Análisis de precios de Capital Federal

Informe TP2 - Data Mining

Ing. Christian Jorge Marcusa

Maestría en Data Mining, Facultad de Ciencias Exactas y Naturales, Universidad de Buenos Aires
Presentado el 9 de Julio de 2019

Abstract

La fluctuación de precios en los mercados porteños es un factor fundamental en la vida de los ciudadanos. Con variables y factores externos influyentes en ello, como la inflación, la oferta y demanda, el mercado cambiario, e incluso la elaboración nacional y extranjera de los mismos, se hace muy difícil encontrar un precio acorde a cada producto. En este desarrollo se intenta encontrar asociaciones entre las subas de los precios en los puntos de venta al consumidor.

Keywords: Data Mining, Precios, Asociacion, Reglas, Arules, KDD

Introducción

Al hablar de precios, y su análisis, debemos siempre recurrir a datos objetivos e históricos de los mismos. Por lo tanto se ha buscado información confiable provista por el gobierno de la ciudad para realizar este análisis.
Se ha planteado una estructura de datos sencilla y durante su desarrollo se irá simplificando aún más para análisis específicos.

Preparación de datos de precios

Si bien el mayor análisis ya fue realizado en etapas previas, se opta por hacer una preparación de cero de los datos.

En primer lugar el dataframe de precios contiene muchos datos faltantes y está en formato vertical. Se ha decidido pasar a horizontal la colección de datos y agregar las distintas mediciones como columnas de cada subconjunto “producto - sucursal”. Para ello se ha utilizado los paquetes dplyr y tidyr. (En particular las funciones select y spread de los mismos respectivamente).

	producto	sucursal	1	2	3	4	5	6	7	8	9	10
1	0000040084107	10-1-214	36.00	36.00	36.00	36.00	36.00	36.00	43.50	43.50	43.50	55.00
2	0000040084107	10-1-219	36.00	36.00	36.00	36.00	36.00	36.00	43.50	43.50	46.00	55.00
3	0000040084107	10-1-220	36.00	36.00	36.00	36.00	36.00	36.00	NA	43.50	43.50	55.00
4	0000040084107	10-1-26	36.00	36.00	36.00	36.00	36.00	NA	43.50	43.50	43.50	55.00
5	0000040084107	10-1-271	36.00	36.00	36.00	36.00	36.00	36.00	43.50	43.50	46.00	55.00
6	0000040084107	10-1-2	36.00	36.00	36.00	36.00	36.00	36.00	43.50	43.50	43.50	55.00

Luego para limpiar y discretizar las mediciones se han creado una serie de funciones que iteran sobre los datos. Para el desarrollo de las mismas se ha partido en un conjunto de test (Usando la función head) para agilizar su desarrollo, y luego ser aplicadas para el conjunto total (Debido al gran volumen de mediciones)

Las funciones desarrolladas fueron

- **completarFaltantes**(dataframe) : Recibe un dataframe y para cada valor NA dentro de sus mediciones, se le completa con la media entre sus adyacentes, o su adyacente contiguo si es una medición extrema (1 o 10), en el caso que sus adyacentes contengan NAs, ignora la operación.
- **eliminarFaltantes**(dataframe) : Utiliza la función drop_na de tidyr para descartar los registros que no pudieron ser llenados por la función anterior.
- agregarColumnasPeriodos(dataframe) : Recibe un dataframe y para cada fila, separa las mediciones en 4 periodos (1 a 3, 4 a 5, 6 a 7 y 8 a 10), luego calcula el precioPromedio para todos los periodos.
- **agregarVariaciones**(dataframe) : Recibe un dataframe y para cada fila, calcula y agrega las variaciones para cada periodo. Es decir [p(x) - p(x-1)] / p(x-1). Luego también calcula y agrega la variación total, del periodo 1 al 4.
- **discretizacionDeVariaciones**(dataframe) : Discretiza las variaciones según la siguiente tabla.

Categorías	Rango
Disminución Fuerte	$[-\infty;\ -0.05)$
Disminución Media	$[-0.5;\ -0.02)$
Disminución Leve	$[-0.02;\ -0.005)$
Mantiene	$[-0.005;\ 0.005)$
Aumento Leve	$[0.005; 0.05)$
Aumento Medio	$[0.05; 0.1)$
Aumento Fuerte	$[0.1; \infty]$

- **mediasPorProducto**(dataframe) : Recibe un dataframe y utilizando dplyr, calcula las medias de precios para cada producto (Agrupa por el mismo, dejando fuera las sucursales). Retorna un nuevo dataframe con las medias para cada periodo, y la media total del precio.
- **preciosRelativos**(dataframe, promedios) : Recibe un dataframe, y otro dataframe con las medias calculadas por la función antes explicada. Para cada elemento del primer dataframe, calcula su diferencia con las medias calculadas en el segundo.

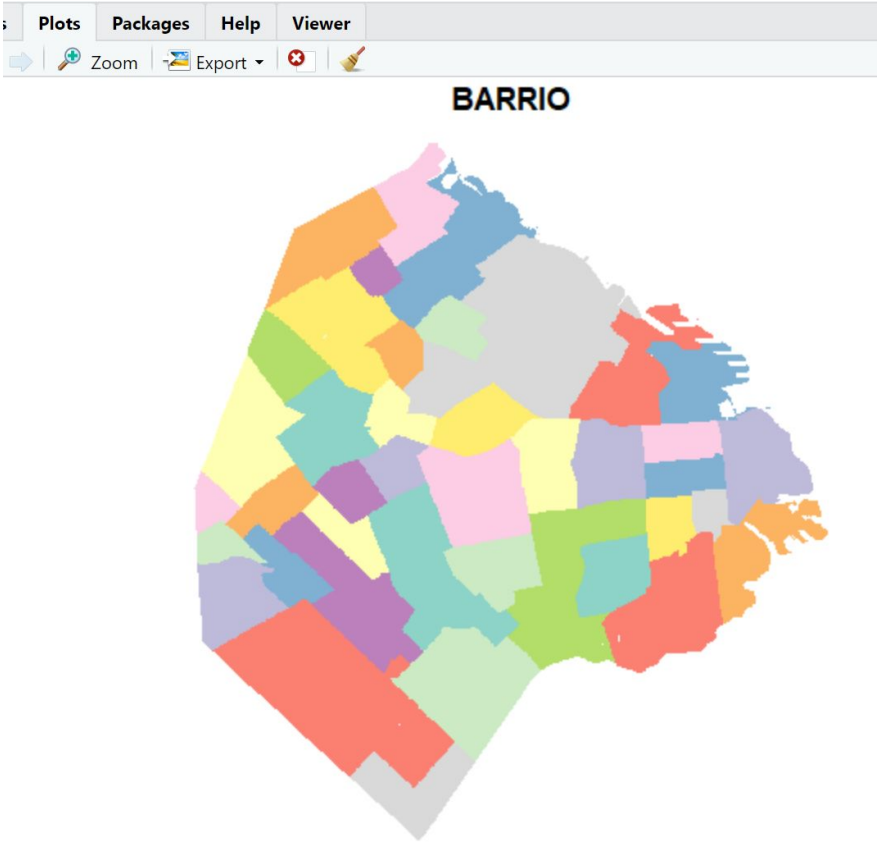
- **discretizacionesDePrecio**(dataframe) : Recibe un dataframe y discretiza las mediciones de la función explicada en el apartado anterior, según la tabla que se detalla a continuación.

Categorías	Rango
Muy caro	$[\infty; 0.1)$
Medio caro	$[0.1; 0.05)$
Levemente caro	$[0.05; 0.01)$
Medio	$[-0.01; 0.01)$
Levemente barato	$[-0.05; -0.01)$
Medianamente barato	$[-0.1; -0.05)$
Muy barato	$[-\infty;\ 0.1)$

Con este procedimiento ya se disponen los datos de precios para trabajar con las reglas, detallado en los siguientes apartados.

Preparación de datos de sucursales

Para los datos de sucursales se han utilizado datos geográficos obtenidos de distintas APIs en la red. La primera que se ha incorporado es el dataset "https://bitsandbricks.github.io/data/CABA_rc.geojson" el cual contiene objetos Simple Feature (sf a partir de ahora) para plotear informacion geografica. El dataset contiene mucha información no pertinente a este análisis pero permite graficar las sucursales según los barrios.



A su vez también se necesitó obtener información pertinente a la geolocalización de las sucursales, ya que solo se dispone de la LATITUD y la LONGITUD.

Para esto se ha utilizado la API Open Cage Data (https://opencagedata.com), la cual expone una sencilla API HTTPS que con una Key (por login), una coordenada de latitud y una de longitud, brinda información pertinente a nuestro análisis.

El problema que se tiene es que parsear la respuesta desde R fue complicado porque está en formato JSON con mucha información no importante, de la cual solo interesaba lo expuesto en el siguiente snippet:


```
"components" : {
    "ISO_3166-1_alpha-2" : "AR",
    "ISO_3166-1_alpha-3" : "ARG",
    "_type" : "building",
    "city" : "Buenos Aires",
    "continent" : "South America",
    "country" : "Argentina",
    "country_code" : "ar",
    "house_number" : "4877",
    "neighbourhood" : "Parque Cornelio Saavedra",
    "postcode" : "1603",
    "road" : "Avenida Doctor Ricardo Balb\u00edn",
    "state" : "Ciudad Aut\u00f3noma de Buenos Aires",
    "state_code" : "C",
    "suburb" : "Saavedra"
}
```


Y más en particularmente el campo suburb (Barrio), ya que el resto de la información ya se dispone (Ciudad, país) o es información más específica no útil (como la dirección exacta) o mismo información redundante (como el campo “neighbourhood”).

Para extraer esta información se desarrolló un pequeño módulo en la tecnología NodeJS (Debido a su facilidad para consumir APIs y trabajar con JSON) el cual obtiene el dataset guardado desde R, lo parsea en JSON, consume las APIs y para cada elemento del dataset le agrega el barrio y lo vuelve a guardar, y esta informacion ya esta disponible para ser usada por el código de R.

Luego el dataset de sucursales ya puede ser agrupado y filtrado por barrio.

	field1	id	lat	lng	barrio
1	46	3-1-3	-34.546397	-58.451884	Belgrano
2	95	15-1-2	-34.5785767	-58.4871575	Villa Urquiza
3	23	10-3-373	-34.543233	-58.463148	Núñez
4	6	10-3-300	-34.540539	-58.472052	Núñez
5	7	15-1-498	-34.5512734	-58.4871771	Saavedra

Preparación de datos de productos

Los datos de productos tienen 3 campos de texto que brindan información redundante, los mismos son nombre, marca y descripción.

Se han utilizado los paquetes tm , stringi, stringr y arules para darle forma al dataset. En particular las funciones removeNumbers, removePunctuation, stri_trans_general, stripWhitespace, unique y la funcion removeWords con stopwords (la cual recibe un parametro kind='sp' y permite obtener las palabras no útiles al análisis, como las preposiciones y los artículos). Para guardar el dataset generado y no hacer denuevo el procesamiento, se ha utilizado write.csv, y se ha guardado el dataset limpio, ejemplificado a continuación:

	id	nombre	presentacion	marca
1	0000040084107	huevo chocolate sorpresa	gr	kinder
2	0000075027513	desodorante original	gr	dove
3	0000075032715	cerveza rubia	cc	corona
4	0000077903518	galletitas obleas rellena	gr	opera
5	0000077940131	turron	gr	arcor
6	0000077940704	desodorante crema pote odorono	gr	rexona
7	0000077953063	postre dulce leche pack	gr	danette

Metodologia

Ya disponiendo de los dataset limpios, se procede a hacer el análisis correspondiente. En el caso de los productos, que ya se venía trabajando con arules, se generó un vocabulario de palabras utilizando una frecuencia mínima de 20 ocurrencias, con la función findFreqTerms de tm.

Lo mismo arrojó un vocabulario de 32 palabras importantes:

[1] "chocolate" "desodorante" "galletitas" "crema" "dulce" "leche"
[7] "pack" "aerosol" "blanco" "agua" "jabon" "liquido"
[13] "doypack" "limon" "cafe" "manzana" "polvo" "jugo"
[19] "naranja" "vainilla" "light" "frutilla" "queso" "yogur"
[25] "fideos" "mate" "vino" "tinto" "malbec" "gas"
[31] "saborizada" "gaseosa"

Una aclaración importante, es que si se aumenta la frecuencia del vocabulario a 30, el mismo se reduce a 15 elementos, y ya subiendolo a 100, da cero. Esto quiere decir que hay cero términos que estén en 100 productos simultáneamente. Se probó con varias opciones y se eligió el número 20 porque 32 palabras parece simbólico en un total de 1000 productos.

Luego utilizando la función merge, se obtuvo un dataset de trabajo general de 44 columnas, a saber: *variacionesDisc(periodo 1a2, periodo 2a3, periodo 3a4, total), preciosRelativosDisc(periodo 1, periodo 2, periodo 3, periodo 4,total), marca, presentación, barrio, y el vocabulario de 32 palabras expuesto en el apartado anterior*.

Análisis descriptivo

Primera iteracion, generación de reglas con support=0.1 y confidence=0.1, arroja 359 reglas.

lhs rhs support confidence lift count
[1] {v1d=Mantiene,
v2d=Mantiene,

v3d=Mantiene} => {variacionTotalDiscreta=Mantiene} 0.1348126 0.9969033 6.7637844 21247
[2] {v1d=Mantiene,
v3d=Mantiene} => {variacionTotalDiscreta=Mantiene} 0.1350410 0.8169117 5.5425782 21283
[3] {v2d=Mantiene,
v3d=Mantiene} => {variacionTotalDiscreta=Mantiene} 0.1349458 0.5861860 3.9771517 21268
[4] {v1d=Mantiene,
v2d=Mantiene} => {variacionTotalDiscreta=Mantiene} 0.1348379 0.5384089 3.6529941 21251
[5] {pr2d=Levemente barato,
pr4d=Levemente barato} => {prtd=Levemente barato} 0.1105112 0.8917618 3.5291590 17417

La primera iteración indica que los productos no suelen cambiar su precio en el tiempo. Buscando reglas con menos LIFT, se encuentran patrones significativos, por ejemplo:

[160] {presentacion= gr} => {v1d=Aumento Fuerte} 0.1555100 0.3201782 1.2078743 24509

Los productos cuya presentación es en gramo, sufrieron un aumento fuerte entre el periodo 1 y el 2 con un 32% de confianza.

[162] {v2d=Aumento Medio} => {variacionTotalDiscreta=Aumento Fuerte} 0.1555481 0.8185036 1.2057261 24515

Con un 81% de confianza, los productos que sufrieron un aumento medio en entre el periodo 2 y el 3, tuvieron un aumento fuerte en el total.

[164] {v1d=Aumento Fuerte, variacionTotalDiscreta=Aumento Fuerte} => {presentacion= gr} 0.1479404 0.5853585 1.2051894 23316

Los productos que tuvieron un aumento fuerte entre el 1 y el 2, y en el total, son presentados en gramos. 58% confianza.

[165] {pr1d=Medio} => {v1d=Mantiene} 0.1401868 0.5603632 1.1984893 22094

Los productos con un precio relativo medio en el primer periodo, se mantuvieron asi hasta el segundo. 56% confianza.

[168] {variacionTotalDiscreta=Aumento Fuerte} => {v3d=Aumento Medio} 0.1822035 0.2684014 1.1941041 28716

Los productos que tuvieron un aumento fuerte en el total, en general tuvieron un aumento medio entre los periodos 3 y 4. 26%

En este punto se sube la confianza a 0.5 para obtener reglas mas fuertes. Se obtienen 188 reglas.

[141] {presentacion= lt} => {variacionTotalDiscreta=Aumento Fuerte} 0.1023324 0.7972712 1.174449 16128

Los productos en litros, tuvieron un aumento fuerte general con un 79% de confianza.

En este punto se observó que debido al soporte, la presencia de las variaciones y precios discretizados es mucho mayor que los campos del vocabulario, y el barrio. Por lo tanto se decide usar el paquete dplyr para obtener subsets de la información y conseguir reglas con ellos.

En primer lugar se trabaja con los productos muy caros. Usando filter y select, se extraen las columnas de precios relativos y se vuelven a generar reglas (132 reglas).

[29] {variacionTotalDiscreta=Aumento Fuerte,presentacion= ml} => {v3d=Aumento Fuerte} 0.1073124 0.7911319 2.7737908

Con un 79% se afirma que los productos que aumentaron fuertemente, y se presentan en mililitros, tuvieron su aumento fuerte entre el tercer y el cuarto periodo.

[37] {v3d=Aumento Fuerte,variacionTotalDiscreta=Aumento Fuerte} => {vino} 0.1517885 0.5492554 2.2875385

Con un 54% de confianza se afirma que los aumentos fuertes en el total y en el último periodo, se dieron en el vino y de la misma manera:

[39] {vino} => {v3d=Aumento Fuerte} 0.1517885 0.6321688 2.2164496

El vino aumentó fuertemente en un 63% en la ultima variacion.

[116] {vino} => {v1d=Mantiene} 0.1394429 0.5807515 1.0158327

[117] {vino} => {variacionTotalDiscreta=Aumento Fuerte} 0.1620766 0.6750165 1.0056011

También se obtuvieron reglas por geolocalización

[126] {barrio=Recoleta} => {variacionTotalDiscreta=Aumento Fuerte} 0.1063628 0.6524272 0.9719488

[127] {barrio=Palermo} => {variacionTotalDiscreta=Aumento Fuerte} 0.1060462 0.6473430 0.9643747

En Recoleta y Palermo (zonas de alto poder adquisitivo) hubo aumentos fuertes en el 65 y 64% de los productos caros.

Se hizo el mismo análisis para productos muy baratos. Con Supp=0.1 y Conf=0.4 se obtuvieron 85 reglas. Se tuvo que bajar la confianza a 0.4 porque las reglas con 0.5 no aportan información. (Ejemplo vino => tinto)

{} => {variacionTotalDiscreta=Aumento Fuerte} 0.4258973 0.4258973 1.000000 3026

Esta regla me llamo la atencion, nos dice que el 42% de los productos baratos aumentó fuertemente.

[77] {presentacion= gr} => {v1d=Mantiene} 0.1532723 0.4488871 1.159761 1089

[78] {v1d=Mantiene} => {vino} 0.1753695 0.4530909 1.153839 1246

[79] {vino} => {v1d=Mantiene} 0.1753695 0.4465950 1.153839 1246

[80] {presentacion= ml} => {v2d=Mantiene} 0.1156932 0.4584495 1.150984 822

[81] {tinto} => {v1d=Mantiene} 0.1242787 0.4212786 1.088431 883

[82] {vino,tinto} => {v1d=Mantiene} 0.1242787 0.4212786 1.088431 883

A diferencia de los productos caros, el vino se mantuvo mejor en los productos baratos.

Luego se opta subdividir por barrios opuestos de la ciudad, se toman Recoleta y Villa Lugano.

En recoleta se encuentran 144 reglas con 0.1 y 0.5, se destacan:

[134] {} => {variacionTotalDiscreta=Aumento Fuerte} 0.6765391 0.6765391 1.0000000 9868

En un 67% de confianza, en los productos hubo un aumento fuerte.

Se decide refinar el dataset, no incluyendo la información interperiodo, para obtener reglas más específicas de los barrios. Se destacan las siguientes reglas:

En Recoleta:

[1] {presentacion= lt} => {variacionTotalDiscreta=Aumento Fuerte} 0.1031126 0.7974549 1.1787270 1504

[8] {prtd=Medio} => {variacionTotalDiscreta=Aumento Fuerte} 0.1493898 0.6698432 0.9901027 2179

[4] {galletitas} => {variacionTotalDiscreta=Aumento Fuerte} 0.05628685 0.8112648 1.1991395 821

[5] {presentacion= gr,galletitas} => {variacionTotalDiscreta=Aumento Fuerte} 0.05628685 0.8112648 1.1991395 821

En Villa Lugano:

[1] {presentacion= lt} => {variacionTotalDiscreta=Aumento Fuerte} 0.1026895 0.8092486 1.1584972 420
[7] {prtd=Medio} => {variacionTotalDiscreta=Aumento Fuerte} 0.1963325 0.6928387 0.9918481 803
[8] {presentacion= gr} => {variacionTotalDiscreta=Aumento Fuerte} 0.3349633 0.6853427 0.9811171 1370

A esta altura, explorando el dataset, se decide extraer algunas palabras que generan reglas redundantes, por ejemplo “vino, tinto, malbec” podrían ser una sola, y se puede bajar el Support sin generar reglas redundantes. Se extraen las palabras *tinto, malbec, blanco, liquido, doypack, gas, polvo*.

Para analizar la desaceleración de precios en el último periodo, se generan un set de reglas y lo filtramos cuya variación en el último periodo sea “Mantiene” o cualquiera de las tres “disminuciones”. Se filtran por el RHS (Siendo uno de los valores del último periodo, mantiene, o disminución). Con un supp de 0.1 y una confianza de 0.5 se obtuvieron las siguientes reglas (filtradas según la utilidad del informe):

[15] {v1d=Aumento Fuerte,
v2d=Mantiene} => {v3d=Mantiene} 0.06658460 0.5179153 1.499945 10494
[16] {v1d=Aumento Fuerte,
presentacion= gr} => {v3d=Mantiene} 0.08487729 0.5457995 1.580701 13377
[32] {v1d=Aumento Fuerte,
v2d=Mantiene,
variacionTotalDiscreta=Aumento Fuerte} => {v3d=Mantiene} 0.06656557 0.5429281 1.572385 10491
[33] {v1d=Aumento Fuerte,
variacionTotalDiscreta=Aumento Fuerte,
presentacion= gr} => {v3d=Mantiene} 0.08472501 0.5726969 1.658599 13353

Como se puede observar, hay reglas con confianza de más del 50% en las que se produjo un aumento fuerte, tanto en períodos individuales como en el total, y en el último periodo se mantuvo.

Trabajando sobre un lote particular de productos, se eligen las galletitas, y se generan reglas sobre ellas con un supp de 0.2 y una confianza de 0.5. Se eligen las reglas detalladas a continuación (De un total de 47):

[12] {v2d=Aumento Fuerte,variacionTotalDiscreta=Aumento Fuerte} => {v3d=Mantiene} 0.2474858 0.7857765
[13] {v3d=Mantiene,variacionTotalDiscreta=Aumento Fuerte} => {v1d=Aumento Fuerte} 0.4105869 0.8564073
[14] {v2d=Aumento Fuerte} => {v3d=Mantiene} 0.2477601 0.7469680
[15] {v1d=Aumento Fuerte,variacionTotalDiscreta=Aumento Fuerte} => {v3d=Mantiene} 0.4105869 0.7266990
[16] {v1d=Aumento Fuerte} => {v3d=Mantiene}
[35] {} => {v3d=Mantiene} 0.5394039 0.5394039
[36] {} => {v2d=Mantiene} 0.5701225 0.5701225
[37] {} => {v1d=Aumento Fuerte} 0.6005668 0.6005668
[38] {} => {variacionTotalDiscreta=Aumento Fuerte} 0.8086487 0.8086487
[41] {prtd=Levemente barato} => {variacionTotalDiscreta=Aumento Fuerte} 0.2517828 0.7497958

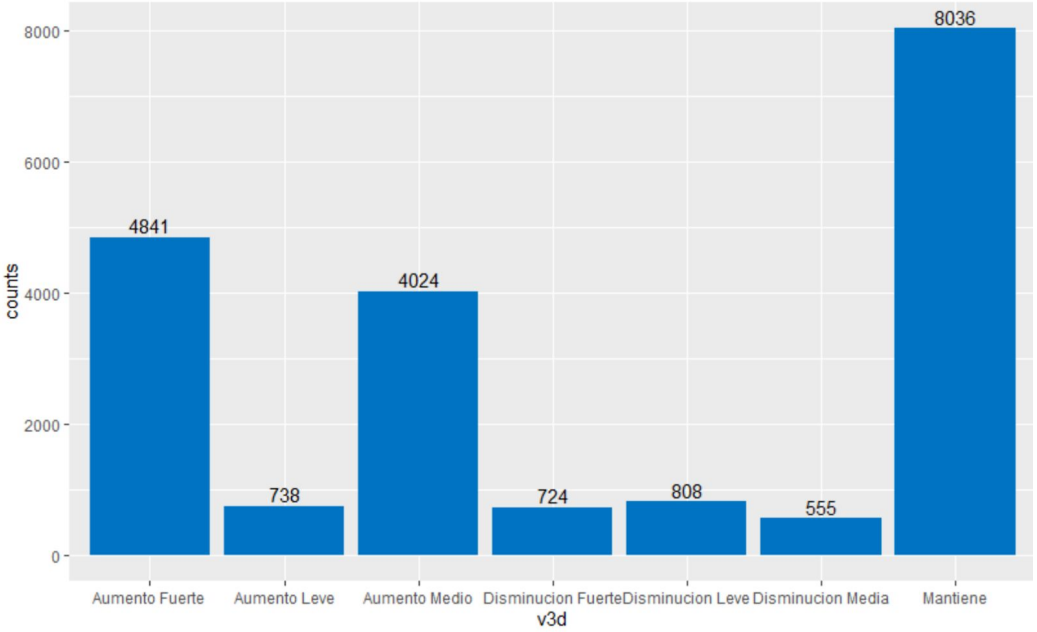
Estas reglas marcan que el aumento fuerte de las galletitas (0.80), estuvo dado más que nada en la primer variación, y en general se mantuvo en la última

Analisis predictivo

Para abordar este análisis se genera un conjunto de reglas utilizando el dataset sin las mediciones de variación y precio relativo totales, y sin el precio y la variación del último período, luego se eligen algunas reglas sobre la variación del segundo periodo, y se contrastan con las métricas del tercero.

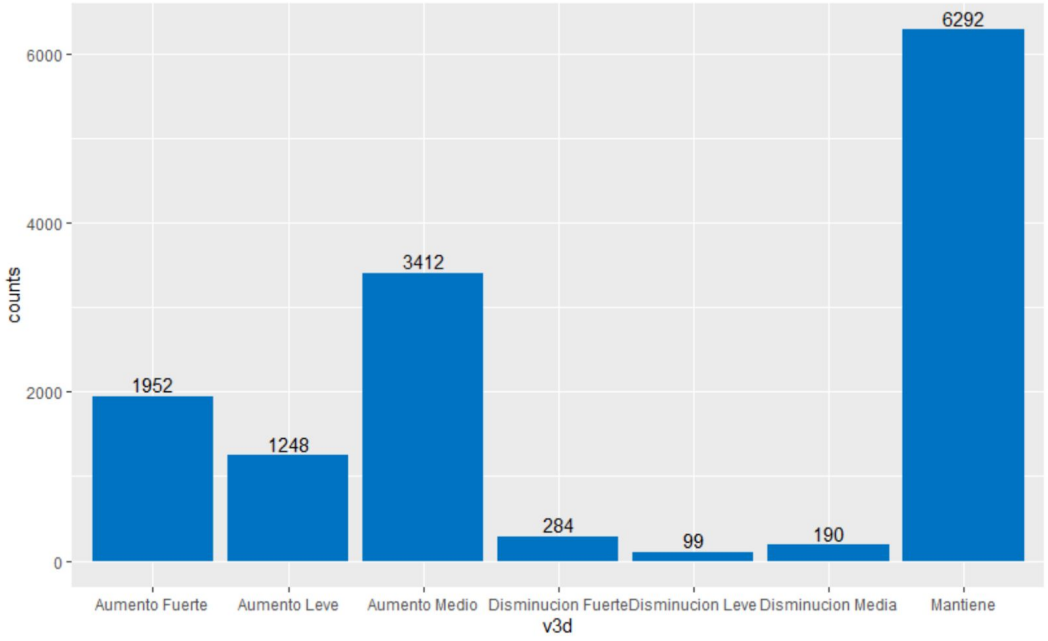
[105] {pr3d=Levemente barato,presentacion= gr} => {v2d=Mantiene} 0.06878633 0.5495792 1.211055 10841

Se filtra el dataset en los productos levemente baratos y que se presentan en gr. Luego se analiza la v3d.



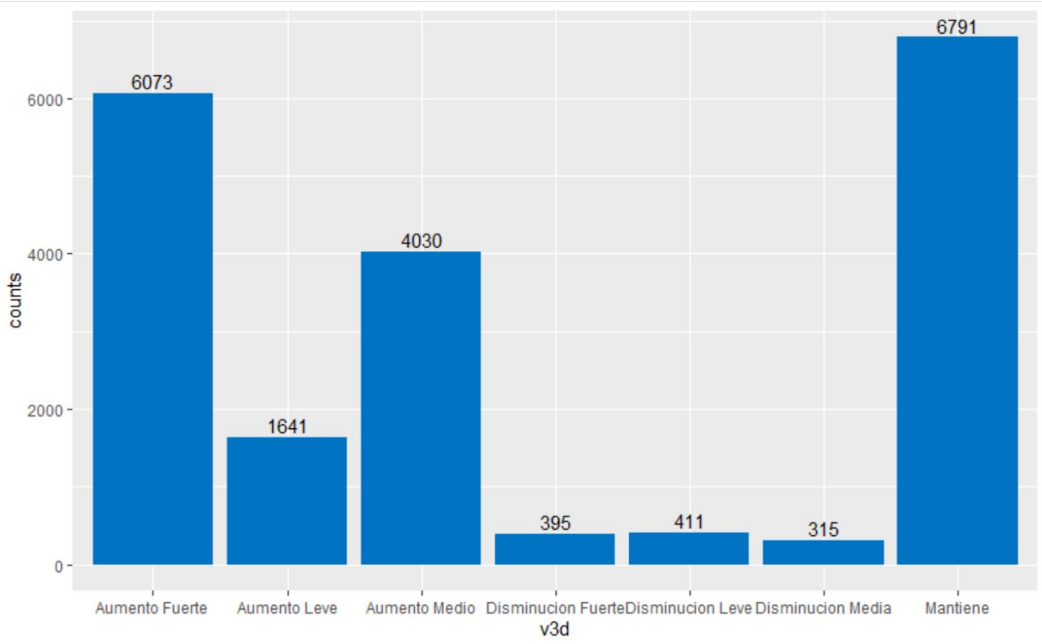
Como se puede observar, menos del 50% mantuvo la tendencia en el tercer periodo para los productos en gr levemente baratos. La regla arrojaba una confianza del 54%.

[80] {v1d=Mantiene,pr2d=Medio,pr3d=Medio} => {v2d=Mantiene} 0.05343138 0.6248423 1.376905 8421



Como se observa, la regla había arrojado un 62% de confianza para la variación de v2, y vemos que ronda el por debajo del 50% para los productos con un precio medio en los periodos 2 y 3, que se mantenían.

[82] {v1d=Mantiene,pr1d=Levemente caro} => {v2d=Mantiene} 0.07565798 0.6066341 1.336782 11924



Esta regla directamente no se cumple en el último periodo, ya que la confianza era de un 0.60 y en el último el % de “Mantiene” no llega ni a un 33%.

Conclusiones

Los resultados de este informe demuestran varias hipótesis sobre el comportamiento de los precios.

- Primero que los precios en general se desaceleraron en el último periodo.
- Luego que en los barrios más pudientes los productos caros sufrieron un aumento fuerte, mientras que en los demás no hay evidencia para afirmarlo.
- Las predicciones de los primeros periodos no se cumplieron en el último. Se observa una desaceleración, pero en los productos donde se predecía que se mantenían con una fuerte confianza, no se pudo sustentar en el último.
- Los productos baratos aumentaron fuertemente en su mayoría.
- Los productos que se venden por litro sufrieron aumentos considerables.
- Las galletitas sufrieron su mayor aumento en el primer periodo, y luego desaceleraron.
- Al igual que los datos arrojados por el TP1, se valida que el vino es de los grandes afectados por la suba, en especial el de mediana y alta gama (Medianamente caro +).

Referencias

- Código fuente utilizado para este trabajo está disponible en GitHub, solicitar acceso. (https://github.com/Chaitooler).
- "https://bitsandbricks.github.io/data/CABA_rc.geojson"
- http://opencagedata.com
- Dplyr-Documentation (https://dplyr.tidyverse.org/)
- R-Documentation (https://www.rdocumentation.org/)
- https://dmuba.github.io/
- MongoDB-Documentation https://docs.mongodb.com/