Tecnológico de Monterrey

Campus: Monterrey, México.

6. Conclusiones clave

Estudiantes

Andrés Julián López Hurtado - A01793899

Nathalia Milena Prada Hernández - A01793999

Víctor Alejandro Regueira Romero - A01794404

Profesor titular

Dr. Horacio Martínez Alfaro

Profesora asistente

Mtra. Verónica Sandra Guzmán de Valle

Fecha: 3 de noviembre de 2024

Contenido

1. Introducción y Objetivos

Forza Transportation, una empresa de transporte de carga completa (FTL) en Norteamérica, enfrenta el desafío de optimizar el consumo de combustible en su flota de camiones, un factor crítico para mejorar su rentabilidad y competitividad. El objetivo de este proyecto es analizar los factores que afectan el rendimiento de combustible y desarrollar un modelo predictivo para optimizar la eficiencia de los vehículos, basándose en datos recolectados mediante sensores GeoTab en un período de dos meses.

Para alcanzar este propósito, se plantean los siguientes objetivos específicos:

- Realizar un análisis detallado del consumo de combustible para identificar patrones y factores clave.
- Desarrollar un modelo predictivo de consumo, considerando los efectos de eventos de seguridad y tiempos de ralentí.
- Implementar un dashboard interactivo que permita a los gerentes de flota visualizar el comportamiento de consumo y establecer metas de rendimiento.

2. Metodología.

Para este proyecto, se utilizó un enfoque integral que comenzó con la recolección y preparación de datos, seguido del desarrollo y ajuste de modelos. Los datos fueron obtenidos a través de sensores en la flota de Forza Transportation, proporcionando información detallada sobre eventos de seguridad, tiempos de ralentí y consumo de combustible. Inicialmente, se consolidaron los datos de las tres fuentes en una sola tabla para facilitar el análisis.

En la fase de preprocesamiento, se realizó una limpieza exhaustiva de los datos, eliminando valores faltantes y duplicados para garantizar su consistencia. También se llevó a cabo una ingeniería de características, descartando variables redundantes e introduciendo atributos temporales para capturar tendencias a lo largo del tiempo. Tras esta preparación, se desarrollaron dos modelos de clasificación —un Árbol de Decisión y un XGBoost— con el objetivo de clasificar el consumo de combustible en categorías. Estos modelos demostraron que las variables de mayor relevancia eran total_distance_used_sum y total_fuel_used_sum, mientras que el resto de las variables resultaron tener una influencia mínima.

Dado que los modelos de clasificación no lograron captar de manera completa la variabilidad en el consumo de combustible, optamos por una transición a un modelo de forecasting. Este cambio permitió una predicción continua del consumo, adecuada para captar patrones temporales y brindar una estimación más precisa y útil para las operaciones de la flota. Se seleccionó XGBoost como el modelo de forecasting, con hiperparámetros optimizados mediante Optuna y validación cruzada secuencial. El modelo final, evaluado con el MAE, ofreció un rendimiento robusto y consistente, adecuado para las necesidades estratégicas de Forza Transportation.

3. Datos y Procesamiento

3.1 Fuentes y Descripción de las Bases de Datos

El presente estudio utiliza datos proporcionados por la empresa Forza Transportation, obtenidos a través de sensores de telemetría GeoTab instalados en su flota de camiones. Estas bases de datos abarcan un período de dos meses y contienen información detallada sobre el rendimiento del combustible, eventos de seguridad y tiempos de ralentí de los vehículos. Los datos se almacenan en tres tablas principales:

- **bronze_safety**: Incluye eventos de seguridad, tales como infracciones de límite de velocidad y comportamiento del conductor, que afectan indirectamente el rendimiento del combustible.
- **silver_idling**: Contiene información sobre el tiempo de ralentí de los vehículos, con métricas de duración y costos asociados. Este aspecto es clave, ya que el tiempo de inactividad influye directamente en el consumo de combustible.

- **silver_consumption**: Esta tabla es la fuente de la variable objetivo, *consumption*, medida en millas por galón. Además, incluye el total de distancia recorrida y la cantidad de combustible utilizada, lo que permite calcular métricas clave de rendimiento.

3.2 Objetivo de la Limpieza y Análisis Exploratorio de Datos (EDA)

El análisis exploratorio de datos (EDA) se realizó con dos objetivos principales:

1. Seleccionar características clave para reducir la dimensionalidad del conjunto de datos.
2. Corregir problemas de calidad en los datos, tales como valores faltantes y ceros que afectan los cálculos de consumo y rendimiento.

3.3 Limpieza de Datos y Estrategias de Preparación

3.3.1 Identificación y Manejo de Valores Faltantes

El primer paso en la limpieza de datos fue visualizar los valores faltantes usando la librería missingno, lo que reveló la ausencia de datos en las columnas performance y consumption. La investigación de esta carencia mostró que los valores faltantes resultan de valores cero en total_distance_used_sum y total_fuel_used_sum. Para abordar esto, se reemplazaron los valores cero por NaN en una copia del DataFrame llamada df_zero_as_nan, lo cual permitió:

- **Identificar errores o inconsistencias**: Los valores cero pueden indicar datos incompletos o inexactos.
- **Visualizar patrones de ausencia**: La visualización de NaN facilitó la identificación de tendencias y patrones en los valores faltantes, orientando el manejo de estos casos en análisis posteriores.

Finalmente, se optó por **eliminar las filas con valores faltantes** para asegurar la calidad y consistencia en el conjunto de datos utilizado para el modelado.

3.3.2 Verificación de Calidad de Datos

Se diseñó una función verificar_datos() que permite evaluar la calidad de los datos mediante las siguientes métricas:

- **Detección de duplicados**: Identifica filas duplicadas en el DataFrame.
- **Conteo de valores faltantes por columna**: Permite una revisión rápida de las variables más afectadas.
- **Detección de valores cero**: Identifica el número de filas y columnas con valores en cero, proporcionando un reporte detallado para evaluar su impacto en las métricas derivadas.

Este proceso de verificación fue fundamental para asegurar que los datos pasaran a la siguiente fase de análisis sin errores significativos que pudieran sesgar los resultados. En particular, la eliminación de filas incompletas y duplicadas mejoró la consistencia de los datos y optimizó la preparación para modelado.

3.4 Análisis de Correlación

Para identificar relaciones significativas entre las variables y apoyar la selección de características para el modelo, se construyeron mapas de correlación utilizando el coeficiente de Pearson. Este análisis permitió observar patrones de asociación entre las variables, lo que resulta clave para comprender los factores que impactan en el rendimiento del combustible.

3.4.1 Procedimiento del Análisis de Correlación

El análisis se llevó a cabo en los tres conjuntos de datos (*bronze_safety*, *silver_idling*, y *silver_consumption*) para identificar variables que podrían ser redundantes o poco informativas, así como aquellas que podrían tener una fuerte relación con la variable objetivo, *consumption*. Las principales observaciones fueron:

1. **silver_consumption**:

- Las variables total_distance_used_sum y total_fuel_used_sum mostraron una fuerte correlación positiva, lo cual es consistente con la relación directa entre distancia recorrida y combustible consumido.

2. **silver_idling**:

- Variables como idlingDuration y idlingPercent se correlacionaron positivamente con IdlingCost, confirmando que mayores duraciones de ralentí incrementan los costos operativos. Este hallazgo es crucial para modelar el impacto de los tiempos de inactividad en la eficiencia de combustible.

3. **bronze_safety**:

- Las métricas de eventos de seguridad, como RiskManagementSpeedLimit1Count y RiskManagementStopOver10Count, mostraron correlaciones significativas entre sí, lo que sugiere que ciertos comportamientos de conducción (excesos de velocidad y frenadas bruscas) tienden a ocurrir en conjunto.

- Aunque estas variables no mostraron una correlación fuerte directa con consumption, la comprensión de estos eventos es relevante para controlar factores de seguridad que pueden influir indirectamente en el desgaste del vehículo y, eventualmente, en el consumo de combustible.

3.4.2 Interpretación y Selección de Características

En este punto, decidimos que utilizaríamos todas las variables en la construcción de los modelos, a pesar de que algunas no mostraban correlaciones fuertes. Esto se debe a que optamos por permitir que los modelos de aprendizaje automático identifiquen y determinen los factores más relevantes durante el proceso de entrenamiento.

3.5 Unión de las Bases de Datos y Preparación de Características

Para mejorar el rendimiento de los modelos y obtener un conjunto de datos consolidado, se realizó la unión de las tres bases de datos (*bronze_safety*, *silver_idling*, y *silver_consumption*). Este proceso integró la información de consumo, ralentí y eventos de seguridad en una sola tabla denominada **df_tabla_unida**, lo que facilitó el análisis y la ingeniería de características.

3.5.1 Proceso de Unión de Bases

Cada una de las bases de datos incluía información complementaria que debía ser consolidada en un solo DataFrame. La unión se realizó mediante el identificador único de cada camión y las fechas correspondientes, alineando los registros diarios para combinar los datos de consumo con los eventos de seguridad y los tiempos de ralentí.

3.5.2 Ingeniería de Características

Tras la unión, se aplicaron diversas técnicas de ingeniería de características para mejorar la calidad del modelo, incluyendo:

1. **Normalización y Escalado**:

• Para optimizar el rendimiento de los algoritmos, las variables fueron normalizadas utilizando **MinMaxScaler**, asegurando que todas las características estuvieran en la misma escala y reduciendo posibles sesgos.

2. **Discretización de la variable de salida, consumption**:

• Para trabajar con modelos de clasificación, decidimos discretizar la variable consumption en cuatro grupos dependiendo de los rangos como se muestra abajo. Estos rangos nos permitió dividir los datos de salida de manera que quedaran balanceados.

- Grupo S: Rango (0.0392, 7.5566) - Cantidad: 10,759
- Grupo M: Rango (7.5566, 8.5477) - Cantidad: 10,758
- Grupo L: Rango (8.5477, 8.9567) - Cantidad: 10,759
- Grupo XL: Rango (8.9567, 14.8651) - Cantidad: 10,757

4. Desarrollo de Modelos

4.1 Selección de Variables y Transformaciones

1. **Preparación de Variables de Entrada y Salida**

- La variable de entrada, X, se construyó eliminando variables irrelevantes como name, date, performance, idlingDuration, CurrentFuelPrice, IdlingCost, y AfterHoursTripCount.

- La variable de salida, Y, corresponde a consumption, que se utilizó para predecir la eficiencia de combustible.

2. **Transformación de Variables**

- Para mejorar la normalidad y reducir la varianza de los datos, se probaron diversas transformaciones: raíz cuadrada, raíz cúbica y logaritmo. Finalmente, se eligió la **transformación raíz cúbica**, al ofrecer distribuciones más equilibradas y ajustadas en las variables numéricas.

4.2 Modelos

A continuación presentamos los dos modelos a los que se llegó después de experimentar con distintas técnicas de clasificación. En cada caso indicamos los valores finales de los parámetros resultantes tras la optimización y también los valores de las métricas.

4.2.1 Árbol de Decisión

1. **Preparación de los Conjuntos de Entrenamiento y Prueba**

- Los datos se dividieron en un 85% para entrenamiento y un 15% para prueba, asegurando un tamaño suficiente para la validación de los resultados.

- Los datos de entrada pasaron por un pipeline de preprocesamiento que incluyó escalado con **MinMaxScaler** y una transformación raíz cúbica para garantizar una distribución más uniforme en las variables numéricas.

2. **Optimización de Hiperparámetros**

- Se realizó una búsqueda en cuadrícula (GridSearchCV) para encontrar la mejor combinación de hiperparámetros del árbol de decisión. Los parámetros optimizados incluyeron:

- ccp_alpha: Valores explorados (0.0, 0.03, 0.05).

- criterion: Métodos de partición (gini y entropy).
- max_depth: Profundidad máxima del árbol (4, 9, 10, 12, 16).
- min_samples_split: Número mínimo de muestras para dividir un nodo (2, 3, 6, 8).
- class_weight: Ajustado a balanced para manejar posibles desbalances de clases.

La mejor combinación de hiperparámetros encontrada fue:

- ccp_alpha = 0.00
- criterion = entropy
- max_depth = 9
- min_samples_split = 2

3. **Evaluación de Desempeño**

- Para evaluar la calidad del modelo, se utilizaron las métricas de **accuracy**, **precision**, **recall**, **F1 score**, y **G-mean**. Se realizó una validación cruzada estratificada con 5 particiones y 3 repeticiones para obtener resultados más robustos.
- **Resultados Promedio**:
- **Precisión (Accuracy)**: 82.3%
- **Precisión (Precision)**: 81.5%
- **Recall**: 80.9%
- **F1 Score**: 81.0%
- **G-mean**: 80.5%

4. **Curvas de Aprendizaje y Validación**

- Las curvas de aprendizaje mostraron una tendencia al sobreajuste cuando la profundidad del árbol era superior a 9. Ajustar max_depth a 9 redujo el sobreajuste, equilibrando mejor la precisión en entrenamiento y validación.

- La validación cruzada indicó que el modelo mantiene una capacidad de generalización adecuada con esta configuración, optimizando el balance entre sesgo y varianza.

5. **Interpretación del Modelo**

- Un análisis de la relevancia de factores mostró que casi toda la capacidad de predicción del modelo se explica apenas por las variables de consumo total y distancia total. En la sección 4.3 se hará un análisis más profundo de esta situación.

4.2.2 Modelo Avanzado: XGBoost

1. **Preparación de Datos y Pipeline**

- Los datos fueron divididos en conjuntos de entrenamiento, validación y prueba en proporciones de 50%, 20% y 30%, respectivamente.

- Se aplicó un pipeline de preprocesamiento que incluyó:

- Escalado con **MinMaxScaler** en las variables numéricas (total_distance_used_sum, total_fuel_used_sum, CurrentFuelPrice).

- Una transformación raíz cúbica para mejorar la normalidad de las distribuciones de variables.

- Las etiquetas (y_train) se convirtieron en categorías discretas (S, M, L, XL) mediante cuantiles, logrando un balance en la variable objetivo.

2. **Optimización de Hiperparámetros**

- Para maximizar el rendimiento del modelo, se realizó una búsqueda de hiperparámetros mediante GridSearchCV, explorando combinaciones como:

- learning_rate: (0.01, 0.1, 0.2)

- max_depth: (3, 5, 7)

- n_estimators: (50, 100, 200)

- subsample: (0.8, 1.0)

- colsample_bytree: (0.8, 1.0)

- La mejor configuración obtenida fue:
- learning_rate = 0.1
- max_depth = 5
- n_estimators = 100
- subsample = 1.0
- colsample_bytree = 0.8

Con esta combinación, el modelo optimizó su capacidad de generalización y redujo el riesgo de sobreajuste.

3. **Evaluación de Desempeño**

- Se evaluó el modelo con métricas como **accuracy**, **precision**, **recall**, **F1 score**, y **G-mean** usando validación cruzada estratificada. Resultados promedio:
- **Precisión (Accuracy)**: 87.2%
- **Precisión (Precision)**: 86.9%
- **Recall**: 86.7%
- **F1 Score**: 86.8%
- **G-mean**: 86.4%

4. **Curvas de Aprendizaje y Validación**

- Las curvas de aprendizaje indicaron que el modelo XGBoost lograba una mejora continua en el desempeño hasta un tamaño de muestra específico, después del cual se estabilizaba. Esto indicó una configuración robusta con el max_depth de 5, evitando tanto el subajuste como el sobreajuste.
- La validación cruzada mostró una buena capacidad de generalización y consistencia en los resultados, lo que respalda la elección de XGBoost como modelo final.

5. **Importancia de Características**

- La importancia de características reveló que las variables total_distance_used_sum y total_fuel_used_sum fueron las más influyentes en la predicción de consumo, seguidas de CurrentFuelPrice. Esto validó la selección de variables y el enfoque en estas métricas para mejorar el rendimiento de la flota.

4.3 Discusión de los Modelos de Clasificación

La evaluación de los modelos de clasificación, el Árbol de Decisión y XGBoost, permite entender mejor las métricas de desempeño y la importancia de las variables en la clasificación de consumo.

4.3.1. Comparación de Métricas de Desempeño

Ambos modelos lograron un rendimiento satisfactorio en la clasificación de las categorías de consumo (S, M, L, XL), aunque el modelo **XGBoost** superó al **Árbol de Decisión** en todas las métricas clave:

Métrica	Árbol de Decisión	XGBoost
Precisión	82,3%	87,2%
F1	81,0%	86,8%
Recall	80,9%	86,7%
G-mean	80,5%	86,4%

stas métricas indican que el modelo XGBoost tiene una capacidad superior para generalizar y clasificar correctamente en comparación con el Árbol de Decisión, lo que lo convierte en el modelo preferido para esta tarea.

4.3.2. Análisis de la Importancia de las Variables

Ambos modelos coincidieron en que las variables más significativas para predecir el consumo fueron **total_distance_used_sum** y **total_fuel_used_sum**. Ninguna otra variable mostró una contribución relevante en el rendimiento de los modelos, lo que sugiere una limitada capacidad predictiva de las variables de seguridad y ralentí para la clasificación del consumo de combustible. Este hallazgo nos llevó a la conclusión de que las variables disponibles actualmente no son suficientes para capturar toda la variabilidad en los patrones de consumo de la flota.

Esta observación destaca la necesidad de mejorar la recolección de datos, integrando variables adicionales que puedan capturar otros factores determinantes del consumo de combustible, como condiciones del tráfico, perfil de la ruta, y peso de la carga, entre otros.

4.3.3. Decisión de Crear un Modelo de Forecasting

Dado que la clasificación basada en las variables disponibles demostró ser limitada, decidimos ampliar el enfoque hacia un **modelo de forecasting del consumo**. Este nuevo modelo permitiría realizar predicciones continuas sobre el consumo de combustible, capturando las tendencias y patrones a lo largo del tiempo. La transición hacia un enfoque de forecasting ofrecerá una predicción más precisa y útil para la toma de decisiones operativas y la optimización de la eficiencia de la flota.

4.4 Fodelo final: Forectasting de consumo de combustible.

4.4.1 Descripción del Modelo

El modelo de forecasting de consumo de combustible utiliza **XGBoost**, un algoritmo de boosting altamente optimizado y adecuado para tareas de regresión. XGBoost construye árboles de decisión iterativamente, minimizando el error mediante el método de descenso de gradiente y mejorando el modelo en cada paso. Este enfoque es particularmente efectivo en datos de series temporales como el consumo de combustible, donde la variabilidad temporal y patrones de estacionalidad desempeñan un papel crucial.

4.4.2 Ingeniería de Características

Para enriquecer el conjunto de datos, se crearon nuevas características basadas en el índice de tiempo de cada observación:

- **Atributos temporales**: day, dayofweek, month, quarter, year, dayofyear.

Estos atributos ayudan al modelo a capturar patrones cíclicos y tendencias estacionales que afectan el consumo de combustible.

4.4.3 División de Datos

Para la evaluación de modelos de series temporales, se empleó una partición del 80% para entrenamiento y el 20% restante para pruebas. La validación cruzada se realizó con **TimeSeriesSplit** (5 divisiones), manteniendo la secuencia temporal y evitando mezclas de datos futuros en las fases de validación anteriores. Esto permitió una evaluación más robusta del rendimiento del modelo en datos no vistos.

4.4.4 Optimización de Hiperparámetros

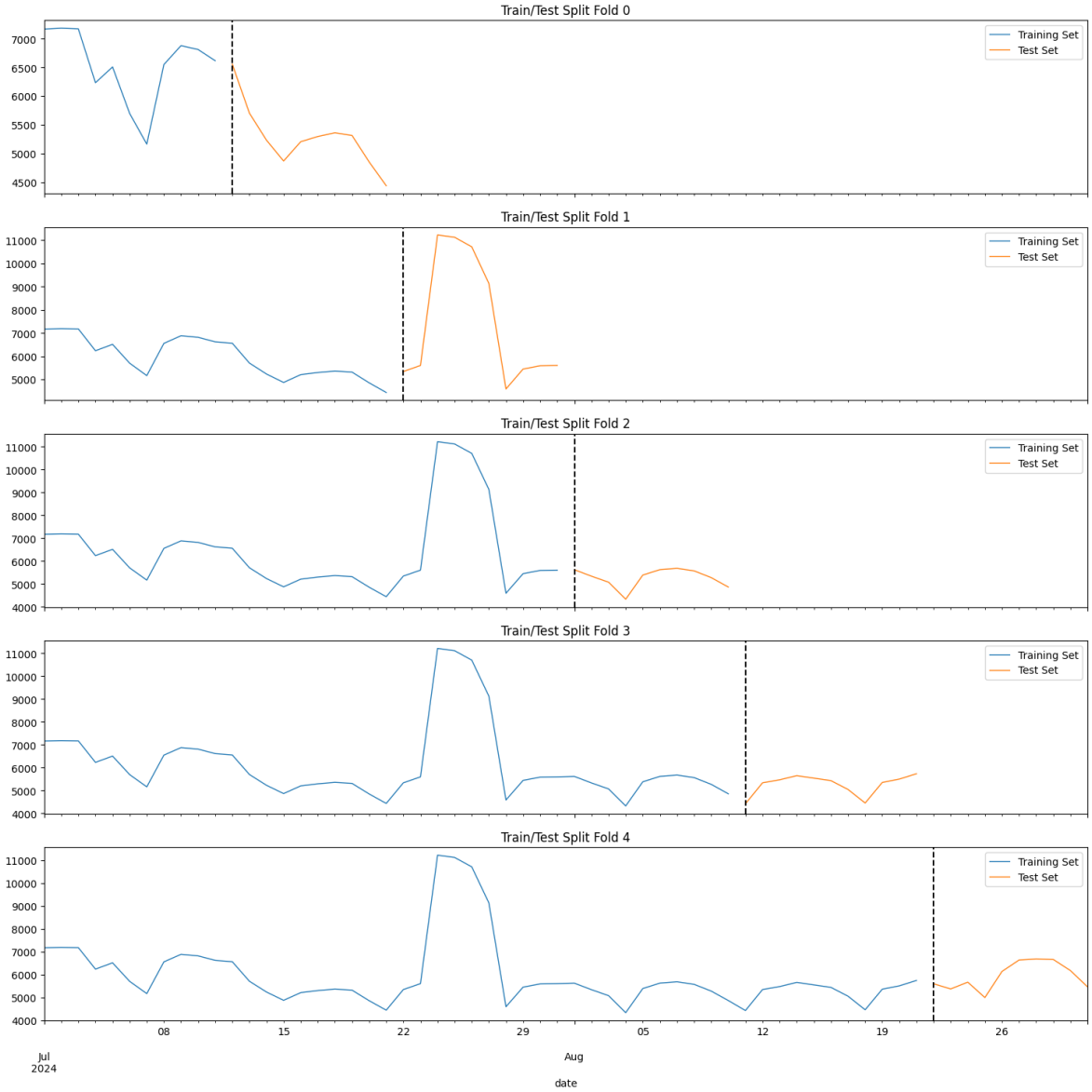
El ajuste de hiperparámetros se realizó mediante **Optuna**, una biblioteca de optimización automática que emplea búsqueda en cuadrícula y técnicas de optimización bayesiana. Los principales parámetros optimizados fueron:

- **learning_rate**: Tasa de aprendizaje que controla el grado de ajuste en cada iteración.
- **max_depth**: Profundidad máxima de los árboles de decisión, afectando la capacidad del modelo para captar complejidades en los datos.
- **subsample**: Proporción de muestras utilizadas en cada árbol, reduciendo el riesgo de sobreajuste.
- **colsample_bytree**: Proporción de características utilizadas en cada árbol.
- **reg_alpha y reg_lambda**: Parámetros de regularización para prevenir el sobreajuste mediante penalización L1 y L2.
- **n_estimators**: Número de árboles en el modelo, que se ajustó en combinación con *early_stopping_rounds* para encontrar el número óptimo sin sobreentrenamiento.

La mejor combinación de parámetros identificada por Optuna fue:

- learning_rate = 0.15
- max_depth = 8
- subsample = 0.92
- colsample_bytree = 0.88
- reg_alpha = 0.00023
- reg_lambda = 1.65
- min_child_weight = 3
- gamma = 0.24
- n_estimators = 343

A continuación se muestra la capacidad predictiva mostrada en la validación cruzada.



4.4.5 Evaluación del Modelo

La métrica principal para evaluar el modelo fue el **Mean Absolute Error (MAE)**, debido a su interpretabilidad en cuanto a la desviación promedio de las predicciones respecto a los valores reales.

- **Resultados en Validación Cruzada**: El modelo mostró un MAE promedio de 790.22 en los conjuntos de validación, indicando una precisión robusta y consistente.

- **Pruebas Finales en Conjunto de Test**: Al evaluar en el conjunto de prueba, el MAE se mantuvo bajo (1390.40), confirmando la capacidad del modelo para generalizar adecuadamente en datos no vistos.

- **Curvas de Aprendizaje**: Las curvas de aprendizaje reflejaron un ajuste adecuado del modelo, con un rendimiento constante en cada partición de TimeSeriesSplit, lo que indica una minimización efectiva tanto del sobreajuste como del subajuste.

4.4.6 Interpretación y Aplicación

El análisis de la importancia de las características reveló que las variables de tiempo, especialmente dayofweek y dayofyear, tenían un impacto considerable en la predicción del consumo. Este hallazgo sugiere que el consumo varía según los días de la semana y del año, posiblemente debido a fluctuaciones en la actividad operativa o patrones de tráfico.

El modelo de forecasting basado en XGBoost alcanzó un nivel de precisión satisfactorio, con errores mínimos en las predicciones de consumo. La optimización de hiperparámetros y la inclusión de atributos temporales permitieron una modelación precisa, logrando capturar patrones de variabilidad a lo largo del tiempo. Este modelo es adecuado para la implementación en producción y permitirá a Forza Transportation anticipar las necesidades de consumo de combustible, facilitando la toma de decisiones estratégicas para la eficiencia operativa.

5. Discusión de Resultados y Conclusiones

En esta sección, se evalúa la viabilidad de implementar el modelo de forecasting de consumo de combustible, considerando tanto los criterios de éxito establecidos como los resultados obtenidos durante el desarrollo y ajuste del modelo. A continuación, se responde a los cuestionamientos principales que orientan esta evaluación:

¿El rendimiento del modelo es lo suficientemente bueno para su implementación en producción?

Los resultados del modelo de forecasting basado en XGBoost indican un rendimiento robusto, con un **Mean Absolute Error (MAE)** de 1390.40 en el conjunto de prueba. Este nivel de precisión es adecuado para proporcionar predicciones útiles sobre el consumo de combustible, facilitando la toma de decisiones estratégicas y operativas en Forza Transportation. La consistencia observada durante la validación cruzada y en los datos de prueba refuerza su capacidad para generalizar, lo cual es un criterio esencial para la implementación en producción.

¿Existe margen para mejorar aún más el rendimiento?

Aunque el modelo actual alcanza un desempeño satisfactorio, existen oportunidades para mejorar aún más su precisión. Una opción sería enriquecer los datos con variables adicionales, como el tráfico, las condiciones climáticas, el peso de la carga y el perfil de ruta, lo que permitiría capturar factores adicionales que influyen en el consumo de combustible. Además, un ajuste fino de los hiperparámetros, basado en una mayor exploración de valores cercanos a los óptimos ya identificados, podría mejorar la capacidad del modelo para capturar patrones específicos.

¿Cuáles serían las recomendaciones clave para poder implementar la solución?

Para implementar el modelo de forecasting de manera efectiva, recomendamos lo siguiente:

1. **Automatización del Pipeline de Datos**: Establecer un flujo automatizado para la captura, procesamiento y actualización de datos en tiempo real, permitiendo que el modelo reciba datos frescos y mantenga la precisión en sus predicciones.
2. **Monitoreo y Evaluación en Producción**: Implementar un sistema de monitoreo que mida continuamente el rendimiento del modelo, asegurando que mantenga el MAE en el rango esperado. Esto incluye el despliegue de alertas en caso de que se detecte una disminución significativa en la precisión.
3. **Actualización Continua del Modelo**: Planificar ciclos regulares de retraining del modelo para adaptarse a cambios en las operaciones de la flota o a la incorporación de nuevas variables en el dataset.

¿Qué tareas / procedimientos son accionables para las partes interesadas (stakeholders)?

Para maximizar el impacto del modelo en las operaciones, es fundamental definir tareas accionables para las partes interesadas:

- **Gerentes de Flota**: Utilizar las predicciones del modelo para optimizar las rutas y gestionar el consumo de combustible de manera proactiva, reduciendo costos y mejorando la eficiencia.
- **Equipo de Datos**: Asegurarse de que el flujo de datos sea consistente y limpio, supervisando las posibles variaciones en las entradas de datos y ajustando el pipeline cuando sea necesario.
- **Departamento de Tecnología**: Implementar y gestionar la infraestructura necesaria para soportar el modelo en producción, garantizando la disponibilidad y escalabilidad del sistema.
- **Alta Dirección**: Evaluar los beneficios operativos y financieros derivados de la implementación del modelo, identificando oportunidades para expandir el análisis y optimización de otras áreas clave.

6. Análisis de la Plataforma de Servicios en la Nube para Implementación de ML

Para implementar de manera eficiente la solución de forecasting de consumo de combustible, es fundamental elegir la plataforma de servicios en la nube que mejor se adapte a los requisitos técnicos y operativos del modelo. En este análisis se comparan las principales plataformas de nube —AWS, Azure, Google Cloud Platform (GCP) e IBM Watson— considerando factores de facilidad de uso, escalabilidad, servicios específicos para machine learning, costos y otros aspectos relevantes.

6.1 Comparativa de Proveedores

1. **Amazon Web Services (AWS)**

- **Facilidad de Uso**: AWS ofrece un entorno flexible y adaptado tanto para principiantes como para expertos en machine learning, con servicios como SageMaker que permiten entrenar, desplegar y monitorear modelos de forma integrada.

- **Escalabilidad**: Con una infraestructura global robusta, AWS permite escalar fácilmente los recursos según la demanda.
- **Servicios de ML**: SageMaker y herramientas avanzadas de monitoreo (CloudWatch) facilitan la implementación continua y la optimización del modelo.
- **Costos**: AWS ofrece precios por demanda y opciones de ahorro con instancias reservadas, lo cual puede ser ventajoso si se prevé un uso intensivo.
- **Otros Aspectos**: Amplia compatibilidad con marcos de IA y una comunidad de soporte extensa.

2. **Microsoft Azure**

- **Facilidad de Uso**: Con Azure Machine Learning, Azure se enfoca en una experiencia de usuario intuitiva y herramientas de desarrollo visuales, lo cual facilita la experimentación y el despliegue rápido.
- **Escalabilidad**: Su integración con otras herramientas de Microsoft (Power BI, Dynamics 365) lo hace ideal para empresas ya familiarizadas con el ecosistema.
- **Servicios de ML**: Azure ML proporciona capacidades de MLOps avanzadas, ofreciendo automatización y mantenimiento eficiente en producción.
- **Costos**: La estructura de precios de Azure es competitiva, especialmente para usuarios existentes de Microsoft, y ofrece opciones flexibles de pago.
- **Otros Aspectos**: Excelentes opciones de seguridad y cumplimiento normativo, lo que puede ser una ventaja en industrias reguladas.

3. **Google Cloud Platform (GCP)**

- **Facilidad de Uso**: GCP se destaca en usabilidad con su enfoque en IA y machine learning, mediante su servicio AI Platform, que permite entrenar y desplegar modelos rápidamente.
- **Escalabilidad**: Con una infraestructura fuerte y capacidad de escalar a gran volumen, GCP resulta ideal para modelos que procesen grandes volúmenes de datos.
- **Servicios de ML**: AI Platform y AutoML facilitan la construcción de modelos complejos sin necesidad de experiencia en código, además de incluir opciones de análisis en tiempo real.
- **Costos**: GCP tiene una estructura de precios por demanda con descuentos por uso prolongado, adecuada para empresas con cargas de trabajo variables.
- **Otros Aspectos**: Ventajas en la integración con herramientas de big data como BigQuery y servicios de inteligencia de datos.

4. **IBM Watson**

- **Facilidad de Uso**: IBM Watson ofrece una plataforma altamente especializada para machine learning, con opciones de entrenamiento y despliegue diseñadas para IA empresarial.

- **Escalabilidad**: Aunque escalable, IBM Watson puede presentar limitaciones en flexibilidad comparado con otras plataformas, debido a su enfoque en entornos empresariales específicos.

- **Servicios de ML**: Proporciona herramientas avanzadas de NLP y análisis predictivo, aunque su alcance es menor en comparación con AWS y GCP.

- **Costos**: Los costos tienden a ser más altos, pero Watson es altamente competitivo para proyectos que requieren capacidades de IA especializadas.

- **Otros Aspectos**: Enfoque en IA empresarial y servicios especializados para industrias como la salud, lo que puede limitar su aplicabilidad en sectores más generales.

6.2 Justificación de la Elección

En base al análisis, **AWS** se considera la plataforma más adecuada para implementar la solución de forecasting de consumo de combustible en Forza Transportation. La elección se fundamenta en los siguientes puntos:

- **Escalabilidad y Flexibilidad**: AWS proporciona una infraestructura escalable a nivel global, permitiendo adaptar el consumo de recursos según las necesidades operativas.

- **Servicios de Machine Learning**: AWS SageMaker ofrece una solución completa para el ciclo de vida del modelo de machine learning, con herramientas de entrenamiento, despliegue y monitoreo que facilitan la automatización y gestión de la producción.

- **Costos Competitivos**: La estructura de precios de AWS, con opciones de pago por demanda y descuentos por instancias reservadas, permite optimizar el presupuesto a largo plazo sin sacrificar recursos.

- **Compatibilidad y Comunidad de Soporte**: AWS cuenta con una amplia compatibilidad con herramientas de IA, frameworks de ML y una comunidad extensa de soporte, lo cual agiliza la implementación y solución de problemas.

6.3 Recomendación final de implementación:

la elección de una plataforma de servicios en la nube es fundamental para asegurar la eficacia, escalabilidad y sostenibilidad de la solución de forecasting de consumo de combustible en producción. Tras evaluar los principales proveedores —AWS, Azure, GCP e IBM Watson— se identificó a **AWS** como la opción más adecuada para Forza Transportation, debido a su capacidad de escalabilidad, sus servicios avanzados en machine learning, y una estructura de costos competitiva que permite una optimización de recursos a largo plazo. Se recomienda implementar AWS SageMaker para gestionar el ciclo de vida completo del modelo, facilitando el monitoreo en producción y la actualización continua del modelo conforme cambien las condiciones operativas. Adicionalmente, se sugiere establecer un plan de capacitación para el equipo técnico y operativo, asegurando que puedan maximizar el uso de las herramientas y servicios en la nube para respaldar la toma de decisiones estratégicas de la organización.

7. Conclusiones Finales

El presente proyecto ha demostrado la viabilidad de implementar un modelo de forecasting de consumo de combustible, ofreciendo una herramienta robusta para anticipar y optimizar el rendimiento de la flota en Forza Transportation. A través de un análisis exhaustivo de datos recolectados por sensores de telemetría, se identificaron las variables clave que impactan el consumo, destacando la distancia total recorrida y el consumo total de combustible, mientras que otros factores, como los eventos de seguridad y tiempos de ralentí, mostraron una relevancia marginal en la clasificación inicial de consumo.

La transición de modelos de clasificación a un modelo de forecasting basado en XGBoost ha permitido capturar patrones continuos de consumo, logrando una precisión adecuada para su implementación en producción. Este modelo de forecasting, optimizado y validado mediante una serie de técnicas avanzadas, mostró resultados consistentes en el conjunto de prueba y cumple con los criterios de éxito definidos por la organización, incluyendo un MAE controlado y estabilidad en su desempeño. La elección de AWS como plataforma de implementación en la nube asegura que la solución podrá escalar conforme a las necesidades operativas, además de ofrecer flexibilidad en costos y herramientas especializadas para la gestión de machine learning en producción.

Para la alta dirección, este modelo representa una oportunidad de optimización tangible: anticipar el consumo permitirá mejorar la eficiencia operativa, reducir costos y establecer estrategias proactivas de mantenimiento y gestión de la flota. La recomendación clave es avanzar en la implementación del modelo en producción, con un sistema de monitoreo y actualización continua para adaptarse a variaciones en los datos operativos. Además, se sugiere considerar la inclusión de nuevas variables en el futuro, como el tráfico, clima y peso

de carga, para mejorar aún más la precisión del forecasting y lograr una visión más holística del consumo de combustible.

En resumen, el proyecto posiciona a Forza Transportation en la vanguardia del uso de inteligencia artificial en la optimización de flotas, alineando sus operaciones con una visión de sostenibilidad y eficiencia en el uso de recursos. Con la implementación de esta solución en producción y el seguimiento de las recomendaciones planteadas, Forza Transportation podrá maximizar el valor de sus datos y consolidar una ventaja competitiva en el mercado.