Kiosko

Pronosticar las ventas diarias a nivel de producto (SKU)

06 de Septiembre 2024

VISIÓN GENERAL

La cadena de tiendas enfrenta desafíos en la gestión eficiente de inventarios debido a la variabilidad en las ventas de productos, que se ve influenciada por múltiples factores como promociones, estacionalidad y eventos especiales. Un modelo preciso de predicción de ventas ayudará a reducir costos de almacenamiento, minimizar pérdidas por productos no vendidos y mejorar la disponibilidad de productos en las tiendas.

OBJETIVOS

1. Determinar los factores que afectan las unidades vendidas de productos
2. Desarrollar un modelo predictivo que permita pronosticar las ventas diarias a nivel de producto (SKU) en cada una de las tiendas de la cadena.
3. Analizar resultados y generar recomendaciones para una mejor gestión de inventarios

ESPECIFICACIONES

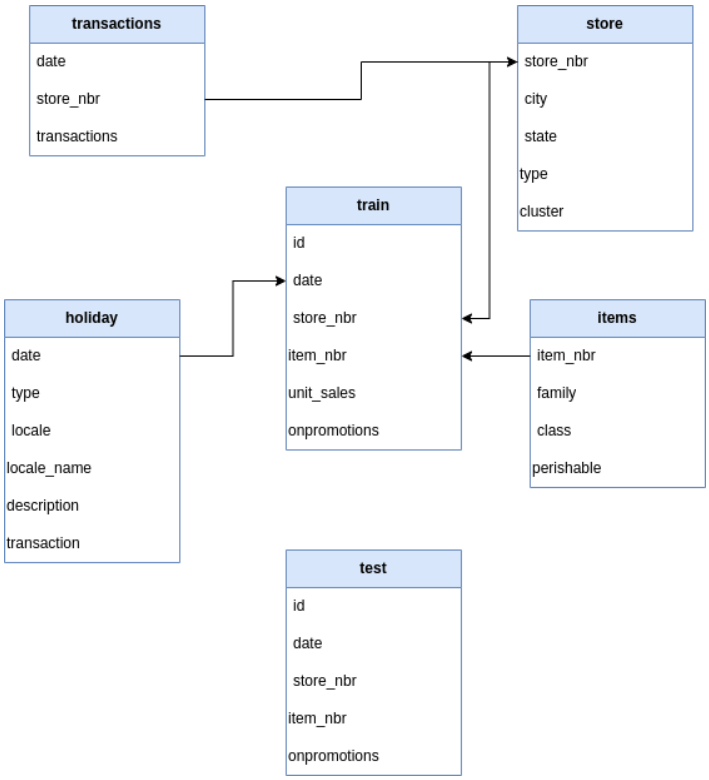
En esta sección se van a describir los elementos que participan en este caso de estudio. El objetivo es tener presento las posibles limitaciones o alcances de esta prueba.

Datos

El primer elemento a considerar son los datos. En este caso han sido proporcionados en un archivo comprimido con la siguiente estructura:

Name	Size
holidays_events.csv	22.3 kB
items.csv	101.8 kB
sample_submission.csv	40.4 MB
stores.csv	1.4 kB
test.csv	126.2 MB
train.csv	5.0 GB
transactions.csv	1.6 MB

Son siete archivos en formato csv, teniendo como el archivo más pesado a train.csv de 5.0 gb. Cada uno de estos archivos almacenan tablas de la siguiente manera:



En el diagrama anterior se muestran las tablas que se generan con cada archivo y como podemos relacionar las tablas. A continuación se mostrará un análisis más detallado del contenido de las tablas.

Holidays

En esta tabla se encuentran las fechas que son días especiales en cierta región y que deberían ser considerados como uno de las principales características a analizar.

Esta tabla tiene los siguientes tamaño y tipos de datos:

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 350 entries, 0 to 349
Data columns (total 6 columns):
 #   Column       Non-Null Count  Dtype 
---  ------       --------------  ----- 
 0   date         350 non-null    object
 1   type         350 non-null    object
 2   locale       350 non-null    object
 3   locale_name  350 non-null    object
 4   description  350 non-null    object
 5   transferred  350 non-null    bool  
dtypes: bool(1), object(5)
memory usage: 14.1+ KB
```

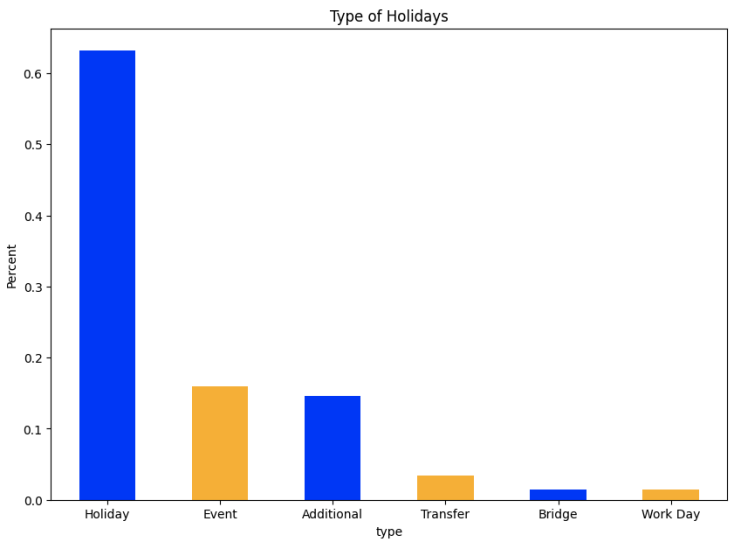
Contiene 350 elementos no nulos de 6 distintas columnas, un tipo de dato booleano, un columna de fecha, cuatro columnas categorias dado que la columna description es una frase corta que hace referencia al holiday.

Esta tabla tiene como inicio de registro la fecha 2012-03-02 y como la fecha de registro más grande 2017-12-26.

Las columnas categóricas contienen los siguiente tipos y sus frecuencias:

- type
    - Holiday:221
    - Event: 56
    - Additonal: 51
    - Transfer: 12
    - Bridge: 5
    - Work Day: 5

Como se puede observar el tipo Holiday es el que más fechas tiene registradas.



La columna categórica locale contiene los siguientes valores:

- National: 174
- Local: 152
- Regional: 24

Siendo los holidays de tipo National los que contienen mayor registro de fechas. La columna transferred es una columna booleana con: 12 valores verdaderos y 338 valores falsos. Una clase muy desbalanceado.

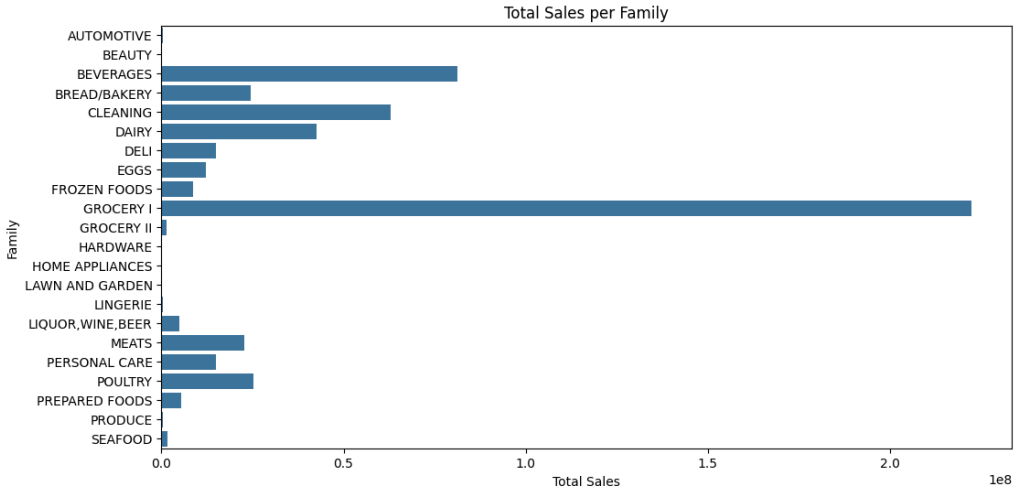
Items

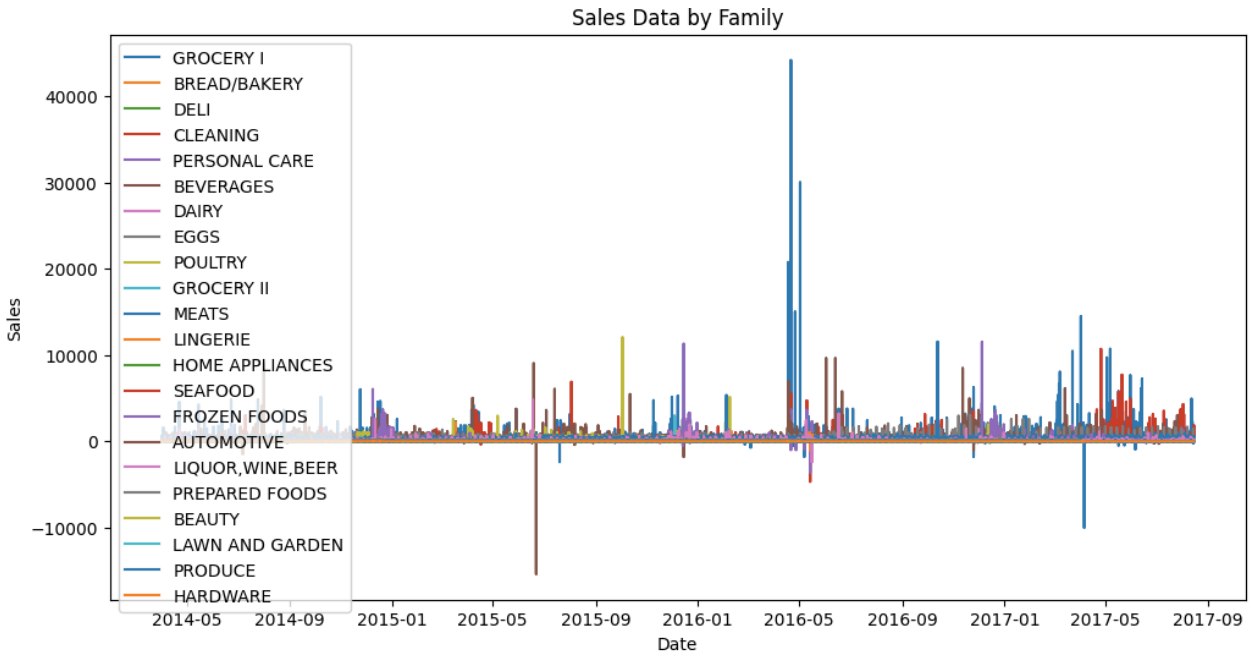
La tabla de los items contiene 4 columnas con 4100 datos no nulos por columna, tres columnas catgóricas: family, class y perishable, y un índice de los items: item_nbr.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 4100 entries, 0 to 4099
Data columns (total 4 columns):
 #   Column      Non-Null Count  Dtype 
---  ------      --------------  ----- 
 0   item_nbr    4100 non-null   int64 
 1   family      4100 non-null   object
 2   class       4100 non-null   int64 
 3   perishable  4100 non-null   int64 
dtypes: int64(3), object(1)
memory usage: 128.2+ KB
```

Dentro de las columnas importantes a revisar está el tipo y la clase:

- Número de elementos en family: 33
- Número de clases: 337
- Número de productos perecederos:
    - No: 3114
    - SI: 986





Stores

Esta tabla contiene 5 columnas con 54 datos no nulos por columna. Contiene cuatro columnas categóricas y una columna de índice de los stores.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 54 entries, 0 to 53
Data columns (total 5 columns):
 #   Column     Non-Null Count  Dtype 
---  ------     --------------  ----- 
 0   store_nbr  54 non-null     int64 
 1   city       54 non-null     object
 2   state      54 non-null     object
 3   type       54 non-null     object
 4   cluster    54 non-null     int64 
dtypes: int64(2), object(3)
memory usage: 2.2+ KB
```

Las columnas contienen los siguientes datos categóricos:

- **Número de Ciudades**: 22
    - ['Quito' 'Santo Domingo' 'Cayambe' 'Latacunga' 'Riobamba' 'Ibarra' 'Guaranda' 'Puyo' 'Ambato' 'Guayaquil' 'Salinas' 'Daule' 'Babahoyo' 'Quevedo' 'Playas' 'Libertad' 'Cuenca' 'Loja' 'Machala' 'Esmeraldas' 'Manta' 'El Carmen']

- **Número de Estados**: 16

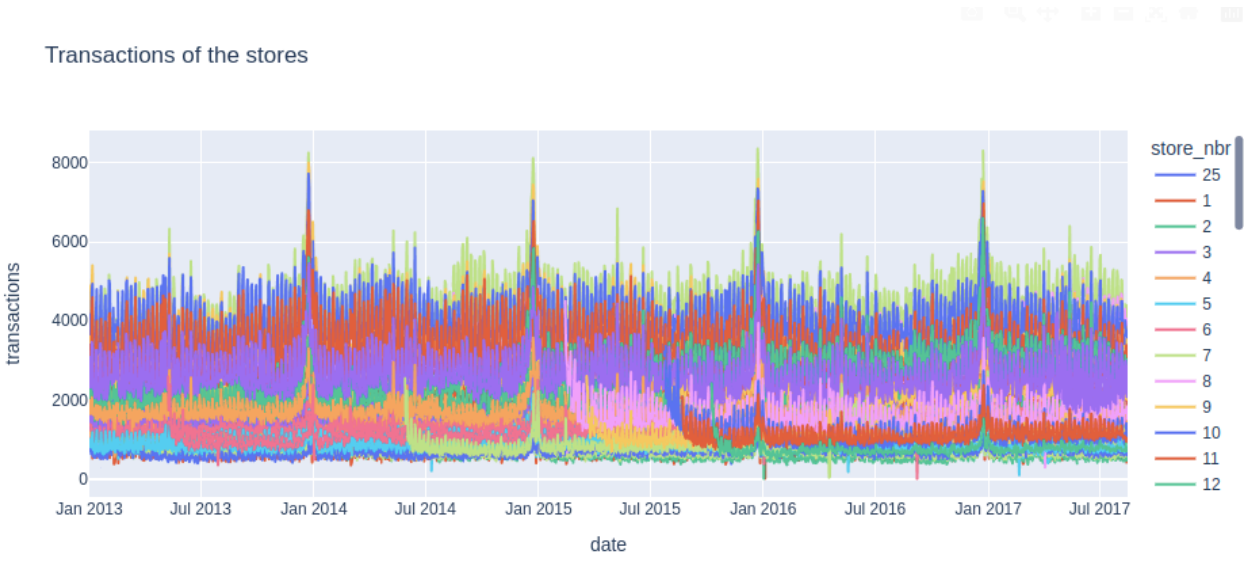
    - ['Pichincha' 'Santo Domingo de los Tsachilas' 'Cotopaxi' 'Chimborazo' 'Imbabura' 'Bolivar' 'Pastaza' 'Tungurahua' 'Guayas' 'Santa Elena' 'Los Rios' 'Azuay' 'Loja' 'El Oro' 'Esmeraldas' 'Manabi']
- **Tipos de Stores**: 5
    - ['D' 'B' 'C' 'E' 'A']
- **Stores Similares (clusters)**: 17
    - [13 8 9 4 6 15 7 3 12 16 1 10 2 5 11 14 17]

Transacciones

Esta tabla contiene 3 columnas con 83488 datos no nulos en cada columna, es una columna de fecha, un identificador de tiendas y un entero que hace referencia a las transacciones por tienda

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 83488 entries, 0 to 83487
Data columns (total 3 columns):
 #   Column        Non-Null Count  Dtype 
---  ------        --------------  ----- 
 0   date          83488 non-null  object
 1   store_nbr     83488 non-null  int64 
 2   transactions  83488 non-null  int64 
dtypes: int64(2), object(1)
memory usage: 1.9+ MB
```

En la siguiente gráfica podemos observar la distribución y frecuancia de las transacciones en la línea de tiempo.



Como se puede observar se ven unos picos muy grandes al incio de cada año, al parcer solo es para cierto tipo de stores.

Train

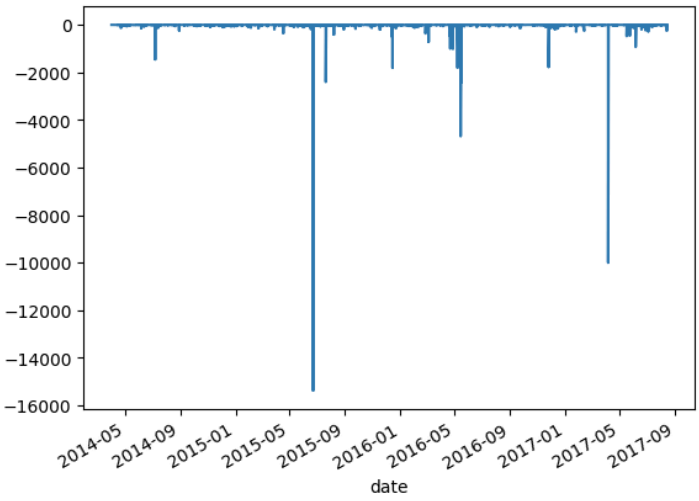
El dataset importante de este caso de uso contiene seis columnas con 125497040 de entradas pesando más de 5 gb de memoria. Esta tabla contiene una columna de fecha y de id, tres columnas categóricas que involucran el store, el item del producto y onpromotion.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 125497040 entries, 0 to 125497039
Data columns (total 6 columns):
 #   Column       Non-Null Count      Dtype  
---  ------       --------------      -----  
 0   id           125497040 non-null  int64  
 1   date         125497040 non-null  object 
 2   store_nbr    125497040 non-null  int64  
 3   item_nbr     125497040 non-null  int64  
 4   unit_sales   125497040 non-null  float64
 5   onpromotion  103839389 non-null  object 
dtypes: float64(1), int64(3), object(2)
memory usage: 5.6+ GB
```

Como se puede observar la columna onpromotion contiene datos nulos, realizando un analisis son: 21657651 datos nulos. Esta a sido el primer criterio para recorta el dataset y solo quedarnos con 103839389 elementos. El siguiente paso fué considerar los items que tengan información al inicio de la fecha donde comienzan los registros, es decir, existen algunos stores que se han agregado en una fecha que ya inició el registro. Eliminando todos estos items tenemos finalmente 67029280 registros en el dataset para realizar el análisis.

Analizando este dataset, una de las primeras opbservaviones fué encontrar picos muy altos dentro del histórico, datos negativos y datos con punto decimal.

Los datos negativos son 3548, analizando las unidades de venta en este conjunto de datos se puede observar que el menor número es -15372 y el máximo es un decimal igual a -0.002. Acá tenemos el caso de números negativos que en este momento se pueden considerar como pérdias, además números decimales que pueden deberse a productos que se pueden comerciar en porciones y no enteros, debido por ejemplo a la familia, cluster del item.




```
count     3548.000000
mean       -26.693853
std        344.960479
min     -15372.000000
25%         -4.000000
50%         -1.000000
75%         -1.000000
max         -0.002000
Name: unit_sales, dtype: float64
```

Analizando el valor mas pequeño se puede ver que el item aparece dos veces en este conjunto negativo, el item hace referencia a una familia de bebidas. Si revisamos en una fuente externa estas fechas podemos notar los siguiente:

	id	date	store_nbr	item_nbr	unit_sales	onpromotion	family	class	perishable	city	state	type	cluster
1204	49592112	2015-06-22	18	1166474	-15372.0	False	BEVERAGES	1120	0	Quito	Pichincha	B	16
1429	56041960	2015-09-08	20	1166474	-1.0	False	BEVERAGES	1120	0	Quito	Pichincha	B	6

El `item_nbr` = 1166474 muestra dos valles en las fechas: 2015-06-22 y 2015-09-08. Revisando el calendario podemos observar que:

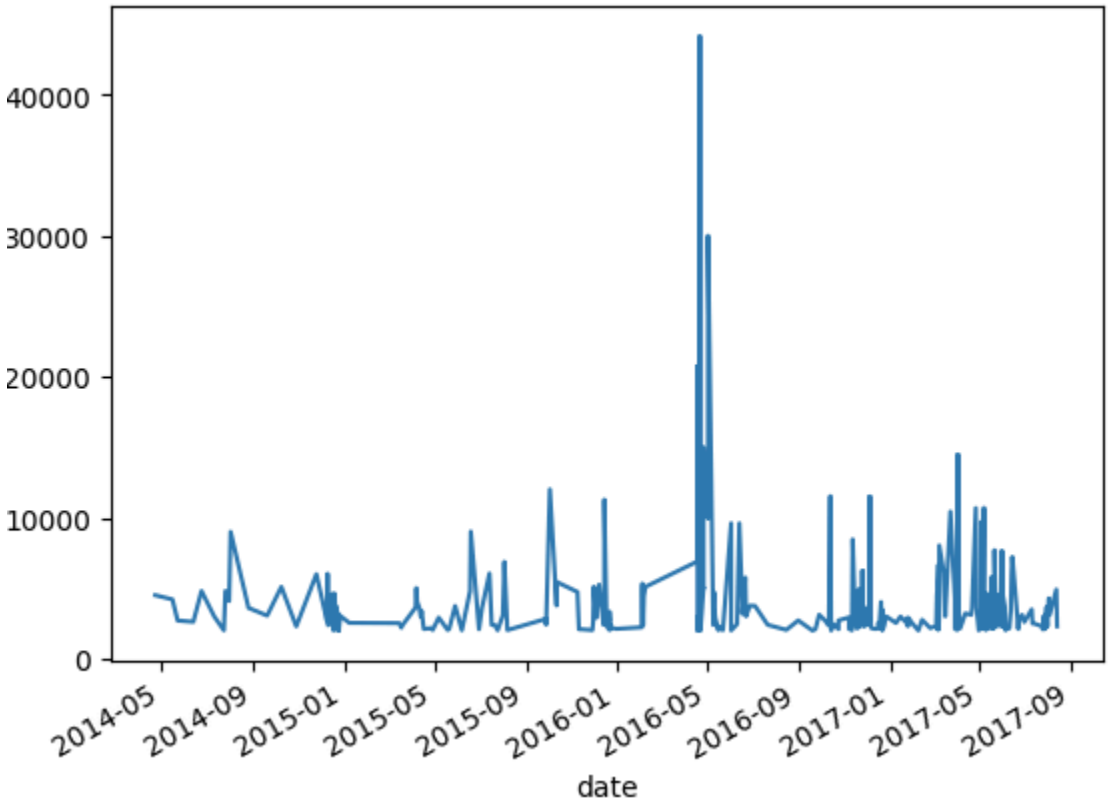
- 2015-06-22 = Fiesta del Inti Raymi: La preparación de la festividad era estricta, pues en los previos «tres días no se comía sino un poco de maíz blanco, crudo, y unas pocas de yerbas que llaman chúcam y agua pura.
- 2015-09-08 = Día Internacional de Gato

Estos valles han sido generados sobre productos de la familia BEVERAGES por las celebraciones y cancelación del consumo de bebidas. Ahora los valores negativos con decimal están asociadas a las siguientes familias:

family	frequency
POULTRY	53
MEATS	43
DELI	4
SEAFOOD	3
FROZEN FOODS	2

Cabe mencionar que un holiday asociado y con mucha frecuencia a los valores negativos es: **Terremoto Manabi**

Ahora analicemos los valores más grandes del histórico.



Como se puede observar a continuación estos valores están solo concentrados en menos 1% de los valores son 445 valores con el máximo en 44142 y el mínimo en 2001 unidades vendidas.

```
count      445.000000
mean      3669.848081
std       3283.263507
min       2001.000000
25%       2302.000000
50%       2735.000000
75%       3674.000000
max      44142.000000
Name: unit_sales, dtype: float64
```

Estos son algunos de los Holidays registrados en el rango de valores más altos

	date	type	locale	locale_name	description	transferred	m_d
106	2014-06-12	Event	National	Ecuador	Inauguracion Mundial de futbol Brasil	False	06-12
109	2014-06-23	Holiday	Local	Guaranda	Cantonizacion de Guaranda	False	06-23
126	2014-07-23	Holiday	Local	Cayambe	Cantonizacion de Cayambe	False	07-23
128	2014-07-25	Holiday	Local	Guayaquil	Fundacion de Guayaquil	False	07-25
148	2014-12-08	Holiday	Local	Loja	Fundacion de Loja	False	12-08

Si analizamos uno de los items de este grupo se encuentra el item que se mostró el sección de negativos:

	id	date	store_nbr	item_nbr	unit_sales	onpromotion	family	class	perishable	city	state	type	cluster
97	53365584	2015-08-07	45	1166474	2032.0	False	BEVERAGES	1120	0	Quito	Pichincha	A	11
87	49347648	2015-06-19	18	1166474	9025.0	False	BEVERAGES	1120	0	Quito	Pichincha	B	16

Las fechas en donde están estos picos son:

El `item_nbr` = 1166474 muestra dos picos en las fechas: 2015-08-07 y 2015-06-19. Revisando el calendario podemos observar que:

- 2015-08-07 = Día Internacional de la Cerveza
- 2015-06-19 = Día Internacional para la Eliminación de la Violencia Sexual en los Conflictos

Estos picos han sido generados sobre productos de la familia BEVERAGES por las celebraciones.

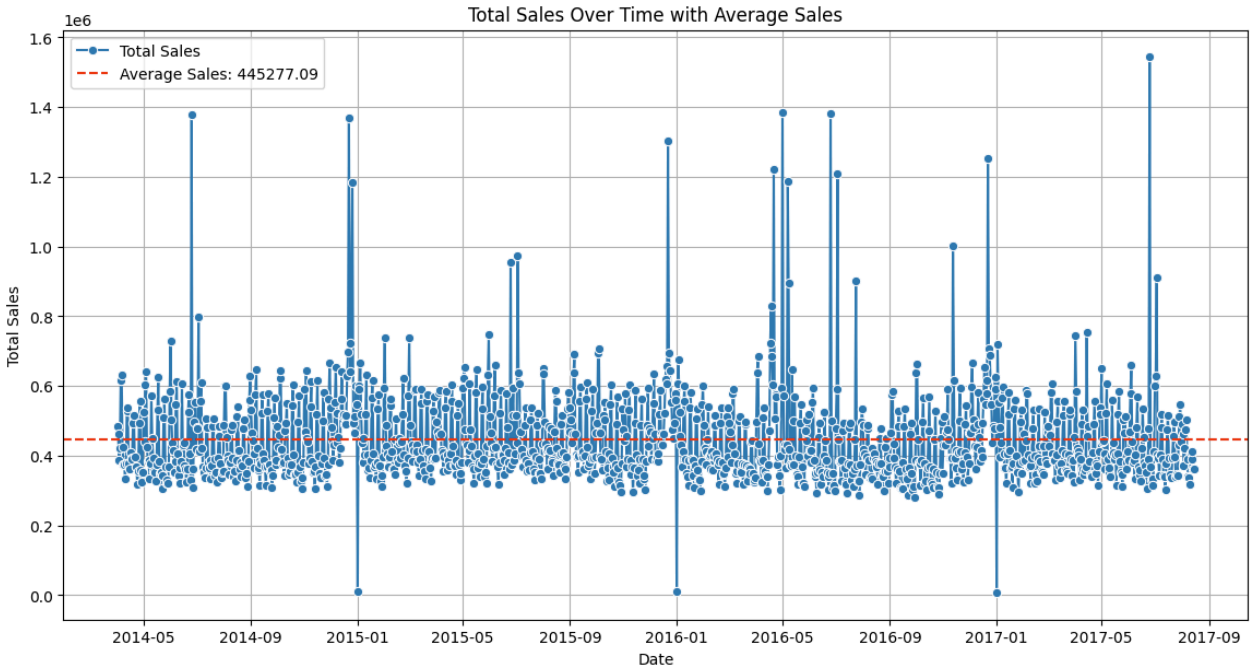
Una vez analizado el contenido y estructura de los datos, hemos observado lo siguiente:

- El dataset de entrenamiento final contiene 67,029,280
- La feature unit_sales, muestra valores decimales, negativos y muy grandes (afectados por los holidays)
- Existen más datos que pueden ayudar al modelo a identificar o aprender sobre la configuración de días de este tipo
- Agregar columnas de tablas como:
    - holidays: date,type,local y transferred
    - item: family, perishable
    - store: type, clluster
    - transaction:transactions

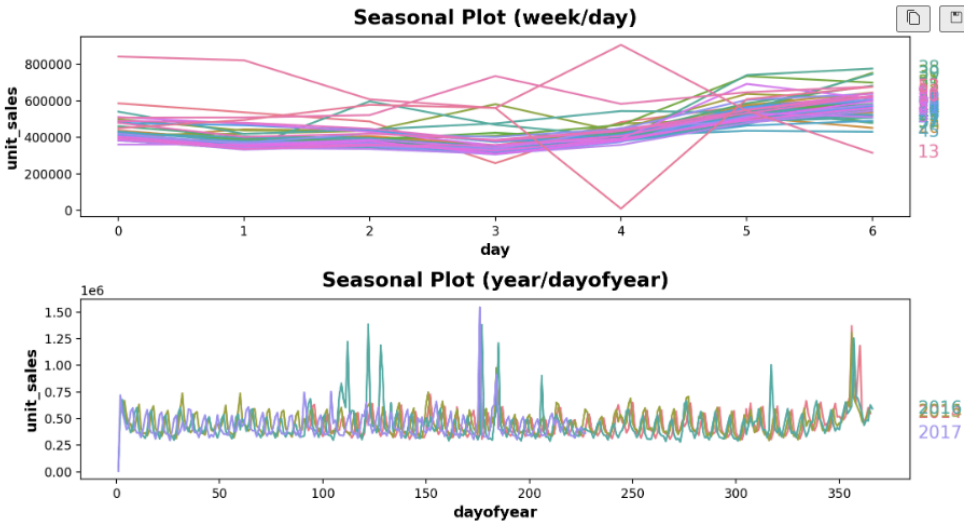
Esas son las nuevas columnas que se aregaran al dataset de entrenamiento.

Estacionalidad y tendencia de los datos

Una vez que se han realizado modificaciones al dataset para concentrarnos en la cantidad y distribución de valores. Tenemos la siguiente gráfica de las unidades vendidas:



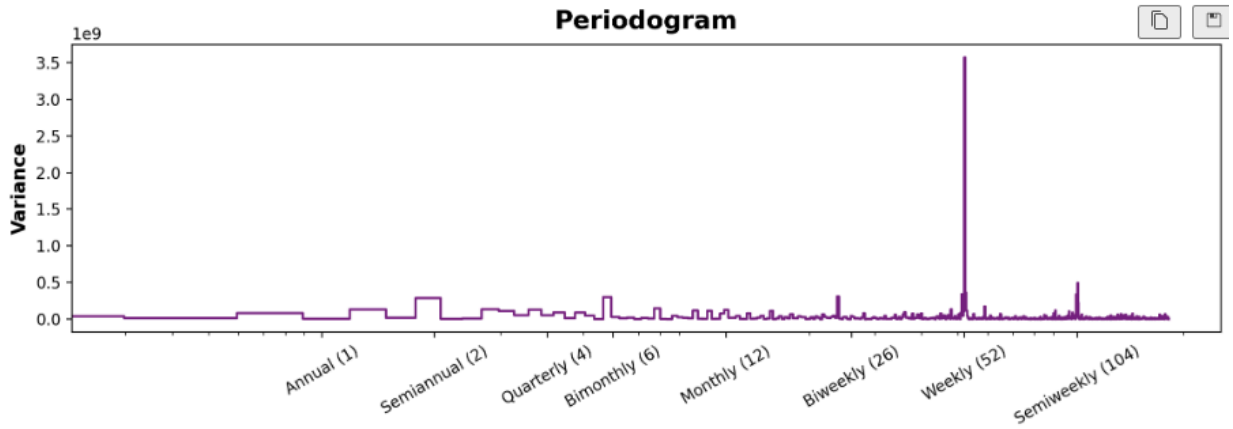
Como se puede observar la gráfica muestra picos y valles en fechas como inicio de año, mediados de año. La media muestral de los datos es una línea horizontal con pendiente cero. Lo que nos hable sobre la direcciónde los datos.



Dado que la estacionalidad semanal es fuerte, puede modelarla utilizando variables indicadoras (o ficticias). Estas variables capturarían el efecto semanal asignando diferentes valores para diferentes días de la semana (o cualquier otro subperíodo relevante dentro de la semana).

También se pude usar las características de Fourier para capturar patrones estacionales complejos, especialmente cuando la estacionalidad no es tan simple como un patrón semanal. Dado que la estacionalidad anual es más débil pero aún está presente, el uso de características de Fourier es un enfoque adecuado. Las series de Fourier pueden modelar funciones periódicas al descomponerlas en una suma de funciones seno y coseno, que puede capturar la naturaleza cíclica de los datos.

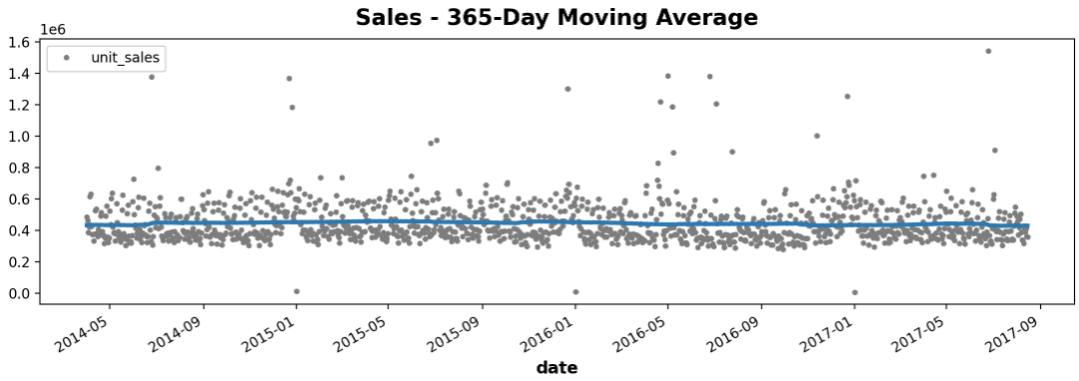
Selección del número de pares de Fourier: la varianza del periodograma disminuye entre "Bimestral (6)" y "Mensual (12)", lo que sugiere que la complejidad de la estacionalidad disminuye a medida que se pasa de períodos más cortos a períodos más largos. Al elegir 10 pares de Fourier, se busca capturar una parte suficiente de la estacionalidad anual sin sobreajuste. Esto significa que estás utilizando 10 funciones seno y 10 funciones coseno para aproximar la estacionalidad anual.

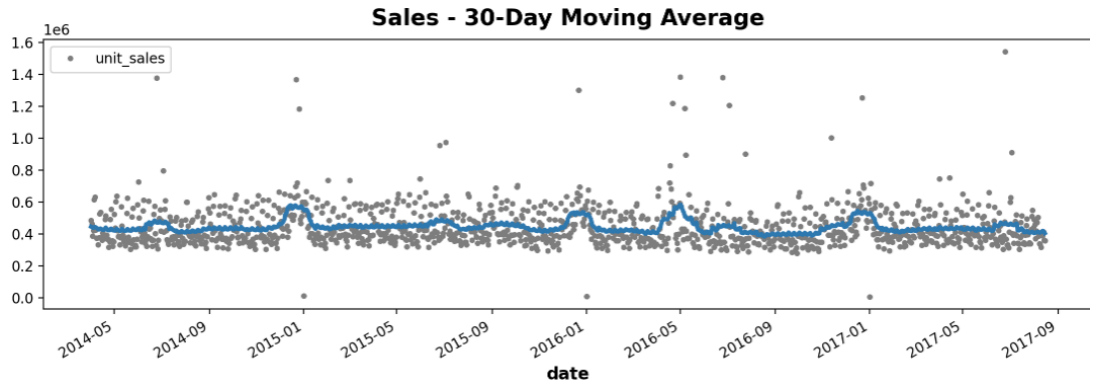


El periodograma muestra picos claros, en particular en la frecuencia "Semanal (52)", lo que indica que hay una cantidad significativa de variación en esta frecuencia. Esto sugiere que los datos de la serie temporal subyacente tienen un fuerte componente semanal.

Hay picos más pequeños en las frecuencias "Anual (1)" y "Semestral (2)", lo que indica cierto nivel de patrones estacionales o semestrales en los datos. Estos picos pueden representar tendencias o ciclos anuales o bianuales en los datos.

Para analizar la tendencia debemos considerar que las observaciones son diarias, se comenzara por analizar el comportamiento con una venta de 365 días.





Como se puede observar en la gráfica anteriores usando una ventade 30 días podemos obtener una mejor aproximación de la tendencia de las observaciones.

Para mejora la aproximación a la gráfica se usó DeterministicProcess y modelos de regresión con una ventana de 30 días para aproximar la prueba.

Modelos predictivos

Regresión Lineal

La regresión lineal es un tipo de algoritmo de aprendizaje automático supervisado que calcula la relación lineal entre la variable dependiente y una o más características independientes ajustando una ecuación lineal a los datos observados.

Descenso de gradiente estocástico (SGD)

Una técnica de optimización clave para entrenar modelos en aprendizaje profundo y aprendizaje automático es el descenso de gradiente estocástico (SGD). Utilizando un único punto de datos seleccionado aleatoriamente (o un pequeño lote de datos) en cada iteración, el SGD cambia los parámetros del modelo en contraste con los métodos clásicos de descenso de gradiente, que calculan el gradiente de la función de pérdida teniendo en cuenta todo el conjunto de datos.

Como resultado, se introduce cierta estocasticidad, que acelera y fortalece el proceso de optimización frente a datos ruidosos.

XGBoost

Extreme Gradient Boosting, o XGBoost para abreviar, es una implementación eficiente de código abierto del algoritmo de aumento de gradiente. XGBoost domina los conjuntos de datos estructurados o tabulares en problemas de modelado predictivo de clasificación y regresión.
Resultados

Como primer opción para aproximar la curva de las unidades se usó el méto del uso de las series de tiempo como features, es decir, desplazando las unidades de venta se puede lograr una aproximación a la curva original.

Resultados

SGD

Usando el retraso de un día para construir el conjunto de datos

	Metric	Train Set	Test Set	Model
0	RMSE	21.98	18.58	SGDRegressor
1	MAPE (%)	222.80	222.70	SGDRegressor
2	R2 Score	0.01	0.02	SGDRegressor
3	MAE	7.02	7.01	SGDRegressor
4	Relative Error to Mean (%)	274.50	232.30	SGDRegressor

Regresión lineal

- Las características de paso de tiempo son características que podemos derivar directamente del índice de tiempo. La característica de paso de tiempo más básica es la variable ficticia de tiempo, que cuenta los pasos de tiempo en la serie desde el principio hasta el final. Las características de paso de tiempo le permiten

	Metric	Train Set	Test Set	Model
0	RMSE	125967.89	125967.89	LinearRegression - time step features
1	MAPE (%)	30.30	30.30	LinearRegression - time step features
2	R2 Score	0.11	0.11	LinearRegression - time step features
3	MAE	83676.50	83676.50	LinearRegression - time step features
4	Relative Error to Mean (%)	28.30	28.30	LinearRegression - time step features

- Tendencia, el componente de tendencia de una serie temporal representa un cambio persistente y a largo plazo en la media de la serie.

	Metric	Train Set	Test Set	Model
0	RMSE	133566.23	133566.23	LinearRegression - trend
1	MAPE (%)	31.40	31.40	LinearRegression - trend
2	R2 Score	0.00	0.00	LinearRegression - trend
3	MAE	92617.52	92617.52	LinearRegression - trend
4	Relative Error to Mean (%)	30.00	30.00	LinearRegression - trend

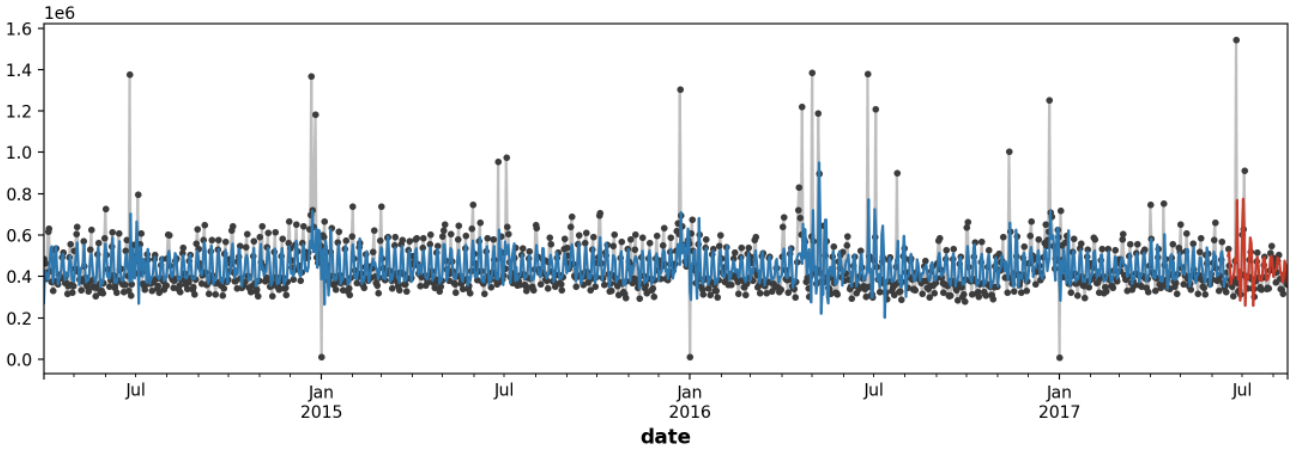
- Para crear una función de retardo, desplazamos las observaciones de la serie objetivo de modo que parezcan haber ocurrido más tarde en el tiempo. En términos más generales, las funciones de retardo permiten modelar la dependencia serial. Una serie temporal tiene dependencia serial cuando una observación se puede predecir a partir de observaciones anteriores. En Hardcover Sales, podemos predecir que las ventas altas en un día generalmente significan ventas altas al día siguiente.

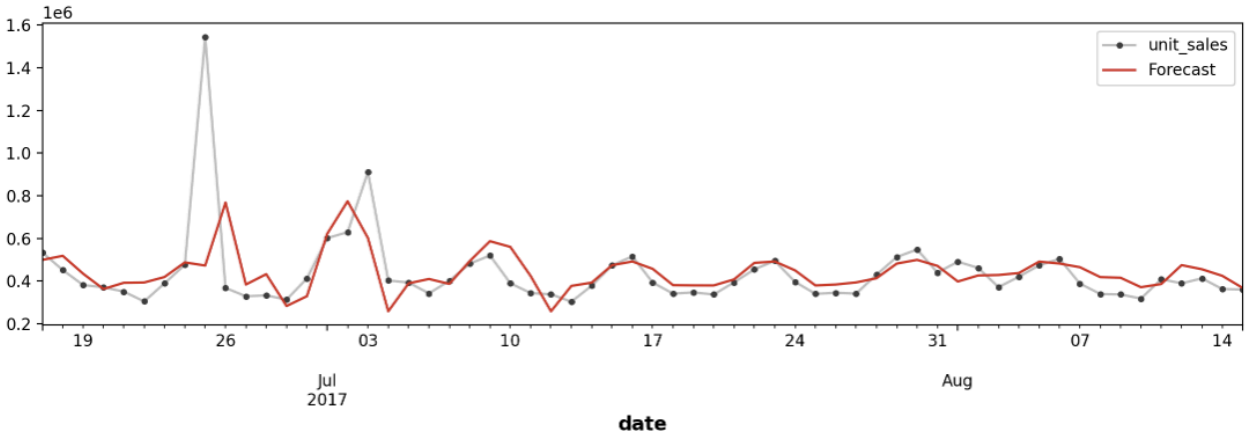
- Estacionalidad, hay dos tipos de características que modelan la estacionalidad.
    - El primer tipo, los indicadores, es mejor para una temporada con pocas observaciones, como una temporada semanal de observaciones diarias.
    - El segundo tipo, las características de Fourier, es mejor para una temporada con muchas observaciones, como una temporada anual de observaciones diarias.

	Metric	Train Set	Test Set	Model
0	RMSE	120737.59	120737.59	LinearRegression - seasonality
1	MAPE (%)	30.60	30.60	LinearRegression - seasonality
2	R2 Score	0.18	0.18	LinearRegression - seasonality
3	MAE	78065.66	78065.66	LinearRegression - seasonality
4	Relative Error to Mean (%)	27.10	27.10	LinearRegression - seasonality

- Retrasos, para investigar la posible dependencia serial (como los ciclos) en una serie temporal, necesitamos crear copias "retrasadas" de la serie. Retrasar una serie temporal significa desplazar sus valores hacia adelante uno o más pasos temporales o, equivalentemente, desplazar los tiempos en su índice hacia atrás uno o más pasos.

	Metric	Train Set	Test Set	Model
0	RMSE	109468.83	164024.87	LinearRegression - lags
1	MAPE (%)	22.80	15.60	LinearRegression - lags
2	R2 Score	0.30	0.11	LinearRegression - lags
3	MAE	63023.09	76463.14	LinearRegression - lags
4	Relative Error to Mean (%)	24.60	37.60	LinearRegression - lags

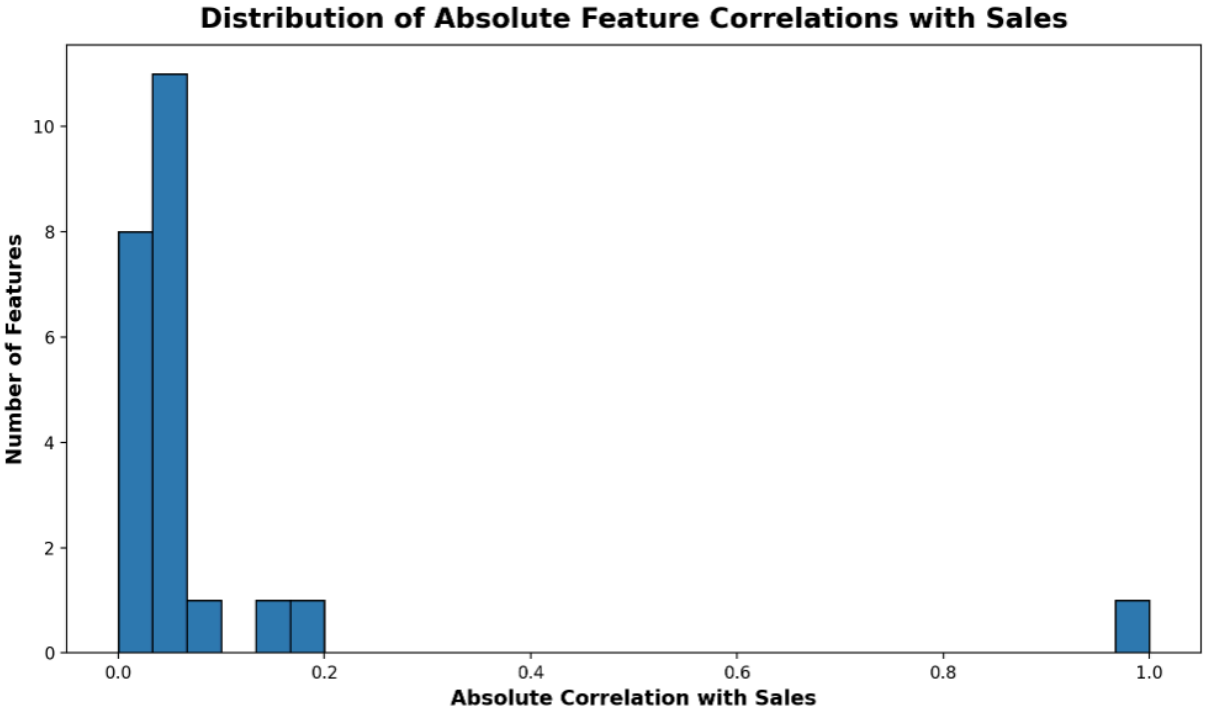




XGBoost

Para el entrenamiento de este modelo se utilizaron las estrategías vistas en los análisis anteriores, por ejemplo:

- Se consideraron agregar más features que solo el tiempo, se agregaron features de las tablas como: 'family', 'perishable', 'type_store', 'cluster', 'type_holiday', 'transferred'
- Para contruir el dataset se tomaron en cuenta tres elementos: tendencia, estacionalidad y ciclos




```
Features with low correlation to 'sales':
cluster         0.002603
type_holiday    0.007006
month           0.008667
payday          0.000055
Name: unit_sales, dtype: float64
```

El número total de features es: 'family', 'perishable', 'type_store', 'cluster', 'type_holiday', 'transferred', weekday, year, month, day, payday, is_weekend, sales_lag_7, sales_lag_30, sales_roll_mean_7, sales_roll_mean_30, sales_ewm_alpha_095_lag_7, sales_ewm_alpha_095_lag_30, sales_ewm_alpha_09_lag_7, sales_ewm_alpha_09_lag_30, sales_ewm_alpha_08_lag_7, sales_ewm_alpha_08_lag_30

Usando la matriz de correlación y eliminando las features menos relevantes se consideraron estas 24 columnas.

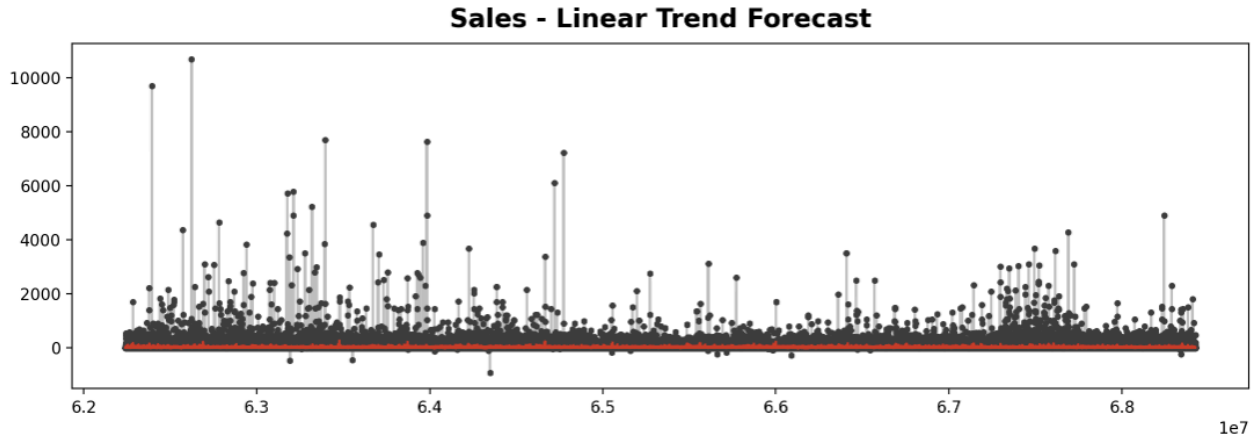
Se realizó una búsqueda de maya para obtener los mejores hiperparámetros:

Best parameters found with Optuna: {'n_estimators': 299, 'learning_rate': 0.010196240577626036, 'max_depth': 14, 'subsample': 0.6264220866247561, 'colsample_bytree': 0.8084523598374298, 'min_child_weight': 23, 'reg_lambda': 0.030801527526937537, 'colsample_bynode': 0.5649473449255678}

```
params = {
    # 'tree_method': 'gpu_hist',
    'tree_method': 'hist',
    'n_jobs': -1,
    'objective': 'reg:squarederror',
    'n_estimators': 299,
    'verbosity': 2,
    'learning_rate': 0.010196240577626036,
    'max_depth': 14,
    'subsample': 0.6264220866247561,
    'colsample_bytree': 0.8084523598374298,
    'min_child_weight': 23,
    'reg_lambda': 0.030801527526937537,
    'colsample_bynode': 0.5649473449255678
}

model = XGBRegressor(**params)
```

Después de 5 hrs de entrenamiento y pruebas el modelo no pudo realizar una buena aproximación sobre los valores reales.



Consideraciones

Cliente

1. Los factores holidays afectan la estabilidad en las ventas de productos, como se puede ver en las gráficas se pueden generar ventas demasiado grandes positivas o negativas, debemos observar que este tipo de afectaciones son anuales. si consideramos ventanas de tiempo mensuales o semanales estas características no determinan el comportamiento de las ventas diarias. Se recomienda considerar un intervalo de tiempo para el histórico de mayor tiempo (10 años por ejemplo).

2. Para desarrollar una estrategia para la reducción de costos de almacenamiento se debe proveer infomación sobre los costos de almacenamiento por tienda, familia o producto.
3. Para los productos no vendidos se debe de proveer datos de las perdidas o ganancias considerado productos perecederos o no, número de productos en promoción.
4. Revisando la lista de elementos con picos y valles se encontraron registros de holidays no considerados en la tabla holidays_events se recomienda actualizar los eventos.
5. En el análisis de importancia de características se pueden mencionar lo siguiente:
    a. perisable: Se recomienda crear un servicio que alerte de productos próximos a caducar para crear estrategias de promoción o creación de nuevos productos
    b. ty_store: Los productos se van a ver afectados dependiendo de su tipo de tienda. así que, la estrategia deberá depender de su localización, giro y de los productos que maneja
    c. family: Los productos se ven afectados dependiendo de la familia que pertenezcan y considerar la familia podrá ayudar a determinar el tipo acción para evitar picos o valles

Científico de datos

1. Realizar una recolección de requisitos para actualizar la etiqueta de cada columna en las tablas
2. Definir la unidad de medida de la característica unit_store