PLANTILLA

PROYECTO FINAL DEL POSTAGRADO EN DATA SCIENCE Y MACHINE LEARNING

ALUMNO 1: Axel Cruz Vázquez

ALUMNO 2: Salvador Escudero Cachay

ALUMNO 3: José Leonardo Chaves Chacón

PROGRAMA:

POSTGRADO EN DATA SCIENCE Y MACHINE LEARNING

NOMBRE DEL PROYECTO: Desarrollo de un modelo en Python para el análisis y proyección de ventas en las tiendas de Walmart

Contenido

RESUMEN	3
INTRODUCCIÓN	3
ESTADO DEL ARTE	6
OBJETIVOS	13
SOLUCIÓN PLANTEADA	13
EVALUACIÓN	29
RESULTADOS	31
CONCLUSIONES Y TRABAJOS FUTUROS	34
REFERENCIAS	36

RESUMEN

El problema planteado para el desarrollo de este proyecto es el análisis, proyección de ventas, efectos de ciertos indicadores y factores externos para las tiendas de Walmart según la base de datos posteada en Kaggle. Esta base de datos muestra información histórica de 45 tiendas ubicadas en diferentes regiones desde el 2010 hasta el 2011, donde se detalla el número de tienda (que incluye el número de departamentos internos de cada tienda), semana de ventas, días festivos, tasa de desempleo del área, índice de precios al consumidor, costo del combustible para la región, las ventas generales y por departamento de cada tienda.

Para obtener los resultados deseados se realizará una depuración de datos, su análisis y la comprensión de estos para determinar indicadores o comportamientos relevantes de la información, usando para este cometido el entorno de distribución de Anaconda y con el uso de lenguaje Python (cuadernos de Jupyter) el cual permitirá el proceso de exploración, análisis de los datos y predicción, de una forma rápida y precisa. Una vez analizados y depurados los datos se procede a aplicar técnicas de Machine Learning de aprendizaje supervisado, las cuales permitirán la predicción y proyección de datos faltantes o bien a crear las estimaciones sobre las ventas futuras en las tiendas de Walmart.

Como objetivo principal, además del análisis, es la proyección de las ventas para Walmart; logrando con ello la determinación de promociones, descuentos, estrategias de merchandising (dando foco a una estrategia por departamentos de venta de cada tienda) y una mejor gestión como la administración del personal, mejores espacios dispuestos para el público logrando así optimizar los costos y elevar los ingresos por ventas.

INTRODUCCIÓN

El presente trabajo se desarrolla con el objetivo de crear una proyección de ventas para las 45 tiendas de Walmart ubicadas en diferentes regiones, la detección de la necesidad u oportunidad se logró al identificar por medio del posteo de su base de datos del 2010 al 2011 en la página de Kaggle donde Walmart solicita o crea un concurso para el desarrollo de algoritmos que permitan el análisis de ventas basados en datos históricos; como factores externos tenemos: temperaturas, costos de combustibles, días festivos, índice de precios al

consumidor, etc.; como datos internos tenemos las ventas por tiendas (45 en total), ventas por departamentos (99 por cada tienda) y la determinación de rebajas; estas variables son claves para la proyección de los datos faltantes (trabajo de predicción).

Durante la investigación se logró identificar, que generalmente las empresas u instituciones, utilizan distintas soluciones aplicadas para los problemas de análisis, donde por ejemplo, las ventas (con sus diversas variables claves para predicciones) son llevadas a hojas de cálculo en MS Excel donde se emplean gráficas, esquemas estadísticos, tablas dinámicas, funciones y formulas para las proyecciones, siendo estas muy básicas y limitadas; también se encontraron soluciones donde se utilizan métodos de probabilidad estadística y modelos matemáticos con logaritmos para el caso de uso especifico, método de incrementos porcentuales, incrementos absolutos o mínimos cuadraros. También se logró constatar, que para la mayoría de los casos, se aplican técnicas de machine learning (que incluyen logaritmos y librerías avanzadas) en entornos de programación con lenguajes Python y/o R con diferentes modelos de aprendizaje automático (supervisado y no supervisado), de los cuales destacan esquemas aplicando regresiones lineales, arboles de decisiones, cuantificadores bayesianos, el método de XGBoost y SMV, etc. los cuales pueden comportarse de distintas formas y con ello se pueden obtener diferentes porcentajes de predicciones.

Cada modelo se configura según el caso de uso a tratar, teniendo en cuenta los conjuntos de datos que se tiene y la taza de precisión para los resultados de predicción.

Con la investigación anterior se determinó que los métodos más recientes y los que brindan mejores resultados (con una forma más eficiente) son los relacionados a los modelos de machine learning.

Analizando las bases de datos de Walmart y debido a que es un problema de regresión (modelo de aprendizaje supervisado), con variables tanto categóricas como continuas se determinó aplicar el algoritmo del Random Forest por ser el mejor para el caso ya que generalmente maneja bien los conjuntos de datos con este tipo de particularidades, donde dependiendo de los parámetros configurados, este algoritmo minimiza la posibilidad de sobre ajustarse al incorporar múltiples árboles de decisión en paralelo y tomar el valor más frecuente de los resultados, es decir evita o reduce la posibilidad de ajustarse al conjunto de datos de entrenamiento y no de forma generalizada, perdiendo precisión al recibir algún dato diferente al que fue entrenado el modelo.

Para el desarrollo de la solución planteada de Walmart, con el uso de técnicas de Machine Learning para el manejo de los datos, se inició con el proceso de análisis empleando y ejecutando código en el entorno de Python, importando las librerías necesarias para el manejo de dataframe, estadísticas y graficas para visualización de resultados; permitiendo ver los valores de los conjuntos de datos de train (datos de entrenamiento), test (datos de prueba), stores (datos de las tiendas), features (datos y descripciones generales), etc., efectuando un tratamientos a los formatos de los datos, convirtiendo los mismos a los adecuados y se realizaron uniones de las bases de datos para ejecutar los análisis y predicciones de forma precisa.

Dentro del tratamiento y análisis realizado, se trató los datos de fecha configurando como formato "fecha" (Date), también se efectúa un análisis estadístico de las bases para considerar e identificar información clave o trascendente; realizando un muestreo descriptivo presentando tablas y gráficos de las variables suministradas de las tiendas, ventas por semanas, ventas por departamento, análisis de los outliers (datos anómalos), análisis de las ventas en días festivos o no para determinar si existe una afectación; análisis de las ventas mensuales y análisis de las correlaciones entre los descuentos o rebajas; y también de las variables de ventas y su correspondiente correlación con los descuentos durante los días festivos.

Con el tratamiento de los datos y los análisis obtenidos según cada paso aplicado se decide realizar, o aplicar a los datos ya depurados, un modelo de predicción de Random Forest, con este mismo modelo se utiliza el método de eliminación de rasgos recursivos para identificar, también, cuáles son las variables más relevantes, con lo anterior se determina que las principales variable son departamento, tamaño, tienda, numero de semana, entre otras y sobre las variables se aplican modelos de predicción que consideran la afectación con todas las variables, la cual brinda una predicción del 96,8%; utilizando, otro modelo con solo 8 de las variables más relevantes, se obtuvo un porcentaje 97,3%; luego se aplicó otra prueba, con un modelo con 4 de las variables más relevantes donde se obtuvo un 97,2% de precisión.

Como se puede observar el modelo de mayor precisión fue al utilizar 8 variables, sobre este punto se ejecutó un código para hacer el modelo aún mas robusto donde se ajustan los parámetros de Random Forest para obtener el WMAE más pequeño con la herramienta GridSearchCV logrando una precisión final de 97,7% y un WMAE de 1528,95; una vez que se logro aumentar y mejorar el modelo se ejecuta, con este, el método de predicción de ventas y

sobre este, como valor agregado, se genera o se extrae de todo el análisis un archivo de Excel (.csv) con la predicción de las ventas.

Con esto y como punto final se obtiene un archivo en Python con la siguiente estructura que permite el análisis y la predicción de los datos:

- Carga de librerías para el desarrollo del modelo
- Carga de bases de datos suministradas
- Analisis de bases de datos
- Depuración de bases de datos
- Analisis de indicadores sobre las bases de datos
- Analisis de correlaciones
- Analisis de los factores más importantes para utilizar en el modelo de predicción
- Analisis y ejecución de los modelos de predicción con diferentes factores
- Optimización del mejor modelo identificado
- Ejecución y desarrollo del grafico sobre el modelo desarrollado
- Extracción o generación de los datos en formato excel

ESTADO DEL ARTE

A continuación, se realizará un breve resumen sobre 15 de los artículos investigados los cuales brindan soluciones o justifican el uso de técnicas tales como machine learning o aprendizaje autónomo para la proyección de indicadores financieros:

1. **Desarrollo de un modelo basado en Machine Learning para la predicción de la demanda de habitaciones en el sector hotelero:** Se entrenaron y validaron diferentes modelos para predecir la ocupación diaria de un hotel, todo esto con el propósito de facilitar a los administradores hoteleros la toma de decisiones. El entrenamiento y validación se realizó utilizando cuatro técnicas de Machine Learning (Ridge Regression, Kernel Ridge Regression, Redes Neuronales Artificiales perceptron multicapa y de Función Base Radial). Los datos se separaron en tres conjuntos: entrenamiento, validación y pruebas, para

completar este proyecto se desarrolla un algoritmo utilizando el lenguaje de programación Python que permite realizar experimentos utilizando las técnicas anteriores, con esto se logra realizar una proyección adecuada de la demanda para el hotel.

2. **Predicción de la Demanda de un Nuevo Producto para una Empresa Importadora usando Series de Tiempo:** Para el desarrollo de este proyecto se realizó una investigación mediante series de tiempo, utilizando la metodología CRISP-DM donde se intenta encontrar un modelo que permita realizar pronósticos sobre la tendencia y estacionalidad de las ventas de este nuevo producto. El desarrollo central del trabajo incluye secciones de análisis del negocio y de datos, que permiten tener una base sólida de conocimiento para enfrentar la tarea. Se realizan varias iteraciones que van intentando mejorar la precisión predictiva a través de distintos métodos como la descomposición, ARIMA, suavizamiento exponencial, árbol de decisión y redes neuronales. Posteriormente se ejecuta una técnica llamada stacking¸ la que propone utilizar más de un método para la obtención de un modelo de pronósticos, permitiendo mejorar de manera considerable la precisión de los resultados obtenidos.

3. **Machine learning y retail:** El comercio minorista se basa en datos que deben ser analizados, por eso la tecnología ha adquirido gran importancia, al punto de que hoy podemos hablar de la existencia de una relación entre el Machine learning y retail, es por eso que los minoristas han posado su mirada sobre el Machine learning. Dentro de los beneficios son: Respuesta inmediata, conocer al cliente, predecir el comportamiento del comprador y análisis de datos, esta tecnología sabe que productos son comprados periódicamente por el cliente y cuales han sido adquiridos rara vez.

4. **Pronostico de ventas usando redes neuronales:** Para la solución al problema se propone una red neuronal que sea alimentada con los niveles de ventas de períodos de tiempo anteriores al que se desea estimar y con un parámetro que informe el período que se desea estimar dado que el problema que se está

abordando es de tipo estacional. Con toda esta información la red debe pronosticar un nivel de ventas partiendo del comportamiento histórico de dicha variable. La red usada para esta aplicación es una backpropagation con momentum (parámetro). Después de realizar varios ensayos se obtuvo que el número de datos de períodos anteriores con que se podría hacer una buena predicción es de un trimestre para predecir las ventas del cuarto mes. Además, este valor permite que el número de patrones aún sea representativo.

5. **Construcción de modelo de forecast para estimación de demanda en retail:** Se desarrollo un modelo para la empresa Mars, empresa multinacional productora de dulces. El objetivo es mejorar la manera en que se predice la demanda, teniendo en cuenta la historia de venta de cada producto en cada cliente, donde se mejorará o mantendrá el indicador de precisión con el cual la compañía trabaja: un mínimo de 70 %. Para realizar lo anterior se utilizó la herramienta Azure ML junto con el lenguaje de programación R, con los que se usaron las funciones de red neuronal artificial para la predicción. Los resultados obtenidos fueron satisfactorios.

6. **Como venderás más con machine learning en retail:** El Machine Learning en Retail te servirá para optimizar la satisfacción del cliente. Le darás un producto especializado, concreto y que se adapta a sus necesidades, gustos y presupuesto. Por ejemplo, acudes a una cafetería por tu café, quien te atiende te pondrá el café ardiendo, el pan muy tostado y con dos sobres de azúcar. Sin embargo, conforme vayas yendo más a menudo, el personal irá conociendo tus gustos y se adaptará a ellos, todo con tal de darte el mejor servicio. El Machine Learning en Retail busca dar esa satisfacción de cliente que no podrá dar nunca una gran superficie. Aunque la señora que regenta la mercería del barrio conozca tus necesidades al vestir, difícilmente lo hará la chica que te atiende en un centro comercial. Y ahí entra en juego el robot donde podremos conocer al cliente tan bien como él mismo, podremos ofrecerle lo mejor, cuando necesite y de la forma que necesite. El machine learning ayuda a crear mejores campañas

de marketing, generar una experiencia de compra óptima, reducir la pérdida de clientela en el proceso de compra.

7. **Machine learning en la predicción de la demanda:** Cuando un sistema de machine learning se alimenta de datos, busca patrones y aún va más allá, puede utilizar los patrones que identifica en los datos para tomar mejores decisiones. El aprendizaje automático posibilita que al pronóstico de retail se le pueda incorporar la gran variedad de factores y relaciones que afectan la demanda a diario. Esto tiene una gran importancia, ya que sólo los datos meteorológicos por sí solos pueden estar compuestos por cientos de factores que potencialmente, pueden impactar en la demanda. Los algoritmos de machine learning generan automáticamente modelos de mejora continúa usando solamente los datos que se les proporciona, ya sean de nuestro negocio o de flujos de datos externos. El principal beneficio es que un sistema como éste puede procesar conjuntos de datos a escala retail desde una variedad de fuentes, todo sin intervención humana.

8. **Beneficios que aporta machine learning al sector retail:** Con el Machine Learning, los retails pueden mejorar procesos administrativos, la gestión de inventarios, la cadena de suministro, la atención a clientes, campañas publicitarias, promociones, sistemas de cobro. Con machine learning podemos analizar tendencias, patrones y hábitos de compra, optimización del espacio de venta, optimizar los espacios en el piso de ventas, diseño de promociones y fijación de precios considerando la oferta y la demanda o temporadas y automatización de la atención al cliente.

9. **Big Data y Retail, como mejorar los objetivos de negocio:** Recordemos que las predicciones con Machine Learning mejoran cuando se disponen de muchos datos, de diferentes tipos y que se actualizan con el tiempo. Precisamente en retail los sistemas transaccionales generan gran cantidad de datos, se pueden extraer datos de comportamiento de la información de los clientes identificados en un gran número de casos a través de los programas de fidelización y además

los sistemas de posicionamiento en el punto de venta generan información sobre la ubicación de los clientes.

10. **Big Data y el analisis predictivo para aumentar ventas:** Para poder realizar un análisis predictivo es necesario capturar suficiente información del tema que nos interese. Pues el análisis no es más que la reducción de la información contenida en los cuantiosos datos. De allí, la relación entre ambos términos, de este modo, el primer análisis toma toda la información sobre prospectos y clientes considerando una recopilación detallada de la base de datos. Como segundo paso, analiza la web en busca de información complementaria sobre estos contactos, es decir, si ya se contaba con el email de un usuario, el análisis relacionará ese mismo email en busca de más información. Por lo que es probable que también se vincule a redes sociales del mismo usuario. Los alcances del análisis predictivo podrían indicar qué prospectos tienen más probabilidades de cerrar una venta con la marca, así como qué porcentaje de los clientes podrían recurrir a una segunda compra.

11. **El Poder de Big Data en logística:** Los estudios de caso de Big Data en logística revelan cómo los datos y el análisis en gestión de almacenes se pueden aplicar a los procedimientos y procesos para mejorar la eficiencia y la precisión. Los datos pueden ayudar a mejorar el almacenamiento, la manipulación, el transporte y otros procesos. Los datos en logística se pueden utilizar para reducir las ineficiencias en la entrega, brindar transparencia a la cadena de suministro, optimizar las entregas, proteger los productos perecederos y automatizar toda la cadena de suministro.

12. **Análisis de Big Data para la gestión de riesgos:** Ser capaz de prever un riesgo potencial y mitigarlo antes de que ocurra es fundamental para que la empresa siga siendo rentable. Hasta ahora, el análisis de big data ha contribuido al desarrollo de soluciones de gestión de riesgos. Las herramientas permiten a las empresas cuantificar y modelar los riesgos a los que se enfrentan todos los días.

Un sistema de análisis de big data adecuado ayuda a garantizar que se identifiquen las áreas de debilidades o riesgos potenciales.

13. **Cómo Netflix resolvió su problema de recomendación con la ciencia de datos.** En 2006, cuando Netflix quería entrar en el mercado de la transmisión, comenzó con una competencia por la predicción de la clasificación de películas. Proporcionó un premio de $ 1 millón a quien haya aumentado la precisión de su plataforma "Cinematch" en un 10%. Al final de la competencia, el equipo de BellKor presentó su solución que aumentó la precisión de la predicción en un 10.06%. Con más de 200 horas de trabajo y un conjunto de 107 algoritmos les proporcionó este resultado. Su modelo final dio un RMSE de 0.8712. Para su solución, utilizaron el algoritmo de vecino más cercano K para el posprocesamiento de los datos. Luego implementaron un modelo de factorización que se conoce popularmente como Descomposición de Valor Singular para proporcionar una integración dimensional óptima a sus usuarios.

14. **Predecir la mejor ubicación de tienda minorista**. Uno de los verdaderos factores del éxito empresarial es "Ubicación ". Probablemente hayas visto que esto es cierto cuando ves un lugar que siempre tiene un nuevo restaurante o tienda. Por alguna razón, nunca tendrá éxito. Esto obliga a las empresas a pensar detenidamente cuál es la mejor ubicación para su negocio. La respuesta es dónde están sus clientes cuando piensan en su producto. ¿Pero dónde está eso? De hecho, algunas empresas están adoptando este ejemplo. Un ejemplo es Buxtonco. ¡Buxtonco está respondiendo dónde debería abrir su próximo negocio con datos! Su sitio exclama: "Que cualquier minorista puede lograr un mayor éxito y crecimiento al comprender a su cliente y que existe una ciencia detrás de identificar quién es ese cliente, dónde viven los clientes potenciales y qué clientes son los más valiosos". Al buscar dónde pueden pasar el tiempo sus clientes y qué podrían estar haciendo en determinadas ubicaciones, la tecnología puede ayudar a determinar dónde sería mejor abrir su próximo negocio.

15. **Predecir por qué los pacientes de hospitales están siendo readmitidos.** Ser capaz de predecir la readmisión de pacientes puede ayudar a los hospitales a reducir sus costos y a mejorar la salud de la población. Saber quién es probable que sea readmitido también puede ayudar al científico de datos a encontrar el "por qué" detrás de la readmisión de poblaciones específicas. Uno de los enfoques comunes es investigar los vínculos entre la readmisión y los puntos de datos socioeconómicos como los ingresos, las direcciones, las tasas de delincuencia y la contaminación del aire. Similar a la forma en que los especialistas en marketing se dirigen a los clientes mediante el aprendizaje automático y los sistemas de recomendación de productos que tienen en cuenta los puntos de datos socioeconómicos para indicar cómo vender a un cliente. Los hospitales están tratando de adaptar mejor su atención para ayudar a sus pacientes basándose en cómo otros pacientes similares han respondido en el pasado. Por lo tanto, poder averiguar el por qué detrás de la readmisión puede a su vez solucionarlo.

OBJETIVOS

Objetivo general

Desarrollar un modelo que permita el análisis y predicción de las ventas para las tiendas de Walmart logrando la eficientización de estrategias, procesos y costos aplicando las técnicas estudiadas en el programa en un caso real y con datos válidos

Objetivos específicos:

1. Establecer un modelo que permite la predicción de las ventas para las tiendas Walmart, mejorando con ello costos y creación de mejores estrategias.

2. Determinar las técnicas de machine Learning para analizar las ventas por departamento y analizar los efectos de los descuentos realizados por las tiendas Walmart en días festivos.

3. Determinar el impacto que las tiendas Walmart reciben en sus ventas en relación con los diferentes factores.

4. Establecer las recomendaciones basadas en las técnicas y modelos para las tiendas Walmart.

SOLUCIÓN PLANTEADA

Como solución al problema presentado por Walmart para el análisis de predicción de las ventas, se dio la tarea de investigar los diferentes modelos que pueden ser ejecutados dentro de machine learning además de una investigación profunda sobre los casos ya completados por terceros, una vez que se logro comprender las dimensiones del caso, se procedió a analizar dentro de la página de Kaggle las descripciones y el detalle sobre el contenido de cada una de las bases suministradas para poder aplicar los criterios y desarrollar el modelo acorde a lo solicitado.

El conjunto de datos utilizados fueron las bases que Walmart posteo en la página de Kaggle como un concurso donde se extrae, se analiza y se utiliza los archivos “features”, “sample Submission”, “stores”, “test” y “train”.

Para los procesos de manejo de datos se utilizó ETL (extraer, transformar y cargar), en la primer fase se extraen las bases de las fuentes suministradas desde la página de Kaggle, una vez extraídos y cargados en Python se realiza el procesos de transformación donde se modifican los datos para obtener los formatos correctos o deseados para evitar complicaciones durante el desarrollo o ejecución de los modelos, para ello se realizó la normalización donde se define que datos entraran en juego (ventas semanales, ventas mensuales, tipos de tiendas, indicadores como combustible, tasa de desempleo, días festivos, etc.) se realiza la eliminación de datos duplicados, se eliminan los datos nulos o faltantes, se verifican los datos y se clasifican logrando una mayor eficiencia al agrupar los mismos y por ultimo al completar todo el análisis y desarrollo de los modelos planteados se extrae un archivo de MS Excel (.csv) que es cargado en el computador.

Para poder generar una respuesta al caso por medio de machine learning y como se comento anteriormente se cargaron las distintas librerías necesarias para el análisis, creación de gráficos, para omisión de mensajes de advertencia, procesamiento de la data y ejecución de los modelos de predicción:

```
In [1]: from IPython.core.display import display, HTML
        HTML("""<style>.output_png{display: table-cell;    text-align: center;    vertical-align: middle;}</style>""")
        import numpy as np
        import pandas as pd # procesamiento de la data
        import matplotlib.pyplot as plt # Visualizacion
        import seaborn as sns # Visualizacion
        from scipy import stats
        from scipy.stats import norm
        import warnings
        warnings.filterwarnings('ignore') #ignorar alertas

        %matplotlib inline
        import gc
```

Una vez cargadas las librerías, se extraen los datos tomados desde la página de kaggle, estos fueron alojados en el computador local inicialmente y posterior a ello cargados en la libreta de Python, para ello se cargó los archivos “train”, “test”, “ stores” y “features” además a su vez se realiza el ajuste de las fechas para los documentos train y test.

```
In [2]:
train = pd.read_csv('train.csv', parse_dates=["Date"])
test = pd.read_csv('test.csv', parse_dates=["Date"])
stores = pd.read_csv('stores.csv')
features = pd.read_csv('features.csv')
```

Cargados los datos se ejecuta la visualización de cada una de las bases suministradas para determinar y conocer su contenido, esto con el propósito de familiarizarse con los datos, conocer e identificar valores trascendentes e identificar valores nulos o incompletos a tratar

```
In [3]: train.head()
```

Out[3]:

	Store	Dept	Date	Weekly_Sales	IsHoliday
0	1	1	2010-02-05	24924.50	False
1	1	1	2010-02-12	46039.49	True
2	1	1	2010-02-19	41595.55	False
3	1	1	2010-02-26	19403.54	False
4	1	1	2010-03-05	21827.90	False

```
In [4]: test.head()
```

Out[4]:

	Store	Dept	Date	IsHoliday
0	1	1	2012-11-02	False
1	1	1	2012-11-09	False
2	1	1	2012-11-16	False
3	1	1	2012-11-23	True
4	1	1	2012-11-30	False

```
In [5]: stores.head()
```

Out[5]:

	Store	Type	Size
0	1	A	151315
1	2	A	202307
2	3	B	37392
3	4	A	205863
4	5	B	34875

```
In [6]: features.head()
```

Out[6]:

	Store	Date	Temperature	Fuel_Price	MarkDown1	MarkDown2	MarkDown3	MarkDown4	MarkDown5	CPI	Unemployment	IsHoliday
0	1	2010-02-05	42.31	2.572	NaN	NaN	NaN	NaN	NaN	211.096358	8.106	False
1	1	2010-02-12	38.51	2.548	NaN	NaN	NaN	NaN	NaN	211.242170	8.106	True
2	1	2010-02-19	39.93	2.514	NaN	NaN	NaN	NaN	NaN	211.289143	8.106	False
3	1	2010-02-26	46.63	2.561	NaN	NaN	NaN	NaN	NaN	211.319643	8.106	False
4	1	2010-03-05	46.50	2.625	NaN	NaN	NaN	NaN	NaN	211.350143	8.106	False

Una vez que se logra visualizar la data y conocer su contenido se procede con la fase de depuración con la finalidad de excluir o reemplazar todos los valores nulos e incompletos, esto aplica para las bases llamada “train”, “test”

```
In [8]: train = train.replace('None', np.nan)
        train = train.replace('NaN', np.nan)
        train = train.replace('NaT', np.nan)
        train = train.replace('', np.nan)
        train_nulls = (train.isnull().sum(axis = 0)/len(train))*100
        train_nulls

Out[8]: Store           0.0
        Dept            0.0
        Date            0.0
        Weekly_Sales    0.0
        IsHoliday       0.0
        dtype: float64
```

```
In [9]: test = test.replace('None', np.nan)
        test = test.replace('NaN', np.nan)
        test = test.replace('NaT', np.nan)
        test = test.replace('', np.nan)
        test_nulls = (test.isnull().sum(axis = 0)/len(test))*100
        test_nulls

Out[9]: Store        0.0
        Dept         0.0
        Date         0.0
        IsHoliday    0.0
        dtype: float64
```

```
In [10]: train = train.fillna(0)
         test = test.fillna(0)

         train.isnull().sum()

Out[10]: Store           0
         Dept            0
         Date            0
         Weekly_Sales    0
         IsHoliday       0
         dtype: int64
```

Una vez que los datos se encuentran depurados se procede a realizar un análisis estadístico de la base “train” donde se determina que posee un total de 421.570 líneas con información, con la máxima se identifica que existen 99 departamentos, 45 tiendas.

```
In [11]: train.describe().T
```

Out[11]:

	count	mean	std	min	25%	50%	75%	max
Store	421570.0	22.200546	12.785297	1.00	11.00	22.00	33.0000	45.00
Dept	421570.0	44.260317	30.492054	1.00	18.00	37.00	74.0000	99.00
Weekly_Sales	421570.0	15981.258123	22711.183519	-4988.94	2079.65	7612.03	20205.8525	693099.36

Se procede a analizar un análisis por tipo de tienda para determinar como se distribuyen sus ventas y conocer los tamaños sobre cada una de las mismas, por medio de este se rescata que son un total de 45 tiendas y se encuentran conformadas por 3 grupos, el grupo A hace referencia a las tiendas de mayor tamaño, seguido por el grupo B y posterior a ello por el grupo C

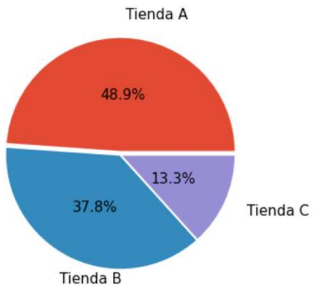
De las 45 tiendas existentes, 22 son tipo A, 17 son tipo B y 6 tipo C

```
In [12]: print("el tamaño del data set es", stores.shape)
         print("Los valores de las tiendas son", stores['Store'].unique())
         print("Los valores de los tipos de tiendas son", stores['Type'].unique())
```

```
el tamaño del data set es (45, 3)
Los valores de las tiendas son [ 1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24
 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45]
Los valores de los tipos de tiendas son ['A' 'B' 'C']
```

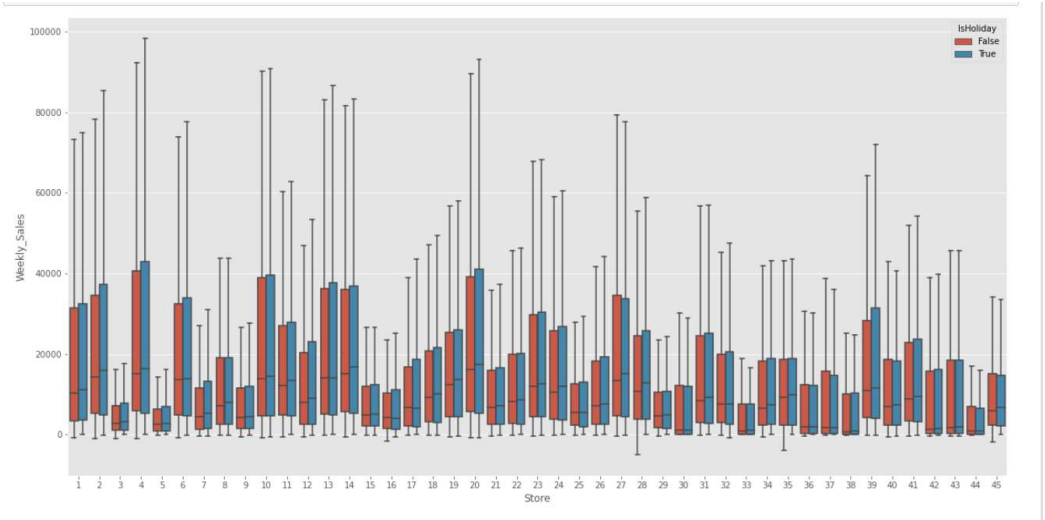
Y se grafican los datos anteriores de forma manual con los resultados obtenidos en el paso anterior para una mejor representación y visualización de la distribución por tipo de tienda

```
In [14]: plt.style.use('ggplot')
labels=['Tienda A','Tienda B','Tienda C']
sizes=grouped.describe()['Size'].round(2)
sizes=[(22/(17+6+22))*100,(17/(17+6+22))*100,(6/(17+6+22))*100]


fig, axes = plt.subplots(1,1, figsize=(7,7))

wprops={'edgecolor':'white',
      'linewidth':2}

tprops = {'fontsize':15}

axes.pie(sizes,
        labels=labels,
        explode=(0.02,0,0),
        autopct='%1.1f%%',
        pctdistance=0.5,
        labeldistance=1.2,
        wedgeprops=wprops,
        textprops=tprops,
        radius=0.8,
        center=(0.5,0.5))
plt.show()
```



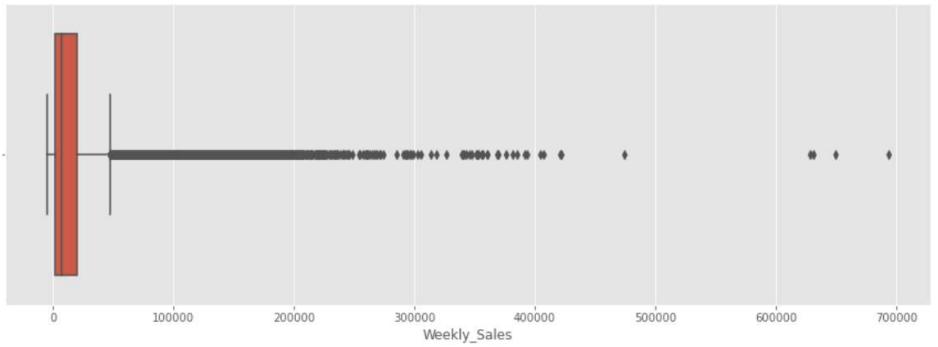
Conocida la distribución de las tiendas, los grupos y el tamaño de estas, se procedió a realizar un análisis de los factores que pueden influir en las ventas de Walmart, como primer factor se contempló la afectación que obtienen las ventas cuando un día es festivo o no. Se logro conocer que los días festivos si tienen un impacto positivo en las ventas de Walmart, pero este no es un impacto gran impacto ya que solo afecta o modifica las ventas en un promedio del 7%

```
In [15]:  data = pd.concat([train['Store'], train['Weekly_Sales'], train['IsHoliday']], axis=1)
          f, ax = plt.subplots(figsize=(20, 10))
          fig = sns.boxplot(x='Store', y='Weekly_Sales', data=data, showfliers=False, hue="IsHoliday")
```



Se procedió a identificar sobre los datos de train posibles outliers y se determinó que los mismos son equivalentes a un 8.4% de la data

```
In [18]:
plt.figure(figsize=(15,5))
sns.boxplot(x=train['Weekly_Sales'])
plt.show()
```

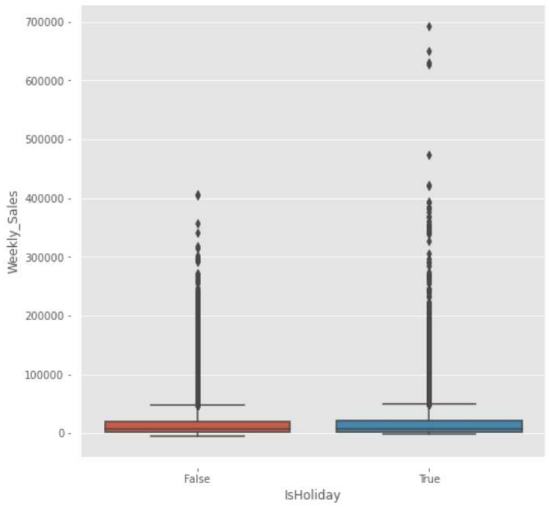



```
In [19]:
Q1 = train['Weekly_Sales'].quantile(0.25)
Q3 = train['Weekly_Sales'].quantile(0.75)
IQR = Q3 - Q1
Noutliers = (train['Weekly_Sales'] > (Q3 + 1.5 * IQR)).sum()
print(round((Noutliers/len(train))*100,1), '% de los datos corresponden a posibles outliers en la base')
```

```
8.4 % de los datos corresponden a posibles outliers en la base
```

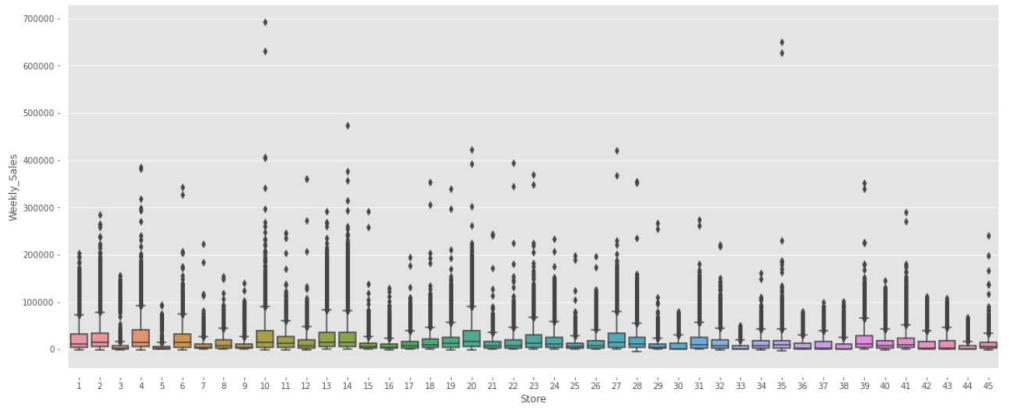
Se realizo este mismo análisis sobre los outliers para las ventas según si el día es festivo o no, logrando identificar que los días festivos presentan una mayor cantidad de posibles datos "outliers"

```
In [21]:
plt.figure(figsize=(8, 8))
sns.boxplot(x="IsHoliday",
            y="Weekly_Sales",
#             orient="h",
            data=data)
sns.despine(offset=10, trim=True)
```



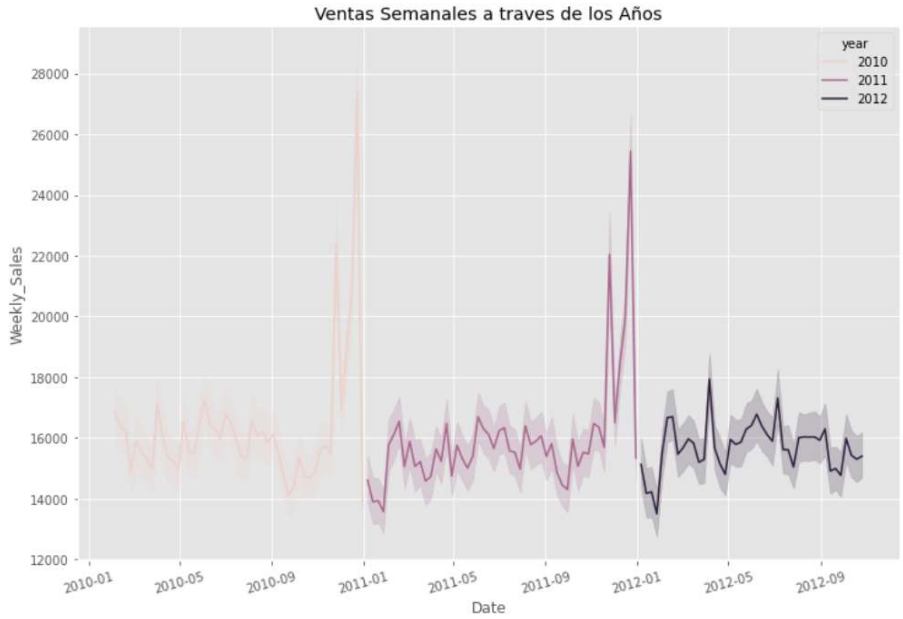
Analizado el factor tienda y los outliers en las ventas semanales sobre la data train y sobre el factor festivo o no se realiza un análisis por tiendas versus las ventas señales para determinar la distribución de estas y su impacto

```
In [22]:
plt.figure(figsize=(20,8))
sns.boxplot(x="Store",
            y="Weekly_Sales",
#             orient="h",
            data=data)
sns.despine(offset=10, trim=True)
```



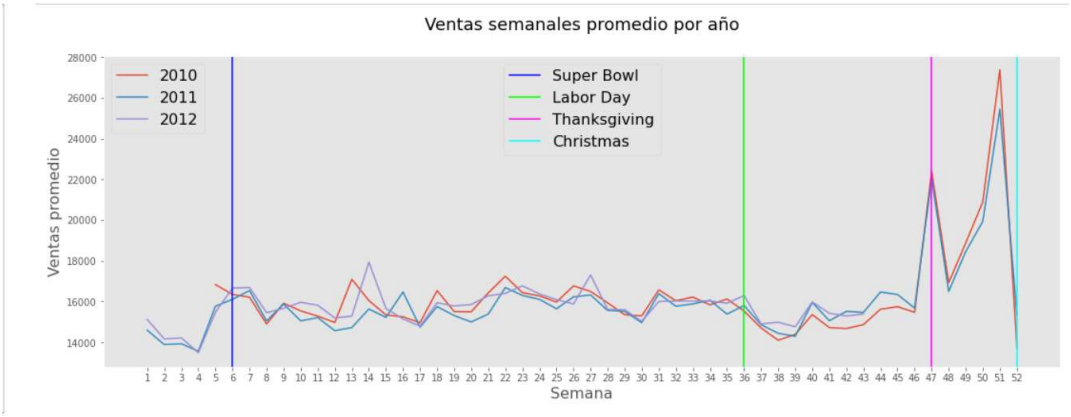
Se realiza un análisis de las ventas semanales a través de los años para poder identificar semanas con mayores impactos en las ventas, y con ello poder a futuro obtener una mejor distribución del personas y recursos. Se logra identificar por medio del siguiente grafico que Walmart posee grandes incrementos en las ventas durante las ultimas semanas de cada año y aparenta ser un comportamiento constante

```
In [27]:
plt.figure(figsize=(12, 8))
sns.lineplot(x="Date", hue="year", y="Weekly_Sales", data=dataTime)
plt.xticks(rotation=15)
plt.title('Ventas Semanales a traves de los Años')
plt.show()
```

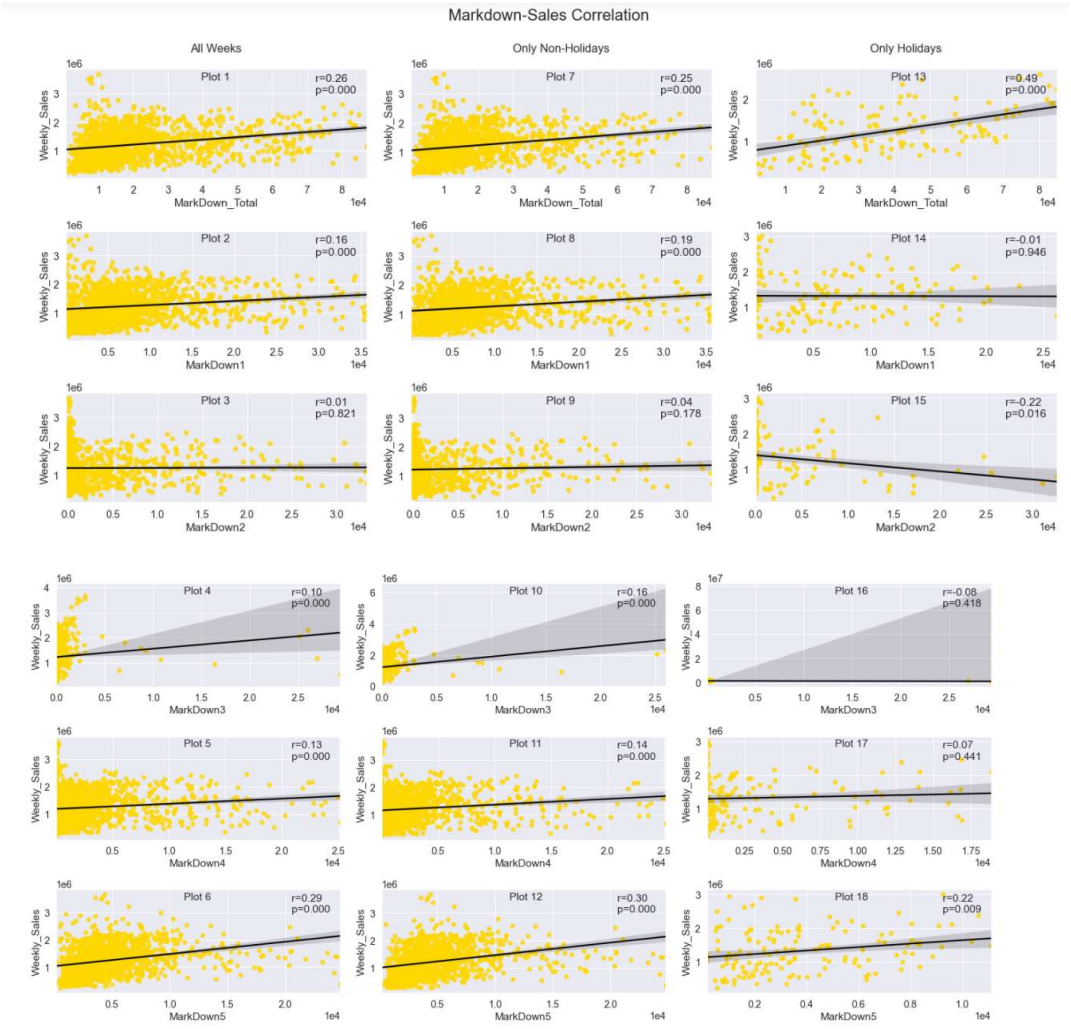


Al identificar que las ventas posee ciertos aumentos o picos además que si tiene una afectación por los días festivos se decide realizar una grafica que permita mostrar el impacto y conocer cual de los mismos es más influyente que el resto, permitiendo con ello brindar como resultado del caso un mayor conocimiento de sus ventas, cuando se pueden esperar crecimientos o afectaciones importantes por un día festivo y con ello un optimo manejo de los recursos e incluyo una distribución o manejo de los horarios de ventas. Con el siguiente grafico se logra determinar que los días festivos que mayor impacto causan en las ventas son el día de acción de gracias y navidad, sobre el resto de los festivos se puede observar que la afectación es mínima o no sale de las ventas promedio

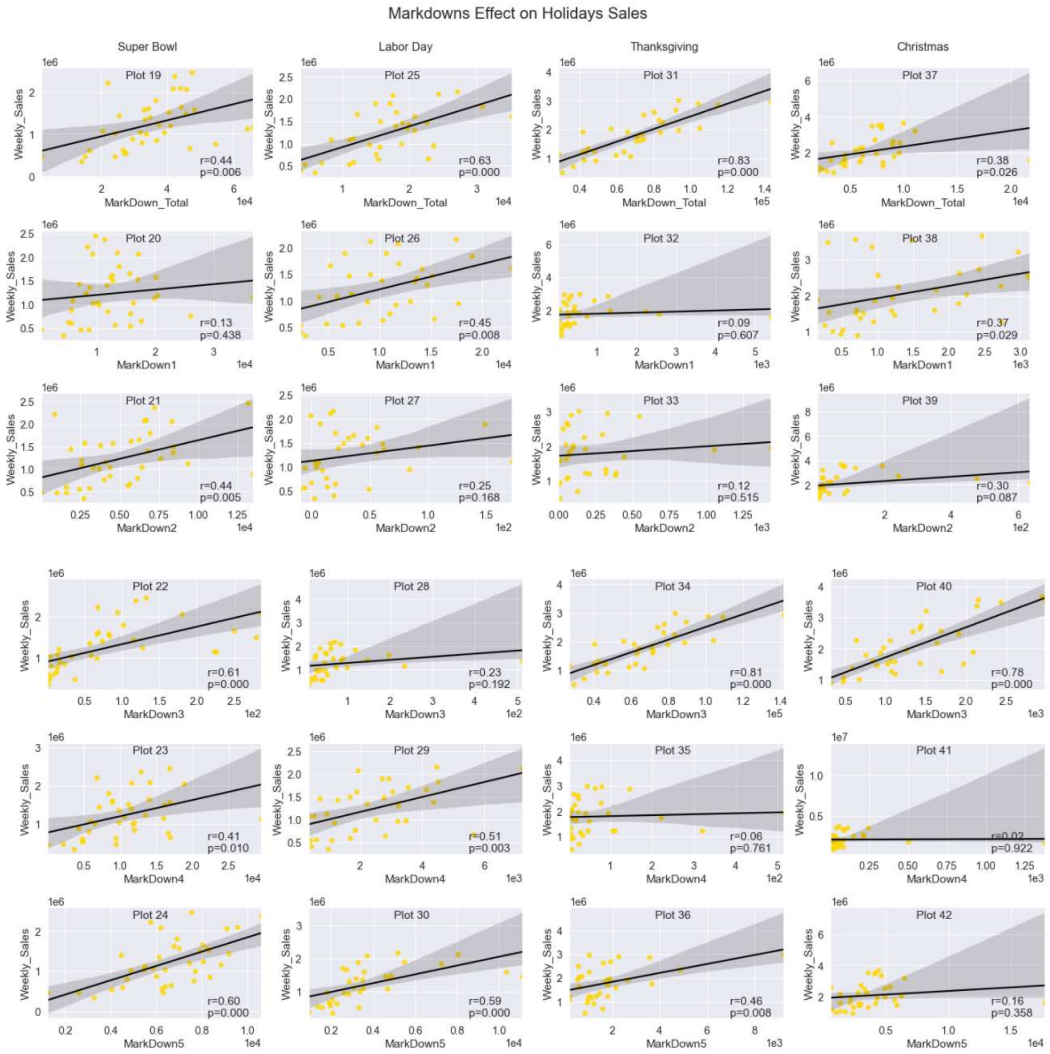
In [29]:
```
Weekly_Sales_2010 = train[train.Year==2010]['Weekly_Sales'].groupby(train['WeekNo']).mean()
Weekly_Sales_2011 = train[train.Year==2011]['Weekly_Sales'].groupby(train['WeekNo']).mean()
Weekly_Sales_2012 = train[train.Year==2012]['Weekly_Sales'].groupby(train['WeekNo']).mean()
plt.figure(figsize=(18,6))
sns.lineplot(x=Weekly_Sales_2010.index, y=Weekly_Sales_2010.values)
sns.lineplot(x=Weekly_Sales_2011.index, y=Weekly_Sales_2011.values)
sns.lineplot(x=Weekly_Sales_2012.index, y=Weekly_Sales_2012.values)
plt.grid()
plt.xticks(range(min(train['WeekNo']), max(train['WeekNo'])+1,1))
plt.legend(['2010', '2011', '2012'], loc='best', fontsize=16)
plt.gca().add_artist(plt.legend(['2010', '2011', '2012'], loc='upper left', fontsize=16))
plt.title('Ventas semanales promedio por año\n', fontsize=18)
plt.ylabel('Ventas promedio', fontsize=16)
plt.xlabel('Semana', fontsize=16)
i=0
colors = ['blue', 'lime', 'magenta', 'cyan']
for x in Holidays.iloc[:3,:].drop_duplicates().stack():
    plt.axvline(x, color = colors[i], label = Holidays.columns[i])
    plt.legend(loc='upper center',fontsize=16)
    i = i + 1
plt.show()
```



Posterior a ello se realiza un análisis de las correlaciones ventas semanales versus si es un día festivo o no



Y adicional se analiza la correlación segregando la información por sólo festivos y analizamos los 4 tipos de festividades correlacionándolo vs weekly sales



Una vez que se lograron analizar los diferentes indicadores, correlaciones, impactos de los factores y al tener un mayor conocimiento del comportamiento de las ventas para Walmart se procede a iniciar con la construcción del modelo de predicción, sobre el mismo se toma la decisión de utilizar le método o modelo Random Forest ya por el tipo de datos encontrados y las variables brindadas este modelo maneja de mejor forma los conjuntos de datos y a su vez se minimiza la posibilidad de un sobre ajuste que pueda entorpecer los resultados deseados.

Utilizando el método de eliminación de rasgos recursivos de modelo Random Forest se permite almacenar la precisión encontrada y a su vez determinar las variables más importantes o bien las que causan un mayor impacto en las ventas, entre las cuales resaltan departamentos, tamaños, tiendas, números de semana entre otros como se puede apreciar en la siguiente tabla

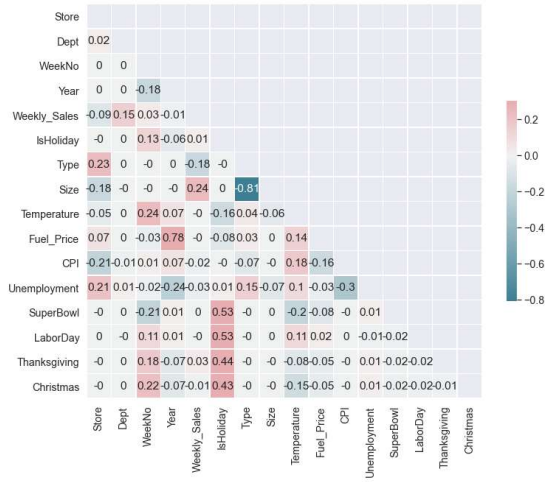
In [39]:
```
from sklearn.ensemble import RandomForestRegressor
from sklearn.feature_selection import RFE

trees = 5
rfe = RFE(RandomForestRegressor(n_estimators=trees), step=1).fit(train_data.loc[:, train_data.columns != 'Weekly_Sales'], tra
rfe.ranking_
rfe.support_

RF_imp = RandomForestRegressor(n_estimators=trees)
RF_imp.fit(train_data.loc[:, train_data.columns != 'Weekly_Sales'], train_data['Weekly_Sales'])
importance = RF_imp.feature_importances_
xy = pd.DataFrame(
        {'Variables':[(train_data.loc[:, train_data.columns != 'Weekly_Sales'].columns[x]) for x in range(len(importance))],
                      'Importance (%)':importance*100})
xy.insert(2,'RFE',rfe.support_)
feature_importance = xy.sort_values(by=['Importance (%)'], ascending=False)
feature_importance.style.format({'Importance (%)': '{:,.1f}'.format})
```

Out[39]:

	Variables	Importance (%)	RFE
1	Dept	63.2	True
6	Size	18.7	True
0	Store	5.6	True
2	WeekNo	3.0	True
9	CPI	2.7	True
13	Thanksgiving	1.9	True
5	Type	1.5	True
10	Unemployment	1.3	False
7	Temperature	1.2	False
8	Fuel_Price	0.5	False
14	Christmas	0.2	False
4	IsHoliday	0.1	False
3	Year	0.1	False
11	SuperBowl	0.0	False
12	LaborDay	0.0	False

Sobre estos indicadores y para poder conocer de forma más precisa cuales utilizar en el modelo de predicción se ejecutan nuevamente correlaciones, pero esta vez por medio de gráficos

In [41]:
```
# Computo de la matriz de correlación
corr = train_data.corr().round(decimals=2)

# Creación del gráfico de correlación
mask = np.triu(np.ones_like(corr, dtype=np.bool))
f, ax = plt.subplots(figsize=(12, 10))
cmap = sns.diverging_palette(220, 10, as_cmap=True)
sns.heatmap(corr, mask=mask, cmap=cmap, vmax=.3, center=0,
            square=True, linewidths=.5, cbar_kws={"shrink": .5}, annot=True)
```



Identificados los factores que pueden representar un mayor impacto sobre las ventas se decide ejecutar los modelos de Random Forest para predecir y conocer los porcentajes de precisión sobre cada uno de ellos, se decide ejecutar el modelo con distintas cantidades de variables para poder identificar su afectación y a la vez cual de los mismos puede significar una mayor precisión al desarrollo. Para ello de ejecutan el modelo contemplando todas las variables, luego se procede a ejecutar con 8 variables y por último con 4 de las variables obteniendo como resultado que el modelo brindar mayor precisión al utilizar o contemplar únicamente 4 varíales con un resultado de 97,4% versus el modelo que tenía todas las variables que alcanzaba un total d 96,8 % y el modelo de 8 variables que alcanzaba 97,2%

In [42]:
```
from sklearn.model_selection import train_test_split
from sklearn import metrics

def Model(train_data):
    Train, Test = train_test_split(train_data, random_state=1)

    X_Train = Train.sort_index().drop(columns=['Weekly_Sales'])
    Y_Train = Train['Weekly_Sales'].sort_index()

    X_Test = Test.sort_index().drop(columns=['Weekly_Sales'])
    Y_Test = Test['Weekly_Sales'].sort_index()

    # Regresion de Random Forest
    print('Calculando')
    trees = 5
    RF = RandomForestRegressor(n_estimators=trees)
    RF.fit(X_Train, Y_Train)
    SalesPrediction = RF.predict(X_Test)
    RF_accuracy = RF.score(X_Test, Y_Test)
    RF_accuracy = metrics.r2_score(Y_Test, SalesPrediction)
    print("Model Accuracy:", round(RF_accuracy*100,1),"%")

    # Evaluación WMAE
    soma_SalesPred = 0
    w = np.zeros(len(X_Test.index))
    for l in range(len(X_Test.index)):
        if X_Test['IsHoliday'].iloc[l] == 0:
            w[l] = 1
        else: w[l] = 5
        soma_SalesPred += w[l]*abs(Y_Test.iloc[l] - SalesPrediction[l])
    WMAE = round(soma_SalesPred/np.sum(w),5)
    print('WMAE =',round(WMAE,2))
```

In [43]:
```
print('Se consideran todas las variables:')
Model(train_data)
```

```
Se consideran todas las variables:
Calculando
Model Accuracy: 96.8 %
WMAE = 1786.7
```

In [44]:
```
print('Se consideran solamente las 8 variables más relevantes:')
train_data_filtered8 = train_data.drop(columns=['Unemployment','Fuel_Price','Christmas','LaborDay','SuperBowl'])
Model(train_data_filtered8)
```

```
Se consideran solamente las 8 variables más relevantes:
Calculando
Model Accuracy: 97.2 %
WMAE = 1743.95
```

In [46]:
```
print('Se consideran solamente las 4 variables más relevantes:')
train_data_filtered4 = train_data.loc[:,['Store','Dept','Size','WeekNo','IsHoliday','Weekly_Sales','Year']]
Model(train_data_filtered4)
```

```
Se consideran solamente las 4 variables más relevantes:
Calculando
Model Accuracy: 97.4 %
WMAE = 1659.08
```

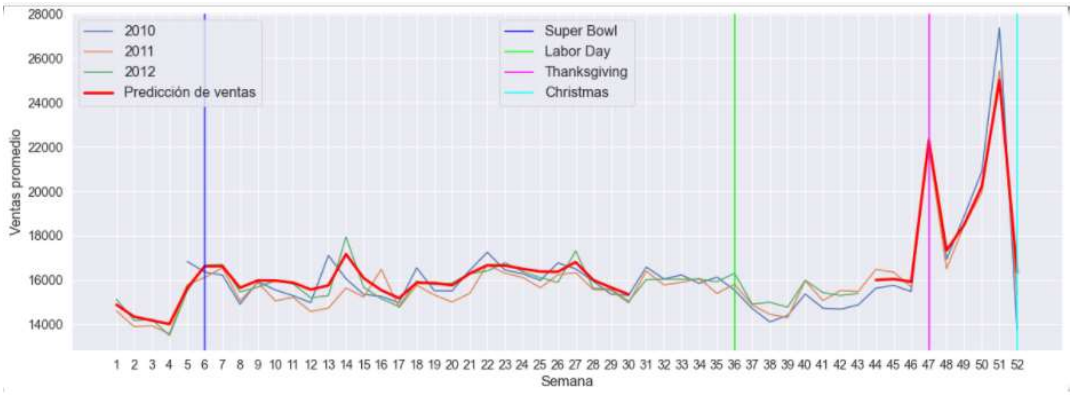
Una vez que se determina el porcentaje de predicción para los modelos ejecutados y donde se logra identificar que el modelo de mayor precisión es el que utiliza únicamente 4 variables se decide hacer mas robusto el mismo con la finalidad de encontrar el WMAE con la herramienta GridSearchCV.

```
In [48]: from sklearn.model_selection import GridSearchCV

In [49]: def Optmized_Model(train_data):
             Train, Test = train_test_split(train_data, random_state=1)

             X_Train = Train.sort_index().drop(columns=['Weekly_Sales'])
             Y_Train = Train['Weekly_Sales'].sort_index()

             X_Test = Test.sort_index().drop(columns=['Weekly_Sales'])
             Y_Test = Test['Weekly_Sales'].sort_index()

             # Regresion Random Forest
             print('Calculando')
             RF = RandomForestRegressor(n_estimators=70, min_samples_split=4)
             RF.fit(X_Train, Y_Train)
             SalesPrediction = RF.predict(X_Test)
             RF_accuracy = RF.score(X_Test, Y_Test)
             RF_accuracy = metrics.r2_score(Y_Test, SalesPrediction)
             print("Model Accuracy:", round(RF_accuracy*100,1),"%")

             # Validación WMAE
             soma_SalesPred = 0
             w = np.zeros(len(X_Test.index))
             for l in range(len(X_Test.index)):
                 if X_Test['IsHoliday'].iloc[l] == 0:
                     w[l] = 1
                 else: w[l] = 5
                 soma_SalesPred += w[l]*abs(Y_Test.iloc[l] - SalesPrediction[l])
             WMAE = round(soma_SalesPred/np.sum(w),5)
             print('WMAE =',round(WMAE,2))
```

Optimizado el modelo se ejecuta nuevamente para determinar el nuevo porcentaje de confiabilidad y sobre el mismo se logra mejorar en un 0.3% obteniendo un nuevo porcentaje de confiabilidad del 97,7%

```
In [50]: Optmized_Model(train_data_filtered4)

         Calculando
         Model Accuracy: 97.7 %
         WMAE = 1534.23
```

Para finalizar y al obtener un modelo con un buen porcentaje de precisión se decide ejecutar un grafico que permita mostrar como se comporta este modelo predictivo y a su vez se compara contra las ventas de los años pasados, logrando determinar que el modelo creado tiene un muy buen comportamiento, este se mueve y se comporta de forma regular aumentando con ello la certeza que el modelo obtiene o genera resultados precisos (se puede apreciar el comportamiento sobre la línea en color rojo)



Obtenido la comparativa de manera gráfica y determinando que el modelo presenta un buen comportamiento se desarrolló un código que permite extraer toda la información trabajada o de predicción en un archivo en formato de MS Excel (.csv), pues este puede ser parte del entregable final para la compañía.

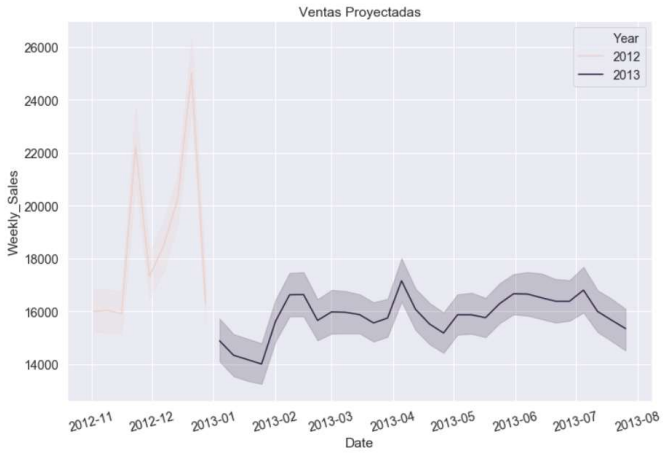
```
In [54]: Submission_File = pd.read_csv('sampleSubmission.csv')
         Submission_File['Weekly_Sales'] = Weekly_Sales_Test
         Submission_File.to_csv('Submission_File.csv',index=False)
         Submission_File.head()
         print('Se generó el Archivo Submission_File.csv con la predicción de ventas solicitada')

         Se generó el Archivo Submission_File.csv con la predicción de ventas solicitada
```

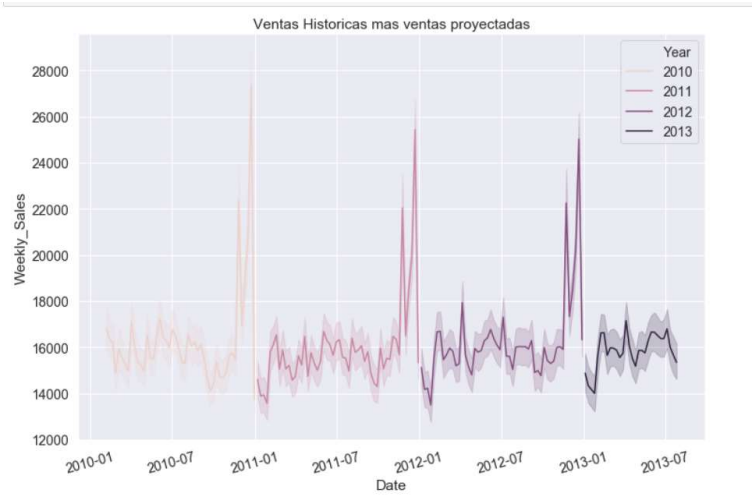
```
In [56]: final = pd.read_csv('Submission_File.csv')
         final.head()

Out[56]:
```

	Id	Weekly_Sales
0	1_1_2012-11-02	29984.097513
1	1_1_2012-11-09	19327.716111
2	1_1_2012-11-16	19266.940640
3	1_1_2012-11-23	20365.272114
4	1_1_2012-11-30	24569.241569

Adicional se genera la grafica sobre los datos proyectados y se realiza, la también, la unión de los datos en un solo grafico

```
In [93]: plt.figure(figsize=(12, 8))
         sns.lineplot(x="Date", hue="Year", y="Weekly_Sales", data=final)
         plt.xticks(rotation=15)
         plt.title('Ventas Proyectadas')
         plt.show()
```



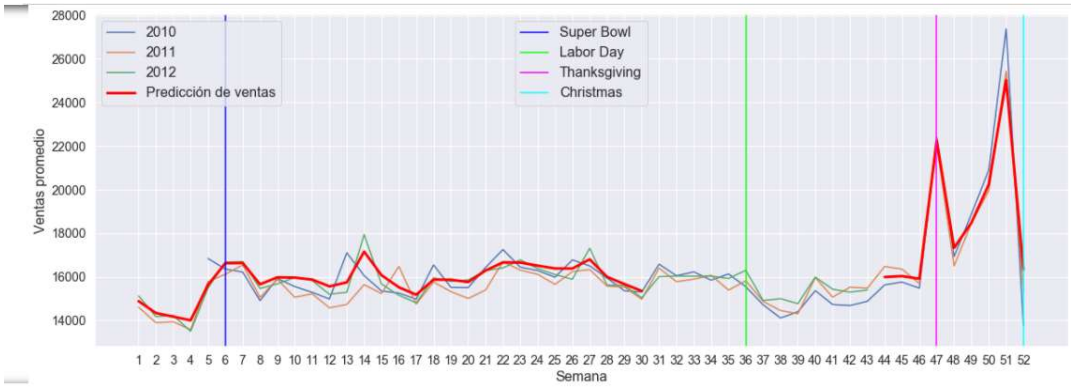


EVALUACIÓN

Para realizar la evaluación de la solución planteada primero se realizó un análisis de la precisión del modelo creado donde se obtuvieron porcentajes muy buenos con valores y entre ellos el porcentaje final de precisión con el modelo ya optimizado fue del 97,7% con un error absoluto medio ponderado (WMAE) de 1529.26 indicando con ello gran confiabilidad

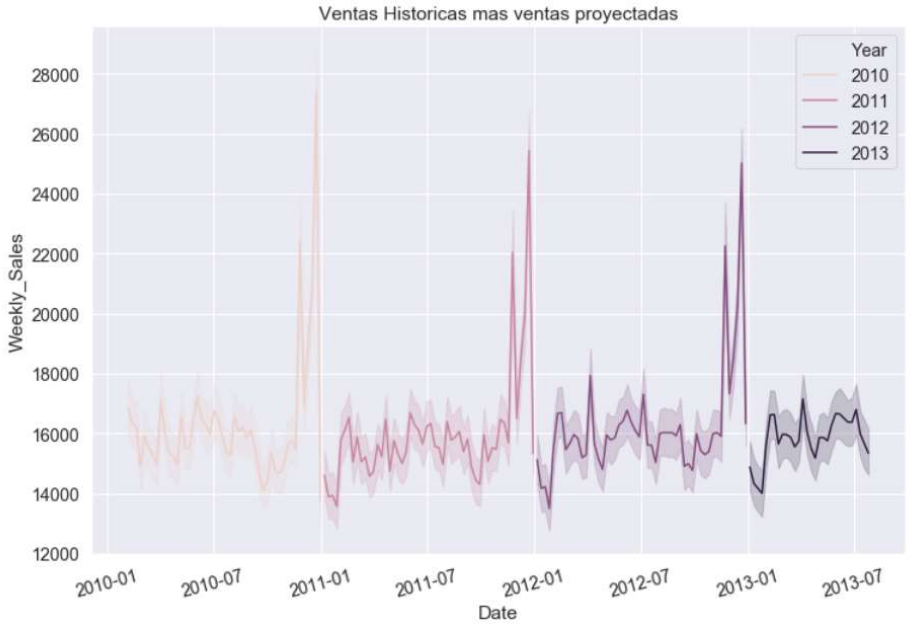
```
In [54]:  Optmized_Model(train_data_filtered4)

          Calculando
          Model Accuracy: 97.7 %
          WMAE = 1529.26
```

Adicional a la validación sobre los porcentajes del promedio absoluto ponderado del error se decidió crear también un código o algoritmo que nos permite gráficamente comparar los resultados del modelo de predicción creado contra los resultados o los valores de los años anteriores, esto nos permite de forma visual poder identificar como se comporta nuestro modelo, nos permite identificar si nuestro modelo puede contemplar las afectación de factores reduciendo o incrementando las ventas en fechas específicas, si las ventas se encuentran posiblemente ajustadas a la media o bien si se tienen un comportamiento similar al entorno real. Los resultados de este grafico fueron muy positivos ya que como se puede observar en la siguiente imagen la línea de color rojo o la línea que hace referencia a los datos de predicción se encuentran completamente acorde a la realidad, tiene la misma tendencia o comportamiento de las ventas de Walmart en los años anteriores, considera también los picos o alzas de las ventas sobre los días de acción de gracias y de navidad que si recordamos son los días festivos que tenían un mayor impacto.



Por último y como un método más para poder determinar los resultados obtenidos se realizo una grafica que muestra las ventas históricas, más los datos de proyección y sobre este mismo se logra visualizar que la tendencia o el comportamiento de cada año es similar, no

presentan fluctuaciones importantes o considerables de forma anual que pueden indicar que el modelo desarrollado presentara algún error u oportunidad para seguir desarrollando.

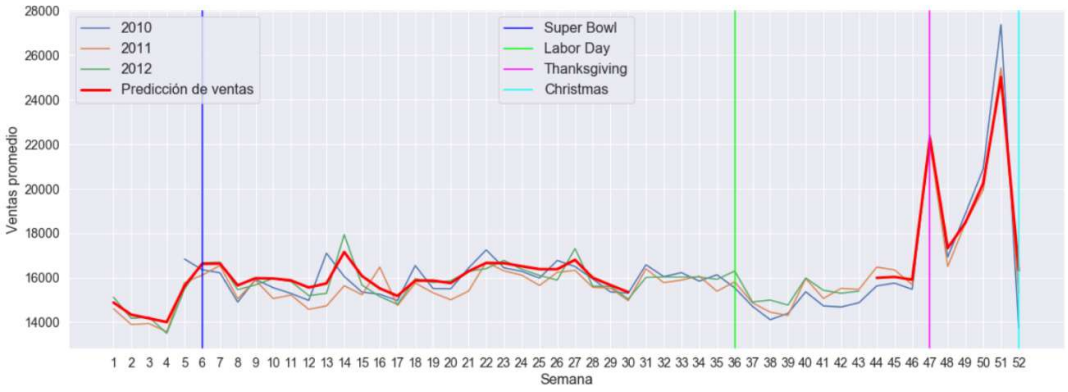


RESULTADOS

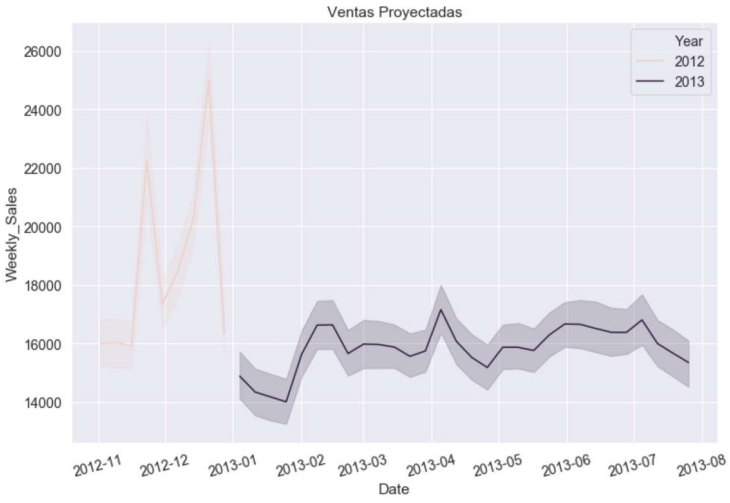
Los resultados obtenidos, con el modelo desarrollado, fueron muy positivos, ya que con el código elaborado y con la ejecución de los mismos se logran cargar los datos, depurarlos de una forma eficiente, también se logra graficar la información para observar tendencias o diferentes comportamientos, se logra identificar que factores externos pueden afectar las ventas y como estos pueden impactar en ellas; se logra, luego del análisis, seleccionar el modelo de machine learning “Random Forest”.

Sobre este modelo se ejecuta la predicción de las ventas y la afectación o impacto de cada una de sus variables, obteniendo porcentajes de precisión altos. Como método de evaluación de resultados se grafican los datos de predicción y se comparan con las ventas promedio de años anteriores por semana y la afectación de los diferentes días festivos, se logra evidenciar que los datos obtenidos del modelo desarrollado tienen un comportamiento

constante y lógico como se puede apreciar en la siguiente imagen ya que la recta en color rojo presenta sus aumentos y bajas de ventas acorde a los datos recolectados

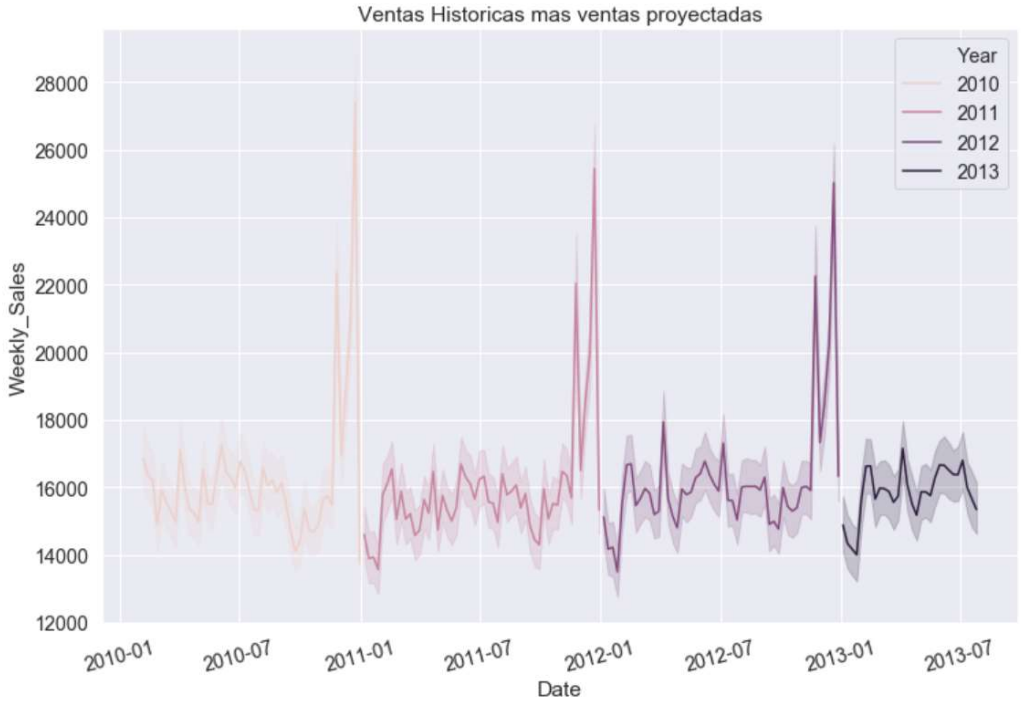


Se realizo también un gráfico específicamente sobre las ventas proyectadas durante 2012 y 2013 el cual permite determinar, también, un comportamiento lógico, ventas promedio muy similares y aumentos durante los últimos meses del año relacionados a el día de acción de gracias y navidad.



Por ultimo se creo un nuevo grafico donde se reúne toda la información, datos históricos y datos proyectados logrando aun más determinar que el comportamiento del modelo es bueno ya que los picos de ventas y la disminución hacen sentido con la data viendo todo esto desde un

solo grafico además que el mismo modelo como tal ya indica que el porcentaje de predicción es de un 97,7%



CONCLUSIONES Y TRABAJOS FUTUROS

Como conclusión al trabajo desarrollado, se obtiene que el modelo de Random Forest aplicado a los datos logra crear un método de predicción de datos con un alto porcentaje de confiabilidad además se logró, por medio del código desarrollado, la extracción o creación de un archivo de Excel final (.csv) que contiene el resultado de los datos de la proyección de ventas, el modelo fue desarrollado dentro de Python ya que es el lenguaje que durante el curso se logró obtener mayor conocimiento y el que se acopla mejor para el desarrollo del mismo, sobre este modelo se analizaron las diferentes variables que pueden impactar las ventas en donde se identificó que los factores que mayor afectación tienen son los departamentos, el tamaño, las tiendas, los números de las semanas, el índice de precios al consumidor y los días festivos.

Sobre el método de Random forest implementado se aplico un modelo de optimización que logro mejorar la predicción alcanzada llevando la misma a un porcentaje de precisión del 97,7%. Sobre los mismos y como modo de análisis para determinar la calidad del modelo se logró crear graficas que permiten mostrar que los comportamientos sobre los nuevos datos o los datos de predicción son muy similares al comportamiento de las ventas en los años anteriores, logrando evidenciar buenos resultados.

Sobre los datos analizados se logró también crear un código que permite realizar una depuración de forma eficiente de los datos recolectados y con ello, se logró determinar sobre los mismos que las ventas estaban compuestas por 45 tiendas, agrupadas en 3 tipos (A, B y C) y estos presentan diferentes tamaños entre las cuales encontramos las tiendas tipo A que representan un 48,9% de las ventas totales, las tiendas tipo B representan un 37,8% de las ventas y las tiendas tipo C representan el 13,3% restante. Según el análisis realizado a los diferentes factores que pueden tener un impacto en las ventas, se identificó que los días festivos si afectan y que la afectación de las mismas equivale a un 7% que impacta de manera positiva, lo anterior no es un gran impacto pero los días donde la afectación es de mayor importancia es sobre el día de acción de gracias y navidad por ende y por medio de estos indicadores se puede brindar a Walmart información para que realice un mejor control de su personal, en donde se puede indicar que días requieren de una mayor atención por un aumento de las ventas y con ello lograr una optimización de costos y recursos e incluso una modificación en los horarios que impacte positivamente las ventas sobre aquellos días con picos o aumentos de las mismas.

Para seguir mejorando la solución planteada, se sugiere continuar con la recolección de datos añadiendo más y mejores factores para el análisis; es decir aumentar la cantidad de variables que pueden impactar las ventas, se puede ampliar el análisis sobre la rotación de inventarios con datos de productos con mayores ventas, productos con menores ventas, productos con mayor rentabilidad, cantidad de productos vendidos por tipos y marca, por semana, por mes o año, también considerar el recorrido de los clientes dentro de la tienda conociendo que existen departamentos con importante participación con respecto al total de ventas por tienda, con lo anterior se puede realizar un análisis para obtener o llevar a cabo negociaciones estratégicas con los proveedores y realizar mejoras internas en la disposición de los productos por departamento (merchandising); se puede también recolectar información sobre las preferencias de compra de los usuarios para creación de promociones estratégicas dirigidas como cross selling (ventas cruzadas), ventas en pack u combos; se puede también recolectar información de la posible competencia que se pueda tener y la afectación que esta realiza a las ventas como forma de determinar y realizar ventajas estratégicas, en resumen a lo anterior y como parte del mejoramiento que se puede realizar sobre el modelo entregado es el enfoque en la recolección de datos ya que sobre los mismos y con los métodos de machine learning se pueden realizar análisis de forma rápida y precisa para generar nuevas conclusiones o recomendaciones.

REFERENCIAS

Analitica del retail. (25 de 02 de 2019). *Machine learning y retail: ¿cómo puede impulsar la estrategia de ventas?* Obtenido de http://analiticaderetail.com/machine-learning-y-retail/

Cespedes Urrutia, A. (21 de 11 de 2017). *Construccion de modelo de forecast para estimación de demanda en una empresa multinacion de retail.* Obtenido de https://repositorio.usm.cl/bitstream/handle/11673/41250/3560902038636UTFSM.pdf?sequence=1&isAllowed=y

Feliu, C. (s.f.). *4 relevant Big Data case studies in Logistics.* Obtenido de https://blog.datumize.com/4-relevant-big-data-case-studies-in-logistics

Kopanakis, J. (s.f.). *5 Real-World Examples of How Brands are Using Big Data Analytics.* Obtenido de https://www.mentionlytics.com/blog/5-real-world-examples-of-how-brands-are-using-big-data-analytics/

Navarro, J. (s.f.). *Clever Data.* Obtenido de https://cleverdata.io/big-data-retail/

Pallares Cabrera, F. (01 de 11 de 2014). *Bliblioteca UTB.* Obtenido de https://biblioteca.utb.edu.co/notas/tesis/0068209.pdf

Rawat, R. (s.f.). *Data Science at Netflix – A Must Read Case Study for Aspiring Data Scientists.* Obtenido de https://data-flair.training/blogs/data-science-at-netflix/

Relex. (s.f.). *La Guía Completa sobre Machine Learning en la Previsión de la Demanda en Retail.* Obtenido de https://www.relexsolutions.com/es/publicaciones/la-guia-completa-sobre-machine-learning-en-la-prevision-de-la-demanda-en-retail/

SeattleDataGuy. (02 de 28 de 2018). *7 Use Cases For Data Science And Predictive Analytics.* Obtenido de https://medium.com/coriers/7-use-cases-for-data-science-and-predictive-analytics-e3616e9331f9

Smart Business technologies. (s.f.). Obtenido de https://www.sb-tec.com/machine-learning-en-retail/

StoreCheck. (s.f.). *5 beneficios que aporta el machine learning al sector retail.* Obtenido de https://blog.storecheck.com.mx/5-beneficios-que-aporta-el-machine-learning-al-sector-retail

Talend. (s.f.). *¿En qué consiste un proceso de ETL (Extraer, Transformar y Cargar?* Obtenido de https://www.talend.com/es/resources/what-is-etl/#:~:text=White%20Papers-,%C2%BFEn%20qu%C3%A9%20consiste%20un%20proceso%20de%20ETL%20(Extraer%2C%20Transformar,centralizaci%C3%B3n%20en%20un%20%C3%BAnico%20repositorio.

Toro, E., Mejía, D., & Salazar, H. (01 de 01 de 2014). *Pronóstico de ventas usando redes neuronales.* Obtenido de

https://www.researchgate.net/publication/44131165_PRONOSTICO_DE_VENTAS_USANDO_REDES_NEURONALES

Uribe Mujica, J. I. (09 de 29 de 2019). *Predicción de la Demanda de un Nuevo Producto para una Empresa.* Obtenido de https://www.mti.cl/wp-content/uploads/2020/01/TesinaUribeMTI2017_27092019.pdf

Way to sucess. (s.f.). *Big Data y el análisis predictivo para aumentar ventas.* Obtenido de https://www.wtseo.co/big-data-y-el-analisis-predictivo-para-aumentar-las-ventas/