

75.06/95.58 Organización de Datos

Segundo Cuatrimestre de 2018

Trabajo Práctico 2

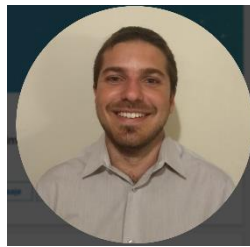
Equipo de oyentes: *“los de arriba son mejores”*

Integrantes:

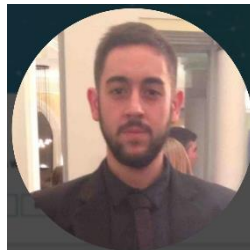
- Joaquín Pascual
joacox86@gmail.com



- Juan Augusto Sanchez
juanaugusto_sanchez@hotmail.com



- Matías Moris
moris.matias@yahoo.com.ar



- Rigoberto Malca
rigoxu@outlook.com



LINK AL CÓDIGO DESARROLLADO

https://github.com/rigoxu/tp2_organizacion_de_datos

CONSIGNA

Determinar cuál es la probabilidad de que ese usuario realice una conversión en Trocafone en un periodo determinado.

ARCHIVOS UTILIZADOS

events_up_to_01062018.csv → data de los eventos

labels_training_set.csv → tiene las etiquetas de conversión

trocafone_kaggle_test.csv → listado a predecir

PREPROCESAMIENTO DE DATOS Y FEATURING ENGINEERING

1. Breve análisis descriptivo del data set events_up_to_01062018.csv

Tamaño del data set: el data set tiene

2341681 observaciones, 23 variables

Las columnas de las que disponemos son

timestamp', 'event', 'person', 'url', 'sku', 'model', 'condition',
'storage', 'color', 'skus', 'search_term', 'staticpage',
'campaign_source', 'search_engine', 'channel', 'new_vs_returning',
'city', 'region', 'country', 'device_type', 'screen_resolution',
'operating_system_version',

2. Transformaciones y creación de variables necesarias

Variables en base al dataset original:

NOMBRE DE VARIABLE	CÓDIGO DE TRANSFORMACIÓN	VARIABLE O VARIABLES FUENTES
fecha	df['fecha'] = pd.to_datetime(df['timestamp'], format='%Y-%m-%d').dt.date df['fecha'] = df['fecha'].astype('datetime64[ns]')	timestamp
numero mes	df['month']= pd.DatetimeIndex(df['timestamp']).month	timestamp
numero dia	df['day']= pd.DatetimeIndex(df['timestamp']).day	timestamp
nombre mes	df.loc[df.month==1,'nombre_mes']='ENERO' df.loc[df.month==2,'nombre_mes']='FEBRERO' df.loc[df.month==3,'nombre_mes']='MARZO' df.loc[df.month==4,'nombre_mes']='ABRIL' df.loc[df.month==5,'nombre_mes']='MAYO' df.loc[df.month==6,'nombre_mes']='JUNIO' df.loc[df.month==7,'nombre_mes']='JULIO' df.loc[df.month==8,'nombre_mes']='AGOSTO' df.loc[df.month==9,'nombre_mes']='SEPTIEMBRE'	month

día de la semana	<pre>df['fecha2'] = df['fecha'].dt.dayofweek+1 # Sumo 1 para que no empiece en 0. df.loc[df.fecha2==1,'dia'] = 'LUNES' df.loc[df.fecha2==2,'dia'] = 'MARTES' df.loc[df.fecha2==3,'dia'] = 'MIERCOLES' df.loc[df.fecha2==4,'dia'] = 'JUEVES' df.loc[df.fecha2==5,'dia'] = 'VIERNES' df.loc[df.fecha2==6,'dia'] = 'SABADO' df.loc[df.fecha2==7,'dia'] = 'DOMINGO' df.loc[df.fecha2==1,'lunes'] =1 df.loc[df.fecha2==2,'martes'] =1 df.loc[df.fecha2==3,'miercoles'] =1 df.loc[df.fecha2==4,'jueves'] =1 df.loc[df.fecha2==5,'viernes'] =1 df.loc[df.fecha2==6,'sabado'] =1 df.loc[df.fecha2==7,'domingo'] =1</pre>	fecha
week o weekend	<pre># Definimos si es durante la semana o el finde: df.loc[df.fecha2<=5,'week'] = 'WEEK' df.loc[df.fecha2>5,'week'] = 'WEEKEND' df.loc[df.fecha2<=5,'flag_week']=1 df.loc[df.fecha2>5,'flag_weekend']=1</pre>	fecha
flags de marca vs brand_viewed	<pre>df.loc[(df['model'].str.lower().str.contains('iphone', na=False)==True),'brand_viewed']='IPHONE' df.loc[(df['model'].str.lower().str.contains('samsung', na=False)==True)&(pd.isna(df.brand_viewed)),'brand_viewed']='SAMSUNG' df.loc[(df['model'].str.lower().str.contains('motorola', na=False)==True)&(pd.isna(df.brand_viewed)),'brand_viewed']='MOTOROLA' df.loc[(df['model'].str.lower().str.contains('sony', na=False)==True)&(pd.isna(df.brand_viewed)),'brand_viewed']='SONY' df.loc[(df['model'].str.lower().str.contains('lg', na=False)==True)&(pd.isna(df.brand_viewed)),'brand_viewed']='LG' df.loc[(df['model'].str.lower().str.contains('asus', na=False)==True)&(pd.isna(df.brand_viewed)),'brand_viewed']='ASUS' df.loc[(df['model'].str.lower().str.contains('ipad', na=False)==True)&(pd.isna(df.brand_viewed)),'brand_viewed']='IPAD' df.loc[(df['model'].str.lower().str.contains('lenovo', na=False)==True)&(pd.isna(df.brand_viewed)),'brand_viewed']='LENOVO' df.loc[(df.event=='viewed product')&(pd.isna(df.brand_viewed)),'brand_viewed']='OTROS'</pre>	brand_viewed,model

flags por tipo de evento	<pre> df.loc[df.event=='visited site','visit']=1 df.loc[df.event=='viewed product','viewprod']=1 df.loc[df.event=='checkout','checkout']=1 df.loc[df.event=='brand listing','brand_listing']=1 df.loc[df.event=='ad campaign hit','ad_camp_hit']=1 df.loc[df.event=='generic listing','generic_listing']=1 df.loc[df.event=='searched products','searched_products']=1 df.loc[df.event=='search engine hit','search_engine_hit']=1 df.loc[df.event=='staticpage','staticpage']=1 df.loc[df.event=='conversion','conversion']=1 df.loc[df.event=='lead','lead']=1 </pre>	event
flag por modelos buscado	<pre> # Modelo buscado: df.loc[(df.searched_products==1)&(df['search_term'].str.lower().str.contains('iphone', na=False)==True),'searched_iphone']=1 df.loc[(df.searched_products==1)&(df['search_term'].str.lower().str.contains('6s', na=False)==True),'searched_iphone_6s']=1 df.loc[(df.searched_products==1)&(df['search_term'].str.lower().str.contains('7s', na=False)==True),'searched_iphone_7s']=1 df.loc[(df.searched_products==1)&(df['search_term'].str.lower().str.contains('iphone 7', na=False)==True),'searched_iphone_7']=1 df.loc[(df.searched_products==1)&(df['search_term'].str.lower().str.contains('iphone 6', na=False)==True),'searched_iphone_6']=1 df.loc[(df.searched_products==1)&(df['search_term'].str.lower().str.contains('iphone 5', na=False)==True),'searched_iphone_5']=1 df.loc[(df.searched_products==1)&(df['search_term'].str.lower().str.contains('samsung', na=False)==True),'searched_samsung']=1 df.loc[(df.searched_products==1)&(df['search_term'].str.lower().str.contains('sony', na=False)==True),'searched_sony']=1 df.loc[(df.searched_products==1)&(df['search_term'].str.lower().str.contains('xperia', na=False)==True),'searched_xperia']=1 df.loc[(df.searched_products==1)&(df['search_term'].str.lower().str.contains('motorola', na=False)==True),'searched_motorola']=1 df.loc[(df.searched_products==1)&(df['search_term'].str.lower().str.contains('a5', na=False)==True),'searched_a5']=1 df.loc[(df.searched_products==1)&(df['search_term'].str.lower().str.contains('a7', na=False)==True),'searched_a7']=1 df.loc[(df.searched_products==1)&(df['search_term'].str.lower().str.contains('j2', na=False)==True),'searched_j2']=1 df.loc[(df.searched_products==1)&(df['search_term'].str.lower().str.contains('j3', na=False)==True),'searched_j3']=1 df.loc[(df.searched_products==1)&(df['search_term'].str.lower().str.contains('j4', na=False)==True),'searched_j4']=1 df.loc[(df.searched_products==1)&(df['search_term'].str.lower().str.contains('j5', na=False)==True),'searched_j5']=1 df.loc[(df.searched_products==1)&(df['search_term'].str.lower().str.contains('j6', na=False)==True),'searched_j6']=1 df.loc[(df.searched_products==1)&(df['search_term'].str.lower().str.contains('j7', na=False)==True),'searched_j7']=1 df.loc[(df.searched_products==1)&(df['search_term'].str.lower().str.contains('s6', na=False)==True),'searched_s6']=1 df.loc[(df.searched_products==1)&(df['search_term'].str.lower().str.contains('s7', na=False)==True),'searched_s7']=1 df.loc[(df.searched_products==1)&(df['search_term'].str.lower().str.contains('s8', na=False)==True),'searched_s8']=1 df.loc[(df.searched_products==1)&(df['search_term'].str.lower().str.contains('moto g', na=False)==True),'searched_moto_g']=1 </pre>	searched_products

```
df.loc[(df.searched_products==1)&(df['search_term'].str.lower().str.contains('moto',
na=False)==True),'searched_moto']=1
df.loc[(df.searched_products==1)&(df['search_term'].str.lower().str.contains('galaxy',
na=False)==True),'searched_galaxy']=1
df.loc[(df.searched_products==1)&(df['search_term'].str.lower().str.contains('lg',
na=False)==True),'searched_lg']=1
df.loc[(df.searched_products==1)&(df['search_term'].str.lower().str.contains('asus',
na=False)==True),'searched_asus']=1
```

flags por modelos
vistos

```
# Modelo visto:
df.loc[(df.brand_viewed=='IPHONE')&(df['model'].str.lower().str.contains('iphone se',
na=False)==True), 'iphone_se_viewed']=1
df.loc[(df.brand_viewed=='IPHONE')&(df['model'].str.lower().str.contains('iphone 5',
na=False)==True), 'iphone_5_viewed']=1
df.loc[(df.brand_viewed=='IPHONE')&(df['model'].str.lower().str.contains('iphone 6',
na=False)==True), 'iphone_6_viewed']=1
df.loc[(df.brand_viewed=='IPHONE')&(df['model'].str.lower().str.contains('iphone 7',
na=False)==True), 'iphone_7_viewed']=1
df.loc[(df.brand_viewed=='SAMSUNG')&(df['model'].str.lower().str.contains('galaxy s8 plus',
na=False)==True), 'samsung_s8_plus']=1
df.loc[(df.brand_viewed=='SAMSUNG')&(df['model'].str.lower().str.contains('galaxy s8',
na=False)==True), 'samsung_s8']=1
df.loc[(df.brand_viewed=='SAMSUNG')&(df['model'].str.lower().str.contains('galaxy s7',
na=False)==True), 'samsung_s7']=1
df.loc[(df.brand_viewed=='SAMSUNG')&(df['model'].str.lower().str.contains('galaxy s6',
na=False)==True), 'samsung_s6']=1
df.loc[(df.brand_viewed=='SAMSUNG')&(df['model'].str.lower().str.contains('galaxy s5',
na=False)==True), 'samsung_s5']=1
df.loc[(df.brand_viewed=='SAMSUNG')&(df['model'].str.lower().str.contains('galaxy s3',
na=False)==True), 'samsung_s5']=1
df.loc[(df.brand_viewed=='SAMSUNG')&(df['model'].str.lower().str.contains('galaxy s7 edge',
na=False)==True), 'samsung_s7_viewed']=1
df.loc[(df.brand_viewed=='SAMSUNG')&(df['model'].str.lower().str.contains('galaxy s6 edge',
na=False)==True), 'samsung_s6_viewed']=1
df.loc[(df.brand_viewed=='SAMSUNG')&(df['model'].str.lower().str.contains('galaxy note',
na=False)==True), 'samsung_note']=1
df.loc[(df.brand_viewed=='SAMSUNG')&(df['model'].str.lower().str.contains('galaxy note 8',
na=False)==True), 'samsung_note_8']=1
df.loc[(df.brand_viewed=='SAMSUNG')&(df['model'].str.lower().str.contains('galaxy a7',
na=False)==True), 'samsung_a7']=1
df.loc[(df.brand_viewed=='SAMSUNG')&(df['model'].str.lower().str.contains('galaxy a8',
na=False)==True), 'samsung_a8']=1
df.loc[(df.brand_viewed=='SAMSUNG')&(df['model'].str.lower().str.contains('galaxy a5',
na=False)==True), 'samsung_a5']=1
df.loc[(df.brand_viewed=='SAMSUNG')&(df['model'].str.lower().str.contains('galaxy j7',
na=False)==True), 'samsung_j7']=1
df.loc[(df.brand_viewed=='SAMSUNG')&(df['model'].str.lower().str.contains('galaxy j5',
na=False)==True), 'samsung_j5']=1
df.loc[(df.brand_viewed=='SAMSUNG')&(df['model'].str.lower().str.contains('galaxy j4',
na=False)==True), 'samsung_j4']=1
df.loc[(df.brand_viewed=='SAMSUNG')&(df['model'].str.lower().str.contains('galaxy j3',
na=False)==True), 'samsung_j3']=1
df.loc[(df.brand_viewed=='SAMSUNG')&(df['model'].str.lower().str.contains('galaxy j2',
na=False)==True), 'samsung_j2']=1
df.loc[(df.brand_viewed=='SAMSUNG')&(df['model'].str.lower().str.contains('galaxy j1',
na=False)==True), 'samsung_j1']=1
df.loc[(df.brand_viewed=='MOTOROLA')&(df['model'].str.lower().str.contains('moto g5',
na=False)==True), 'moto_g5']=1
df.loc[(df.brand_viewed=='MOTOROLA')&(df['model'].str.lower().str.contains('moto g4',
na=False)==True), 'moto_g4']=1
df.loc[(df.brand_viewed=='MOTOROLA')&(df['model'].str.lower().str.contains('moto g3',
na=False)==True), 'moto_g3']=1
df.loc[(df.brand_viewed=='MOTOROLA')&(df['model'].str.lower().str.contains('moto g2',
na=False)==True), 'moto_g2']=1
df.loc[(df.brand_viewed=='MOTOROLA')&(df['model'].str.lower().str.contains('moto g',
na=False)==True), 'moto_g']=1
df.loc[(df.brand_viewed=='MOTOROLA')&(df['model'].str.lower().str.contains('moto z',
```

brand_viewed

```
na=False)==True),'moto_z']=1
df.loc[(df.brand_viewed=='MOTOROLA')&(df['model'].str.lower().str.contains('x play',
na=False)==True),'moto_x_play']=1
df.loc[(df.brand_viewed=='MOTOROLA')&(df['model'].str.lower().str.contains('moto x2',
na=False)==True),'moto_x2']=1
df.loc[(df.brand_viewed=='MOTOROLA')&(df['model'].str.lower().str.contains('moto x',
na=False)==True),'moto_x']=1
df.loc[(df.brand_viewed=='MOTOROLA')&(df['model'].str.lower().str.contains('moto e2',
na=False)==True),'moto_e2']=1
df.loc[(df.brand_viewed=='SONY')&(df['model'].str.lower().str.contains('xperia m4',
na=False)==True),'xperia_m4']=1
df.loc[(df.brand_viewed=='SONY')&(df['model'].str.lower().str.contains('xperia z',
na=False)==True),'xperia_z']=1
df.loc[(df.brand_viewed=='SONY')&(df['model'].str.lower().str.contains('xperia z2',
na=False)==True),'xperia_z2']=1
df.loc[(df.brand_viewed=='SONY')&(df['model'].str.lower().str.contains('xperia z3',
na=False)==True),'xperia_z3']=1
df.loc[(df.brand_viewed=='SONY')&(df['model'].str.lower().str.contains('xperia z4',
na=False)==True),'xperia_z4']=1
df.loc[(df.brand_viewed=='SONY')&(df['model'].str.lower().str.contains('xperia z5',
na=False)==True),'xperia_z5']=1
```


flags por modelos
checkout

```
# Modelo checkout:
df.loc[(df.checkout==1)&(df['model'].str.lower().str.contains('iphone se',
na=False)==True),'iphone_se_viewed_cho']=1
df.loc[(df.checkout==1)&(df['model'].str.lower().str.contains('iphone 5',
na=False)==True),'iphone_5_viewed_cho']=1
df.loc[(df.checkout==1)&(df['model'].str.lower().str.contains('iphone 6',
na=False)==True),'iphone_6_viewed_cho']=1
df.loc[(df.checkout==1)&(df['model'].str.lower().str.contains('iphone 7',
na=False)==True),'iphone_7_viewed_cho']=1
df.loc[(df.checkout==1)&(df['model'].str.lower().str.contains('galaxy s8 plus',
na=False)==True),'samsung_s8_plus_cho']=1
df.loc[(df.checkout==1)&(df['model'].str.lower().str.contains('galaxy s8',
na=False)==True),'samsung_s8_cho']=1
df.loc[(df.checkout==1)&(df['model'].str.lower().str.contains('galaxy s7',
na=False)==True),'samsung_s7_cho']=1
df.loc[(df.checkout==1)&(df['model'].str.lower().str.contains('galaxy s6',
na=False)==True),'samsung_s6_cho']=1
df.loc[(df.checkout==1)&(df['model'].str.lower().str.contains('galaxy s5',
na=False)==True),'samsung_s5_cho']=1
df.loc[(df.checkout==1)&(df['model'].str.lower().str.contains('galaxy s3',
na=False)==True),'samsung_s5_cho']=1
df.loc[(df.checkout==1)&(df['model'].str.lower().str.contains('galaxy s7 edge',
na=False)==True),'samsung_s7_viewed_cho']=1
df.loc[(df.checkout==1)&(df['model'].str.lower().str.contains('galaxy s6 edge',
na=False)==True),'samsung_s6_viewed_cho']=1
df.loc[(df.checkout==1)&(df['model'].str.lower().str.contains('galaxy note',
na=False)==True),'samsung_note_cho']=1
df.loc[(df.checkout==1)&(df['model'].str.lower().str.contains('galaxy note 8',
na=False)==True),'samsung_note_8_cho']=1
df.loc[(df.checkout==1)&(df['model'].str.lower().str.contains('galaxy a7',
na=False)==True),'samsung_a7_cho']=1
df.loc[(df.checkout==1)&(df['model'].str.lower().str.contains('galaxy a8',
na=False)==True),'samsung_a8_cho']=1
df.loc[(df.checkout==1)&(df['model'].str.lower().str.contains('galaxy a5',
na=False)==True),'samsung_a5_cho']=1
df.loc[(df.checkout==1)&(df['model'].str.lower().str.contains('galaxy j7',
na=False)==True),'samsung_j7_cho']=1
df.loc[(df.checkout==1)&(df['model'].str.lower().str.contains('galaxy j5',
na=False)==True),'samsung_j5_cho']=1
df.loc[(df.checkout==1)&(df['model'].str.lower().str.contains('galaxy j4',
na=False)==True),'samsung_j4_cho']=1
df.loc[(df.checkout==1)&(df['model'].str.lower().str.contains('galaxy j3',
na=False)==True),'samsung_j3_cho']=1
df.loc[(df.checkout==1)&(df['model'].str.lower().str.contains('galaxy j2',
na=False)==True),'samsung_j2_cho']=1
df.loc[(df.checkout==1)&(df['model'].str.lower().str.contains('galaxy j1',
na=False)==True),'samsung_j1_cho']=1
df.loc[(df.checkout==1)&(df['model'].str.lower().str.contains('moto g5',
na=False)==True),'moto_g5_cho']=1
df.loc[(df.checkout==1)&(df['model'].str.lower().str.contains('moto g4',
na=False)==True),'moto_g4_cho']=1
df.loc[(df.checkout==1)&(df['model'].str.lower().str.contains('moto g3',
na=False)==True),'moto_g3_cho']=1
df.loc[(df.checkout==1)&(df['model'].str.lower().str.contains('moto g2',
na=False)==True),'moto_g2_cho']=1
df.loc[(df.checkout==1)&(df['model'].str.lower().str.contains('moto g',
na=False)==True),'moto_g_cho']=1
df.loc[(df.checkout==1)&(df['model'].str.lower().str.contains('moto z',
na=False)==True),'moto_z_cho']=1
```

checkout

```
df.loc[(df.checkout==1)&(df['model'].str.lower().str.contains('x play',
na=False)==True),'moto_x_play_cho']=1
df.loc[(df.checkout==1)&(df['model'].str.lower().str.contains('moto x2',
na=False)==True),'moto_x2_cho']=1
df.loc[(df.checkout==1)&(df['model'].str.lower().str.contains('moto x',
na=False)==True),'moto_x_cho']=1
df.loc[(df.checkout==1)&(df['model'].str.lower().str.contains('moto e2',
na=False)==True),'moto_e2_cho']=1
df.loc[(df.checkout==1)&(df['model'].str.lower().str.contains('xperia m4',
na=False)==True),'xperia_m4_cho']=1
df.loc[(df.checkout==1)&(df['model'].str.lower().str.contains('xperia z',
na=False)==True),'xperia_z_cho']=1
df.loc[(df.checkout==1)&(df['model'].str.lower().str.contains('xperia z2',
na=False)==True),'xperia_z2_cho']=1
df.loc[(df.checkout==1)&(df['model'].str.lower().str.contains('xperia z3',
na=False)==True),'xperia_z3_cho']=1
df.loc[(df.checkout==1)&(df['model'].str.lower().str.contains('xperia z4',
na=False)==True),'xperia_z4_cho']=1
df.loc[(df.checkout==1)&(df['model'].str.lower().str.contains('xperia z5',
na=False)==True),'xperia_z5_cho']=1
```

flags por modelos
conversion

```
# Modelo conversion:
df.loc[(df.conversion==1)&(df['model'].str.lower().str.contains('iphone se',
na=False)==True),'iphone_se_viewed_conv']=1
df.loc[(df.conversion==1)&(df['model'].str.lower().str.contains('iphone 5',
na=False)==True),'iphone_5_viewed_conv']=1
df.loc[(df.conversion==1)&(df['model'].str.lower().str.contains('iphone 6',
na=False)==True),'iphone_6_viewed_conv']=1
df.loc[(df.conversion==1)&(df['model'].str.lower().str.contains('iphone 7',
na=False)==True),'iphone_7_viewed_conv']=1
df.loc[(df.conversion==1)&(df['model'].str.lower().str.contains('galaxy s8 plus',
na=False)==True),'samsung_s8_plus_conv']=1
df.loc[(df.conversion==1)&(df['model'].str.lower().str.contains('galaxy s8',
na=False)==True),'samsung_s8_conv']=1
df.loc[(df.conversion==1)&(df['model'].str.lower().str.contains('galaxy s7',
na=False)==True),'samsung_s7_conv']=1
df.loc[(df.conversion==1)&(df['model'].str.lower().str.contains('galaxy s6',
na=False)==True),'samsung_s6_conv']=1
df.loc[(df.conversion==1)&(df['model'].str.lower().str.contains('galaxy s5',
na=False)==True),'samsung_s5_conv']=1
df.loc[(df.conversion==1)&(df['model'].str.lower().str.contains('galaxy s3',
na=False)==True),'samsung_s5_conv']=1
df.loc[(df.conversion==1)&(df['model'].str.lower().str.contains('galaxy s7 edge',
na=False)==True),'samsung_s7_viewed_conv']=1
df.loc[(df.conversion==1)&(df['model'].str.lower().str.contains('galaxy s6 edge',
na=False)==True),'samsung_s6_viewed_conv']=1
df.loc[(df.conversion==1)&(df['model'].str.lower().str.contains('galaxy note',
na=False)==True),'samsung_note_conv']=1
df.loc[(df.conversion==1)&(df['model'].str.lower().str.contains('galaxy note 8',
na=False)==True),'samsung_note_8_conv']=1
df.loc[(df.conversion==1)&(df['model'].str.lower().str.contains('galaxy a7',
na=False)==True),'samsung_a7_conv']=1
df.loc[(df.conversion==1)&(df['model'].str.lower().str.contains('galaxy a8',
na=False)==True),'samsung_a8_conv']=1
df.loc[(df.conversion==1)&(df['model'].str.lower().str.contains('galaxy a5',
na=False)==True),'samsung_a5_conv']=1
df.loc[(df.conversion==1)&(df['model'].str.lower().str.contains('galaxy j7',
na=False)==True),'samsung_j7_conv']=1
df.loc[(df.conversion==1)&(df['model'].str.lower().str.contains('galaxy j5',
na=False)==True),'samsung_j5_conv']=1
df.loc[(df.conversion==1)&(df['model'].str.lower().str.contains('galaxy j4',
na=False)==True),'samsung_j4_conv']=1
df.loc[(df.conversion==1)&(df['model'].str.lower().str.contains('galaxy j3',
na=False)==True),'samsung_j3_conv']=1
df.loc[(df.conversion==1)&(df['model'].str.lower().str.contains('galaxy j2',
na=False)==True),'samsung_j2_conv']=1
df.loc[(df.conversion==1)&(df['model'].str.lower().str.contains('galaxy j1',
na=False)==True),'samsung_j1_conv']=1
df.loc[(df.conversion==1)&(df['model'].str.lower().str.contains('moto g5',
na=False)==True),'moto_g5_conv']=1
df.loc[(df.conversion==1)&(df['model'].str.lower().str.contains('moto g4',
na=False)==True),'moto_g4_conv']=1
df.loc[(df.conversion==1)&(df['model'].str.lower().str.contains('moto g3',
na=False)==True),'moto_g3_conv']=1
df.loc[(df.conversion==1)&(df['model'].str.lower().str.contains('moto g2',
na=False)==True),'moto_g2_conv']=1
df.loc[(df.conversion==1)&(df['model'].str.lower().str.contains('moto g',
na=False)==True),'moto_g_conv']=1
df.loc[(df.conversion==1)&(df['model'].str.lower().str.contains('moto z',
na=False)==True),'moto_z_conv']=1
```

conversion

	<pre> df.loc[(df.conversion==1)&(df['model'].str.lower().str.contains('x play', na=False)==True),'moto_x_play_conv']=1 df.loc[(df.conversion==1)&(df['model'].str.lower().str.contains('moto x2', na=False)==True),'moto_x2_conv']=1 df.loc[(df.conversion==1)&(df['model'].str.lower().str.contains('moto x', na=False)==True),'moto_x_conv']=1 df.loc[(df.conversion==1)&(df['model'].str.lower().str.contains('moto e2', na=False)==True),'moto_e2_conv']=1 df.loc[(df.conversion==1)&(df['model'].str.lower().str.contains('xperia m4', na=False)==True),'xperia_m4_conv']=1 df.loc[(df.conversion==1)&(df['model'].str.lower().str.contains('xperia z', na=False)==True),'xperia_z_conv']=1 df.loc[(df.conversion==1)&(df['model'].str.lower().str.contains('xperia z2', na=False)==True),'xperia_z2_conv']=1 df.loc[(df.conversion==1)&(df['model'].str.lower().str.contains('xperia z3', na=False)==True),'xperia_z3_conv']=1 df.loc[(df.conversion==1)&(df['model'].str.lower().str.contains('xperia z4', na=False)==True),'xperia_z4_conv']=1 df.loc[(df.conversion==1)&(df['model'].str.lower().str.contains('xperia z5', na=False)==True),'xperia_z5_conv']=1 </pre>	
flags de condition	<pre> # condition: df.loc[(df['condition'].str.lower().str.contains('bom', na=False)==True),'condition_bom']=1 df.loc[(df['condition'].str.lower().str.contains('excelente', na=False)==True),'condition_excelente']=1 df.loc[(df['condition'].str.lower().str.contains('muito', na=False)==True),'condition_muito_bom']=1 df.loc[(df['condition'].str.lower().str.contains('touch', na=False)==True),'condition_touch']=1 df.loc[(df['condition'].str.lower().str.contains('novo', na=False)==True),'condition_novo']=1 </pre>	condition
flags checkout vs condition	<pre> df.loc[(df.checkout==1)&(df['condition'].str.lower().str.contains('bom', na=False)==True),'condition_bom_cho']=1 df.loc[(df.checkout==1)&(df['condition'].str.lower().str.contains('excelente', na=False)==True),'condition_excelente_cho']=1 df.loc[(df.checkout==1)&(df['condition'].str.lower().str.contains('muito', na=False)==True),'condition_muito_bom_cho']=1 df.loc[(df.checkout==1)&(df['condition'].str.lower().str.contains('touch', na=False)==True),'condition_touch_cho']=1 df.loc[(df.checkout==1)&(df['condition'].str.lower().str.contains('novo', na=False)==True),'condition_novo_cho']=1 </pre>	condition, checkout

flags conversion vs condition	df.loc[(df.conversion==1)&(df['condition'].str.lower().str.contains('bom', na=False)==True), 'condition_bom_conv']=1 df.loc[(df.conversion==1)&(df['condition'].str.lower().str.contains('excelente', na=False)==True), 'condition_excelente_conv']=1 df.loc[(df.conversion==1)&(df['condition'].str.lower().str.contains('muito', na=False)==True), 'condition_muito_bom_conv']=1 df.loc[(df.conversion==1)&(df['condition'].str.lower().str.contains('touch', na=False)==True), 'condition_touch_conv']=1 df.loc[(df.conversion==1)&(df['condition'].str.lower().str.contains('novo', na=False)==True), 'condition_novo_conv']=1	condition, conversion
flags de search_engine_hit	# Search engine hit: df.loc[(df.search_engine_hit==1)&(df['search_engine'].str.lower().str.contains('google', na=False)==True), 'searched_google']=1 df.loc[(df.search_engine_hit==1)&(df['search_engine'].str.lower().str.contains('bing', na=False)==True), 'searched_bing']=1 df.loc[(df.search_engine_hit==1)&(df['search_engine'].str.lower().str.contains('yahoo', na=False)==True), 'searched_yahoo']=1	search_engine_hit
flags de marca	df.loc[(df['model'].str.lower().str.contains('iphone', na=False)==True), 'iphone_viewed']=1 df.loc[(df['model'].str.lower().str.contains('samsung', na=False)==True), 'samsung_viewed']=1 df.loc[(df['model'].str.lower().str.contains('motorola', na=False)==True), 'motorola_viewed']=1 df.loc[(df['model'].str.lower().str.contains('sony', na=False)==True), 'sony_viewed']=1 df.loc[(df['model'].str.lower().str.contains('lg', na=False)==True), 'lg_viewed']=1 df.loc[(df['model'].str.lower().str.contains('asus', na=False)==True), 'asus_viewed']=1 df.loc[(df['model'].str.lower().str.contains('ipad', na=False)==True), 'ipad_viewed']=1	model
flags de storage	df.loc[df['storage']=='8GB', 'storage_8']=1 df.loc[df['storage']=='16GB', 'storage_16']=1 df.loc[df['storage']=='32GB', 'storage_32']=1 df.loc[df['storage']=='64GB', 'storage_64']=1 df.loc[df['storage']=='128GB', 'storage_128']=1 df.loc[df['storage']=='4GB', 'storage_4']=1 df.loc[df['storage']=='256GB', 'storage_256']=1 df.loc[df['storage']=='512MB', 'storage_512MB']=1	storage
flags de visitas vs nombre mes	df.loc[df.event=='visited site', 'visit']=1 df.loc[(df.event=='visited site')&(df.nombre_mes=='MAYO'), 'visit_mayo']=1 df.loc[(df.event=='visited site')&(df.nombre_mes=='ABRIL'), 'visit_abril']=1 df.loc[(df.event=='visited site')&(df.nombre_mes=='MARZO'), 'visit_marzo']=1 df.loc[(df.event=='visited site')&(df.nombre_mes=='FEBRERO'), 'visit_febrero']=1 df.loc[(df.event=='visited site')&(df.nombre_mes=='ENERO'), 'visit_enero']=1	event, nombre_mes
flags de conversion vs nombre mes	df.loc[df.event=='conversion', 'conversion']=1 df.loc[(df.event=='conversion')&(df.nombre_mes=='MAYO'), 'conv_mayo']=1 df.loc[(df.event=='conversion')&(df.nombre_mes=='ABRIL'), 'conv_abril']=1 df.loc[(df.event=='conversion')&(df.nombre_mes=='MARZO'), 'conv_marzo']=1 df.loc[(df.event=='conversion')&(df.nombre_mes=='FEBRERO'), 'conv_febrero']=1 df.loc[(df.event=='conversion')&(df.nombre_mes=='ENERO'), 'conv_enero']=1	event, conversion
busco en el buscador	df.loc[df.search_term.notnull(), 'busqueda_term']=1	search_term
flags de operating_system_version	df.loc[(df['operating_system_version'].str.lower().str.contains('windows', na=False)==True), 'sist_oper_windows']=1 df.loc[(df['operating_system_version'].str.lower().str.contains('android', na=False)==True), 'sist_oper_android']=1 df.loc[(df['operating_system_version'].str.lower().str.contains('ios', na=False)==True), 'sist_oper_ios']=1	operating_system_version

	df.loc[(df['operating_system_version'].str.lower()).str.contains('mac', na=False)==True), 'sist_oper_mac']=1 df.loc[(df['operating_system_version'].str.lower()).str.contains('linux', na=False)==True), 'sist_oper_linux']=1	
flags de color	df.loc[(df['color'].str.lower()).str.contains('preto', na=False)==True), 'col_preto']=1 df.loc[(df['color'].str.lower()).str.contains('dourado', na=False)==True), 'col_dourado']=1 df.loc[(df['color'].str.lower()).str.contains('cinza', na=False)==True), 'col_cinza']=1 df.loc[(df['color'].str.lower()).str.contains('branco', na=False)==True), 'col_branco']=1 df.loc[(df['color'].str.lower()).str.contains('prateado', na=False)==True), 'col_prateado']=1 df.loc[(df['color'].str.lower()).str.contains('rosa', na=False)==True), 'col_rosa']=1	color
flags de campaign_source	df.loc[(df['campaign_source'].str.lower()).str.contains('google', na=False)==True), 'camp_google']=1 df.loc[(df['campaign_source'].str.lower()).str.contains('criteo', na=False)==True), 'camp_criteo']=1 df.loc[(df['campaign_source'].str.lower()).str.contains('rtbhouse', na=False)==True), 'camp_rtbhouse']=1 df.loc[(df['campaign_source'].str.lower()).str.contains('zanox', na=False)==True), 'camp_zanox']=1	campaign_source
flags de checkout por mes	# Flags de checkout por mes df.loc[(df.event=='checkout') & (df.nombre_mes=='MAYO'), 'CHO_MAYO']=1 df.loc[(df.event=='checkout') & (df.nombre_mes=='ABRIL'), 'CHO_ABRIL']=1 df.loc[(df.event=='checkout') & (df.nombre_mes=='MARZO'), 'CHO_MARZO']=1 df.loc[(df.event=='checkout') & (df.nombre_mes=='FEBRERO'), 'CHO_FEBRERO']=1 df.loc[(df.event=='checkout') & (df.nombre_mes=='ENERO'), 'CHO_ENERO']=1	checkout, nombre mes
flags de channel	df.loc[(df['channel'].str.lower()).str.contains('paid', na=False)==True), 'chan_paid']=1 df.loc[(df['channel'].str.lower()).str.contains('direct', na=False)==True), 'chan_direct']=1 df.loc[(df['channel'].str.lower()).str.contains('referral', na=False)==True), 'chan_referral']=1 df.loc[(df['channel'].str.lower()).str.contains('organic', na=False)==True), 'chan_organic']=1	channel
flags de new vs returning	df.loc[(df['new_vs_returning'].str.lower()).str.contains('returning', na=False)==True), 'nvsv_return']=1 df.loc[(df['new_vs_returning'].str.lower()).str.contains('new', na=False)==True), 'nvsv_new']=1	new_vs_returning
flags de ciudad	df.loc[(df['city'].str.lower()).str.contains('são paulo', na=False)==True), 'city_saopaulo']=1 df.loc[(df['city'].str.lower()).str.contains('rio de janeiro', na=False)==True), 'city_rio']=1 df.loc[(df['city'].str.lower()).str.contains('belo horizonte', na=False)==True), 'city_belo']=1 df.loc[(df['city'].str.lower()).str.contains('salvador', na=False)==True), 'city_salv']=1	city
flags de device_type	df.loc[(df['device_type'].str.lower()).str.contains('smartphone', na=False)==True), 'disp_smart']=1 df.loc[(df['device_type'].str.lower()).str.contains('computer', na=False)==True), 'disp_comp']=1 df.loc[(df['device_type'].str.lower()).str.contains('tablet', na=False)==True), 'disp_tablet']=1	device_type
flags de resolution	df.loc[(df['screen_resolution'].str.lower()).str.contains('360x640', na=False)==True), 'res_360x640']=1 df.loc[(df['screen_resolution'].str.lower()).str.contains('1366x768', na=False)==True), 'res_1366x768']=1 df.loc[(df['screen_resolution'].str.lower()).str.contains('1920x1080', na=False)==True), 'res_1920x1080']=1	resolution

variables de promedio, max	<pre> # Columna timestamp del registro anterior df['timestamp_anterior']=df['timestamp2'].shift() # Armo el registro anterior de persona df['persona_anterior']=df['person'].shift() # Calculo la diferencia de tiempo df.loc[df.event!='visited sit','time_diff']=df.timestamp2-df.timestamp_anterior # Convierto la diferencia a minutos df['tiempo_conv_mins'] = df['time_diff'].dt.total_seconds().div(60) df['tiempo_conv_mins'] = df[df.person==df.persona_anterior].tiempo_conv_mins # Creo dos datasets ya agrupados con el promedio y el maximo por persona. Después los joeo: df_mean_time=df.groupby(['person']).agg({'tiempo_conv_mins':'mean'}) df_max_time=df.groupby(['person']).agg({'tiempo_conv_mins':'max'}) </pre>	timestamp
flags de url	<pre> ## Flag por URL #Flag de Url por marca df.loc[(df['url'].str.lower().str.contains('comprar', na=False)==True),'url_comprar']=1 df.loc[(df['url'].str.lower().str.contains('vender', na=False)==True),'url_vender']=1 df.loc[(df['url'].str.lower().str.contains('iphone', na=False)==True),'url_iphone']=1 df.loc[(df['url'].str.lower().str.contains('samsung', na=False)==True),'url_samsung']=1 df.loc[(df['url'].str.lower().str.contains('motorola', na=False)==True),'url_motorola']=1 df.loc[(df['url'].str.lower().str.contains('lg', na=False)==True),'url_lg']=1 df.loc[(df['url'].str.lower().str.contains('ipad', na=False)==True),'url_ipad']=1 df.loc[(df['url'].str.lower().str.contains('asus', na=False)==True),'url_asus']=1 df.loc[(df['url'].str.lower().str.contains('apple', na=False)==True),'url_apple']=1 df.loc[(df['url'].str.lower().str.contains('sony', na=False)==True),'url_sony']=1 df.loc[(df['url'].str.lower().str.contains('quantum', na=False)==True),'url_quantum']=1 </pre>	url
semanas de mayo	<pre> # semanas de mayo: df.loc[(df['day']<=7) & (df['month']==5),'Primera_Semana']=1 df.loc[(df['day']>7) & (df['day']<=15) & (df['month']==5) ,'Segunda_Semana']=1 df.loc[(df['day']>15) & (df['day']<=21) & (df['month']==5),'Tercera_Semana']=1 df.loc[(df['day']>21) & (df['day']<=28) & (df['month']==5),'Cuarta_Semana']=1 df.loc[(df['day']>28) & (df['month']==5),'Fin_de_Mes']=1 </pre>	month

semanas y productos	<pre> df.loc[(df.Primer_Semana==1)&(df['model'].str.lower().str.contains('iphone', na=False)==True),'iphone_1eraSem']=1 df.loc[(df.Segunda_Semana==1)&(df['model'].str.lower().str.contains('iphone', na=False)==True),'iphone_2daSem']=1 df.loc[(df.Tercera_Semana==1)&(df['model'].str.lower().str.contains('iphone', na=False)==True),'iphone_3eraSem']=1 df.loc[(df.Cuarta_Semana==1)&(df['model'].str.lower().str.contains('iphone', na=False)==True),'iphone_4taSem']=1 df.loc[(df.Fin_de_Mes==1)&(df['model'].str.lower().str.contains('iphone', na=False)==True),'iphone_FindeMes']=1 df.loc[(df.Primer_Semana==1)&(df['model'].str.lower().str.contains('samsung', na=False)==True),'samsung_1eraSem']=1 df.loc[(df.Segunda_Semana==1)&(df['model'].str.lower().str.contains('samsung', na=False)==True),'samsung_2daSem']=1 df.loc[(df.Tercera_Semana==1)&(df['model'].str.lower().str.contains('samsung', na=False)==True),'samsung_3eraSem']=1 df.loc[(df.Cuarta_Semana==1)&(df['model'].str.lower().str.contains('samsung', na=False)==True),'samsung_4taSem']=1 df.loc[(df.Fin_de_Mes==1)&(df['model'].str.lower().str.contains('samsung', na=False)==True),'samsung_FindeMes']=1 df.loc[(df.Primer_Semana==1)&(df['model'].str.lower().str.contains('motorola', na=False)==True),'motorola_1eraSem']=1 df.loc[(df.Segunda_Semana==1)&(df['model'].str.lower().str.contains('motorola', na=False)==True),'motorola_2daSem']=1 df.loc[(df.Tercera_Semana==1)&(df['model'].str.lower().str.contains('motorola', na=False)==True),'motorola_3eraSem']=1 df.loc[(df.Cuarta_Semana==1)&(df['model'].str.lower().str.contains('motorola', na=False)==True),'motorola_4taSem']=1 df.loc[(df.Fin_de_Mes==1)&(df['model'].str.lower().str.contains('motorola', na=False)==True),'motorola_FindeMes']=1 df.loc[(df.Primer_Semana==1)&(df['model'].str.lower().str.contains('', na=False)==True),'motorola_1eraSem']=1 df.loc[(df.Segunda_Semana==1)&(df['model'].str.lower().str.contains('motorola', na=False)==True),'motorola_2daSem']=1 df.loc[(df.Tercera_Semana==1)&(df['model'].str.lower().str.contains('motorola', na=False)==True),'motorola_3eraSem']=1 df.loc[(df.Cuarta_Semana==1)&(df['model'].str.lower().str.contains('motorola', na=False)==True),'motorola_4taSem']=1 df.loc[(df.Fin_de_Mes==1)&(df['model'].str.lower().str.contains('motorola', na=False)==True),'motorola_FindeMes']=1 df.loc[(df.Fin_de_Mes==1)&(df['model'].str.lower().str.contains('iphone se', na=False)==True),'Iphonese_FindeMes']=1 df.loc[(df.Fin_de_Mes==1)&(df['model'].str.lower().str.contains('iphone 5', na=False)==True),'Iphone5_FindeMes']=1 df.loc[(df.Fin_de_Mes==1)&(df['model'].str.lower().str.contains('iphone 6', na=False)==True),'Iphone6_FindeMes']=1 df.loc[(df.Fin_de_Mes==1)&(df['model'].str.lower().str.contains('iphone 7', na=False)==True),'Iphone7_FindeMes']=1 df.loc[(df.Fin_de_Mes==1)&(df['model'].str.lower().str.contains('galaxy s8 plus', na=False)==True),'galaxy_s8_plus_FindeMes']=1 df.loc[(df.Fin_de_Mes==1)&(df['model'].str.lower().str.contains('galaxy s8', na=False)==True),'galaxy_s8_FindeMes']=1 df.loc[(df.Fin_de_Mes==1)&(df['model'].str.lower().str.contains('galaxy s7', na=False)==True),'galaxy_s7_FindeMes']=1 </pre>	semanas, produtos
---------------------	--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	-------------------


```

df.loc[(df.Fin_de_Mes==1)&(df['model'].str.lower().str.contains('galaxy s6',
na=False)==True), 'galaxy_s6_FindeMes']=1
df.loc[(df.Fin_de_Mes==1)&(df['model'].str.lower().str.contains('galaxy s5',
na=False)==True), 'galaxy_s5_FindeMes']=1
df.loc[(df.Fin_de_Mes==1)&(df['model'].str.lower().str.contains('galaxy s3',
na=False)==True), 'galaxy_s3_FindeMes']=1
df.loc[(df.Fin_de_Mes==1)&(df['model'].str.lower().str.contains('galaxy s7 edge',
na=False)==True), 'galaxy_s7_edge_FindeMes']=1
df.loc[(df.Fin_de_Mes==1)&(df['model'].str.lower().str.contains('galaxy s6 edge',
na=False)==True), 'galaxy_s6_edge_FindeMes']=1
df.loc[(df.Fin_de_Mes==1)&(df['model'].str.lower().str.contains('galaxy note',
na=False)==True), 'galaxy_note_FindeMes']=1
df.loc[(df.Fin_de_Mes==1)&(df['model'].str.lower().str.contains('galaxy a7',
na=False)==True), 'galaxy_a7_FindeMes']=1
df.loc[(df.Fin_de_Mes==1)&(df['model'].str.lower().str.contains('galaxy a8',
na=False)==True), 'galaxy_a8_FindeMes']=1
df.loc[(df.Fin_de_Mes==1)&(df['model'].str.lower().str.contains('galaxy a5',
na=False)==True), 'galaxy_a5_FindeMes']=1
df.loc[(df.Fin_de_Mes==1)&(df['model'].str.lower().str.contains('galaxy j7',
na=False)==True), 'galaxy_j7_FindeMes']=1
df.loc[(df.Fin_de_Mes==1)&(df['model'].str.lower().str.contains('galaxy j5',
na=False)==True), 'galaxy_j5_FindeMes']=1
df.loc[(df.Fin_de_Mes==1)&(df['model'].str.lower().str.contains('galaxy j4',
na=False)==True), 'galaxy_j4_FindeMes']=1
df.loc[(df.Fin_de_Mes==1)&(df['model'].str.lower().str.contains('galaxy j3',
na=False)==True), 'galaxy_j3_FindeMes']=1
df.loc[(df.Fin_de_Mes==1)&(df['model'].str.lower().str.contains('galaxy j2',
na=False)==True), 'galaxy_j2_FindeMes']=1
df.loc[(df.Fin_de_Mes==1)&(df['model'].str.lower().str.contains('galaxy j1',
na=False)==True), 'galaxy_j1_FindeMes']=1
df.loc[(df.Fin_de_Mes==1)&(df['model'].str.lower().str.contains('moto g5',
na=False)==True), 'moto_g5_FindeMes']=1
df.loc[(df.Fin_de_Mes==1)&(df['model'].str.lower().str.contains('moto g4',
na=False)==True), 'moto_g4_FindeMes']=1
df.loc[(df.Fin_de_Mes==1)&(df['model'].str.lower().str.contains('moto g3',
na=False)==True), 'moto_g3_FindeMes']=1
df.loc[(df.Fin_de_Mes==1)&(df['model'].str.lower().str.contains('moto g2',
na=False)==True), 'moto_g2_FindeMes']=1
df.loc[(df.Fin_de_Mes==1)&(df['model'].str.lower().str.contains('moto g',
na=False)==True), 'moto_g_FindeMes']=1
df.loc[(df.Fin_de_Mes==1)&(df['model'].str.lower().str.contains('moto z',
na=False)==True), 'moto_z_FindeMes']=1
df.loc[(df.Fin_de_Mes==1)&(df['model'].str.lower().str.contains('x play',
na=False)==True), 'xplay_FindeMes']=1
df.loc[(df.Fin_de_Mes==1)&(df['model'].str.lower().str.contains('moto x2',
na=False)==True), 'moto_x2_FindeMes']=1
df.loc[(df.Fin_de_Mes==1)&(df['model'].str.lower().str.contains('moto x',
na=False)==True), 'moto_x_FindeMes']=1
df.loc[(df.Fin_de_Mes==1)&(df['model'].str.lower().str.contains('moto e2',
na=False)==True), 'moto_e2_FindeMes']=1
df.loc[(df.Fin_de_Mes==1)&(df['model'].str.lower().str.contains('xperia m4',
na=False)==True), 'xperia_m4_FindeMes']=1
df.loc[(df.Fin_de_Mes==1)&(df['model'].str.lower().str.contains('xperia z',
na=False)==True), 'xperia_z_FindeMes']=1
df.loc[(df.Fin_de_Mes==1)&(df['model'].str.lower().str.contains('xperia z2',
na=False)==True), 'xperia_z2_FindeMes']=1
df.loc[(df.Fin_de_Mes==1)&(df['model'].str.lower().str.contains('xperia z3',
na=False)==True), 'xperia_z3_FindeMes']=1
df.loc[(df.Fin_de_Mes==1)&(df['model'].str.lower().str.contains('xperia z4',

```

```

na=False)==True),'xperia_z4_FindeMes']=1
df.loc[(df.Fin_de_Mes==1)&(df['model'].str.lower().str.contains('xperia z5',
na=False)==True),'xperia_z5_FindeMes']=1

df.loc[(df.Primer_Semana==1)&(df['model'].str.lower().str.contains('iphone se',
na=False)==True),'Iphonese_1eraSemana']=1
df.loc[(df.Primer_Semana==1)&(df['model'].str.lower().str.contains('iphone 5',
na=False)==True),'Iphone5_1eraSemana']=1
df.loc[(df.Primer_Semana==1)&(df['model'].str.lower().str.contains('iphone 6',
na=False)==True),'Iphone6_1eraSemana']=1
df.loc[(df.Primer_Semana==1)&(df['model'].str.lower().str.contains('iphone 7',
na=False)==True),'Iphone7_1eraSemana']=1
df.loc[(df.Primer_Semana==1)&(df['model'].str.lower().str.contains('galaxy s8 plus',
na=False)==True),'galaxy_s8_plus_1eraSemana']=1
df.loc[(df.Primer_Semana==1)&(df['model'].str.lower().str.contains('galaxy s8',
na=False)==True),'galaxy_s8_1eraSemana']=1
df.loc[(df.Primer_Semana==1)&(df['model'].str.lower().str.contains('galaxy s7',
na=False)==True),'galaxy_s7_1eraSemana']=1
df.loc[(df.Primer_Semana==1)&(df['model'].str.lower().str.contains('galaxy s6',
na=False)==True),'galaxy_s6_1eraSemana']=1
df.loc[(df.Primer_Semana==1)&(df['model'].str.lower().str.contains('galaxy s5',
na=False)==True),'galaxy_s5_1eraSemana']=1
df.loc[(df.Primer_Semana==1)&(df['model'].str.lower().str.contains('galaxy s3',
na=False)==True),'galaxy_s3_1eraSemana']=1
df.loc[(df.Primer_Semana==1)&(df['model'].str.lower().str.contains('galaxy s7 edge',
na=False)==True),'galaxy_s7_edge_1eraSemana']=1
df.loc[(df.Primer_Semana==1)&(df['model'].str.lower().str.contains('galaxy s6 edge',
na=False)==True),'galaxy_s6_edge_1eraSemana']=1
df.loc[(df.Primer_Semana==1)&(df['model'].str.lower().str.contains('galaxy note',
na=False)==True),'galaxy_note_1eraSemana']=1
df.loc[(df.Primer_Semana==1)&(df['model'].str.lower().str.contains('galaxy a7',
na=False)==True),'galaxy_a7_1eraSemana']=1
df.loc[(df.Primer_Semana==1)&(df['model'].str.lower().str.contains('galaxy a8',
na=False)==True),'galaxy_a8_1eraSemana']=1
df.loc[(df.Primer_Semana==1)&(df['model'].str.lower().str.contains('galaxy a5',
na=False)==True),'galaxy_a5_1eraSemana']=1
df.loc[(df.Primer_Semana==1)&(df['model'].str.lower().str.contains('galaxy j7',
na=False)==True),'galaxy_j7_1eraSemana']=1
df.loc[(df.Primer_Semana==1)&(df['model'].str.lower().str.contains('galaxy j5',
na=False)==True),'galaxy_j5_1eraSemana']=1
df.loc[(df.Primer_Semana==1)&(df['model'].str.lower().str.contains('galaxy j4',
na=False)==True),'galaxy_j4_1eraSemana']=1
df.loc[(df.Primer_Semana==1)&(df['model'].str.lower().str.contains('galaxy j3',
na=False)==True),'galaxy_j3_1eraSemana']=1
df.loc[(df.Primer_Semana==1)&(df['model'].str.lower().str.contains('galaxy j2',
na=False)==True),'galaxy_j2_1eraSemana']=1
df.loc[(df.Primer_Semana==1)&(df['model'].str.lower().str.contains('galaxy j1',
na=False)==True),'galaxy_j1_1eraSemana']=1
df.loc[(df.Primer_Semana==1)&(df['model'].str.lower().str.contains('moto g5',
na=False)==True),'moto_g5_1eraSemana']=1
df.loc[(df.Primer_Semana==1)&(df['model'].str.lower().str.contains('moto g4',
na=False)==True),'moto_g4_1eraSemana']=1
df.loc[(df.Primer_Semana==1)&(df['model'].str.lower().str.contains('moto g3',
na=False)==True),'moto_g3_1eraSemana']=1
df.loc[(df.Primer_Semana==1)&(df['model'].str.lower().str.contains('moto g2',
na=False)==True),'moto_g2_1eraSemana']=1
df.loc[(df.Primer_Semana==1)&(df['model'].str.lower().str.contains('moto g',
na=False)==True),'moto_g_1eraSemana']=1
df.loc[(df.Primer_Semana==1)&(df['model'].str.lower().str.contains('moto z',

```

```

na=False)==True),'moto_z_1eraSemana']=1
df.loc[(df.Primer_Semana==1)&(df['model'].str.lower().str.contains('x play',
na=False)==True),'xplay_1eraSemana']=1
df.loc[(df.Primer_Semana==1)&(df['model'].str.lower().str.contains('moto x2',
na=False)==True),'moto_x2_1eraSemana']=1
df.loc[(df.Primer_Semana==1)&(df['model'].str.lower().str.contains('moto x',
na=False)==True),'moto_x_1eraSemana']=1
df.loc[(df.Primer_Semana==1)&(df['model'].str.lower().str.contains('moto e2',
na=False)==True),'moto_e2_1eraSemana']=1
df.loc[(df.Primer_Semana==1)&(df['model'].str.lower().str.contains('xperia m4',
na=False)==True),'xperia_m4_1eraSemana']=1
df.loc[(df.Primer_Semana==1)&(df['model'].str.lower().str.contains('xperia z',
na=False)==True),'xperia_z_1eraSemana']=1
df.loc[(df.Primer_Semana==1)&(df['model'].str.lower().str.contains('xperia z2',
na=False)==True),'xperia_z2_1eraSemana']=1
df.loc[(df.Primer_Semana==1)&(df['model'].str.lower().str.contains('xperia z3',
na=False)==True),'xperia_z3_1eraSemana']=1
df.loc[(df.Primer_Semana==1)&(df['model'].str.lower().str.contains('xperia z4',
na=False)==True),'xperia_z4_1eraSemana']=1
df.loc[(df.Primer_Semana==1)&(df['model'].str.lower().str.contains('xperia z5',
na=False)==True),'xperia_z5_1eraSemana']=1

df.loc[(df.Segunda_Semana==1)&(df['model'].str.lower().str.contains('iphone se',
na=False)==True),'Iphonese_2daSemana']=1
df.loc[(df.Segunda_Semana==1)&(df['model'].str.lower().str.contains('iphone 5',
na=False)==True),'Iphone5_2daSemana']=1
df.loc[(df.Segunda_Semana==1)&(df['model'].str.lower().str.contains('iphone 6',
na=False)==True),'Iphone6_2daSemana']=1
df.loc[(df.Segunda_Semana==1)&(df['model'].str.lower().str.contains('iphone 7',
na=False)==True),'Iphone7_2daSemana']=1
df.loc[(df.Segunda_Semana==1)&(df['model'].str.lower().str.contains('galaxy s8 plus',
na=False)==True),'galaxy_s8_plus_2daSemana']=1
df.loc[(df.Segunda_Semana==1)&(df['model'].str.lower().str.contains('galaxy s8',
na=False)==True),'galaxy_s8_2daSemana']=1
df.loc[(df.Segunda_Semana==1)&(df['model'].str.lower().str.contains('galaxy s7',
na=False)==True),'galaxy_s7_2daSemana']=1
df.loc[(df.Segunda_Semana==1)&(df['model'].str.lower().str.contains('galaxy s6',
na=False)==True),'galaxy_s6_2daSemana']=1
df.loc[(df.Segunda_Semana==1)&(df['model'].str.lower().str.contains('galaxy s5',
na=False)==True),'galaxy_s5_2daSemana']=1
df.loc[(df.Segunda_Semana==1)&(df['model'].str.lower().str.contains('galaxy s3',
na=False)==True),'galaxy_s3_2daSemana']=1
df.loc[(df.Segunda_Semana==1)&(df['model'].str.lower().str.contains('galaxy s7 edge',
na=False)==True),'galaxy_s7_edge_2daSemana']=1
df.loc[(df.Segunda_Semana==1)&(df['model'].str.lower().str.contains('galaxy s6 edge',
na=False)==True),'galaxy_s6_edge_2daSemana']=1
df.loc[(df.Segunda_Semana==1)&(df['model'].str.lower().str.contains('galaxy note',
na=False)==True),'galaxy_note_2daSemana']=1
df.loc[(df.Segunda_Semana==1)&(df['model'].str.lower().str.contains('galaxy a7',
na=False)==True),'galaxy_a7_2daSemana']=1
df.loc[(df.Segunda_Semana==1)&(df['model'].str.lower().str.contains('galaxy a8',
na=False)==True),'galaxy_a8_2daSemana']=1
df.loc[(df.Segunda_Semana==1)&(df['model'].str.lower().str.contains('galaxy a5',
na=False)==True),'galaxy_a5_2daSemana']=1
df.loc[(df.Segunda_Semana==1)&(df['model'].str.lower().str.contains('galaxy j7',
na=False)==True),'galaxy_j7_2daSemana']=1
df.loc[(df.Segunda_Semana==1)&(df['model'].str.lower().str.contains('galaxy j5',
na=False)==True),'galaxy_j5_2daSemana']=1
df.loc[(df.Segunda_Semana==1)&(df['model'].str.lower().str.contains('galaxy j4',

```

```

na=False)==True), 'galaxy_j4_2daSemana']=1
df.loc[(df.Segunda_Semana==1)&(df['model'].str.lower().str.contains('galaxy j3',
na=False)==True), 'galaxy_j3_2daSemana']=1
df.loc[(df.Segunda_Semana==1)&(df['model'].str.lower().str.contains('galaxy j2',
na=False)==True), 'galaxy_j2_2daSemana']=1
df.loc[(df.Segunda_Semana==1)&(df['model'].str.lower().str.contains('galaxy j1',
na=False)==True), 'galaxy_j1_2daSemana']=1
df.loc[(df.Segunda_Semana==1)&(df['model'].str.lower().str.contains('moto g5',
na=False)==True), 'moto_g5_2daSemana']=1
df.loc[(df.Segunda_Semana==1)&(df['model'].str.lower().str.contains('moto g4',
na=False)==True), 'moto_g4_2daSemana']=1
df.loc[(df.Segunda_Semana==1)&(df['model'].str.lower().str.contains('moto g3',
na=False)==True), 'moto_g3_2daSemana']=1
df.loc[(df.Segunda_Semana==1)&(df['model'].str.lower().str.contains('moto g2',
na=False)==True), 'moto_g2_2daSemana']=1
df.loc[(df.Segunda_Semana==1)&(df['model'].str.lower().str.contains('moto g',
na=False)==True), 'moto_g_2daSemana']=1
df.loc[(df.Segunda_Semana==1)&(df['model'].str.lower().str.contains('moto z',
na=False)==True), 'moto_z_2daSemana']=1
df.loc[(df.Segunda_Semana==1)&(df['model'].str.lower().str.contains('x play',
na=False)==True), 'xplay_2daSemana']=1
df.loc[(df.Segunda_Semana==1)&(df['model'].str.lower().str.contains('moto x2',
na=False)==True), 'moto_x2_2daSemana']=1
df.loc[(df.Segunda_Semana==1)&(df['model'].str.lower().str.contains('moto x',
na=False)==True), 'moto_x_2daSemana']=1
df.loc[(df.Segunda_Semana==1)&(df['model'].str.lower().str.contains('moto e2',
na=False)==True), 'moto_e2_2daSemana']=1
df.loc[(df.Segunda_Semana==1)&(df['model'].str.lower().str.contains('xperia m4',
na=False)==True), 'xperia_m4_2daSemana']=1
df.loc[(df.Segunda_Semana==1)&(df['model'].str.lower().str.contains('xperia z',
na=False)==True), 'xperia_z_2daSemana']=1
df.loc[(df.Segunda_Semana==1)&(df['model'].str.lower().str.contains('xperia z2',
na=False)==True), 'xperia_z2_2daSemana']=1
df.loc[(df.Segunda_Semana==1)&(df['model'].str.lower().str.contains('xperia z3',
na=False)==True), 'xperia_z3_2daSemana']=1
df.loc[(df.Segunda_Semana==1)&(df['model'].str.lower().str.contains('xperia z4',
na=False)==True), 'xperia_z4_2daSemana']=1
df.loc[(df.Segunda_Semana==1)&(df['model'].str.lower().str.contains('xperia z5',
na=False)==True), 'xperia_z5_2daSemana']=1

df.loc[(df.Tercera_Semana==1)&(df['model'].str.lower().str.contains('iphone se',
na=False)==True), 'Iphonese_3eraSemana']=1
df.loc[(df.Tercera_Semana==1)&(df['model'].str.lower().str.contains('iphone 5',
na=False)==True), 'Iphone5_3eraSemana']=1
df.loc[(df.Tercera_Semana==1)&(df['model'].str.lower().str.contains('iphone 6',
na=False)==True), 'Iphone6_3eraSemana']=1
df.loc[(df.Tercera_Semana==1)&(df['model'].str.lower().str.contains('iphone 7',
na=False)==True), 'Iphone7_3eraSemana']=1
df.loc[(df.Tercera_Semana==1)&(df['model'].str.lower().str.contains('galaxy s8 plus',
na=False)==True), 'galaxy_s8_plus_3eraSemana']=1
df.loc[(df.Tercera_Semana==1)&(df['model'].str.lower().str.contains('galaxy s8',
na=False)==True), 'galaxy_s8_3eraSemana']=1
df.loc[(df.Tercera_Semana==1)&(df['model'].str.lower().str.contains('galaxy s7',
na=False)==True), 'galaxy_s7_3eraSemana']=1
df.loc[(df.Tercera_Semana==1)&(df['model'].str.lower().str.contains('galaxy s6',
na=False)==True), 'galaxy_s6_3eraSemana']=1
df.loc[(df.Tercera_Semana==1)&(df['model'].str.lower().str.contains('galaxy s5',
na=False)==True), 'galaxy_s5_3eraSemana']=1
df.loc[(df.Tercera_Semana==1)&(df['model'].str.lower().str.contains('galaxy s3',

```

```

na=False)==True),'galaxy_s3_3eraSemana']=1
df.loc[(df.Tercera_Semana==1)&(df['model'].str.lower().str.contains('galaxy s7 edge',
na=False)==True),'galaxy_s7_edge_3eraSemana']=1
df.loc[(df.Tercera_Semana==1)&(df['model'].str.lower().str.contains('galaxy s6 edge',
na=False)==True),'galaxy_s6_edge_3eraSemana']=1
df.loc[(df.Tercera_Semana==1)&(df['model'].str.lower().str.contains('galaxy note',
na=False)==True),'galaxy_note_3eraSemana']=1
df.loc[(df.Tercera_Semana==1)&(df['model'].str.lower().str.contains('galaxy a7',
na=False)==True),'galaxy_a7_3eraSemana']=1
df.loc[(df.Tercera_Semana==1)&(df['model'].str.lower().str.contains('galaxy a8',
na=False)==True),'galaxy_a8_3eraSemana']=1
df.loc[(df.Tercera_Semana==1)&(df['model'].str.lower().str.contains('galaxy a5',
na=False)==True),'galaxy_a5_3eraSemana']=1
df.loc[(df.Tercera_Semana==1)&(df['model'].str.lower().str.contains('galaxy j7',
na=False)==True),'galaxy_j7_3eraSemana']=1
df.loc[(df.Tercera_Semana==1)&(df['model'].str.lower().str.contains('galaxy j5',
na=False)==True),'galaxy_j5_3eraSemana']=1
df.loc[(df.Tercera_Semana==1)&(df['model'].str.lower().str.contains('galaxy j4',
na=False)==True),'galaxy_j4_3eraSemana']=1
df.loc[(df.Tercera_Semana==1)&(df['model'].str.lower().str.contains('galaxy j3',
na=False)==True),'galaxy_j3_3eraSemana']=1
df.loc[(df.Tercera_Semana==1)&(df['model'].str.lower().str.contains('galaxy j2',
na=False)==True),'galaxy_j2_3eraSemana']=1
df.loc[(df.Tercera_Semana==1)&(df['model'].str.lower().str.contains('galaxy j1',
na=False)==True),'galaxy_j1_3eraSemana']=1
df.loc[(df.Tercera_Semana==1)&(df['model'].str.lower().str.contains('moto g5',
na=False)==True),'moto_g5_3eraSemana']=1
df.loc[(df.Tercera_Semana==1)&(df['model'].str.lower().str.contains('moto g4',
na=False)==True),'moto_g4_3eraSemana']=1
df.loc[(df.Tercera_Semana==1)&(df['model'].str.lower().str.contains('moto g3',
na=False)==True),'moto_g3_3eraSemana']=1
df.loc[(df.Tercera_Semana==1)&(df['model'].str.lower().str.contains('moto g2',
na=False)==True),'moto_g2_3eraSemana']=1
df.loc[(df.Tercera_Semana==1)&(df['model'].str.lower().str.contains('moto g',
na=False)==True),'moto_g_3eraSemana']=1
df.loc[(df.Tercera_Semana==1)&(df['model'].str.lower().str.contains('moto z',
na=False)==True),'moto_z_3eraSemana']=1
df.loc[(df.Tercera_Semana==1)&(df['model'].str.lower().str.contains('x play',
na=False)==True),'xplay_3eraSemana']=1
df.loc[(df.Tercera_Semana==1)&(df['model'].str.lower().str.contains('moto x2',
na=False)==True),'moto_x2_3eraSemana']=1
df.loc[(df.Tercera_Semana==1)&(df['model'].str.lower().str.contains('moto x',
na=False)==True),'moto_x_3eraSemana']=1
df.loc[(df.Tercera_Semana==1)&(df['model'].str.lower().str.contains('moto e2',
na=False)==True),'moto_e2_3eraSemana']=1
df.loc[(df.Tercera_Semana==1)&(df['model'].str.lower().str.contains('xperia m4',
na=False)==True),'xperia_m4_3eraSemana']=1
df.loc[(df.Tercera_Semana==1)&(df['model'].str.lower().str.contains('xperia z',
na=False)==True),'xperia_z_3eraSemana']=1
df.loc[(df.Tercera_Semana==1)&(df['model'].str.lower().str.contains('xperia z2',
na=False)==True),'xperia_z2_3eraSemana']=1
df.loc[(df.Tercera_Semana==1)&(df['model'].str.lower().str.contains('xperia z3',
na=False)==True),'xperia_z3_3eraSemana']=1
df.loc[(df.Tercera_Semana==1)&(df['model'].str.lower().str.contains('xperia z4',
na=False)==True),'xperia_z4_3eraSemana']=1
df.loc[(df.Tercera_Semana==1)&(df['model'].str.lower().str.contains('xperia z5',
na=False)==True),'xperia_z5_3eraSemana']=1

df.loc[(df.Cuarta_Semana==1)&(df['model'].str.lower().str.contains('iphone se',

```

```

na=False)==True),'Iphonese_4taSemana']=1
df.loc[(df.Cuarta_Semana==1)&(df['model'].str.lower()).str.contains('iphone 5',
na=False)==True),'Iphone5_4taSemana']=1
df.loc[(df.Cuarta_Semana==1)&(df['model'].str.lower()).str.contains('iphone 6',
na=False)==True),'Iphone6_4taSemana']=1
df.loc[(df.Cuarta_Semana==1)&(df['model'].str.lower()).str.contains('iphone 7',
na=False)==True),'Iphone7_4taSemana']=1
df.loc[(df.Cuarta_Semana==1)&(df['model'].str.lower()).str.contains('galaxy s8 plus',
na=False)==True),'galaxy_s8_plus_4taSemana']=1
df.loc[(df.Cuarta_Semana==1)&(df['model'].str.lower()).str.contains('galaxy s8',
na=False)==True),'galaxy_s8_4taSemana']=1
df.loc[(df.Cuarta_Semana==1)&(df['model'].str.lower()).str.contains('galaxy s7',
na=False)==True),'galaxy_s7_4taSemana']=1
df.loc[(df.Cuarta_Semana==1)&(df['model'].str.lower()).str.contains('galaxy s6',
na=False)==True),'galaxy_s6_4taSemana']=1
df.loc[(df.Cuarta_Semana==1)&(df['model'].str.lower()).str.contains('galaxy s5',
na=False)==True),'galaxy_s5_4taSemana']=1
df.loc[(df.Cuarta_Semana==1)&(df['model'].str.lower()).str.contains('galaxy s3',
na=False)==True),'galaxy_s3_4taSemana']=1
df.loc[(df.Cuarta_Semana==1)&(df['model'].str.lower()).str.contains('galaxy s7 edge',
na=False)==True),'galaxy_s7_edge_4taSemana']=1
df.loc[(df.Cuarta_Semana==1)&(df['model'].str.lower()).str.contains('galaxy s6 edge',
na=False)==True),'galaxy_s6_edge_4taSemana']=1
df.loc[(df.Cuarta_Semana==1)&(df['model'].str.lower()).str.contains('galaxy note',
na=False)==True),'galaxy_note_4taSemana']=1
df.loc[(df.Cuarta_Semana==1)&(df['model'].str.lower()).str.contains('galaxy a7',
na=False)==True),'galaxy_a7_4taSemana']=1
df.loc[(df.Cuarta_Semana==1)&(df['model'].str.lower()).str.contains('galaxy a8',
na=False)==True),'galaxy_a8_4taSemana']=1
df.loc[(df.Cuarta_Semana==1)&(df['model'].str.lower()).str.contains('galaxy a5',
na=False)==True),'galaxy_a5_4taSemana']=1
df.loc[(df.Cuarta_Semana==1)&(df['model'].str.lower()).str.contains('galaxy j7',
na=False)==True),'galaxy_j7_4taSemana']=1
df.loc[(df.Cuarta_Semana==1)&(df['model'].str.lower()).str.contains('galaxy j5',
na=False)==True),'galaxy_j5_4taSemana']=1
df.loc[(df.Cuarta_Semana==1)&(df['model'].str.lower()).str.contains('galaxy j4',
na=False)==True),'galaxy_j4_4taSemana']=1
df.loc[(df.Cuarta_Semana==1)&(df['model'].str.lower()).str.contains('galaxy j3',
na=False)==True),'galaxy_j3_4taSemana']=1
df.loc[(df.Cuarta_Semana==1)&(df['model'].str.lower()).str.contains('galaxy j2',
na=False)==True),'galaxy_j2_4taSemana']=1
df.loc[(df.Cuarta_Semana==1)&(df['model'].str.lower()).str.contains('galaxy j1',
na=False)==True),'galaxy_j1_4taSemana']=1
df.loc[(df.Cuarta_Semana==1)&(df['model'].str.lower()).str.contains('moto g5',
na=False)==True),'moto_g5_4taSemana']=1
df.loc[(df.Cuarta_Semana==1)&(df['model'].str.lower()).str.contains('moto g4',
na=False)==True),'moto_g4_4taSemana']=1
df.loc[(df.Cuarta_Semana==1)&(df['model'].str.lower()).str.contains('moto g3',
na=False)==True),'moto_g3_4taSemana']=1
df.loc[(df.Cuarta_Semana==1)&(df['model'].str.lower()).str.contains('moto g2',
na=False)==True),'moto_g2_4taSemana']=1
df.loc[(df.Cuarta_Semana==1)&(df['model'].str.lower()).str.contains('moto g',
na=False)==True),'moto_g_4taSemana']=1
df.loc[(df.Cuarta_Semana==1)&(df['model'].str.lower()).str.contains('moto z',
na=False)==True),'moto_z_4taSemana']=1
df.loc[(df.Cuarta_Semana==1)&(df['model'].str.lower()).str.contains('x play',
na=False)==True),'xplay_4taSemana']=1
df.loc[(df.Cuarta_Semana==1)&(df['model'].str.lower()).str.contains('moto x2',
na=False)==True),'moto_x2_4taSemana']=1

```

	<pre> df.loc[(df.Cuarta_Semana==1)&(df['model'].str.lower().str.contains('moto x', na=False)==True),'moto_x_4taSemana']=1 df.loc[(df.Cuarta_Semana==1)&(df['model'].str.lower().str.contains('moto e2', na=False)==True),'moto_e2_4taSemana']=1 df.loc[(df.Cuarta_Semana==1)&(df['model'].str.lower().str.contains('xperia m4', na=False)==True),'xperia_m4_4taSemana']=1 df.loc[(df.Cuarta_Semana==1)&(df['model'].str.lower().str.contains('xperia z', na=False)==True),'xperia_z_4taSemana']=1 df.loc[(df.Cuarta_Semana==1)&(df['model'].str.lower().str.contains('xperia z2', na=False)==True),'xperia_z2_4taSemana']=1 df.loc[(df.Cuarta_Semana==1)&(df['model'].str.lower().str.contains('xperia z3', na=False)==True),'xperia_z3_4taSemana']=1 df.loc[(df.Cuarta_Semana==1)&(df['model'].str.lower().str.contains('xperia z4', na=False)==True),'xperia_z4_4taSemana']=1 df.loc[(df.Cuarta_Semana==1)&(df['model'].str.lower().str.contains('xperia z5', na=False)==True),'xperia_z5_4taSemana']=1 </pre>	
url vs marca	<pre> #Flag de Url por marca df.loc[(df['url'].str.lower().str.contains('6s', na=False)==True),'url_6s']=1 df.loc[(df['url'].str.lower().str.contains('6', na=False)==True),'url_6']=1 df.loc[(df['url'].str.lower().str.contains('iphone-5s', na=False)==True),'url_iphone-5s']=1 df.loc[(df['url'].str.lower().str.contains('galaxy', na=False)==True),'url_galaxy']=1 df.loc[(df['url'].str.lower().str.contains('galaxy-j5', na=False)==True),'url_galaxy-j5']=1 df.loc[(df['url'].str.lower().str.contains('galaxy-s7', na=False)==True),'galaxy-s7']=1 df.loc[(df['url'].str.lower().str.contains('galaxy-s6', na=False)==True),'url_galaxy-s6']=1 df.loc[(df['url'].str.lower().str.contains('galaxy-j7', na=False)==True),'url_galaxy-j7']=1 df.loc[(df['url'].str.lower().str.contains('galaxy-s8', na=False)==True),'url_galaxy-s8']=1 </pre>	url, marca

Variables racionales en base al dataset agrupado:

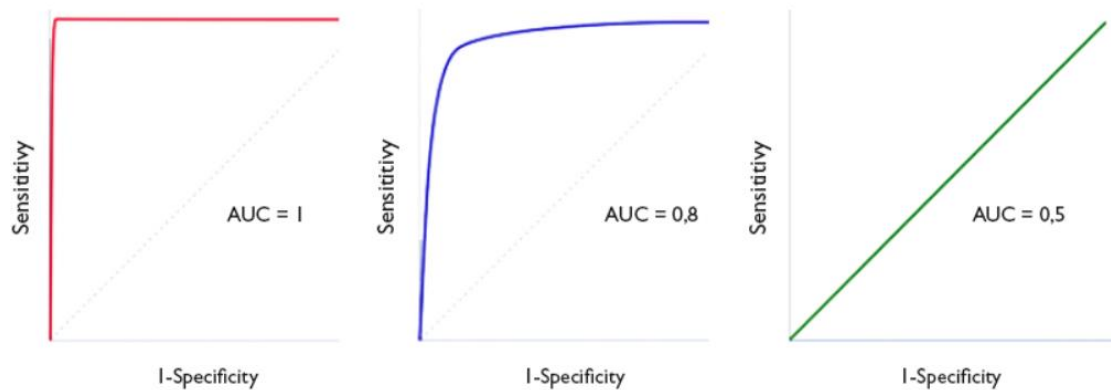
NOMBRE DE VARIABLE	CÓDIGO DE TRANSFORMACIÓN	VARIABLE O VARIABLES FUENTES
ratios de conversion, checkout y visit	$df_grouped['r_conv_visit'] = df_grouped['conversion'] / df_grouped['visit']$ $df_grouped['r_conv_cho'] = df_grouped['conversion'] / df_grouped['checkout']$ $df_grouped['r_cho_conv'] = df_grouped['checkout'] / df_grouped['conversion']$ $df_grouped['r_iphone_cho'] = df_grouped['checkout'] / df_grouped['iphone_viewed']$ $df_grouped['r_samsung_cho'] = df_grouped['checkout'] / df_grouped['samsung_viewed']$	conversion, checkout, visit
ratio conversion vs busqueda de terminos	$df_grouped['r_conv_busq_term'] = df_grouped['conversion'] / df_grouped['busqueda_term']$ $df_grouped['r_conv_may_busq_term'] = df_grouped['conv_mayo'] / df_grouped['busqueda_term']$	conversion, search_term
ratio conversion vs visita del mes	<p># ratio de conversion sobre visitas del mes</p> $df_grouped['r_vis_conv_may'] = df_grouped['conv_mayo'] / df_grouped['visit_mayo']$ $df_grouped['r_vis_conv_apr'] = df_grouped['conv_abril'] / df_grouped['visit_abril']$ $df_grouped['r_vis_conv_mar'] = df_grouped['conv_marzo'] / df_grouped['visit_marzo']$ $df_grouped['r_vis_conv_feb'] = df_grouped['conv_febrero'] / df_grouped['visit_febrero']$ $df_grouped['r_vis_conv_jan'] = df_grouped['conv_enero'] / df_grouped['visit_enero']$ <p># idem pero meses distintos:</p> $df_grouped['r_conv_may_vis_apr'] = df_grouped['conv_mayo'] / df_grouped['visit_abril']$ $df_grouped['r_conv_apr_vis_mar'] = df_grouped['conv_abril'] / df_grouped['visit_marzo']$ $df_grouped['r_conv_mar_vis_feb'] = df_grouped['conv_marzo'] / df_grouped['visit_febrero']$ <p>#df.loc[df.event=='checkout','checkout']=1</p> $df_grouped['r_cho_vis_may'] = df_grouped['checkout'] / df_grouped['visit_mayo']$	conversion, visit
ratio sistema operativo, checkout, conversion	$df_grouped['r_cho_windows'] = df_grouped['checkout'] / df_grouped['sist_oper_windows']$ $df_grouped['r_conv_windows'] = df_grouped['conversion'] / df_grouped['sist_oper_windows']$ $df_grouped['r_conv_may_windows'] = df_grouped['conv_mayo'] / df_grouped['sist_oper_windows']$	checkout, sistema operativo
ratio viewed producto, conversion, condition	$df_grouped['r_vis_may_viewprod'] = df_grouped['visit_mayo'] / df_grouped['viewprod']$ $df_grouped['r_conv_viewprod'] = df_grouped['conversion'] / df_grouped['viewprod']$ $df_grouped['r_condbom_viewprod'] = df_grouped['condition_bom'] / df_grouped['viewprod']$	conversion, condition, viewproduct
ratio conversion vs checkout	$df_grouped['conv_cho_vis_may'] = (df_grouped['conversion'] + df_grouped['checkout']) / df_grouped['visit_mayo']$ $df_grouped['vis_may_cho_conv'] = (df_grouped['visit_mayo'] + df_grouped['checkout']) / df_grouped['conversion']$	conversion, checkout
promedios por mes	$df_grouped['prom_conv_mes'] = df_grouped['conversion'] / 5$ $df_grouped['prom_cho_mes'] = df_grouped['checkout'] / 5$	conversion
ratios de campaña	$df_grouped['r_conv_may_camp_hit'] = df_grouped['conv_mayo'] / df_grouped['ad_camp_hit']$ $df_grouped['r_vis_may_camp_hit'] = df_grouped['visit_mayo'] / df_grouped['ad_camp_hit']$ $df_grouped['r_cho_camp_hit'] = df_grouped['checkout'] / df_grouped['ad_camp_hit']$ $df_grouped['r_conv_camp_hit'] = df_grouped['conversion'] / df_grouped['ad_camp_hit']$ $df_grouped['r_viewp_camp_hit'] = df_grouped['viewprod'] / df_grouped['ad_camp_hit']$ $df_grouped['r_cond_bom_camp_hit'] = df_grouped['condition_bom'] / df_grouped['ad_camp_hit']$ $df_grouped['r_search_eng_camp_hit'] = df_grouped['search_engine_hit'] / df_grouped['ad_camp_hit']$	campaign, conversion

ratios de busqueda de termino	df_grouped['r_conv_busqterm']=df_grouped['conversion']/df_grouped['busqueda_term'] df_grouped['r_cho_busqterm']=df_grouped['checkout']/df_grouped['busqueda_term'] df_grouped['r_vis_may_busqterm']=df_grouped['visit_mayo']/df_grouped['busqueda_term'] df_grouped['r_viewprod_busqterm']=df_grouped['viewprod']/df_grouped['busqueda_term']	conversion, search_term, checkout, viewp
ratios condition	df_grouped['r_conv_conbom']=df_grouped['conversion']/df_grouped['condition_bom'] df_grouped['r_cho_conbom']=df_grouped['checkout']/df_grouped['condition_bom'] df_grouped['r_vis_may_conbom']=df_grouped['visit_mayo']/df_grouped['condition_bom'] df_grouped['r_conv_conmuybom']=df_grouped['conversion']/df_grouped['condition_muito_bom'] df_grouped['r_cho_conmuybom']=df_grouped['checkout']/df_grouped['condition_muito_bom'] df_grouped['r_vis_may_conmuybom']=df_grouped['visit_mayo']/df_grouped['condition_muito_bom'] df_grouped['r_conv_conexc']=df_grouped['conversion']/df_grouped['condition_excelente'] df_grouped['r_cho_conexc']=df_grouped['checkout']/df_grouped['condition_excelente'] df_grouped['r_vis_may_conexc']=df_grouped['visit_mayo']/df_grouped['condition_excelente']	conversion, condition, visit, checkout
% del total	denominador=df_grouped.size # a esto lo uso como denominador df_grouped['pt_cho']=df_grouped['checkout']/denominador df_grouped['pt_conv']=df_grouped['conversion']/denominador df_grouped['pt_conv_may']=df_grouped['conv_mayo']/denominador df_grouped['pt_conv_apr']=df_grouped['conv_abril']/denominador df_grouped['pt_conv_mar']=df_grouped['conv_marzo']/denominador df_grouped['pt_vis_may']=df_grouped['visit_mayo']/denominador df_grouped['pt_viewprod']=df_grouped['viewprod']/denominador	checkout, conversion, viewproduct

3. Modelado

Los modelos se trabajaron con una partición del 70% para el training y 30% para el testing. Teniendo en cuenta que el entrenamiento final se realiza con el 100%

La medida de valuación que utilizamos es la curva ROC, representada gráficamente de la siguiente manera:

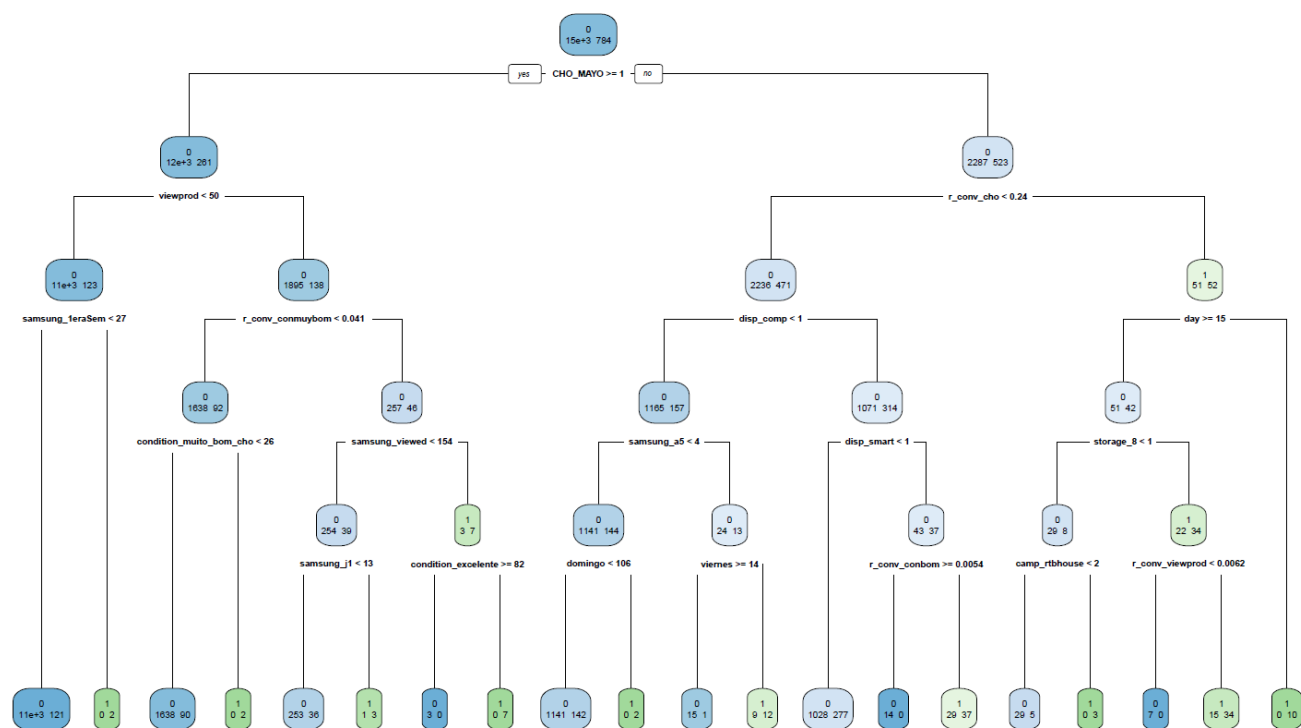


de manera tal que, el resultado, mientras más cercano a 1 se encuentre, el modelo ajustará mejor.

Como primer paso realizamos una prueba utilizando la funcionalidad de “RPART” y de ello inferimos nuestras variables más importantes para poder ingresar al algoritmo de XGboost.

```
Data_set <- rpart(label~.,data=df.train,  
  control=rpart.control(maxdepth = 5,minsplit = 1,minbucket = 1,cp = 0.001,xval = 10) ,  
  method = 'class')
```

```
rpart.plot(x =Data_set,type = 2,extra = 1)
```



Adicionalmente tomamos en cuenta otras metricas de evaluacion que consideramos importantes a la hora de la toma de decisiones:

- Armamos la matriz de confusión
`confusion <- table(df.val$label,predicho)`
 #Paso a Matriz para poder manipularla
`confusion <- as.matrix(confusion)`
 #Ordenamos
`confusion <- confusion[c(2,1),c(2,1)]`
`confusion`
- Accuracy:
`accuracy<-sum(diag(confusion))/sum(confusion)`
- Precisión
`precision<-confusion[1,1]/sum(confusion[,1])`
`precision`
- Sensibilidad
`sensibilidad<-confusion[1,1]/sum(confusion[1,])`

- Especificidad

```
especificidad<-confusion[2,2]/sum(confusion[2,])
```

accuracy

precision

sensibilidad

especificidad

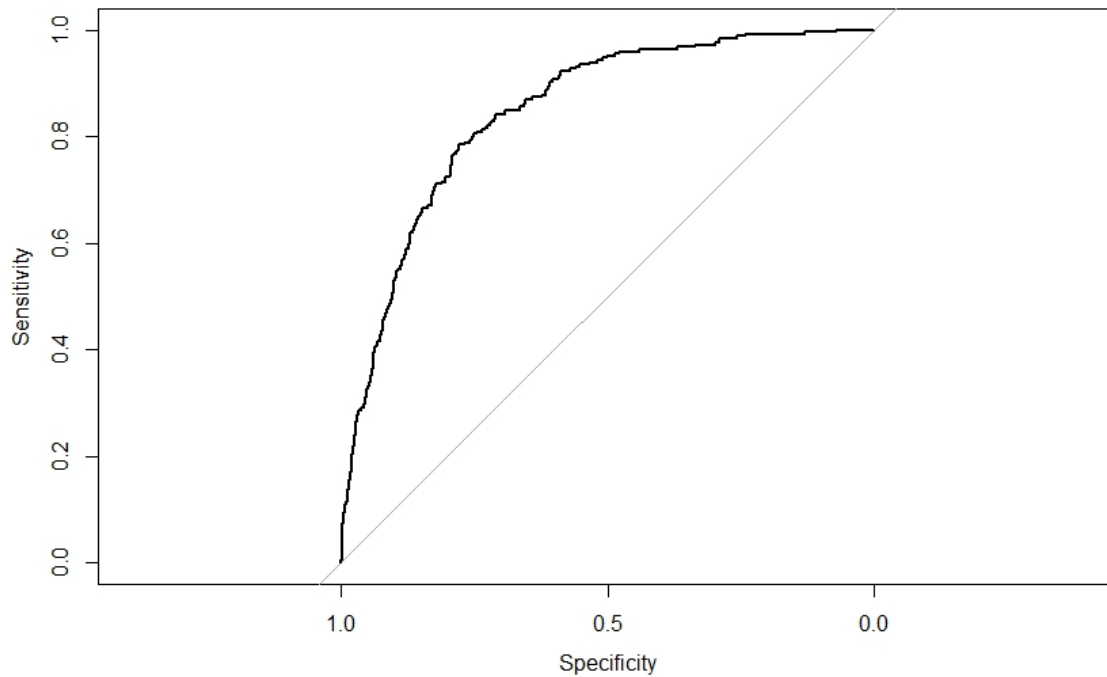
4. Entrenamiento algoritmo XGBoost

```
modelo <- xgboost(data=trMatrix,
  label=as.integer(trTarget)-1,
  eval_metric = "auc",
  #nfold=10,metrics = list("auc","rmse"),
  max_depth=p.max_depth,
  print_every_n = 50,
  #verbose = 0,
  colsample_bytree=p.colsample_bytree,
  min_child_weight=p.min_child_weight,
  nrounds=p.nrounds,
  gamma=p.gamma,
  eta=p.eta,
  nthread=4,
  #lambda=p.lambda,
  #early_stopping_rounds = 100,
  objective = "binary:logistic")

prModelo<- predict(modelo,tsMatrix)
corte_prob<-data.frame(a=tsTarget,b=ifelse(prModelo>0.2,yes=1,no = 0))
table(corte_prob)

roc_test <- roc( tsTarget, prModelo, algorithm = 2)
plot(roc_test)
auc(roc_test)
```

Area Under the Curve (AUC) = 0.8468



Para ajustar el modelo se codificó un grid-search y así encontrar los mejores parámetros que ajustan a este modelo utilizado.

```
for(p.colsample_bytree in c(0.9)){
  for(p.min_child_weight in c(7)){
    for(p.eta in c(0.01)){
      for(p.max_depth in c(3)){
        for(p.gamma in c(20,25,30)){
          #for(p.lambda in c(0,1,5,10,100,200)){
            for(p.nrounds in c(100)){

modelo <- xgboost(data=trMatrix,
  label=as.integer(trTarget)-1,
  eval_metric = "auc",
  #nfold=10,metrics = list("auc","rmse"),
  max_depth=p.max_depth,
  #print_every_n = 100,
  verbose = 0,
  colsample_bytree=p.colsample_bytree,
  min_child_weight=p.min_child_weight,
  nrounds=p.nrounds,
  gamma=p.gamma,
  eta=p.eta,
  nthread=4,
  #lambda=p.lambda,
  #early_stopping_rounds = 100,
  objective = "binary:logistic")
```

```

prModelo<- predict(modelo,tsMatrix)

roc_test <- roc( tsTarget, prModelo, algorithm = 2)

#res<-paste((auc(roc_test ),'p.max_depth=',p.max_depth,',','p.colsample_bytree',p.colsample_bytree)
#res<-paste(round(auc(roc_test),6),
res<-paste(round(auc(roc_test),6),
            ',p.max_depth=',p.max_depth,
            ',p.colsample_bytree=',p.colsample_bytree,
            ',p.min_child_weight=',p.min_child_weight,
            ',p.eta=',p.eta,
            ',p.lambda=',p.lambda,
            ',p.gamma=',p.gamma,
            ',p.nrounds=',p.nrounds
            )
res.total<-rbind(res.total,res)

AUC<-auc(roc_test)
AUC.total<-c(AUC.total,AUC)

    #}
    }
  }
}
}
}
}

```

¡ESTO ES TODO AMIGOS!

