(1,1) e IMA(1,2). El resultado del AIC del modelo IMA(1,1) es 1189, menor que el IMA(1,2). Adicionalmente el parámetro θ_2 tiende a cero, en el modelo IMA(1,2), lo que no hace idóneo sugerir este modelo. La función `auto.arima()` propone que el mejor modelo para predecir la serie de ventas C115 es un ARIMA (0,1,1) sin sugerencia de un modelo estacional.

El análisis descriptivo de la serie de C141 sugiere no diferenciar; además, teóricamente sugiere un modelo MA(2) o un modelo estacional anual con un componente regresivo 1 o 2. El modelo MA(2) entrega un AIC de 2836. Atendiendo la sugerencia de un proceso estacional anual se sensibiliza un proceso SARIMA (0,0,2)(1,0,0)12 y SARIMA (0,0,2)(2,0,0)12; se identifica que los dos modelos generan un AIC de 2837 pero el modelo SARIMA (0,0,2)(1,0,0)12 tiene un BIC 2851, menor al SARIMA (0,0,2)(2,0,0)12. Una observación relevante es que en ambos casos el parámetro $\phi_i \cong 0$ para $i = 1$ y 2. La función `auto.arima()` propone que el mejor modelo para predecir la serie de ventas C141 es un ARIMA (2,0,0) sin sugerencia de un modelo estacional. La Tabla 4 presenta los resultados obtenidos para la serie C115 y C141.

Se propone entonces para la serie C115 el modelo ARIMA (0,1,1). Para la serie de ventas C141 se propone un ARIMA (2,0,0) y se tendrá en cuenta en los análisis el SARIMA (0,0,2)(1,0,0)12.

Tabla 4*Resumen de resultados de la estimación*

Serie	Modelo	(p, d, q)	Desviación estándar	AIC	BIC
C115	MA (1)	$\theta_1 = 0.438$	$\widehat{\sigma}_{\theta_1} = 0.089$	1218	1224
C115	IMA (1,1)	$\theta_1 = 0.696$	$\widehat{\sigma}_{\theta_1} = 0.089$	1189	1194
C115	IMA (1,2)	$\theta_1 = -0.686$ $\theta_2 = -0.009$	$\widehat{\sigma}_{\theta_1} = 0.1727$ $\widehat{\sigma}_{\theta_2} = 0.2006$	1191	1198
C115	AR (1)	$\phi_1 = 0.464$	$\widehat{\sigma}_{\phi_1} = 0.1123$	1215	1221
C115	ARI (1,1)	$\phi_1 = -0.39$	$\widehat{\sigma}_{\phi_1} = 0.09$	1203	1208
C141	MA (2)	$\theta_1 = 0.272$ $\theta_2 = 0.2914$	$\widehat{\sigma}_{\theta_1} = 0.085$ $\widehat{\sigma}_{\theta_2} = 0.08$	2836	2847
C141	SARIMA (0,0,2)(1,0,0)12	$\theta_1 = 0.292$ $\theta_2 = 0.304$ $\phi_1 = 0.099$	$\widehat{\sigma}_{\theta_1} = 0.087$ $\widehat{\sigma}_{\theta_2} = 0.081$ $\widehat{\sigma}_{\phi_1} = 0.093$	2837	2851
C141	AR (1)	$\phi_1 = 0.3585$	$\widehat{\sigma}_{\phi_1} = 0.083$	2840	2848
C141	AR (2)	$\phi_1 = 0.2726$ $\phi_2 = 0.2464$	$\widehat{\sigma}_{\phi_1} = 0.085$ $\widehat{\sigma}_{\phi_2} = 0.085$	2833	2845

Se establecen las ecuaciones de pronóstico para C115 ARIMA(0,1,1)

$$C115_t = C115_{t-1} + a_t - 0.696a_{t-1} \quad (63)$$

$$Si \ h = 1; \widehat{C115}_t(1) = 0.304C115_t + 0.696a_{t-1}(1) \quad (64)$$

$$Si \ h \geq 2; \widehat{C115}_t(h-1) \quad (65)$$

En el caso de C41 en el modelo AR(2)

$$C141_t = 0.272C141_{t-1} + 0.2464C141_{t-2} + a_t \quad (66)$$

$$Si \ h = 1; \widehat{C141}_t(1) = 0.272C141_t + 0.2464\widehat{C141}_{t-1} \quad (67)$$

$$Si \ h \geq 2; \widehat{C141}_t(h) = 0.272\widehat{C141}_t(h-1) + 0.2464\widehat{C141}_t(h-2) \quad (68)$$

Al realizar la verificación de supuestos sobre el error, para los dos modelos, se encuentra que no se encuentra evidencia de correlación entre los errores; así lo soportan la prueba de Durbin-Watson, el estadístico Q y Q'. A partir de la prueba de Shapiro-Wilk (>0.05) se asume que no hay evidencia para decir que los errores no siguen una distribución normal (ver Tabla 5). Se utiliza la gráfica de residuales versus el tiempo para validar la varianza constante de los residuales entornos a la media cero (Figura 12).

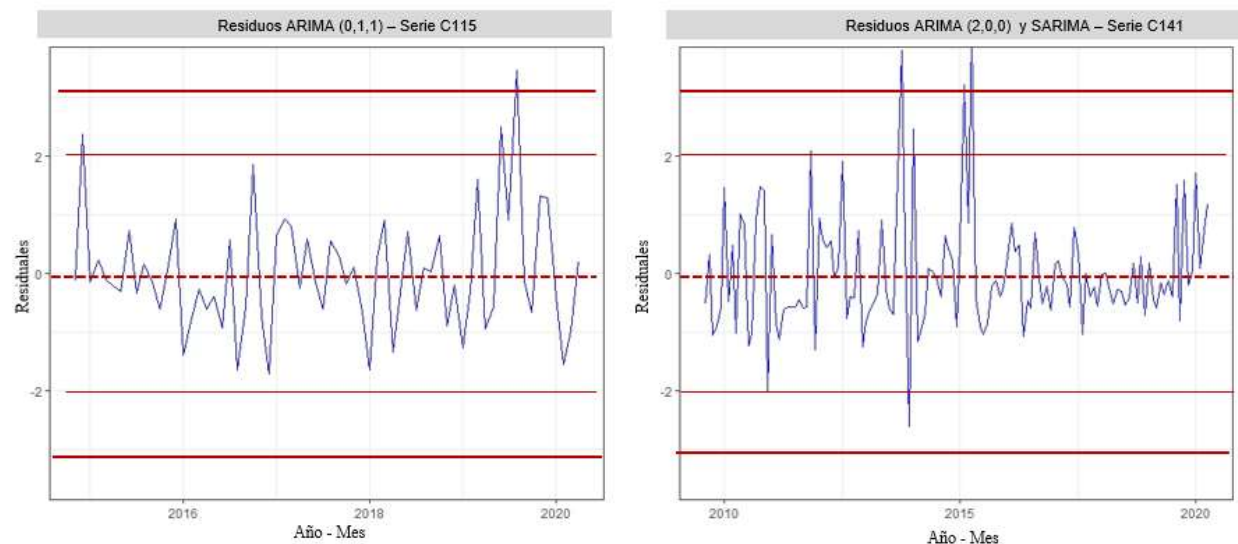
Tabla 5

Supuestos sobre los residuos

Serie	Shapiro Wilk	Q' de Ljung - Box	Q de Box - Pierce	Durbin Watson
C115-IMA(1,1)	0.065	0.846	0.849	1.92
C141-AR(2)	0.088	0.911	0.912	1.97
C141-SARIMA	0.07	0.769	0.767	1.93

Figura 14

Serie residuos Vs. el tiempo



Una vez identificado los modelos ARIMA, se procede a realizar la selección del modelo Prophet utilizando la misma serie de entrenamiento. El modelo del Prophet implementado en R

con la función `prophet_reg()` requiere la calibración de diversos hiperparámetros de entrada para optimizar el ajuste a la serie temporal y obtener unas predicciones con mayor precisión. Se debe calibrar el tipo de crecimiento lineal/logístico (`growth`), cantidad de cambios en tendencia de la serie de tiempo (`changepoint_num`), tiempo hasta ocurrencia de cambios de tendencia (`changepoint_range`), cambios estacionales según año, semana o día (`seasonality_yearly`), definición de estacionalidad como aditiva o multiplicativa (`season`), flexibilidad de puntos de cambio, estacionalidad o frecuencia de días feriados (`prior_scale`) y la de saturación en caso de crecimiento logístico (`logistic_cap`).

Apoyados en el conocimiento de la serie de ventas C115 y C141 para la calibración de parámetros vía simulación de puntos de cambio, tiempo hasta que ocurre un punto de cambio y se dejarán los demás parámetros en detección automática. El modelo seleccionado para cada serie será aquel con mejor error cuadrático medio (RMSE).

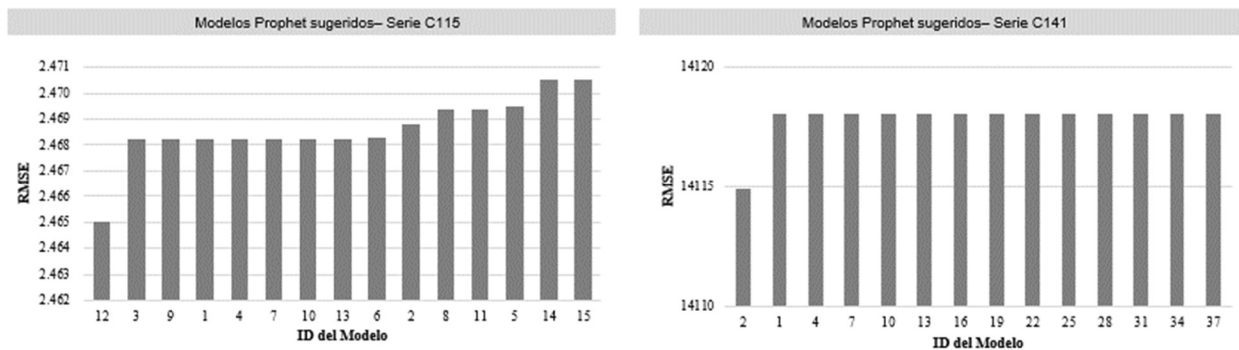
Para el modelo del C115 se asume un crecimiento lineal de la serie de ventas, de la Figura 6 se evidencia un punto de cambio de tendencia a partir de 2019 y no hay evidencia de factor estacionalidad, según los análisis descriptivos y lo sugerido por el modelo ARIMA. Se configuró el set de hiperparámetros con un `changepoint_num` con la posibilidad de detectar de 2 hasta 10 posibles puntos de cambio de tendencia (la Figura 10 sugiere hasta 5), el `changepoint_range` variara de 0 a 1 espaciado por 0.2 (cubriendo el total de rango permitido de este parámetro) dado que no se tiene claridad de la distancia temporal en que suceden los cambios de tendencia, se dejó libre la posibilidad de detección de estacionalidad y el resto de parámetros estarán bajo selección automática.

De los modelos sugeridos para la serie C115, el ID = 12 presenta el mejor desempeño a partir de las métricas evaluadas. No solo tienen el menor RMSE = 2.465 sino también el menor MAE, MAPE y MASE. Sin embargo, se puede decir que los modelos sugeridos tienen desempeños muy homogéneos (figura 13 - izquierda). Los parámetros del modelo seleccionado (ID = 12) son:

crecimiento lineal, ocho puntos de cambio, no puede existir un punto de cambio antes del primer 20% de los datos y se describe la estacionalidad como aditiva.

Para el modelo C141 se asume un crecimiento lineal de la serie de ventas y según la Figura 10, existen varios puntos de cambios de tendencia (se podrían asumir como ciclos de largo plazo). Los meses de julio, agosto y diciembre las ventas parecen caer y esto no permite configurar una estacionalidad anual o estrictamente semestral. En este punto es importante traer que los la metodología ARIMA sugieren un AR(2) para esta serie y deja la posibilidad de un SARIMA(0,0,1)(1,0,0)¹². Se configuró el set de parámetros con un changepoint_num con la posibilidad de detectar de 2 hasta 28 posibles puntos de cambio de tendencia (la Figura 10 sugiere hasta 14), el changepoint_range variara de 0 a 1 espaciado por 0.2 (cubriendo el total de rango permitido de este parámetro) dado que no se tiene claridad de la distancia temporal en que suceden los cambios de tendencia, se dejó libre la posibilidad de detección de estacionalidad y el resto de parámetros estarán bajo selección automática.

De los modelos sugeridos para la serie C141, el que presenta el mejor desempeño a partir de las métricas evaluadas tiene un RMSE = 17.063 (ID = 2). También tiene menor MAE, MAPE, SMAPE y MASE. Sin embargo, el total de modelos corridos en la calibración de parámetros, tienen desempeño homogéneo. El modelo seleccionado (ID = 2) presenta crecimiento lineal, detecta dos puntos de cambio, no puede existir un punto de cambio antes de 12 meses y se detecta estacionalidad aditiva.

Figura 15*Selección del mejor modelo Prophet*

Una vez identificado los modelos ARIMA y PROPHET, se procede a estimar el modelo XGBoost utilizando las mismas series de entrenamiento. Inicialmente se crea un set de variables a partir de la misma serie asociadas a la temporalidad (ver Tabla 6).

Tabla 6*Creación de variables Componente115y Componente141*

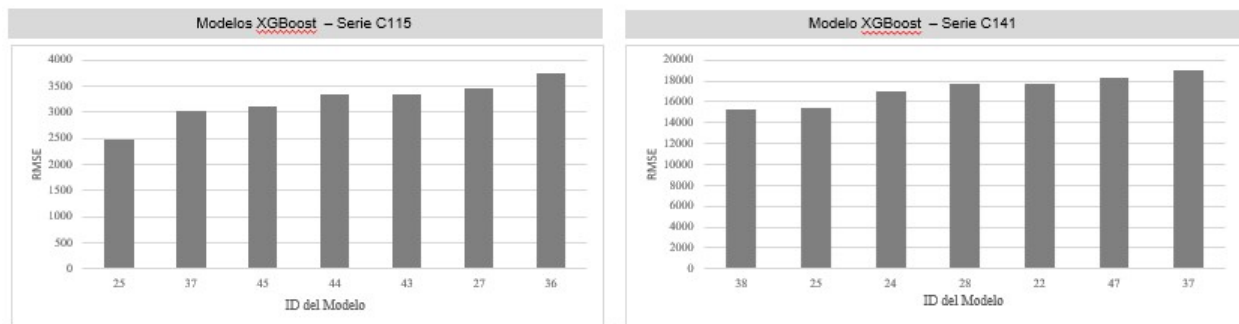
Variable	Valores	Variable	Valores
Semestre	1 y 2	Semana del mes	1, 2, ...,4
Trimestre	1, 2, 3 y 4	Mes (indicadora)	1 si es mes. 0 caso contrario
Mes	1, 2, ...,12	Día (indicadora)	1 si es día. 0 caso contrario
Día de la semana	1, 2, ...,7	Ventas t-6	≥ 0
Día del trimestre	1, 2, ...,63	Ventas t-12	≥ 0
Día del año	1, 2, ...,365	Ventas t-24	≥ 0
Semana del año	1, 2, ...,48		

El modelo del XGBoost implementado en R en el paquete {parsnip} la función `boost_tree()` requiere la calibración de diversos parámetros de entrada para optimizar el ajuste a la serie temporal y obtener unas predicciones con mayor precisión. Se debe calibrar la tasa de aprendizaje, el número de árboles, tamaño del nodo, profundidad del árbol y mínima pérdida entre otros. Se realiza ajuste de parámetros a partir de la sensibilización de cada uno de ellos.

Para las dos series se configura una grilla con valores para la tasa de aprendizaje (0.001, 0.01, 0.1, 0.2, 0.4, 0.6, 0.8, 1.0), número de árboles (5, 10, 50), tamaño mínimo del nodo (4, 10) y selección aleatoria de predictores (10); el resto de los parámetros estarán bajo selección automática. El modelo seleccionado será aquel con mejor error cuadrático medio (RMSE).

Figura 16

Modelos de pronóstico XGBoost



Se identifica que, de los modelos sugeridos, que el modelo ID = 25 y el ID = 38 presenta el mejor desempeño a partir de las métricas evaluadas para C115 y C141 respectivamente. En el C115 este modelo tiene el menor RMSE = 2463 y también el menor MAE, MAPE, SMAPE y MASE. Algo similar con el C141 con RMSE = 15282 y menores métricas MAE, MAPE, SMAPE y MASE.

6.5 Fase 5 – Evaluación

Para lograr una línea base de comparación, el equipo de planeación de demanda compartió las medidas de desempeño promedio históricas desde el 2019 y las del pronóstico a 12 meses (datos de prueba) de la serie de datos analizada. El pronóstico de la serie de ventas C115 presenta mejora con los modelos ARIMA, XGBoost y Prophet respecto al desempeño histórico. En términos comparativos del MAPE mejoró un 15% con el ARIMA, 38% con XGBoost y 40% con Prophet. Al comparar respecto al promedio móvil de los datos de prueba, en términos de MAPE, el modelo ARIMA presenta un desempeño inferior respecto al promedio móvil (-11%), mientras los

algoritmos Prophet (21%) y XGBoost (19%) logran un mejor desempeño. El mejor desempeño para la serie C115, según MAPE, es el modelo Prophet. Sin embargo, el MASE sugiere al XGBoost. Ambos, Prophet y XGBoost, con desempeños muy similares.

Tabla 7

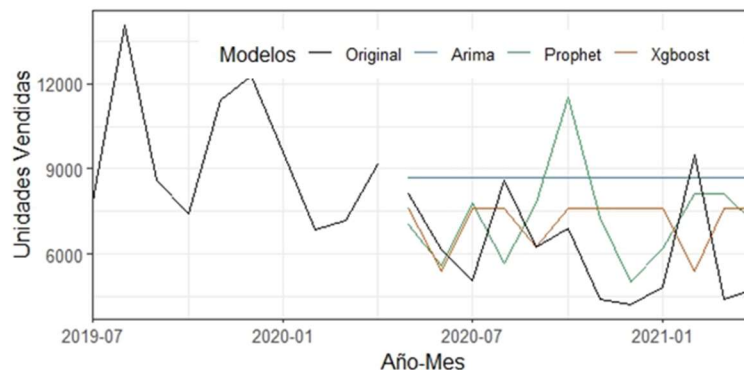
Componente115 – Medidas de desempeño

Modelo	RMSE	MAE	MAPE %	MASE	SMAPE %
Promedio Móvil Histórico	3684	3356	64.3	1.54	48.1
Promedio Móvil	3003	2699	49	1.21	38
ARIMA (0,1,1)	3106	2710	54.5	1.30	39.4
Prophet	2465	2155	38.4	1.03	32
XGBoost	2463	2081	39.6	0.99	32.5

Seleccionar el modelo Prophet es optar por un modelo que responde a las variaciones de la serie de ventas, mientras que el XGBoost responde a “ciclos” de caída de las ventas y posterior estabilización. El modelo ARIMA se proyecta como línea recta en el tiempo; algo de esperar según la Ecuación 65. En el caso de planeación de demanda, es importante un modelo que responda a los movimientos del mercado y así, poder estimar los esfuerzos requeridos por la cadena de suministro, por ende, se sugiere al Prophet como modelo de pronóstico de demanda.

Figura 17

Pronóstico Componente115



En cuanto a la serie de ventas del C141 es evidente la mejora de los modelos estimados respecto al desempeño histórico, en términos comparativos el MAPE mejoró 3.3% con el ARIMA, 1.8% al SARIMA, 29% con Prophet y 46% con XGBoost. Al comparar los modelos estimados respecto al promedio móvil estimado con los datos de prueba, en términos de MAPE, todos los modelos presentan mejor rendimiento. El modelo ARIMA (2,0,0) presentan mejor desempeño en todas las métricas (excepto el MAPE) respecto a los otros modelos. Esto se da porque es una línea en el tiempo que promedia las variaciones de la serie.

Tabla 8

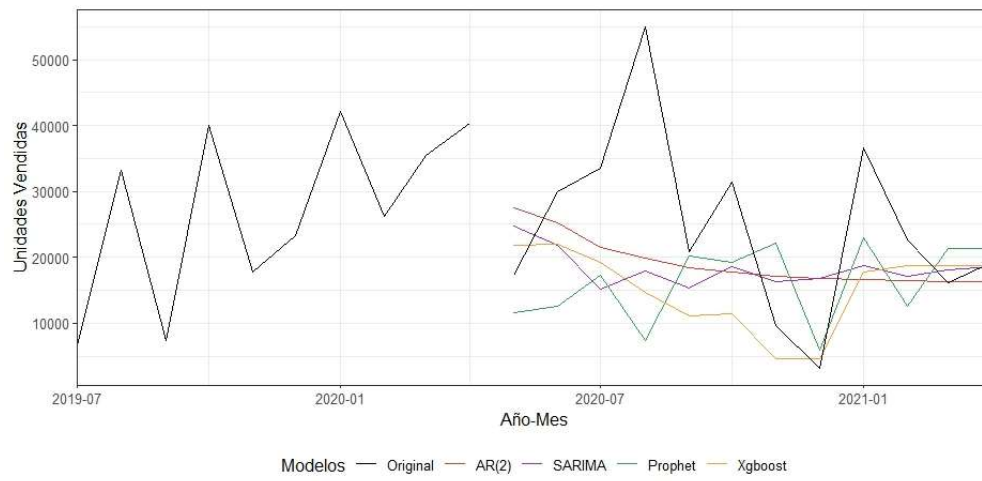
Componente 141 – Medidas de desempeño

Modelos	RMSE	MAE	MAPE %	MASE	SMAPE %
Promedio Móvil Histórico	18521	14376	72	0.934	58.17
Promedio Móvil	14310	1119	109	0.734	57
ARIMA (2,0,0)	14095	10698	69.6	0.704	48.7
SARIMA (0,0,1)(1,0,0)12	14756	11289	70.7	0.742	51.7
Prophet	17063	12186	50.9	0.801	55.9
XGBoost	15280	10730	38.5	0.706	49.1

Seleccionar el modelo XGBoost es optar por un modelo que responde a las variaciones de la serie de ventas y asertivo. Algo que intenta hacer el modelo Prophet, pero parece que los incrementos en ventas le restan capacidad de pronóstico. El modelo ARIMA proyecta la tendencia de la serie mitigando la varianza, Mientras el SARIMA tiene un comportamiento cíclico que marca la tendencia de los cambios de la serie, pero estable en media. Aquí se puede sugerir el uso del modelo XGBoost para el corto plazo (hasta 3 meses) dado su foco en la planeación de capacidades productivas y el ARIMA o SARIMA para la planeación de abastecimiento y proyecciones financieras dado su foco en mitigar alteraciones del mercado.

Figura 18

Pronóstico Componente141



7 Conclusiones

Del análisis de resultados, siguiendo la metodología CRISP-DM, se puede concluir utilizar modelos ARIMA y los algoritmos de aprendizaje automático XGBoost y Prophet para la proyección de ventas a 12 meses de los componente115 y componente141, mejora las métricas de desempeño respecto a la actual practica de pronóstico de la compañía.

En el caso del componente115, el algoritmo Prophet alcanza una mejora del 40% respecto al MAPE histórico y del 21% respecto al último ejercicio de pronóstico realizado. El algoritmo XGBoost y el Modelo ARIMA (0,1,1) mejoran el MAPE histórico en 16% y 9.8% respectivamente. Respecto al último ejercicio de pronóstico, XGBoost logró mejorar en 2% el MAPE, mientras el modelo ARIMA presentó incremento del +5.5%.

Para el componente141, XGBoost alcanza una mejora del 46% respecto al MAPE histórico y del 64% respecto al último ejercicio de pronóstico realizado. En el caso del algoritmo Prophet logra una mejora respecto al MAPE histórico de 29% y test de 53%. Los modelos AR (2) y SARIMA mejoran los ejercicios histórico y test de pronóstico realizado por la compañía con mejoras superiores al hasta del 3% y del 35% respectivamente.

Es importante resaltar que no hay un modelo único para las dos series, validando lo descrito en los antecedentes. Esto es, realizar análisis por cada caso soportado por el entendimiento del negocio y el análisis exploratorio. Adicionalmente se podría, como ejercicio teórico – practico, explorar los algoritmos híbridos, las ANN's y el algoritmo LightGBM. Así mismo evaluar la idoneidad de modelos ARIMAX y buscar incorporar variables del entorno través de modelos adecuados para este tipo de ejercicios.

Los resultados obtenidos abren la puerta a escalar el proyecto a otros productos. Se sugiere incluir la selección automática de un modelo y apalancar en impacto financiero para el negocio.

8 Bibliografía

Adhikari, R., & Agrawal, R. (2013). *An Introductory Study on Time series Modeling and Forecasting*. New Delhi.

Alón, I., Qi, M., & Sadiwski, R. (2001). Forecasting aggregate retail sales: A comparison of artificial neural networks and traditional methods. *Journal of Retailing and Consumer Services*, 147-156.

Amalnick, M., Habibifar, N., & Bastan, M. (2019). An intelligent algorithm for final product demand forecasting in pharmaceutical units. *International Journal of Systems Assurance Engineering and Management*, 481-493.

Anshuman, G., Costas, M., & Conor, M. (2000). Mid-term supply chain planning under demand uncertainty: customer demand satisfaction and inventory management. *Computers and Chemical Engineering*, 2613–2621.

Apichatibutarapong, S. (Febrero de 2015). Business Forecasting Technique on Mobile Devices by Using R-Programming. *Procedia - Social and Behavioral Sciences*, 1377–1382.

Armstrong, j. s. (1985). *Long-range Forecasting: From Crystal Ball to Computer*. Estados Unidos: John Wiley & Sons.

Armstrong, S. (2001). *Principles of forecasting: a handbook for researchers and practitioners. Forecasting Audit checklist*. Filadelfia: University of Pennsylvania.

Aslanargun, A., Mammadov, M., Yazici, B., & Yolacan, S. (2007). Comparison of ARIMA, neural networks and hybrid models in time series: Tourist arrival forecasting. *Journal of Statistical Computation and Simulation* , 29-53.

Aytac, E. (2021). Forecasting Turkey's Hazelnut Export Quantities with Facebook's Prophet Algorithm and Box-Cox Transformations. *Advance in Distributed Computing and Artificial Intelligence Journal*, 33-47.

Blackburn R, L. K. (2014). A predictive analytics approach for demand forecasting in the process industry. *International Transaction in Operation Research*.

Box, G., Gwlym, J., & Gregory, R. (1994). *Time Series Analysis: Forecasting and Control* (3ra ed.). New Jersey: Prentice Hall.

- Breiman, L. (2001). *Random Forest*. Berkeley: University of California.
- Breiman, L., Friedman, J., Olshen, R., & Stone, C. (1984). *Classification And Regression Trees*. New York: Routledge.
- Candan, G., Taşkin, M., & Yazgan, H. (2014). Demand forecasting in pharmaceutical industry using neuro-fuzzy approach. *Journal of Military and Information Science*, 41-49.
- Carbonneau, R., Laframboise, K., & Vahidov, R. (2008). Application of machine learning techniques for supply chain demand forecasting. *European Journal of Operational Research*, 1140-1154.
- Chakraborty, K., Mehorotra, K., Mohan, C., & Ranka, S. (1990). Forecasting the Behavior of Multivariate Time Series using Neural. *Electrical Engineering and Computer Science - technical Reports*, 1-26.
- Chen, T., & Guestrin, C. (2016). XGBoost: A Scalable Tree Boosting System. *International Conference of Knowledge and Data Mining*, 785-794.
- Christopher, M., & Holweg, M. (2011). Supply Chain 2.0: managing supply chains in the era of turbulence. *International Journal of Physical Distribution & Logistics Management*, 63-82.
- Diaz-Robles, L. A., JOrtega-Bravo, J., & Fu, J. (2008). A Hybrid ARIMA and Artificial Neural Networks Model to Forecast Particulate Matter in Urban Areas: The Case of Temuco, Chile. *Atmospheric Environment*, 8331-8340.
- Doganis, P., Alexandridis, A., Patrino, P., & Sarimveis, S. (2006). Time series sales forecasting for short shelf-life food products based on artificial neural networks and evolutionary computing. *Journal of Food Engineering*, 196-204.
- Doganis, P., Alexandridis, A., Patrinos, P., & Sarimveis, H. (2006). Time series sales forecasting for short shelf-life food products based on artificial neural networks and evolutionary computing. *Journal of Food engineering*, 75(2), 196-204.
- Eksoz, C., Mansoury, A., & Bourkakis, M. (2014). Collaborative forecasting in the food supply chain: A conceptual framework. *International Journal Producción Economics*, 120-135.
- Espinosa, J. (2020). Application of Random Forest and XGBoost algorithms based on a credit card application database. *Ingeniería e Investigación Tecnológica*, 3.

- Friedman, J. (2000). Greedy Function Approximation: A Gradient Boosting Machine. *The Annals of Statistics*, 1-39.
- Goh, C., & Law, R. (2002). Modeling and forecasting tourism demand for arrivals with stochastic nonstationary seasonals and trends by exponentially weighted moving averages. *Tourism Management*, 499-510.
- Guerrero, V. (2003). *Análisis estadístico de series de tiempo económicas*. Mexico D.F.: Thomson.
- Hamzaçebi, C. (2008). Improving artificial neural networks performance in seasonal time series forecasting. *Information Sciences*, 4550-4559.
- Hastie, T., Tibshirani, R., & Jerome, F. (2008). *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Stanford: Springer.
- Hernandez, J. (2019). *Análisis de procesos con SIPOC*. <https://agileexperience.es/2019/12/30/analisis-de-procesos-con-sipoc/>.
- Hyndman, R. (2020). A brief history of forecasting competitions. *International Journal of Forecasting*, 7-14.
- Hyndman, R., & Khandakar, Y. (2008). Automatic time series forecasting: The forecast package for R. *Journal of Statistical Software*, 1-10.
- Hyndman, R., & Koehler, A. (2006). Another look at measures of forecast accuracy. *International Journal of Forecasting*, 679-688.
- Jain, A., & Kummar, A. (2007). Hybrid Neural Network Models for Hydrologic Time Series Forecasting. *Applied Soft Computing*, 585-592.
- Jain, C. L. (2006). Benchmarking forecasting practices in America. *The Journal of Business Forecasting*, 28-30.
- Javier, E. (2020). Application of Random Forest and XGBoost algorithms based on a credit card application database. *Ingeniería e Investigación Tecnológica*, 1-16.
- Kuo, R., Wu, P., & Wang, C. (2002). An intelligent sales forecasting system through integration of artificial neural network and fuzzy neural network. *Neural Networks*, 909-925.

- Lawrence, M., Goodwin, P., O'Connor, M., & Önkal, D. (2006). Judgmental forecasting: A review of progress over the last 25 years. *International Journal of Forecasting*, 22(3), 493-518.
- Lunenburg, F. (2012). Human Resource Planning: Forecasting Demand and Supply. *International journal of management, business and administration*, 15(2), 1-10.
- Makridakis, S., Andersen, A. ., Carbone, R., Fildes, R., Hibon, M., Lewandowski, R., . . . Winkler, R. (1982). The accuracy of extrapolation (time series) methods: Results of a forecasting competition. *Jorunal of Forecasting*, 111-153.
- Makridakis, S., Chatfield, C., Hibon, M., Lawrence, M., Mills, T., Ord, K., & Simmons, L. F. (1993). La competencia M2: un estudio de previsión basado en juicios en tiempo real. *Revista Internacional de pronósticos*, 5-22.
- Makridakis, S., Spiliotis, E., & Assimakopoulos, V. (2020). The M4 Competition: 100,000 time series and 61 forecasting Methods. *international journal of forecasting* , 54-74.
- Makridakis, S., Spiliotis, E., & Assimakopoulos, V. (2020). The M5 Competition: Results, findings and conclusions. *international journal of forecasting*, 54-74.
- Mason, L., Baxter, J., Bartlett, P., & Frean, M. (1999). Boosting Algorithms as Gradient Descent in Function Space. *Advances in Neural Information Processing Systems*, 1-29.
- Medina, F., & Ñique, C. (2017). Bosques aleatorios como extensión de los árboles de clasificación con los programas R y Python. *INTERFASES*, 165-189.
- Mendoza, R. (2013). *Boosting en el modelo de parentizaje PAC*. bogotá: Institución Universitaria Politécnico Grancolombiano.
- Mentzer, J., & Cox, J. (1984). Familiarity, Application, and Performance of Sales Forecasting Techniques. *Journal of Forecasting*, 27-36.
- Mentzer, J., & Kahn, K. (1995). Forecasting Technique Familiarity, Satisfaction, Usage, and Application. *Journal of Forecasting*, 465 - 476.
- Mentzer, J., Bienstock, C., & Kahn, K. (1999). Benchmarking Sales Forecasting Management, *Business Horizons*. 48-56.

- Merkuryeva, G., Valberga, A., & Smirnov, A. (2019). Demand forecasting in pharmaceutical supply chains: A case study. *Procedia Computer Science*, 3-10.
- Moon, M. (2013). *Demand and Supply Integration: The Key to World-class Demand Forecasting*. Estados Unidos: Pearson Education.
- Murphy, A. (1993). What is a good forecast? An essay on the nature of goodness in weather forecasting. *American Meteorological Society*, 281-293.
- Nowadly, K., & Jung, S. (2020). Using Machine Learning Approaches to Improve Long-Range Demand Forecasting. *MIT Libraries*, 2-71.
- Papacharalampous, G., & Tyrallis, H. (2020). Hydrological time series forecasting using simple combinations: Big data testin and investivations on one-year ahead river flow preictability. *Journal of Hydrology*, 590.
- Park, J., Chang, B., & Mok, N. (2019). Time Series Analysis and Forecasting Daily Emergency Department Visits Utilizing Facebook's Prophet Method. *Annals of emergency Medicine*, S57.
- Peña, D. (2005). *Análisis de series temporales*. Madrid: Alianza Editora.
- Ramos, P., Nicolau, S., & Rui, R. (2015). Performance of state space and ARIMA models for consumer retail sales forecasting. *Robotic and Computer Integrated Manufacturing*, 151-163.
- RStudio Team. (2002). *RStudio: Integrated Development Environment for R*. Boston: PBC.
- Satrio, C., Darmanwan, W., Unrica, B., & Hanafiah, N. (2021). Time series analysis and forecasting of coronavirus disease in Indonesia using ARIMA model and PROPHET. *Procedia Computer Science*, 524-532.
- Shalev-Shwartz, S., Ben-David, & Shai. (2014). *Understanding Machine Learning: From Theory to Algorithms*. New York: Cambridge University Press.
- Subashini, A., Sandhiya, K., Saranya, S., & Harsha, U. (2019). Forecasting Website Traffic Using. *International Research Journal of Multidisciplinary Technovation*, 56-63.
- Tandon, C., Revankar, S., Parihar, H., & Singh, S. (2021). How can we predict the impact of the social media messages on the value of cryptocurrency? Insights from big data analytics. *International Journal of Information Management Data Insights*, 1-8.

- Tang, Z., Almeida, c., & Fishwick, P. (1991). Time series forecasting using neural networks vs. Box- Jenkins methodology. *Simulation*, 303-310.
- Taylor, S., & Letham, B. (2017). Forecasting at Scale. *PeerJ*, 1-25.
- Team, R. (2002). *RStudio: Integrated Development Environment for R*. Boston: PBC.
- Van der Vorst, J., & Beulens, A. (2002). Identifying sources of uncertainty to generate supply chain redesign. *International Journal of Physical Distribution & Logistics Management*, 409-430.
- Vidal Holguin, C. J. (2010). *Fundamentos de control y gestión de inventarios*. Santiago de Cali: Universidad del Valle.
- Wang, L., Zou, H., Su, J., & Li, L. (2013). An ARIMA-ANN hybrid model for time series forecasting. *Systems Research and Behavioral Science*, 244-259.
- Williams, B., & Hoel, L. (2003). Modeling and forecasting Vehicular traffic flow as a seasonal ARIMA process: Theoretical basis and empirical results . *Journal of Transportation Engineering*, 664-672.
- Yang, H., Li, X., Qiang, W., Zhao, Y., Zhang, W., & Tang, C. (2021). A network traffic forecasting method based on SA optimized ARIMA–BP neural network. *Computer Networks*, 193.
- Zadeh, N., Sepehri, M., & Farvaresh, H. (2014). Intelligent Sales Prediction for Pharmaceutical Distribution Companies: A Data Mining Based Approach. *Mathematical Problems in Engineering*, 1-15.
- Zhang, P., & Kline, D. (2007). Quarterly Time-Series Forecasting With Neural Networks. *IEEE Transactions on Neural Networks*, 1800-1814.
- Zhao, N., Liu, Y., Vanos, J., & Cao, G. (2018). Day-of-week and seasonal patterns of PM2.5 concentrations over the US: Times series analyses using the Prophet procedure. *Atmospheric Environment*, 116-127.

Apéndice A. Aspectos de calidad de un pronóstico

Tabla A 1

Resumen aspecto de calidad de los pronósticos

Aspecto	Definición
Sesgo	Correspondencia entre la media del pronóstico y la media de las observaciones
Asociación	Nivel de relación lineal entre los pronósticos y las observaciones
Precisión	Correspondencia media entre pares individuales de pronósticos y observaciones
Capacidad	Precisión del pronóstico de interés en relación con la exactitud de los pronósticos producidos por la norma de referencia
Confiabilidad	Correspondencia entre el promedio condicional de las observaciones y los pronósticos condicionados, como promedio de todos los pronósticos
Resolución	Diferencia entre el promedio condicional de las observaciones y el promedio no condicional de las observaciones, como promedio de todos los pronósticos
Nitidez	Variabilidad del pronóstico como se describe en la distribución del pronóstico
Discriminación	Correspondencia entre el pronóstico promedio condicional y la observación condicionada, sobre el promedio de todas las observaciones
Incertidumbre	Variabilidad de las observaciones como se describe en la distribución de las observaciones

Nota. Adaptado de MURPHY, Allan. What is a good forecast? An essay on the nature of goodness in weather forecasting. American Meteorological Society. June, 1993, vol 8. p. 284.

Apéndice B. Modelos de desarrollo aditivo

Las funciones base $f(x) = \sum_{m=1}^M \beta_m b(x, \gamma_m)$, con $\beta_m, m = 1, 2, \dots, M$ como coeficientes de desarrollo y $b(x, \gamma_m)$ funciones $\in R$ y que dependen de x y parámetros γ , se optimizan a partir de la pérdida promediada (error cuadrático medio) o pérdida basada en la verosimilitud.

$$\min_{\{\beta_m, \gamma_m\}_1^M} \sum_{i=1}^N L \left(y_i, \sum_{m=1}^M \beta_m b(x, \gamma_m) \right) \quad (69)$$

Mediante el modelo aditivo por pasos hacia adelante se puede resolver añadiendo secuencialmente nuevas funciones base a la expansión sin ajustar los parámetros y los coeficientes de las que ya se han agregado. Primero se establece $f_0(x) = 0$, después desde $m = 1$ Hasta $m = M$ se calcula $\{\beta_m, \gamma_m\} = \min_{\{\beta, \gamma\}} \sum_{i=1}^N L(y_i, f_{m-1}(x_i) + \beta b(x_i, \gamma))$ y se establece $f_m(x) = f_{m-1}(x) + \beta_m b(x, \gamma_m)$. Si la función de pérdida es $L(y, f(x)) = (y - f(x))^2$ (error cuadrático medio), entonces

$$L(y, f(x)) = (y - f_{m-1}(x) - \beta_m b(x, \gamma_m))^2 = (r_i - \beta_m b(x, \gamma_m))^2 \quad (70)$$

Con $r_i = y - f_{m-1}(x)$ como el residuo i -ésimo. El término $\beta_m b(x, \gamma_m)$ que mejor se adapta a los residuos en cada etapa. Si la función de pérdida es $(y, f(x)) = e^{-yf(x)}$, entonces

$$\min_{\beta, \gamma} \sum_{i=1}^N e^{-y_i(f_{m-1}(x_i) + \beta b(x_i, \gamma))} = \min_{\beta, \gamma} \sum_{i=1}^N w_i^m e^{-y_i \beta b(x_i, \gamma)} \quad (71)$$

Con $w_i^m = e^{-y_i f_{m-1}(x_i)}$. Para hallar solución al parámetro β :

$$\sum_{i=1}^N w_i^m e^{-y_i \beta G(x_i)} = e^{-\beta} \sum_{i=1}^N w_i^m + e^{\beta} \sum_{i=1}^N w_i^m \quad (72)$$

$$= (e^{\beta} - e^{-\beta}) \sum_{i=1}^N w_i^m I(y_i \neq G(x_i)) + e^{-\beta} \sum_{i=1}^N w_i^m = e^{\beta} err_m + e^{-\beta} (1 - err_m) \quad (73)$$

El momento donde β se hace mínimo $e^{\beta} err_m - e^{-\beta} (1 - err_m) = 0$, es dado por

$$\beta = \frac{1}{2} \log \left(\frac{1 - err_m}{err_m} \right) \quad (74)$$

Entonces se actualiza la ecuación $f_m(x) = f_{m-1}(x) + \beta_m G_m(x)$. Los pesos dados para la iteración siguiente vienen dados por

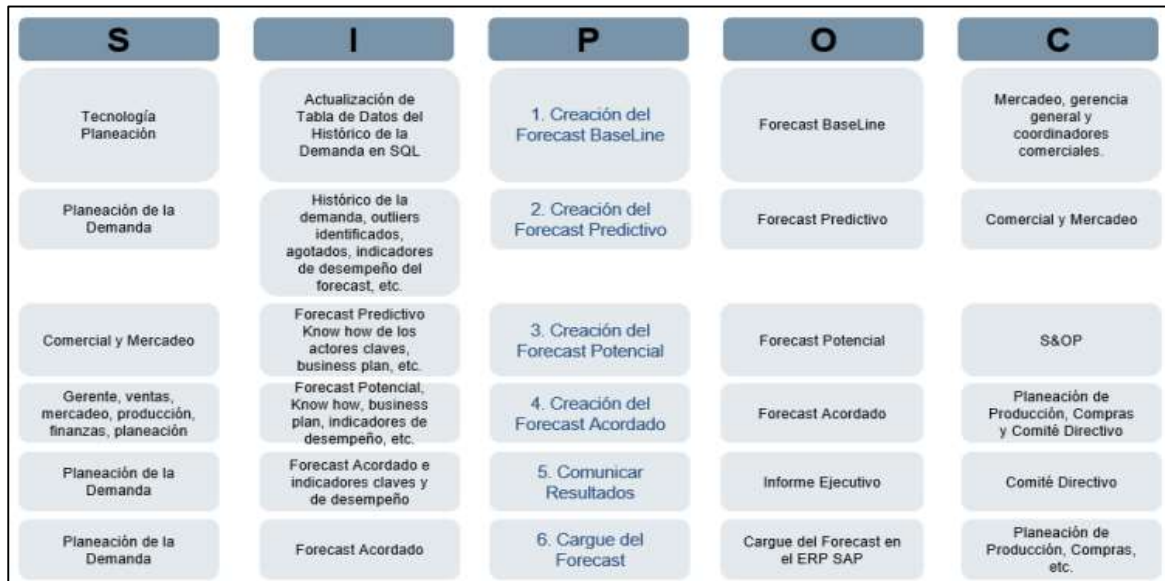
$$w_i^{m+1} = e^{-y_i f_m(x_i)} = w_i^m * e^{-y_i \beta_m G_m(x_i)} \quad (75)$$

Teniendo en cuenta que $-y_i G_m(x_i) = 2I(y_i \neq G(x_i)) - 1$, entonces

$$w_i^{m+1} = w_i^m * e^{-\alpha_m I(y_i \neq G(x_i))} e^{-\beta_m} \quad (76)$$

Donde $\alpha = 2\beta_m$.

Flujo del proceso actual de pronóstico de demanda



NOTA. Elaboración propia a partir de entrevista con el grupo de interés del proceso.

Apéndice D. Líneas de comando R

```
library(fs)
library(janitor)
library(readxl)
library(modeltime)
library(tidyverse)
library(astsa)
library(Rssa)
library(ggExtra)
library(tidymodels)
library(dplyr)
library(lubridate)
library(forecast)
library(tseries)
library(TSstudio)
library(ggfortify)
library(ggplot2)
library(fpp2)
library(knitr)
library(gt)

raw_tbl <- excel_sheets('data/C115 y C141.xlsx') %>% as_tibble_col('id') %>%
mutate(data = map(id, ~ read_xlsx('data/C115 y C141.xlsx', sheet = .x,
.name_repair = make_clean_names))) %>% unnest()

# Procesamiento de las series ----
```

```

ts_list <- raw_tbl %>% rename(pedido = pedido_number) %>% select(- c(cod_unico,
descr_unica)) %>% group_by(id) %>% summarise_by_time(mes, .by = 'month', pedido =
sum(pedido)) %>% pad_by_time(mes, .by = 'month') %>% group_split(id)

c115_tbl <- ts_list[[1]] %>% filter_by_time(mes, .start_date = '2014-11-01')
c141_tbl <- ts_list[[2]]
data_tbl <- bind_rows(c115_tbl, c141_tbl)

# Visualización ----

series_plot <- data_tbl %>% ggplot(aes(mes, pedido))+ geom_line() +
facet_wrap(vars(id), scales = 'free_y', nrow = 1)

names <- c("C70" , "C30")

series_plot1 <- data_tbl %>% ggplot(aes(mes, pedido)) + geom_line(color="blue") +
labs(title='Ventas Históricas', subtitle='2010 - 2020', x='Año-Mes', y='Unidades
Vendidas') + facet_wrap(vars(id), scales = 'free_y', strip.position="left", ncol = 1)

# Partición en entrenamiento y prueba ----

series_list <- data_tbl %>% group_split(id)

series_split <- series_list %>% map(~ .x %>% time_series_split(mes, assess = 12,
cumulative = TRUE))

# * Visualización

training_list<-series_split %>% map(~ training(.x) %>% mutate(tipo='Entrenamiento'))
testing_list <- series_split %>% map(~ testing(.x) %>% mutate(tipo = 'Prueba'))

split_plot <- training_list %>% bind_rows() %>% bind_rows(testing_list %>% bind_rows())
%>% ggplot(aes(mes, pedido, color = tipo)) + geom_line() +

```

```
labs(title='Ventas Históricas', subtitle='2010 - 2020', x='Año-Mes', y='Unidades
Vendidas') + facet_wrap(vars(id), ncol = 1, scales = 'free_y') +

scale_color_manual(values = c('blue', 'red')) + theme(text = element_text(size = 10),
legend.position = 'bottom') + scale_x_datetime(expand = c(0, 0))

#Exploratorio

tsC115 <- ts(series_list[[1]]$pedido, start = c(2014,11), frequency = 12)
tsC141 <- ts(series_list[[2]]$pedido, start = c(2009,8), frequency = 12)

#Para la serie C115

ts_seasonal(tsC115, type = "all")
ts_heatmap(tsC115)
adf.test(tsC115)
adf.test(diff(tsC115))
ts_cor(tsC115)
ts_cor(diff(tsC115))

#Para la serie C141

ts_seasonal(tsC141, type = "all")
ts_heatmap(tsC141)
adf.test(tsC141)
adf.test(diff(tsC141))
acf(tsC141)

# Datos SERIE C141 - Estructura aplica para seri C15

split_series <- series_split

data_tbl <- split_series %>% map(~ training(.x) %>% mutate(tipo ='Entrenamiento'))
%>% bind_rows()
```

```

raw_tbl <- data_tbl %>% filter(id == 'C141')

ts_plot <- raw_tbl %>% ggplot(aes(mes, pedido)) + gg_ts()

plotly::ggplotly(ts_plot)

# Datos de entrenamiento y prueba

data_tbl <- raw_tbl %>% select(id, mes, pedido)

prueba_tbl <- split_series %>% map(~ testing(.x) %>% mutate(tipo = 'Prueba'))%>%
bind_rows() %>% filter(id == 'C141')

# Modelos ----

# * ARIMA ----

arima_fit1 <- arima_reg(mode="regression",seasonal_period = 12, non_seasonal_ar =
NULL, non_seasonal_differences = NULL, non_seasonal_ma = 2, seasonal_ar = 1,
seasonal_differences = NULL, seasonal_ma = NULL) %>%
set_engine('arima') %>% fit(pedido ~ mes, data = data_tbl)

arima_fit2 <- arima_reg(mode="regression",seasonal_period = NULL,
non_seasonal_ar = 2, non_seasonal_differences = NULL, non_seasonal_ma
= NULL, seasonal_ar = NULL, seasonal_differences = NULL,
seasonal_ma = NULL) %>%
set_engine('arima') %>% fit(pedido ~
mes, data = data_tbl)

arima_fit <- arima_reg() %>% set_engine('auto_arima') %>% fit(pedido ~ mes, data =
data_tbl)

arima_modeltime_table <- modeltime_table(arima_fit)

arima_modeltime_table1 <- modeltime_table(arima_fit1)

arima_modeltime_table%>%modeltime_calibrate(new_data=data_tbl)%>%
modeltime_accuracy()

```



```

arima_modeltime_table %>% modeltime_calibrate(new_data = data_tbl) %>%
modeltime_residuals()%>%select(.index, .residuals)%>%
mutate(promedio=mean(.residuals))%>%mutate(StandDev = sd(.residuals)) %>%
mutate(diferencia = (.residuals - mean(.residuals))/sd(.residuals)) %>%
ggplot(aes(.index, diferencia))+geom_line(color="blue")+labs(y = 'Residuales', x =
'Año-Mes')+ coord_cartesian(ylim = c(-3.5, 3.5))

arima_modeltime_table %>% modeltime_calibrate(new_data = data_tbl) %>%
modeltime_residuals() %>% modeltime_residuals_test()

arima_modeltime_table1%>%modeltime_calibrate(new_data=data_tbl)%>%
modeltime_accuracy()

arima_modeltime_table1 %>% modeltime_calibrate(new_data = data_tbl) %>%
modeltime_residuals()%>%select(.index, .residuals)%>%
mutate(promedio=mean(.residuals))%>%mutate(StandDev = sd(.residuals)) %>%
mutate(diferencia = (.residuals - mean(.residuals))/sd(.residuals)) %>%
ggplot(aes(.index, diferencia))+geom_line(color="blue")+labs(y = 'Residuales', x =
'Año-Mes')+ coord_cartesian(ylim = c(-3.5, 3.5))

arima_modeltime_table1 %>% modeltime_calibrate(new_data = data_tbl) %>%
modeltime_residuals() %>% modeltime_residuals_test()

# * Prophet ----

prophet_recipe <- recipe(pedido ~ mes, data_tbl)

prophet_grid_tbl <- tibble(changepoint_num = seq(from = 2, to = 28, by = 2)) %>%
expand_grid(changepoint_range = c(0, 1, by = 0.2)) %>%
mutate(seasonality_yearly = 'auto') %>% create_model_grid(f_model_spec
= prophet_reg,engine_name='prophet')

```

```

prophet_wfset <- workflow_set(preproc = list(prophet_recipe),
models = prophet_grid_tbl$.models,cross = TRUE)

prophet_wfset %>% write_rds()

# ** Fit prophet ----

parallel_start(3)

prophet_modeltime_table <- prophet_wfset %>%
modeltime_fit_workflowset(data = data_tbl,ontrol = control_fit_workflowset(
verbose= TRUE ,allow_par = TRUE))

parallel_stop()

# *** Mejor prophet ----

prophet_calib_tbl          <-          prophet_modeltime_table          %>%
modeltime_calibrate(data_tbl)

best_prophet <- prophet_calib_tbl %>% modeltime_accuracy() %>%
arrange(desc(rsq), rmse) %>% slice(1) %>% pull(.model_id)

prophet_calib_tbl %>% modeltime_accuracy() %>% arrange(desc(rsq), rmse) %>%
gt() %>% tab_header('Prophet - Métricas de desempeño')

prophet_calib_tblt <- prophet_modeltime_table %>% modeltime_calibrate(data_tbl)

best_prophet_modeltime_table          <-          prophet_modeltime_table          %>%
filter(.model_id == best_prophet)

best_prophet_modeltime_table %>% pull_modeltime_model(2)

# * Xgboost ----

# ** Feature engineering ----

# *** Rolling lags para mejorar el modelamiento de tendencias ----

data_xgboost_full_tbl <- data_tbl %>%

```

```

bind_rows(future_frame(.data = ., mes, .length_out = 12)) %>%

tk_augment_lags(pedido, .lags = 12) %>% tk_augment_slidify(.value = pedido_lag12, .f
= mean, .period = c(6, 12, 24),.align = 'center',

.partial = TRUE) %>% rename_with(.cols = contains('lag'), .fn = ~ str_c("lag_", .))
%>% select(- lag_pedido_lag12)


data_xgboost_full_tbl %>% select(-id) %>% pivot_longer(-mes)%>%
ggplot(aes(mes, value, color=name))+gg_ts()


data_xgboost_tbl <- data_xgboost_full_tbl %>% filter(!is.na(pedido))
forecast_xgboost_tbl <- data_xgboost_full_tbl %>% filter(is.na(pedido))
prueba_xgboost_tbl <- prueba_tbl %>% left_join(forecast_xgboost_tbl,by = 'mes') %>%
janitor::remove_empty('cols') %>% rename(id = id.x, pedido = pedido.x)


# **** Xgboost recipe ----

xgboost_recipe <- recipe(pedido ~ ., data_xgboost_tbl) %>%

step_timeseries_signature(mes) %>% step_fourier(mes, period = c(18, 2*18), K = 2) %>%
step_rm(mes) %>% step_rm(matches("(xts$)|(iso$)")) %>% step_zv(all_predictors()) %>%
step_dummy(all_nominal_predictors(), one_hot = TRUE) %>%
step_naomit(starts_with("lag_")) %>% step_range(mes_index.num)


xgboost_recipe %>% prep() %>% juice() %>% glimpse() %>%
gt() %>% tab_header('Creación de Variables')


xgboost_grid_tbl <- tibble(

learn_rate = c(0.001, 0.01, 0.1, 0.2, 0.4, 0.6, 0.8, 1.0)) %>% expand_grid(mtry = 10,
trees = c(5, 10, 50), min_n = c(4, 10)) %>% create_model_grid( f_model_spec =
boost_tree, engine_name = "xgboost", mode = "regression")

```

```

xgboost_wfset <- workflow_set(preproc = list(xgboost_recipe), models =
xgboost_grid_tbl$.models, cross = TRUE)

# ***** Fit Xgboost ----

parallel_start(3)

xgboost_modeltime_table <- xgboost_wfset %>%
modeltime_fit_workflowset(data = data_xgboost_tbl,
ontrol = control_fit_workflowset(verbose = TRUE, allow_par = TRUE))
parallel_stop()

xgboost_calib_tbl <- xgboost_modeltime_table %>%
modeltime_calibrate(prueba_xgboost_tbl)

best_xgboost <- xgboost_calib_tbl %>% modeltime_accuracy() %>% filter(rsq > 0.1) %>%
arrange(rmse, desc(rsq)) %>% slice(1) %>% pull(.model_id)

best_xgboost_modeltime_table <- xgboost_modeltime_table %>% filter(.model_id ==
best_xgboost)

best_xgboost_modeltime_table %>% pull_modeltime_model(38)

# Calibración ----

calib_tbl <- combine_modeltime_tables(arima_modeltime_table,
arima_modeltime_table1, best_prophet_modeltime_table,
best_xgboost_modeltime_table) %>%
modeltime_calibrate(prueba_xgboost_tbl)

# Desempeño ----

calib_tbl %>% modeltime_accuracy() %>% arrange(rmse)

# Prueba de pronóstico ----

forecast_tbl <- calib_tbl %>% modeltime_forecast(new_data = prueba_xgboost_tbl,

```

```

actual_data = data_tbl) %>% mutate(.model_desc = factor(.model_desc, levels =
c('ACTUAL', 'ARIMA(2,0,0) WITH NON-ZERO MEAN',
'ARIMA(0,0,2)(1,0,0)[12] WITH NON-ZERO MEAN',
'RECIPE_PROPHET_REG_2', 'RECIPE_BOOST_TREE_38'),
labels = c('Original', 'AR(2)', 'SARIMA', 'Prophet', 'Xgboost'))

forecast_tbl %>% arrange(.index) %>% filter_by_time(.index, .start_date = '2019-
07') %>% ggplot() + gg_ts(aes(.index, .value, color = .model_desc)) +
geom_line(data = prueba_tbl, mapping = aes(mes, pedido), linetype=1, size=0.2) +
scale_color_manual(values = c('black', "#D43F3A", "#9632B8", "#31a354", '#EEA236')) +
labs(color = 'Modelos') + theme(legend.position = 'bottom') +
ggtitle('C141')

```