

TRABAJO DE FIN DE MÁSTER

ABC PROPHET: Innovación en la Predicción Automatizada del Tráfico de Llamadas y AHT

Autor:
Laura ROCA

Tutor:
Verónica RUIZ

26 de agosto de 2024

Índice general

1. Introducción	1
1.1. Descripción del problema	2
1.2. Objetivos	3
1.2.1. Objetivo general	3
1.2.2. Objetivos específicos	3
1.3. Casos de uso	4
1.4. Estructura de la memoria	5
2. Estado del arte	6
3. Metodología	8
3.1. Entendimiento del negocio	9
3.2. Entendimiento de los datos	9
3.3. Preparación de los datos	10
3.4. Modelado	12
3.5. Evaluación	12
3.5.1. ¿Por qué WAPE?	13
3.6. Despliegue	13
4. Análisis exploratorio	15
4.1. Línea 1	15
4.1.1. Tráfico de llamadas	15
4.1.2. AHT	17
4.2. Línea 2	19
4.2.1. Tráfico de llamadas	19
4.2.2. AHT	21
4.3. Línea 3	22
4.3.1. Tráfico de llamadas	22
4.3.2. AHT	24
4.4. Línea 4	26
4.4.1. Tráfico de llamadas	26
4.4.2. AHT	27
4.5. Línea 5	28
4.5.1. Tráfico de llamadas	28
4.5.2. AHT	30
5. Modelos	32
5.1. Pycaret Regresión	32
5.2. Pycaret Time Series	34
5.3. Facebook Prophet	35
5.4. Gated recurrent unit (GRU)	37

6. Resultados	39
6.1. Línea 1	39
6.1.1. Tráfico de llamadas	39
6.1.2. AHT	41
6.2. Línea 2	43
6.2.1. Tráfico de llamadas	43
6.2.2. AHT	45
6.3. Línea 3	46
6.3.1. Tráfico de llamadas	46
6.3.2. AHT	48
6.4. Línea 4	49
6.4.1. Tráfico de llamadas	49
6.4.2. AHT	51
6.5. Línea 5	53
6.5.1. Tráfico de llamadas	53
6.5.2. AHT	54
6.6. App: ABC PROPHET	56
7. Conclusiones	58
Bibliografía	59

Capítulo 1

Introducción

En la última década, los avances en modelos predictivos y las técnicas de machine learning han revolucionado múltiples industrias, facilitando la automatización y optimización de procesos críticos. Estos enfoques permiten a las organizaciones prever comportamientos futuros basados en el análisis de grandes volúmenes de datos históricos, lo que a su vez impulsa la toma de decisiones más informadas y precisas. La capacidad de anticipar demandas, optimizar recursos y mejorar la eficiencia operativa se ha convertido en un diferenciador clave en un mercado cada vez más competitivo y exigente.

En el sector Business Process Outsourcing (BPO), la predicción del tráfico de llamadas y del Tiempo Medio de Manejo (AHT) es fundamental para garantizar una operación eficiente. Las empresas BPO, que ofrecen servicios como atención al cliente y soporte técnico, dependen en gran medida de la precisión de estas predicciones para asignar adecuadamente sus recursos humanos y tecnológicos. Una predicción precisa no solo ayuda a mitigar los costos asociados con la subdotación o sobrecarga de personal, sino que también es esencial para mantener altos niveles de calidad en el servicio.

Este trabajo de fin de máster (TFM) se centrará en la realización de múltiples predicciones utilizando diversas librerías de Python y modelos predictivos, con el objetivo de garantizar una predicción precisa, sencilla y rápida del tráfico de llamadas y AHT. En particular, el estudio se enfocará en una campaña específica dentro de una empresa del sector BPO en Colombia, que abarca cinco líneas de servicio. Para cada una de estas líneas, se llevarán a cabo dos predicciones: una para el volumen de llamadas y otra para el AHT, lo que en total representa diez series de tiempo a predecir. Además, como parte integral de este proyecto, se desarrollará una aplicación que automatizará el proceso de predicción, generando nuevas estimaciones de manera automática cada semana. Esto permitirá a la empresa recibir actualizaciones continuas y precisas, facilitando una gestión más dinámica y reactiva de sus operaciones.

1.1. Descripción del problema

En el contexto de la empresa “ABC” del sector BPO en Colombia, la predicción del tráfico de llamadas y AHT ha sido un desafío constante debido a la necesidad de optimizar la asignación de recursos y garantizar un servicio de calidad. Hasta hace poco, estas predicciones se realizaban utilizando un modelo de media móvil sencillo en Excel, que si bien proporcionaba una base para la planificación, no ofrecía la precisión ni la flexibilidad necesarias para anticipar adecuadamente las fluctuaciones en la demanda. Este enfoque resultaba en predicciones que a menudo debían ajustarse manualmente, lo que no solo incrementaba el riesgo de errores, sino que también limitaba la capacidad de respuesta ante cambios abruptos en los patrones de llamadas.

Además, la dependencia de un proceso manual para ajustar las predicciones generaba un considerable gasto de tiempo y recursos humanos. Cada línea de servicio en la campaña contaba con una persona dedicada a realizar estos ajustes, lo que implicaba un esfuerzo continuo y repetitivo para revisar y modificar las estimaciones a medida que se obtenían nuevos datos. Este proceso no solo era ineficiente, sino que también desviaba recursos que podrían haberse utilizado en tareas de mayor valor añadido. La necesidad de una solución más automatizada y precisa se hacía evidente, ya que el método actual no solo era costoso en términos de tiempo, sino que también limitaba la capacidad de la empresa para optimizar sus operaciones de manera proactiva y eficiente.

Este problema ha llevado a la búsqueda de un enfoque más avanzado y automatizado para la predicción del tráfico de llamadas y del AHT, que no solo mejore la precisión de las estimaciones, sino que también reduzca la carga de trabajo manual. La implementación de modelos predictivos más sofisticados y la automatización del proceso de predicción semanalmente mediante la aplicación “ABC PROPHET” se presentan como la solución necesaria para superar las limitaciones del enfoque actual, permitiendo una gestión operativa más eficiente y efectiva.

1.2. Objetivos

1.2.1. Objetivo general

Desarrollar un modelo predictivo y una aplicación automatizada para mejorar la precisión y eficiencia en la estimación del tráfico de llamadas y AHT en una campaña de una empresa BPO en Colombia.

1.2.2. Objetivos específicos

- Comparar y evaluar la efectividad de diferentes modelos predictivos, para identificar el enfoque que ofrezca la mayor precisión en las predicciones de tráfico de llamadas y AHT.
- Desarrollar e implementar una aplicación automatizada que genere modelos predictivos precisos, permitiendo la generación de predicciones semanales de manera automática, reduciendo así el tiempo y esfuerzo humano necesario para ajustar manualmente las estimaciones.

1.3. Casos de uso

Las predicciones del tráfico de llamadas y AHT generadas por modelos predictivos avanzados y automatizados ofrecen diversas aplicaciones prácticas dentro de una empresa BPO. A continuación, se describen los principales casos de uso:

- **Optimización de la Planificación de Turnos:** Las predicciones semanales del tráfico de llamadas permiten diseñar turnos de trabajo más precisos para cada línea de servicio. Esto asegura que haya suficiente personal durante los períodos de mayor demanda y evita tener agentes ociosos en horas valle, optimizando así los costos laborales.
- **Optimización de la Carga de Trabajo:** Conociendo de antemano el AHT, se puede distribuir mejor la carga de trabajo entre los agentes, asegurando una gestión equilibrada de las llamadas y evitando la saturación de personal.
- **Planificación Presupuestaria:** Las predicciones precisas permiten planificar el presupuesto de personal para cada línea de servicio de manera más eficiente. Al conocer de antemano la demanda esperada, se pueden asignar recursos humanos de forma óptima, evitando tanto la subdotación como la sobredotación, y asegurando que el presupuesto de personal se ajuste a las necesidades reales de cada línea.

1.4. Estructura de la memoria

El trabajo se estructura en seis capítulos principales.

1. **Estado del arte:** Se revisa la literatura existente sobre predicción de tráfico de llamadas y AHT en el sector BPO. Se analizan los modelos predictivos más relevantes y las herramientas utilizadas en machine learning, justificando la elección de los enfoques adoptados en el proyecto.
2. **Metodología:** En este apartado, se presenta una descripción exhaustiva del proceso seguido para la realización del proyecto, abarcando tanto la planificación como la ejecución. Se detallan las herramientas tecnológicas, técnicas analíticas y métodos empleados.
3. **Análisis exploratorio:** Se realiza un análisis preliminar de los datos disponibles, identificando patrones y tendencias clave. Este análisis guía las decisiones metodológicas posteriores y proporciona una base para la modelización predictiva.
4. **Modelos:** se ofrece una explicación detallada de cada uno de los modelos implementados en el proyecto. Se aborda la teoría subyacente de cada modelo, destacando los fundamentos conceptuales que los sustentan. Además, se describe minuciosamente el proceso de adaptación de cada modelo a las necesidades específicas del proyecto.
5. **Resultados:** Se presentan y evalúan los resultados obtenidos, comparando las predicciones con los datos reales. Se discuten los beneficios en términos de precisión y eficiencia, así como las mejoras operativas logradas mediante la automatización.
6. **Conclusiones:** Se resumen los hallazgos principales y se discuten las implicaciones para la empresa y el sector. También se reflexiona sobre las limitaciones del estudio y se sugieren posibles mejoras para futuras investigaciones.

Capítulo 2

Estado del arte

A continuación, se presentan y analizan las diferentes fuentes y bibliografías sobre el tema a abordar. El análisis se seguirá de manera coherente y ordenada con la finalidad de conocer el estado actual de la discusión científica sobre las predicciones en el sector BPO.

La predicción de llamadas entrantes en un centro de contacto es crucial para anticipar la demanda y planificar los recursos necesarios. Un estudio [6] analizó datos históricos utilizando modelos de aprendizaje automático y series de tiempo para identificar patrones en la demanda. Este enfoque permite optimizar operaciones, reducir costos y mejorar la experiencia del cliente al gestionar eficazmente los recursos del centro de contacto.

En colaboración con el grupo MásMóvil [10], se realizó un estudio para predecir el número de llamadas entrantes al centro de contacto de Pepephone utilizando un conjunto de datos disponible. El análisis comparó diversas metodologías, desde aproximaciones estadísticas hasta redes neuronales avanzadas como LSTM. La comparación se centró en métricas como MAE y RMSE, evaluando modelos con y sin variables exógenas. Se utilizó Python en Jupyter Project, implementando SARIMA con la librería statsmodels y redes neuronales con TensorFlow Keras, demostrando la efectividad de cada enfoque en diferentes escenarios.

En estudios recientes [6], se ha destacado el uso de PyCaret como una herramienta altamente eficiente en la construcción de modelos de aprendizaje automático para la predicción de la demanda de llamadas entrantes en centros de contacto. Según los autores, PyCaret automatiza gran parte del proceso de modelado, lo que no solo ahorra tiempo y esfuerzo en comparación con enfoques que requieren la prueba manual de múltiples modelos, sino que también es particularmente efectivo en el análisis de series de tiempo. La capacidad de PyCaret para optimizar automáticamente los hiperparámetros del modelo se traduce en una mejora significativa en la precisión de las predicciones, lo que lo convierte en una herramienta valiosa en la gestión de la complejidad predictiva en entornos dinámicos como los centros de contacto.

El estudio de múltiples series de tiempo ha llevado al desarrollo de proyectos [10] donde se ha utilizado PyCaret como parte de las herramientas de AutoML para la predicción en series temporales. En su estudio, se evaluaron herramientas como AutoGluon, Auto-Sklearn y PyCaret, destacando cómo estas plataformas automatizan el proceso de construcción de modelos predictivos sin requerir una profunda experiencia en ciencia de datos. Los resultados subrayan que la eficacia de PyCaret, al igual que las otras herramientas, depende en gran medida de la

especificidad del conjunto de datos utilizado, lo cual es crítico para la precisión en la predicción de series temporales.

Por otro lado, Facebook Prophet ha sido destacado por su capacidad para manejar datos de series temporales con características estacionales y tendencias no lineales. Por ejemplo, [2] utilizaron Prophet para predecir el tráfico a corto plazo en una carretera de Hyderabad, logrando predicciones precisas en términos de tendencia y límites de confianza. Aunque compararon Prophet con SARIMA, se destacó que Prophet fue efectivo al establecer límites superiores e inferiores alrededor de las predicciones, lo que permitió capturar adecuadamente la variabilidad del tráfico. De manera similar, en otro estudio [8] demostraron la precisión de Prophet en la predicción de ventas en supermercados, subrayando su utilidad en la elaboración de pronósticos con bajo margen de error en datos de series temporales. Estos estudios refuerzan la relevancia de Prophet como una herramienta robusta para predicciones en diversos contextos.

Capítulo 3

Métodología

En esta sección se describirá detalladamente el procedimiento y los pasos seguidos en el desarrollo de este trabajo. Para comenzar, se presentará un esquema de la metodología utilizada, la cual está basada en la metodología CRISP-DM. Este enfoque es ampliamente utilizado en proyectos de ciencia de datos, especialmente cuando el problema de negocio y los resultados esperados están claramente definidos. Su enfoque secuencial facilita la comprensión y gestión del proyecto por parte del cliente, incluso si no tiene experiencia en proyectos de datos. Además, CRISP-DM ayuda a coordinar actividades y gestionar expectativas al prever fases y resultados. [7].

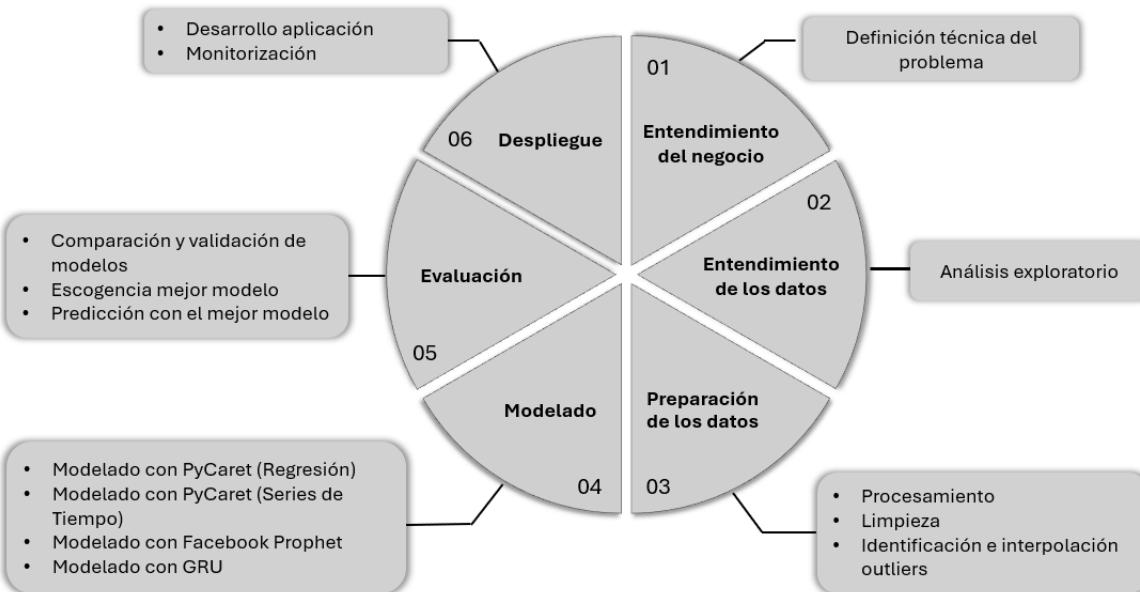


FIGURA 3.1: Metodología

3.1. Entendimiento del negocio

En esta primera fase, el equipo de trabajo se reunió con el equipo de Workforce Management (WFM), quienes son responsables de realizar las predicciones de tráfico de llamadas y aht, asignar turnos de trabajo y entregar estas predicciones al área de negocio para la elaboración de presupuestos. Estas reuniones fueron fundamentales para identificar los objetivos clave del proyecto y comprender las necesidades específicas de los involucrados.

El objetivo principal que se estableció fue mejorar la precisión de las predicciones actuales, tanto a nivel diario como mensual, con la capacidad de actualizar estas predicciones semanalmente. Esta mejora en la precisión es crucial para optimizar la asignación de recursos y asegurar que las operaciones diarias se alineen con las expectativas del negocio. Además, se definió la necesidad de desarrollar una interfaz que permitiera al equipo de WFM no solo acceder a las predicciones de manera automatizada, sino también ajustar los parámetros según sea necesario y realizar un análisis detallado de los datos. Esta solución facilitará que el equipo de WFM gestione de manera más eficiente la asignación de turnos y la elaboración de presupuestos, garantizando que las decisiones se basen en datos actualizados y precisos.

3.2. Entendimiento de los datos

Es importante destacar que tanto los datos históricos de cada línea como el conjunto de datos relacionados con los días festivos de cada línea fueron proporcionados por el equipo de WFM. Estos datos corresponden a la misma información que ellos utilizan actualmente para las predicciones que realizan, lo que garantiza la consistencia y relevancia de los análisis llevados a cabo en este proyecto.

En esta fase del proyecto, se realizó un análisis exploratorio exhaustivo para comprender en profundidad la naturaleza de los datos disponibles y su comportamiento a lo largo del tiempo. Inicialmente, se graficaron las series de tiempo para visualizar su evolución. Además, se visualizó cuáles datos fueron identificados como atípicos para comprender mejor las anomalías presentes en la serie.

Se procedió a realizar la descomposición de la serie de tiempo con el fin de analizar sus componentes principales: tendencia, estacionalidad y componente aleatorio. Este análisis permitió identificar patrones subyacentes en los datos, como tendencias a largo plazo y variaciones estacionales que son recurrentes en períodos específicos.

Adicionalmente, se llevó a cabo un análisis estacional detallado, segmentando los datos por año, por mes y por día de la semana. Este enfoque permitió visualizar cómo varía la estacionalidad a lo largo del tiempo y cómo se comportan las series de tiempo en diferentes períodos. Esta comprensión detallada de la estacionalidad es fundamental para desarrollar modelos predictivos que capturen de manera precisa estas variaciones temporales y mejoren la exactitud de las predicciones diarias y mensuales, asegurando que se alineen con los patrones reales observados en el negocio.

Los resultados y detalles específicos de esta fase se presentarán en la sección **Análisis exploratorio** donde se discutirá en profundidad cada uno de los hallazgos y su impacto en el desarrollo del proyecto.

3.3. Preparación de los datos

Para simplificar el alcance de este proyecto, se decidió enfocarse en los datos de una campaña específica que cuenta con cinco líneas de servicio. Se trabajará con dos objetivos principales (targets) para las predicciones: el tráfico de llamadas y el Tiempo Medio de Manejo (AHT). A continuación, se detallan los conjuntos de datos utilizados para cada línea de servicio:

Línea 1

- **TS Llamadas:** 1528 registros desde el 1 de febrero de 2020 hasta el 7 de abril de 2024.
- **TS AHT:** 1528 registros desde el 1 de febrero de 2020 hasta el 7 de abril de 2024.

Línea 2

- **TS Llamadas:** 1528 registros desde el 1 de febrero de 2020 hasta el 7 de abril de 2024.
- **TS AHT:** 1528 registros desde el 1 de febrero de 2020 hasta el 7 de abril de 2024.

Línea 3

- **TS Llamadas:** 828 registros desde el 1 de enero de 2022 hasta el 7 de abril de 2024.
- **TS AHT:** 555 registros desde el 1 de octubre de 2022 hasta el 7 de abril de 2024.

Línea 4

- **TS Llamadas:** 394 registros desde el 11 de marzo de 2023 hasta el 7 de abril de 2024.
- **TS AHT:** 665 registros desde el 13 de junio de 2022 hasta el 7 de abril de 2024.

Línea 5

- **TS Llamadas:** 1191 registros desde el 1 de enero de 2021 hasta el 5 de abril de 2024.
- **TS AHT:** 1526 registros desde el 1 de febrero de 2022 hasta el 5 de abril de 2024.

Adicionalmente, se agregaron a los datos variables temporales tales como el año, mes, día, día de la semana y la identificación de días festivos. Estas variables se incluyeron con el objetivo de capturar patrones estacionales y efectos específicos del calendario, los cuales pueden influir significativamente en el comportamiento de las series temporales de tráfico de llamadas y AHT. Además, para la predicción del AHT, se incorporó el tráfico de llamadas como una variable exógena, ya que existe una relación directa entre el volumen de llamadas y el tiempo medio de manejo, lo que permite mejorar la precisión del modelo predictivo al considerar la influencia del tráfico en los tiempos de atención.

Así, la base de datos utilizada para las predicciones está dada por:

Variable	Tipo de variable	Descripción
fecha	Cualitativa	Fecha de registro del dato
real_calls	Cuantitativa	Volumen de llamadas recibidas
real_aht	Cuantitativa	Tiempo medio de manejo
año	Cuantitativa	Año en número
mes	Cuantitativa	Mes en número
dia	Cuantitativa	Día del mes en número
dia_semana	Cuantitativa	Día de la semana en número, lunes (0), ..., festivo (7)
state_holiday	Cuantitativa	Indicador si la fecha es un día festivo (1) o no (0)

CUADRO 3.1: Descripción de las variables utilizadas en el análisis

Y se presenta de la siguiente manera en python:

	año	mes	dia	dia_semana	state_holiday	fecha	linea	real_aht	real_calls
0	2020	2	1	5	0	2020-02-01	linea_1	338.18	293.0
1	2020	2	2	6	0	2020-02-02	linea_1	370.12	29.0
2	2020	2	3	0	0	2020-02-03	linea_1	395.45	2881.0
3	2020	2	4	1	0	2020-02-04	linea_1	404.66	2776.0
4	2020	2	5	2	0	2020-02-05	linea_1	399.82	2586.0

FIGURA 3.2: Primeras observaciones del archivo de datos

Después de recopilar los datos, se realizó un proceso de limpieza enfocado en asegurar la calidad y consistencia del conjunto de datos. Aunque no se encontraron valores nulos o faltantes, se llevaron a cabo varias acciones clave para preparar los datos para su análisis.

Primero, se corrigieron posibles datos incorrectos, realizando la verificación de que los datos fueran consistentes en términos de formato y contenido. Se revisaron las variables para asegurarse de que los valores fueran coherentes y que no hubiera errores de entrada, como fechas mal formateadas o valores fuera de rango.

A continuación, se procedió a estandarizar los nombres de las variables y categorías. Los nombres de las variables se ajustaron para ser claros y consistentes, eliminando ambigüedades y asegurando que siguieran una convención uniforme.

Por último, se implementó una función personalizada para identificar y tratar outliers, considerando los datos agrupados por día de la semana de cada año para cada línea. Se aplicó el modelo Isolation Forest para detectar outliers dentro de estos grupos específicos y asignarlos como NA. Luego, los outliers identificados fueron reemplazados por el promedio entre el dato anterior y el dato siguiente, y cualquier valor faltante residual fue imputado utilizando K-Nearest Neighbors. Este enfoque asegura un tratamiento efectivo de los outliers, manteniendo la integridad de los datos en función de las variaciones por día de la semana en cada año.

3.4. Modelado

En esta fase, cada serie de tiempo fue abordada utilizando cuatro enfoques distintos: PyCaret regresión, PyCaret time series, Facebook Prophet y una Red Neuronal GRU. Para esto, el conjunto de datos fue dividido en tres partes: conjunto de entrenamiento, validación y prueba. Los datos de prueba, correspondientes a los registros del año 2024, se reservaron para el final, con el fin de comparar el desempeño de todos los modelos de manera objetiva. Los conjuntos de entrenamiento y validación se utilizaron para ajustar y optimizar los modelos.

PyCaret regresión es una herramienta automatizada que simplifica la construcción y evaluación de modelos de regresión, probando múltiples algoritmos y seleccionando el mejor basado en métricas de rendimiento. PyCaret time series extiende esta funcionalidad a las series de tiempo, adaptando los modelos de regresión para capturar patrones temporales específicos. Adicionalmente, se calculó el promedio de las predicciones generadas ambas herramientas, creando un modelo combinado que también se incluyó para comparar su rendimiento con los otros enfoques.

Facebook Prophet es un modelo diseñado para realizar pronósticos de series de tiempo con componentes estacionales y de tendencia, y es particularmente robusto en la gestión de datos con patrones irregulares o estacionales.

Finalmente, GRU (Gated Recurrent Unit) es una arquitectura de red neuronal recurrente especializada en la captura de dependencias a largo plazo en datos secuenciales, lo que la hace adecuada para modelar series de tiempo complejas.

Cada una de estas herramientas y modelos se explicará y detallará en profundidad en el capítulo [Modelos](#).

3.5. Evaluación

En la fase de evaluación, los resultados obtenidos por cada uno de los modelos desarrollados fueron comparados utilizando la métrica WAPE (Weighted Absolute Percentage Error).

Para cada modelo: PyCaret regresión, PyCaret time series, Facebook Prophet, GRU, y el modelo combinado de PyCaret, se calculó el WAPE utilizando los datos de prueba reservados del año 2024. El WAPE permitió cuantificar el error relativo de las predicciones de cada modelo, proporcionando una base uniforme para la comparación.

Tras calcular el WAPE para todos los modelos, se seleccionó el modelo con el WAPE más bajo como el más preciso. Este modelo seleccionado es el que se utilizará para realizar las predicciones futuras, garantizando que las estimaciones sean las más confiables según los datos disponibles y el rendimiento evaluado.

3.5.1. ¿Por qué WAPE?

El Error Ponderado Absoluto Porcentual (WAPE, por sus siglas en inglés) es una métrica clave para evaluar la precisión de los modelos de pronóstico, especialmente en el contexto de los centros de llamadas. Se calcula dividiendo la suma de los errores absolutos por el volumen total de datos, lo que da como resultado una métrica que pondera el impacto de los errores en función del volumen de llamadas. La fórmula es:

$$\text{WAPE} = \frac{\sum_{t=1}^T |\hat{y}_t - y_t|}{\sum_{t=1}^T y_t}$$

donde \hat{y}_t representa el valor pronosticado, y_t el valor real, y T el número total de observaciones.

El WAPE se elige como métrica preferida en los centros de llamadas debido a varias razones críticas. En primer lugar, a diferencia de otras métricas como el MAPE (Mean Absolute Percentage Error), el WAPE maneja mejor la variabilidad en los volúmenes de llamadas. El MAPE, aunque comúnmente usado, puede verse sesgado por períodos con bajos volúmenes, ya que magnifica los errores cuando los volúmenes son pequeños o incluso puede fallar debido a la división por cero en estos casos.^[9]

El WAPE, al ponderar los errores según el volumen de llamadas, asegura que los períodos con altos volúmenes de llamadas—que son más críticos para las operaciones—tienen un mayor peso en la evaluación del pronóstico. Esto es crucial en centros de llamadas, donde los errores en la predicción durante períodos de alta demanda pueden llevar a un sobredimensionamiento o falta de personal, afectando tanto los costos operativos como la calidad del servicio. Además, el WAPE está directamente relacionado con el esfuerzo necesario para gestionar en tiempo real los recursos del centro de llamadas, lo que lo convierte en una métrica alineada con los objetivos operativos del centro.^[5]

Por estas razones, el WAPE es preferido sobre otras métricas como el MAPE en el entorno de los centros de llamadas, ya que proporciona una medida más práctica y relevante de la precisión del pronóstico en relación con los costos y la eficiencia operativa.

3.6. Despliegue

Se desarrolló una interfaz en Shiny para facilitar la visualización y análisis de las series de tiempo modeladas. La aplicación permite explorar los datos históricos, validar los modelos mediante la comparación de métricas como el WAPE, y generar predicciones diarias y mensuales. Además, incluye una sección para ajustar manualmente las predicciones y ofrece la opción de descargar las predicciones ajustadas o no en formato Excel.

El proceso de modelado, predicción y uso de la aplicación será monitoreado de manera constante. Cada viernes, se ejecutarán nuevamente los modelos para ajustar las predicciones con datos actualizados, asegurando así su precisión. Además, se realizará un acompañamiento constante a los usuarios de la aplicación en el equipo

de WFM para garantizar que puedan utilizarla de manera efectiva.

Esta aplicación será explicada con más detalle en el apartado ??, donde se abordarán sus funcionalidades y su integración en el proceso de análisis y predicción.

Capítulo 4

Análisis exploratorio

Se examina cada línea utilizando diversos gráficos y herramientas visuales. El análisis se centró en identificar patrones, tendencias y posibles anomalías en los datos históricos, lo cual es crucial para comprender el comportamiento del tráfico de llamadas y AHT y para la posterior modelización.

4.1. Línea 1

4.1.1. Tráfico de llamadas

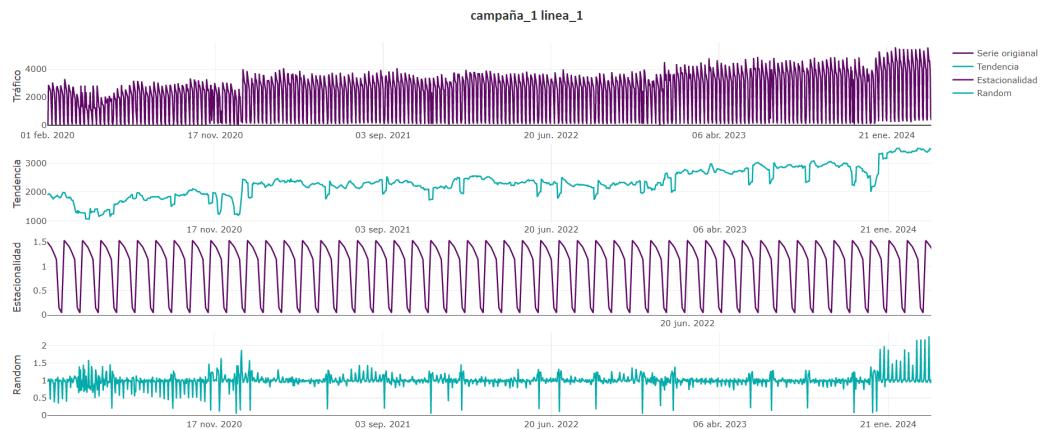


FIGURA 4.1: Línea 1- Llamadas: Descomposición diaria

El gráfico de descomposición del tráfico de la Línea 1 muestra varios componentes clave. La serie original refleja el tráfico diario con una tendencia ligeramente ascendente a lo largo del tiempo, aunque se observan varias fluctuaciones notables. La tendencia muestra esta inclinación hacia un aumento en el tráfico, a pesar de los altibajos. La estacionalidad revela un patrón fuerte y regular que se repite diariamente: los días con mayor tráfico son los lunes, con una disminución gradual hasta el domingo, que es el día con menor tráfico, formando una onda característica. El componente de ruido captura las variaciones aleatorias que no siguen los patrones anteriores, siendo más pronunciado en ciertos períodos pero generalmente bajo. Este análisis confirma la presencia de una estacionalidad diaria significativa y una tendencia general de aumento en el tráfico a lo largo del tiempo.

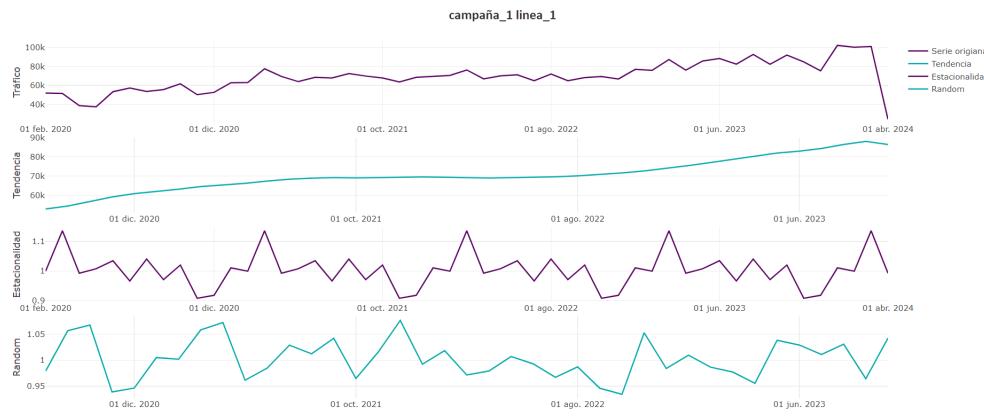


FIGURA 4.2: Línea 1- Llamadas: Descomposición mensual

Se observa una fluctuación considerable en el tráfico a lo largo de los meses, con algunos picos y valles que sugieren variaciones estacionales o eventos específicos. La tendencia muestra un aumento gradual y sostenido en el tráfico a lo largo del tiempo, indicando un crecimiento continuo en el tráfico. El componente de estacionalidad destaca un patrón cíclico mensual, aunque menos pronunciado que el diario, en donde se observa que el mes de marzo es el mes con más tráfico, contrario a noviembre. Finalmente, el componente de ruido muestra variaciones aleatorias que no siguen los patrones de tendencia o estacionalidad, siendo más notorias en algunos períodos específicos.

Observando más a fondo la estacionalidad día-mes,

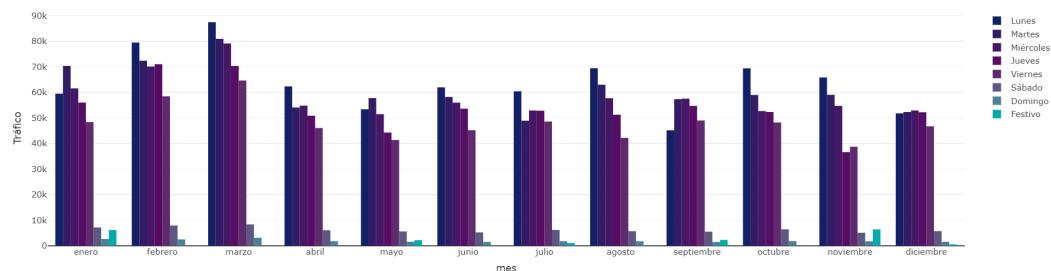


FIGURA 4.3: Línea 1- Llamadas: Estacionalidad por día de la semana-mes

Notamos que, efectivamente, los lunes, especialmente durante el mes de marzo, son los días con mayor tráfico de llamadas, mientras que los sábados, domingos y días festivos presentan el menor tráfico de llamadas.

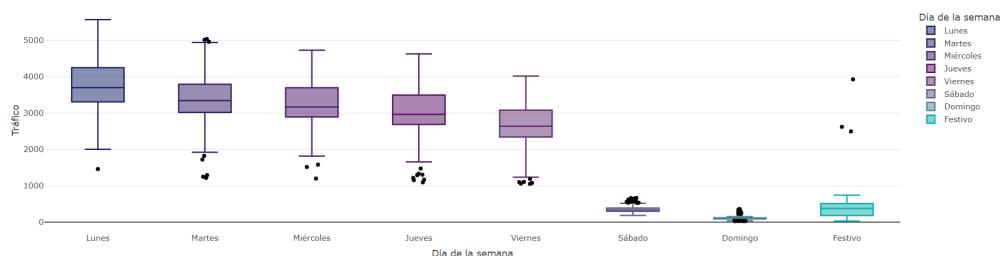


FIGURA 4.4: Línea 1- Llamadas: Box-plot día de la semana

Este gráfico de cajas (boxplot) ilustra la variabilidad del tráfico de llamadas en cada día de la semana. Se puede observar que los días de semana con mayor tráfico (lunes y martes) presentan una mayor variabilidad, con una amplia dispersión en los valores, lo que indica fluctuaciones significativas en el volumen de llamadas en estos días. En contraste, los días con menos tráfico (sábados, domingos y festivos) muestran una variabilidad mucho menor, con rangos más estrechos, lo que sugiere que el tráfico en estos días es más constante y predecible. Los puntos atípicos en algunos días indican excepciones donde el tráfico fue notablemente diferente al habitual para ese día específico.

4.1.2. AHT

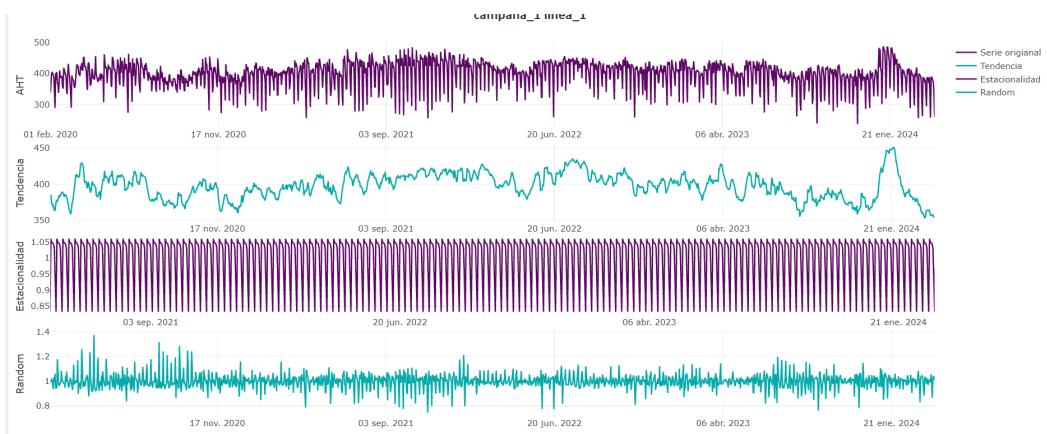


FIGURA 4.5: Línea 1-AHT: Descomposición diaria

La descomposición diaria del AHT para la Línea 1 revela una tendencia con fluctuaciones a lo largo del tiempo, incluyendo un reciente aumento en el AHT. La estacionalidad es fuerte y muestra un patrón diario bien definido, indicando ciclos predecibles. El componente de ruido es generalmente bajo, lo que sugiere que la mayoría de las variaciones en el AHT son explicadas por la tendencia y la estacionalidad, aunque se observan algunos picos que podrían reflejar eventos atípicos o condiciones excepcionales.

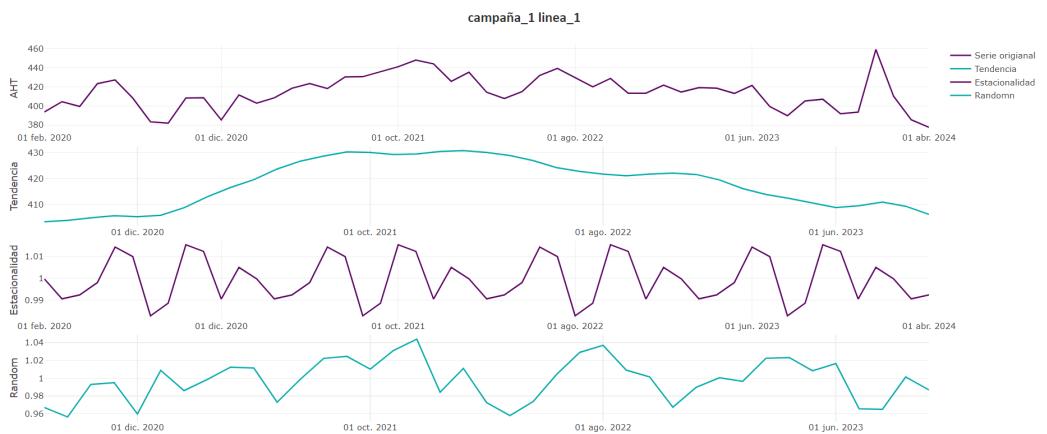


FIGURA 4.6: Línea 1-AHT: Descomposición mensual

La descomposición mensual del AHT para la Línea 1 revela una tendencia ascendente constante, lo que indica un aumento gradual en el tiempo promedio de manejo de llamadas a lo largo de los meses. Además, se observa una estacionalidad con fluctuaciones cíclicas, donde ciertos meses presentan sistemáticamente tiempos de manejo más altos o bajos. El componente de ruido muestra variaciones menores, lo que sugiere que la mayor parte de las fluctuaciones en el AHT se explican por la tendencia y la estacionalidad, con poca influencia de factores aleatorios.

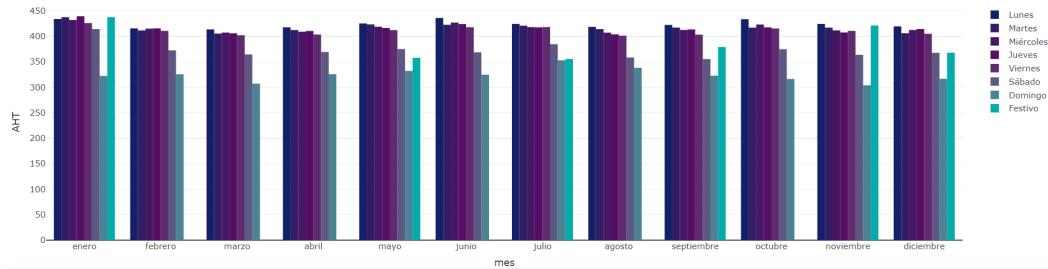


FIGURA 4.7: Línea 1- AHT: Estacionalidad por día de la semana-mes

Se observa que el AHT es generalmente más alto durante los días laborables (lunes a viernes) y tiende a ser más bajo los sábados, domingos y días festivos. Además, la variabilidad del AHT a lo largo del año es relativamente constante, aunque algunos meses, como noviembre y diciembre, presentan ligeras diferencias en el AHT entre los distintos días de la semana. Este patrón sugiere que los días de mayor actividad laboral tienden a tener tiempos de manejo más altos, mientras que los fines de semana y festivos, cuando el volumen de llamadas es menor, presentan tiempos de manejo más bajos.

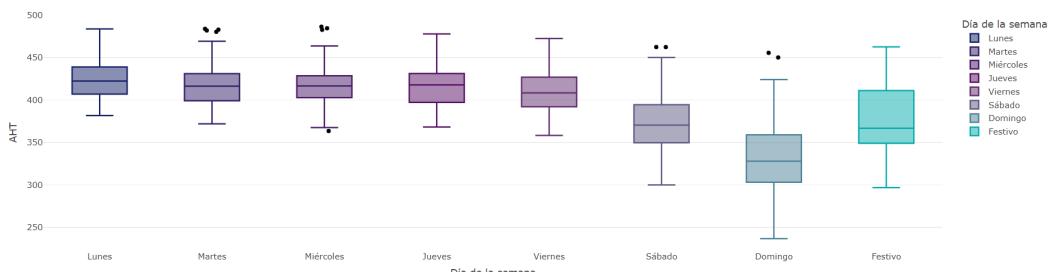


FIGURA 4.8: Línea 1- AHT: Box-plot día de la semana

Se observa que los días laborables (lunes a viernes) presentan una menor variabilidad en el AHT, con distribuciones más compactas, mientras que los sábados, domingos y festivos muestran una mayor dispersión, especialmente los domingos y festivos, lo que indica mayor variabilidad en el tiempo de manejo durante estos días. Además, los días laborables tienen algunos valores atípicos, pero son más consistentes en comparación con los fines de semana y festivos, donde la variabilidad y los outliers son más pronunciados. Esto sugiere que, aunque los días laborables son más predecibles en términos de AHT, los días no laborables son menos consistentes, posiblemente debido a la menor cantidad de datos o a fluctuaciones en el volumen de llamadas.

4.2. Línea 2

4.2.1. Tráfico de llamadas

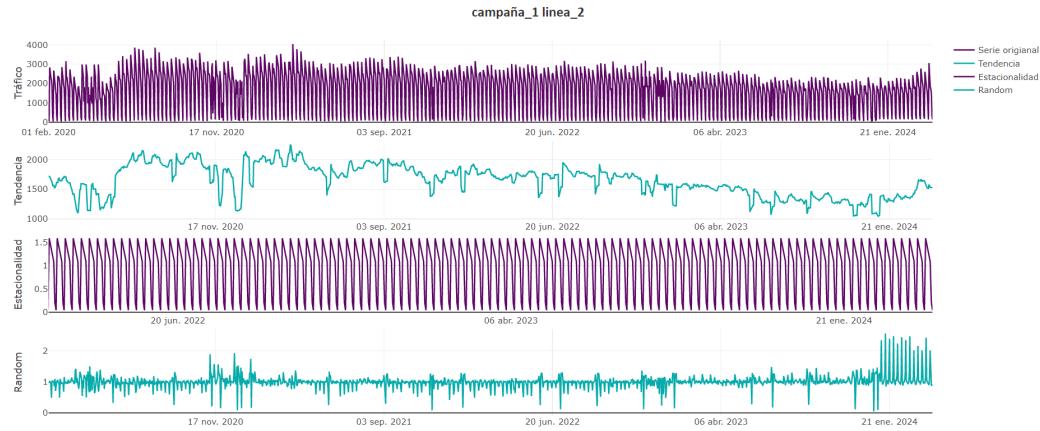


FIGURA 4.9: Línea 2- Llamadas: Descomposición diaria

La tendencia muestra fluctuaciones a lo largo del tiempo, con un aumento inicial seguido por una serie de altibajos, y una ligera recuperación hacia el final del período, lo que podría indicar variaciones en la demanda o cambios en el comportamiento de las llamadas. La estacionalidad es claramente visible y sigue un patrón diario fuerte, similar a la Línea 1, lo que sugiere un ciclo regular en el tráfico de llamadas a lo largo de la semana. Finalmente, el ruido muestra variabilidad, pero en general, es moderado, indicando que la mayoría de las fluctuaciones en el tráfico de llamadas son explicadas por la tendencia y la estacionalidad, con algunas excepciones.

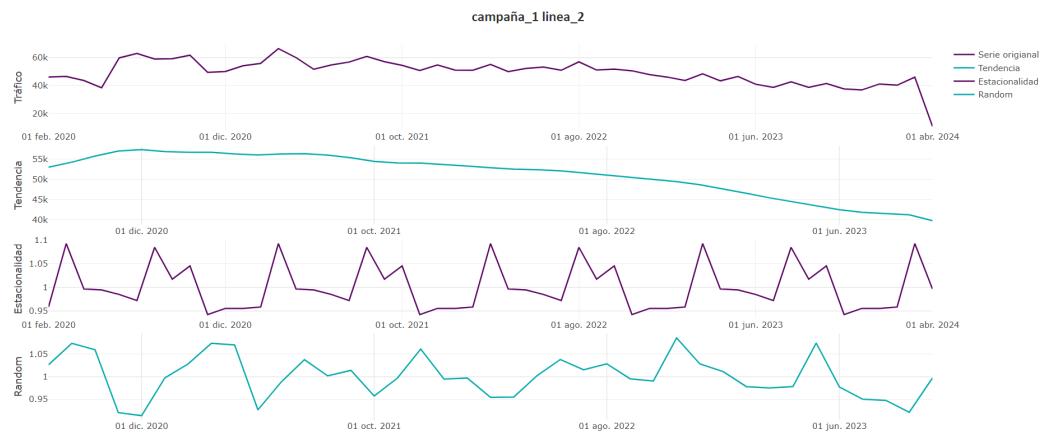


FIGURA 4.10: Línea 2- Llamadas: Descomposición mensual

La estacionalidad muestra un patrón cíclico mensual con fluctuaciones moderadas, lo que sugiere que ciertos meses tienen consistentemente más o menos tráfico de llamadas. El ruido muestra variaciones, pero en general, las fluctuaciones aleatorias son relativamente contenidas.

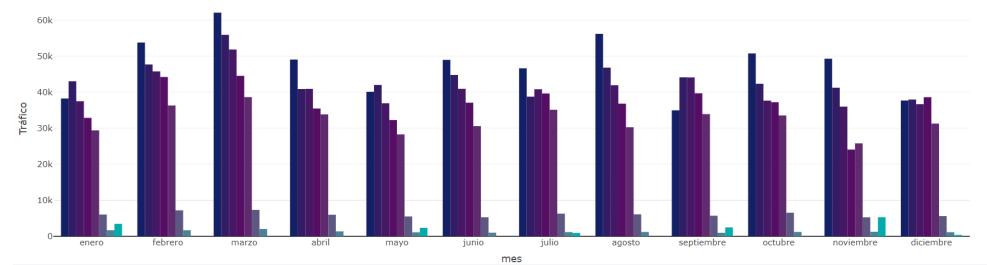


FIGURA 4.11: Línea 2- Llamadas: Estacionalidad por día de la semana-mes

Se observa que los lunes suelen tener el mayor tráfico en cada mes, seguidos de cerca por los otros días de la semana, con una disminución progresiva hacia los fines de semana. Los sábados, domingos y festivos presentan el menor tráfico, lo que es consistente a lo largo de todos los meses. Este patrón sugiere que el tráfico es más alto al comienzo de la semana y decrece hacia el fin de semana, independientemente del mes, lo que refleja una rutina operativa predecible donde los días laborables concentran la mayor cantidad de llamadas, muy similar al comportamiento de la línea 1.

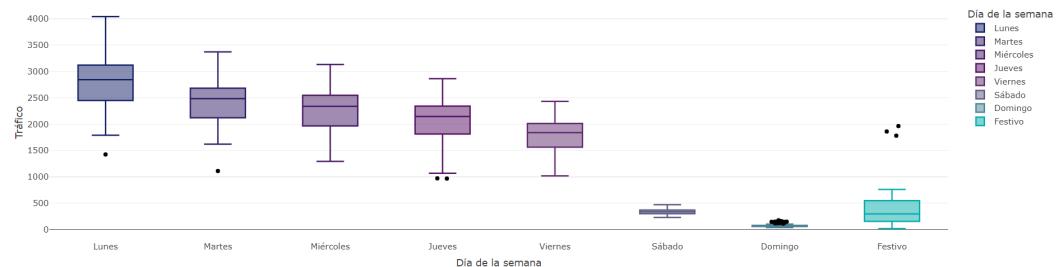


FIGURA 4.12: Línea 2- Llamadas: Box-plot día de la semana

El tráfico de llamadas, se comporta de manera muy similar a la línea 1, en donde los lunes tienen la mayor variabilidad en el tráfico de llamadas, mientras que los fines de semana y festivos presentan menor tráfico y pero los festivos con una mayor dispersión. Los días laborables mantienen una distribución más consistente, pero con algunos valores atípicos notables.

4.2.2. AHT

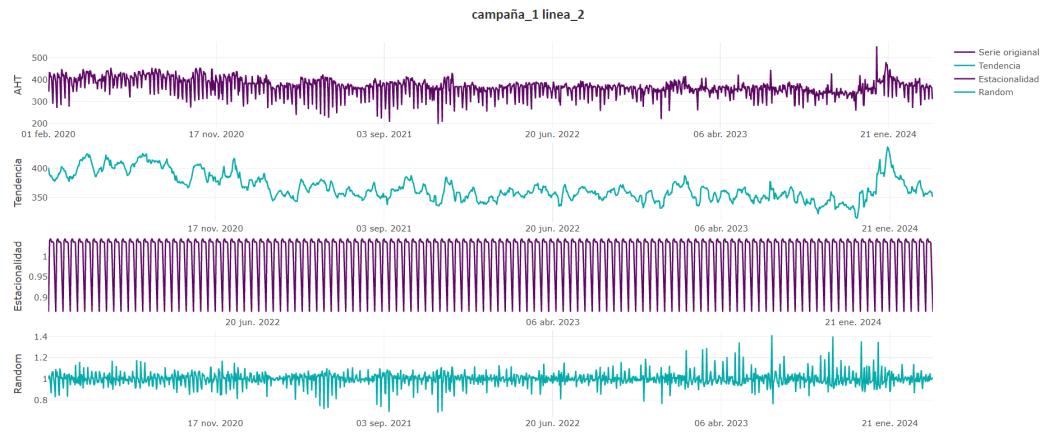


FIGURA 4.13: Línea 2-AHT: Descomposición diaria

La descomposición diaria del AHT para la Línea 2 revela una tendencia general a la baja, aunque con algunos aumentos recientes. La estacionalidad es consistente, mostrando ciclos diarios regulares en el AHT. El componente de ruido es moderado, indicando que la mayoría de las variaciones en el AHT están influenciadas principalmente por la tendencia y la estacionalidad, con algunas fluctuaciones menores.

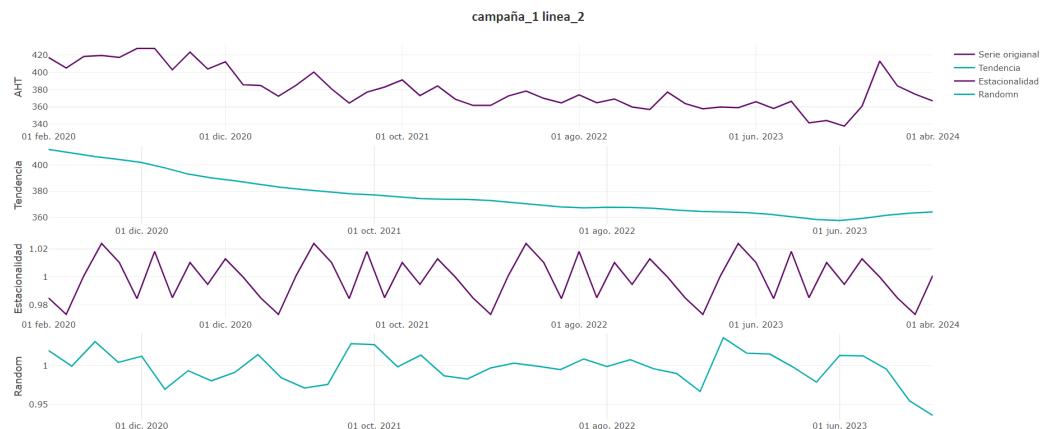


FIGURA 4.14: Línea 2-AHT: Descomposición mensual

Se observa una tendencia a la baja a lo largo del tiempo, indicando una disminución gradual en el tiempo promedio de manejo de llamadas. La estacionalidad refleja variaciones cíclicas mensuales, con algunos meses consistentemente mostrando tiempos de manejo más altos o bajos. El componente de ruido es moderado.

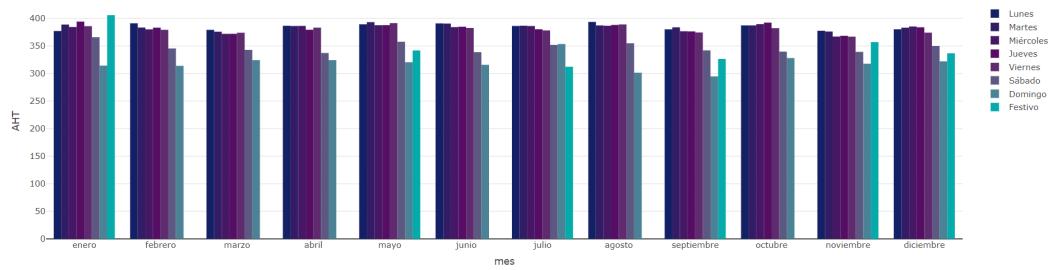


FIGURA 4.15: Línea 2- AHT: Estacionalidad por día de la semana-mes

Se observa que el AHT es bastante consistente a lo largo de los diferentes meses y días de la semana, con pequeñas variaciones entre días laborables y no laborables. Los días laborables tienden a mantener un AHT más elevado y estable, mientras que los fines de semana y festivos (con excepciones) suelen mostrar un AHT ligeramente menor.

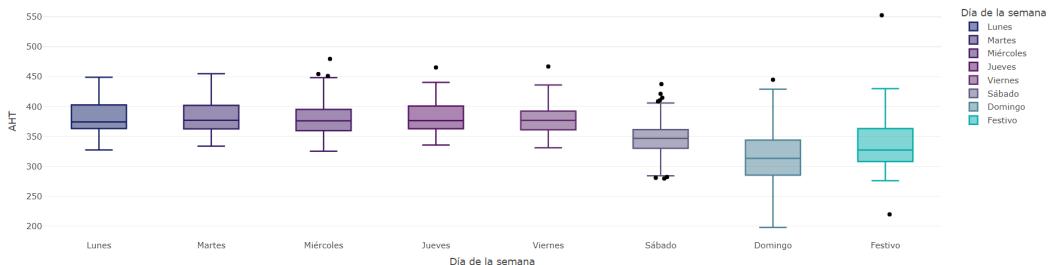


FIGURA 4.16: Línea 2- AHT: Box-plot día de la semana

Al igual que la línea 1, los días laborables (lunes a viernes) tienen una distribución del AHT más compacta y consistente, con menos variabilidad. Los sábados, domingos y festivos, por otro lado, presentan mayor dispersión y variabilidad.

4.3. Línea 3

4.3.1. Tráfico de llamadas

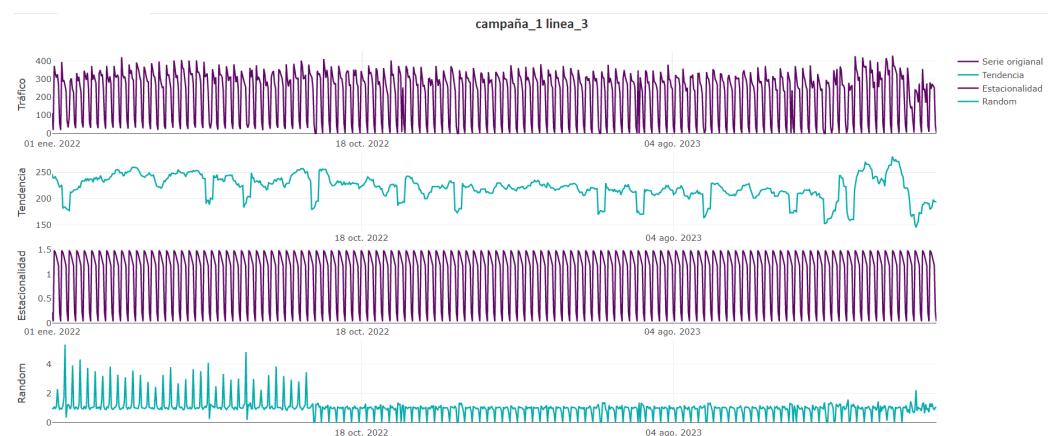


FIGURA 4.17: Línea 3- Llamadas: Descomposición diaria

La tendencia refleja oscilaciones con descensos y repuntes en el tiempo, lo que podría señalar cambios en la demanda de llamadas. La estacionalidad es marcada y consistente, mostrando un ciclo diario bien definido que sugiere un comportamiento constante en el tráfico de llamadas. El componente de ruido muestra picos esporádicos, lo que apunta a la presencia de eventos o anomalías puntuales que interrumpen el patrón regular.

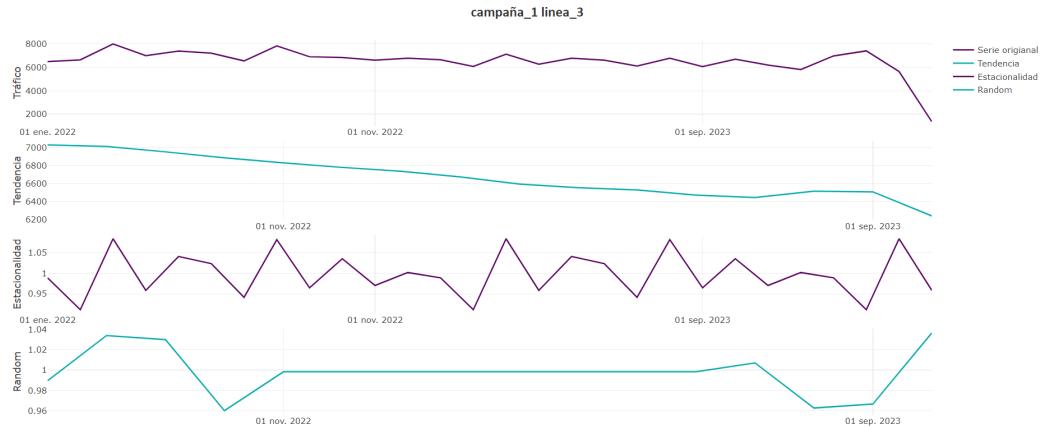


FIGURA 4.18: Línea 3-Llamadas: Descomposición mensual

La descomposición mensual del tráfico de llamadas para la Línea 3 muestra una tendencia descendente, lo que indica una disminución gradual en el volumen de llamadas a lo largo del tiempo. La estacionalidad presenta variaciones moderadas de un mes a otro, sugiriendo ciclos regulares pero no muy pronunciados en el tráfico mensual. El ruido es relativamente bajo.

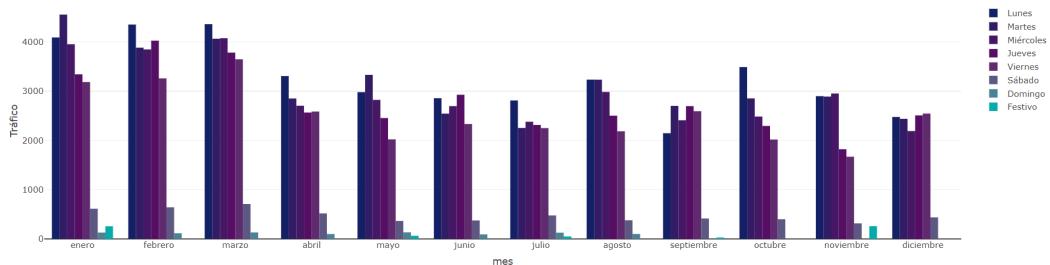


FIGURA 4.19: Línea 3- Llamadas: Estacionalidad por día de la semana-mes

Se observa el mismo comportamiento de las líneas anteriores, lunes tienden a tener el mayor tráfico, seguido por los otros días laborables, mientras que los fines de semana y festivos muestran un tráfico significativamente menor. Este patrón se mantiene de manera consistente a lo largo de los meses.

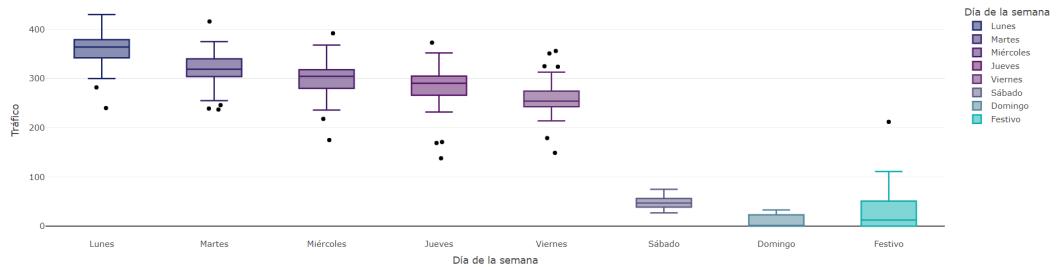


FIGURA 4.20: Línea 3- Llamadas: Box-plot día de la semana

A medida que avanza la semana, la variabilidad y el tráfico tienden a disminuir, especialmente los fines de semana , exceptuando los festivos que presentan mayor variabilidad Además, se identifican varios valores atípicos, especialmente durante los días laborables, lo que sugiere la presencia de días con tráfico anormalmente alto o bajo en comparación con el promedio semanal.

4.3.2. AHT



FIGURA 4.21: Línea 3-AHT: Descomposición diaria

En la descomposición diaria del AHT para la Línea 3, la tendencia se mantiene relativamente estable con algunas fluctuaciones notables. La estacionalidad muestra un ciclo diario constante, indicando un patrón regular en el tiempo de manejo de llamadas. El ruido es bajo, salvo por un pico aislado que sugiere un evento atípico. En general, la serie es estable y predecible, con pocas interrupciones.

Esta línea cuenta con pocos datos, lo que impidió realizar la descomposición mensual.

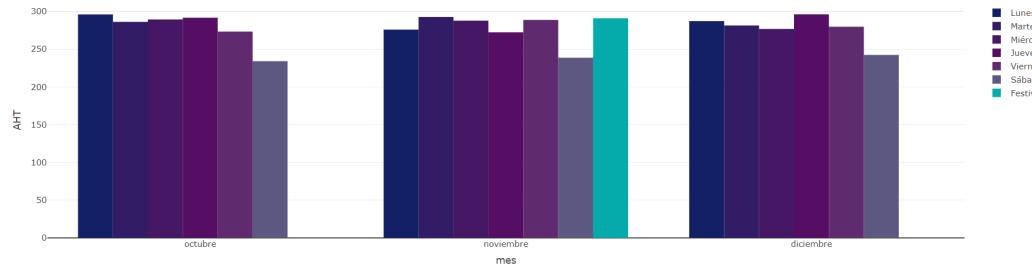


FIGURA 4.22: Línea 3- AHT: Estacionalidad por día de la semana-mes

Debido a la cantidad limitada de datos, solo se pudo analizar el AHT para algunos meses en esta línea. El gráfico muestra una consistencia en el AHT entre los días laborables durante octubre, noviembre y diciembre, con algunas diferencias menores en los fines de semana y festivos. Esto sugiere que, a pesar de la falta de datos, el AHT se mantiene relativamente estable a lo largo de estos meses.

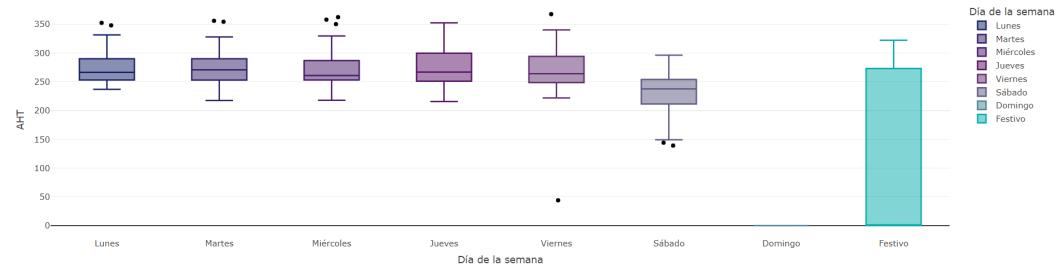


FIGURA 4.23: Línea 3- AHT: Box-plot día de la semana

Este gráfico de cajas muestra que el AHT para los domingos es siempre 0, lo que indica que no hay llamadas manejadas en esos días. Los días laborables presentan una distribución consistente del AHT, con medianas similares y algunas variaciones en la dispersión. Los días festivos, en contraste, muestran un AHT significativamente más alto y con mayor dispersión.

4.4. Línea 4

4.4.1. Tráfico de llamadas

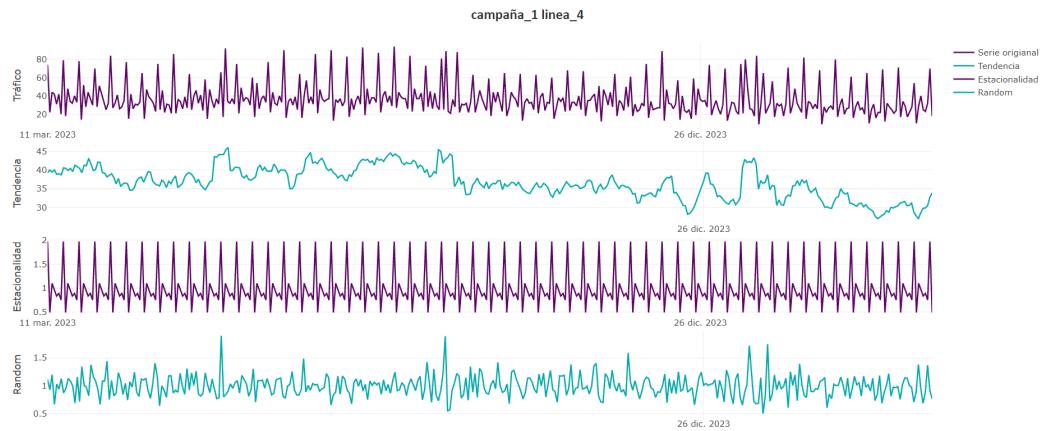


FIGURA 4.24: Línea 4- Llamadas: Descomposición diaria

En la descomposición diaria del tráfico de llamadas para la Línea 4, se observa una tendencia con ligeras fluctuaciones y algunas caídas y recuperaciones notables, lo que podría reflejar cambios en la demanda a lo largo del tiempo. La estacionalidad es bastante fuerte, mostrando un patrón diario claro y repetitivo. El ruido presenta algunas variaciones, con picos esporádicos que sugieren la presencia de eventos atípicos.

Esta línea cuenta con pocos datos, lo que impidió realizar la descomposición mensual.

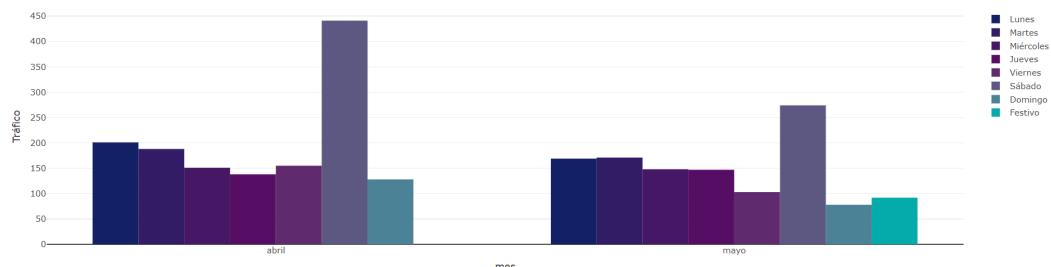


FIGURA 4.25: Línea 4- Llamadas: Estacionalidad por día de la semana-mes

Contrario a las líneas anteriores, se observa que el tráfico varía significativamente entre los días de la semana, con los sábados presentando el mayor volumen de llamadas.

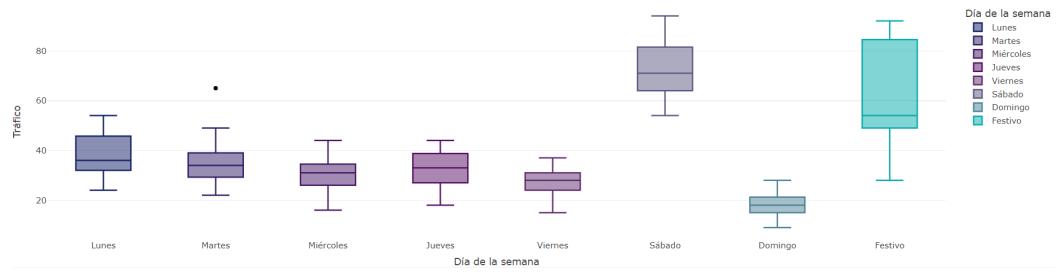


FIGURA 4.26: Línea 4- Llamadas: Box-plot día de la semana

Los días festivos presentan una mayor variabilidad en el tráfico, con un rango de valores más amplio y un tráfico considerablemente más alto en comparación con otros días. Los sábados también muestran una mayor variabilidad, aunque menor que los festivos, mientras que los días de semana (lunes a viernes) tienen un tráfico más consistente, con menores diferencias entre días. Los domingos muestran el menor tráfico y una variabilidad reducida, lo que indica un patrón de llamadas más estable y bajo en esos días.

4.4.2. AHT

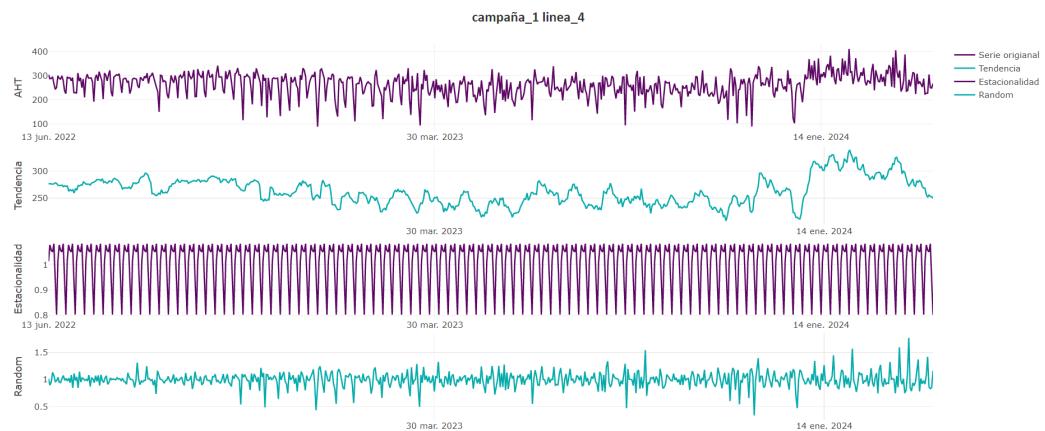


FIGURA 4.27: Línea 4-AHT: Descomposición diaria

Esta línea, presenta una tendencia general ligeramente descendente, con fluctuaciones. La estacionalidad muestra un patrón diario consistente, lo que sugiere que la duración promedio de las llamadas sigue un ciclo regular. El componente de ruido revela algunas variaciones y picos esporádicos, lo que podría señalar la presencia de eventos no habituales.

Esta línea cuenta con pocos datos, lo que impidió realizar la descomposición mensual.

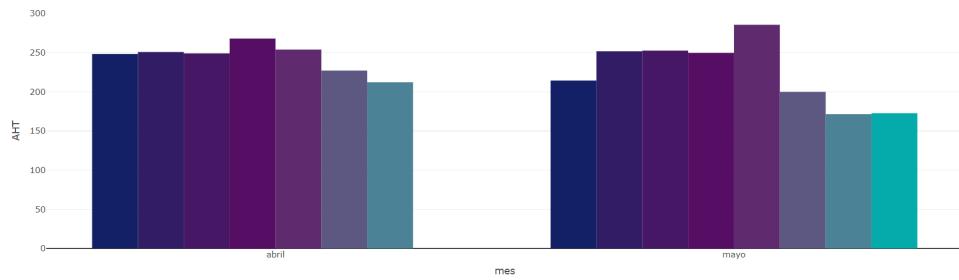


FIGURA 4.28: Línea 4- AHT: Estacionalidad por día de la semana-mes

sigue un patrón consistente durante la semana, con variaciones más marcadas en los días festivos y los fines de semana. Los días de semana tienen tiempos de manejo similares, mientras que los días festivos tienden a mostrar un aumento en el AHT.

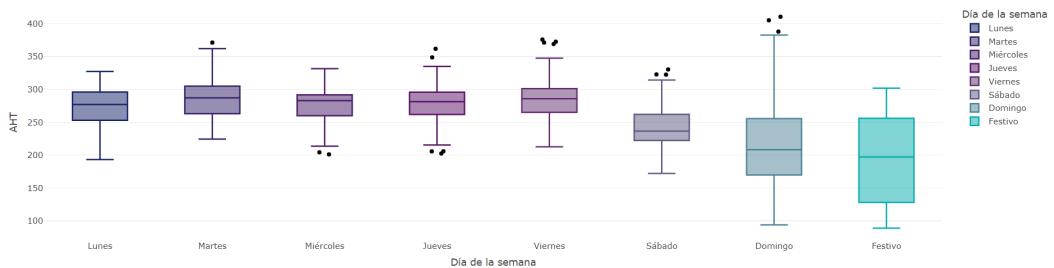


FIGURA 4.29: Línea 4- AHT: Box-plot día de la semana

Se observa que los días festivos y los domingos presentan una mayor variabilidad en el AHT, con una tendencia a tiempos más altos. Los días de semana, en cambio, tienen una distribución más consistente, aunque con algunos valores atípicos, especialmente hacia el final de la semana.

4.5. Línea 5

4.5.1. Tráfico de llamadas

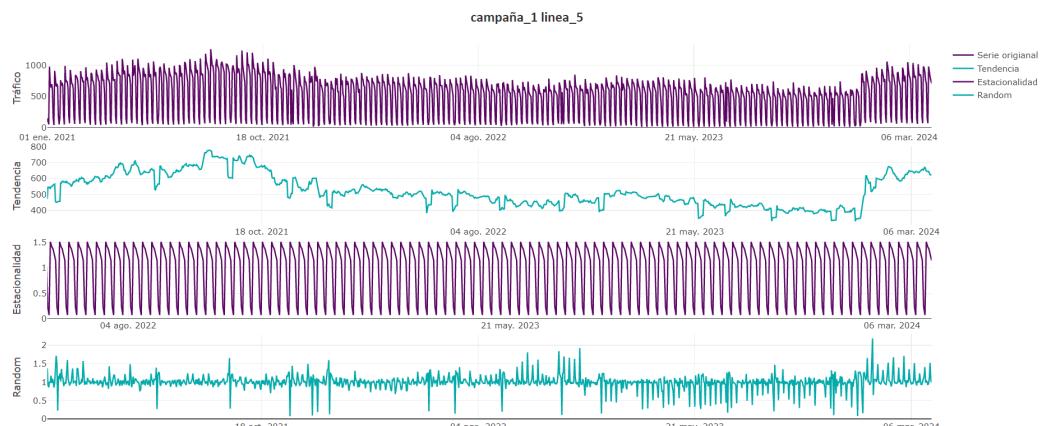


FIGURA 4.30: Línea 5- Llamadas: Descomposición diaria

En la Línea 5, el análisis de llamadas muestra una tendencia general descendente, seguida por una recuperación más reciente. La estacionalidad es muy marcada y regular, lo que indica patrones de tráfico consistentes y previsibles a lo largo del tiempo. El componente aleatorio tiene picos ocasionales, lo que sugiere la presencia de eventos imprevistos que impactan el tráfico de llamadas de manera esporádica.

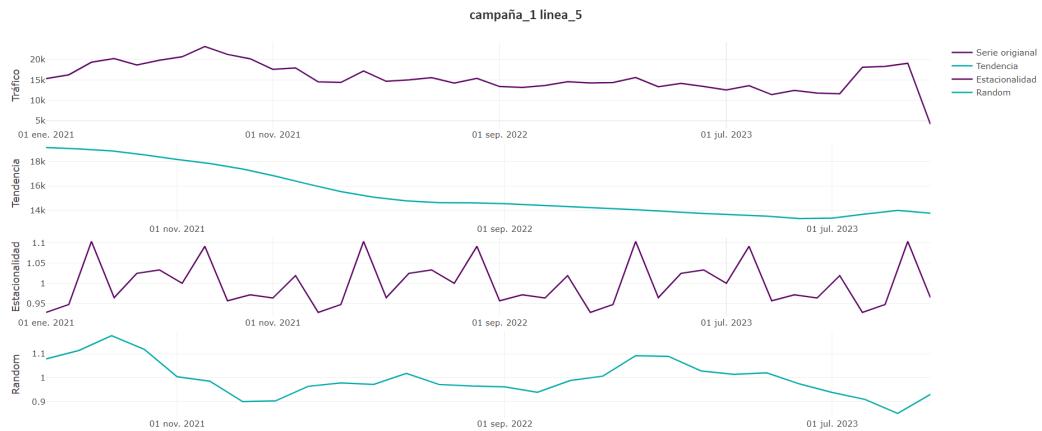


FIGURA 4.31: Línea 5-Llamadas: Descomposición mensual

En la descomposición, las llamadas muestran una tendencia decreciente a lo largo del tiempo. La estacionalidad indica fluctuaciones regulares, pero con variaciones menores. El componente aleatorio refleja una disminución en la variabilidad a lo largo del tiempo, lo que sugiere una estabilización en la variabilidad de los datos.

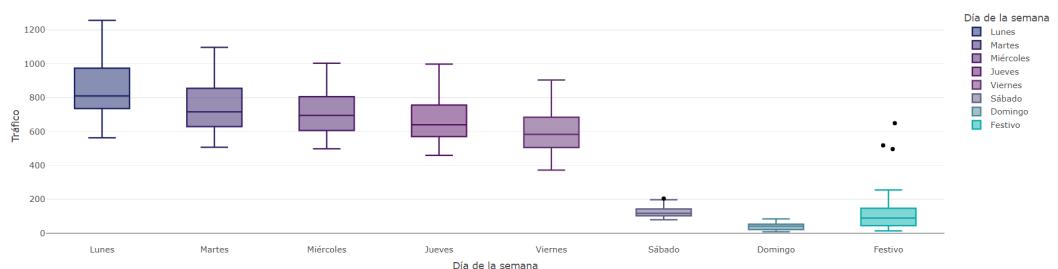


FIGURA 4.32: Línea 5- Llamadas: Box-plot día de la semana

En la Línea 5, el tráfico de llamadas es más alto los lunes, disminuyendo progresivamente a lo largo de la semana, con mínimos significativos durante los sábados y domingos. Los días festivos también presentan variabilidad, pero generalmente muestran un mayor tráfico en comparación con los fines de semana.

4.5.2. AHT

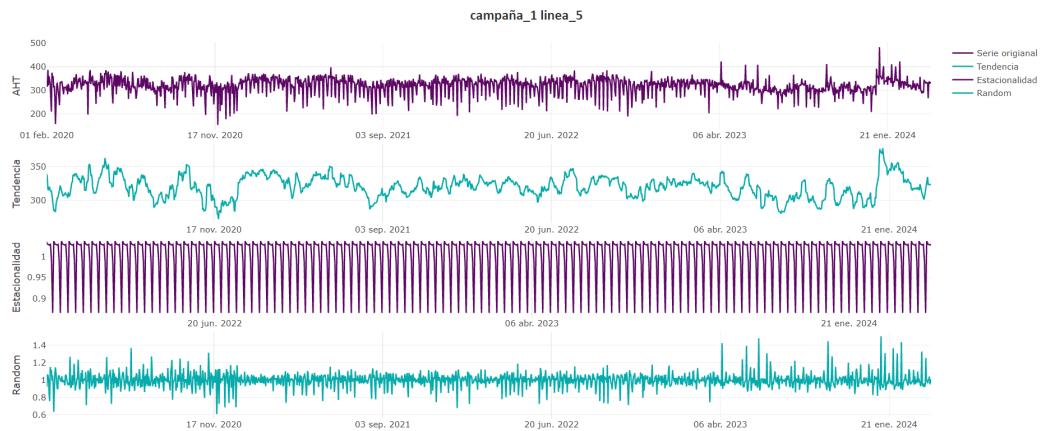


FIGURA 4.33: Línea 5-AHT: Descomposición diaria

En la Línea 5, el AHT diario muestra una tendencia estable con ligeras fluctuaciones, reflejando posibles variaciones en el tráfico de las llamadas. La estacionalidad es consistente, indicando un patrón regular en la duración de las llamadas a lo largo del tiempo. El componente aleatorio presenta algunos picos, lo que sugiere la presencia de eventos aislados o anomalías en el proceso.

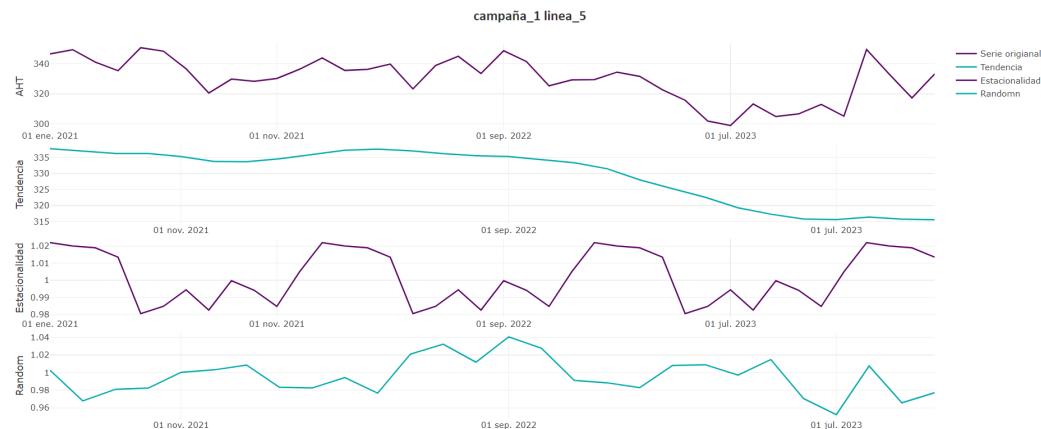


FIGURA 4.34: Línea 5-AHT: Descomposición mensual

Mensualmente, se presenta una ligera disminución a lo largo del tiempo, aunque con algunos picos y valles que indican fluctuaciones periódicas. La estacionalidad es baja, en relación a la diaria. El componente aleatorio muestra variabilidad moderada, con algunos picos que podrían corresponder a eventos específicos.

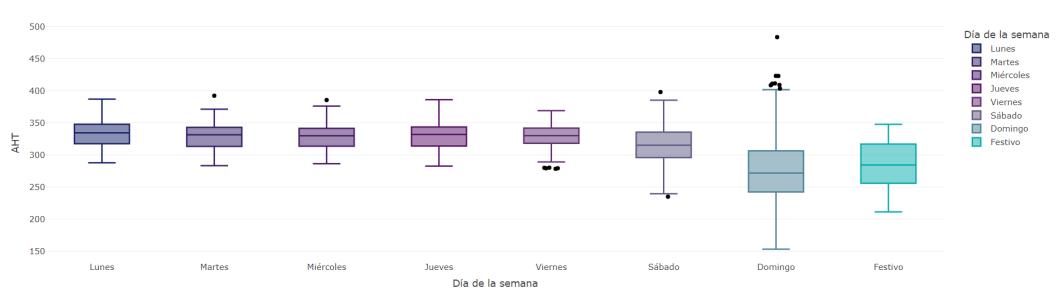


FIGURA 4.35: Línea 5- AHT: Box-plot día de la semana

Se observa que los lunes tienen una duración de llamadas medianamente alta y consistente. Los martes y miércoles muestran una ligera disminución, manteniéndose estables. Los jueves y viernes presentan una variabilidad moderada. Los sábados muestran mayor dispersión con algunos valores atípicos. Los domingos, aunque tienen una media relativamente baja, presentan una gran variabilidad, reflejando varios picos inusuales. Los días festivos muestran una variabilidad notable, similar a la de los domingos.

Capítulo 5

Modelos

5.1. Pycaret Regresión

PyCaret es una biblioteca de código abierto para el aprendizaje automático desarrollada en Python. Se caracteriza por ser de código bajo, lo que implica que los usuarios pueden ejecutar programas de aprendizaje automático con muchas menos líneas de código. Gracias a sus opciones de código reducido, se evita la necesidad de escribir cientos de líneas de código. [1]

El módulo de Regresión de PyCaret (`pycaret.regression`) es una herramienta de aprendizaje supervisado utilizada para predecir valores o resultados continuos mediante diversas técnicas y algoritmos. La regresión puede aplicarse para anticipar valores o resultados como ventas, unidades vendidas, temperaturas o cualquier número que sea continuo. Este modulo incluye más de 25 algoritmos y 10 gráficos para evaluar el rendimiento de los modelos. Ya sea para el ajuste de hiperparámetros, ensamblaje o técnicas avanzadas como el stacking.

El procedimiento o metodología empleada para realizar el modelado con esta herramienta fue el siguiente: configuración, comparación de modelos de regresión, ajuste de hiperparámetros y predicción.

Configuración

Para la configuración, se emplea la función `setup()`, que inicializa el entorno en PyCaret y crea la pipeline de transformación para preparar los datos para el modelado e implementación. Esta función cuenta con diversos parámetros, como `train_size`, `numeric_imputation`, `preprocess`, `remove_outliers`, y `fold_strategy`, entre otros, que permiten transformar y preprocesar los datos de manera efectiva.

En este proyecto, la configuración se llevó a cabo sin aplicar transformaciones ni preprocesamientos adicionales, dado que los datos ya habían sido preprocesados previamente, solo se separó el conjunto en `train` y `val`. Sin embargo, se estableció la estrategia de "foldz el número de pliegues para la validación cruzada, utilizando una técnica específica para series temporales conocida como rolling window". Esta técnica consiste en dividir los datos en múltiples subconjuntos o "ventanas" que se desplazan a lo largo del tiempo. En cada iteración, se entrena el modelo en una ventana de datos pasados y se evalúa en la siguiente ventana de datos futuros. Esta estrategia respeta la secuencia temporal de los datos, evitando el uso de información futura para predecir el pasado, lo que es crucial para mantener la integridad temporal en la modelización de series temporales.

Se tomó fold = 3, lo que consiste en entrenar y evaluar cada modelo en tres fases consecutivas. En cada fase, la ventana de datos utilizada para entrenar el modelo se expande para incluir más datos anteriores, mientras que la ventana de evaluación avanza al siguiente conjunto de datos.

Comparación de modelos

Para este paso, se utilizó la función `compare_models()`. Esta función entrena todos los modelos en la biblioteca de modelos y los califica en el conjunto de validación utilizando la validación cruzada rolling para la evaluación de métricas (WAPE). Los modelos comparados con sus parámetros iniciales, son:

- Linear Regression
- Lasso Regression
- Ridge Regression
- Elastic Net
- Least Angle Regression
- Lasso Least Angle Regression
- Orthogonal Matching Pursuit
- Bayesian Ridge
- Automatic Relevance Determination
- Passive Aggressive Regressor
- Random Sample Consensus
- TheilSen Regressor
- Huber Regressor
- Kernel Ridge
- Support Vector Regression
- K Neighbors Regressor
- Decision Tree Regressor
- Random Forest Regressor
- Extra Trees Regressor
- AdaBoost Regressor
- Gradient Boosting Regressor
- MLP Regressor
- Light Gradient Boosting Machine
- CatBoost Regressor

- Dummy Regressor

Ajuste de hiperparámetros

Para ajustar los hiperparámetros, se utilizó la función `tune_model()`. Esta función ajusta automáticamente los hiperparámetros de un modelo en un espacio de búsqueda predefinido y lo califica mediante la validación cruzada y escoge los mejores parámetros teniendo en cuenta la métrica de error (WAPE).

Predicción

Por último, se realiza la predicción en el conjunto de prueba. Este paso es crucial para validar que el modelo funciona adecuadamente con datos que no ha visto durante el entrenamiento, asegurando así su aplicabilidad en las predicciones futuras.

5.2. Pycaret Time Series

Esta herramienta aplica la misma metodología que PyCaret Regression, pero está adaptada para trabajar con series temporales utilizando los siguientes modelos:

- ARIMA
- Auto ARIMA
- Croston
- Linear w/ Cond. Deseasonalize & Detrending
- Elastic Net w/ Cond. Deseasonalize & Detrending
- Ridge w/ Cond. Deseasonalize & Detrending
- Lasso w/ Cond. Deseasonalize & Detrending
- Lasso Least Angular Regressor w/ Cond. Deseasonalize & Detrending
- Bayesian Ridge w/ Cond. Deseasonalize & Detrending
- Huber w/ Cond. Deseasonalize & Detrending
- Orthogonal Matching Pursuit w/ Cond. Deseasonalize & Detrending
- K Neighbors w/ Cond. Deseasonalize & Detrending
- Decision Tree w/ Cond. Deseasonalize & Detrending
- Random Forest w/ Cond. Deseasonalize & Detrending
- Extra Trees w/ Cond. Deseasonalize & Detrending
- Gradient Boosting w/ Cond. Deseasonalize & Detrending
- AdaBoost w/ Cond. Deseasonalize & Detrending
- Light Gradient Boosting w/ Cond. Deseasonalize & Detrending
- CatBoost Regressor w/ Cond. Deseasonalize & Detrending

Para este proyecto, además se calculó el promedio de los resultados obtenidos por los modelos PyCaret Time Series y PyCaret Regression. Observamos que uno de los modelos ajustaba mejor la altura de la serie de tiempo y el otro la tendencia, por lo que, al combinarlos, se obtenían a menudo mejores resultados. Por ello, este modelo promedio, denominado Promedio TS-R, también se incluyó en la comparación de modelos.

5.3. Facebook Prophet

Facebook Prophet es una herramienta de pronóstico desarrollada internamente por Facebook, diseñada para abordar ciertos desafíos que surgían con las metodologías de pronóstico tradicionales. A menudo, las herramientas automáticas de pronóstico disponibles en el mercado resultaban ser demasiado rígidas, dificultando la inclusión de suposiciones adicionales, o bien, las herramientas más robustas requerían un analista experimentado con habilidades especializadas en ciencia de datos. Para atender la alta demanda de pronósticos comerciales de alta calidad, Facebook decidió liberar Prophet como software de código abierto en 2017.

Características y Capacidades

Prophet está diseñado específicamente para manejar tareas de pronóstico empresarial, las cuales suelen presentar las siguientes características:

- **Datos de series temporales:** Capturados a nivel horario, diario o semanal, con preferencia por series que incluyan al menos un año completo de datos históricos.
- **Efectos de estacionalidad fuertes:** Que ocurren a diario, semanal o anualmente.
- **Días festivos y otros eventos especiales únicos:** Que no siguen necesariamente los patrones de estacionalidad, ocurriendo de manera irregular.
- **Datos faltantes y valores atípicos:** Comunes en muchos conjuntos de datos históricos.
- **Cambios significativos en la tendencia:** Que pueden ocurrir, por ejemplo, con el lanzamiento de nuevas características o productos.
- **Tendencias asintóticas:** Que se acercan a un límite superior o inferior.

En esencia, Prophet es un modelo de regresión aditiva. Esto significa que el modelo es simplemente la suma de varios componentes (opcionales), que pueden incluir:

- Una curva de crecimiento lineal o logística.
- Una curva de estacionalidad anual.
- Una curva de estacionalidad semanal.
- Una curva de estacionalidad diaria.
- Días festivos y otros eventos especiales.

- Curvas de estacionalidad adicionales definidas por el usuario, tales como curvas estacionales horarias o trimestrales.

Este enfoque permite que Prophet maneje eficazmente la complejidad de los patrones de series temporales, ofreciendo pronósticos que son tanto precisos como fáciles de interpretar y ajustar según las necesidades específicas de cada negocio[12].

El modelo se expresa matemáticamente de la siguiente manera:[13]

$$y(t) = g(t) + s(t) + h(t) + \epsilon_t, \quad (5.1)$$

donde:

- $g(t)$ es la función de tendencia que modela los cambios no periódicos.
- $s(t)$ es la estacionalidad que representa cambios periódicos (diarios, semanales, anuales).
- $h(t)$ captura los efectos de los días festivos que ocurren en fechas específicas del calendario.
- ϵ_t es el término de error que captura las variaciones no modeladas por los otros componentes.

Para la implementación de este modelo, inicialmente, los datos históricos se dividieron en tres subconjuntos: entrenamiento, validación y prueba. El conjunto de entrenamiento se utilizó para ajustar el modelo, mientras que el conjunto de validación ayudó a ajustar los hiperparámetros del modelo. El conjunto de prueba, por su parte, sirvió para evaluar la precisión del modelo en la predicción de períodos futuros. Además, se incluyeron los festivos de cada línea.

Posteriormente, se definió un espacio de búsqueda de hiperparámetros, que incluyó la consideración de diversos aspectos como el tipo de crecimiento de la tendencia, la inclusión de estacionalidades anual, semanal y diaria, y la configuración de diferentes escalas de regularización para los puntos de cambio y la estacionalidad. Este espacio de búsqueda permitió explorar diversas configuraciones del modelo para encontrar la más adecuada para los datos específicos.

El ajuste de los hiperparámetros se llevó a cabo mediante un proceso de búsqueda aleatoria, en el cual se probaron múltiples combinaciones de parámetros. Cada combinación fue evaluada utilizando el WAPE. El objetivo fue minimizar este error para identificar la configuración óptima del modelo.

Una vez identificados los mejores hiperparámetros, se entrenó el modelo final utilizando todo el conjunto de datos de entrenamiento. Finalmente, se realizó el pronóstico sobre el conjunto de prueba para evaluar la capacidad predictiva del modelo, obteniendo así las predicciones finales para los períodos futuros.

Este enfoque basado en Prophet permitió aprovechar sus capacidades para capturar tendencias no lineales, efectos estacionales y días festivos.

5.4. Gated recurrent unit (GRU)

El modelo GRU, o Gated Recurrent Unit, es una variante de las redes neuronales recurrentes (RNN) diseñada para abordar algunas de las limitaciones comunes en las RNN tradicionales, como el problema del desvanecimiento del gradiente, que afecta la capacidad de estas redes para aprender dependencias a largo plazo en secuencias de datos. Introducido por Cho et al. en 2014 [3], el GRU es una simplificación de la arquitectura LSTM (Long Short-Term Memory), pero con un rendimiento similar en muchas tareas de modelado secuencial, como la predicción de series temporales, procesamiento de lenguaje natural y análisis de secuencias en general.[14]

Arquitectura del GRU

La arquitectura de un GRU combina los estados de la celda y el estado oculto en una sola estructura, simplificando así la red en comparación con las LSTM. A diferencia de las LSTM, que utilizan tres puertas (puerta de entrada, puerta de olvido y puerta de salida), los GRU emplean solo dos puertas: la puerta de actualización y la puerta de reinicio.

- **Puerta de actualización:** Controla cuánto de la información pasada debe llevarse al futuro, decidiendo la cantidad de información previa que se mantiene y la nueva información que se incorpora. Esta puerta es crucial para la conservación de la memoria a largo plazo en la secuencia.
- **Puerta de reinicio:** Decide cuánta parte de la información anterior debe olvidarse en el cálculo de la salida actual. Esta puerta permite a la red olvidar estados previos que no son relevantes para la predicción actual, haciendo que la red sea más eficiente en términos de aprendizaje de dependencias a corto plazo.

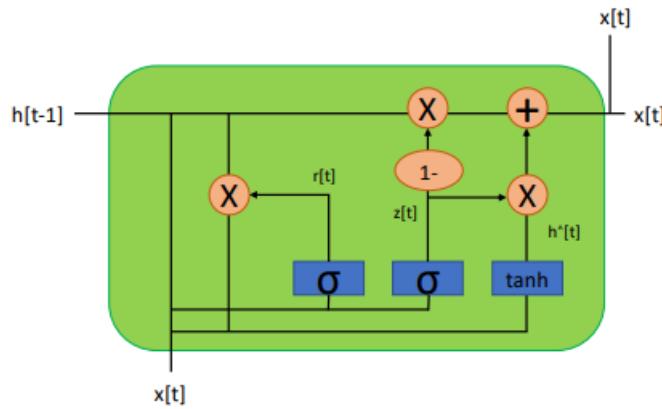


FIGURA 5.1: Arquitectura GRU[11]

Matemáticamente, las operaciones de un GRU se pueden describir como sigue:[4]

- **Cálculo de la puerta de actualización:**

$$z_t = \sigma(W_z x_t + U_z h_{t-1} + b_z)$$

donde z_t es la puerta de actualización, W_z , U_z son los pesos respectivos, x_t es la entrada actual, h_{t-1} es el estado oculto anterior, y b_z vector que se debe aprender durante el entrenamiento.

- **Cálculo de la puerta de reinicio:**

$$r_t = \sigma(W_r x_t + U_r h_{t-1} + b_r)$$

donde r_t es la puerta de reinicio y W_r es el peso asociado.

- **Cálculo del nuevo vector de activación candidato:**

$$\tilde{h}_t = \tanh(W_h x_t + U_h (r_t \odot h_{t-1}) + b_h)$$

donde \tilde{h}_t es el estado candidato.

- **Cálculo del vector de salida:**

$$h_t = (1 - z_t) \odot h_{t-1} + z_t \odot \tilde{h}_t$$

donde h_t es el vector de salida, que se utiliza para predecir la salida en el tiempo t .

En la implementación del modelo GRU, el proceso comienza dividiendo los datos en conjuntos de entrenamiento y prueba en función de fechas específicas. Para ello, se ajusta la fecha de inicio del conjunto de prueba para asegurar que el modelo tenga acceso a un número suficiente de datos anteriores (n_step=14)) para entrenar adecuadamente la secuencia temporal. Los datos seleccionados se normalizan usando StandardScaler para garantizar que las características estén en una escala comparable, lo que facilita el entrenamiento del modelo.

Posteriormente, se crean secuencias de datos normalizados que el modelo GRU utilizará. Este modelo se construye y optimiza mediante una búsqueda de hiperparámetros con la función kt.RandomSearch. El modelo incluye capas GRU con unidades configurables, seguidas de capas densas para refinar la salida. El WAPE se emplea como métrica de evaluación, enfocándose en minimizarlo. La búsqueda de hiperparámetros considera múltiples configuraciones para encontrar la arquitectura óptima.

Finalmente, el modelo se entrena utilizando los mejores hiperparámetros encontrados, y las predicciones se realizan sobre el conjunto de prueba. Las predicciones se desnormalizan para interpretar los resultados en su escala original. Este enfoque asegura que el modelo esté bien ajustado a los patrones temporales presentes en los datos y que las predicciones sean precisas y útiles.

Se decidió utilizar este modelo para evaluar el rendimiento de una red neuronal y compararlo con los otros modelos. El modelo GRU fue elegido por su simplicidad en comparación con las LSTM, manteniendo una alta capacidad para aprender dependencias a largo plazo en secuencias de datos. Además, su menor complejidad computacional lo hace más rápido de entrenar y menos exigente en recursos.

Capítulo 6

Resultados

6.1. Línea 1

6.1.1. Tráfico de llamadas

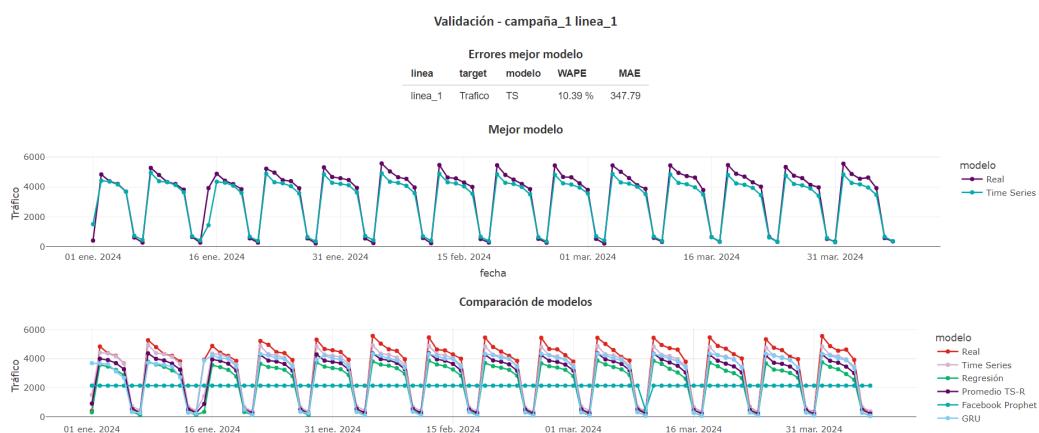


FIGURA 6.1: Línea 1-Llamadas: Validación diaria

El mejor modelo identificado, utilizando la herramienta Pycaret Time Series, mostró un WAPE del 10.39 % y un MAE de 347.79, lo que indica una alta precisión en la predicción del tráfico de llamadas. Sin embargo, es importante destacar que el modelo falló notablemente el 15 de enero, debido a que en la base de datos esta fecha estaba marcada como festivo. A pesar de esto, la línea experimentó un tráfico similar al de un lunes normal, lo cual es inusual y contribuyó al error en la predicción.

De manera específica, el mejor modelo encontrado por Pycaret fue un ARIMA(order=(0, 0, 0), seasonal_order=(1, 0, 1, 7)) con los siguientes parámetros:

- 'concentrate_scale': False
- 'enforce_invertibility': True
- 'enforce_stationarity': True
- 'hamilton_representation': False
- 'maxiter': 50
- 'measurement_error': False
- 'method': 'lbfgs'

- 'mle_regression': True
- 'order': (0, 0, 0)
- 'out_of_sample_size': 0
- 'scoring': 'mse'
- 'scoring_args': None
- 'seasonal_order': (1, 0, 1, 7)
- 'simple_differencing': False
- 'start_params': None
- 'suppress_warnings': False
- 'time_varying_regression': False
- 'trend': None
- 'with_intercept': True

Para realizar el pronóstico mensual, se sumaron los resultados de los pronósticos diarios, obteniendo los siguientes resultados:

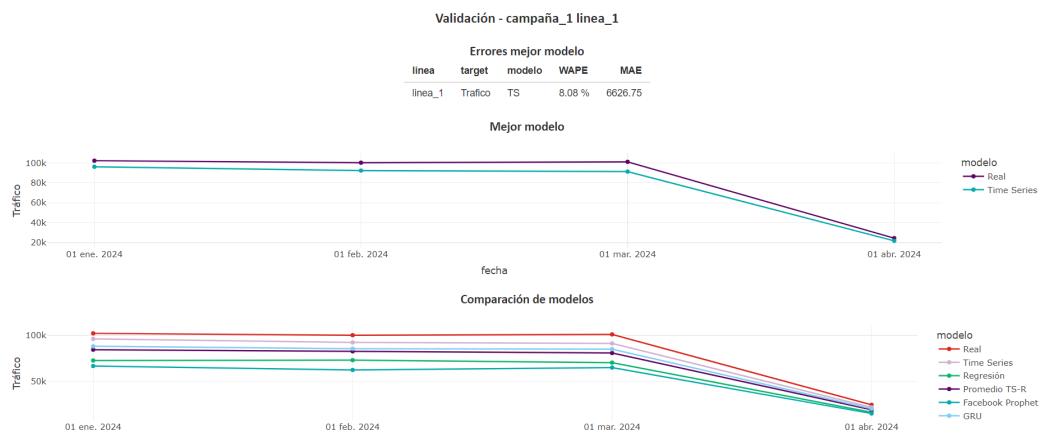


FIGURA 6.2: Línea 1-Llamadas: Validación mensual

Se muestra que el modelo anterior es el más preciso, con un WAPE del 8.08 % y un MAE de 6626.75. Este modelo logró capturar eficazmente las tendencias mensuales, como se observa en la gráfica, donde las predicciones se ajustan muy bien a los datos reales. Esto sugiere que este enfoque es altamente adecuado para la planificación mensual de recursos, proporcionando predicciones confiables que pueden respaldar la toma de decisiones estratégicas.

A partir, de este mejor modelo, pronosticamos las llamadas que se recibirán desde el 8 de abril hasta el 31 de diciembre del 2024.

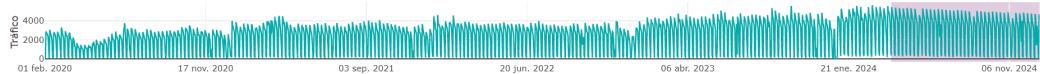


FIGURA 6.3: Línea 1-Llamadas: Predicción diaria



FIGURA 6.4: Línea 1-Llamadas: Predicción mensual

6.1.2. AHT

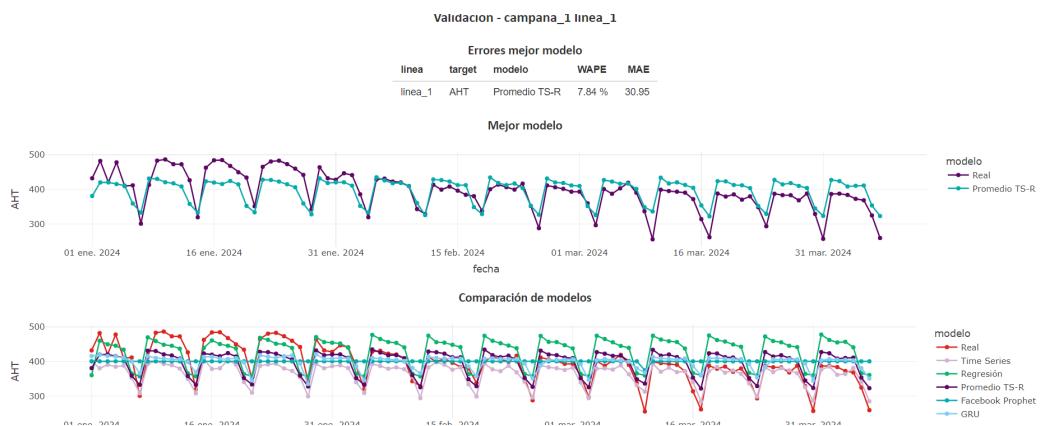


FIGURA 6.5: Línea1-AHT: Validación diaria

Los resultados obtenidos para el AHT en la línea 1 muestran que el modelo basado en el promedio de los resultados de los modelos de Pycaret TS y Pycaret Regresión (Promedio TS-R) es el más efectivo, con un WAPE de 7.84 % y un MAE de 30.95. Como se observa en la gráfica superior, este modelo logra capturar con precisión las fluctuaciones diarias del AHT, manteniéndose cercano a los valores reales.

Los modelos promediados fueron, GradientBoostingRegressor con los siguientes parámetros:

- 'learning_rate'=0.16867803151940922
- 'max_depth'=2
- 'max_features'=1.0
- 'min_impurity_decrease'=2.69946419220041e-05
- 'min_samples_leaf'=2
- 'min_samples_split'=3
- 'n_estimators'=224
- 'random_state'=47
- 'subsample'=1

y OrthogonalMatchingPursuit con los siguientes parámetros:

- 'fit_intercept': False
- 'n_nonzero_coefs': 3,
- 'precompute': 'auto',
- 'tol': None

Mensual, obtenemos los siguientes resultados:

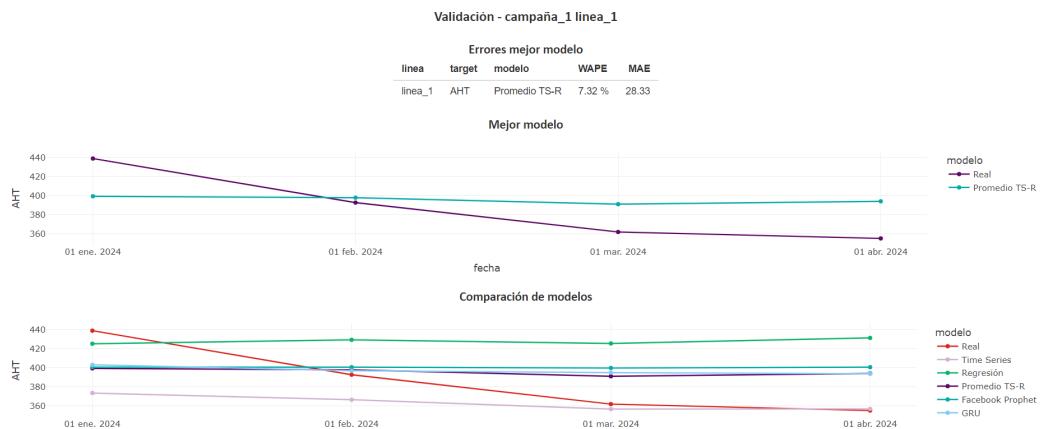


FIGURA 6.6: Línea1-AHT: Validación mensual

Se obtiene un WAPE de 7.32 % y un MAE de 28.33. Como se observa en la gráfica, este modelo captura bien las variaciones mensuales del AHT, manteniéndose cercano a los valores reales, aunque no toma muy bien los valores muy altos o bajos.

Con lo cual, se obtienen los siguientes pronósticos para 2024,

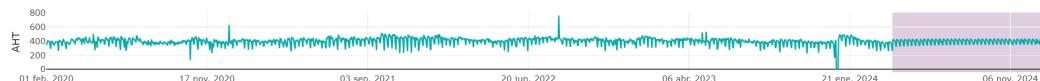


FIGURA 6.7: Línea 1-AHT: Predicción diaria

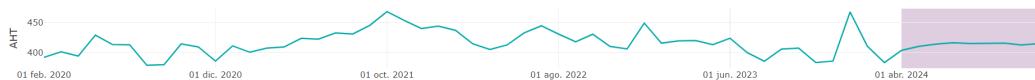


FIGURA 6.8: Línea 1-AHT: Predicción mensual

6.2. Línea 2

6.2.1. Tráfico de llamadas

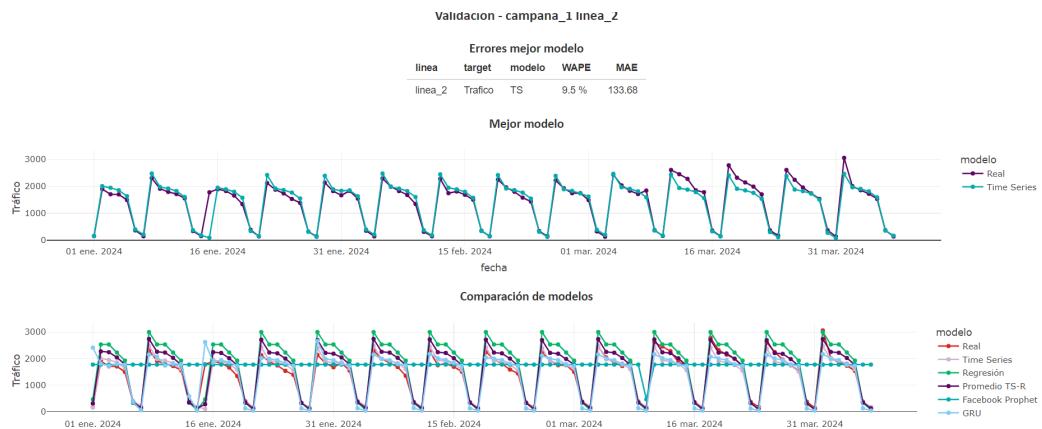


FIGURA 6.9: Línea 2-Llamadas: Validación diaria

El modelo más preciso, identificado utilizando Pycaret Time Series, logró un WAPE del 9.5 % y un MAE de 133.68, lo que refleja su capacidad para seguir de cerca las variaciones diarias del tráfico. A demás, se observa que igual a la línea 1, el 15 de enero marcado como festivo, tiene comportamiento como un lunes normal, lo que genera un error en la predicción.

De manera específica, el mejor modelo encontrado por Pycaret fue un ARIMA($\text{order}=(0, 0, 0)$, $\text{seasonal_order}=(0, 1, 0, 7)$) con los siguientes parámetros:

- 'concentrate_scale': False
- 'enforce_invertibility': True
- 'enforce_stationarity': True
- 'hamilton_representation': False
- 'maxiter': 50
- 'measurement_error': False
- 'method': 'lbfgs'
- 'mle_regression': True
- 'order': (0, 0, 0)
- 'out_of_sample_size': 0
- 'scoring': 'mse'
- 'scoring_args': None
- 'seasonal_order': (0, 1, 0, 7)
- 'simple_differencing': False
- 'start_params': None

- 'suppress_warnings': False
- 'time_varying_regression': False
- 'trend': None
- 'with_intercept': True

Para la validación mensual, se obtienen los siguientes resultados:

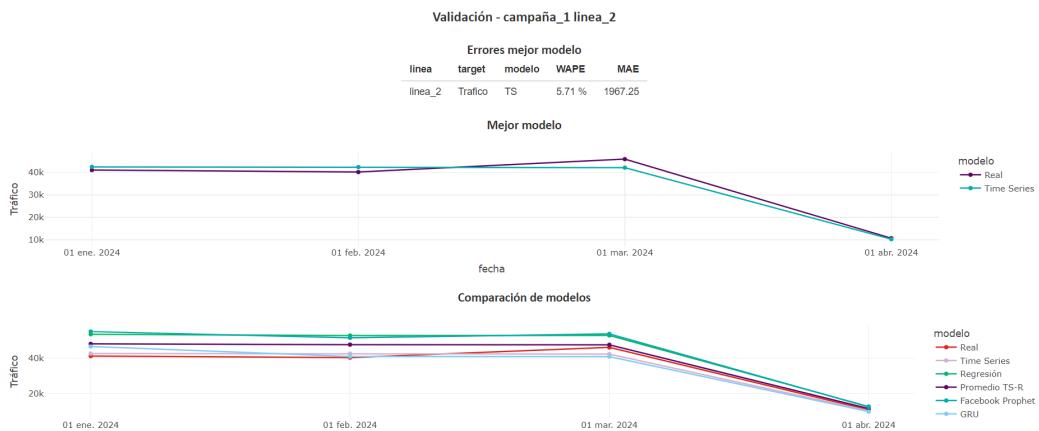


FIGURA 6.10: Línea 2-Llamadas: Validación mensual

Se observa un WAPE del 5.71 % y un MAE de 1967.25 y vemos qye la predicción se ajusta notoriamente, con lo cual observamos que en el mes de enero completo se compensa el error del 15 festivo.

Con lo que se obtienen los siguientes pronósticos futuros,

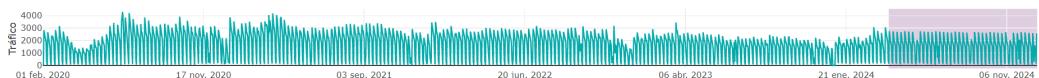


FIGURA 6.11: Línea 2-Llamadas: Predicción diaria



FIGURA 6.12: Línea 2-Llamadas: Predicción mensual

6.2.2. AHT

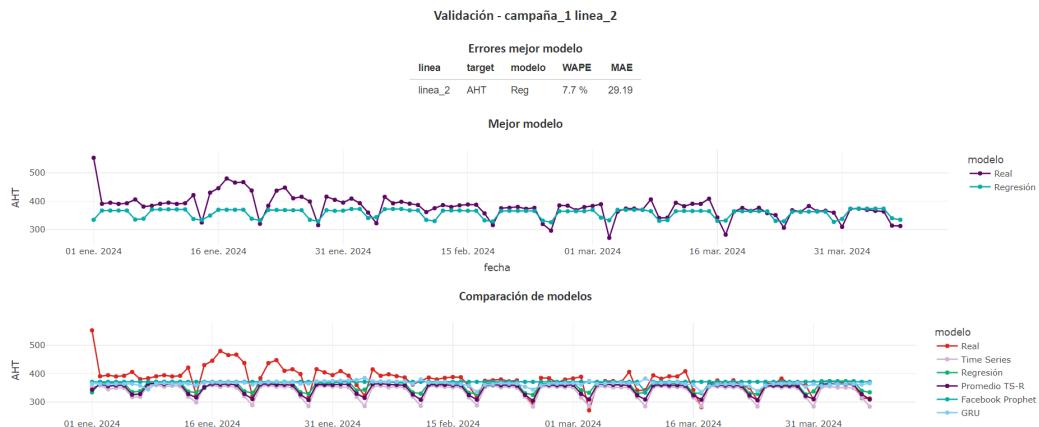


FIGURA 6.13: Línea2-AHT: Validación diaria

Para la validación diaria del AHT en la línea 2, el mejor modelo identificado fue utilizando Pycaret Regresion, que obtuvo un WAPE del 7.7 % y un MAE de 29.19. Este modelo demostró ser altamente preciso en la predicción del AHT, como se observa en la gráfica, donde sigue de cerca las fluctuaciones reales del tiempo medio de manejo. Sin embargo, no logró captar la subida del 15 al 20 de enero, que puede catalogarse como inusual.

Específicamente, el mejor modelo fue CatBoostRegressor con los siguientes parámetros:

- 'depth': 4
- 'l2_leaf_reg': 3
- 'max_features'=1.0
- "loss_function": 'RMSE'
- 'border_count': 254
- 'random_strength': 0.4
- 'n_estimators'=224
- 'n_estimators': 20
- "eta": 0.5

Mensual, obtenemos los siguientes resultados:



FIGURA 6.14: Línea2-AHT: Validación mensual

Se obtiene un WAPE de 9.5 % y un MAE de 35.53. Obteniendo los siguientes valores futuros:

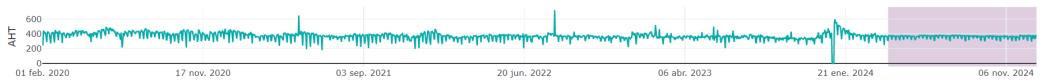


FIGURA 6.15: Línea 2-AHT: Predicción diaria



FIGURA 6.16: Línea 2-AHT: Predicción mensual

6.3. Línea 3

6.3.1. Tráfico de llamadas

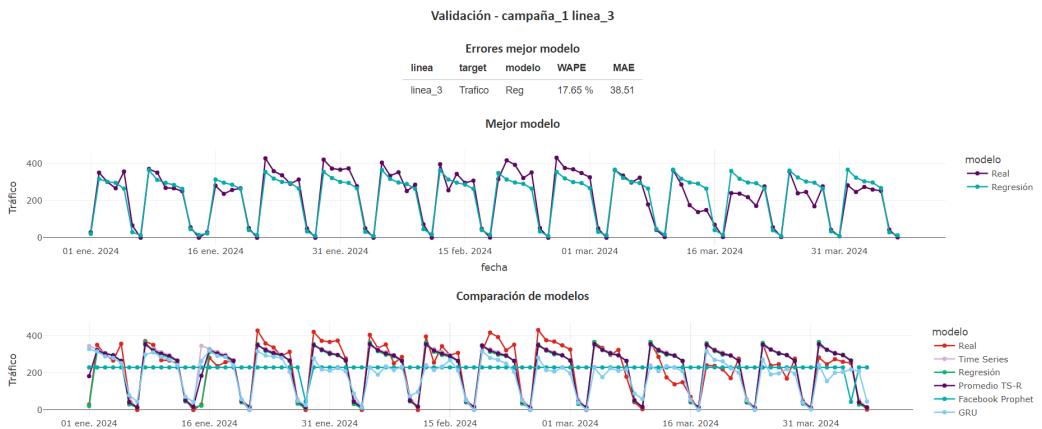


FIGURA 6.17: Línea 3-Llamadas: Validación diaria

El mejor modelo fue el modelo hallado con Pycaret Regresión, con un WAPE del 17.65% y un MAE de 38.51. Aunque el modelo sigue relativamente bien las tendencias generales del tráfico, se observan ciertas desviaciones en comparación con los valores reales, especialmente en días con fluctuaciones más pronunciadas.

De manera específica, el mejor modelo encontrado por Pycaret fue un AdaBoostRegressor con los siguientes parámetros:

- 'estimator': None
- 'learning_rate': 0.3
- 'loss': 'square'
- 'n_estimators': 280

Para la validación mensual, se obtienen los siguientes resultados:



FIGURA 6.18: Línea 3-Llamadas: Validación mensual

Se observa un WAPE del 11.58% y un MAE de 619, observando que se compensa un poco los errores de las predicciones diarias.

Con lo que se obtienen los siguientes pronósticos futuros,



FIGURA 6.19: Línea 3-Llamadas: Predicción diaria



FIGURA 6.20: Línea 3-Llamadas: Predicción mensual

6.3.2. AHT

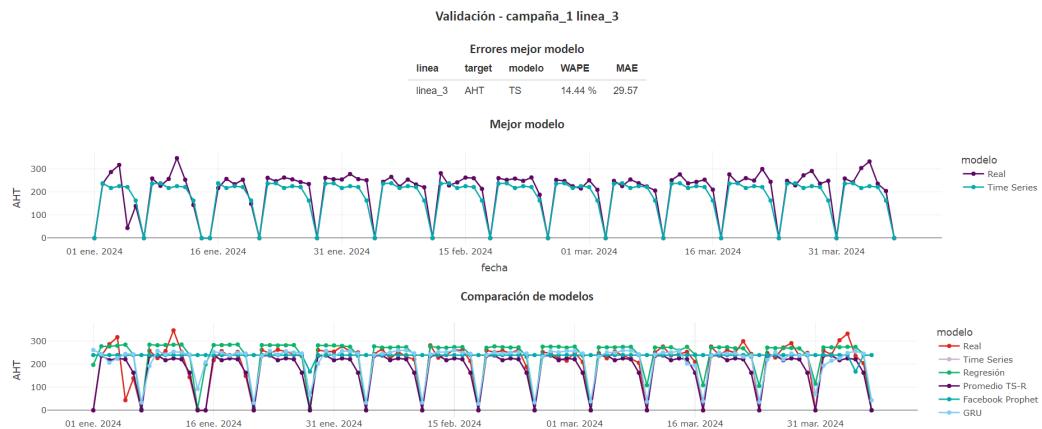


FIGURA 6.21: Línea 3-AHT: Validación diaria

En la validación diaria del AHT para la línea 3, el mejor modelo identificado fue el hallado con Pycaret TS, con un WAPE del 14.44 % y un MAE de 29.57. El modelo sigue de cerca las variaciones diarias del AHT, reflejando una capacidad adecuada para captar las fluctuaciones dentro de la línea, aunque con algunas discrepancias observables en los días con mayores variaciones.

De manera específica, el mejor modelo encontrado por Pycaret fue un BaseCdsDtForecaster con los siguientes parámetros:

- 'degree': 1
- 'deseasonal_model': 'additive'
- 'fe_target_rr': [WindowSummarizer(lag_feature='lag': [7, 6, 5, 4, 3, 2, 1], n_jobs=1)]
- "regressor": LinearRegression(n_jobs=-1)
- 'window_length': 7
- "regressor_copy_X": True
- 'max_samples': None
- 'regressor_fit_intercept': True
- 'regressor_positive': False

Para la validación mensual, se obtienen los siguientes resultados:

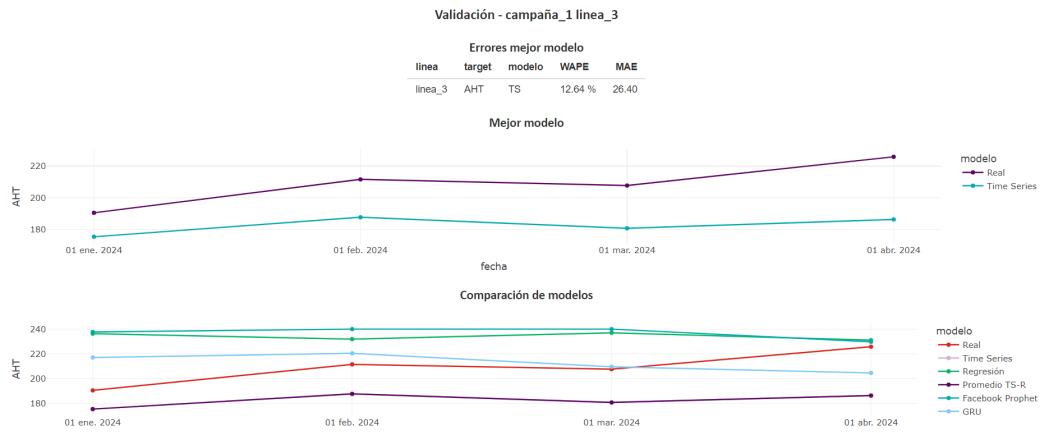


FIGURA 6.22: Línea 3-AHT: Validación mensual

Se observa un WAPE del 12.64 % y un MAE de 26.40, observando que se compensa un poco los errores de las predicciones diarias.

Con lo que se obtienen los siguientes pronósticos futuros,

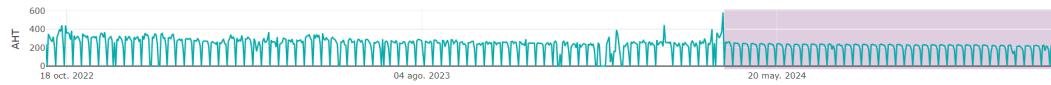


FIGURA 6.23: Línea 3-AHT: Predicción diaria

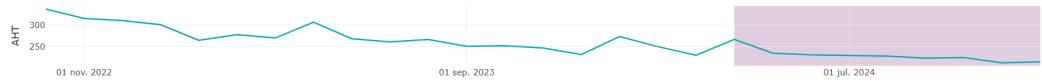


FIGURA 6.24: Línea 3-AHT: Predicción mensual

6.4. Línea 4

6.4.1. Tráfico de llamadas

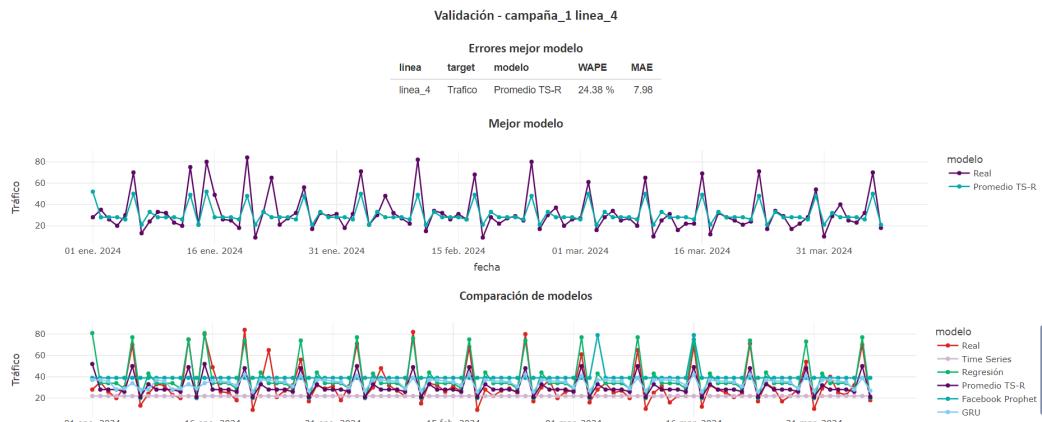


FIGURA 6.25: Línea 4-Llamadas: Validación diaria

En la validación diaria del tráfico de llamadas para la línea 4, el mejor desempeño se obtuvo con el promedio entre los modelos de Pycaret Time Series y Pycaret Regresión, alcanzando un WAPE del 24.38 % y un MAE de 7.98. Es importante destacar que esta serie temporal cuenta con datos históricos desde marzo de 2023, lo que significa que el limitado historial disponible podría ser una de las razones detrás de las predicciones menos precisas. La falta de un historial más amplio limita la capacidad del modelo para identificar patrones robustos, lo que afecta la exactitud de las proyecciones.

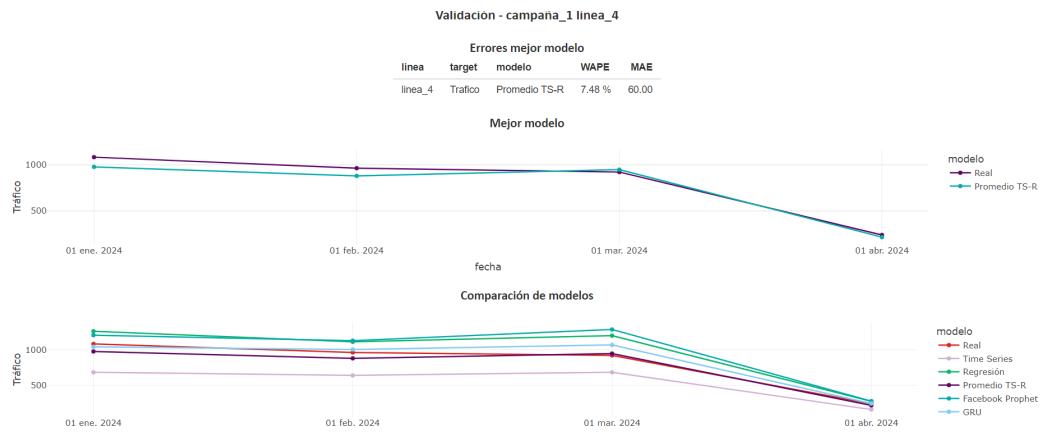


FIGURA 6.26: Línea 4-Llamadas: Validación mensual

Sin embargo, es importante destacar que estos errores en las predicciones diarias tienden a compensarse cuando se consideran a nivel mensual, resultando en predicciones mensuales más precisas.

Los modelos utilizados para hacer el promedio son: AdaBoostRegressor con los siguientes parámetros,

- 'estimator': None,
- 'learning_rate': 0.005,
- 'loss': 'square',
- 'n_estimators': 150,

y Croston con 'smoothing': 0.8891318935804967.

Con lo que se obtienen las siguientes predicciones futuras

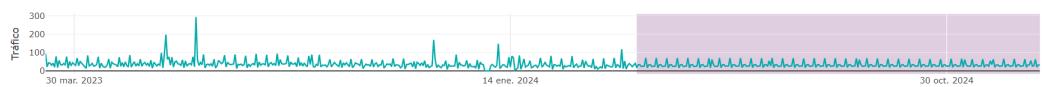


FIGURA 6.27: Línea 4-Llamadas: Predicción diaria

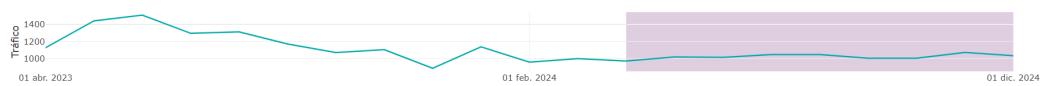
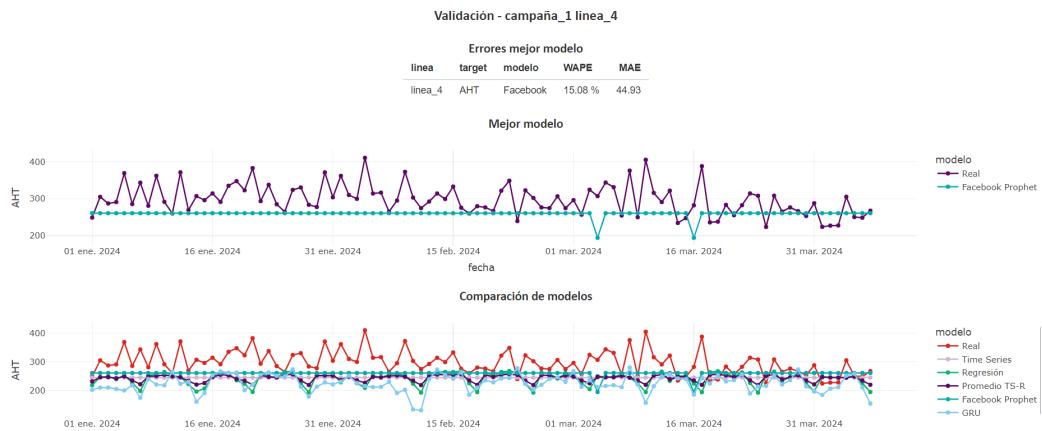


FIGURA 6.28: Línea 4-Llamadas: Predicción mensual

6.4.2. AHT



En la validación diaria del AHT para la línea 4, el modelo con mejor desempeño fue Facebook Prophet, obteniendo un WAPE del 15.08 % y un MAE de 44.93. No obstante, aunque Prophet mantiene cierta consistencia, no captura adecuadamente las fluctuaciones observadas en los datos reales, como se muestra en la gráfica superior. Este comportamiento puede explicarse por la limitada cantidad de datos históricos disponibles, con registros que comienzan en 2022. Además, es importante señalar que desde 2024 se observa un notable incremento en la altura de la serie temporal, lo que sugiere un posible cambio operacional en la línea. Tal cambio es extremadamente difícil de prever por cualquier modelo, ya que no existían patrones previos que lo anticiparan. Esta combinación de un historial reducido y un cambio abrupto en la tendencia complica la capacidad de Prophet para generar predicciones precisas.

Los parámetros utilizados para el modelo de Facebook Prophet fueron los siguientes:

- 'changepoint_prior_scale': 0.011348847189364953
- 'changepoint_range': 0.6279962156489811
- 'daily_seasonality': True
- 'growth': 'flat'
- 'interval_width': 0.6294926339715619
- 'n_changepoints': 45
- 'seasonality_mode': 'additive'
- 'seasonality_prior_scale': 19.95612477805556
- 'uncertainty_samples': 1000
- 'weekly_seasonality': False
- 'yearly_seasonality': False

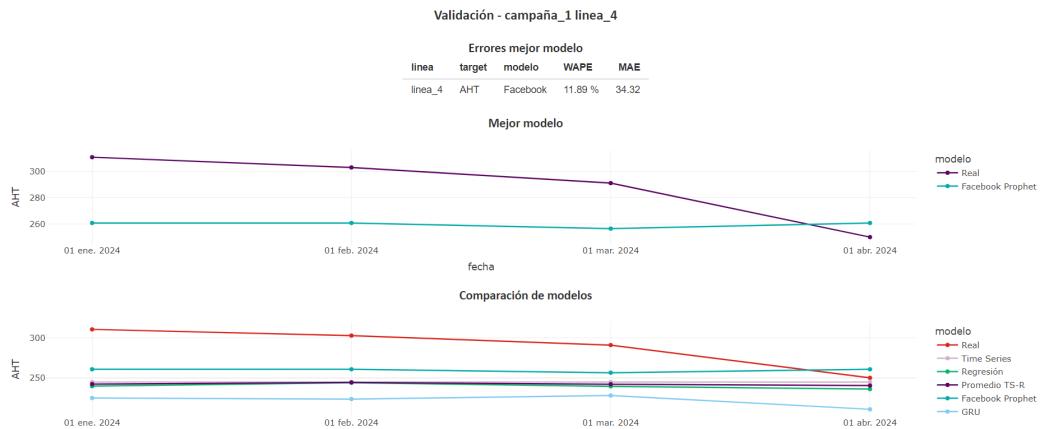


FIGURA 6.30: Línea 4-AHT: Validación mensual

Mensualmente,
obtenemos un WAPE del 11.89 % y un MAE de 34.32, con lo que observamos
que se compensan un poco los errores diarios.

Con lo que obtenemos las siguientes predicciones,



FIGURA 6.31: Línea 4-AHT: Predicción diaria

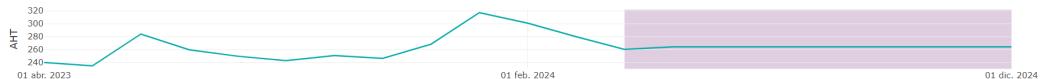


FIGURA 6.32: Línea 4-AHT: Predicción mensual

Estas predicciones casi constantes pueden atribuirse a la falta de un historial de datos más extenso y a la inestabilidad inherente de la serie temporal. La combinación de estos factores limita la capacidad del modelo para identificar y aprender patrones significativos, lo que resulta en proyecciones que no reflejan adecuadamente las fluctuaciones reales de la serie. Sin embargo, es importante destacar que esta tendencia a mantenerse en una predicción constante tiene una ventaja: al ubicarse en el centro, este enfoque genera menores errores en comparación con otros modelos que tienden a desviarse significativamente hacia arriba o hacia abajo. Al evitar estas desviaciones extremas, el modelo logra mantener un nivel de error más controlado, lo que puede ser preferible en situaciones donde la estabilidad y la previsibilidad son más valoradas que la precisión en capturar cada fluctuación.

6.5. Línea 5

6.5.1. Tráfico de llamadas

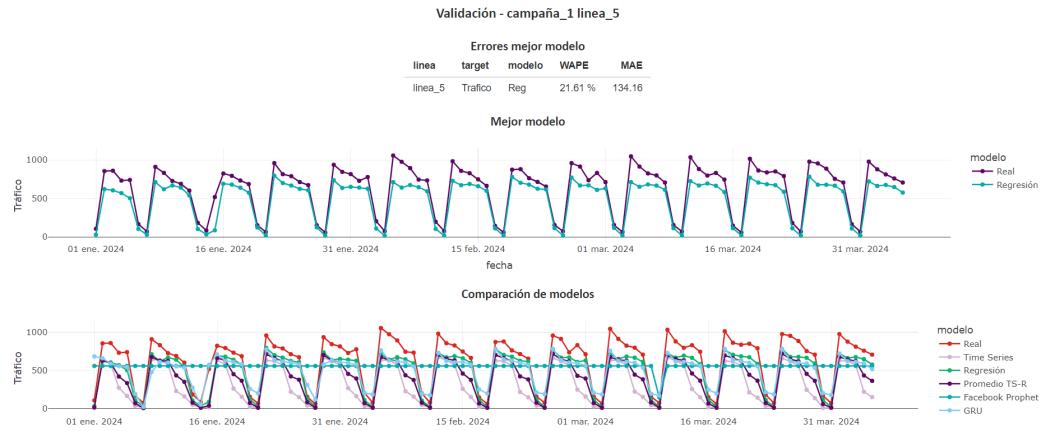


FIGURA 6.33: Línea 5-Llamadas: Validación diaria

En la validación diaria del tráfico de llamadas para la línea 5, el modelo de regresión fue el que presentó el mejor desempeño, alcanzando un WAPE del 21.61 % y un MAE de 134.16. Aunque este modelo logra capturar las tendencias generales del tráfico, se observan algunas desviaciones respecto a los valores reales, especialmente en los picos de mayor tráfico.

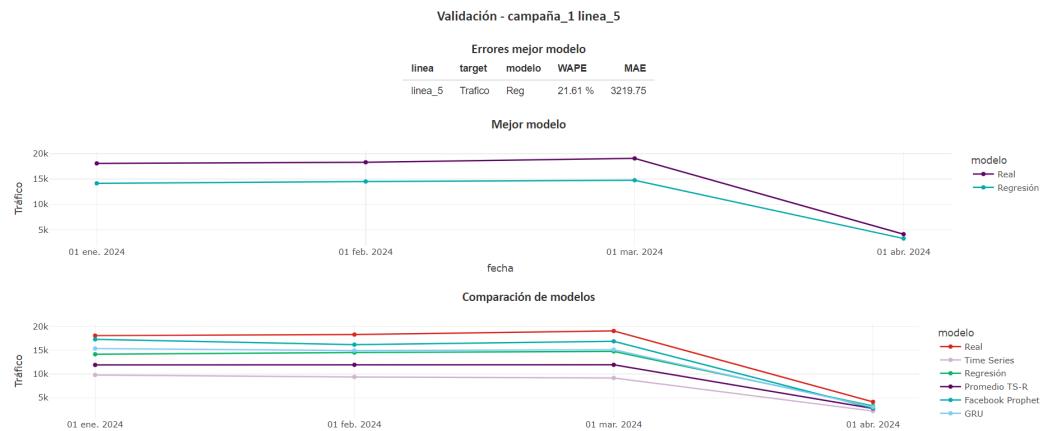


FIGURA 6.34: Línea 5-Llamadas: Validación mensual

Las desviaciones en los picos altos afectan significativamente la precisión mensual del modelo, ya que este tiende a quedarse por debajo de los valores reales de la serie original. Esto se debe a que el modelo de regresión, aunque eficaz para suavizar las fluctuaciones, no logra capturar completamente los picos de mayor tráfico. Como resultado, las predicciones mensuales tienden a subestimar el tráfico total.

El modelo específico utilizado para realizar la predicción es: ExtraTreesRegressor con los siguientes parámetros,

- 'bootstrap': False
- 'ccp_alpha': 0.0

- 'criterion': 'squared_error'
- 'max_depth': None
- 'max_features': 1.0
- 'max_leaf_nodes': None
- "max_samples": None
- 'min_impurity_decrease': 0.0
- 'min_samples_leaf': 1
- 'min_samples_split': 2
- 'min_weight_fraction_leaf': 0.0
- 'monotonic_cst': None
- 'n_estimators': 100

Con lo que se obtienen las siguientes predicciones futuras

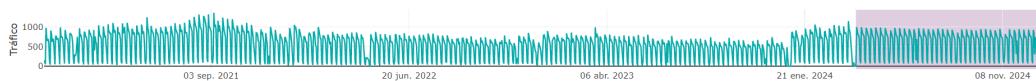


FIGURA 6.35: Línea 5-Llamadas: Predicción diaria



FIGURA 6.36: Línea 5-Llamadas: Predicción mensual

6.5.2. AHT

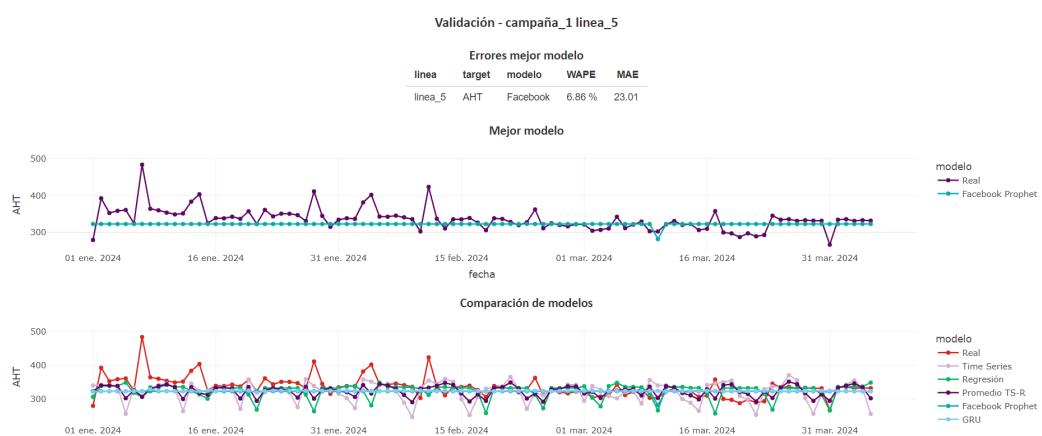


FIGURA 6.37: Línea 5-AHT: Validación diaria

Para la validación diaria del AHT para la línea 5, el modelo con mejor desempeño fue Facebook Prophet, obteniendo un WAPE del 4.61 % y un MAE de 15.44. Al

igual que la línea 4, desde 2024 se observa un notable incremento en la altura de la serie temporal, lo que sugiere un posible cambio operacional en la línea. Aunque en esta línea no se ve afectada en gran medida por este cambio abrupto, no podemos garantizar que el modelo genere una línea base estable, que no es capaz de reflejar las fluctuaciones y picos reales que aparecen en los datos, que si bien minimiza grandes errores, puede resultar en subestimaciones significativas de las fluctuaciones reales.

Los parámetros utilizados para el modelo de Facebook Prophet fueron los siguientes:

- 'changepoint_prior_scale': 0.011348847189364953
- 'changepoint_range': 0.6279962156489811
- 'daily_seasonality': True
- 'growth': 'flat'
- 'interval_width': 0.6294926339715619
- 'n_changepoints': 45
- 'seasonality_mode': 'additive'
- 'seasonality_prior_scale': 19.95612477805556
- 'uncertainty_samples': 1000
- 'weekly_seasonality': False
- 'yearly_seasonality': False

Mensualmente,

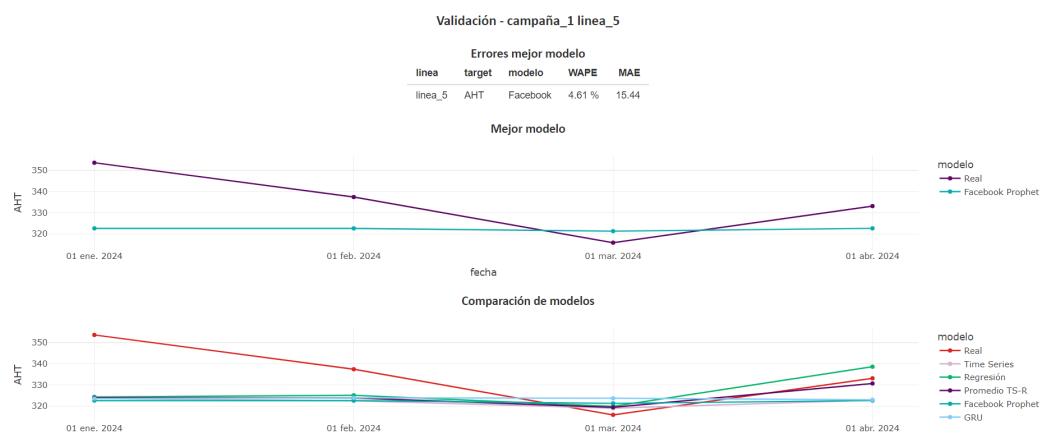


FIGURA 6.38: Línea 5-AHT: Validación mensual

obtenemos un WAPE del 4.61 % y un MAE de 15.44, con lo que observamos que se compensan un poco los errores diarios.

Con lo que obtenemos las siguientes predicciones,



FIGURA 6.39: Línea 4-AHT: Predicción diaria



FIGURA 6.40: Línea 4-AHT: Predicción mensual

6.6. App: ABC PROPHET

En este apartado se presentará en detalle la aplicación desarrollada, "ABC Prophet", la cual se estructura en tres módulos principales: Descripción, Validación y Pronóstico. Cada módulo está diseñado para cumplir funciones específicas dentro del proceso de análisis y predicción, proporcionando una interfaz intuitiva y funcional para los usuarios de WFM.

Descripción

En este módulo se lleva a cabo el análisis exploratorio de cada serie de tiempo. El módulo se organiza en cuatro submódulos clave: Serie, Descomposición, Análisis Estacional Mensual y Análisis Estacional Diario. Cada submódulo proporciona una perspectiva específica para entender mejor los patrones y características de las series temporales, facilitando un análisis detallado y exhaustivo.

Validación

En este módulo se visualizan los resultados de la validación, utilizando herramientas de PyCaret junto con modelos de Facebook Prophet y GRU. El propósito es comparar los resultados de cada modelo y determinar cuál se ajusta mejor a la serie de tiempo seleccionada, ofreciendo una visión clara de la eficacia de cada enfoque en el contexto específico de los datos analizados.

Predicción

Este módulo se compone de dos submódulos: Predicción y Ajuste de Predicción. En el primer submódulo, se presentan las predicciones futuras para cada serie de tiempo, permitiendo a los usuarios visualizar las proyecciones. Además, se muestran las diferencias y desviaciones mensuales, facilitando la identificación de posibles cambios o tendencias emergentes en la serie. Esta funcionalidad es crucial para anticipar variaciones significativas y ajustar las estrategias operativas en consecuencia.

El segundo submódulo se centra en el ajuste de las predicciones. Aquí, se proporciona una herramienta que permite a los usuarios modificar las proyecciones ajustando los valores en un porcentaje específico para cada mes. Esta capacidad es especialmente útil para capturar y reflejar cambios operativos que no se hayan detectado previamente en los datos históricos. Con esta función, es posible hacer ajustes manuales que optimicen las predicciones según las nuevas realidades del negocio, garantizando que las proyecciones se mantengan precisas y alineadas con los cambios organizacionales y del entorno.

Capítulo 7

Conclusiones

Bibliografía

- [1] M. Ali. *PyCaret: An open source, low-code machine learning library in Python.* <https://www.pycaret.org>. PyCaret version 2.3.1. Abr. de 2020.
- [2] Naveen Kumar Chikkakrishna et al. «Short-Term Traffic Prediction Using Sari-ma and FbPROPHET». En: *2019 IEEE 16th India Council International Conference (INDICON)*. 2019, págs. 1-4. DOI: [10.1109/INDICON47234.2019.9028937](https://doi.org/10.1109/INDICON47234.2019.9028937).
- [3] Kyunghyun Cho et al. *Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation*. 2014. arXiv: [1406.1078 \[cs.CL\]](https://arxiv.org/abs/1406.1078). URL: <https://arxiv.org/abs/1406.1078>.
- [4] Junyoung Chung et al. «Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling». En: *arXiv preprint arXiv:1412.3555* (2014). arXiv: [1412.3555 \[cs.NE\]](https://arxiv.org/abs/1412.3555).
- [5] Sihan Ding y Ger Koole. «Optimal call center forecasting and staffing». En: *Probability in the Engineering and Informational Sciences* 36.2 (2022), págs. 254-263. DOI: [10.1017/S0269964820000595](https://doi.org/10.1017/S0269964820000595).
- [6] D. M. Escobar A. A. & Soto. «Predicción de la demanda de llamadas entrantes en un centro de contacto mediante el uso de técnicas de aprendizaje automático.» En: *Journal of Systems Integration* (2023). URL: <http://hdl.handle.net/11371/5753>.
- [7] Pablo Haya. *La metodología CRISP-DM en ciencia de datos.* <https://www.iic.uam.es/innovacion/metodologia-crisp-dm-ciencia-de-datos/>. Instituto de Ingeniería del Conocimiento, Blog. 2021.
- [8] Bineet Jha y Shilpa Pande. «Time Series Forecasting Model for Supermarket Sales using FB-Prophet». En: abr. de 2021, págs. 547-554. DOI: [10.1109/ICCMC51019.2021.9418033](https://doi.org/10.1109/ICCMC51019.2021.9418033).
- [9] Ger Koole. *Webinar on call center WFM: Why WAPE is KING.* <https://www.ccmath.com/erlang-calculators/>. Webinar presented on April 29, 2022. 2022.
- [10] Adrián Martín Sanabria. «Comparación de modelos para predicción de número de llamadas recibidas mediante análisis de series temporales». No Publicado. Madrid, mayo de 2022. URL: <https://oa.upm.es/71966/>.
- [11] Shiva Nosouhian, Fereshteh Nosouhian y Abbas Kazemi Khoshouei. «A review of recurrent neural network architecture for sequence learning: Comparison between LSTM and GRU». En: (2021).
- [12] Greg Rafferty. *Forecasting Time Series Data with Facebook Prophet: Build, improve, and optimize time series forecasting models using the advanced forecasting tool*. Packt Publishing Ltd, 2021.
- [13] Toni Toharudin et al. «Employing long short-term memory and Facebook prophet model in air temperature forecasting». En: *Communications in Statistics-Simulation and Computation* 52.2 (2023), págs. 279-290.

- [14] WildML. *Recurrent Neural Network Tutorial, Part 4 – Implementing a GRU/LSTM RNN with Python and Theano.* <https://wildml.com/2015/10/recurrent-neural-network-tutorial-part-4-implementing-a-gru-lstm-rnn-with-python-and-theano/>. Archived from the original on 2021-11-10, Retrieved May 18, 2016. 2015.