CARRERA: ESPECIALIZACIÓN EN CIENCIA DE DATOS

TRABAJO FINAL INTEGRADOR

TITULO

Optimización de modelos predictivos para Series de Tiempo Jerárquicas (HTS).

Nombre y Apellido del Alumno/a: Patricio Alex Woodley

Título de grado o posgrado (último): Ingeniería Industrial

Profesores:

Dra. Marcela Riccillo

Lugar y Fecha: Ciudad de Buenos Aires 21 de Octubre 2021

Resumen. Las series de tiempo jerárquicas (HTS) son un subconjunto de series de tiempo que están contenidas dentro de una estructura o jerarquía. Esta estructura puede estar dada por características propias del producto/servicio, como su categoría, departamento, etc., o por una división geográfica, como por ejemplo ciudad, estado o país. Las técnicas clásicas como ARIMA o ETS no son óptimas para este tipo de conjunto de series, ya que la estructura puede aportar información valiosa que pasaría inadvertida o al menos no tan fácilmente identificable si modelamos las series de tiempo por separado. Debido a esto, se comparó el desempeño de Redes Neuronales LSTM, las cuales son ampliamente utilizadas para series de tiempo, y de técnicas específicas para series de tiempo jerárquicas como Bottom-Up y Reconciliación Óptima para predecir las ventas de departamentos dentro de tiendas de Walmart en EE.UU. Se analizaron en detalle 70 series de tiempo a nivel tienda-departamento y se demostró la eficacia de estas técnicas para cada una de ellas.

Abstract. Hierarchical Time Series (HTS) are a subset of time series that are contained inside of a structure or hierarchy. This kind of structure may be defined by important characteristics of the product/service, like its category or department, or it may be defined by some geographic division, for instance, city, state, country and so on. Classic approaches for time series, like ARIMA or ETS are suboptimal for this kind of series due to the fact that this hierarchy could give us interesting information for forecasting that may be hidden or not easily identifiable if we model all the series separately. Because of that, we compared the performance of LSTM Neural Networks, which are widely used for time series, and some specific approaches for hierarchical time series as Bottom-up and Optimal Reconciliation in order to forecast department sales for Walmart stores in the USA. We performed an in-depth analysis for 70 store-departamental time series and showed the efficacy of these techniques for each one.

Indice

1. Introducción

Existen series de tiempo que pueden ser desagregadas por ciertas variables de interés. Estas variables pueden ser características asociadas al bien o servicio que representan, como su categoría, departamento, etc, o cuestiones geográficas como ciudad, provincia o país. Este tipo de series de tiempo que se encuentran dentro de una estructura o jerarquía son llamadas Series de Tiempo Jerárquicas (HTS).

En este trabajo, se analizarán las ventas de 10 tiendas de la cadena Walmart repartidas en 3 estados de Estados Unidos. Dada la naturaleza de los productos que vende Walmart, éstos se encuentran dentro de una estructura jerárquica que los organiza. Es decir, tenemos departamentos y categorías que nos definen si un producto es comida, cosas para el hogar, pasatiempos, etc

Una correcta estimación de las ventas futuras es fundamental para el planeamiento de cualquier empresa, en especial para aquellas en las que la demanda es fluctuante a lo largo del tiempo. En particular para Walmart, con 11.000 tiendas bajo 65 marcas en 28 países del mundo, una predicción precisa de sus miles de productos significa una disminución en costos de transporte, distribución, mantenimiento y almacenamiento de productos al disminuir el sobre inventario. En caso contrario, una previsión por debajo de la demanda real puede derivar en oportunidades perdidas y clientes insatisfechos.

Ha habido grandes progresos en este campo gracias a la introducción de algoritmos de Machine Learning, como Redes Neuronales o Gradient Boosting. El objetivo de este trabajo es comparar modelos que tienen antecedentes de tener un poder predictivo sobre series de tiempo complejas, como Redes Neuronales LSTM, y ver cuál es el que mejor se adapta a este tipo de series de tiempo jerárquicas.

2. Estado del Arte

Los métodos clásicos estadísticos para predecir series de tiempo son ARIMA (AutoRegressive Integrated Moving Average) en el caso de métodos lineales y GARCH (Generalized AutoRegressive Conditional Heteroskedasticity) para métodos no lineales. Aunque no suelen tener la flexibilidad suficiente para reflejar los patrones inherentes de la serie de tiempo.

En los recientes años se han utilizado algoritmos no paramétricos (Machine Learning) para predecir series de tiempo con resultados prometedores. Como por ejemplo en Ghassen Chniti, Houda Bakir, and Hédi Zaher (2017) [4], donde predicen precios de teléfonos en mercados europeos, las Redes Neuronales LSTM y SVM son los modelos de Machine Learning más utilizados.

Además, como en Mergani A. Khairalla and Xu Ning. (2017) [8], donde analizan el tipo de cambio real entre el euro y la libra sudanesa, se utilizaron modelos híbridos entre los métodos clásicos y los actuales de modo tal que se intenta utilizar lo mejor de ambos contextos, obteniendo mejores métricas que los modelos por separado.

Finalmente, en los últimos años, otros modelos que han ganado particular importancia en las series de tiempo son los llamados Ensemble, como los algoritmos de Gradient Boosting, por ejemplo Light GBM y XGBoost, y Stacking para mejorar las métricas de los modelos. Además, la técnica Ensemble se ha utilizado con diferentes algoritmos, como se menciona en A. O. Akyuz, M. Uysal, B. A. Bulbul and M. O. Uysal (2017) [1], para predicciones de demanda en la cadena minorista SOK en Turquía.

Esta técnica se basa en utilizar más de un modelo para predecir los valores objetivo. Estos modelos deben ser independientes entre sí, de modo tal que los errores de un modelo sean compensados por los aciertos de otro. En el caso de la regresión, la predicción final es el promedio de cada uno de los resultados de todos los modelos.

En cambio, en la técnica Stacking, los resultados de varios modelos (que pueden pertenecer al mismo algoritmo o no) se utilizan como datos de entrada de un modelo final que luego le servirá como información para obtener una predicción más precisa de la serie de tiempo.

Estas dos técnicas tienen como ventajas poder generar predicciones más robustas, con menor varianza y un menor sesgo, pero al mismo tiempo, es cada vez más difícil poder explicar cómo llegan a esas conclusiones y operan como cajas negras.

3. Definición del Problema

En el presente trabajo, la serie de tiempo es jerarquizada. Es decir, los datos se encuentran desagregados por varias características de interés. Debido a esto, la información se describe a nivel producto, categoría, departamento y detalle de las tiendas. Por ello debe realizarse un tratamiento especial para este tipo de series de tiempo. Como es explicado en Athanasopoulos, G. and Kourentzes, N. (2020) [2] y en Hyndman, R. J., Ahmed, R. A., Athanasopoulos, G., & Shang, H. L. (2011) [7], debe haber una coherencia en la predicción de las ventas a través de los niveles de la jerarquía. Es decir, la sumarización de los nodos del nivel más bajo próximo debe ser igual a cada uno de los nodos del nivel de estudio.

Para ello existen diversas técnicas para abordar este problema, como vemos en Hyndman, R.J., & Athanasopoulos, G. (2018) [5] : Bottom-up, Top-Down, Middle-Out y Reconciliación Óptima.
En Bottom-up, se generan las predicciones en el último nivel de la jerarquía y luego se suman para producir predicciones en los niveles más altos.
En cambio, para Top-Down, envuelve generar una una predicción para el total de la serie de tiempo y luego se desagrega hacia abajo en la jerarquía a través de pesos ponderados.
Además, existe una combinación de ambos llamada Middle-Out. En este caso, se elige un nivel de la jerarquía y se generan predicciones para cada una de las series de tiempo pertenecientes a ese nivel. Luego se aplica Bottom-up para las series superiores al nivel elegido y Top-Down para las inferiores.

Por último, un método superador de estas técnicas, es la reconciliación óptima, en donde se busca minimizar el error de las predicciones coherentes, es decir, de las predicciones base (modeladas por separado) luego de haber sido reconciliadas para tener en cuenta la estructura jerárquica.
Este tipo de problemática no es sencilla de predecir, dada la alta variabilidad de las ventas para cada una de las series que representan el último nivel. Aunque ha habido grandes avances, no existe aún una técnica satisfactoria para abordar el problema.
En este trabajo se modelará la problemática de Walmart, enfocándonos en las ventas de productos en 10 tiendas pertenecientes a 3 estados en EE.UU. (California, Wisconsin y Texas).
Walmart es la mayor cadena de tiendas en el mundo, como vemos en (3), con casi 11.000 tiendas bajo 65 marcas en 28 países. Es el minorista más grande del mundo, con sede principal en Bentoville, Arkansas. Poder predecir cuánto venderá cada una de sus tiendas por día es de vital importancia para la empresa dado que una buena predicción conducirá a un buen manejo de los inventarios o a un mejor nivel de servicio. En cambio, una predicción errónea se traduce en oportunidades perdidas, como la pérdida de una venta debido a la falta de stock o mayores costos por exceso de inventario (debido a su manipulación y almacenamiento).

4. Justificación del estudio

En los últimos años se han aplicado diversas técnicas para series de tiempo, pero no existe un estudio aún que sea satisfactorio para series de tiempo enmarcadas dentro de una estructura jerárquica, y en particular para una tienda que vende miles de productos como Walmart.
Esta estructura con diferentes niveles nos proporciona información que puede ser de gran utilidad para predecir ventas de cada una de ellas. Las series que comparten un departamento, categoría, tienda o estado tienen características compartidas que pasarían inadvertidas o al menos no serían fácil de identificar, si modelamos todas por separado y sin tener en cuenta la jerarquía.
Por ello, un enfoque clásico de series de tiempo sería insuficiente para poder modelar de forma correcta este grupo de series. Las Redes Neuronales LSTM pueden aprender cómo se relacionan todas las series dentro de la estructura jerárquica, dado que se entrenan todos las series al mismo tiempo, y de aprender que tanto dependen de sus desfases. De igual manera, la técnica de series de tiempo jerárquicas (HTS) toman como base a los distintos clásicos como ARIMA, para capturar estas relaciones con sus desfases y le suman una capa adicional para reconciliar las series según la estructura jerárquica en la que se encuentran. Por lo tanto, tanto LSTM como HTS son buenos candidatos para modelar este tipo de problemática donde la estructura en la que se encuentran las series de tiempo ocupa un rol importante.

5. Alcances del trabajo y limitaciones

El estudio analizará las ventas de 3049 productos repartidos en 3 categorías (Comidas, Hobbies y Productos del Hogar) y 7 departamentos para 10 tiendas localizadas en 3 estados de Estados Unidos: California, Wisconsin y Texas. El dataset proviene de una competencia realizada por Kaggle [23].

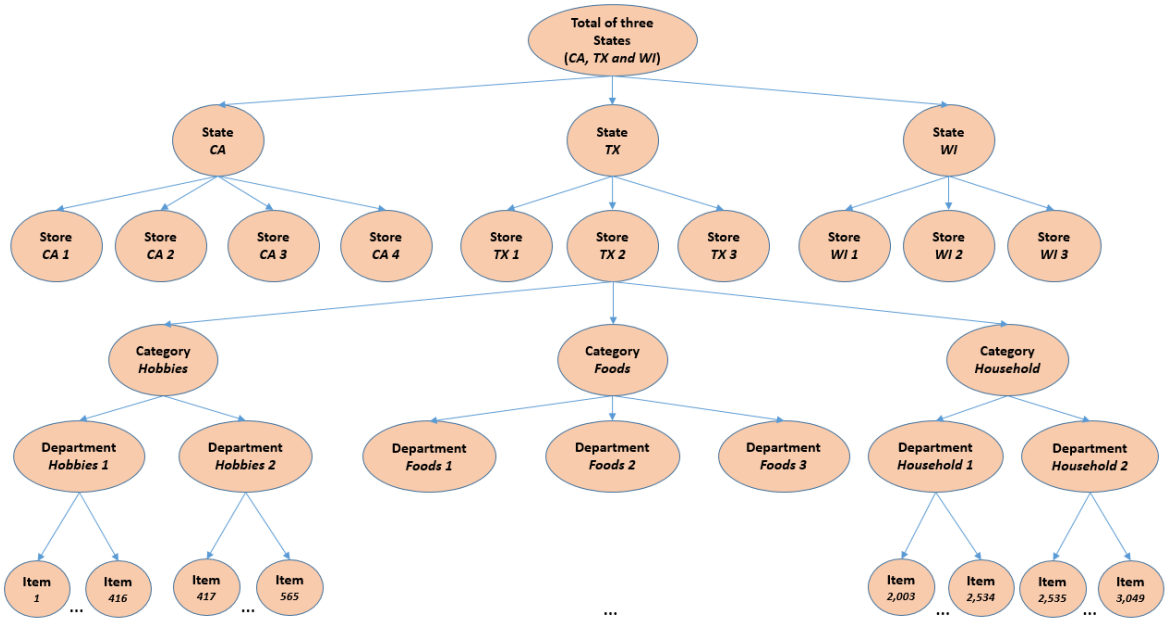

Gráfico 5.1 Estructura jerárquica de las tiendas de Walmart
***Fuente**: https://mofc.unic.ac.cy/m5-competition/*

La información histórica comprende desde el 29 de enero de 2011 hasta el 19 de junio de 2016.
Siendo información de casi 5 años para más de 3000 productos, existe una fuerte limitante en el hardware utilizado para poder procesar todo en memoria.
Dada esta limitante en el hardware, se analizará la información a nivel tienda-departamento, de manera tal que se puede tener toda la información en memoria.

6. Hipótesis

- La técnica de Series de Tiempo Jerárquicas (HTS) es más eficiente para la predicción de ventas de una tienda de Walmart, a lo largo de un mes, que modelos de machine learning especializados para series de tiempo, como las Redes Neuronales LSTM, ya que toman en cuenta la información que les proporciona la jerarquía.

Variables dentro de la hipótesis:

- Ventas de una tienda.
- Predicción de la venta de una tienda.
- Eficiencia de una técnica en series de tiempo.

Ventas de una tienda

- Definición nominal: son todos los intercambios de productos realizados por la tienda a lo largo de un mes.
- Definición operacional: es la sumatoria de las ventas de todos los productos de una tienda a lo largo de un mes.
- Medición cuantitativa: Se miden en unidades vendidas.

Predicción de las ventas de una tienda

- Definición nominal: es una estimación a través de un modelo predictivo de las ventas totales que genera una tienda a lo largo de un mes.
- Definición operacional: es la sumatoria de las predicciones de las ventas de todos los productos de una tienda a lo largo de un mes.
- Medición cuantitativa: Se miden en unidades vendidas.

Eficiencia de una técnica en series de tiempo

- Definición nominal: es la capacidad de predicción del modelo. Es decir, que tanto se ajusta la predicción al valor real de la serie de tiempo.
- Definición operacional: es la sumatoria de los errores de predicción. Cuánto menor error, mayor la eficiencia.
- Medición cuantitativa: Se mide a través de la métrica RMSE (raíz del error cuadrático medio.

7. Objetivos

7.1 Objetivo General

- Desarrollar un modelo predictivo que mejor se ajuste para la predicción de ventas jerárquicas para una tienda de Walmart en un mes, utilizando como métrica RMSD o RMSE (raíz del error cuadrático medio).

$$\mathrm{RMSD} = \sqrt{\frac{\sum_{t=1}^{T} (\hat{y}_t - y_t)^2}{T}}.$$

Donde Y_t es el valor actual futuro de la serie de tiempo examinada en el tiempo t, $\hat{Y}_t$ la predicción generada y T el tamaño de la muestra de entrenamiento (número de observaciones históricas).

7.2 Objetivos Específicos

- Desarrollar modelos de HTS para la predicción de las ventas.
- Elegir el candidato de menor RMSE como representante del primer conjunto.
- Desarrollar modelos de Redes Neuronales LSTM para la predicción de estas ventas.
- Elegir el candidato de menor RMSE como representante del segundo conjunto.
- Elegir el modelo con mejor performance de todos los conjuntos.

8. Metodología

Para los distintos tipos de modelos se utilizarán las siguientes técnicas:

- Técnicas de Machine Learning:
    1. Redes Neuronales Recurrentes (en particular LSTM).

- Técnicas para el manejo de de Series de Tiempo Jerárquicas (HTS):
    1. Bottom-Up.
    2. Reconciliación óptima.

8.1 Modelos ARIMA

Box & Jenkins (1976), como se ve en Hyndman, R.J., & Athanasopoulos, G. (2018) [5], generaron una metodología para la predicción de series de tiempo a través de los modelos autorregresivos integrados de media móvil (ARIMA), los cuales utilizan valores pasados de estas series. La principal suposición de estos modelos es que la serie de tiempo es estacionaria, es decir que las propiedades de ésta no cambian a través del tiempo. Estos modelos se descomponen en 3 partes principales:

- Parte Autorregresiva: Indica si la serie es dependiente de sus desfases
- Parte Integrada: Indica si es necesario diferenciar a la serie de tiempo para que pueda ser considerada estacionaria.
- Parte de Media Móvil: Indica si el error de la regresión es una combinación lineal de los errores del pasado pertenecientes a un proceso estocástico.

8.2 ETS

Son una familia de series de tiempo en los que se descompone a la serie en 3 términos:

- Error
- Tendencia
- Estacionalidad

Estos términos pueden ser considerados aditivos o multiplicativos entre sí según el tipo de modelo, como se detalla en Hyndman, R.J., & Athanasopoulos, G. (2018) [5].

8.3 Series de Tiempo Jerárquicas (HTS)

Las series de tiempo jerárquicas (*Hierarchical Time Series* en inglés) son series que pueden ser desagregadas por una o más variables de interés. Esta variable puede ser de tipo geográfico como ciudad, estado o provincia o de acuerdo a una característica o atributo de la variable. En Athanasopoulos, G., Ahmed, R. A., & Hyndman, R. J. (2009) [6] y Hyndman, R. J., Ahmed, R. A., Athanasopoulos, G., & Shang, H. L. (2011) [7] se desarrolló una metodología particular para este tipo de series.

La idea principal de estos métodos es obtener unas predicciones base, en las que las series de tiempo se modelan por separado, para luego reconciliarse entre sí de modo tal de producir predicciones coherentes con la estructura jerárquica en la que se encuentran. Es decir, la suma de las series de tiempo de un nivel tiene que ser coherente con las del nivel inmediato superior.

Las diversas técnicas que se detallan en Hyndman, R.J., & Athanasopoulos, G. (2018) [5], son:

- Bottom-up: simplemente se predicen las series de tiempo del nivel más bajo de la estructura y luego se suman para predecir las series de niveles más altos.
- Top-Down: primero se intenta predecir la serie de tiempo que se encuentra más arriba en la jerarquía y luego se desagrega en los niveles inferiores utilizando proporciones para cada serie de tiempo del nivel inferior.
- Middle-Out: es una combinación de los dos enfoques anteriores. Primero se elige un nivel del medio de la estructura jerárquica y se generan predicciones para este nivel. Para las series de tiempo arriba de este nivel se utiliza el enfoque Bottom Up, sumando las predicciones del nivel medio. Para las series que se encuentran en un nivel por debajo del nivel medio, las predicciones se realizan utilizando el enfoque Top Down
- Reconciliación óptima: los 3 enfoques anteriores se pueden generalizar de forma matricial utilizando la fórmula $\bar{y}_h = S\,G\,\hat{y}_h$

  Donde G es una matriz que mapea las predicciones base en el nivel inferior y la matriz S los suma utilizando la estructura de agregación para producir predicciones coherentes $\bar{y}_h$.

  La reconciliación óptima ocurre cuando obtenemos una matriz G que minimiza el error en las predicciones coherentes.

8.4 Redes Neuronales Recurrentes

La principal característica de las Redes Neuronales Recurrentes (RNN), y que las distingue de los demás tipos de Redes como se especifica en Goodfellow, I., Bengio, Y., & Courville, A. (2016) [12], es que tienes un bucle o loop dentro de ellas. Esta característica les da la capacidad de reconocer dependencias temporales y espaciales. Por ello son ampliamente utilizadas para series de tiempo o NLP (procesamiento de lenguaje natural), entre otras cosas, donde la información pasada aporta información clave para predecir el futuro.

8.4.1 Redes Neuronales LSTM (Long Short-Term Memory)

Existe un tipo de Red Neuronal Recurrente llamada LSTM, que es capaz de captar dependencias de largo término (es decir que sucedieron varios pasos atrás en el tiempo). Esta característica hace que sean ampliamente utilizadas para series de tiempo. Fueron introducidas por Hochreiter & Schmidhuber (1997) [11].

La estructura básica de las Redes LSTM se compone de una celda Estado y 3 compuertas, como vemos en el gráfico 8.1:

- Celda Estado: es la parte principal de la Red, donde se remueve o agrega información.
- Compuerta de olvido: se define qué información previa se olvida y cual se mantiene en la celda Estado.
- Compuerta de nueva información: se define qué información se agrega a la celda Estado.
- Compuerta de devolución de Vector de salida: Se filtra la celda Estado y se devuelve el vector final.

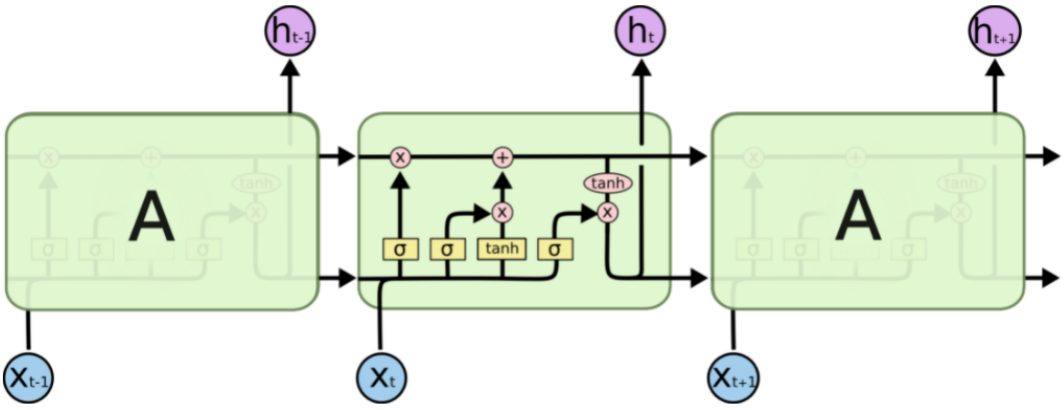

Gráfico 8.1 Estructura básica de una Red Neuronal LSTM

Fuente: *https://colah.github.io/posts/2015-08-Understanding-LSTMs/img/LSTM3-chain.png*

9. Experimentación

9.1 Agregación de nivel producto a nivel departamental

La estructura jerárquica tiene 5 niveles: Estado, Tienda, Categoría, Departamento y Producto.
Los valores posibles para cada uno de los niveles son:
Estados : California, Texas y Wisconsin
Tiendas: CA1, CA2, CA3, CA4, TX1, TX2, TX3, WI1, WI2, WI3
Categorias: Foods, Household, Hobbies
Departamentos: FOOD1, FOOD2, FOOD3, HOUSEHOLD1, HOUSEHOLD2, HOBBIES1, HOBBIES2

Cada serie de tiempo lleva en su nombre la siguiente codificación:

ESTADO | NRO TIENDA | CATEGORÍA | NRO DEPARTAMENTO

Por ejemplo, la serie de tiempo CA1FOO1 es el Departamento FOOD1 de la categoría Food perteneciente a la Tienda CA1 en California.

Todos los departamentos y categorías están en las 10 tiendas (no hay departamentos exclusivos por tienda).

A nivel producto, existen 3049 series de tiempo (una por cada producto separado por tienda). Teniendo en cuenta esto, y las dificultades de tener tanta información en memoria, se seleccionó comparar ambos algoritmos a nivel departamental. Es decir, agregando las unidades de los productos vendidos al departamento que les corresponde.
Al hacer esta agregación, este estudio analizará 70 series de tiempo (7 departamentos para cada una de las 10 tiendas).

9.2 Análisis Exploratorio (EDA)

Se realizó un análisis de la importancia de las distintas categorías, tiendas y estados en las ventas históricas en el período comprendido entre 29-01-2011 y el 24-04-2016.
Todos los gráficos se realizaron utilizando las siguientes librerías de Python, desarrollado por G. van Rossum (1995) [13]:

- Pandas [17] y Numpy [14] para preprocesamiento
- Matplotlib [15] y Seaborn [16] para crear los gráficos

Existe información dentro de la estructura jerárquica que puede ser valiosa para la predicción de ventas. Aunque sólo queremos predecir a nivel tienda-departamento, saber a qué categoría o a qué estado pertenece, nos da información que pasaría inadvertida si las series se modelan por separado y sin tener en cuenta las demás.

De las 3 categorías pertenecientes al estudio, Foods es la que más vendió a lo largo de los años.

9.2.1 Análisis de las ventas

Se realizó un análisis de la importancia de las ventas a nivel estado, tienda, categoría y producto. De esta manera, se puede visualizar diferencias entre los mismos y si hay, por ejemplo, tiendas que expliquen la mayoría de las ventas de un estado.

9.2.1.1 Ventas totales por categoría

Un 58% de las ventas se realizaron en la categoría Foods, mientras que en segundo lugar se encuentra la categoría Household con casi un 30%, como se observa en el gráfico 9.1.

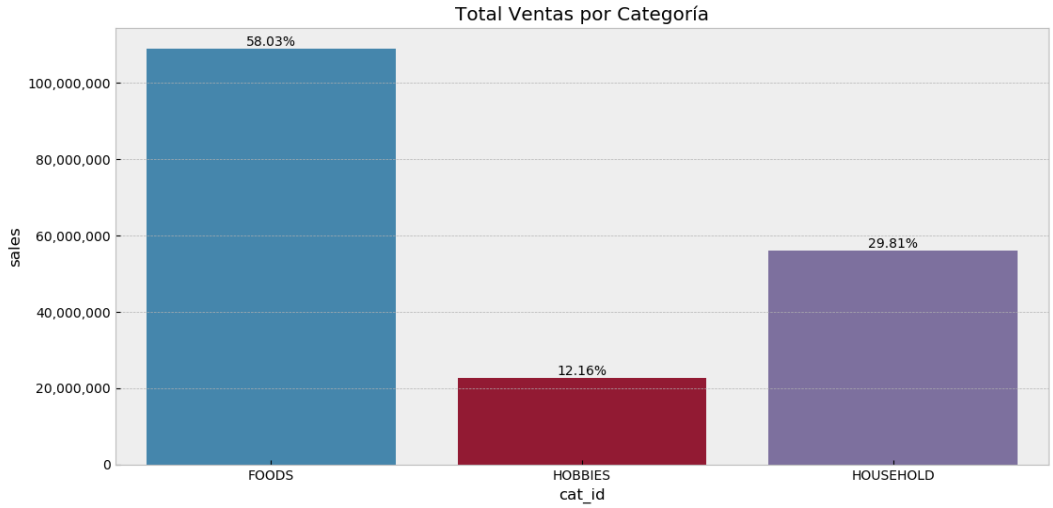

Gráfico 9.1. Ventas total por categoría.

9.2.1.2 Ventas por estado

Si analizamos las ventas por estado, como se ve en el gráfico 9.2, California se lleva el 45%. Este estado es el más grande de EE.UU. en cuanto a población, con casi 40 millones de personas en 2019, seguido por Texas con alrededor de 29 millones de personas.

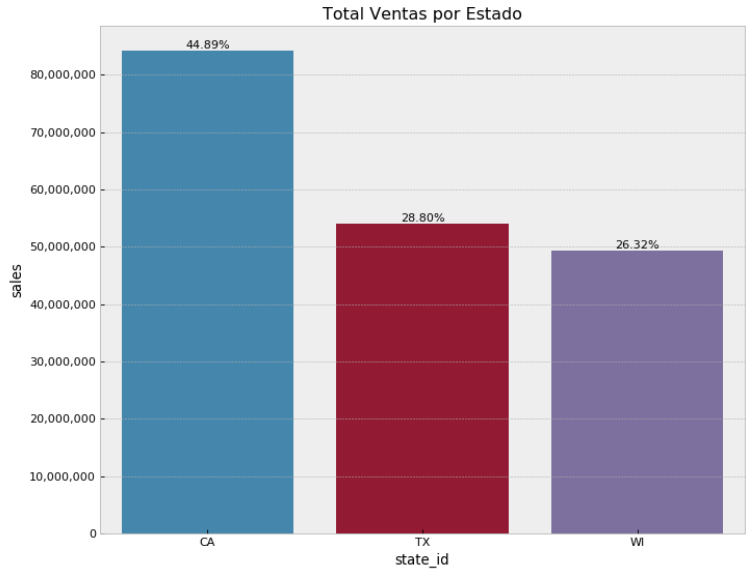

Gráfico 9.2. Ventas por estado.

9.2.1.3 Ventas por tienda para cada estado.

En cambio, como vemos en el gráfico 9.3, vemos que CA_3 es la tienda que más vendió por amplia diferencia.

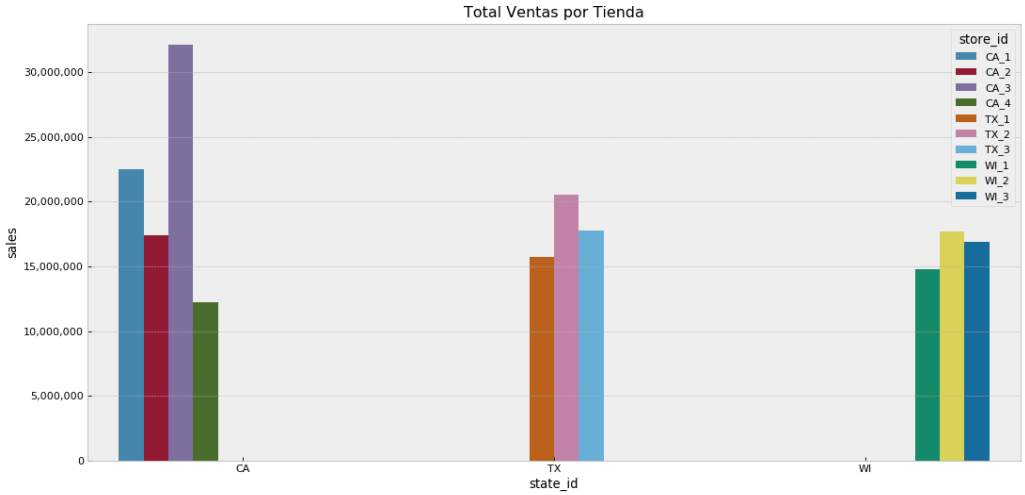

Gráfico 9.3. Ventas por tienda para cada estado.

En los estados de Texas y Wisconsin, las tiendas tienen ventas similares entre sí. Se puede observar que las demás tiendas de California tienen ventas similares a la de los demás estados.

9.2.1.4 Productos con más ventas históricas

Por último, del top 10 de productos con más ventas históricas, 9 pertenecen a Foods y 1 a Hobbies, como se observa en el gráfico 9.4.

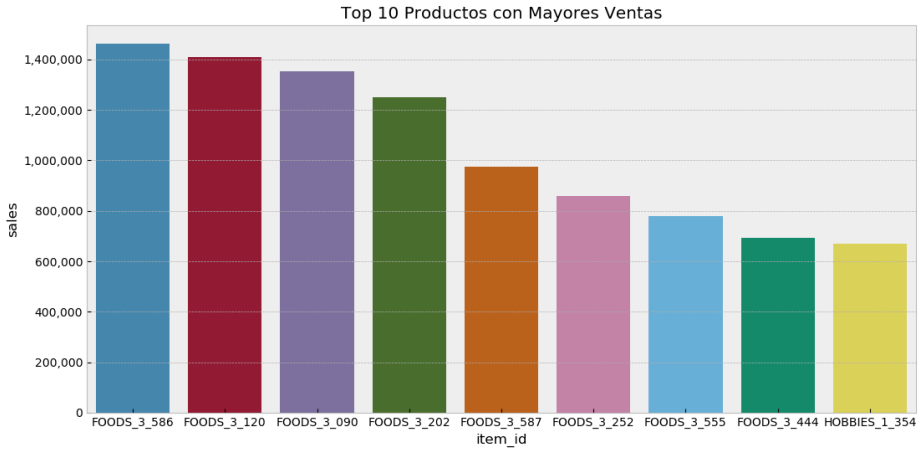

Gráfico 9.4. Los 10 productos con más ventas históricas.

9.2.2 Análisis de las series de tiempo

Para entender las características de estas series de tiempo, se analizará una serie representativa de cada nivel y de las 70 series de tiempo a nivel departamental, con la siguiente información:

- Gráfico de ventas diarias desde 2015-01-01 hasta 2015-12-31.
- Gráficos de ACF y PACF (Autocorrelación y autocorrelación parcial).
- Gráfico donde se descompone la serie de tiempo en tendencia, estacionalidad y residuo.
- Otras agrupaciones interesantes.

9.2.2.1 Autocorrelación (ACF) y Autocorrelación parcial PACF

La autocorrelación, como se detalla en Hyndman, R.J., & Athanasopoulos, G. (2018) [5], es una característica importante en una serie de tiempo y nos indica cuál es la dependencia que tiene un punto Xt respecto a los puntos anteriores Xt-n siendo n = 1,2,3…
Este indicador se calcula como la correlación de la serie de tiempo respecto a la misma desfasada en t = n.
Si por ejemplo queremos calcular la autocorrelación con un lag (o desfase) de 1, la fórmula es la siguiente:

$$R(k) = \frac{E[(X_i - \mu) - (X_{i-1} - \mu)}{\sigma^2}$$

Si la autocorrelación es alta, Xt-1 es un candidato a ser un buen regresor para predecir el valor de Xt. Además es capaz de mostrar tendencias semanales o mensuales si Xt depende de un desfase a 7 o 30 días por ejemplo.

En cambio, la autocorrelación parcial es la autocorrelación entre la serie de tiempo respecto a la misma desfasada en t= n pero descontando los valores de los intervalos intermedios. Dicho de otra manera, es la autocorrelación entre Xt y Xt-n que no se explica por retrasos de 1 a t-1, inclusive.

$$\alpha(1) = Cor(X_t, X_{t-1})$$

$$\alpha(k) = Cor(X_{t-n} - P_{t,n}(X_{t-n}), X_t - P_{t,n}(X_t)), \text{ para n >= 2}$$

Donde $P_{t,n}(X)$ denota la proyección de x en el espacio abarcado por X_{t-1},..... X_{t-n-1}

Estas características son usadas dentro de la metodología de Box-Jenkins para determinar las partes autorregresivas (AR), los cuales indican la correlación lineal con sus desfases, y otra de media móvil (MA), la cual indica si X_t depende de valores estocasticos anteriores, en los modelos ARIMA.

9.2.2.2 Análisis de las serie de tiempo Ventas Totales

Primero se analizará la serie de tiempo de Ventas Totales para el año 2015. Para tener un panorama inicial de cómo se comportan estas series de tiempo, se suman las ventas totales de las 10 tiendas de modo tal que se puedan observar estacionalidades generales y no de una categoría o tienda en particular.
Cómo se observa en el gráfico 9.5, existe una fuerte estacionalidad semanal. Además se señalan algunas fechas con eventos importantes. Lo interesante es que muchos de estos eventos generan picos mínimos o máximos (otros generan el pico el día anterior) y cómo veremos más adelante, influyen de manera distinta según el estado, tienda, categoría o departamento. Esta diferencia en patrones dificulta el modelado ya que no es igual para todas las series de tiempo. En este caso, El Día de Acción de Gracias es uno de los puntos más bajos en ventas y Navidad es el único día del año en el que Walmart cierra sus tiendas.

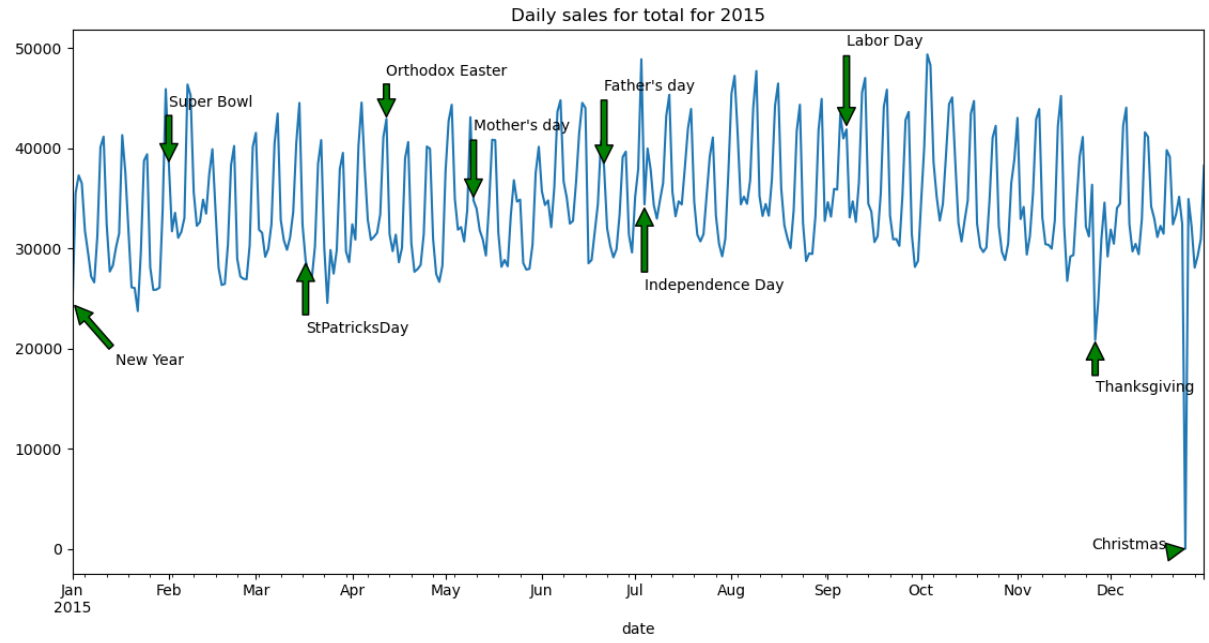

Gráfico 9.5. Ventas diarias para todas las tiendas del estudio en el 2015.

Se observa en el gráfico 9.6 que hay una estacionalidad marcada a nivel semanal, y una tendencia creciente desde enero hasta octubre y luego decae en los últimos meses del año.
Esto nos da una idea de la naturaleza de esta serie de tiempo. Una feature que nos diga el día de la semana seguramente tenga más sentido que el mes en el que se encuentra.

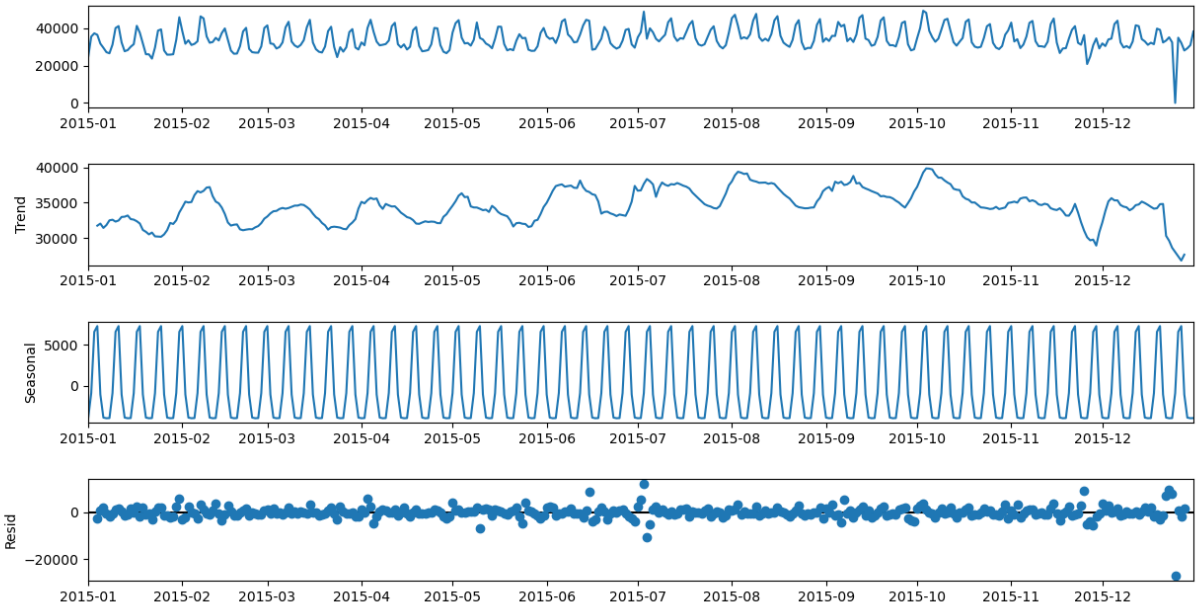

Gráfico 9.6. Ventas diarias para todas las tiendas descompuesta en tendencia, estacionalidad y residuos.

El gráfico 9.7 nos muestra otra forma de ver la estacionalidad semanal de la serie de tiempo, dado que hay una fuerte correlación en los desfases 7, 14, 21, etc. Además es significativa la dependencia con la venta en el día anterior (lo cual es lógico salvo eventos especiales).

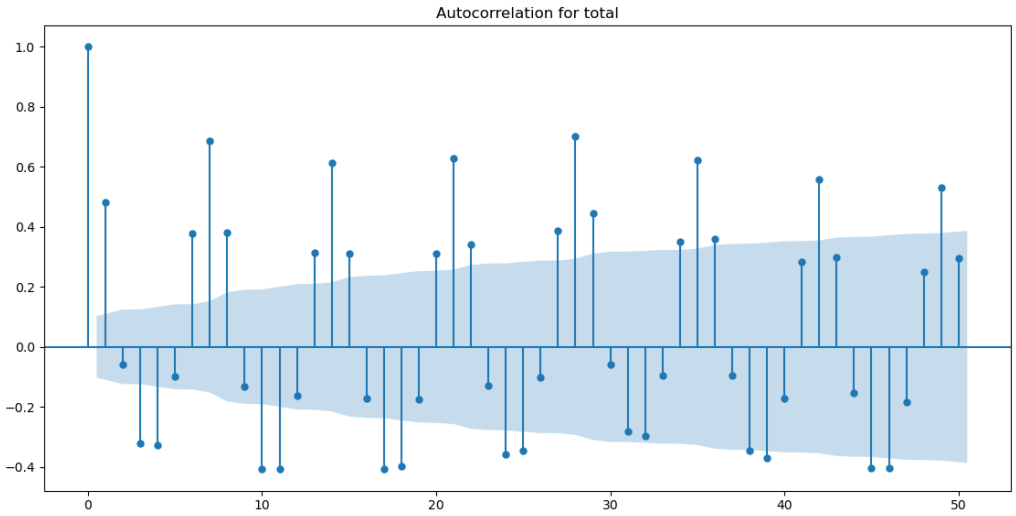

Gráfico 9.7. Autocorrelaciones para las ventas diarias para todas las tiendas.

Por último, la autocorrelación parcial nos confirma la estacionalidad semanal y como su influencia baja a medida que pasa el tiempo y pasa a ser no significativa (el desfase 0 siempre da 1 porque es contra sí misma sin desfase), como se observa en el gráfico 9.8.

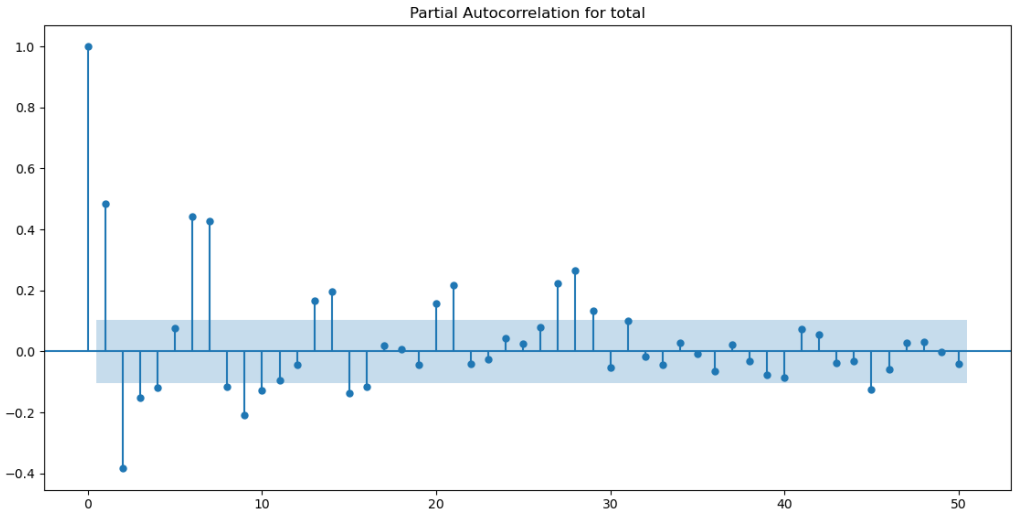

Gráfico 9.8. Autocorrelaciones parciales para las ventas diarias para todas las tiendas.

9.2.2.3 Análisis de las series de tiempo a nivel estatal

Se analizará el comportamiento de las series de tiempo de cada uno de los estados del estudio.

9.2.2.3.1 California

California es el estado con más ventas por lo que su serie de tiempo es muy similar al total de los 3 estados. De todas maneras, comienzan a ver diferencias en algunos patrones con respecto al total de ventas. Por ejemplo, en el gráfico 9.9, el Super Bowl genera un pico de ventas, mientras que en el gráfico 9.5 se observa que ese pico se da un día antes del evento.

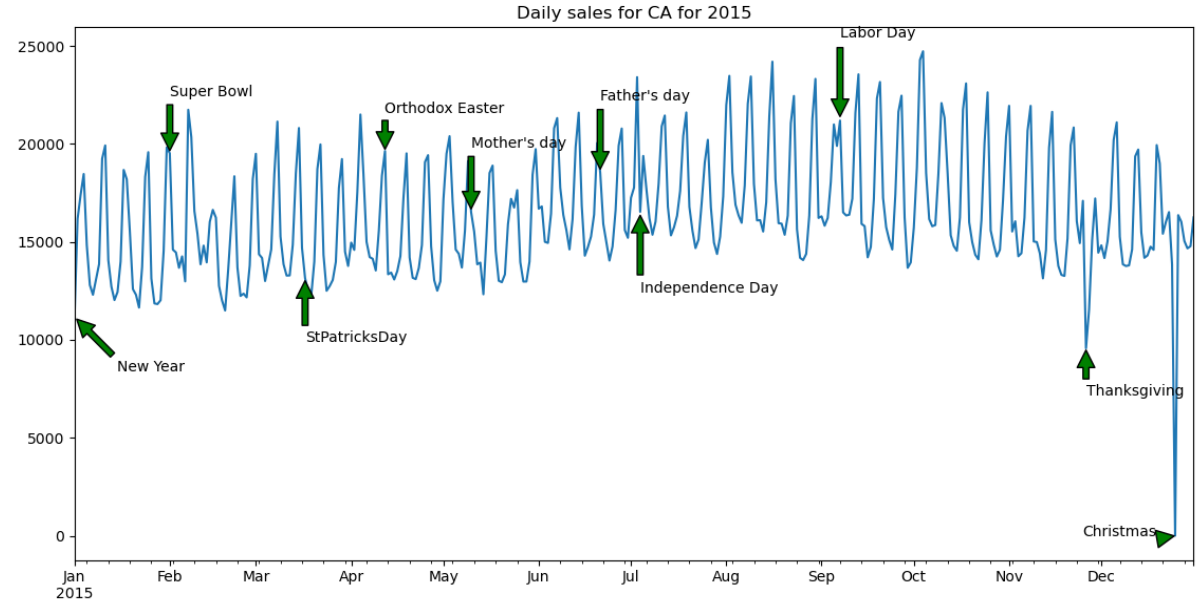

Gráfico 9.9. Ventas diarias para todas las tiendas del estado de California en el 2015.

La tendencia y la estacionalidad en el gráfico 9.10 son similares a los de las ventas totales en el gráfico 9.6. Esto es un indicio de que California es el estado que define la forma y comportamiento de las ventas totales en general.

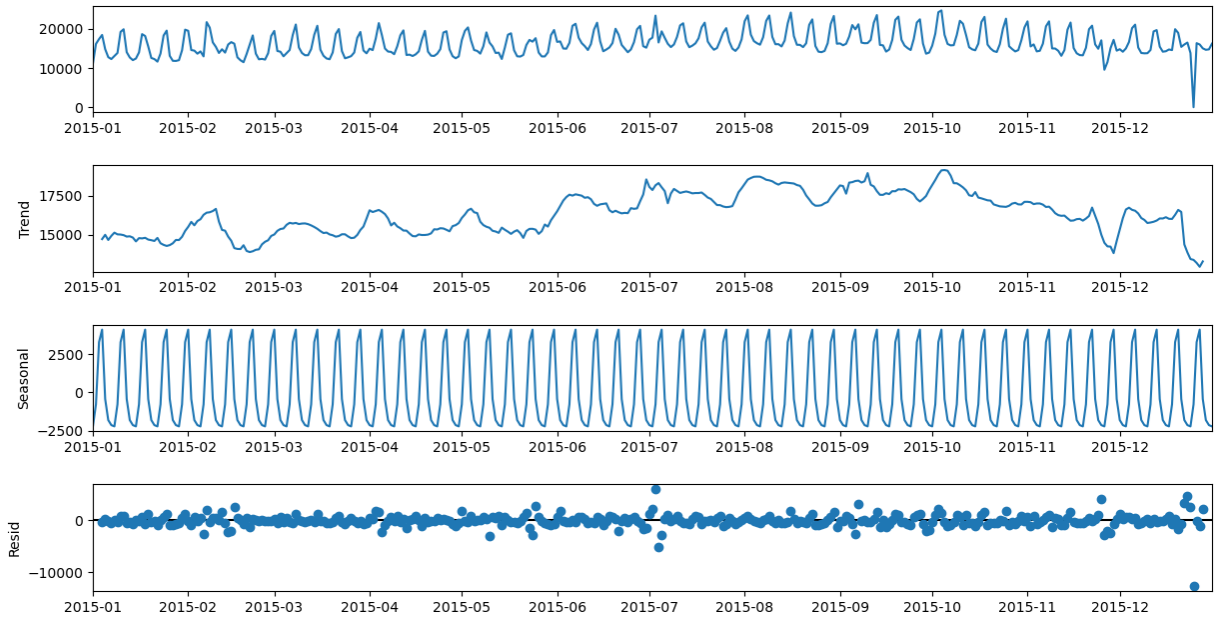

Gráfico 9.10. Ventas diarias para todas las tiendas del estado de California descompuesta en tendencia, estacionalidad y residuos.

En sintonía con lo dicho anteriormente, tanto las autocorrelaciones como las autocorrelaciones parciales son similares a las vistas para las ventas en los gráficos 9.7 y 9.8.

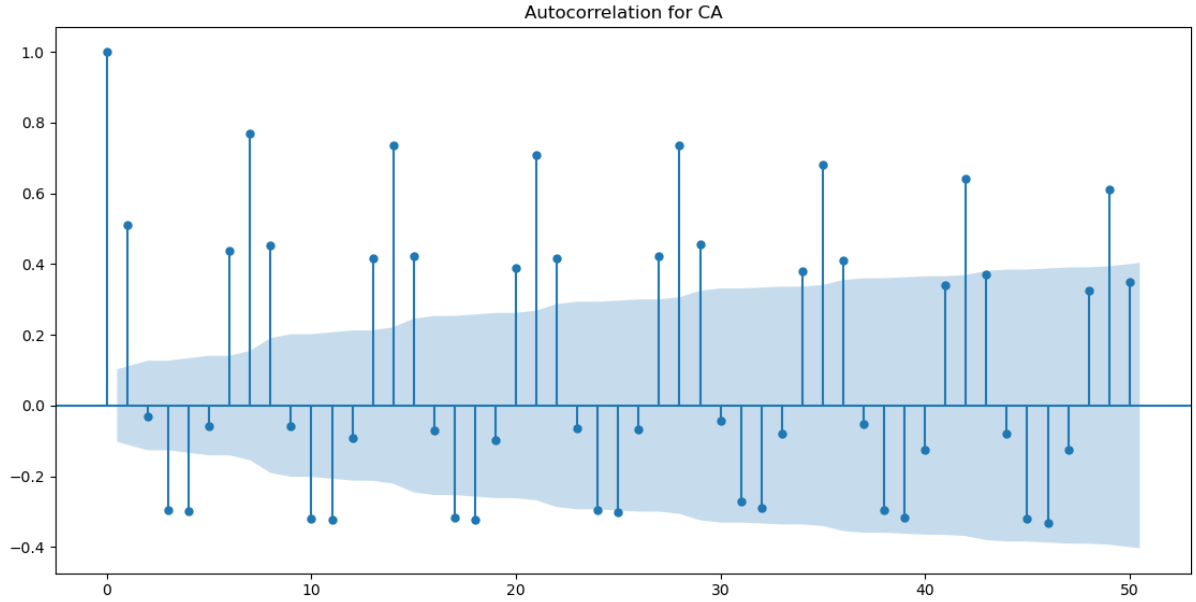

Gráfico 9.11. Autocorrelaciones para las ventas diarias para todas las tiendas del estado de California.

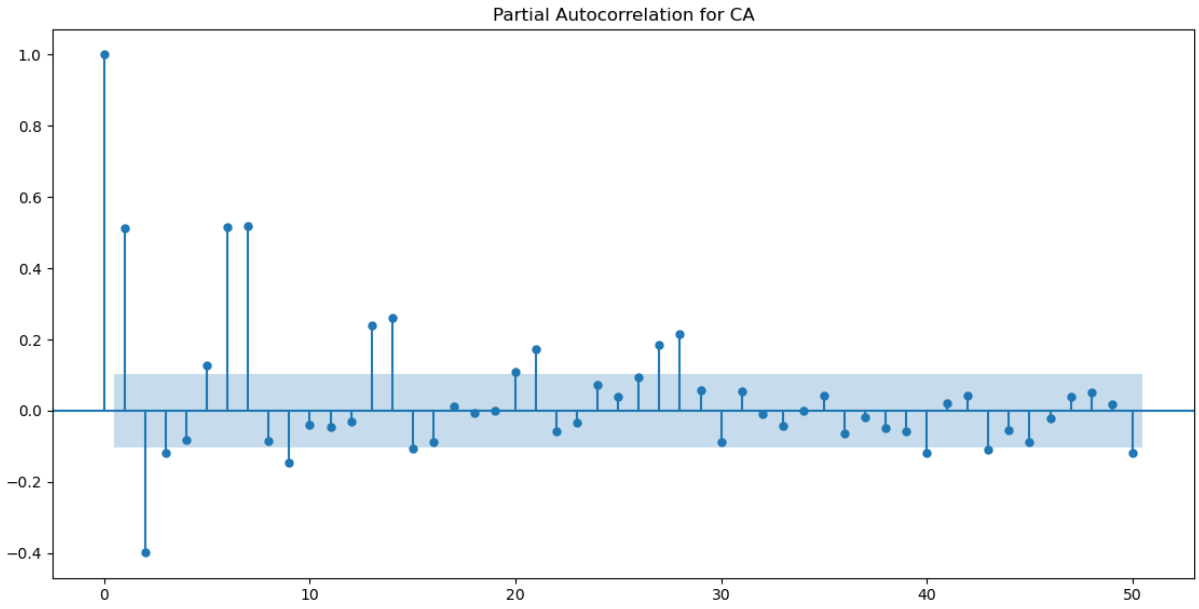

Gráfico 9.12. Autocorrelaciones parciales para las ventas diarias para todas las tiendas del estado de California.

9.2.2.3.2 Texas

Al considerar sólo Texas, estado que acumula el 29% de las ventas históricas, vemos algunas particularidades respecto a California.

Si observamos el gráfico 9.13, se aprecia que la caída de ventas en el Día de Acción de Gracias es similar a otros picos inferiores cercanos, lo que nos indica que este evento no es tan importante para predecir las ventas como lo es en California. Además aparece un pico superior un 50% más grande, a mediados de junio, que no aparece ni en el gráfico 9.5 ni en el gráfico 9.9.

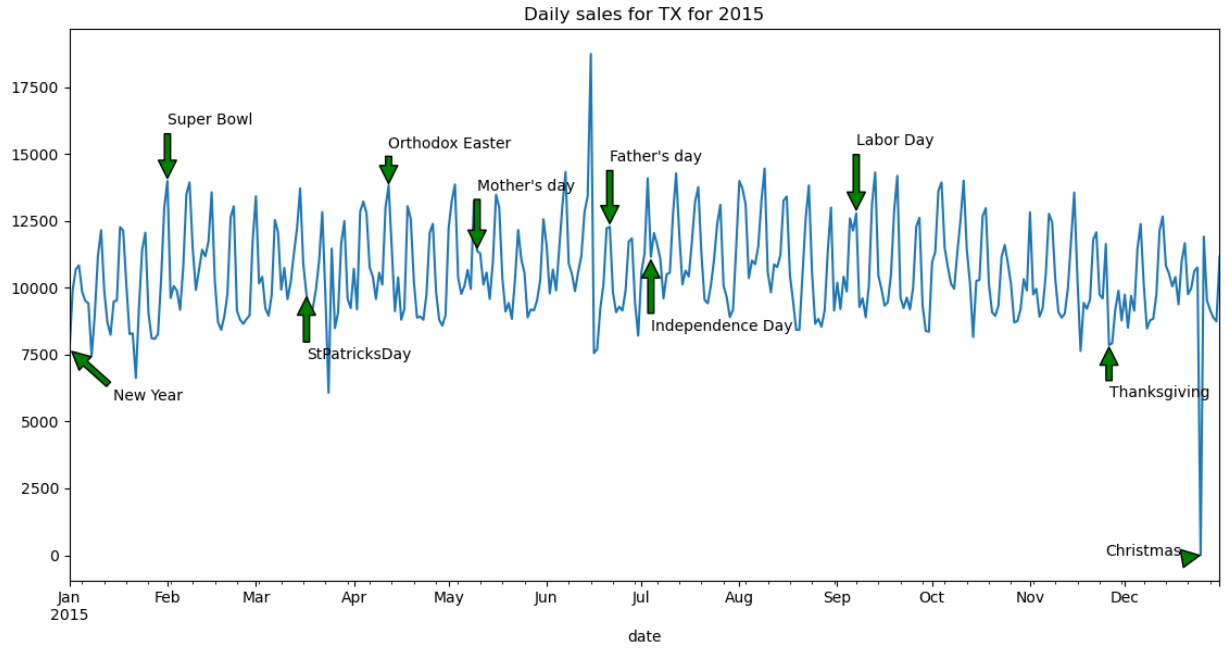

Gráfico 9.13. Ventas diarias para todas las tiendas del estado de Texas en el 2015.

No hay una tendencia marcada para el estado de Texas (parece seguir un proceso estocástico), como se aprecia en el gráfico 9.14. La estacionalidad es semanal al igual que los gráficos anteriores.

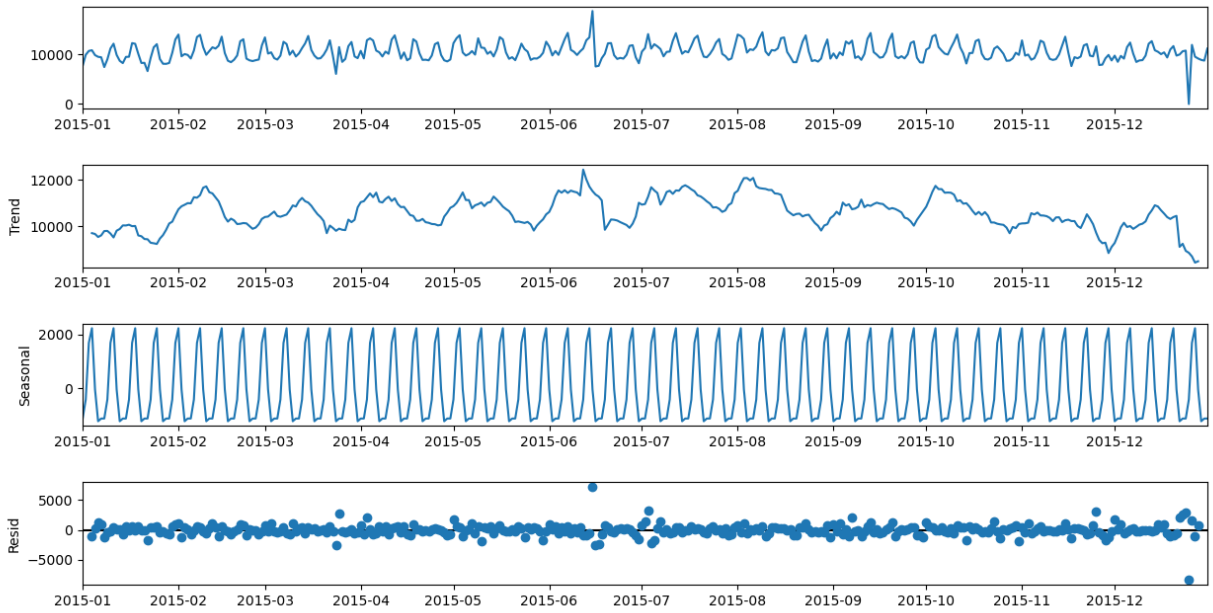

Gráfico 9.14. Ventas diarias para todas las tiendas del estado de Texas descompuesta en tendencia, estacionalidad y residuos.

Tanto el gráfico 9.15 como el 9.16 se observa la estacionalidad semanal de la serie de tiempo, en línea con California y una fuerte dependencia con el día anterior.

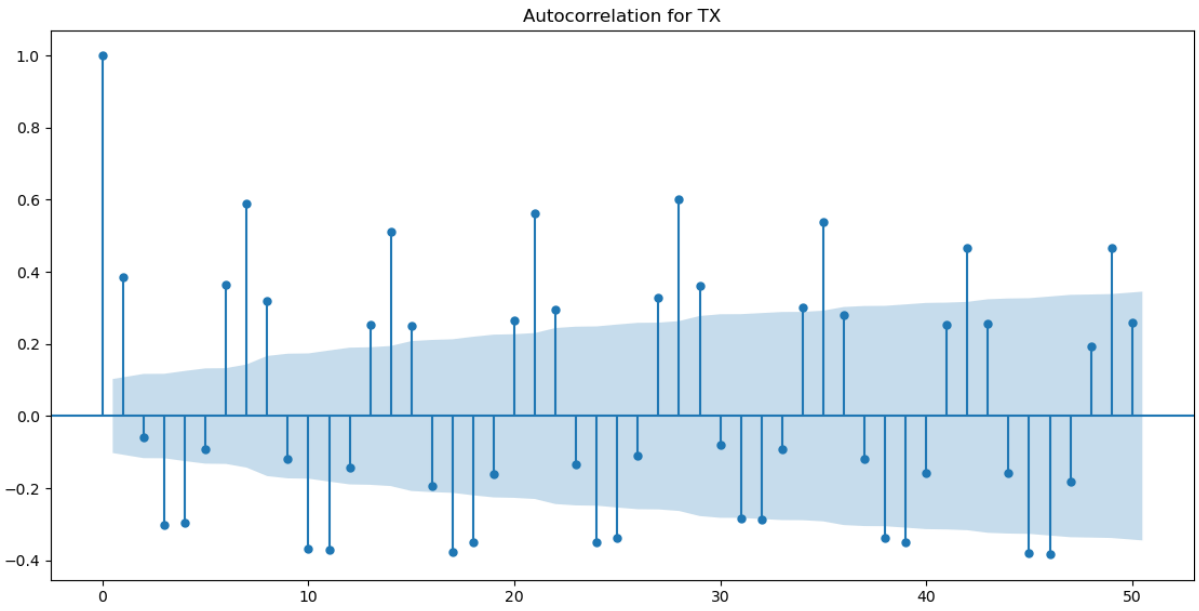

Gráfico 9.15. Autocorrelaciones para las ventas diarias para todas las tiendas del estado de Texas.

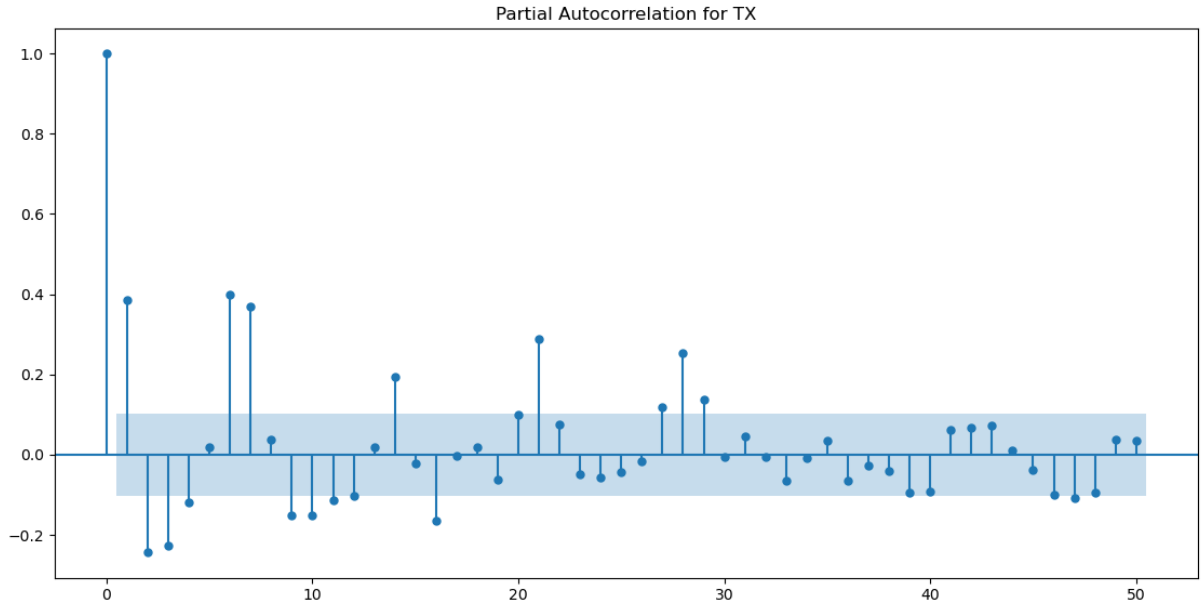

Gráfico 9.16. Autocorrelaciones parciales para las ventas diarias para todas las tiendas del estado de Texas.

9.2.2.3.3 Wisconsin

En este estado con menor población respecto a California y Texas, las ventas siguen un patrón diferente para algunos eventos de interés en EE.UU.

El gráfico 9.17 nos muestra estos cambios marcados en algunos eventos. Por ejemplo, el Super Bowl es uno de los picos más bajos del año (contrario a los otros en los que este día es un pico alto en ventas) pero el día anterior es el pico positivo máximo del año. Esto podría indicar que en el estado de Wisconsin, la población tiene un comportamiento previsor y prefiere prepararse el día anterior al Super Bowl en lugar de comprar el mismo día como en California o Texas.

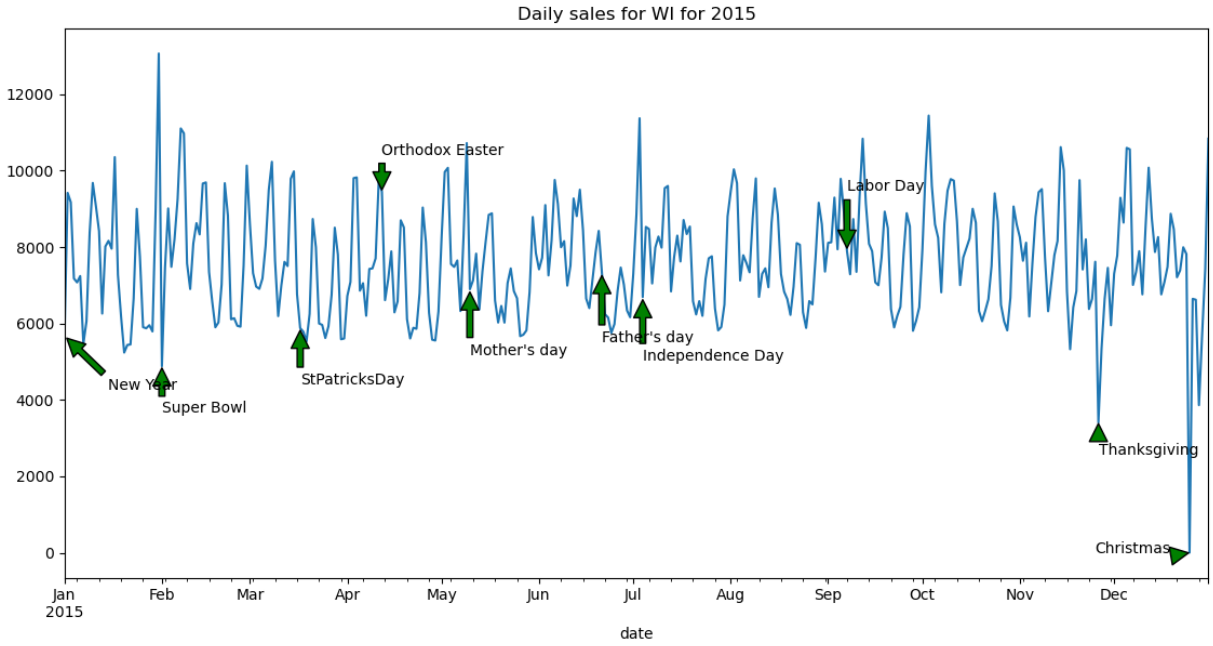

Gráfico 9.17. Ventas diarias para todas las tiendas del estado de Wisconsin en el 2015.

Al igual que en Texas, el gráfico 9.18 nos muestra que no hay una tendencia clara en el año y nos confirma la estacionalidad semanal al igual que las otras series de tiempo.

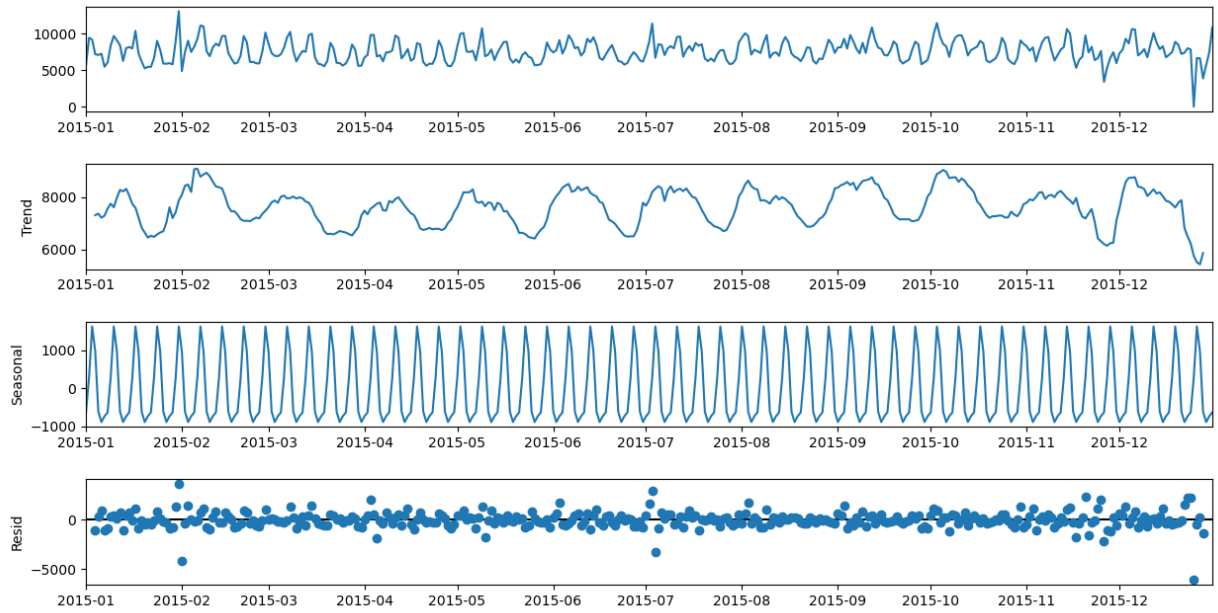

Gráfico 9.18. Ventas diarias para todas las tiendas del estado de Wisconsin descompuesta en tendencia, estacionalidad y residuos.

Al observar los gráficos 9.19 y 9.20, se concluye que Wisconsin tiene una estacional semanal pero no tiene una correlación tan marcada como en California y Texas. Además, este estado tiene una estacionalidad significativa a los 30 días mientras que las otros estados no.

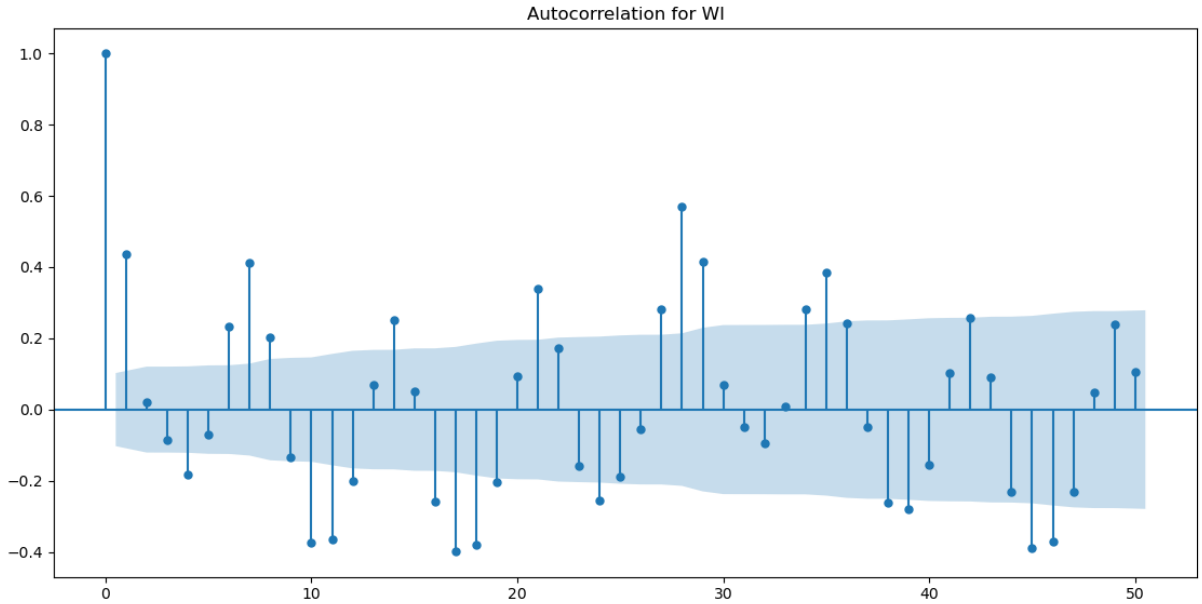

Gráfico 9.19. Autocorrelaciones para las ventas diarias para todas las tiendas del estado de Wisconsin.

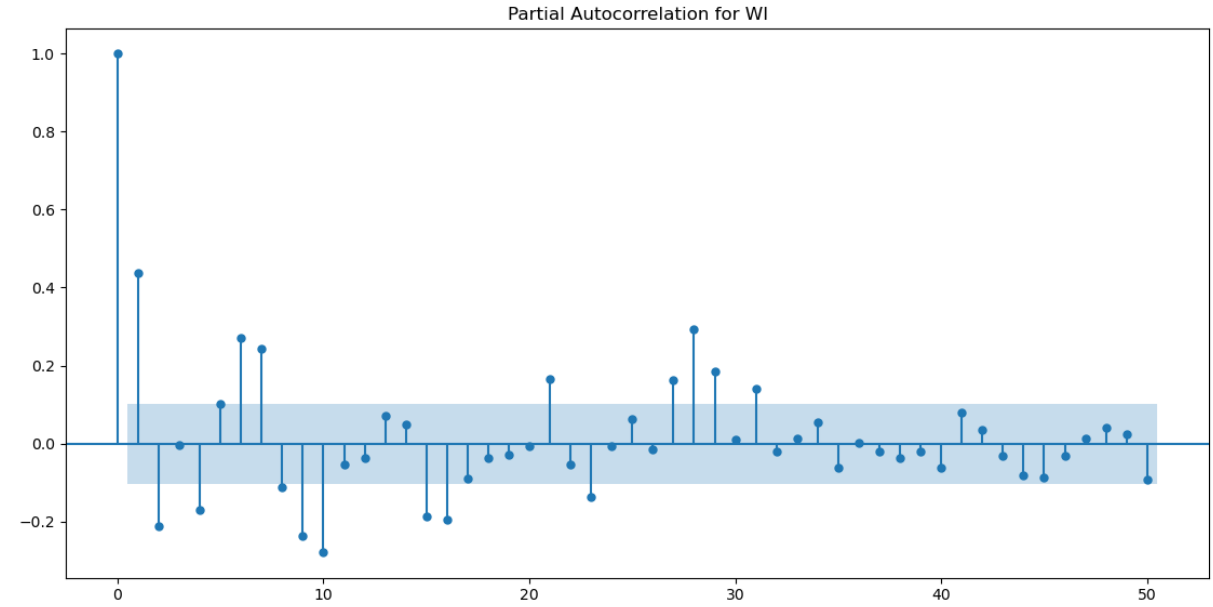

Gráfico 9.20. Autocorrelaciones parciales para las ventas diarias para todas las tiendas del estado de Wisconsin.

9.2.2.3.4 Comparación entre estados

Esto nos muestra que, a pesar de que en términos generales las series de tiempo de los 3 estados tienen un comportamiento similar, existen particularidades pertenecientes a cada estado que complejizan un modelamiento con técnicas clásicas como ARIMA. Un evento tan importante como el Super Bowl puede significar cosas muy distintas según el estado que observemos.

9.2.2.4 Análisis de las series de tiempo a nivel tienda-departamento

Luego de observar las relaciones generales a nivel estatal, se analizarán las series de tiempo a nivel tienda-departamento. Dado que son 70 series en total, se mostrará a continuación el detalle de una serie de cada categoría y de cada estado.

9.2.2.4.1 CA3FOO1 (Departamento FOOD 1 en tienda 3 de California)

El gráfico 9.21 nos muestra las ventas del departamento Food 1, para la tienda 3 en California. Estando en un nivel más bajo en la estructura jerárquica, se observa más volatilidad de las ventas. La varianza de la serie de tiempo no es constante en el tiempo (la distancia entre los picos mínimos y máximos cambia según el mes). Esto lo hace una serie más impredecible y complicada de modelar que las anteriores.

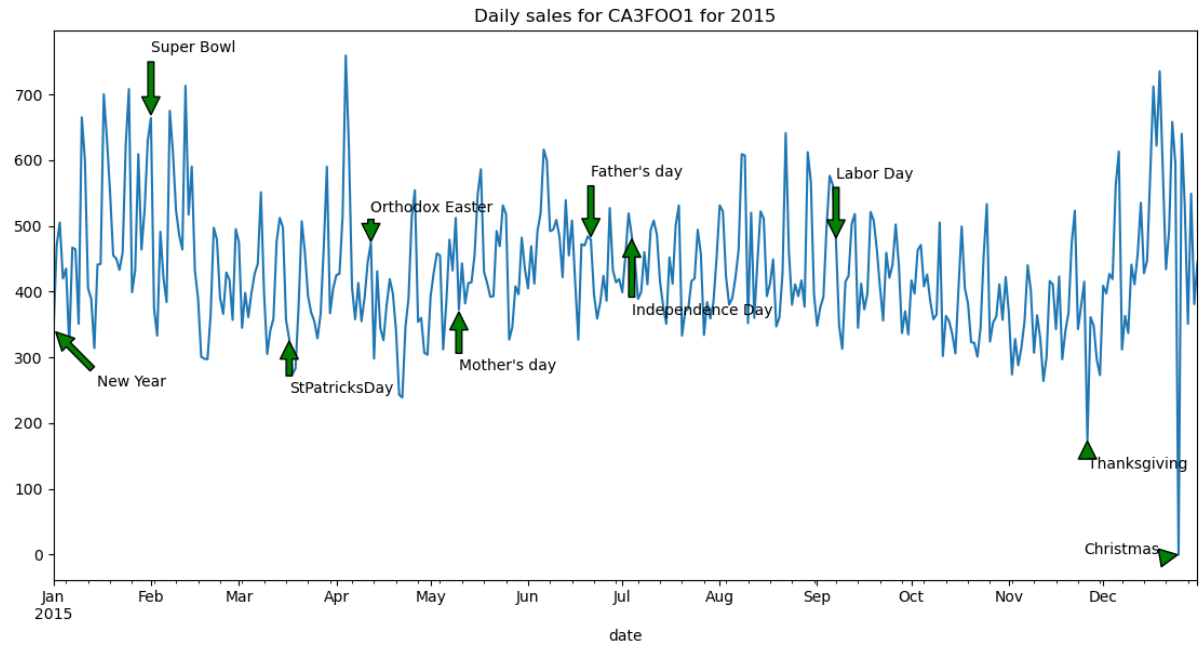

Gráfico 9.21. Ventas diarias del departamento FOOD 1 para la tienda 3 del estado de California en el 2015.

La tendencia es decreciente en el tiempo desde febrero hasta noviembre y luego vuelve a subir (diferente a las demás que eran crecientes o estocásticas). La estacionalidad es principalmente semanal como se observa en el gráfico 9.22.

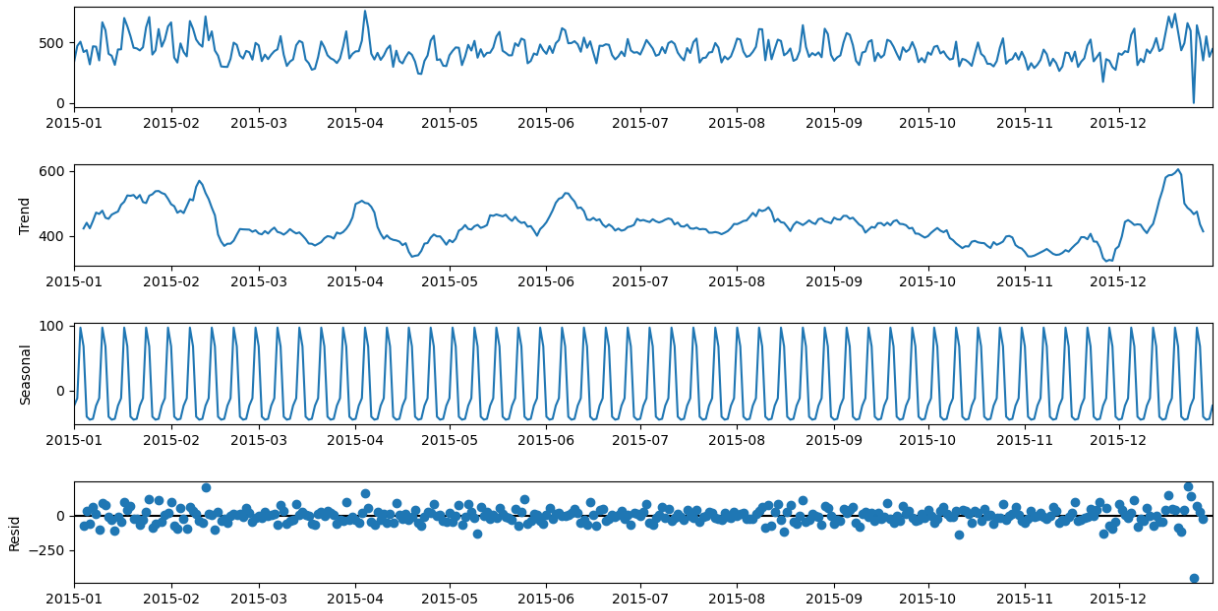

Gráfico 9.22. Ventas diarias del departamento FOOD 1 para la tienda 3 del estado de California descompuesta por tendencia, estacionalidad y residuos.

A pesar de pertenecer a California, las autocorrelaciones que vemos para este departamento en el gráfico 9.23 y 9.24 son distintas a las que encontramos para el estado en general. La estacionalidad semanal es significativa pero disminuye a medida que nos alejamos en desfases.

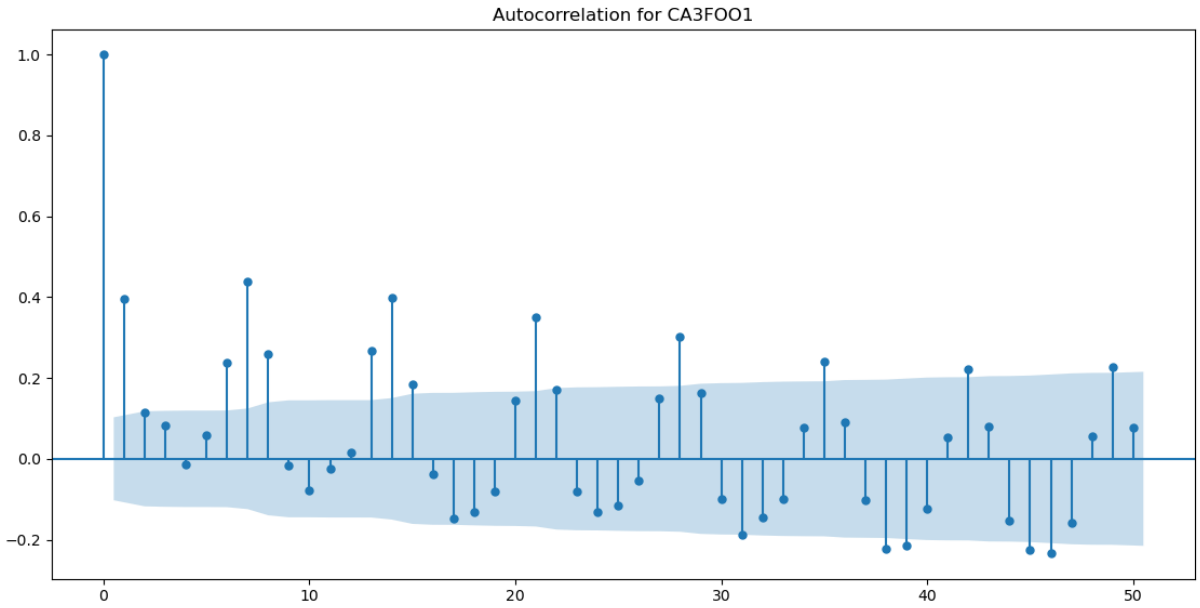

Gráfico 9.23. Autocorrelaciones para las ventas diarias del departamento FOOD 1 para la tienda 3 del estado de California.

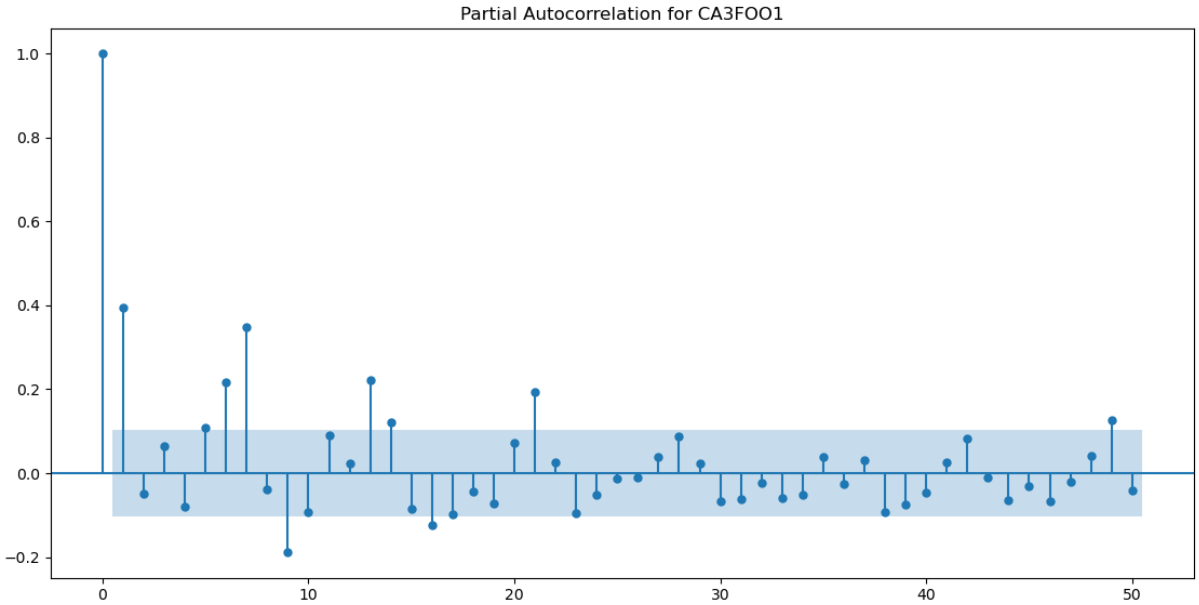

Gráfico 9.24. Autocorrelaciones parciales para las ventas diarias del departamento FOOD 1 para la tienda 3 del estado de California.

9.2.2.4.2 TX2HOB2 (Departamento HOBBIES 2 en tienda 2 de Texas)

En este caso se analizará el departamento Hobbies 2 en la tienda 2 de Texas. El pico más alto en febrero que se observa en el gráfico 9.25 se debe a San Valentín. Este departamento aparenta ser más influenciado por eventos que implican regalos, distintos a los de la categoría Food. La varianza no es constante en el tiempo para esta serie de tiempo.

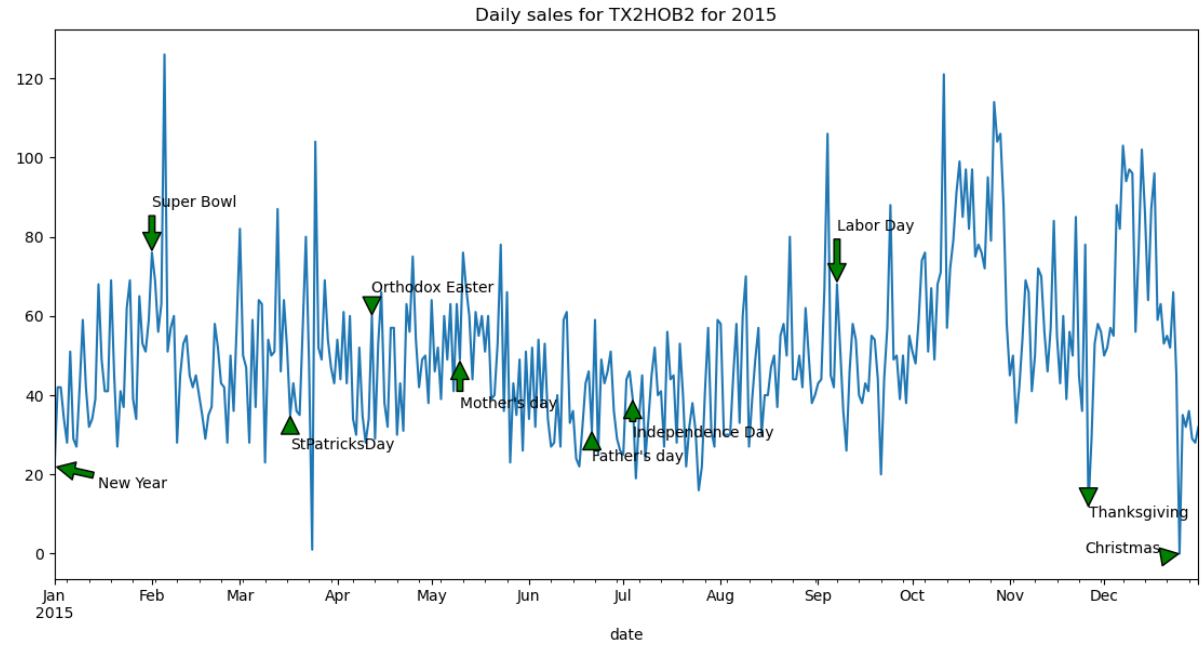

Gráfico 9.25. Ventas diarias del departamento Hobbies 2 para la tienda 2 del estado de Texas en el 2015.

El gráfico 9.26, nos muestra diferencias interesantes respecto a las otras series de tiempo estudiadas. La tendencia no es significativa a lo largo de todo el año. La estacionalidad no es tan lineal, existe un pico intermedio. Por último los residuos parecen estar más alejados de la media 0.

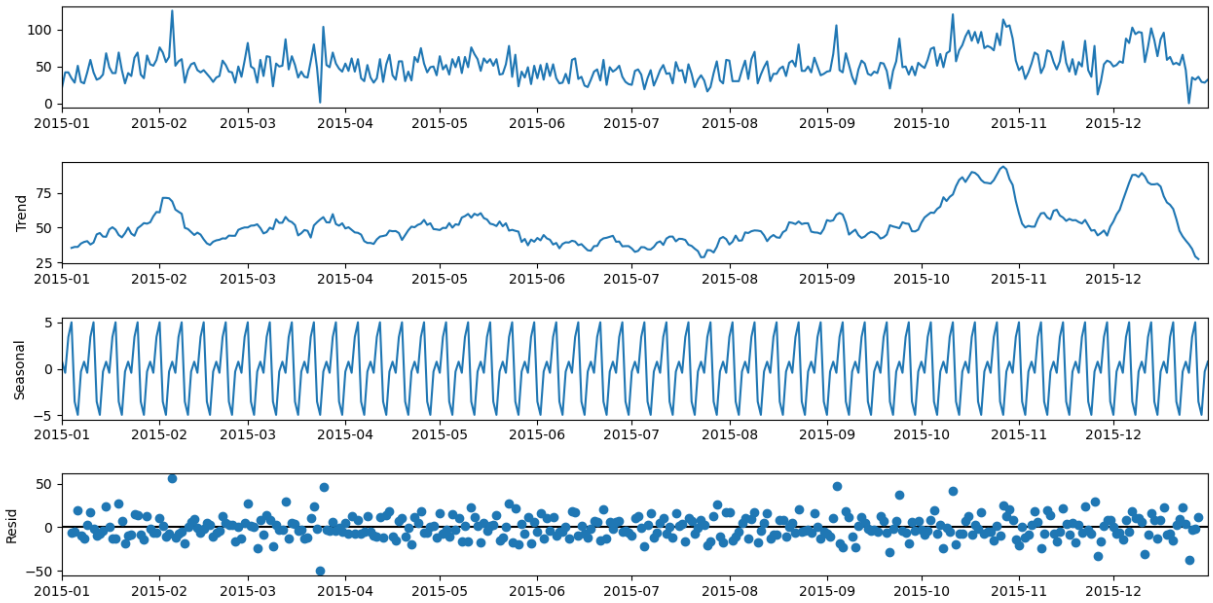

Gráfico 9.26. Ventas diarias del departamento Hobbies 2 para la tienda 2 del estado de Texas descompuesta por tendencia, estacionalidad y residuos.

En el gráfico 9.27 se observa que las autocorrelaciones son totalmente distintas a las otras series. Son sólo significativas hasta el lag 10. No aparenta tener estacionalidad semanal.

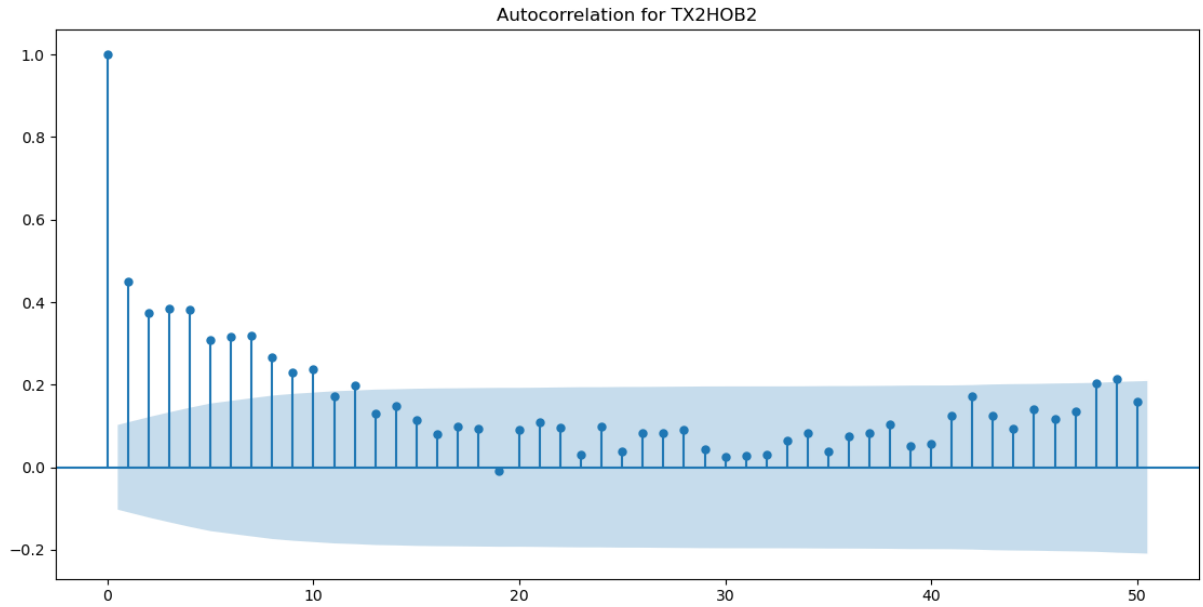

Gráfico 9.27. Autocorrelaciones para las ventas diarias del departamento Hobbies 2 para la tienda 2 del estado de Texas.

Al igual que en el gráfico anterior, vemos que en el gráfico 9.28 no hay una estacionalidad semanal marcada, y la autocorrelación parcial sólo es significativa en los primeros 4 desfases.

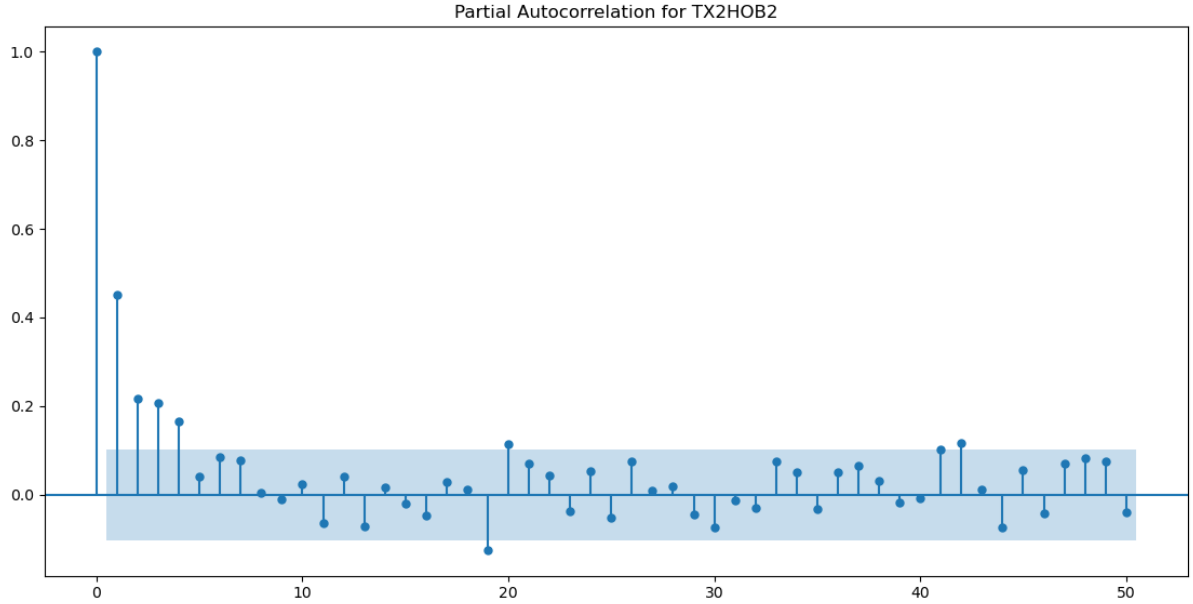

Gráfico 9.28. Autocorrelaciones parciales para las ventas diarias del departamento Hobbies 2 para la tienda 2 del estado de Texas.

9.2.2.4.3 WI1HOU2 (Departamento HOU2 en tienda 1 de Wisconsin)

En este caso se analizará el departamento Household 2 en la tienda 1 de Wisconsin.

El gráfico 9.29 muestra las siguientes particularidades:

- La varianza es mucho más alta en los primeros 4 meses que en el resto de la serie (la distancia entre picos es mayor en estos meses).
- El evento Super Bowl es comparable con el Día de Acción de Gracias.
- El Día del Padre es un pico positivo (contrario a lo que vimos para el departamento Hobbies 2).

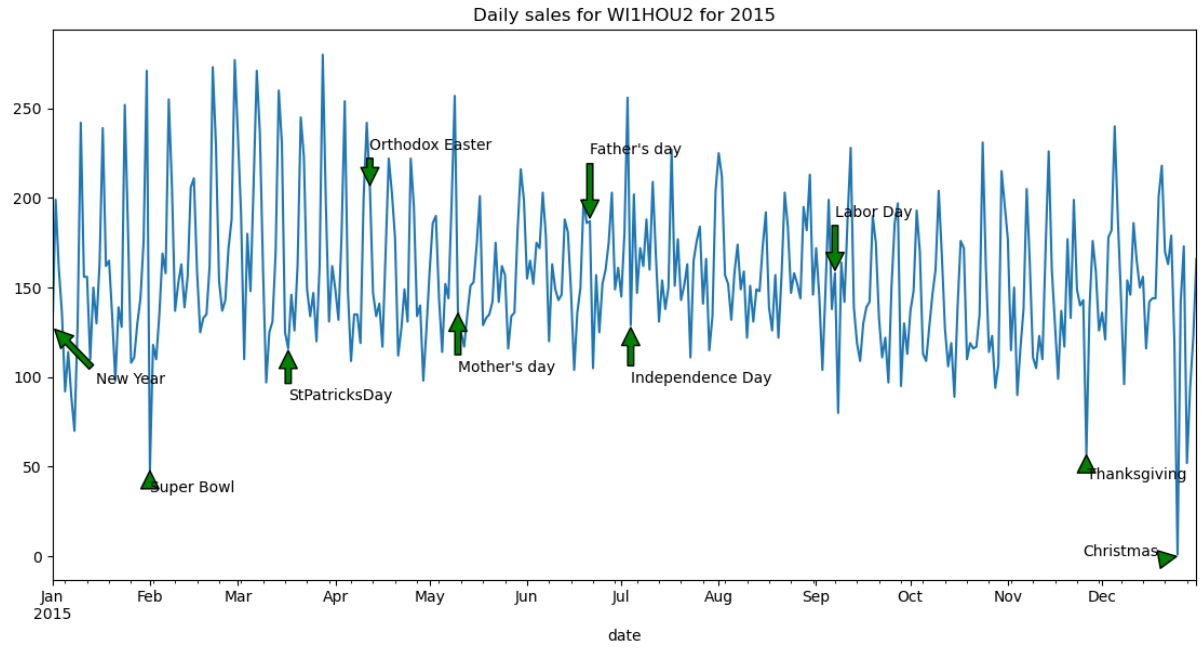

Gráfico 9.29. Ventas diarias del departamento Household 2 para la tienda 1 del estado de Wisconsin en el 2015.

La tendencia es decreciente para esta serie de tiempo, como se ve en el gráfico 9.30. Además la estacionalidad semanal es bastante marcada.

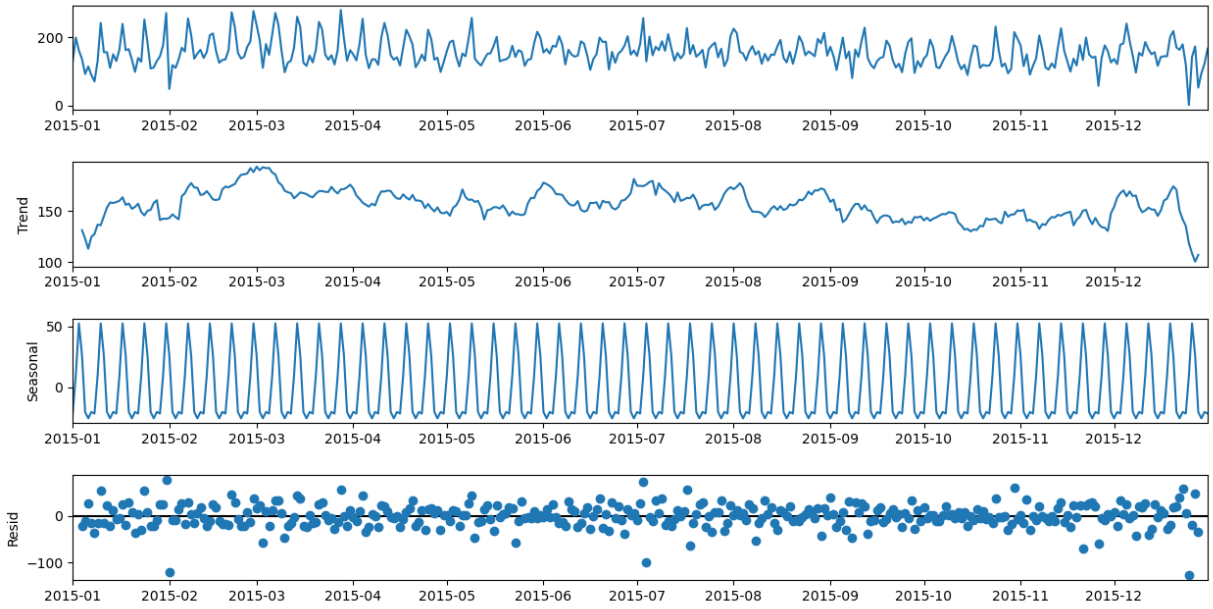

Gráfico 9.30. Ventas diarias del departamento Household 2 para la tienda 1 del estado de Wisconsin descompuesta por tendencia, estacionalidad y residuos.

Las autocorrelaciones y autocorrelaciones parciales en los gráficos 9.31 y 9.32 son casi idénticas a las pertenecientes a California. Hay una fuerte estacionalidad semanal y una dependencia con el primer desfase.

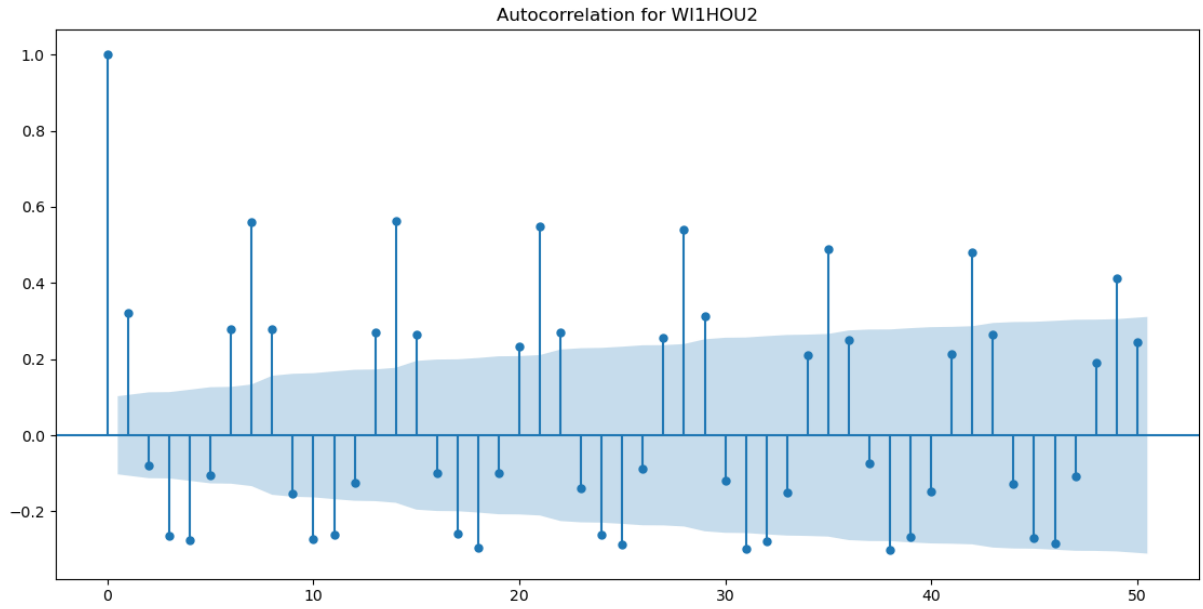

Gráfico 9.31. Autocorrelaciones para las ventas diarias del departamento Household 2 para la tienda 1 del estado de Wisconsin.

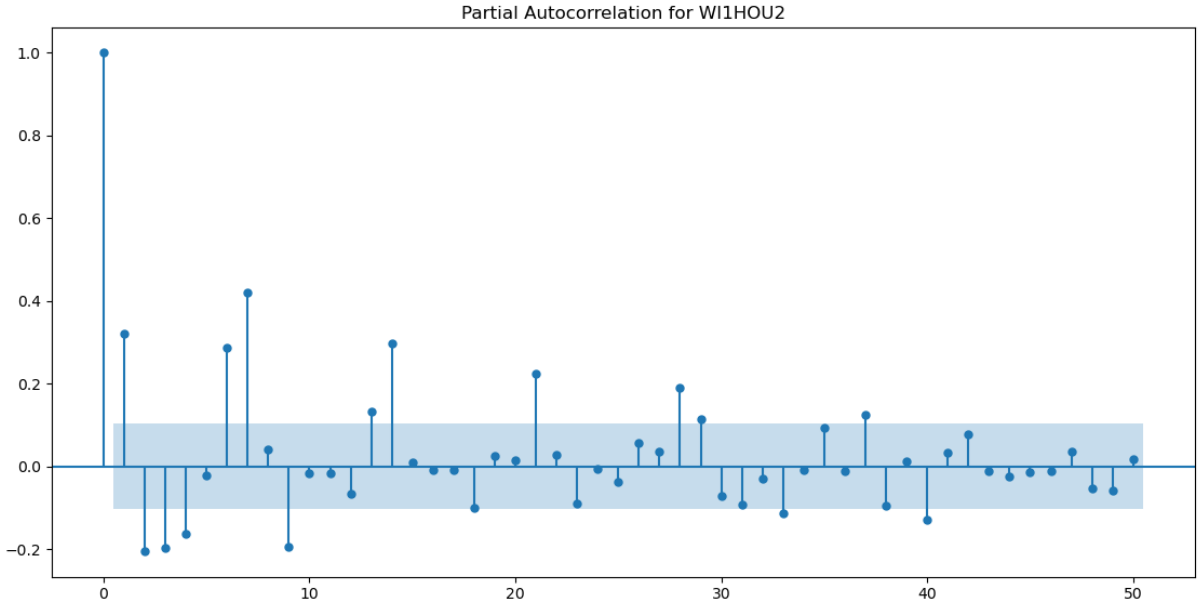

Gráfico 9.32. Autocorrelaciones parciales para las ventas diarias del departamento Household 2 para la tienda 1 del estado de Wisconsin.

9.2.2.4 Comparación entre tienda-departamentos

Cada serie de tiempo responde diferente a eventos importantes y los patrones estacionales son diferentes. Además, existe información valiosa al saber a qué categoría y a que estado pertenece el departamento, que puede ser usada para modelar la serie de tiempo. Por ejemplo, todos los departamentos de Food van a comportarse de forma similar frente a eventos como el Super Bowl, siempre y cuando sean de la misma tienda o estado. Esto nos indica que información de un nivel de la estructura jerárquica puede ser utilizada en un nivel inferior donde la relación no es tan clara o permanece oculta.

9.2.3 Resumen Análisis Exploratorio

- Las series de tiempo comparten similitudes como una marcada estacionalidad semanal, pero luego varían según el nivel que se encuentren en la jerarquía y a que tipo de categoría pertenezcan. Los niveles más altos son más estables, con una varianza constante lo que hace que sean más simples de modelar. Si miramos a nivel tienda-departamental, la varianza deja de ser constante y la tendencia es errática. Probablemente estas diferencias se deban a que los niveles más altos compensan los picos que suceden en los niveles más bajos, bajando la varianza.
- Los días en los que hay un evento importante tienen una influencia significativa pero distinta según el estado, tienda, categoría o departamento. El Super Bowl no afecta de igual manera a la categoría Food (pico positivo) que a Household (pico negativo). Dada la naturaleza de las categorías, probablemente la gente compre el televisor (household) el día anterior a este eventos, pero todo lo referido a comida y snacks unas horas antes del partido.
- La estructura jerárquica nos aporta información importante para la predicción de ventas. Si sabemos a qué categoría o a qué estado pertenece, tenemos información que pasaría inadvertida o al menos no fácilmente identificable si modelamos las series por separado y sin tener en cuenta las demás series.

Por todo esto, un enfoque clásico de series de tiempo sería insuficiente para poder modelar correctamente este tipo de series.

Las Redes Neuronales LSTM tienen la capacidad teórica de aprender la estructura jerárquica, dado que se entrenan todos las series al mismo tiempo, y de aprender las relaciones de cada departamento con sus desfases.

De igual manera, la técnica de series de tiempo jerárquicas (HTS) toma las bondades de los métodos clásicos como ARIMA, para capturar estos desfases y le suman una capa adicional para reconciliar las series según la estructura jerárquica en la que se encuentran.

En resumen, tanto LSTM como HTS son buenos candidatos para modelar este tipo de problemática donde la estructura en la que se encuentran las series de tiempo ocupa un rol importante.

9.3 Modelado

Debemos garantizar la independencia de los resultados sobre la partición de los datos. Como estamos utilizando series de tiempo, debe respetarse el orden en el que ocurren los eventos (no puede haber eventos futuros dentro del dataset Train). Por ello, se toman distintos subconjuntos del dataset y se predicen los 30 días posteriores. De esta manera, garantizamos respetar la dependencia temporal y validamos el modelo para distintos momentos.

Este caso particular se dividió el dataset en 3 subconjuntos y se promedia el error:

- Entrenamiento: Desde 29/01/2011 hasta 20/05/2016, Validación: 21/05/2016 hasta 19/06/2016.
- Entrenamiento: Desde 29/01/2011 hasta 20/04/2016, Validación: 21/04/2016 hasta 20/05/2016.
- Entrenamiento: Desde 29/01/2011 hasta 21/03/2016, Validación: 22/03/2016 hasta 21/04/2016.

Este error se calculará para las 70 series de tiempo que representan al nivel tienda-departamento (dado que tenemos 7 departamentos para cada tienda).

Se analizará si existe un modelo general que minimice el error de todas estas series de tiempo, o si es necesario tener distintos modelos por departamento. Para ello, se calculará la mediana de los errores como métrica para comparar qué tan bueno es un modelo para generalizar respecto a otros. El modelo que minimice esta métrica, tiene la capacidad de predecir con mayor certeza el conjunto total de series.

Además, esta forma de minimizar el error de las series a nivel tienda-departamental se analizará agrupando por estado o categoría para entender si hay diferencias entre estos niveles (siempre viéndolo a nivel tienda-departamental).

9.3.1 Series de tiempo Jerárquicas (HTS)

El paquete de R [21] llamado HTS desarrollado por Hyndman, R. J., Athanasopoulos, G., & Shang, H. L. (2013) [19], nos da la posibilidad de utilizar las técnicas de series de tiempo jerárquicas con los siguientes hiper parámetros:

1. **fmethod** = es el algoritmo que se utiliza para hacer el forecast base, al que luego se aplicará la reconciliación según la jerarquía. Este paquete nos da la opción de utilizar ARIMA o ETS.
2. **method** = Es la forma en la que se van a armonizar los niveles de la jerarquía. Como en este caso también existen relaciones geográficas, sólo pueden usarse los métodos "comb" (reconciliación óptima) y "bu" (bottom up).
Bottom up es el método clásico en el que los niveles superiores se calculan como la suma del nivel inferior, no se toma en cuenta ninguna reconciliación dada la estructura jerárquica del dataset.

La reconciliación óptima busca encontrar la matriz G óptima en la ecuación $\overline{y}_h = S\,G\,\widehat{y}_h$ que minimiza el error de predicción del conjunto de predicciones coherentes

donde $\widehat{y}_h$ = forecast base

$\overline{y}_h$ = forecast coherente o reconciliado

y la matriz S representa cómo deben sumarse los datos según la estructura jerárquica.

3. **weights** = Este hiper parámetro se utiliza cuando el método es la reconciliación óptima. Son distintas maneras de aproximar la matriz G dado que es complicada de calcular de forma exacta. Los valores posibles son:

ols = Se asume que la matriz G es independiente de la data

wls = Calcula G como proporcional a la varianza de los residuos del forecast base.

mint = Se asume que las matrices de error de covarianza son proporcionales entre sí, y se estima directamente la matriz completa de 1 paso de covarianza W_1

nseries = Se asume que cada error de la predicción base del último nivel tiene una varianza constante y no están correlacionadas entre nodos. Este estimador sólo depende de la estructura de la agregación, y no en la data. Por ello, es referido como escalamiento estructural.

4. **algorithms** = Este hiper parámetro es para saber que algoritmo se va a utilizar para realizar los cálculos cuando el método es reconciliación óptima. No debería haber diferencias usando uno u otro. Valores posibles = lu, cg, chol, recursive, slm.
5. **covariance** = Sólo se utiliza cuando weights = mint. Es la forma de calcular la matriz de covarianza W_1. Los valores posibles son:

sam = Usa la covarianza muestral.

shr = Se utiliza un estimador de "contracción" que contrae la covarianza muestral a una matriz diagonal.

Además de las ventas diarias de cada tienda-departamento, se agregaron otras features que resultaron relevantes en el análisis exploratorio:

- Día de la semana.
- Mes
- Evento importante que ocurre ese día.
- Evento importante que ocurre el día anterior.
- Tipo de evento importante.

Se probaron todas las combinaciones posibles de estos hiper parámetros.

9.3.1.1 Modelos que minimizan la mediana del RMSE

En este caso, se observa en la tabla 9.1, las 10 combinaciones de hiper parámetros que minimizan la mediana del RMSE entre todos las series de tiempo tienda-departamentales:

fmethod	method	weight	algorithm	covariance	extra features	Median RMSE
arima	comb	mint	cg	shr	TRUE	72.5920331
arima	comb	mint	chol	shr	TRUE	72.5920334
arima	comb	mint	lu	shr	TRUE	72.5920334
arima	comb	mint	cg	sam	TRUE	73.8473443
arima	comb	mint	chol	sam	TRUE	73.8473662
arima	comb	mint	lu	sam	TRUE	73.8473662
arima	comb	mint	cg	shr	FALSE	75.1083619
arima	comb	mint	chol	shr	FALSE	75.1083660
arima	comb	mint	lu	shr	FALSE	75.1083660
arima	comb	mint	cg	sam	FALSE	75.5076054

Tabla 9.1. Set de hiperparametros con menor mediana de RMSE teniendo en cuenta todas las series de tiempo a nivel tienda-departamento.

Se resume a continuación en el gráfico 9.33 cómo se desempeña el modelo que minimiza la mediana de todos los errores de las series de tiempo tienda-departamentales (el primer modelo en la tabla 9.1), a través de los distintos estados y tiendas.

Se observa que no hay gran diferencia en las distribuciones del error para los distintos estados. Esto nos indica que el modelo tiene capacidad para generalizar ya que se distribuye uniformemente entre estados. Si hubiera un estado con una distribución del error diferente, nos daría un indicio de que debería calcularse un modelo para cada estado por separado.

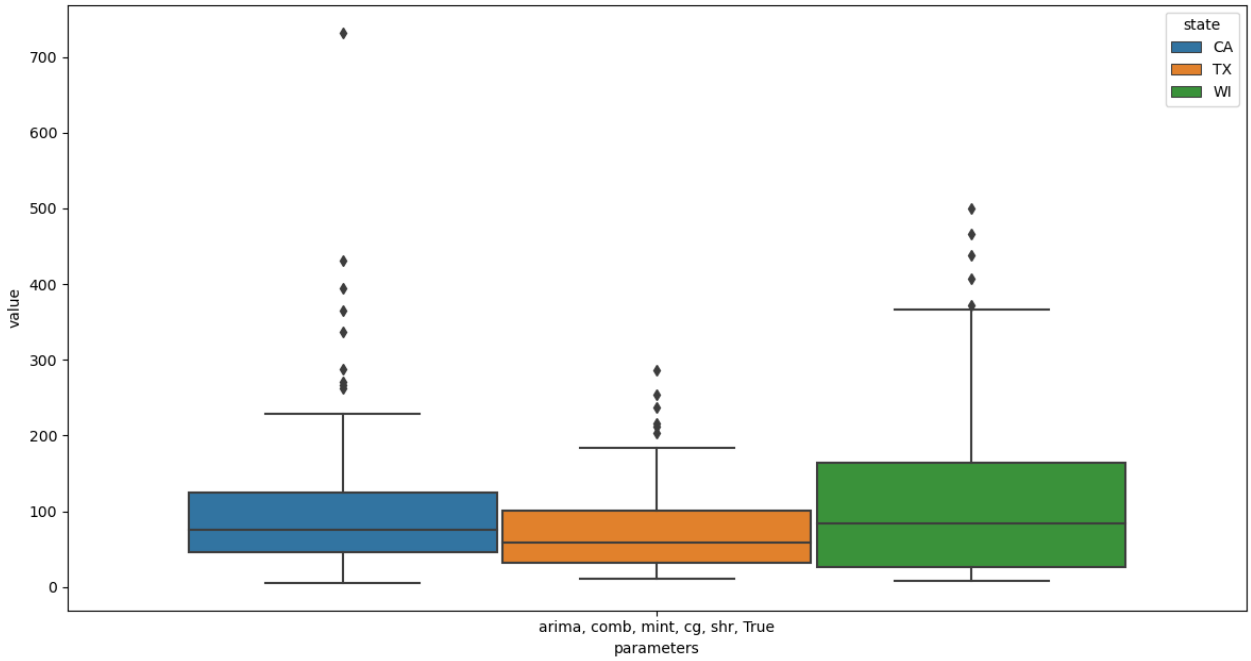

Gráfico 9.33. Distribución de los errores para cada uno de los 3 estados utilizando el modelo que minimiza la mediana de todos los errores.

Si observamos el gráfico 9.34, la distribución de los errores se mantiene en valores parecidos a nivel tienda, salvo por la tienda WI2 donde esta distribución tiene mayor variabilidad que las demás. Probablemente deberíamos calcular un modelo con otro set de hiper parámetros para esta tienda para disminuir ese error.

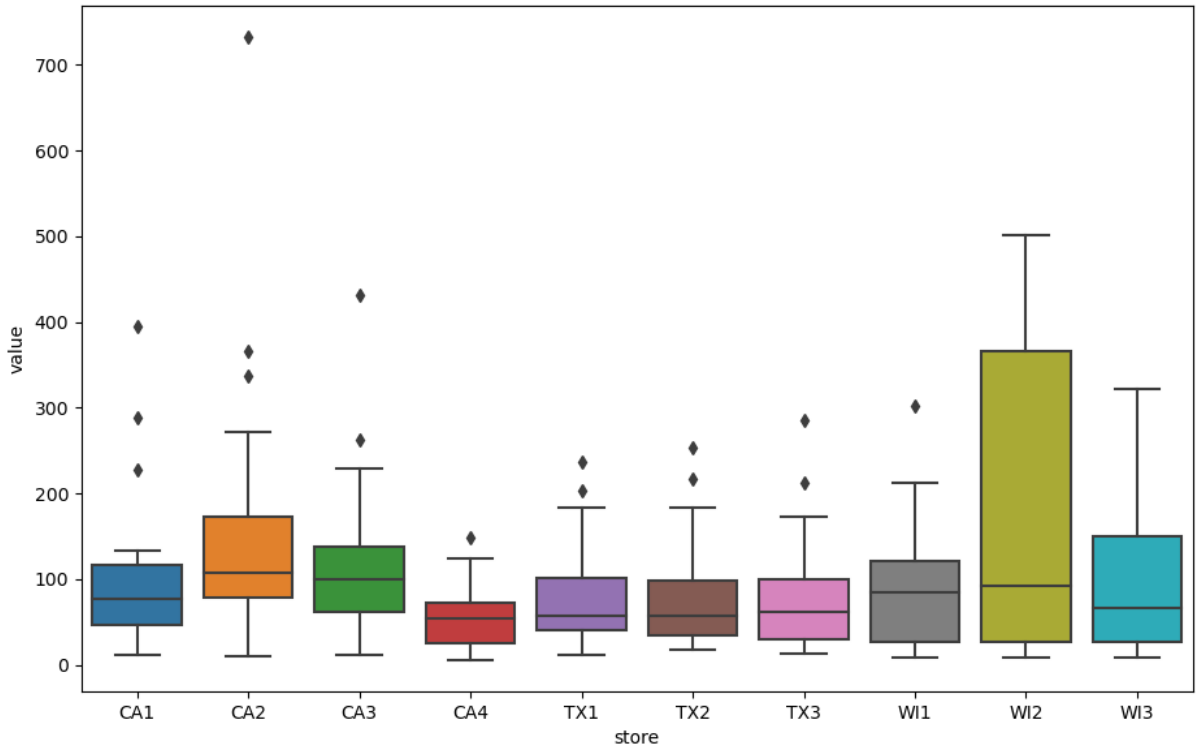

Gráfico 9.34. Distribución de los errores para cada uno de las 10 tiendas utilizando el modelo que minimiza la mediana de todos los errores.

9.3.1.1 Modelos que minimizan la mediana del RMSE agrupando a nivel estatal

Esta misma idea de minimizar la mediana del error de las series de tiempo tienda-departamentales se puede aplicar agrupando a nivel estatal o tienda para ver si existen diferencias entre los modelos óptimos para cada uno.
Como se observa en la tabla 9.2, Wisconsin y California tienen los mismos hiper parámetros óptimos, mientras que Texas utiliza otra forma de calcular la matriz de covarianza.

fmethod	method	weight	algorithm	covariance	extra features	Median RMSE	Position	state
arima	comb	mint	lu	shr	TRUE	75.9790263	1	CA
arima	comb	mint	chol	shr	TRUE	75.9790263	2	CA
arima	comb	mint	cg	shr	TRUE	75.9790273	3	CA
arima	comb	mint	cg	sam	TRUE	77.0294337	4	CA
arima	comb	mint	lu	sam	TRUE	77.0294446	5	CA
arima	comb	mint	lu	sam	TRUE	57.3018220	1	TX
arima	comb	mint	chol	sam	TRUE	57.3018220	2	TX
arima	comb	mint	cg	sam	TRUE	57.3018227	3	TX
arima	comb	mint	chol	sam	FALSE	57.8326003	4	TX
arima	comb	mint	lu	sam	FALSE	57.8326003	5	TX
arima	comb	mint	cg	shr	TRUE	84.1950522	1	WI
arima	comb	mint	lu	shr	TRUE	84.1950564	2	WI
arima	comb	mint	cg	sam	FALSE	84.8854623	3	WI
arima	comb	mint	chol	sam	FALSE	84.8855204	4	WI
arima	comb	mint	lu	sam	FALSE	84.8855204	5	WI

Tabla 9.2. Set de hiperparametros con menor mediana de RMSE teniendo en cuenta las series de tiempo a nivel tienda-departamento separado por estado.

9.3.1.3 Elección final de modelos

Dado que existen diferencias a nivel tienda entre las distribuciones del error, y no se obtienen los mismos hiper parámetros para cada estado, es conveniente calcular el set de hiper parámetros que minimice el error de cada serie de tiempo por separado.

9.3.2 Redes neuronales LSTM

La implementación de esta red neuronal se realizó utilizando la librería Keras [18] en Python.

En este caso los hiper parámetros a optimizar, utilizando grid search, fueron los siguientes:

Timesteps = Cantidad de días para atrás que LSTM toma en cuenta para cada punto. Se probaron 1, 12, 28 y 60 días.
Nro layers used = Número de capas LSTM en serie. Hasta 3 capas.
Layer units = Número de celdas LSTM en paralelo para cada capa. No necesariamente igual para cada capa. Se probaron 3 diferentes configuraciones:
(600, 300, 150), (300, 200, 100) y (200, 100, 80).
Nro de epochs = 5, 10, 20, 30, 50.
Batch size = 16, 32, 64, 128.
Is scaled = Booleano que define si el input fue previamente normalizado o no.
Extra features = Booleano que agrega otras features.

Al igual que en la técnica anterior, además de las ventas diarias de cada tienda-departamento, las features extra que se agregaron son:

- Día de la semana.
- Mes
- Evento importante que ocurre ese día.
- Evento importante que ocurre el día anterior.
- Tipo de evento importante.

9.3.2.1 Modelos que minimizan la mediana del RMSE

Utilizando la misma idea que anteriormente, vemos en la tabla 9.3, los modelos que minimizan la mediana de RMSE de todas las series de tiempo a nivel tienda-departamento.

Timesteps	Is Scaled	Layers Used	Layer Units	Nro Epochs	Batch Size	Extra Features	Median RMSE
1	TRUE	2	(200, 100, 80)	50	16	FALSE	87.8491428
1	TRUE	3	(600, 300, 150)	30	16	FALSE	89.2840153
1	TRUE	1	(500, 300, 150)	50	16	FALSE	94.9818183
1	TRUE	3	(300, 200, 100)	50	32	FALSE	96.3760606
1	TRUE	1	(200, 100, 80)	50	16	FALSE	96.7444484
1	TRUE	2	(600, 300, 150)	50	32	FALSE	98.9402639
1	TRUE	3	(600, 300, 150)	50	16	FALSE	101.8318382
14	TRUE	1	(300, 200, 100)	50	16	FALSE	103.5065730
28	TRUE	1	(300, 200, 100)	50	32	FALSE	103.5647479
1	TRUE	1	(300, 200, 100)	50	32	FALSE	104.0316372

Tabla 9.3. Set de hiperparametros con menor mediana de RMSE teniendo en cuenta todas las series de tiempo a nivel tienda-departamento.

Siguiendo la línea de pensamiento anterior, se observa en el gráfico 9.35 como se desempeña este modelo agrupando a las series de tiempo por los distintos estados y tiendas, de modo tal de ver si es conveniente utilizar este modelo o hay que utilizar uno distinto por serie de tiempo. Se observa que California y Texas tienen distribuciones del error similares pero la primera posee muchos outliers. Esto es un indicador de que lo más conveniente es minimizar el error para cada departamento por separado.

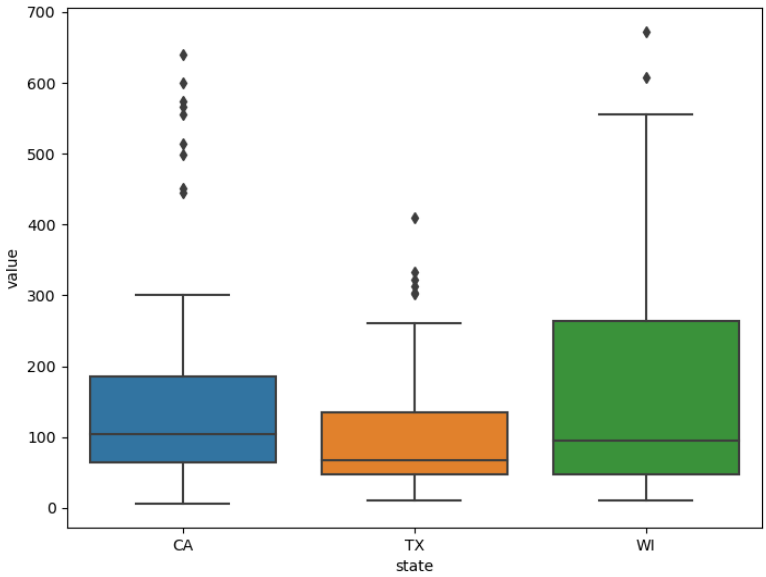

Gráfico 9.35. Distribución de los errores para cada uno de los 3 estados utilizando el modelo que minimiza la mediana de todos los errores

Si observamos el gráfico 9.36, vemos un patrón similar que en 9.34, ya que WI2 tiene una distribución de los errores con mayor variabilidad que los demás.

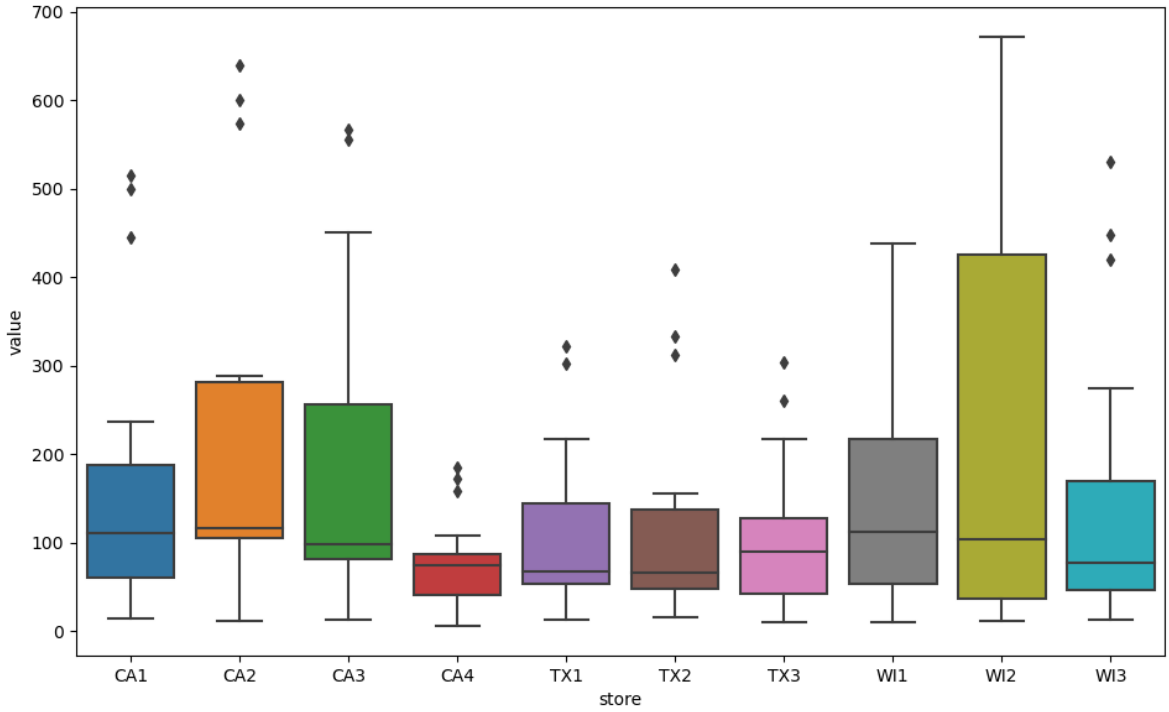

Gráfico 9.36. Distribución de los errores para cada uno de las 10 tiendas utilizando el modelo que minimiza la mediana de todos los errores.

9.3.2.2 Modelos que minimizan la mediana del RMSE agrupando a nivel estatal

Si aplicamos esta idea de minimizar la mediana del error de las series de tiempo tienda-departamentales pero agrupando por estado, podemos observar si existen diferencias entre los modelos óptimos para cada uno.

Si observamos la tabla 9.4, vemos que en los 3 estados los hiper parámetros óptimos son exactamente los mismos. Igualmente vimos anteriormente que este modelo generaba muchos outliers para cada estado.

Timesteps	Is Scaled	Layers Used	Layer Units	Nro Epochs	Batch Size	Extra Features	Median RMSE	Position	State
1	TRUE	2	(200,100,80)	50	16	FALSE	104.8229430	1	CA
1	TRUE	3	(600,300,150)	30	16	FALSE	105.5190495	2	CA
1	TRUE	1	(500,300,150)	50	16	FALSE	106.8341969	3	CA
14	TRUE	1	(300,200,100)	50	16	FALSE	111.8481830	4	CA
14	TRUE	1	(200,100,80)	50	16	FALSE	112.3236759	5	CA
1	TRUE	2	(200,100,80)	50	16	FALSE	68.1842181	1	TX
1	TRUE	1	(500,300,150)	50	16	FALSE	71.7268414	2	TX
1	TRUE	3	(600,300,150)	30	16	FALSE	72.4339853	3	TX
1	TRUE	3	(300,200,100)	50	32	FALSE	74.1240856	4	TX
1	TRUE	1	(200,100,80)	50	16	FALSE	75.8850120	5	TX
1	TRUE	2	(200,100,80)	50	16	FALSE	94.9306514	1	WI
1	TRUE	1	(500,300,150)	50	16	FALSE	95.6937640	2	WI
1	TRUE	3	(300,200,100)	50	32	FALSE	96.4596687	3	WI
1	TRUE	3	(600,300,150)	30	16	FALSE	104.4581354	4	WI
1	TRUE	3	(300,200,100)	30	32	FALSE	108.6075724	5	WI

Tabla 9.4. Set de hiperparametros con menor mediana de RMSE teniendo en cuenta las series de tiempo a nivel tienda-departamento separado por estado.

9.3.2.3 Elección final de modelos

Dado que existen diferencias en los errores tanto a nivel estatal, como a nivel tienda, se concluye que es necesario calcular el set de hiper parámetros que minimicen el error de cada serie de tiempo a nivel tienda-departamento por separado.

10. Resultados

10.1 Comparación entre HTS y LSTM

10.1.1 Comparación de modelos que minimizan la mediana de RMSE

Se observa en la tabla 10.1 que el mejor modelo tanto a nivel general como a nivel estatal de HTS se desempeña mejor que el mejor modelo de LSTM:

Comparación de errores entre HTS y LSTM según el nivel de agregación.

Aggregation	Median RMSE - HTS	Median RMSE - LSTM	Difference
Total	72.5920331	87.8491428	21.02%
CA	75.9790263	104.8229430	37.96%
TX	57.3018220	68.1842181	18.99%
WI	84.1950522	94.9306514	12.75%

Tabla 10.1. Comparación de errores entre HTS y LSTM según el nivel de agregación

Probablemente se deba a que no estamos calculando a nivel producto sino a nivel tienda-departamental. Esto provoca que HTS tenga más posibilidades de funcionar mejor ya que la estructura jerárquica tiene más importancia que la variabilidad que pueda haber en el último nivel.

10.1.1 Comparación de modelos que minimizan el error de cada serie por separado

Dado que en la sección 9.3 concluímos que es necesario calcular un set de hiper parámetros distinto para cada serie de tiempo a nivel tienda-departamento, debemos comparar LSTM y HTS a este nivel y calcular en cuántas series HTS se desempeña mejor que LSTM, y viceversa.

En la tabla 10.2, se observa el RMSE mínimo que se obtiene para cada serie de tiempo, optimizando los hiper parámetros para cada serie en particular, tanto para LSTM como para HTS. La columna “HTS es ganador” es un booleano que nos indica si HTS tiene menor RMSE que LSTM. En este caso vemos que en 61 de las 70 series de tiempo tienda-departamentales performa mejor HTS que LSTM.

Departmento	RMSE - LSTM	RMSE - HTS	Diferencia	HTS es ganador
CA1HOB1	114.59	81.38	-28.99%	1
CA1HOB2	14.33	11.75	-18.02%	1
CA1HOU1	203.13	99.67	-50.93%	1
CA1HOU2	59.49	32.20	-45.87%	1
CA1FOO1	70.22	73.29	4.36%	0
CA1FOO2	122.01	90.80	-25.58%	1
CA1FOO3	467.19	263.32	-43.64%	1
CA2HOB1	110.75	88.02	-20.53%	1
CA2HOB2	15.68	14.18	-9.60%	1
CA2HOU1	274.21	222.53	-18.85%	1
CA2HOU2	101.02	84.34	-16.51%	1
CA2FOO1	125.56	96.08	-23.48%	1
CA2FOO2	186.02	154.81	-16.78%	1
CA2FOO3	560.07	430.81	-23.08%	1
CA3HOB1	95.73	76.55	-20.03%	1
CA3HOB2	17.02	15.80	-7.15%	1
CA3HOU1	273.15	163.58	-40.11%	1
CA3HOU2	76.48	48.35	-36.79%	1
CA3FOO1	86.82	92.42	6.44%	0
CA3FOO2	131.54	97.24	-26.08%	1
CA3FOO3	504.70	281.30	-44.26%	1
CA4HOB1	69.72	61.80	-11.37%	1
CA4HOB2	6.15	6.88	11.81%	0
CA4HOU1	82.99	51.78	-37.60%	1
CA4HOU2	35.08	21.08	-39.90%	1
CA4FOO1	44.87	55.86	24.49%	0
CA4FOO2	81.43	50.67	-37.77%	1
CA4FOO3	159.79	105.58	-33.93%	1
TX1HOB1	63.38	49.78	-21.47%	1
TX1HOB2	13.28	11.67	-12.15%	1
TX1HOU1	146.54	105.06	-28.31%	1
TX1HOU2	57.55	33.22	-42.26%	1

TX1FOO1	43.02	45.74	6.32%	0
TX1FOO2	68.26	59.01	-13.55%	1
TX1FOO3	280.68	201.28	-28.29%	1
TX2HOB1	60.54	51.71	-14.57%	1
TX2HOB2	18.07	17.95	-0.67%	1
TX2HOU1	141.76	101.45	-28.43%	1
TX2HOU2	43.29	26.03	-39.87%	1
TX2FOO1	47.00	57.35	22.03%	0
TX2FOO2	69.53	56.71	-18.44%	1
TX2FOO3	351.76	196.70	-44.08%	1
TX3HOB1	74.98	53.99	-27.99%	1
TX3HOB2	16.36	15.93	-2.63%	1
TX3HOU1	128.69	102.84	-20.09%	1
TX3HOU2	36.73	26.50	-27.86%	1
TX3FOO1	64.63	66.08	2.24%	0
TX3FOO2	104.43	92.09	-11.82%	1
TX3FOO3	246.39	206.12	-16.34%	1
WI1HOB1	112.02	77.09	-31.18%	1
WI1HOB2	12.30	11.98	-2.57%	1
WI1HOU1	172.90	96.03	-44.46%	1
WI1HOU2	46.25	26.01	-43.76%	1
WI1FOO1	73.32	70.73	-3.54%	1
WI1FOO2	222.01	122.12	-44.99%	1
WI1FOO3	409.56	233.76	-42.92%	1
WI2HOB1	71.67	41.38	-42.27%	1
WI2HOB2	12.19	11.13	-8.74%	1
WI2HOU1	272.74	197.70	-27.51%	1
WI2HOU2	32.13	23.18	-27.85%	1
WI2FOO1	109.91	99.45	-9.52%	1
WI2FOO2	454.90	381.91	-16.05%	1
WI2FOO3	557.40	468.17	-16.01%	1
WI3HOB1	58.59	44.55	-23.97%	1
WI3HOB2	10.44	10.78	3.23%	0

WI3HOU1	152.31	98.25	-35.49%	1
WI3HOU2	50.08	24.76	-50.56%	1
WI3FOO1	72.24	65.77	-8.96%	1
WI3FOO2	149.38	155.22	3.91%	0
WI3FOO3	442.60	306.13	-30.83%	1

Tabla 10.2. Comparación de errores entre HTS y LSTM a nivel Tienda-Departamento tomando en cuenta el modelo que minimiza RMSE para cada serie en particular

De las 9 series en las que LSTM es más preciso, 6 pertenecen al departamento FOOD 1.
En estas 9 series, la diferencia en el RMSE es casi insignificante entre HTS y LSTM, como se observa en la tabla 10.3.

Department	RMSE - LSTM	RMSE - HTS	Difference
CA1FOO1	70.22	73.29	4.36%
CA3FOO1	86.82	92.42	6.44%
CA4HOB2	6.15	6.88	11.81%
CA4FOO1	44.87	55.86	24.49%
TX1FOO1	43.02	45.74	6.32%
TX2FOO1	47.00	57.35	22.03%
TX3FOO1	64.63	66.08	2.24%
WI3HOB2	10.44	10.78	3.23%
WI3FOO2	149.38	155.22	3.91%

Tabla 10.3. Series de tiempo donde LSTM se desempeña mejor que HTS tomando en cuenta el modelo que minimiza RMSE para cada serie en particular

10.2 Elección de técnica ganadora

Por todo lo expuesto, podemos concluir que HTS fue capaz de realizar predicciones mucho más precisas y coherentes para casi todas las series de tiempo a nivel tienda-departamento, tanto si tomamos sólo un modelo que minimice la mediana de la distribución de RMSE o si tomamos un modelo en particular que minimice el RMSE de cada departamento.

11. Conclusiones

Las técnicas clásicas para modelar y predecir series de tiempo, como ARIMA y ETS, no son óptimas, por sí solas, cuando el conjunto de series a predecir está contenido dentro de una jerarquía. Esto se debe a que la información disponible dentro de una estructura jerárquica puede ser muy útil a la hora de realizar predicciones ya que un nivel particular de la estructura puede revelar características de la data que son importantes para ser modeladas. Estas características pueden estar completamente ocultas o no fácilmente identificables en otros niveles.

En el análisis exploratorio, se observó cómo el comportamiento de una serie de tiempo cambiaba si pertenecía a una categoría u otra, o si la tienda estaba en California o Wisconsin. El consumidor actuaba distinto según cada una de estas variables. Por ejemplo, para un evento importante como el Super Bowl el patrón de compra era muy distinto según si el departamento pertenecía a Food o Household, ya que satisfacen necesidades diferentes.

En cuanto al modelado, se compararon todas las series de tiempo a nivel tienda-departamento entre LSTM y HTS, optimizando los hiper parámetros para cada serie en particular. Además se analizó si alguna técnica era óptima para algún set de series de tiempo agrupadas por estado o categoría. HTS generó modelos más robustos que LSTM, no sólo a nivel general, sino también agrupando las series por algún estado o categoría. Esto quiere decir que se desempeñó mejor que LSTM sin importar la característica de la serie de tiempo.
El error, calculado como RMSE, fue menor en 61 de las 70 series analizadas. Y en las series en las que tenía un error mayor, la diferencia era baja. Probablemente esto se deba a que el trabajo está hecho a nivel tienda-departamental (es decir ya con una agregación a nivel producto), lo cual le da mayor posibilidad de aprender a HTS ya que depende mucho más de la estructura jerárquica que si hubiese sido a nivel producto, donde la variabilidad juega un papel más importante.

11.1 Próximos pasos - Recomendaciones

- HTS demostró ser una técnica valiosa y bien fundamentada pero queda aún demostrar su eficacia a nivel producto, donde hay mucha mayor variabilidad.
- En LSTM hay muchas otras maneras en las que se podría optimizar y observar si su error disminuye. Por ejemplo, aumentando la cantidad de capas en serie.
- La arquitectura de la Red Neuronal podría complejizarse aún más, siendo LSTM una parte dentro de una Red Neuronal mayor que contenga otro tipo de componentes.
- Ambas técnicas pueden combinarse entre sí para generar modelos más robustos: las técnicas de reconciliación de HTS pueden usarse con predicciones base realizadas por LSTM u otro tipo de técnica que no sea ARIMA o ETS.

12. Referencias-Bibliografía

1. A. O. Akyuz, M. Uysal, B. A. Bulbul and M. O. Uysal.(2017). "Ensemble approach for time series analysis in demand forecasting: Ensemble learning," 2017 IEEE International Conference on Innovations in Intelligent SysTems and Applications (INISTA), Gdynia, 2017, pp. 7-12, doi: 10.1109/INISTA.2017.8001123.

2. Athanasopoulos, G. and Kourentzes, N.(2020). On the evaluation of hierarchical forecasts (Department of Econometrics and Business Statistics Working Paper Series 02/2020).

3. Statista. (2021, July 30). World: Leading retailers 2019, by retail revenue. Retrieved October 18, 2021, from https://www.statista.com/statistics/266595/leading-retailers-worldwide-based-on-revenue/

4. Ghassen Chniti, Houda Bakir, and Hédi Zaher (2017). E-commerce Time Series Forecasting using LSTM Neural Network and Support Vector Regression. In Proceedings of the International Conference on Big Data and Internet of Thing (BDIOT2017). Association for Computing Machinery, New York, NY, USA, 80–84. DOI:https://doi.org/10.1145/3175684.3175695

5. Hyndman, R.J., & Athanasopoulos, G. (2018) Forecasting: principles and practice, 2nd edition, OTexts: Melbourne, Australia. OTexts.com/fpp2. Accessed on October 2021

6. Athanasopoulos, G., Ahmed, R. A., & Hyndman, R. J. (2009). Hierarchical forecasts for Australian domestic tourism. International Journal of Forecasting, 25, 146–166.

7. Hyndman, R. J., Ahmed, R. A., Athanasopoulos, G., & Shang, H. L. (2011). Optimal combination forecasts for hierarchical time series. Computational Statistics & Data Analysis

8. Mergani A. Khairalla and Xu Ning. (2017). Financial Time Series Forecasting Using Hybridized Support Vector Machines and ARIMA Models. In Proceedings of the 2017 International Conference on Wireless Communications, Networking and Applications (WCNA 2017). Association for Computing Machinery, New York, NY, USA, 94–98. DOI:https://doi.org/10.1145/3180496.3180613

9. Box, G.E.P. and Jenkins, G.M. (1976) Time Series Analysis: Forecasting and Control. Revised Edition, Holden Day, San Francisco

10. Hyndman, R. J., Lee, A. J., & Wang, E. (2016). Fast computation of reconciled forecasts for hierarchical and grouped time series. Computational Statistics & Data Analysis, 97, 16–32. doi:10.1016/j.csda.2015.11.007

11. Sepp Hochreiter, Jürgen Schmidhuber; Long Short-Term Memory. Neural Comput 1997; 9 (8): 1735–1780. doi: https://doi.org/10.1162/neco.1997.9.8.1735

12. Goodfellow, I., Bengio, Y., & Courville, A. (2016). Deep Learning. MIT Press.

13. G. van Rossum, Python tutorial, Technical Report CS-R9526, Centrum voor Wiskunde en Informatica (CWI), Amsterdam, May 1995.

14. Charles R. Harris, K. Jarrod Millman, Stéfan J. van der Walt, Ralf Gommers, Pauli Virtanen, David Cournapeau, Eric Wieser, Julian Taylor, Sebastian Berg, Nathaniel J. Smith, Robert Kern, Matti Picus, Stephan Hoyer, Marten H. van Kerkwijk, Matthew Brett, Allan Haldane, Jaime Fernández del Río, Mark Wiebe, Pearu Peterson, Pierre Gérard-Marchant, Kevin Sheppard, Tyler Reddy, Warren Weckesser, Hameer Abbasi, Christoph Gohlke & Travis E. Oliphant. Array programming with NumPy, Nature, 585, 357–362 (2020), DOI:10.1038/s41586-020-2649-2

15. John D. Hunter. Matplotlib: A 2D Graphics Environment, Computing in Science & Engineering, 9, 90-95 (2007), DOI:10.1109/MCSE.2007.55

16. Waskom, M. L. (2021). seaborn: statistical data visualization. Journal of Open Source Software, 6(60), 3021. doi:10.21105/joss.03021

17. Wes McKinney. Data Structures for Statistical Computing in Python, Proceedings of the 9th Python in Science Conference, 51-56 (2010)

18. Chollet, F., & Others. (2015). Keras. Opgehaal van https://keras.io

19. Hyndman, R. J., Athanasopoulos, G., & Shang, H. L. (2013). hts : An R Package for Forecasting Hierarchical or Grouped Time Series.

20. NIST/SEMATECH e-Handbook of Statistical Methods, https://www.itl.nist.gov/div898/handbook/eda/section3/eda35c.htm, 17 Octubre 2021.

21. R Core Team (2020). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. URL https://www.R-project.org/.

22. RStudio Team (2019). RStudio: Integrated Development for R. RStudio, Inc., Boston, MA URL http://www.rstudio.com/.

23. M5 Forecasting - Accuracy | Kaggle. (n.d.). Kaggle. Retrieved June 1, 2020, from https://www.kaggle.com/c/m5-forecasting-accuracy/data.