UNIVERSIDAD DE BUENOS AIRES
FACULTAD DE INGENIERÍA

75.06/95.58 Organización de Datos

Segundo Cuatrimestre de 2018

Trabajo Práctico 2:

Predicting User Conversions - www.trocafone.com

Grupo 42:

Farfán, Verónica

Basconcelo, Eliana

Link de Github:
https://github.com/vpfarfan/Organizacion-de-datos-20182/tree/master/TP2

1. Introducción

El objetivo del siguiente informe es predecir conversiones de usuarios en un período determinado en el sitio web www.trocafone.com mediante el uso de Machine Learning. A modo de continuación del Trabajo Práctico 1 dónde analizamos los datos del sitio web; ahora intentaremos intentar determinar, para cada usuario presentado, cuál es la probabilidad de que ese usuario realice una conversión en Trocafone en un periodo determinado.

Por un lado, los datos obtenidos se encuentran en un archivo .CSV, con el mismo formato utilizado en el TP1 en el cual se encuentra la siguiente información con un límite temporal de datos hasta el 31/05/2018.

Por otro lado, existe un archivo CSV que indica para un subconjunto de los usuarios incluidos en el set de eventos si los mismos realizaron una conversión (columna label = 1) o no (columna label = 0) desde el 01/06/2018 hasta el 15/06/2018.

Repasando los eventos posibles para los usuarios, estos son:

- **“viewed product”:** El usuario visita una página de producto.
- **“brand listing”:** El usuario visita un listado específico de una marca viendo un conjunto de productos.
- **“visited site”:** El usuario ingresa al sitio a una determinada url.
- **“ad campaign hit”:** El usuario ingresa al sitio mediante una campaña de marketing online.
- **“generic listing”:** El usuario visita la homepage.
- **“searched products”:** El usuario realiza una búsqueda de productos en la interfaz de búsqueda del site.
- **“search engine hit”:** El usuario ingresa al sitio mediante un motor de búsqueda web.
- **“checkout”:** El usuario ingresa al checkout de compra de un producto.
- **“staticpage”:** El usuario visita una página
- **“conversion”:** El usuario realiza una conversión, comprando un producto.
- **“lead”:** El usuario se registra para recibir una notificación de disponibilidad de stock, para un producto que no se encontraba disponible en ese momento.

El análisis realizado en el TP1 nos permitió explorar y extraer información pertinente para la empresa en forma de insights que aportan conocimiento de los usuarios y del comportamiento de los mismos cuando interactúan con el sitio web. En instancias del TP2, el objetivo es entrenar un modelo de Machine Learning, de tal forma de poder indicar la probabilidad de que conjunto seleccionado de usuarios realice una conversión desde el 01/06/2018 al 15/06/2018.

2. Repaso de Insight obtenidos

A raíz del TP1, hemos obtenido interesantes insights que le servirán a Trocafone a entender a sus usuarios, a tomar medidas para mejorar y lograr el objetivo del negocio, es decir, concretar más ventas.

- Los usuarios de Trocafone, ingresan al sitio por Smartphones y Computadoras principalmente
- Tanto en sus Smartphones como en sus Computadoras, la mayoría elige Chrome como buscador

Con estos 2 insights Trocafone puede realizar acciones para optimizar su sitio web en vista Smartphone y Computadoras para que la experiencia de usuario sea mejor. Además, la optimización de interfaz debería ser a través de Chrome.

- El 66% del tráfico es pago y el 20% es orgánico
- La campaña publicitaria con más éxito en clicks, por ende visitas, fue realizada a través de Google
- Las campañas en buscadores (ej. Google) tienen mayor efectividad en Smartphones.
- Las campañas en otros sitios tienen igual efectividad en computadora como en dispositivos móviles

Esto significa que la empresa, debería realizar acciones para que su tráfico derive más a orgánico, como posicionarse en el buscador de Google. Pero, también implica que Trocafone debería centrar sus campañas de Ads en Google, ya que ha sido probada exitosa y hacerla disponible para individuos que entren por Smartphone o Computadoras.

- Los usuarios buscan en su mayoría dispositivos de la marca iPhone
- Apple y Samsung son las marcas más vistas y agregadas al carrito de compras (casi por igual)
- A la hora de comprar, los clientes eligen Samsung

Trocafone deberá analizar por qué los clientes desisten de comprar todos los iPhones que son buscados y/o colocados en el carrito de compras. ¿Será por el precio o por los medios de pago? Con este set de datos no lo podemos determinar, pero es un buen punto de partida para investigarlo.

- Cerca del 30% de usuarios que ingresan desde Brasil, lo hacen desde la región de San Pablo
- San Pablo, Minas Gerais, Río de Janeiro y Bahía concentran más del 50% de usuarios
- Lamentablemente hay muchos datos cuya región es Unknown

Para mejorar sus ventas, Trocafone debería hacer más campañas para ser reconocido en más regiones, sin perder de vista que tiene que mantener el volumen de usuarios de San Pablo y las regiones mayoritarias.

- Entre mayo y junio, hubo un gran salto en las visitas y la adición de productos al carrito de compras
- En el mismo período se observa una gran cantidad de hits de una campaña publicitaria en Google

- Este mismo salto no se vio reflejado en ventas, incluso las ventas de Samsung (la marca más comprada) decrecieron respecto del mes de abril. iPhone y Motorola tuvieron un leve crecimiento

Las conversiones no acompañaron bien a esta campaña de Google realizada a mediados de mayo. Pero, la campaña dejó un tráfico que acompaña al sitio hasta el día de corte de este estudio. Tal vez, la campaña logró hacer conocida la empresa e incrementar el número de futuros compradores pero no concretar ventas.

- 6% de productos vistos pasan al carrito de compras
- 0.16% de los productos vistos son comprados
- 2.6% de los productos agregados al carrito son comprados

Si bien todas las métricas son significativas para la estrategia de Trocafone, la que más relación directa tiene con los ingresos son estas 3. ¿Por qué los usuarios buscan tanto antes de decidirse agregarlo al carrito? ¿Por qué una vez agregado al carrito deciden no concretar la compra? En esto pueden influir muchos factores, tal vez la descripción de los productos o el abanico sea muy amplio y los usuarios no puedan decidirse por uno fácilmente. Considerando las características del negocio donde venden Smartphones refurbished (con calificación Excelente, Muy Bueno, Bueno) esto puede ser verdadero y la decisión al potencial comprador le lleva mucho tiempo y vistas de comparación. Pero el porqué de una vez agregado al carrito no se concreta la compra, tiene que ver con otros motivos que pueden ser envío, método de pago, o que prefieren ver el móvil en una tienda física, que exceden a los datos de este set para el análisis.

Con todos estos insights y conclusiones, Trocafone podría establecer estrategias, las que recomendamos u otras, para lograr un mayor tráfico de usuarios en su sitio, mejorar el targeting de Ads, optimización de su sitio e interfaz, mejora en la descripción de productos para disminuir las múltiples vistas de los mismos; todo esto en pos de incrementar la cantidad de productos en el checkout y por ende la cantidad de productos comprados.

3. Transformaciones realizadas a los datos

3.1 Procesamiento de Datos

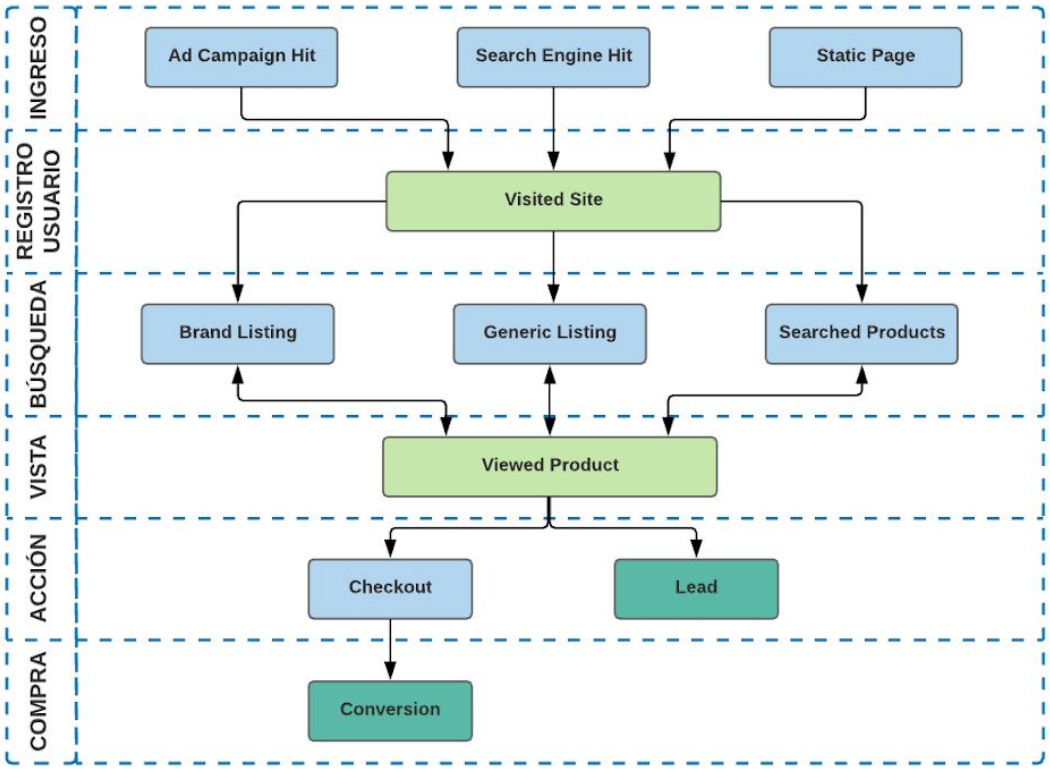
Basados en nuestro trabajo con los datos en el TP1, lo que realizamos en primer lugar es establecer que vamos a utilizar los datos segmentados por sesiones por usuarios.

En primer se agregaron columnas al set de datos relacionados con columnas existentes, a saber:

- ➔ day: día del evento - timestamp
- ➔ month: mes del evento - timestamp
- ➔ brand: marca del dispositivo - model
- ➔ session_id: identificador de sesiones para cada usuario
- ➔ last_timestamp: fecha del evento anterior de un mismo usuario
- ➔ new_session: primer evento de una sesión

Adicionalmente, se realizó el mismo tratamiento con los datos nulls y conversión de tipo de datos.

Como ya hemos definido, los eventos son interacciones con un sitio web, donde se observa cierta progresión de los mismos como muestra el gráfico a continuación. Con esta información hemos definido indicadores de orden a cada evento del 1 al 6 que permite calcular mejor la duración de cada evento.



Para las variables categóricas, reducimos la cantidad de categorías agrupando las menos frecuentes en una nueva categoría llamada 'variable_other'.

- ➔ Campaign_source
- ➔ Url
- ➔ OS
- ➔ Color

3.2 Definición de la sesión

En el TP1 decidimos agrupar eventos por sesiones considerando que los eventos no estén separados entre sí por más de 30 minutos y teniendo en cuenta que cada sesión debería tener un solo evento visited site.

Para realizar pruebas en este trabajo, definimos las sesiones mediante 2 métodos diferentes. Como conclusión podemos decir que los resultados no varían con el método utilizado así que optamos por continuar con la opción 2 que coincide con la elección del TP1.

- Opción 1 - según la duración de los eventos: para cada evento se define una duración máxima según la regla de proximidad de intercuartiles (fórmula de Tukey).
  "Tomando como referencia la diferencia entre el primer cuartil Q1 y el tercer cuartil Q3, o el valor intercuartil, se considera un valor extremo o atípico aquel que se encuentra a 1,5 veces esa distancia de uno de esos cuartiles (atípico leve) o a 3 veces esa distancia (atípico extremo). Las observaciones que están más allá de 3 veces el rango intercuartil se conocen como valores atípicos extremos."
- Opción 2 - por tiempo definido: que una sesión termina cuando el tiempo entre eventos es mayor a un valor T= 30 minutos.

4. Feature Engineering

4.1 Dataframes

Los dataframes creados para el análisis son:

- Dfperson: eventos agrupados por persona sin tener en cuenta las sesiones generadas
- User_session: eventos agrupados por sesiones y por personas. Cada sesión toma un id único

A este último dataframe se le agregan los siguientes features:

1. Cantidad de eventos por sesión
2. Promedio de eventos por sesión
3. Tiempo total en el sitio
4. Tiempo promedio por sesión
5. Cantidad de skus vistos en total
6. Promedio de skus vistos por sesión
7. Cantidad total de eventos de cada tipo (output del dataframe dfperson)
8. Ratio de conversión: cantidad de viewed/cantidad de conversión.
9. Ratio de conversión de checkout: cantidad de checkout/cantidad de conversión.
10. Eventos y sus cantidades

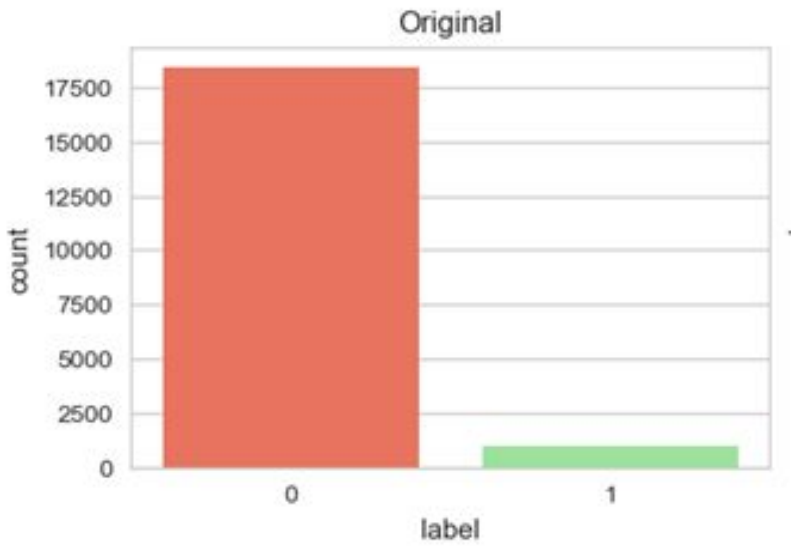
Para sumar información de la última transacción de cada usuario y el tiempo (en segundos) desde la primera a la última se genera un nuevo dataframe last_tx.

Por último, generamos un nuevo dataframe dfcontent que agrupa el contenido de los eventos de cada persona. Se crea una columna por cada valor que pueden acompañar a cada evento y cuenta las ocurrencias para cada usuario.

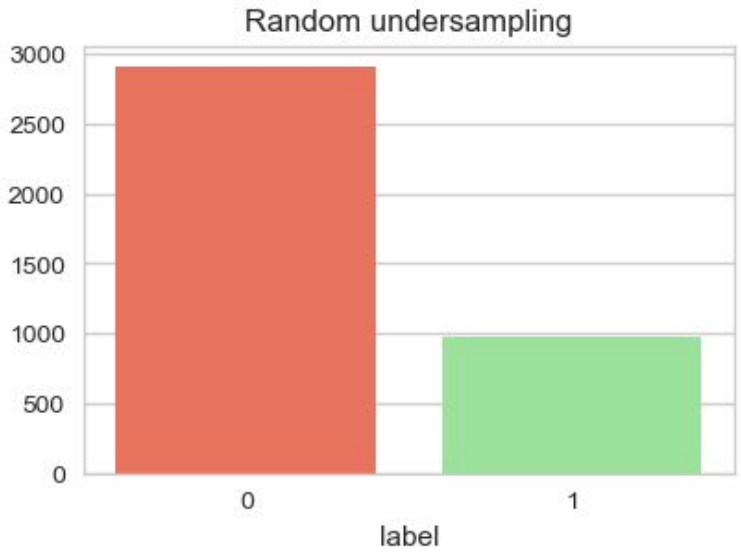
Todos estos features van a formar parte del dataset final para el entrenamiento del modelo.

5. Balanceo de datos y entrenamiento del Modelo

Cuando se incorporan los labels del set de entrenamiento, se observa que existe un desbalanceo de clases.



Undersampling: Como un primer tratamiento, hacemos una selección aleatoria de los casos con label 0 y nos quedamos con el 15% de éstos, mientras que de los casos con label 1 nos quedamos con el 100%.



Posterior al balanceo, separamos aleatoriamente un un set del 25% de los datos para usar como set de validación y aplicamos mean encoding a ambos sets: entrenamiento y testeo.
Una vez obtenidas las features para el modelo, creamos el dataset final con los datos de la última transacción y de cada sesión por persona.

6. Modelos utilizados

Los modelos enunciados a continuación y sus resultados fueron obtenidos luego de realizar el balanceo de datos undersampling en primer lugar y en segundo lugar la división de sets de entrenamiento y testeo. Para el último submit en Kaggle, se incorporó oversampling mediante SMOTE.

6.1 Regresión logística

Los datos fueron escalados usando robust_scaler teniendo en cuenta que los datos no siguen una distribución normal y es menos sensible a outliers.

Score de Entrenamiento:
CV Score : Mean - 0.7372249 | Std - 0.02707962 | Min - 0.6912598 | Max - 0.7659338

Score de Validación:
AUC: 0.73495

6.2 Random forest

Utilización de SMOTE para oversampling
Score de Entrenamiento:
CV Score : Mean - 0.8616016 | Std - 0.1187496 | Min - 0.6961342 | Max - 0.968556

Score de Validación:
AUC: 0.86188

6.3 Balanced Random forest

Score de Entrenamiento:
CV Score : Mean - 0.8585604 | Std - 0.009922133 | Min - 0.8432616 | Max - 0.8717769

Score de Validación:
AUC: 0.86180

6.4 XGBoost

Utilización de SMOTE para oversampling
Score de Entrenamiento:
CV Score : Mean - 0.8302901 | Std - 0.1560823 | Min - 0.6061831 | Max - 0.9707385

Score de Validación:
AUC: 0.86742

XGB con class_weight = 'balanced'
Score de Entrenamiento:
CV Score : Mean - 0.856784 | Std - 0.009116474 | Min - 0.8426599 | Max - 0.8703324

Score de Validación:
AUC: 0.86568

6.5 Gradient Boosting

Score de Entrenamiento:
CV Score : CV Score : Mean - 0.854332 | Std - 0.01300218 | Min - 0.8349086 | Max - 0.8675676

Score de Validación:
AUC: 0.86458

6.6 Ada Boost

Score de Entrenamiento:
CV Score : Mean - 0.8142266 | Std - 0.01617435 | Min - 0.7878269 | Max - 0.8318111

Score de Validación:
AUC: 0.81622

El algoritmo utilizado para submitir en Kaggle fue XGBoost con SMOTE. Para ello, cargamos los datos del set de testeo de Kaggle, le aplicamos encoding a las variables correspondientes y agregamos todas las variables usadas en el entrenamiento. El último submit nos dió un score de 0.85385.

7. Conclusiones

Para lograr el objetivo planteado de entrenar un modelo de Machine Learning para indicar la probabilidad de que conjunto seleccionado de usuarios de Trocafone realice una conversión, testeamos distintos modelos siendo el de mayor éxito XGBoost con SMOTE.

Un punto importante a destacar es que el desbalanceo de clases afecta mucho a la performance de los algoritmos. Utilizando el total de los datos, ninguno de los algoritmos que probamos dio buenos resultados, por eso decidimos disminuir la muestra de casos negativos de forma aleatoria para no introducir sesgo al modelo. La reducción de casos negativos la hicimos usando distintos ratios, llegando a la conclusión de que el mejor resultado (evaluando con AUC) se obtenía con un ratio de 0.15. De esta forma nos quedamos con aproximadamente la mitad de los casos negativos.

Además de reducir los casos negativos, decidimos utilizar un segundo tratamiento para desbalanceo de clases. Para algunos algoritmos, hacer oversampling de la clase minoritaria usando el método SMOTE resultó en una mejora de la métrica evaluada. Y para otros, simplemente utilizamos el tratamiento del desbalanceo de clases incluido en el mismo algoritmo. Este segundo método dió un mejor resultado que usando SMOTE, por ejemplo en Balanced Random Forest.

Respecto a la selección de features, hicimos varias pruebas comprobando cuáles afectan en mayor o menor medida a cada modelo y seleccionamos las que mejoraban la métrica de evaluación. En los casos de variables categóricas, para las que usamos mean encoding, debimos poner especial cuidado, ya que comprobamos que un error en la implementación del mismo puede resultar en overfitting.

Para evitar el overfitting, luego de separar un set de testeo y uno de entrenamiento, realizamos la codificación de variables sólo con los datos del set de entrenamiento y luego aplicamos el resultado al set de testeo. En los casos en que usamos SMOTE, lo hicimos sólo sobre el set de entrenamiento, también para evitar tener overfitting en el modelo. De esta forma el valor de AUC que obtuvimos para los algoritmos en el set de entrenamiento es muy cercano al obtenido en el set de testeo.

Balanced Random Forest y XGBoost con SMOTE, dieron resultados similares. Pero XGBoost con SMOTE fue el último seleccionado para submitir a Kaggle dando un score de 0.85385. Si bien se podrían mejorar ambos modelos haciendo un refinamiento de los hiper parámetros, consideramos que el resultado obtenido es bueno y le sería de mucha utilidad a Trocafone para poder predecir el comportamiento de los usuarios en su sitio.

Los insights obtenidos en el TP1 junto con el modelo armado para la predicción de conversión en este TP2 son herramientas muy útiles para Trocafone ya que le permitiría tomar decisiones estratégicas para aumentar las conversiones, y al mismo tiempo, mediante actualizaciones al modelo con los nuevos datos, predecir el comportamiento del usuario.