COMPORTAMIENTO VENTAS JUGUETES

MÁSTER DATA SCIENCE (4º EDICIÓN KSCHOOL)

ÁNGEL LUIS BARCO BONILLA

1. INTRODUCCIÓN

A continuación se expone una memoria del proyecto realizado indicando el objetivo del mismo. Además, se descubre la procedencia de los datos y el tratamiento necesario para su interpretación.

En esta memoria también se presentan las pautas a seguir para la comprensión del proyecto, el porqué de la metodología escogida para su desarrollo y las conclusiones obtenidas.

2. OBJETIVO

El objetivo del proyecto es analizar el comportamiento de las ventas generadas por la campaña principal de Google Adwords de una empresa de juguetes. Para ello, utilizaremos distintas herramientas y técnicas aprendidas durante el máster. El tiempo es un factor muy importante en el comportamiento de las ventas por lo que el grueso del proyecto se basará en el estudio de las series temporales.

Una vez que se consiga la interpretación y la comprensión correcta de los datos de estudio, la ampliación de este proyecto consistirá en realizar una predicción consistente para el número de ventas futuras y por consiguiente predecir las pujas óptimas que habría que realizar en las distintas segmentaciones de la campaña de Google Adwords. De esta manera, conseguiremos maximizar los beneficios de la empresa minimizando los costes.

3. PROCEDENCIA Y EXPLICACIÓN DE LOS DATOS

Los datos principales han sido facilitados por una empresa de juguetes y se dividen en 3 bases de datos en formato .csv.

Los 3 ficheros tienen idéntica estructura. En la primera base de datos se encuentran las observaciones que se registraron en el año 2015, en la segunda están las observaciones de 2016 y en la tercera base de datos disponemos de las observaciones registradas en 2017 (hasta el mes de septiembre).

A continuación se detallan las dimensiones y métricas incluidas en las bases de datos.

Dimensiones

- **Día**: Fecha en el que se registran las observaciones.
- **Dispositivo** desde el que se producen la observaciones (ordenador, Tablet o móvil).
- **Mes.**
- **Día de la semana.**
- **Región**: Comunidad Autónoma
- **Campaña:** Nombre de la campaña en estudio (Shopping – General).

Métricas

- **Impresiones:** nº de veces que se muestra el anuncio.
- **Clics:** nº de veces que se ha pinchado en el anuncio.
- **Conversiones:** nº de compras que se han realizado.
- **Coste**
- **CTR:** Clics/Impresiones
- **Posic. Media:** Posición media del anuncio (para Shopping siempre es posición 1)
- **CPC medio:** Coste/Clics
- **CPM medio:** Coste por cada 1.000 impresiones.
- **Coste/conv:** Coste/Conversiones
- **Porcentaje de conv:** Conversiones/Clics
- **Valor conv./coste:** Valor de las Convresiones / Coste
- **Valor conv. Total:** Valor de todas las Conversiones.
- **Valor/conv:** Valor de las Conversiones / Conversiones

Viendo las variables, la relación que existe entre las mismas y que vamos a modelar los datos aplicando las técnicas relacionadas con las series temporales, podemos adelantar que la mayoría de estas variables no van a aportar información a nuestro modelo o que la información que van a aportar será puramente descriptiva.

Además de los datos facilitados por la empresa, se ha utilizado una base de datos procedente de la siguiente página web: https://www.arcgis.com/features/index.html

En dicha base de datos figuran los Centroides de las coordenadas geográficas de las Comunidades Autónomas españolas. Nos será de ayuda para la parte de visualización con Tableau.

4. DEPURACIÓN DE LOS DATOS

La depuración de los datos se ha realizado en el notebook “1. Tratamiento_datos”. Algunas de las transformaciones que se realizaron en dicho notebook fueron las siguientes:

- El primer problema al que nos enfrentamos fue al de la lectura de los datos. Algunas de las features numéricas eran interpretadas como string debido al formato inicial de los datos. Para solucionarlo tuvimos que incluir parámetros en la lectura de los datos y crear algunas funciones que transformaran los datos de manera correcta.
- Selección de features interesantes para nuestro estudio.
- Transformación columna Dia para ponerla en formato fecha.
- Agrupar por fecha para obtener un único registro por día. (pd.groupby)
- Reindexar para estudiar la existencia de fechas sin datos.
- Interpolar las observaciones de las fechas sin datos.
- Exploración de los datos con técnicas estudiadas durante el máster (
- Creación de features interesantes para el modelo (“ lagged” features, día de la semana…).
- División de los datos en datasets de entrenamiento y prueba (train y test).

5. INFORMACIÓN ÚTIL PARA EL USUARIO

5.1. ¿POR QUÉ EL USO DE SERIES TEMPORALES?

Una serie temporal es una secuencia de datos, observaciones o valores, medidos en determinados momentos y ordenados cronológicamente.

Las series temporales son un tanto especiales y hay que tratarlas de una manera diferente a los problemas de regresión o análisis multivariante por varios motivos:

- Son dependientes del tiempo por lo que no se cumple el supuesto básico de los modelos de regresión de que las observaciones son independientes.
- Es muy probable que tengan una tendencia y/o estacionalidad cada cierto periodo de tiempo.
- Suelen estar autocorreladas. Esta correlación entre observaciones consecutivas provoca que la mayoría de los métodos estadísticos estándares basados en la suposición de observaciones independientes pueden arrojar resultados inútiles o incluso ser engañosos.

5.2. SERIES TEMPORALES Y PYTHON

Las principales librerías de Python que nos permiten trabajar con series temporales son las siguientes:

- **Pandas:** Nos ofrece las herramientas necesarias para realizar de manera sencilla las operaciones imprescindibles cuando trabajamos con series temporales como convertir fechas, estandarizar el tiempo, identificar datos faltantes, desplazar fechas.
- **Statmodels:** Contiene funciones y objetos vitales para el análisis de series temporales como el modelo autoregresivo (AR), modelo autoregresivo de media móvil (ARMA), autocorrelación…

5.3 CONTENIDO DE LOS NOTEBOOKS

En el segundo notebook (“2. Regresión_Serie”) usamos las librería sklearn para probar distintos modelos que se ajusten a nuestros datos de entrenamiento (train.csv).

En este notebooks se crea un modelo predictivo “Pipeline”. Para poder aplicar este modelo es necesario crear una función que asigne un valor único a cada característica de una variable categórica (One-hot encoding) . Eliminaremos las features que no nos aporten información mediante la validación cruzada (RFECV).

En el tercer notebook (“3. Modelos_TSeries”) construiremos distintos modelos de series temporales incluidos en la librería statsmodels, estimaremos su error a través del Root Mean Square Error (RMSE) y elegiremos el modelo que mejor se aproxime a nuestros datos.

Entre los modelos de series temporales que construiremos están Mean Constant Model, Linear Trend Model, Random Walk Model, Modelo de suavizado exponencial simple (SES).

Por otra parte, haremos una descomposición de series temporales y trataremos de eliminar la tendencia y la estacionalidad de la serie. También se intenta encontrar sin éxito un modelo ARIMA que explique la evolución de las conversiones.

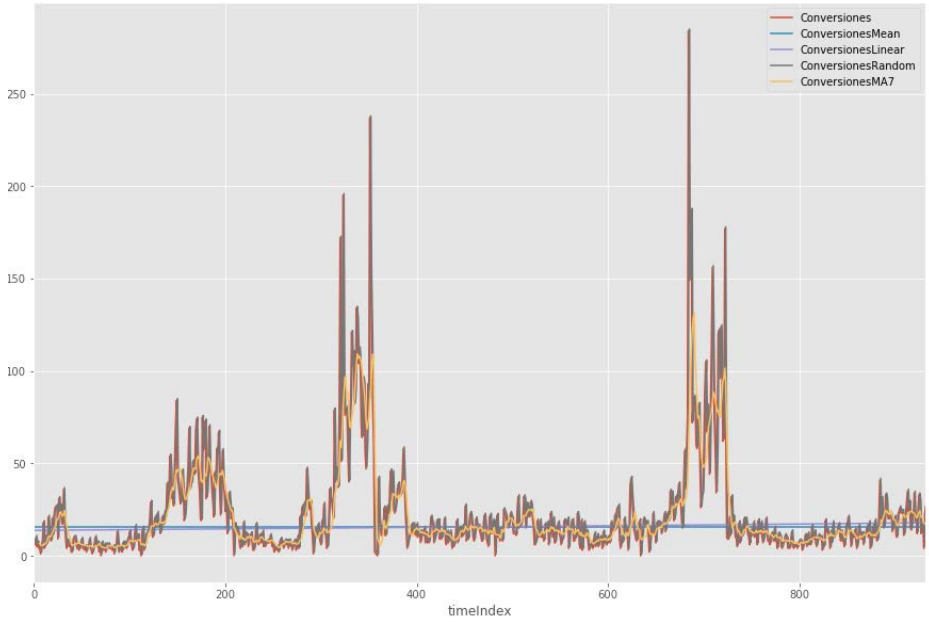
En el interior de los notebooks se especifican los pasos a seguir en cada momento.

6. CONCLUSIONES

Tras construir distintos modelos y evaluar su Root Mean Square Error (RMSE) podemos concluir que el modelo que mejor se ajusta a los datos durante el periodo analizado es el Modelo de media móvil de orden 7.

	Model	Forecast	RMSE
0	Mean	15.6749	28.6295
1	Linear	14.1089	28.6322
2	Random	24	17.6696
3	Moving Average 7	16.9441	16.0638
4	Exp Smoothing 7	20.1684	18.1892

En el siguiente gráfico se observa la aproximación de los modelos construidos con el objetivo de predecir los datos originales. La línea roja corresponde a las conversiones reales y en el gráfico se representan las estimaciones de los diferentes modelos, desde el Mean Constant Model y Linear Trend Model con malos resultados (líneas azul y morada) hasta el Random Walk Model o el Modelo de Media Móvil (líneas verde y amarilla).



Queda pendiente la ampliación de este proyecto que consistirá en ajustar los parámetros de los modelos encontrados para realizar una mejor predicción futura, así como la búsqueda de nuevos modelos.

7. BIBLIOGRAFÍA

Python: https://www.python.org/

Pandas: http://pandas.pydata.org/

Sklearn: http://scikit-learn.org/stable/

Statsmodels: http://www.statsmodels.org/stable/index.html

Matplotlib: https://matplotlib.org/index.html

Tableau: https://www.tableau.com/es-es/support/help

https://relopezbriega.github.io/

http://pythonforengineers.com/

https://www.analyticsvidhya.com/blog/2016/02/time-series-forecasting-codes-python/

https://github.com/silicon-valley-data-science/pydata-sf-2016-arima-tutorial