ENTREGABLE FINAL
H&M Business Intelligence & Data Science

Escuela Técnica Superior de
Ingeniería Informática

Miembros del grupo

- Datos del grupo:
  - Identificador: **H&M BI. NLP, visualización y predicción**
  - Título: **Business Intelligence y Data Science para la empresa de ropa H&M**
- Miembros del grupo:
  - **Pablo Reina Jiménez** Datos de contacto: pabreijim1, pabreijim1@alum.us.es
  - **María Lourdes Linares Barrera** Datos de contacto: marlinbar, marlinbar@alum.us.es

ÍNDICE DE CONTENIDOS

1. HITO ANTERIOR: INFORME INTERMEDIO

En esta sección, hablaremos sobre el estado del proyecto hasta el hito anterior: el entregable intermedio. Presentamos a continuación los principales puntos que se abordaban en este informe.

- **Motivaciones** para la elección del proyecto.
  Entre estas motivaciones destacan:
  i. Disponibilidad de datos corporativos reales.
  ii. Variedad de técnicas.
  iii. Capacidad de extender el trabajo desarrollado a cualquier otro caso de uso dentro del campo del Business Intelligence.

- **Especificación de objetivos**
  En esta sección del informe intermedio se detallaban los objetivos y requisitos planteados en líneas generales en la presentación inicial del proyecto. Por concretar un poco más:
  i. División del proyecto en 3 tareas: análisis de opinión pública, dashboards de visualización y predicción del éxito de las campañas de marketing.
  ii. Fuente de datos de cada una de estas tareas y su modo de extracción (en el caso de web scraping). Se planteaba el preprocesamiento y análisis exploratorio a realizar.
  iii. Descripción sobre las técnicas a utilizar. Por ejemplo: NLP a través de aprendizaje por transferencia para predicción de opinión pública, tipos de informes de visualización (evolución temporal, ventas por departamento de producto, ventas según el perfil del cliente…) y modelos de regresión (regresión lineal y random forest) para predicción de las ventas de las campañas marketing.
  iv. Tecnologías o librerías.

- **Reajuste de objetivos**
  En esta sección se planteaban añadían algunos objetivos nuevos respecto a lo que en un comienzo se propuso en la presentación inicial. Entre ellos:
  i. Obtención de WordClouds en base a las predicciones de la opinión pública.
  ii. Incorporación de los resultados de opinión pública al dashboard sobre ventas que se iba a generar. Con ello, se consigue integrar todas las fuentes de decisión en un informe paginado conjunto y centralizado.
  iii. Nuevos objetivos en la predicción del éxito de las campañas de marketing, abordando la interpretabilidad de los modelos. Para regresión lineal se propuso obtener proyecciones 3-dimensionales del modelo y para random forest se propuso obtener la importancia de cada uno de los medios publicitarios en la decisión.

- **Plan de trabajo**
  Se comentaron las herramientas a utilizar para gestionar el proyecto, la planificación de las tareas y la prioridad que se iba a conceder a cada una de ellas.

- **Estado del proyecto**
  Este capítulo recogía el avance del proyecto a fecha de la realización del informe intermedio, incluyendo problemas encontrados, trabajo realizado y trabajo futuro.

2. ANÁLISIS DE OPINIÓN PÚBLICA: TRABAJO Y RESULTADOS

En esta fase del proyecto consiste en analizar la opinión que de la empresa en distintos medios de comunicación: páginas de reseñas, periódicos, foros, redes sociales... Para ello se aplicarán técnicas de Data Mining y procesamiento de lenguaje natural.

2.1. Extracción de los datos y preprocesado

El dataset a utilizar lo construimos los miembros del grupo utilizando web scraping.

Instancias supervisadas	
Finalidad	Necesitamos reseñas clasificadas (positivas o negativas) para entrenar los modelos.
Fuente de datos	Reseñas de Google Maps de distintas tiendas de H&M del mundo. [1] [2] [3] • Dos tiendas de Reino Unido (en Londres) • Dos tiendas de Estados Unidos (en Lakeland y Boston) • Cuatro en Canadá (dos en Toronto, una de Quebec y otra en Vancouver).
Librerías	Librerías de Python (pandas, sklearn, selenium, cleantext y Vader).
Workflow	1. Scraping utilizando selenium. 1.1. Definimos un driver y establecemos la conexión con la url. 1.2. Cargamos las noticias accediendo a la sección de reseñas. 1.3. Nos desplazamos hasta el final para cargar todas las noticias. 1.4. Ampliamos los detalles de la reseña. 1.5. Almacenamos las reseñas en un dataframe de pandas: fecha, nombre de usuario, texto de reseña, número de estrellas. Repetimos el proceso para cada urls y obtenemos un csv por cada tienda. 2. Preprocesamiento. 2.1. Eliminación de reseñas vacías. 2.2. Etiquetado: positivas (>=3 estrellas) y negativas (<3 estrellas). 2.3. Construir un dataframe conjunto para todas las tiendas solo con el texto de reseña y clasificación (características relevantes para el modelo de NLP). 2.4. Limpieza de reseñas. 2.4.1. Extracción de la traducción al inglés. 2.4.2. Eliminación de emojis utilizando la librería clean text. 2.5. Balanceo del dataset (para entrenar los modelos de NLP). 2.5.1. Filtrar por subjetividad utilizando la librería Vader. Debido al elevado número de reseñas conseguidas, optamos por balancear el dataset eliminando reseñas de la clase más numerosa hasta igualar ambas. Para equiparar las clases extraemos un subconjunto de la clase más numerosa utilizando un filtro de subjetividad. Utilizando la librería Vader obtenemos para cada reseña de la clase más numerosa el porcentaje de subjetividad, ordenamos las reseñas de la clase más numerosa de mayor a menor subjetividad y nos quedamos con las N primeras, donde N es el número de reseñas de la clase menos numerosa. De esta forma, tenemos pareadas las reseñas positivas y negativas y, además, hemos extraído el subconjunto de reseñas de mejor calidad.

Instancias supervisadas	
Finalidad	Recopilamos posts de distintos medios de los que desconocemos su "sentimiento" y le aplicamos los modelos entrenados para extraer nueva información.
Fuente de datos	• Twitter (tuits recientes que citen a H&M o su servicio técnico) • Facebook [4] • Quora [5] • Google News [6].
Librerías	Librerías de Python (pandas, sklearn, selenium, twint, cleantext, deep_translator, vader, text_blob)
Workflow para Twitter	1. Scraping de tweets utilizando twint 1.1. Definimos el usuario sobre el que queremos buscar tweets (H&M). 1.2. Establecemos el límite de tweets que queremos cargar. 1.3. Cargamos los tweets como un DataFrame de Pandas. 1.4. Mantenemos únicamente los tweets que mencionen a H&M y no los haya emitido la propia empresa. 2. Preprocesamiento. 2.1. Eliminación de posibles emoticonos que aparezcan en los tweets. 2.2. Eliminación de usuarios citados, hashtags y links. 2.3. Traducción al inglés de los tweets en otro idioma. 2.4. Eliminación de valores nulos. 2.5. Si es necesario, recorte del tweet a una longitud adecuada. Una vez terminado este proceso, obtenemos un datasets de tweets ya preprocesados en los que aparece mencionada la compañía H&M y no han sido escritos por ella.
Workflow para Google News, Facebook, Quora	1. Scraping utilizando selenium. En este caso seguimos un procedimiento muy similar al caso de las reseñas. 2. Preprocesamiento. El proceso es similar al anterior, pero presenta las siguientes diferencias: - Al ser instancias no supervisadas, no tenemos el número de estrellas así que no tenemos que etiquetarlas como positivas o negativas (es información que desconocemos hasta el momento). - No tenemos que llevar a cabo un balanceo, ya que con estas opiniones no se va a entrenar el modelo, si no para aplicárselas. - Las publicaciones en las redes sociales no se muestran traducidas directamente. Utilizaremos deep_translator para traducirlas al inglés. - En las noticias de Google News puede haber noticias objetivas, para las cuales extraer su sentimiento no es relevante. Para filtrar estas noticias, utilizaremos un filtro de subjetividad-polaridad utilizando Text Blob. El resto del procedimiento es prácticamente análogo.

Tras la extracción, limpieza y filtrado obtenemos una serie de datasets:

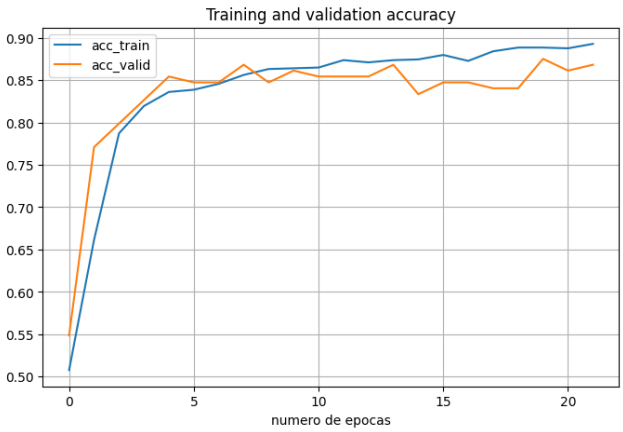
- El dataset google_reviews_balanced será utilizado para entrenar y evaluar los modelos.
- Los datasets de Twitter, Facebook, Google News y Quora serán aplicados al modelo ya entrenado para conocer el "sentimiento" de los usuarios de H&M.

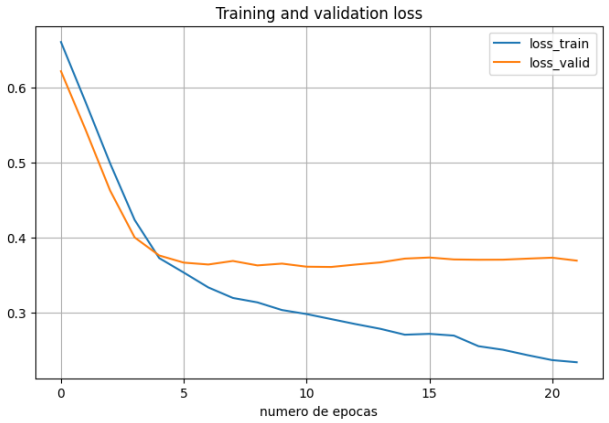
2.2. Modelos de aprendizaje: procesamiento de lenguaje natural

Con las instancias supervisadas se propone entrenar y evaluar (dar la eficiencia y eficacia) un modelo de procesamiento de lenguaje natural que permita discernir si una opinión es positiva o negativa (problema de clasificación). Se utilizarán las librerías: sklearn, Tensorflow y Pytorch.

2.2.1. Transfer leanring

Modelo 1: basado en Transfer learning	
Técnica	**Transfer learning** Tomar un modelo de procesamiento de lenguaje natural ya entrenado, añadirle una serie de capas y reentrenar solo esas capas.
Workflow	1. Dividimos el dataset en entrenamiento, validación y test a razón 80%-10%-10%. 2. Utilizar como arquitectura base el modelo nnlm-en-dim50 [7] disponible en el repositorio de modelos de Tensorflow Hub. 3. Concatenamos las siguientes capas al modelo: • Una capa densa de 64 neuronas con función de activación relu. • Una capa densa de 16 neuronas con función de activación relu. • Una neurona de salida con función de activación 4. Establecemos los parámetros de entrenamiento del modelo. • Entrenamos únicamente las capas finales añadidas. • Utilizamos un optimizador adaptativo: Adam. • Función de error: Entropía cruzada. • Métrica: accuracy (es una buena métrica al estar el dataset balanceado). • Introducimos EarlyStopping para evitar el sobreajuste.
Resultados	Eficiencia del 87%



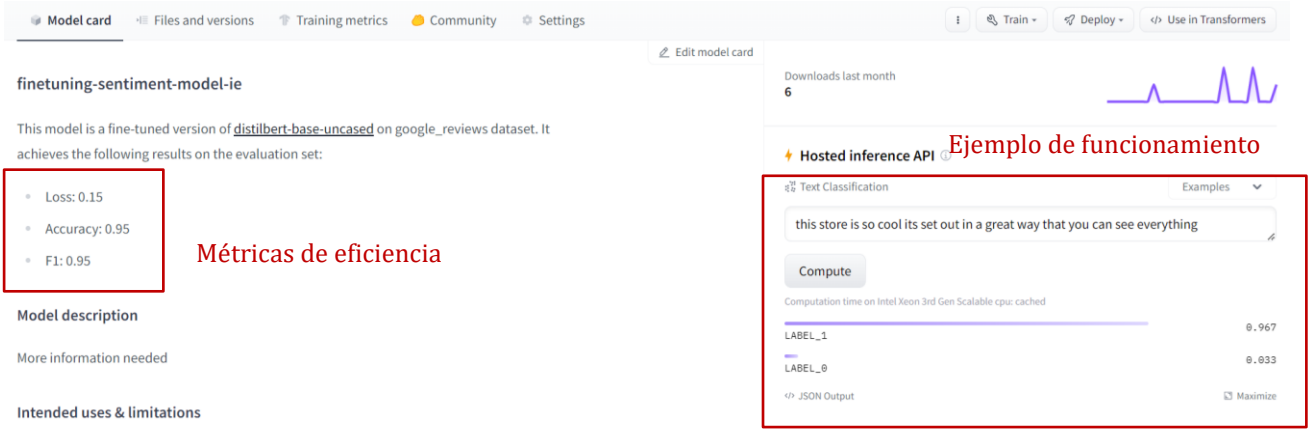



```
36/36 [==============================] - 0s 3ms/step - loss: 0.2551 - accuracy: 0.8839 - val_loss: 0.3703 - val_accuracy: 0.8403
Epoch 19/50
36/36 [==============================] - 0s 3ms/step - loss: 0.2503 - accuracy: 0.8883 - val_loss: 0.3704 - val_accuracy: 0.8403
Epoch 20/50
36/36 [==============================] - 0s 3ms/step - loss: 0.2432 - accuracy: 0.8883 - val_loss: 0.3718 - val_accuracy: 0.8750
Epoch 21/50
36/36 [==============================] - 0s 4ms/step - loss: 0.2366 - accuracy: 0.8874 - val_loss: 0.3730 - val_accuracy: 0.8611
Epoch 22/50
36/36 [==============================] - 0s 13ms/step - loss: 0.2337 - accuracy: 0.8927 - val_loss: 0.3691 - val_accuracy: 0.8681
5/5 - 0s - loss: 0.3230 - accuracy: 0.8750 - 30ms/epoch - 6ms/step
```

2.2.2. Fine Tuning

Modelo 2: basado en Fine Tuning	
Técnica	**Fine tuning** Tomar un modelo de procesamiento de lenguaje natural ya entrenado y reentrenar partiendo con los pesos iniciales del modelo anterior.
Workflow	1. Lectura del dataset de Google Reviews en forma de dataframes de pandas 2. Dividimos en train y test a razón 80%-20% 3. Transformamos los dataframes de pandas en estructuras de tipo datasets 4. Cargamos el modelo base: DistilBERT de Hugging Face en Pytorch [8] [9] 5. Definimos el tokenizador y el padding para adecuar y normalizar nuestros datos a la entrada del modelo. 5. Establecemos los parámetros de entrenamiento del modelo. • Utilizamos un optimizador adaptativo: Adam. • Función de error: Entropía cruzada. • Métrica: accuracy (es una buena métrica al estar el dataset balanceado) y f1. • Tasa de aprendizaje de 2e-5
Resultados	Eficiencia del 95%

Pueden consultarse las métricas de eficiencia y descargar el modelo en el siguiente enlace [10]. A través de dicho enlace, se puede probar el modelo de forma interactiva.



Dado que este modelo es el que mejores resultados proporciona, es el que hemos decidido utilizar finalmente para aplicarlo a la predicción del sentimiento de las publicaciones de redes sociales, periódicos y foros. Los resultados obtenidos son bastante precisos. El modelo proporciona la predicción junto con la probabilidad con la que ha tomado la decisión.

site	review	label	score
facebook	great selection and great service	1	0,97160404920578
facebook	how on earth do you get in touch with an actual store! the "customer service" line is far from customer service with an endless loop and voice	0	0,981667280197144
facebook	5 hours later and 2 email i still haven progressed. i also rang the 1800 number, pressed 3 to speak to a customer service officer. i was prompt	0	0,972457706928253
facebook	the dress i bought shrunk in a very unusual way and i have made 7 phone calls to date. i only getting message, '. . .busy call back later.' or i an	0	0,982921183109283
facebook	this store is so cool its set out in a great way that you can see everything	1	0,967416524887085
facebook	great prices and fun style	1	0,973173201084137
facebook	The clothes are well attended, and a lot of products of very good quality as well as good prices.	1	0,973054766654968
facebook	Terrible service. I'm coming out of a fracture and I can't stand for a long time! The supervisor Alex sent me to form in a great line! what a lack	0	0,979790568351746
facebook	I'm the one who buys shoes today, but you said... they didn't accept my debit card, but they already charged me. and I asked the bank that th	0	0,98026043176651
facebook	nice selection. great prices. found some good deals.	1	0,975844621658325
facebook	good service clean and tidy	1	0,970619261264801
facebook	all current in fashion, affordable prices, diversity of sizes and above all clothing ordered by size.	1	0,939377248287201

3. ANÁLISIS DE VENTA Y DASHBOARDS: TRABAJO Y RESULTADOS

Para esta segunda tarea nos centraremos en el estudio de las ventas de la empresa H&M a lo largo de 2-3 años, en función de sus productos, compradores y transacciones. De forma adicional, haremos uso de los resultados obtenidos anteriormente gracias al análisis de sentimientos.

3.1. Descripción de los datos

El dataset seleccionado proviene de la página web Kaggle [11] y está compuesto por tres tablas.

Tabla de datos	Descripción
Customers	Contiene información sobre los clientes de la empresa: customer_id, club_member_status, fashion_news_frequency, age y postal_code.
Articles	Contiene información sobre los artículos de la empresa: article_id, department_name. product_type_name. product_group_name. index_name, section_name… Entre otros datos, que no resultarán relevantes en nuestro estudio.
Transactions	Contiene información sobre las transacciones de la empresa y posee las siguientes columnas: article_id, costumer_id, t_dat, payment.

3.2. Preprocesamiento

Durante el proceso de carga de las distintas tablas en PowerBI, se cambiaron los tipos de las columnas que no eran los correctos. Una vez realizado este tratamiento, procedimos a la generación de nuevas variables que nos serían de utilidad para las posteriores visualizaciones.

- Grupo de edad: añadido en la tabla Customers. Toma distintos valores en función de la franja de edad en la que se encuentre el comprador: joven (16-30), adulto (30-60) y anciano (+60). Esta columna nos permitirá filtrar por los distintos grupos de edad y obtener una mejor compresión de las ventas en función de ello.
- Estación: añadido en la tabla Transactions. Representa la estación en la que se produjo la venta: invierno (si la venta se produjo entre diciembre y febrero), primavera (entre marzo y mayo), verano (entre junio y agosto) y otoño (entre septiembre y noviembre).

3.3. Dashboards y visualizaciones

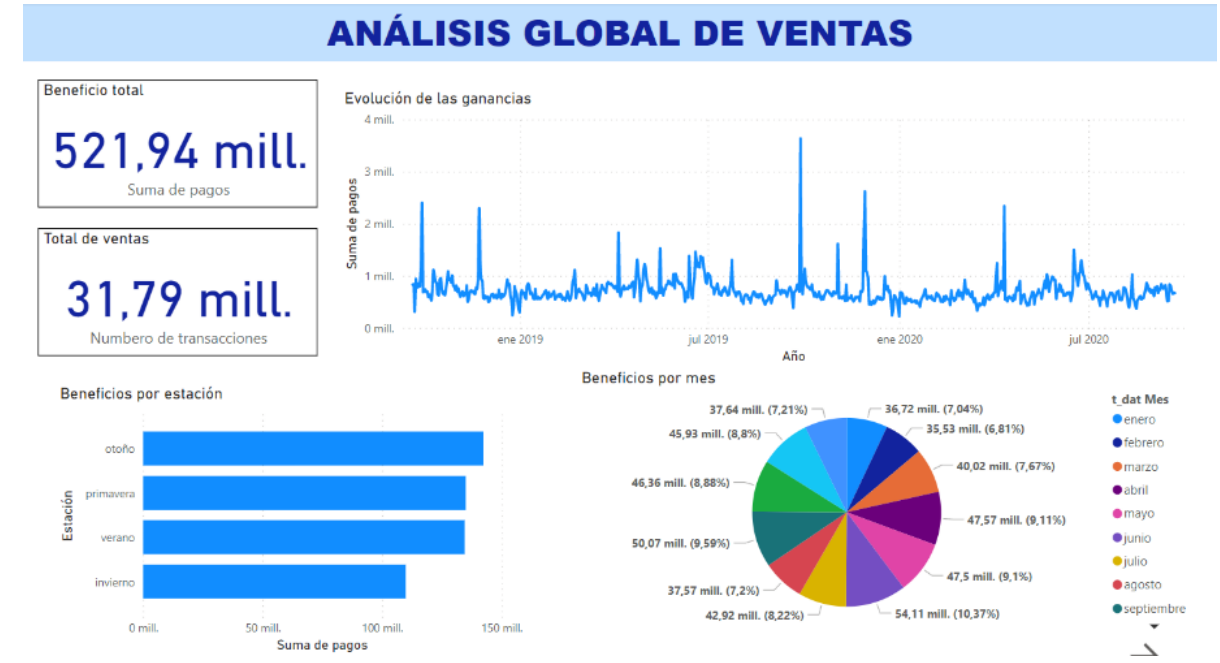
Una vez los datos ya procesados habían sido cargados en PowerBI, pasamos a generar las visualizaciones que nos permitirán analizar los distintos aspectos de la empresa. Decidimos organizar estas visualizaciones en las siguientes secciones: dashboards de evolución análisis de ventas, productos y clientes y dashboards de análisis de opinión pública.

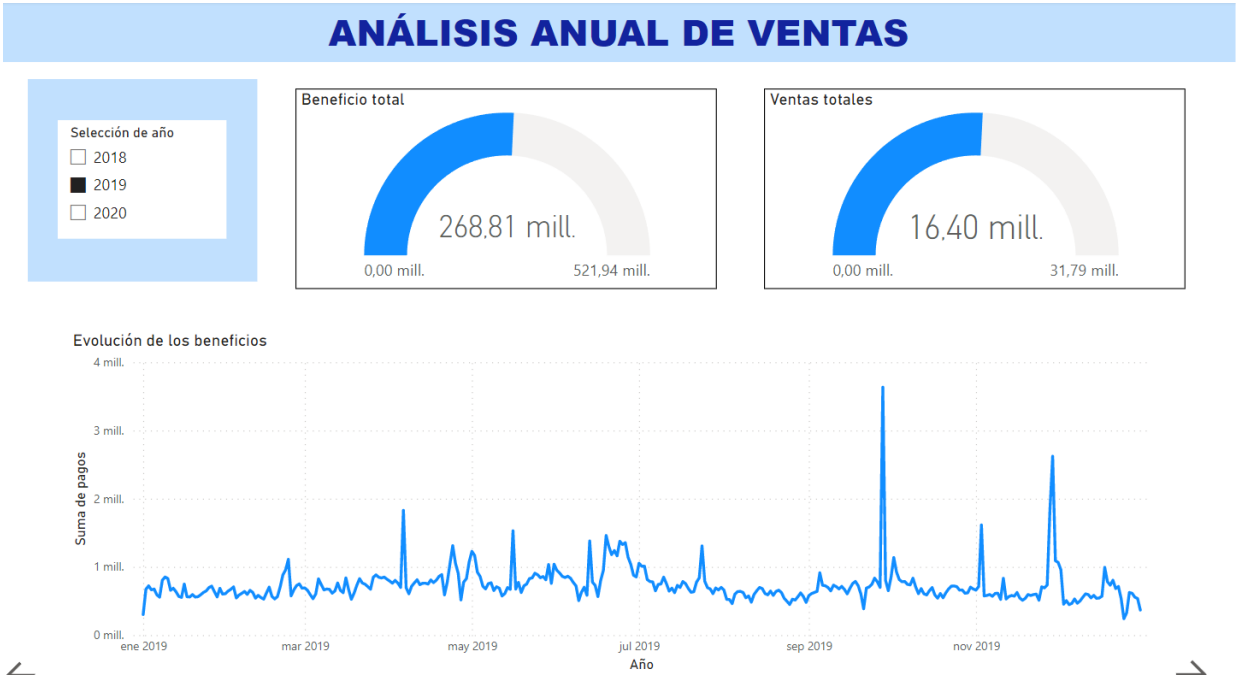
3.4. Sección 1: análisis de ventas

Análisis de ventas

En esta sección nos centraremos en el estudio de la tabla transacciones. Realizaremos diversas representaciones gráficas centradas en la evolución temporal de las ventas y estudiaremos la influencia de los factores estacionales (estación, año…). Podemos distinguir dos subsecciones:

- Un estudio general de las ventas y beneficios por estación y mes
- Un estudio anual de estas características





Análisis de productos

En este subapartado analizaremos la tabla de productos y estudiaremos cómo se distribuyen las ventas en función de ellos. Cuenta con una única página en la que representamos el total de productos y ventas en función de la estación. Adicionalmente, obtenemos representaciones del beneficio por cada tipo de producto, así como el número de unidades vendidas.

Análisis de clientes

Por último, nos centraremos en la tabla de clientes. En ella, estudiaremos como los diversos factores de grupo de edad, estado de la membresía o el interés por las noticias se relacionan con las ventas de la empresa. De nuevo, este apartado cuenta con una única página de visualizaciones, en las que se representa el número de consumidores, pudiendo filtrar por grupo de edad, membresía y la frecuencia de consulta de noticas de moda. Además, se añaden visualizaciones adicionales que representan las ventas y beneficios en función de diversos factores (estación o el grupo de edad).

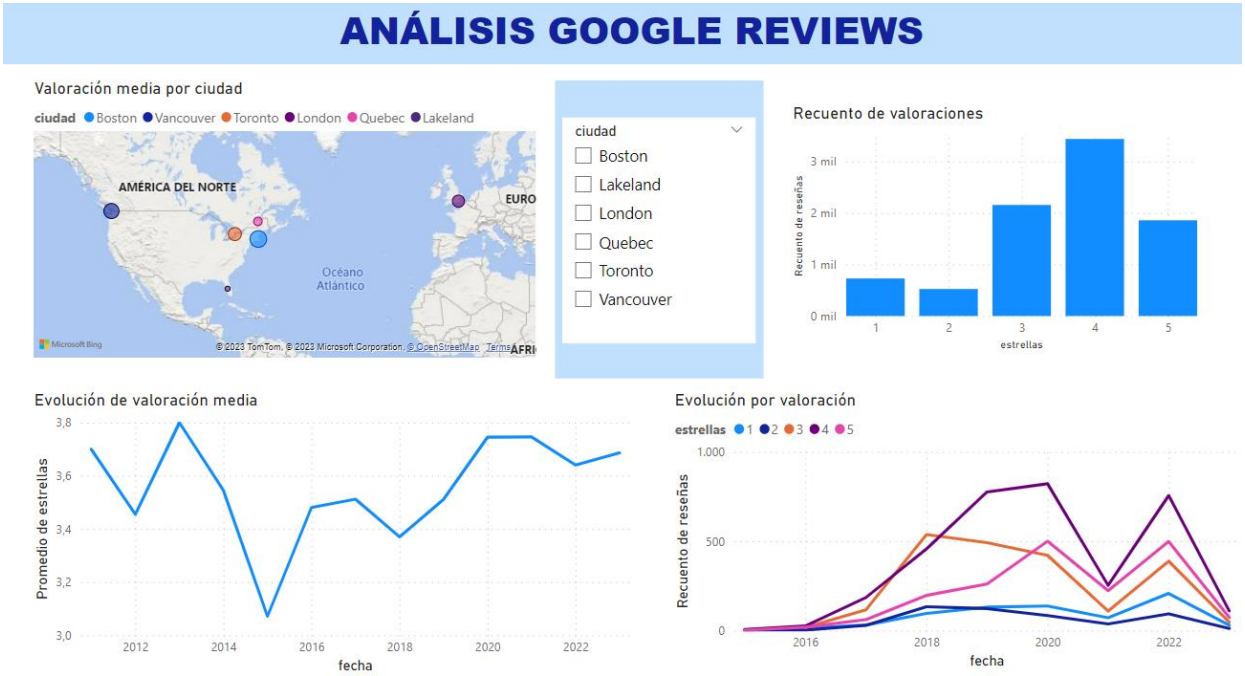
3.5. Sección 2: Análisis de opinión pública

Siguiendo con las visualizaciones obtenidas en PowerBI, hemos hecho uso de los resultados de la sección anterior para generar algunos dashboards relacionados con el análisis de los sentimientos hacia la compañía H&M de distintas fuentes. Para ello se han desarrollado 3 páginas con distintas visualizaciones relacionadas con estos datos, que pasamos a detallar a continuación.

Análisis Google Reviews

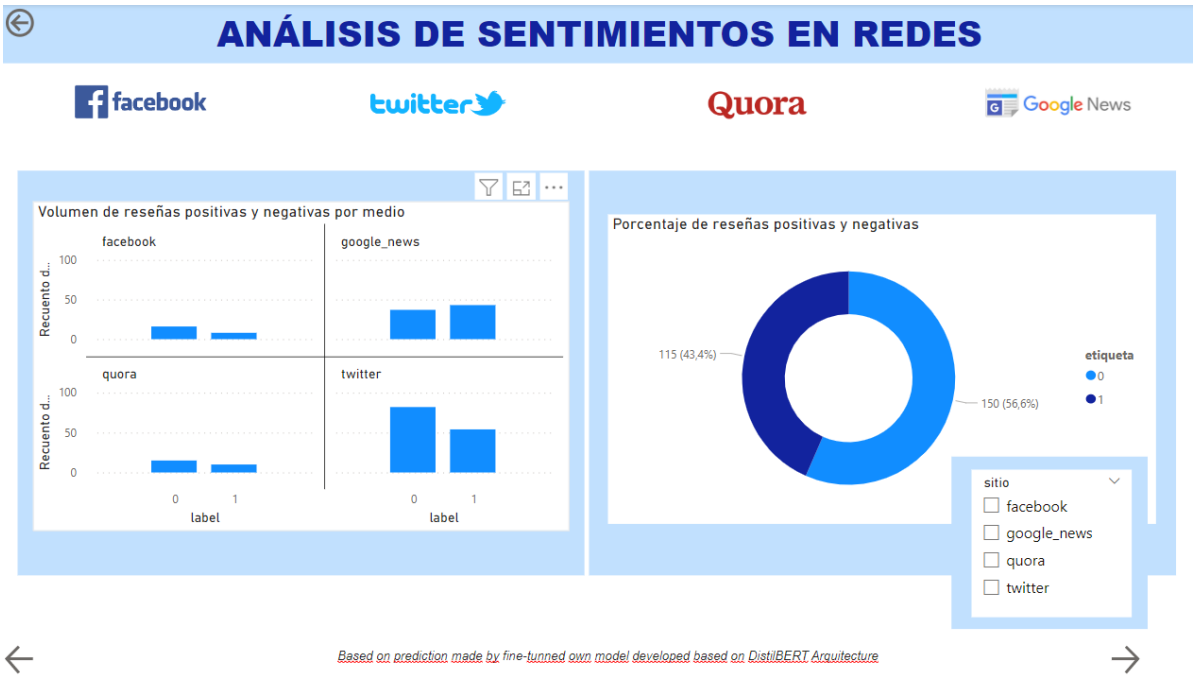
En esta primera página analizamos las reseñas recopiladas de Google Reviews para el entrenamiento de nuestro modelo visto en la sección anterior. Estas visualizaciones están basadas en información extraída por scraping.

- Visualización geográfica (el tamaño de la burbuja depende de la valoración media de la tienda).
- Distribución de las valoraciones o estrellas a lo largo del tiempo y su recuento total.



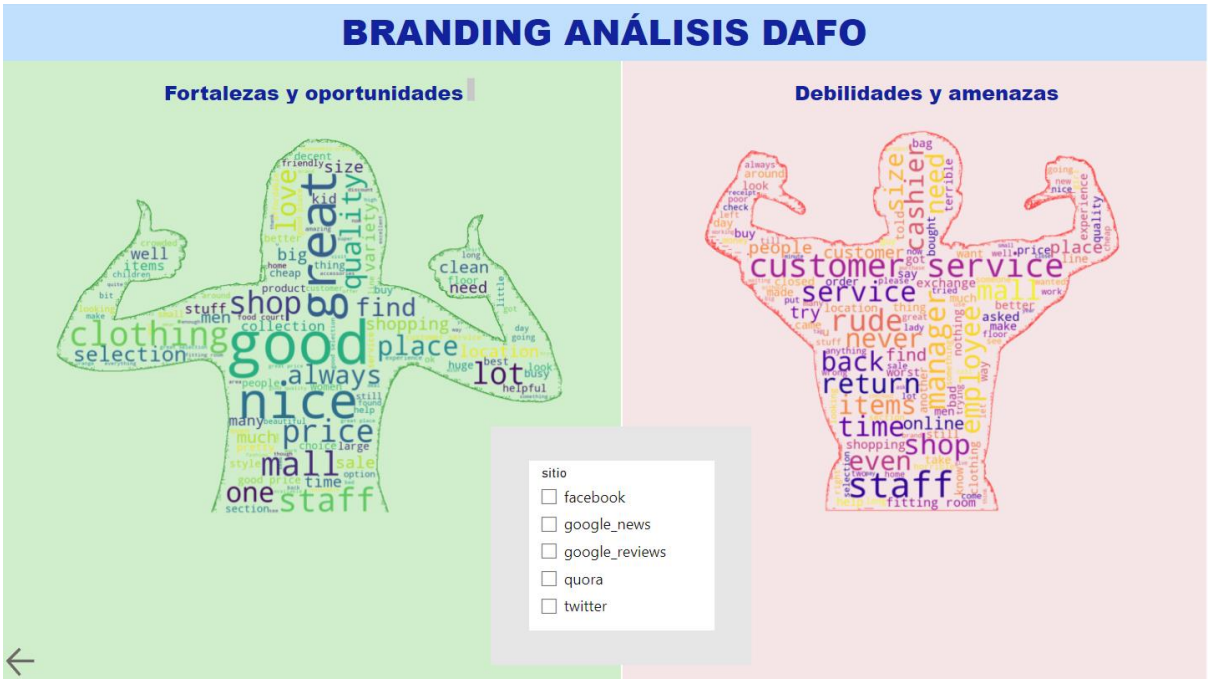
Análisis Sentimientos en Redes

Una vez obtenidas las predicciones de sentimientos para todos los comentarios de las distintas páginas, dedicamos esta sección a realizar un análisis sobre ellos. Para cada una de las fuentes realizamos un recuento de los comentarios que el modelo ha clasificado como positivos o negativos.



Word Cloud

Además de las visualizaciones previas, hemos generado nubes de palabras tanto para los comentarios clasificados como negativos y positivos. Además, hemos añadido un filtro por web para poder generar estas nubes de palabras, permitiendo filtrar por una fuente concreta. El Word Cloud dinámico puede accederse desde el archivo loca, en la app de PowerBI Service es estático.



3.6. Publicación del informe

Una vez se generaron todos los dashboards se decidió que la mejor forma de compartirlos era mediante la opción que ofrece la versión Pro de PowerBI de compartir informes. Esta funcionalidad nos permite mediante un enlace compartir nuestro proyecto con cualquier miembro de la comunidad universitaria, facilitando así su compartición y visualización de los datos. Mediante el siguiente enlace se puede acceder a dicho informe y testear con los distintos filtrados que ofrece.

4. CAMPAÑAS DE MARKETING: TRABAJO Y RESULTADOS

La tercera tarea del proyecto consiste en predecir el éxito de una campaña de marketing en función del presupuesto en distintos medios publicitarios y el tipo de influencer contratado.

4.1. Descripción de los datos

El dataset de referencia es un dataset disponible en el repositorio de Kaggle [12] que cuenta con datos de 4572 campañas publicitarias. Podemos distinguir dos tipos de variables:

- Variables explicativas: Para cada campaña, el dataset contiene el presupuesto (en millones) invertido en cada medio (anuncios de televisión, radio y redes sociales) y magnitud del influencer contratado.
- Variable respuesta: El objetivo de la predicción es el número de ventas (en millones) de una campaña.

	TV	Radio	Social Media	Influencer	Sales
0	16.0	6.566231	2.907983	Mega	54.732757
1	13.0	9.237765	2.409567	Mega	46.677897
2	41.0	15.886446	2.913410	Mega	150.177829
3	83.0	30.020028	6.922304	Mega	298.246340
4	15.0	8.437408	1.405998	Micro	56.594181

4.2. Preprocesamiento

Para el preprocesamiento hemos utilizado las librerías de Python pandas, numpy y seaborn.

Imputación de valores nulos

Para no perder información relevante se ha optado por realizar la imputación de valores nulos en lugar de eliminarlos.

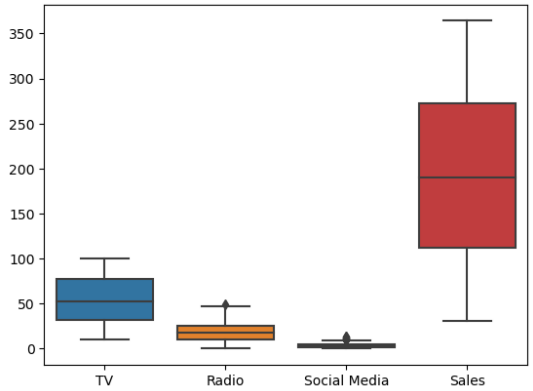
- Los valores nulos de las variables continuas (presupuesto en los distintos medios) han sido imputados por la media.
- Los valores nulos de las variables continuas (influencer contratado) han sido imputados por la moda.

Codificación de los datos

Los únicos datos que era preciso adecuar para la predicción eran los de “Tipo de influencer”. Al tratarse de datos categóricos que implican una relación de orden (nanoinfluencer tiene menor peso que macroinfluencer) se ha optado por utilizar un Ordinal Encoding.

Detección de outliers

El siguiente paso que se ha realizado es detectar los valores outliers o valores atípicos, es decir, valores que se salen del rango de valores normales que toma una variable e introducen ruidos y sesgos innecesarios en nuestro modelo. Para detectar los outliers (y si son superiores o inferiores) hemos utilizado los gráficos de cajas o boxplot, que se pueden observar en la figura inferior. Una vez detectados, para eliminar estos outliers, hemos definido una función que utiliza la técnica del rango intercuartílico (método IRQ).

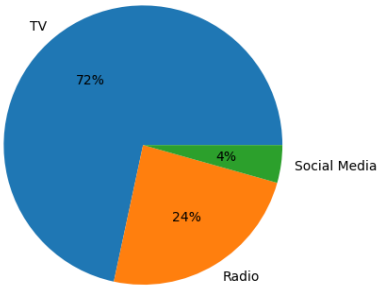


Las variables que presentan valores atípicos son el presupuesto en radio y en redes sociales.

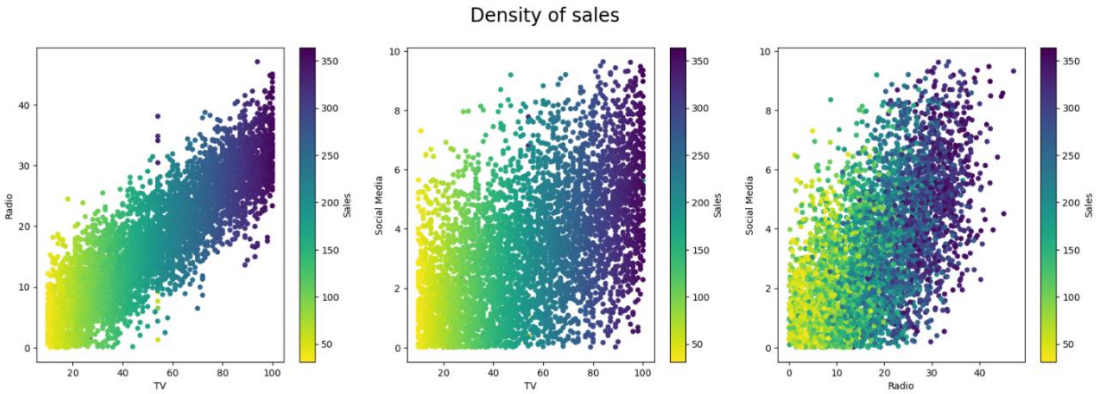
4.3. Análisis exploratorio y visualizaciones preliminares

Además de los datos de predicción, se han obtenido algunas visualizaciones preliminares que pueden resultar de interés desde el punto de vista práctico. Utilizamos las librerías de Matplotlib y Seaborn. En esta memoria, comentaremos únicamente los más relevantes, el resto pueden consultarse en el código fuente.

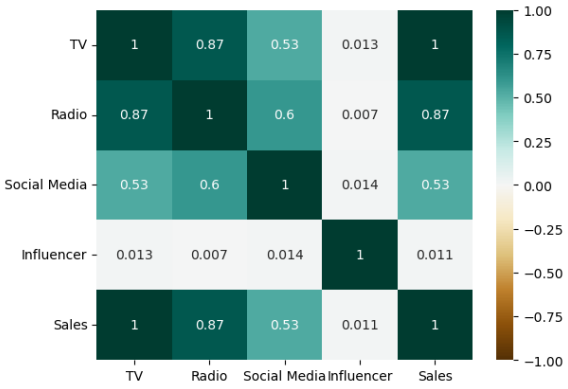
- Presupuesto total invertido en cada uno de los medios.
  Se observa que dónde más se invierte es en anuncios televisivos.



- Concentración del número de ventas. Podemos ver que al aumentar el presupuesto en el medio del eje X y del eje Y, aumenta la densidad de ventas.



- Estudio de correlaciones entre variables.
  Esto ofrece una primera intuición sobre qué medios son más importantes para una campaña de marketing.
  Posteriormente, esta información será complementada con el gráfico de importancia de características obtenido a partir del modelo de random forest.
  Las características más relevantes sobre las ventas son los presupuestos en TV y radio.



4.4. Modelos de predicción y otras consideraciones

Nos encontramos ante un problema de regresión, que abordaremos haciendo uso de la herramienta sklearn. Ante ello, dividimos los datos en entrenamiento y test y procedemos a definir y evaluar los siguientes modelos.

Regresión lineal clásica

- Modelo de regresión

Utilizando la clase LinearRegression, obtenemos un modelo de regresión lineal que predice las ventas a razón del siguiente modelo matemático:

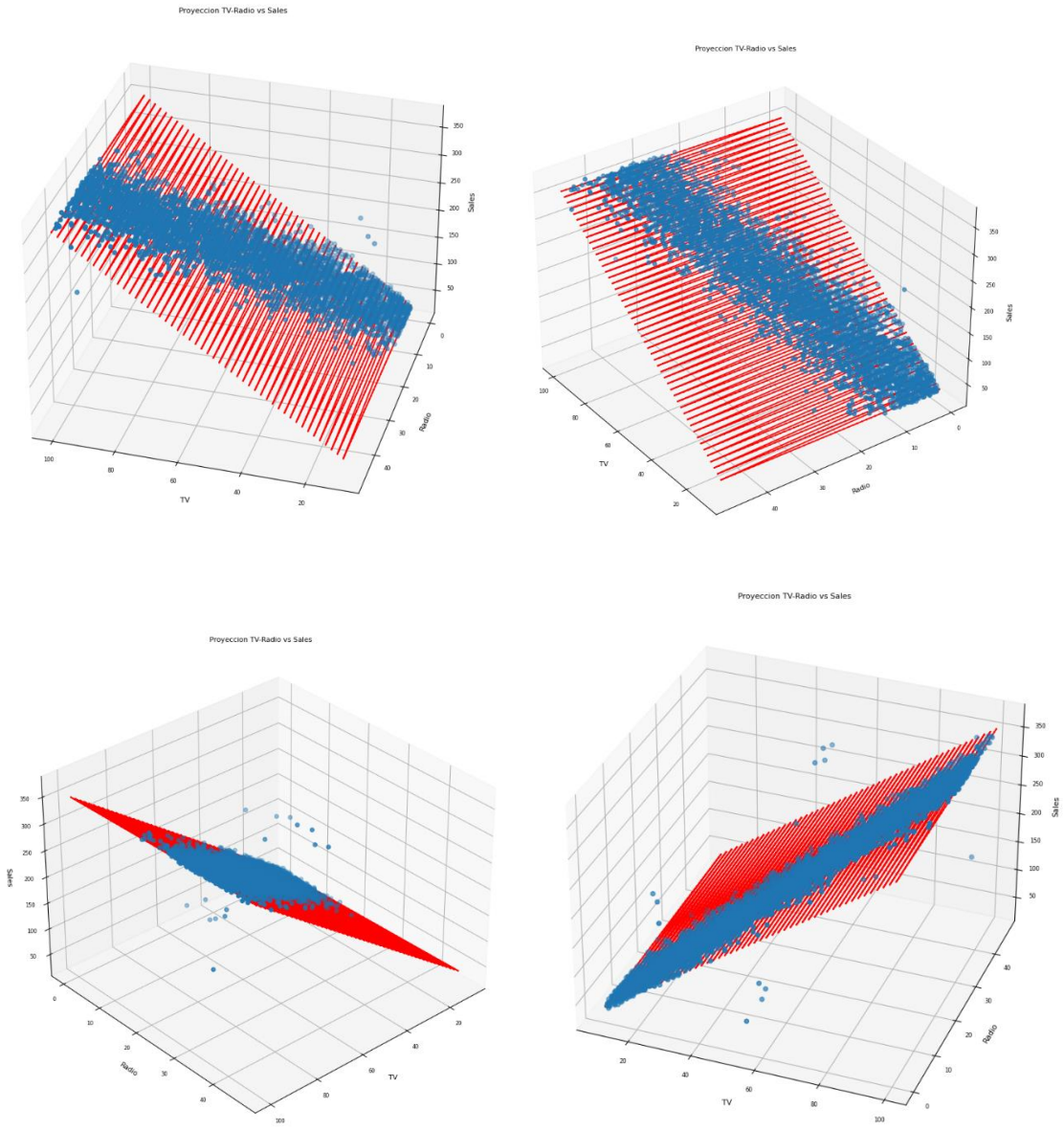
sales = 0.58 + 3.51 * TV + 0.11 * Radio + 0.06 * SocialMedia + -0.10 * Influencer

- Interpretabilidad del modelo: proyección 3D

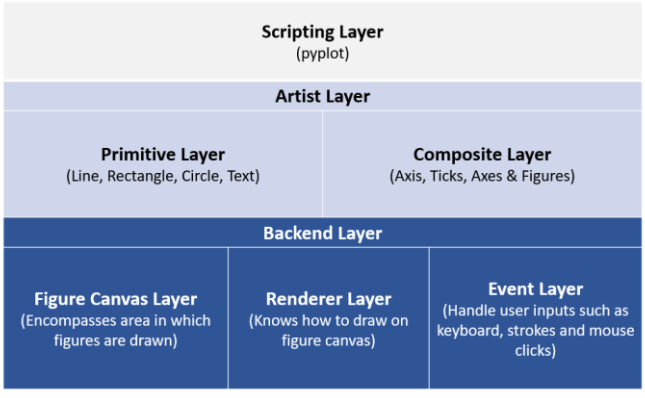
Un modelo matemático de regresión en 3 dimensiones es fácil de interpretar gráficamente. Si el plano solución cubre o se aproxima transversalmente a la nube de puntos a predecir, entonces, el modelo se ajusta a los datos de forma precisa. En nuestro caso contamos con un modelo en 5 dimensiones, por lo que la interpretación gráfica no queda tan clara. Proponemos lo siguiente.

Para dar una idea intuitiva del modelo matemático y dotar al modelo de explicabilidad, proponemos obtener una representación gráfica móvil (gif) de la proyección (ya que el modelo 5-dimensional) de este hiperplano en 3D (sales, TV y Radio). TV, Radio y ventas son las variables que mayor correlación presentan, así que la proyección sobre este triedro tridimensional sería el que representaría mejor el modelo.

Los resultados han sido sorprendentes y refuerzan la precisión del modelo que ya nos habían proporcionado las métricas de error. En el informe no podemos adjuntar el gif, pero añadimos algunos frames representativos.



Para desarrollar estos gráficos, al tratarse de algo más específico, Seaborn no nos proporciona una interfaz sencilla para desarrollarlo. Debemos de acudir a Matplotlib y utilizar la capa “Artist layer” (utilizando las clases Figure, Axes…), situada a un nivel inferior respecto al “Scripting layer” que estamos habituados a utilizar (pyplot).



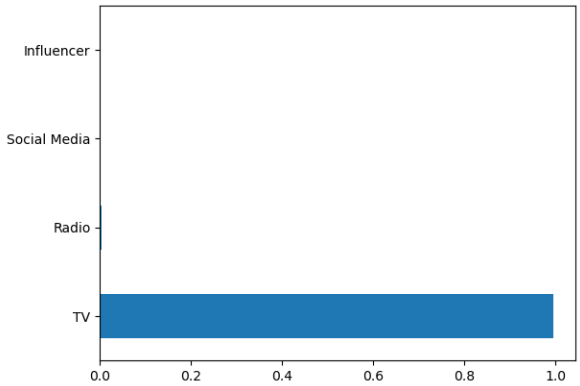
Regresor Random Forest

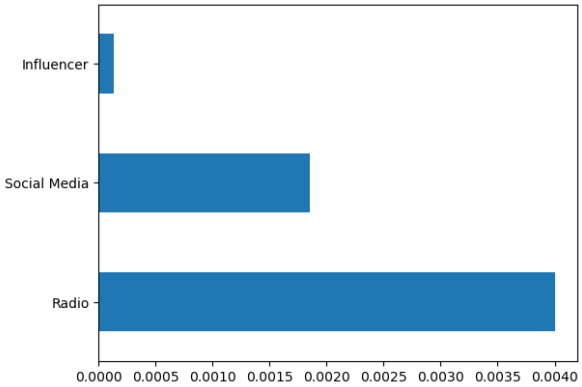
- Modelo de regresión

Utilizando la clase RandomForestRegressor de sklearn obtenemos un modelo de regresión para predecir el número de ventas. Los parámetros considerados para este modelo son una profundidad máxima de 9 nodos y un bosque de 10 árboles (estimadores). A diferencia del caso anterior, con este modelo no obtenemos una expresión matemática.

- Importancia de los atributos

Una de las principales ventajas que nos proporciona el regresor random forest es que reduce el sesgo del modelo al utilizar varios regresores y que nos permite categorizar los atributos según su importancia. En nuestro estudio, esto es especialmente importante ya que, desde el punto de vista del marketing, no solo nos interesa predecir el éxito de una campaña, si no también detectar qué factores son cruciales para el éxito de una campaña, para invertir más en ellos. En este caso el presupuesto en TV es la característica más relevante.





Rendimiento de los modelos

Algunas métricas de interés para medir la bondad de un modelo de regresión son el MSE, el RMSE, el MAE, el R^2 y el MAPE.

$$MSE = \frac{1}{N}\sum_{i=1}^{N} (y_i - \hat{y})^2 \qquad RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N} (y_i - \hat{y})^2} \qquad MAE = \frac{1}{N}\sum_{i=1}^{N} |y_i - \hat{y}|$$

$$MAPE = \frac{100\%}{n}\sum_{i=1}^{N} \left|\frac{y-\hat{y}}{y}\right| \qquad R^2 = 1 - \frac{\sum_{i=1}^{N} (y_i-\hat{y})^2}{\sum_{i=1}^{N} (y_i-\bar{y})^2}$$

$$donde:$$
$$y - variable\ respuesta$$
$$\hat{y} - valor\ predicho\ y$$
$$\bar{y} - \ valor\ medio\ de\ y$$

De ellas, las que tienen mayor interés son el R^2 y el MAPE, por su interpretabilidad.

- El R^2 es la llamada métrica de bondad de ajuste. Toma valores en el rango [0,1], indicando los valores más próximos a 1 que el modelo explica mejor la variable respuesta.
- El MAPE es la métrica del error porcentual. Un valor alto se traduce en un porcentaje medio de error alto y un valor bajo en un porcentaje de error bajo.

Hemos obtenido todas estas métricas para evaluar la bondad de los modelos de regresión lineal y random forest. Mostramos a continuación los resultados para el R^2 y el MAPE. Puede consultar los restantes resultados en el repositorio de código del proyecto.

	LinearRegression	RandomForestRegressor
MAPE	1.966149	2.238034
R^2	0.994919	0.993878

El R^2 es muy próximo a 1 y el error porcentual está entre el 1'97% y el 2'2% (valores porcentuales bajos), por lo tanto, podemos concluir que los modelos obtenidos son bastante precisos.

REFERENCIAS Y ENLACES DE INTERÉS

Código e informes

- Informe de PowerBI
  El enlace anterior le redirigirá al informe público desarrollado por el grupo.
- Carta del modelo de Hugging Face
  Este enlace le permitirá acceder a la página del modelo y testar su funcionamiento.
- Repositorio de código
  Este enlace le dará acceso al código desarrollado para este proyecto.

Bibliografía y referencias

[1] «Reseñas tiendas de H&M en Gran Bretaña (Londres),» [En línea]. Available: https://goo.gl/maps/c7SKunpHWNrwTEXZA, https://goo.gl/maps/c7SKunpHWNrwTEXZA.

[2] «Reseñas tiendas de H&M en USA (Lakeland y Boston),» [En línea]. Available: https://goo.gl/maps/221mTxNH7oP3PU9P9, https://goo.gl/maps/tHCCyazg7CM2AG8w7.

[3] «Reseñas tiendas de H&M Canada (Toronto, Quebec y Vancouver),» [En línea]. Available: https://goo.gl/maps/Qn2VVd9jt78Nq58YA, https://goo.gl/maps/BrzZDyr9rJZBkgaEA, https://goo.gl/maps/uYFmXfEDco1WL4Lr5, https://goo.gl/maps/aE22wex13tEoYXFx8.

[4] «Cadenas de publicaciones sobre H&M en Facebook,» [En línea]. Available: https://www.facebook.com/HM-2045252695742351/reviews, https://www.facebook.com/HM-864963097024789/reviews/?ref=page_internal, https://www.facebook.com/HM-2008699865868134/reviews/?ref=page_internal.

[5] «Cadenas de posts sobre H&M en Quora,» [En línea]. Available: https://www.quora.com/Are-H-M-clothes-good-quality, https://www.quora.com/What-do-you-think-of-the-fashion-brand-H-M.

[6] «Titulares de noticias relacionadas con H&M recogidas en Google News,» [En línea]. Available: https://news.google.com/search?q=%22h%26m%22&hl=es&gl=ES&ceid=ES%3Aes.

[7] TensorHub, «Modelo NNLM-EN-DIM50,» [En línea]. Available: https://tfhub.dev/google/nnlm-en-dim50/2.

[8] Hugging-Face, «DistilBERT base model,» [En línea]. Available: https://huggingface.co/distilbert-base-uncased.

[9] V. Sanh, L. Debut, J. Chaumond y T. Wolf, «DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter,» 2020. [En línea]. Available: https://arxiv.org/pdf/1910.01108.pdf.

[10] «Modelo NLP obtenido, en nuestro repositorio de Hugging face,» [En línea]. Available: https://huggingface.co/lourdesLB/finetuning-sentiment-model?text=this+store+is+so+cool+its+set+out+in+a+great+way+that+you+can+see+everything.

[11] «H&M transactions dataset,» Kaggle, [En línea]. Available: https://www.kaggle.com/competitions/h-and-m-personalized-fashion-recommendations/data.

[12] «Marketing campaigns dataset,» Kaggle, [En línea]. Available: https://www.kaggle.com/datasets/harrimansaragih/dummy-advertising-and-sales-data.