Datamining - Reglas de asociación utilizando el algoritmo apriori con herramienta R

25 Junio 2010

Resultados Técnicos Esperados

Detalle del preprocesamiento de los datos.
Software y algoritmo utilizado
Justificación de la elección de los parámetros del algoritmo (la misma puede ser empírica)
Explicación del criterio utilizado para seleccionar los resultados gerenciales

Resultados Gerenciales Esperados

1. Seleccionar con utilizando el criterio explicitado en los resultados técnicos las 10 reglas más significativas para el primer mes para el cual se tienen datos.
2. Comparar la evolución del criterio utilizado en cada una de las reglas a lo largo de todos los meses del dataset.
3. En algunos casos las personas presentan "saltos" en los productos que tienen de un mes a otro. Estos saltos están definidos por la aparición en un cierto mes de un producto con el cual la persona no había contado hasta ese momento. Estudiar los casos donde se presentan "saltos" y detectar las características de los mismos.
4. Analizar cuáles son las características demográficas de los poseedores de productos de cada tipo de banco

1) Seleccionar, explicitado el criterio utilizado en los resultados técnicos, las 10 reglas más significativas para el primer mes para el cual se tienen datos.

El primer paso fue pasar los archivos a una base de datos. Trabajamos sobre un motor de base de datos Postgres en una de sus ultimas versiones. De ese trabajo se generaron las siguientes tablas:

```
-- Table: bcra
-- DROP TABLE bcra;

CREATE TABLE bcra
(
  id character varying(50),
  banco character varying(100),
  periodo character varying(50),
  situacion character varying(100),
  monto character varying(50)
)
WITH ( OIDS=FALSE );
ALTER TABLE bcra OWNER TO postgres;


-- Table: personas_crudo
-- DROP TABLE personas_crudo;

CREATE TABLE personas_crudo
(
  id character varying(50),
  sexo character varying(100),
  fnacimiento character varying(50),
  afip_activo character varying(100),
  afip_ganancias character varying(100),
  afip_iva character varying(100),
  afip_sociedades character varying(100),
  afip_empleador character varying(100),
  actividad1 character varying(100),
  actividad2 character varying(100),
  monotributo character varying(100),
  mono_categoria character varying(100),
  mono_fdesde character varying(100),
  autonomo character varying(100),
  auto_categoria character varying(100),
  auto_fdesde character varying(100),
  monto_estimado character varying(100),
  auto_anio character varying(100),
  auto_precio character varying(100),
  a_dom_provincia character varying(100),
  cod_postal character varying(100)
)
WITH ( OIDS=FALSE );
ALTER TABLE personas_crudo OWNER TO postgres;
```

Con las tablas bcra y personas_crudo se migraron los datos de los archivos tal cual estaban,

guardando cada dato como una cadena de caracteres.

Luego se generaron tablas, mas estratégicas, con tipos de datos ajustados a cada columna, la información sin normalizar y el agregado de indices que ayuden en los tiempos de respuesta de las consultas realizadas:

```
-- Table: cuentas
-- DROP TABLE cuentas;

CREATE TABLE cuentas
(
  id integer NOT NULL,
  banco character varying(100) NOT NULL,
  periodo date,
  anio smallint NOT NULL,
  mes smallint NOT NULL,
  situacion smallint,
  monto numeric,
  CONSTRAINT "PK" PRIMARY KEY (id, banco, anio, mes)
)
WITH (OIDS=FALSE);
ALTER TABLE cuentas OWNER TO postgres;

-- Index: "MES_IDX"
-- DROP INDEX "MES_IDX";
CREATE INDEX "MES_IDX"
  ON cuentas
  USING btree
  (mes);

-- Index: banco_idx
-- DROP INDEX banco_idx;
CREATE INDEX banco_idx
  ON cuentas
  USING btree
  (banco);


-- Table: personas
-- DROP TABLE personas;
CREATE TABLE personas
(
  id integer,
  sexo character(1),
  fnacimiento date,
  afip_activo boolean,
  afip_ganancias character(2),
  afip_iva character(2),
  afip_sociedades character(1),
  afip_empleador boolean,
  actividad1 integer,
  actividad2 integer,
  monotributo boolean,
  mono_categoria character(1),
```

```
  mono_fdesde date,
  autonomo boolean,
  auto_categoria character varying(100),
  auto_fdesde date,
  monto_estimado numeric(8,2),
  auto_anio smallint,
  auto_precio numeric,
  a_dom_provincia character varying(100),
  cod_postal character varying(100)
)
WITH (OIDS=FALSE);
ALTER TABLE personas OWNER TO postgres;
```

De la base de datos se extrajo diferentes conjuntos de datos para ser procesados usando el R. Para el primer punto se extrajo la informacion del primer mes en un archivo bancos_2_2008.csv. A continuación un extracto de las primeras lineas para ver su formato.

ID	BANCO
2	BANCO HIPOTECARIO S.A.
4	AMERICAN EXPRESS ARGENTIN
6	HSBC BANK ARGENTINA S.A.
6	STANDARD BANK ARGENTINA S.A.
6	AMERICAN EXPRESS ARGENTIN
9	BANCO DE GALICIA Y BUENOS AIRES S.A.
10	BBVA BANCO FRANCES S.A.
10	BANCO DE LA CIUDAD DE BUENOS AIRES

Para leerlo desde el R se uso la siguiente linea:

```
bancos <- read.transactions('bancos_2_2008.csv', format='single', sep=',', cols=c(1,2))
```

El trabajo posterior fue correr el algoritmo apriori variando cada uno de los factores.

Luego experimentamos de varias maneras correr el algoritmo a priori. Sucedió primero que dejando una confianza alta, teniamos que bajar mucho el soporte. En esto casos dejamos 0.6 de confianza, y variamos el soporte con 0.1, 0.01, 0.001 y 0.0001. Obteniendo 0 reglas, 0 reglas, 4 reglas y 365 reglas respectivamente

Luego probamos bajando la confianza para poder subir el soporte. Asi se generaron reglas de longitud uno que fuimos descartando por solo representar los items mas frecuentes. Además de generar las mismas reglas que obtuvimos con los parámetros elegidos que explicamos a continuación.

Finalmente los parametros elegidos para correr el algoritmo de aprioir fueron:

```
apriori(bancos_2_2008, parameter=list(support=0.0001, confidence=0.6));
```

De estas 365 reglas obtenidas, elegimos las 10 siguientes:

	rules	support	confidence	lift
1	{BANCO DE GALICIA Y BUENOS AIRES S.A.,BANCO DE LA PROVINCIA DE CORDOBA S.A.} => {TARJETA NARANJA S.A.}	0.001955	0.884615	8.529341
2	{BANCO COLUMBIA S.A.,BANCO PRIVADO DE INVERSIONES S.A.} => {HSBC BANK ARGENTINA S.A.}	0.000339	0.8	22.196706
3	{BANCO MACRO S.A.,CREDILOGROS COMPANIA FINANCIERA S.A.} => {BANCO COLUMBIA S.A.}	0.000339	0.8	40.662069
4	{BANCO DE LA NACION ARGENTINA,NUEVA CARD S.A.} => {BANCO DE LA PROVINCIA DE BUENOS AIRES}	0.000254	0.6	11.971574
5	{BANCO CETELEM ARGENTINA S.A.,BANCO PATAGONIA S.A.} => {GE COMPANA FINANCIERA S.A.}	0.000254	0.6	26.8
6	{BANCO COLUMBIA S.A., BANCO DE GALICIA Y BUENOS AIRES S.A.} => {TARJETA NARANJA S.A.}	0.001357	0.842105	8.119465
7	{BANCO DE GALICIA Y BUENOS AIRES S.A., COMPAÑIA FINANCIERA ARGENTINA S.A.} => {TARJETA NARANJA S.A.}	0.001866	0.628571	6.0606
8	{BANCO DE GALICIA Y BUENOS AIRES S.A., BANCO DEL TUCUMAN S.A.} => {TARJETA NARANJA S.A.}	0.000848	0.909091	8.765331
9	BANCO DE GALICIA Y BUENOS AIRES S.A., NUEVO BANCO DE ENTRE RÍOS S.A.} => {TARJETA NARANJA S.A.}	0.000848	0.769231	7.416819
10	{BANCO DE GALICIA Y BUENOS AIRES S.A., NUEVO BANCO DE SANTA FE SOCIEDAD ANONIMA} => {TARJETA NARANJA S.A.}	0.000678	0.8	7.713491

El criterio elegido fue el siguiente:

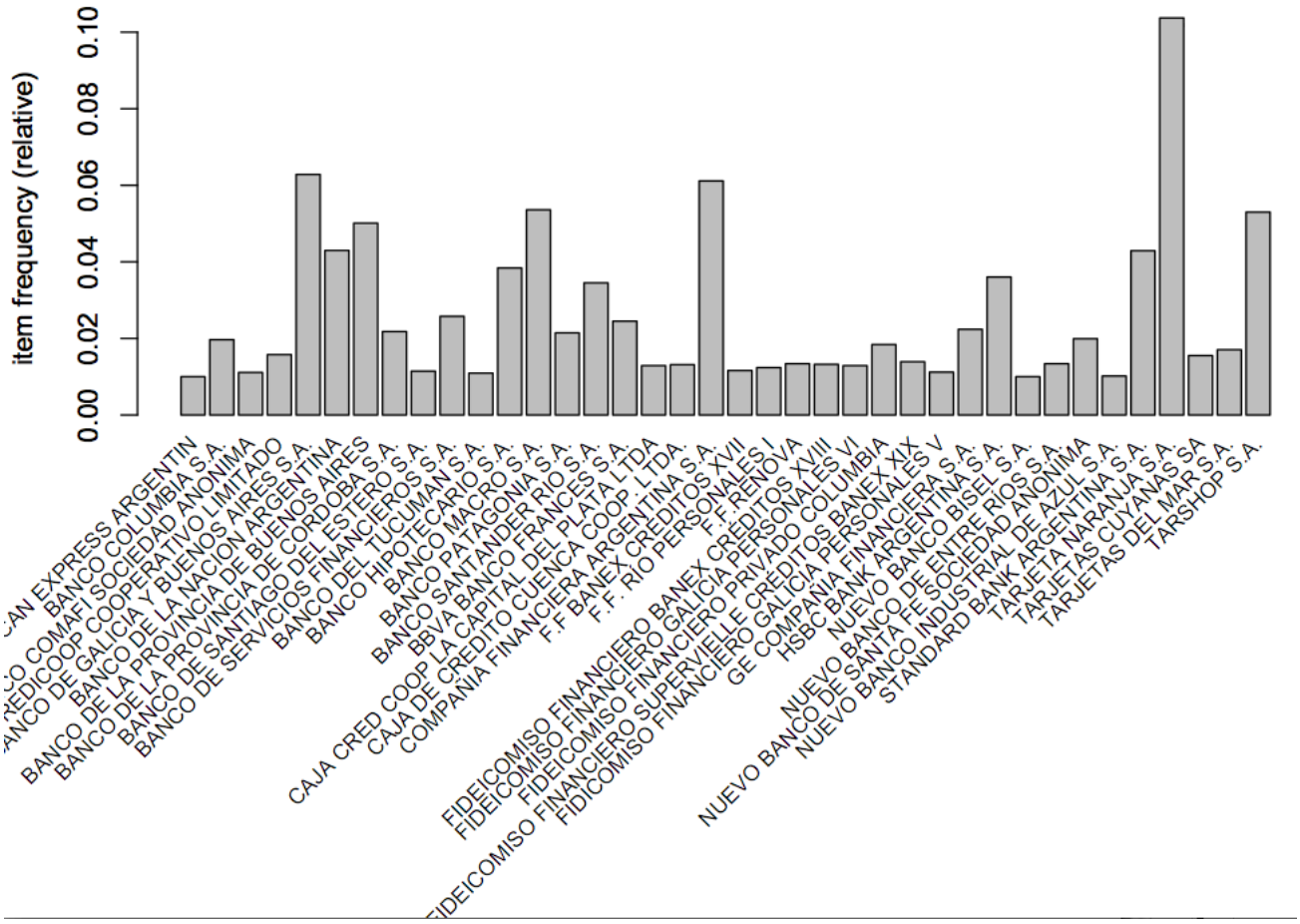
- Ordenamos las 365 reglas, en forma decreciente por soporte.
- Eliminamos las reglas donde los bancos, tanto en el antecedente como en el consecuente, se llamaban igual o tenían una relación basada en el nombre. Por ejemplo la siguiente regla: {FIDEICOMISO FINANCIERO TARJETAS DEL MAR III} => {TARJETAS DEL MAR S.A.}. Decidimos no tomarlas porque deducimos que son entidades ya relacionadas, no nos informarían nada que ya no sepamos.
- Elegimos las 10 primeras reglas según el ordenamiento antes mencionado.

La mayoría de las reglas quedaron asociadas a Banco Galicia y Tarjeta Naranja, debido a que estos dos bancos son los mas frecuentes en el conjunto de datos.
Durante los experimentos habíamos decido no tomar las reglas que tenían estos bancos,

dado que eran los más frecuentes, pero descartamos esta idea porque las siguientes reglas tenían un soporte bajo, y las reglas obtenidas no resultaban significativas.

Usando el R, obtuvimos el siguiente Plot, dónde se puede ver la distribución los bancos:

```
>itemFrequencyPlot(bancos,support=0.1,cex.names=0.8)
```



2 - Comparar la evolución del criterio utilizado en cada una de las reglas a lo largo de todos los meses del dataset.

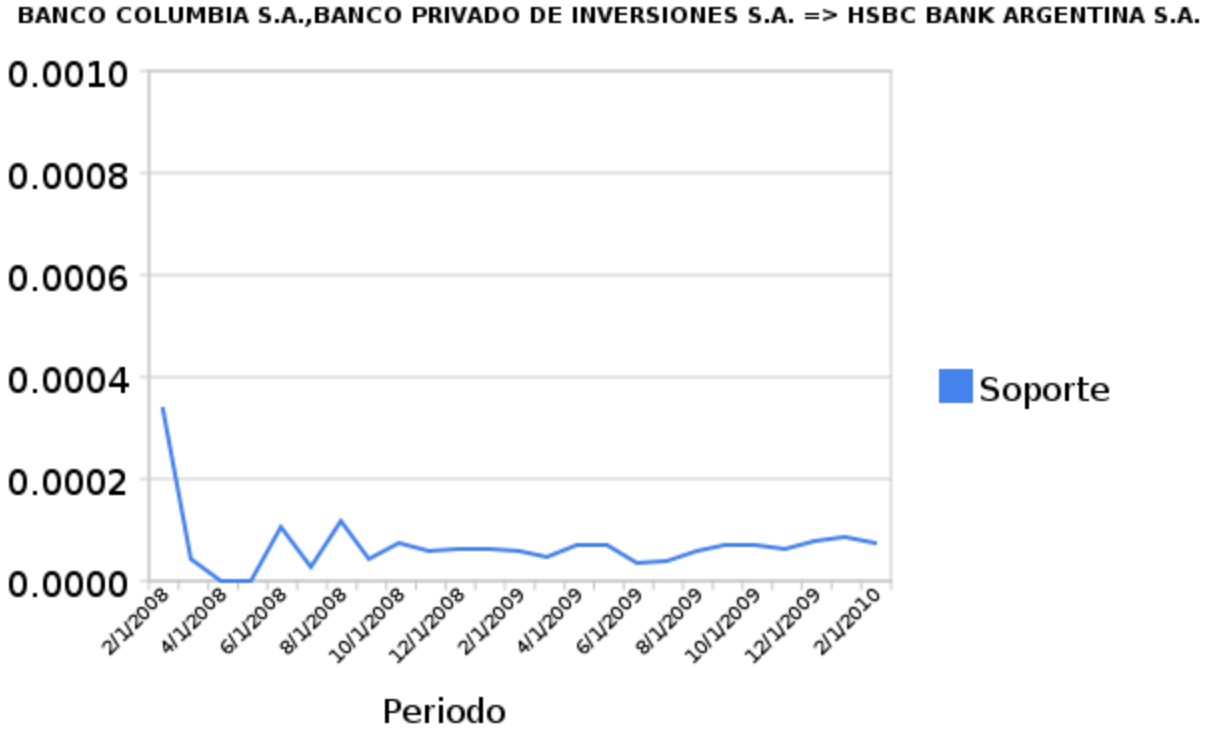
Para comprobar la evolución de las reglas hemos calculado el soporte y la confianza de la regla para cada período a través de una consulta SQL:

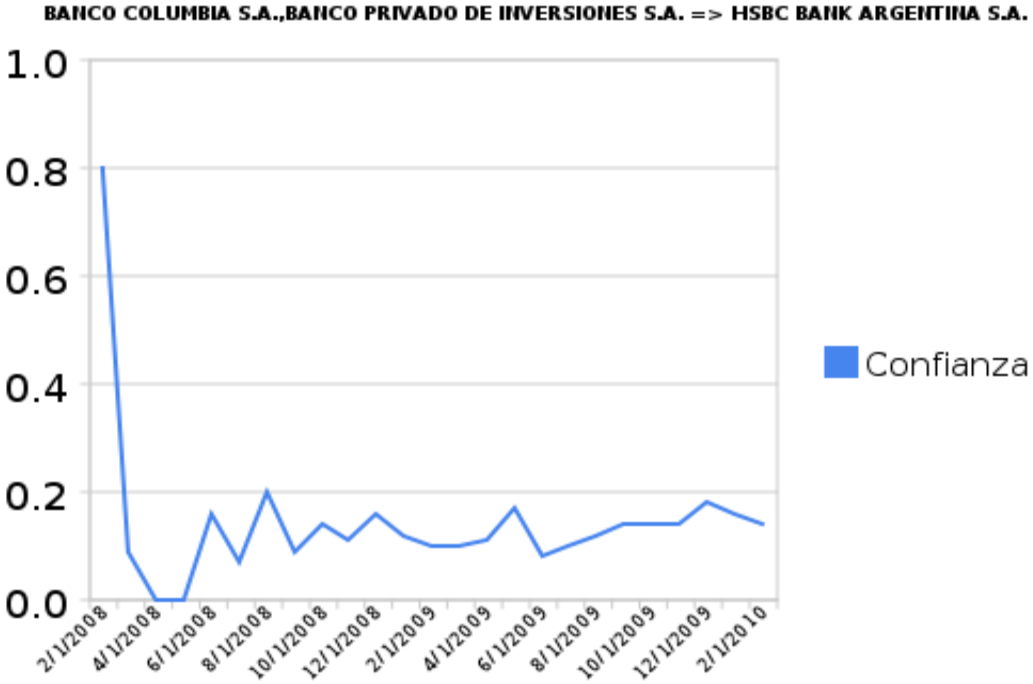
```
select t.anio, t.mes, c.frecuency, t.total, cast(c.frecuency as real) / cast(t.total as real) as support,
cast(c.frecuency as real) /cast(d.frecuency as real) as confidence
from
(select anio, mes, count(distinct id) as total
from cuentas
group by anio, mes) t,
(select c1.anio, c1.mes, count(distinct id) as frecuency
from
cuentas c1
where
 exists (select id from cuentas c2 where c1.id = c2.id and c1.mes = c2.mes and c1.anio = c2.anio and
banco = 'BANCO DE GALICIA Y BUENOS AIRES S.A.')
and exists (select id from cuentas c3 where c1.id = c3.id and c1.mes = c3.mes and c1.anio = c3.anio
and banco = 'BANCO DE LA PROVINCIA DE CORDOBA S.A.')
and exists (select id from cuentas c4 where c1.id = c4.id and c1.mes = c4.mes and c1.anio = c4.anio
and banco = 'TARJETA NARANJA S.A.')
group by c1.anio, c1.mes) c,
(select c1.anio, c1.mes, count(distinct id) as frecuency
from
cuentas c1
where
 exists (select id from cuentas c2 where c1.id = c2.id and c1.mes = c2.mes and c1.anio = c2.anio and
banco = 'BANCO DE GALICIA Y BUENOS AIRES S.A.')
and exists (select id from cuentas c3 where c1.id = c3.id and c1.mes = c3.mes and c1.anio = c3.anio
and banco = 'BANCO DE LA PROVINCIA DE CORDOBA S.A.')
group by c1.anio, c1.mes) d
where t.anio = c.anio
and t.mes = c.mes
and c.anio = d.anio
and c.mes = d.mes
order by t.anio, t.mes
```

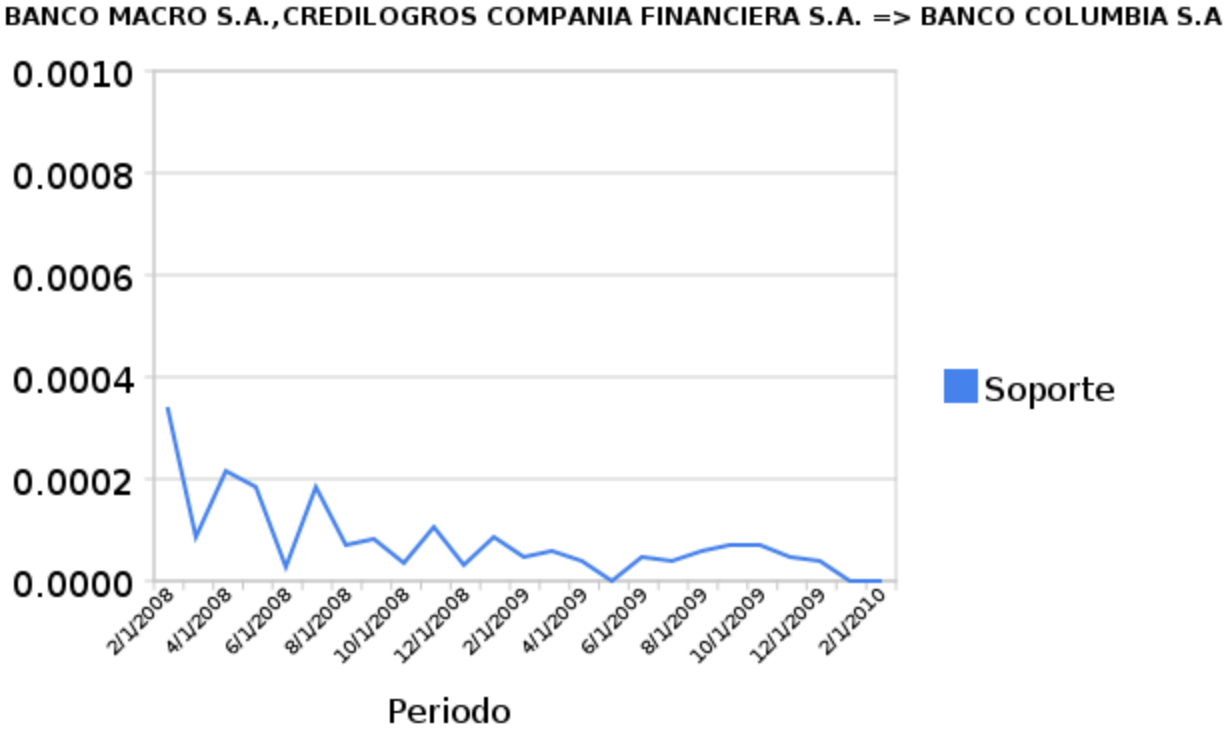
Como resultado de muestran gráficamente la evolución de estos parámetros a lo largo del tiempo.


BANCO DE GALICIA Y BUENOS AIRES S.A.,BANCO DE LA PROVINCIA DE CORDOBA S.A. => {TARJETA NARANJA S.A.}
0.008
0.006
0.004
0.002
0.000
2/1/2008
4/1/2008
6/1/2008
8/1/2008
10/1/2008
12/1/2008
2/1/2009
4/1/2009
6/1/2009
8/1/2009
10/1/2009
12/1/2009
2/1/2010
Periodo
Soporte


BANCO DE GALICIA Y BUENOS AIRES S.A.,BANCO DE LA PROVINCIA DE CORDOBA S.A. => {TARJETA NARANJA S.A.}
1.0
0.8
0.6
0.4
0.2
0.0
2/1/2008
4/1/2008
6/1/2008
8/1/2008
10/1/2008
12/1/2008
2/1/2009
4/1/2009
6/1/2009
8/1/2009
10/1/2009
12/1/2009
2/1/2010
Confianza


BANCO COLUMBIA S.A.,BANCO PRIVADO DE INVERSIONES S.A. => HSBC BANK ARGENTINA S.A.
0.0010
0.0008
0.0006
0.0004
0.0002
0.0000
2/1/2008
4/1/2008
6/1/2008
8/1/2008
10/1/2008
12/1/2008
2/1/2009
4/1/2009
6/1/2009
8/1/2009
10/1/2009
12/1/2009
2/1/2010
Soporte
Periodo

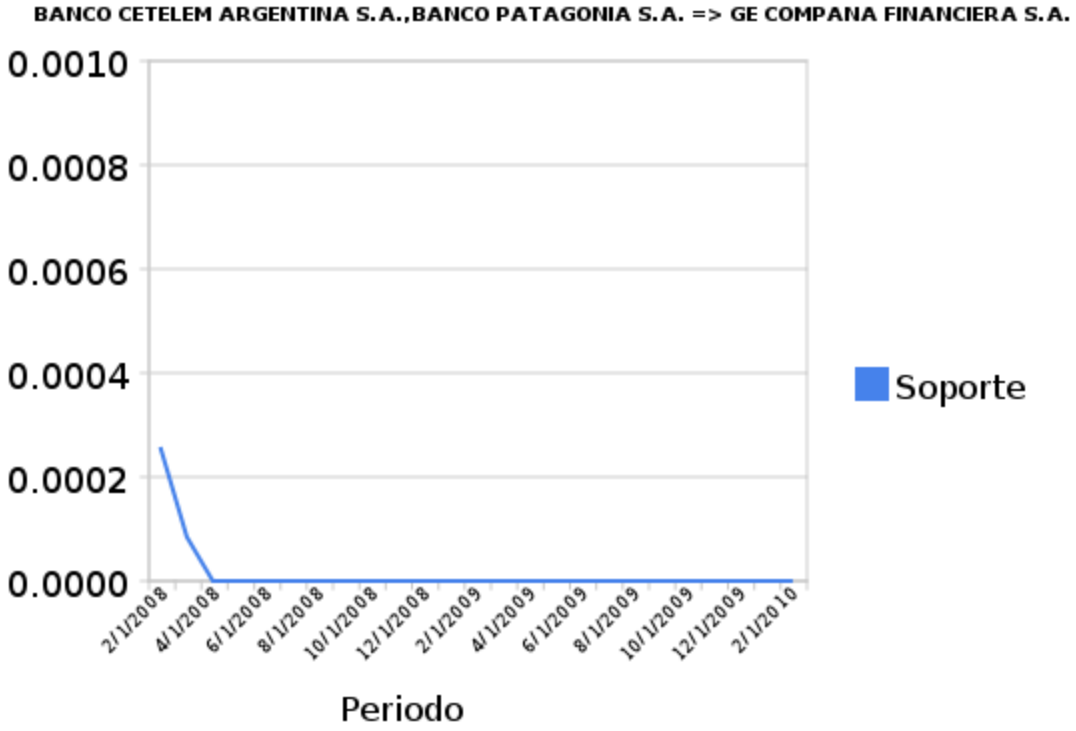

BANCO COLUMBIA S.A.,BANCO PRIVADO DE INVERSIONES S.A. => HSBC BANK ARGENTINA S.A.
1.0
0.8
0.6
0.4
0.2
0.0
2/1/2008
4/1/2008
6/1/2008
8/1/2008
10/1/2008
12/1/2008
2/1/2009
4/1/2009
6/1/2009
8/1/2009
10/1/2009
12/1/2009
2/1/2010
Confianza

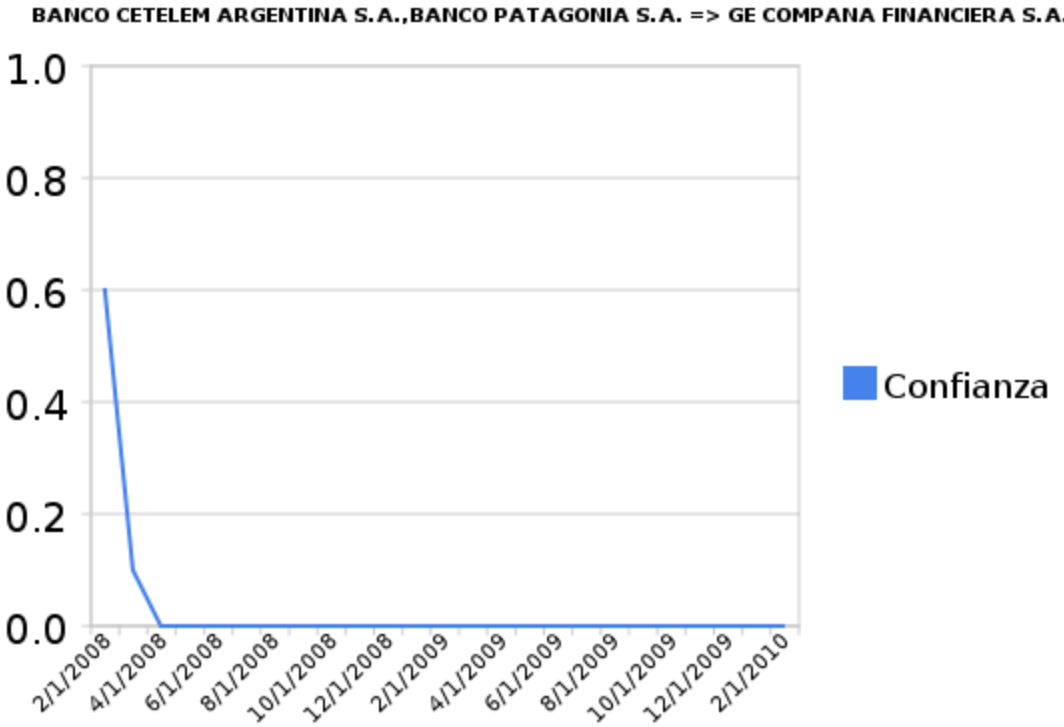

BANCO MACRO S.A.,CREDILOGROS COMPANIA FINANCIERA S.A. => BANCO COLUMBIA S.A
0.0010
0.0008
0.0006
0.0004
0.0002
0.0000
2/1/2008
4/1/2008
6/1/2008
8/1/2008
10/1/2008
12/1/2008
2/1/2009
4/1/2009
6/1/2009
8/1/2009
10/1/2009
12/1/2009
2/1/2010
Soporte
Periodo

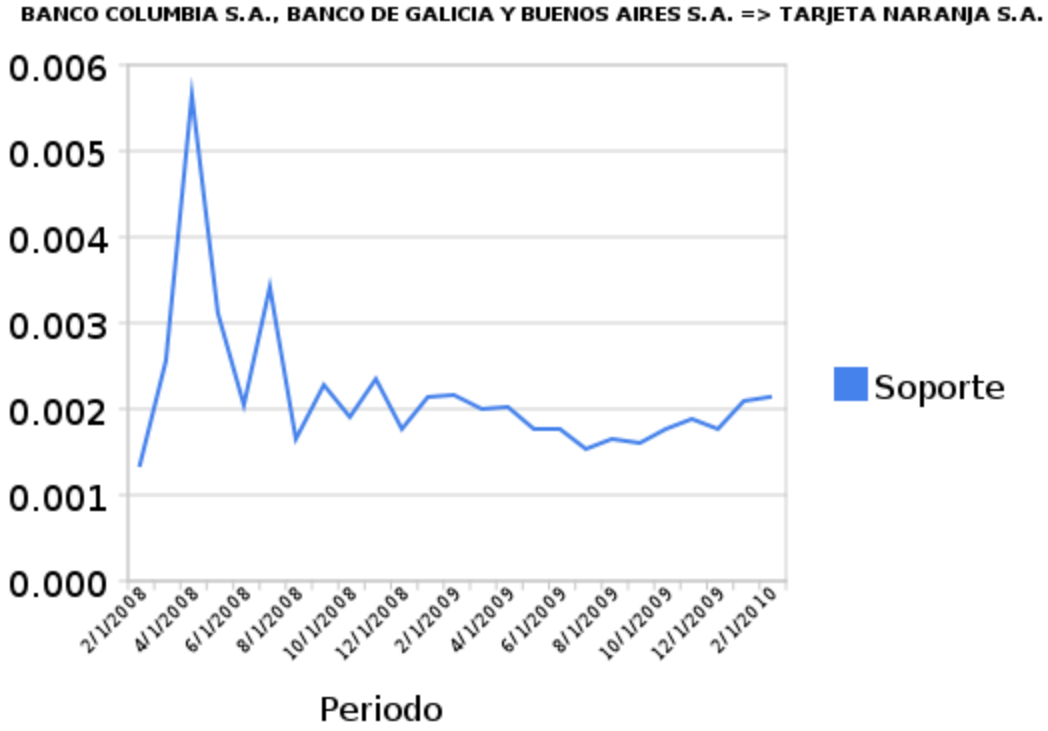

BANCO MACRO S.A.,CREDILOGROS COMPANIA FINANCIERA S.A. => BANCO COLUMBIA S.A
1.0
0.8
0.6
0.4
0.2
0.0
2/1/2008
4/1/2008
6/1/2008
8/1/2008
10/1/2008
12/1/2008
2/1/2009
4/1/2009
6/1/2009
8/1/2009
10/1/2009
12/1/2009
2/1/2010
Confianza

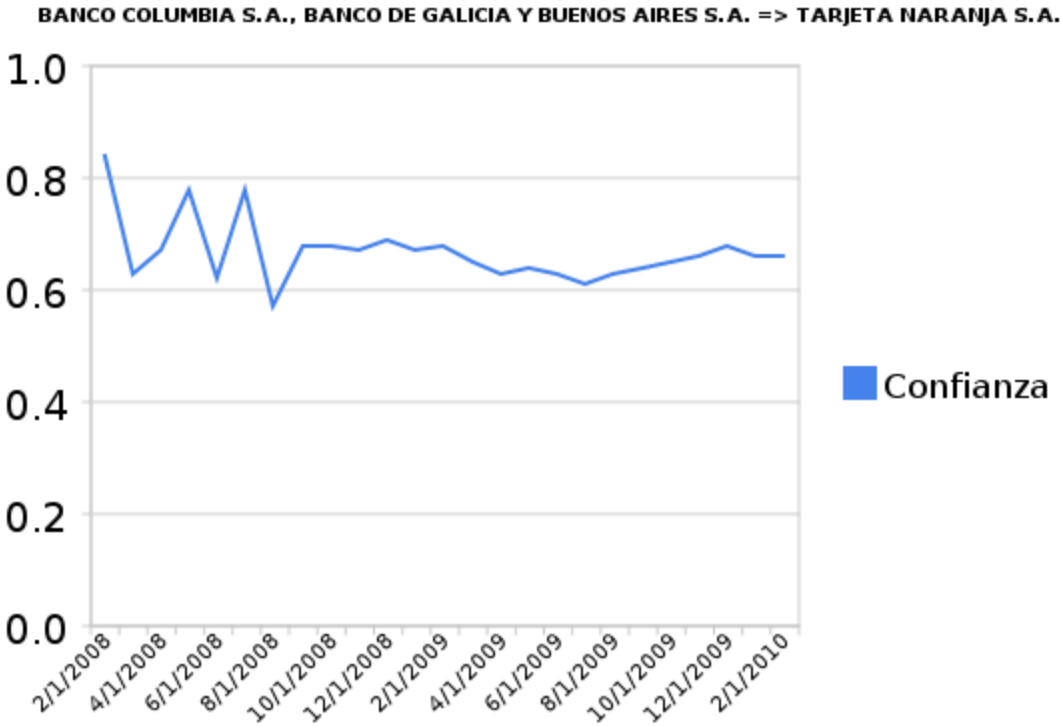

BANCO DE LA NACION ARGENTINA,NUEVA CARD S.A. => BANCO DE LA PROVINCIA DE BUENOS AIRES
0.0010
0.0008
0.0006
0.0004
0.0002
0.0000
2/1/2008
4/1/2008
6/1/2008
8/1/2008
10/1/2008
12/1/2008
2/1/2009
4/1/2009
6/1/2009
8/1/2009
10/1/2009
12/1/2009
2/1/2010
Soporte
Periodo


BANCO DE LA NACION ARGENTINA,NUEVA CARD S.A. => BANCO DE LA PROVINCIA DE BUENOS AIRES
1.0
0.8
0.6
0.4
0.2
0.0
2/1/2008
4/1/2008
6/1/2008
8/1/2008
10/1/2008
12/1/2008
2/1/2009
4/1/2009
6/1/2009
8/1/2009
10/1/2009
12/1/2009
2/1/2010
Confianza


BANCO CETELEM ARGENTINA S.A.,BANCO PATAGONIA S.A. => GE COMPANA FINANCIERA S.A.
0.0010
0.0008
0.0006
0.0004
0.0002
0.0000
2/1/2008
4/1/2008
6/1/2008
8/1/2008
10/1/2008
12/1/2008
2/1/2009
4/1/2009
6/1/2009
8/1/2009
10/1/2009
12/1/2009
2/1/2010
Soporte
Periodo

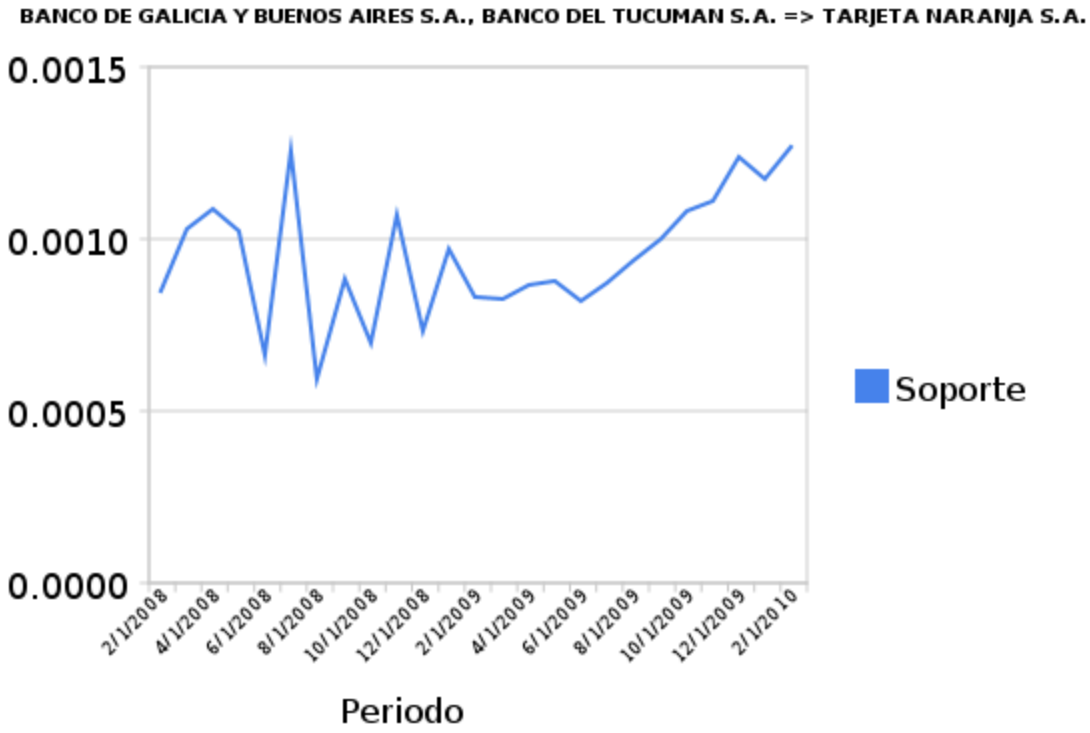

BANCO CETELEM ARGENTINA S.A.,BANCO PATAGONIA S.A. => GE COMPANA FINANCIERA S.A.
1.0
0.8
0.6
0.4
0.2
0.0
2/1/2008
4/1/2008
6/1/2008
8/1/2008
10/1/2008
12/1/2008
2/1/2009
4/1/2009
6/1/2009
8/1/2009
10/1/2009
12/1/2009
2/1/2010
Confianza

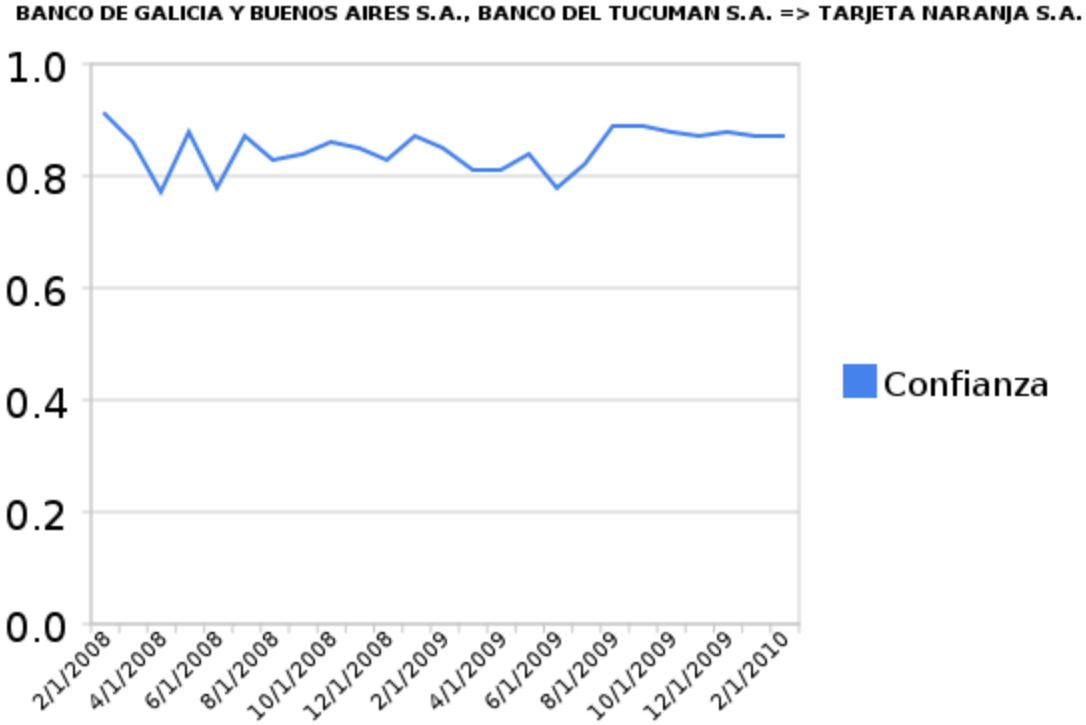

BANCO COLUMBIA S.A., BANCO DE GALICIA Y BUENOS AIRES S.A. => TARJETA NARANJA S.A.
0.006
0.005
0.004
0.003
0.002
0.001
0.000
2/1/2008
4/1/2008
6/1/2008
8/1/2008
10/1/2008
12/1/2008
2/1/2009
4/1/2009
6/1/2009
8/1/2009
10/1/2009
12/1/2009
2/1/2010
Periodo
Soporte


BANCO COLUMBIA S.A., BANCO DE GALICIA Y BUENOS AIRES S.A. => TARJETA NARANJA S.A.
1.0
0.8
0.6
0.4
0.2
0.0
2/1/2008
4/1/2008
6/1/2008
8/1/2008
10/1/2008
12/1/2008
2/1/2009
4/1/2009
6/1/2009
8/1/2009
10/1/2009
12/1/2009
2/1/2010
Confianza

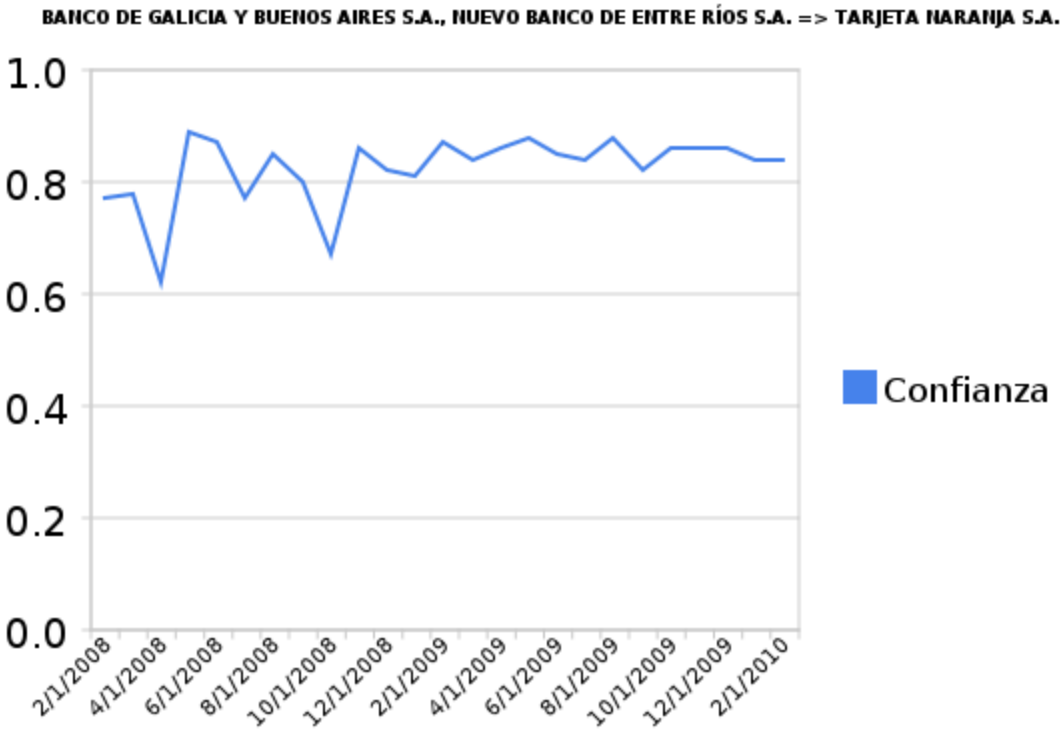

BANCO DE GALICIA Y BUENOS AIRES S.A., COMPAÑIA FINANCIERA ARGENTINA S.A. => TARJETA NARANJA S.A.
0.005
0.004
0.003
0.002
0.001
0.000
2/1/2008
4/1/2008
6/1/2008
8/1/2008
10/1/2008
12/1/2008
2/1/2009
4/1/2009
6/1/2009
8/1/2009
10/1/2009
12/1/2009
2/1/2010
Periodo
Soporte

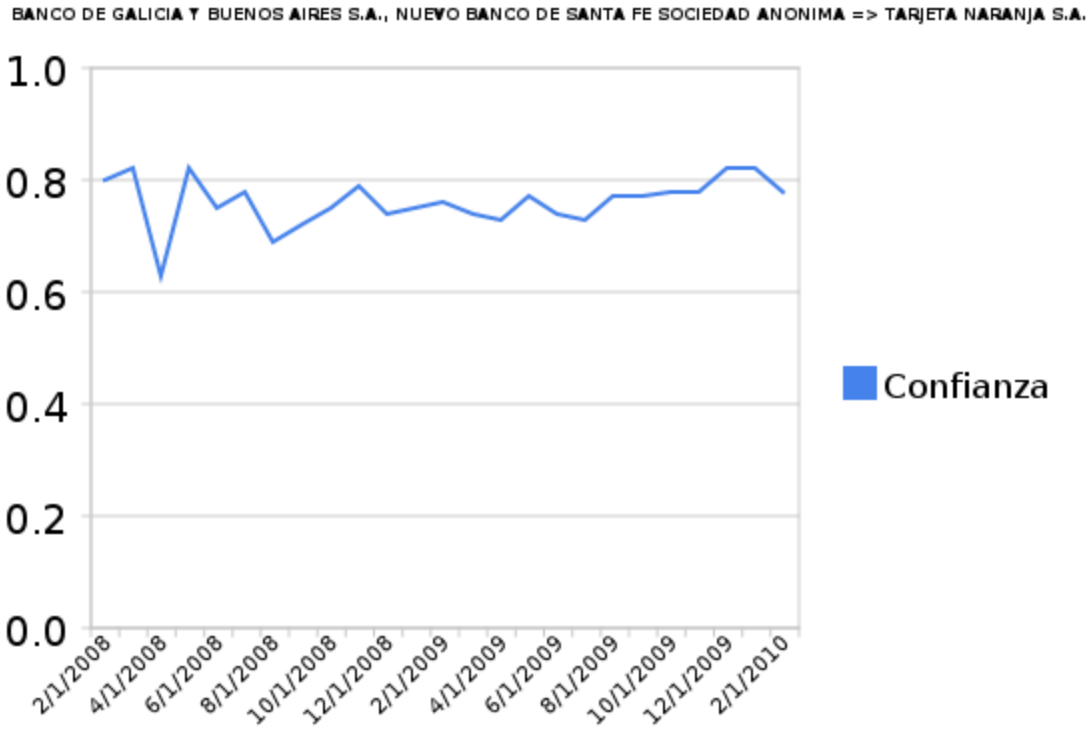

BANCO DE GALICIA Y BUENOS AIRES S.A., COMPAÑIA FINANCIERA ARGENTINA S.A. => TARJETA NARANJA S.A.
1.0
0.8
0.6
0.4
0.2
0.0
2/1/2008
4/1/2008
6/1/2008
8/1/2008
10/1/2008
12/1/2008
2/1/2009
4/1/2009
6/1/2009
8/1/2009
10/1/2009
12/1/2009
2/1/2010
Confianza


BANCO DE GALICIA Y BUENOS AIRES S.A., BANCO DEL TUCUMAN S.A. => TARJETA NARANJA S.A.
0.0015
0.0010
0.0005
0.0000
2/1/2008
4/1/2008
6/1/2008
8/1/2008
10/1/2008
12/1/2008
2/1/2009
4/1/2009
6/1/2009
8/1/2009
10/1/2009
12/1/2009
2/1/2010
Periodo
Soporte


BANCO DE GALICIA Y BUENOS AIRES S.A., BANCO DEL TUCUMAN S.A. => TARJETA NARANJA S.A.
1.0
0.8
0.6
0.4
0.2
0.0
2/1/2008
4/1/2008
6/1/2008
8/1/2008
10/1/2008
12/1/2008
2/1/2009
4/1/2009
6/1/2009
8/1/2009
10/1/2009
12/1/2009
2/1/2010
Confianza


BANCO DE GALICIA Y BUENOS AIRES S.A., NUEVO BANCO DE ENTRE RÍOS S.A. => TARJETA NARANJA S.A.
0.0012
0.0010
0.0008
0.0006
0.0004
0.0002
0.0000
2/1/2008
4/1/2008
6/1/2008
8/1/2008
10/1/2008
12/1/2008
2/1/2009
4/1/2009
6/1/2009
8/1/2009
10/1/2009
12/1/2009
2/1/2010
Soporte
Periodo


BANCO DE GALICIA Y BUENOS AIRES S.A., NUEVO BANCO DE ENTRE RÍOS S.A. => TARJETA NARANJA S.A.
1.0
0.8
0.6
0.4
0.2
0.0
2/1/2008
4/1/2008
6/1/2008
8/1/2008
10/1/2008
12/1/2008
2/1/2009
4/1/2009
6/1/2009
8/1/2009
10/1/2009
12/1/2009
2/1/2010
Confianza



Conclusiones:

1. Las reglas 1 y 8 son las que tienen un comportamiento más homogéneo a lo largo del tiempo.

2. Generalmente las reglas que mejor se comportan son los que contienen Galicia y Tarjeta Naranja en el antecedente, siempre variando con un banco con otro banco. Como el ejemplos podemos decir Banco de Córdoba, Banco de Tucuman, Banco de Santa Fe, Banco de Entre Ríos o Banco Columbia. La mayoría de estos bancos son del interior del País.

3 - En algunos casos las personas presentan “saltos” en los productos que tienen de un mes a otro. Estos saltos están definidos por la aparición en un cierto mes de un producto con el cual la persona no había contado hasta ese momento. Estudiar los casos donde se presentan “saltos” y detectar las características de los mismos.

El analisis de saltos se realizó con un conjunto de querys. Luego se hizo controles para los casos más significativos.
La forma de trabajar fue hacer una consulta a la base de datos para calcular cuales eran los cambios de banco en cada uno de los clientes, mirando dos meses hacia atrás. Esto es, para el mes 6 de 2008, y para un cliente en particular, se miraba si ese cliente tenia en el el mes 6 de 2008 un banco que no habia tenido ni en el mes 4 de 2008 ni en el mes 5 de 2008.

```
select anio, mes ,  banco, count(1)
from
cuentas c1
where
c1.mes > 2
and c1.anio = 2009
and not exists(select 'existe' from cuentas c2
        where c1.id = c2.id
        and c1.anio = c2.anio
        and c1.banco = c2.banco
        and c1.banco - 1 = c2.mes
)
and not exists(select 'existe' from cuentas c2
        where c1.id = c2.id
        and c1.banco = c2.anio
        and c1.banco = c2.banco
        and c1.banco - 2 = c2.mes
        )
group by anio, mes, banco
order by anio, mes, banco, count(1) desc
```

Esto derivó en el analisís de saltos que presentamos a continuación.

a) Banco CREDILOGROS y Banco SERVICIO DE TRANSACCIONES para Enero de 2010.

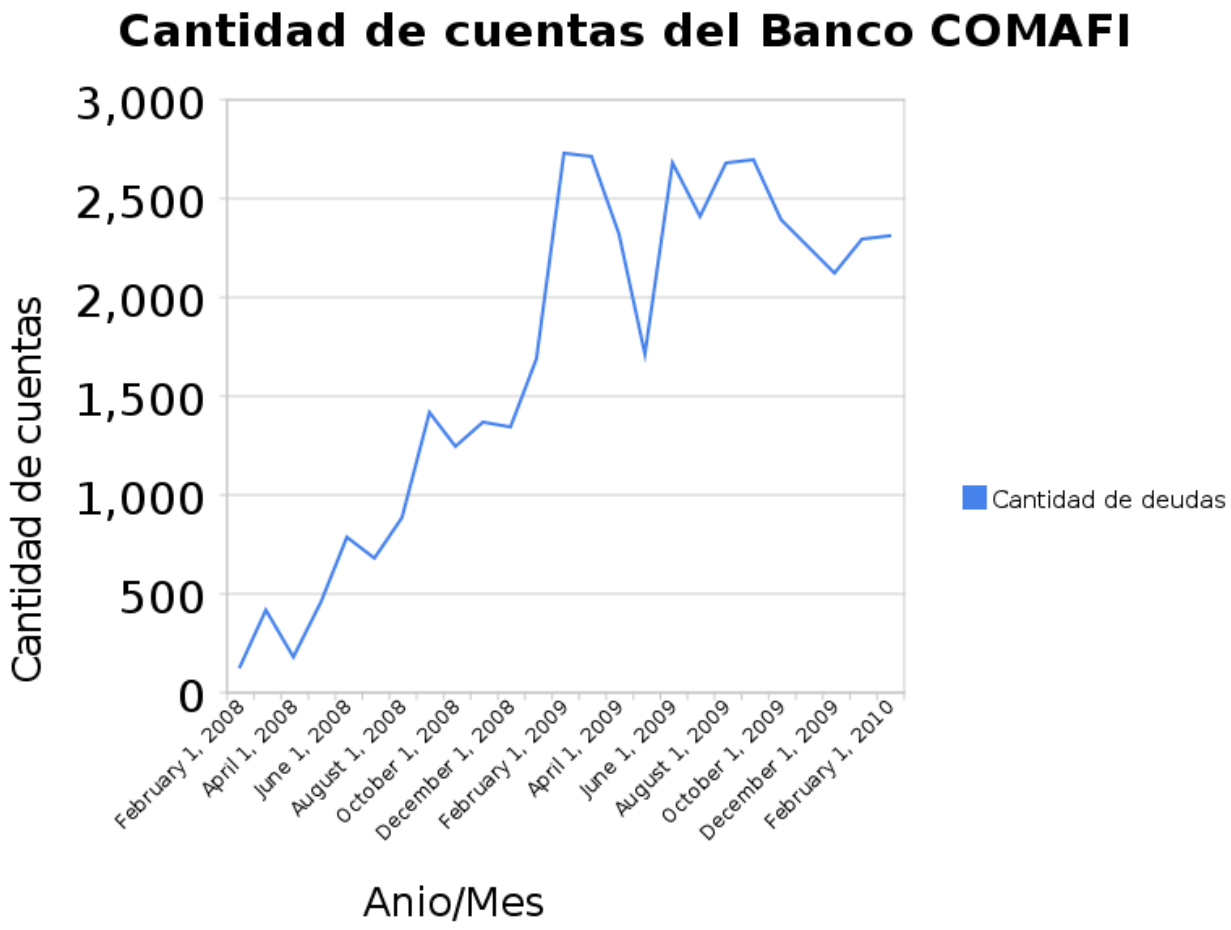
Estamos en un caso de funsion de bancos. Banco SERVICIO DE TRANSACCIONES absorbe el banco CREDILOGROS en Enero de 2010. Esta información, obtenidad de internet, fué la que determinó el claro ejemplo de salto descubierto.

b) Banco COMAFI en febrero de 2009.

Aqui podemos ver un salto importante de Enero a Febrero de 2009. Este es un Banco que presenta deudas para todos los meses. Pero es en febrero de 2009 cuando muestra su pico más alto.

Año/Mes	Cantidad de deudas
February 1, 2008	131
March 1, 2008	417
April 1, 2008	181

May 1, 2008	463
June 1, 2008	785
July 1, 2008	678
August 1, 2008	887
September 1, 2008	1416
October 1, 2008	1244
November 1, 2008	1368
December 1, 2008	1342
January 1, 2009	1689
February 1, 2009	2732
March 1, 2009	2710
April 1, 2009	2316
May 1, 2009	1710
June 1, 2009	2680
July 1, 2009	2413
August 1, 2009	2683
September 1, 2009	2695
October 1, 2009	2395
November 1, 2009	2253
December 1, 2009	2126
January 1, 2010	2297
February 1, 2010	2315



c) Banco Coto Centro Integral de Comercialización. Mayo de 2008.

En este mes, este banco comienza a registrar deudas para muchos clientes. Esto se debe a que es un banco creado para esta fecha, según se puede ver en el siguiente informe: www.bcra.gov.ar/pdfs/snp/SNP3339.pdf

d) Banco FIDEICOMISO FINANCIERO LMF. Agosto de 2008.

Para este caso se empiezan a computar deudas contra este fideicomiso en esta fecha. A partir de allì, hasta la finalización de la info del dataset, este banco estará siempre presente manteniendo similares cantidades.

4 - Analizar cuáles son las características demográficas de los poseedores de productos de cada tipo de banco

Para este punto se trabajo con la tabla Personas. Se realizaron diferentes discretizaciones sobre cada una de las variables. Estas discretizaciones se hicieron teniendo en cuenta la densidad de cada una de esas variables, procurando mantener igual densidad en cada intervalo de la discretización

El paso posterior fue generar diferentes consultas en SQL que servirían como entrada del apriori para poder resolver este punto.

A continuación mostramos algunas de las corridas del apriori y las reglas más significativas elegidas:

```
reglas <- apriori(personas, parameter=list(support=0.1, confidence=0.6))
```

En este caso se encontraron 120 reglas. De esas, rescatamos las siguientes:

Nro	Reglas	Soporte	Confianza	Lift
11	{EDAD_21-32} => {SEXO_M}	0.1204134	0.6374248	1.1085454
14	{EDAD_32-40} => {SEXO_M}	0.1234109	0.6250945	1.0871019
23	{EDAD_40-50} => {SEXO_M}	0.1273730	0.6204061	1.0789483

Estas tres reglas explican algo que tal vez ya sea sabido: Que el sexo masculino es preponderante en el conjunto de datos analizado. Se puede ver que las tres franjas de edad muestran el mismo resultado.

Luego, como la mayoria de las reglas daban como resultado el sexo de las personas, optamos por bajar mucho el soporte para tratar de encontrar reglas que no tuvieran que ver con el sexo de las personas. La siguiente corrida elegida del algoritmo fué:

```
reglas <- apriori(personas, parameter=list(support=0.0001, confidence=0.6))
```

El resultado obtenido se puede detallar, según el R, como se muestra a continuación

```
parameter specification:
 confidence minval smax arem  aval originalSupport support minlen maxlen target   ext
        0.6    0.1    1 none FALSE            TRUE   1e-04      1      5  rules FALSE

algorithmic control:
 filter tree heap memopt load sort verbose
    0.1 TRUE TRUE  FALSE TRUE    2    TRUE

apriori - find association rules with the apriori algorithm
```

version 4.21 (2004.05.09) (c) 1996-2004 Christian Borgelt
set item appearances ...[0 item(s)] done [0.00s].
set transactions ...[65 item(s), 87075 transaction(s)] done [0.05s].
sorting and recoding items ... [59 item(s)] done [0.01s].
creating transaction tree ... done [0.08s].
checking subsets of size 1 2 3 4 5 done [0.02s].
writing ... **[14948 rule(s)]** done [0.01s].
creating S4 object ... done [0.03s].

Aquí, como podemos ver, el número de reglas obtenidas es significativo. Entonces, para tratar de achicar el scope de búsqueda de reglas, hicimos un pre-procesamiento de las reglas obtenidas a través del R mismo. Para ello aplpicamos los siguientes pasos:

```
reglas_null <- subset(reglas, subset = rhs %pin% 'NULL')
```

Nos quedamos con las reglas que tienen NULL en su lado derecho.Luego calculamos el opuesto a través de R como se muestra a continuación:

```
reglas_nonull <- setdiff(reglas, reglas_null)
```

Así obtuvimos las reglas que no tienen NULL en el consecuente. El siguiente paso, fué, como ya dijimos, tratar de quedarnos solo con las reglas que no daban información sobre el SEXO de la personas. Para ello volvimos a aplicar un algoritmo similar al contado hasta aquí. Con:

```
reglas_nosexo <- subset(reglas_nonull, subset = rhs %pin% 'SEXO')
```

Me quedo solo con las reglas que tinen NULL en el consecuente. Y luego me quedo con el conjunto opuesto. Esto es con las reglas que no profesan SEXO en el consecuente:

```
reglas_xxx <- setdiff(reglas_nonull, reglas_nosexo)
```

WRITE(reglas_xxx, file='reglas_personas_reducido_no_null_no_sexo.csv', sep=';', col.names=NA)
Finalmente tomamos esas reglas y las exportamos a un archivo .csv, dónde hacemos un analisís regla por reglas, quedándonos con las que consideramos más relevantes.

La cantidad de reglas obtenidas fueron 78. De esas 78 nos quedamos con las siguientes:

116	{AUTO_ANIO_2006,EDAD_21-32,PROVINCIA_BUENOS AIRES,SEXO_M} => {AUTO_PRECIO_25000-30000}	0.00015	0.63	47.74

En esta reglas podemos ver que una persona con domicilio en Buenos Aires, de Sexo masculino, y que tiene un auto modelo 2006, entonces el precio de ese auto puede variar entre 25000 y 30000 pesos.

105	{AUTO_PRECIO_35000-40000,MONTO_>10000,PROVINCIA_CAPITAL FEDERAL,SEXO_M} => {EDAD_40-50}	0.00049	0.62	3.02

En esta regla podemos ver que los hombres domiciliados en capital deferal con un auto valuado entre 35000 y 40000, y cuyo sueldo es mayor a los 10000 pesos por mes, tienen entre 40 y 50 años.

116	{AUTO_PRECIO_25000-30000,MONTO_5000-10000,PROVINCIA_CAPITAL FEDERAL,SEXO_F} => {EDAD_50-60}	0.000115	0.63	3.71

Aquí podemos ver que las mujeres de Capital Federal, que ganan entre 5000 y 10000 pesos, y que poseen auto con precio en un rango de 25000 y 30000 pesos, tienen entre 50 y 60 años.

626	{EDAD_21-32,MONTO_>10000,SEXO_F} => {PROVINCIA_CAPITAL FEDERAL}	0.000103	0.6	3.94

Esta reglas nos dice que las mujeres de entre 21 y 32 años, que tienen un sueldo mayor a 10000 pesos, estan domiciliadas en Capital Federal.

525	{AUTO_ANIO_2006,MONTO_>10000,SEXO_F} => {AUTO_PRECIO_40000-50000}	0.000138	0.6	47.76

Esta reglas nos dice que las mujeres que ganan más de 10000 pesos y que tienen un auto, que es modelo 2006, entonces el precio de ese auto varia entre 40000 y 50000 pesos.

Para el siguiente experimento se filtro el conjunto de personas quedandose con aquellas que si informaban en la variable afip_activo. Esto se hizo generando un nuevo dataset, aprtir de una nueva consulta SQL, dónde se filtraron solo las personas dónde la columna AFIP estaba informada. Luego se corrio el algortimo a priori con los siguientes parámetros:

```
reglas <- apriori(personas, parameter=list(support=0.7, confidence=0.9))
```

1	{} => {AFIP_ACTIVO_TRUE}	0.9999244	0.9999244	1.000000

Esta regla muestra el item más frecuente. Podemos ver que que el soporte es igual a la confianza, una caracterítica que se da cuando la regla es de tamaño 1. Aquí se ve que el 99% de las personas si informan que estan activos en AFIP.

2	{MONTO_NULL} => {AFIP_ACTIVO_TRUE}	0.704775	1.0000000	1.000076

Esta regla nos dice que las personas que informa sobre su actividad en AFIP, no informan el monto de su sueldo.

EL siguiente experimento fué bajar aún más el soporte para ver que otras reglas se podían tomar:

Luego bajamos amboas parámetros para poder encontrar más reglas

```
reglas <- apriori(personas, parameter=list(support=0.1, confidence=0.6))
```

1	{} => {SEXO_M}	0.6750302	0.6750302	1.000000

Esta regla nos dice que él 60% de las personas pertenecen al Sexo masculino.

2	{AFIP_EMPLEADOR_TRUE} => {SEXO_M}	0.1200877	0.7216167	1.069014

Esta reglas nos dice que las personas que son empledoras son de sexo masculino.

3	{SEXO_F} => {MONOTRIBUTO_TRUE}	0.2287636	0.7051013	1.1.186256

Esta regla asegura que las mujeres son monotributistas.

4	{ACTIVIDAD1_K} => {AFIP_SOCIEDADES_N}	0.1069377	0.9751895	1.8118010

K- Servicios inmobiliarios, empresariales y de alquiler

A continuacion se lista el SQL que muestra las discretizaciones efectuadas para cada una de las variables :

```
--Edad
case
when (current_date - fnacimiento )/ 365 >= 100 then 'EDAD_NULL'
when (current_date - fnacimiento )/ 365 >= 60 then 'EDAD_>60'
when (current_date - fnacimiento )/ 365 >= 50 then 'EDAD_50-60'
when (current_date - fnacimiento )/ 365 >= 40 then 'EDAD_40-50'
when (current_date - fnacimiento )/ 365 >= 32 then 'EDAD_32-40'
when (current_date - fnacimiento )/ 365 >= 21 then 'EDAD_21-32'
else 'EDAD_MENOR'
end as edad,

--Actividad 1 y 2
case
when actividad1 between 11111 and 20390 then 'ACTIVIDAD1_A'
when actividad1 between 50110 and 50300 then 'ACTIVIDAD1_B'
when actividad1 between 101000   AND 142900 then 'ACTIVIDAD1_C'
when actividad1 between 151111   and 372000 then 'ACTIVIDAD1_D'
when actividad1 between 401110   and 410200 then 'ACTIVIDAD1_E'
when actividad1 between 451100   and 455000 then 'ACTIVIDAD1_F'
when actividad1 between 501110   and 526909 then 'ACTIVIDAD1_G'
when actividad1 between 551100   and 552290  then 'ACTIVIDAD1_H'
```

```
when actividad1 between 601100   and 642099 then 'ACTIVIDAD1_I'
when actividad1 between 651100   and 672200 then 'ACTIVIDAD1_J'
when actividad1 between 701010   and 749900 then 'ACTIVIDAD1_K'
when actividad1 between 751100   and 753000 then 'ACTIVIDAD1_L'
when actividad1 between 801000   and 809000 then 'ACTIVIDAD1_M'
when actividad1 between 851110   and 853200 then 'ACTIVIDAD1_N'
when actividad1 between 900010   and 930990 then 'ACTIVIDAD1_O'
when actividad1 between 950000   and 950000 then 'ACTIVIDAD1_P'
when actividad1 between 990000   and 990000 then 'ACTIVIDAD1_Q'
when actividad1 is null then 'ACTIVIDAD1_NULL'
else 'ACTIVIDAD1_NA'
end as actividad1,

--Antiguedad monotributo
case
when (current_date - mono_fdesde )/ 365 >= 10 then 'MONO_ANTIGUEDAD_>10'
when (current_date - mono_fdesde )/ 365 >= 5 then 'MONO_ANTIGUEDAD_5-10'
when (current_date - mono_fdesde )/ 365 >= 2 then 'MONO_ANTIGUEDAD_2-5'
when (current_date - mono_fdesde )/ 365 >= 0 then 'MONO_ANTIGUEDAD_<2'
else 'MONO_ANTIGUEDAD_NULL'
end as mono_antiguedad,

--Antiguedad autonomo
case
when (current_date - auto_fdesde )/ 365 >= 20 then 'AUTO_ANTIGUEDAD_>20'
when (current_date - auto_fdesde )/ 365 >= 15 then 'AUTO_ANTIGUEDAD_15-20'
when (current_date - auto_fdesde )/ 365 >= 10 then 'AUTO_ANTIGUEDAD_10-15'
when (current_date - auto_fdesde )/ 365 >= 5 then 'AUTO_ANTIGUEDAD_5-10'
when (current_date - auto_fdesde )/ 365 >= 2 then 'AUTO_ANTIGUEDAD_2-5'
when (current_date - auto_fdesde )/ 365 >= 0 then 'AUTO_ANTIGUEDAD_<2'
else 'AUTO_ANTIGUEDAD_NULL'
end as auto_antiguedad,

--Monto estimado
case
when monto_estimado >= 10000 then 'MONTO_>10000'
when monto_estimado >= 5000 then 'MONTO_5000-10000'
when monto_estimado >= 4000 then 'MONTO_4000-5000'
when monto_estimado >= 3000 then 'MONTO_3000-4000'
when monto_estimado >= 2000 then 'MONTO_2000-3000'
when monto_estimado >= 1000 then 'MONTO_1000-2000'
when monto_estimado >= 500 then 'MONTO_500-1000'
when monto_estimado >= 0 then 'MONTO_<500'
else 'MONTO_NULL'
end as monto_estimado,

--Auto precio
case
when auto_precio >= 50000 then 'AUTO_PRECIO_>50000'
when auto_precio >= 40000 then 'AUTO_PRECIO_40000-50000'
when auto_precio >= 35000 then 'AUTO_PRECIO_35000-40000'
when auto_precio >= 30000 then 'AUTO_PRECIO_30000-35000'
when auto_precio >= 25000 then 'AUTO_PRECIO_25000-30000'
when auto_precio >= 0 then 'AUTO_PRECIO_<25000'
```

```
else 'AUTO_PRECIO_NULL'
end as auto_precio,
```