Proyecto Final
Detección de fraude en tarjeta de crédito

Reconocimiento de Patrones

Autores

Ignacio Gómez
Ana Clara Rodríguez

Reconocimiento de Patrones
Instituto de Ingeniería Eléctrica
Montevideo, Uruguay

Diciembre, 2016

Contents

Chapter 1

Introducción

1.1 Presentación del problema

La industria de los medios de pago está cada vez más presente en el día a día de las personas. Actualmente, es normal que una persona tenga un plástico para realizar compras diarias en un supermercado, así como también realizar compras en internet. La información de la tarjeta viaja por distintos medios, donde ésta puede ser interceptada por un agente y éste puede utilizar dichos datos para realizar una suplantación de identidad, o vender esa información, para que sean realizadas transacciones a través de la tarjeta de la víctima.

El crecimiento de estas transacciones ilegítimas, llamados fraudes, viene acompañado al aumento del volumen transaccional de los últimos años, por lo que es necesario realizar un monitoreo de dicho volumen para mitigar las pérdidas que pueden llegar a generar a una persona, banco, o cualquier entidad financiera relacionada.

Al referirse a costos relacionados a una institución financiera, se pueden observar distintos tipos, uno está relacionado a la pérdida directa a nivel monetario ya que deberá absorber el monto del fraude cometido, y otro es costo a nivel operativo que implica el monitoreo de las transacciones. Este último es importante en cuanto a cantidad de personal que debe disponer la organización para realizar el monitoreo. Como no es factible validar cada transacción manualmente, las entidades financieras compran herramientas de monitoreo automático, que generan alertas por si una transacción cumple con determinado patrón que determina que tiene una alta probabilidad de que sea fraude.

PayGroup, es una empresa que se dedica a generar y vender este tipo de herramientas de monitoreo, con foco en generar modelos predictivos para la detección de fraude en medios de pago.

Para esta investigación, se provee por dicha empresa un set de datos de una entidad financiera de Brasil, con información transaccional de sus tarjetahabientes, con un año de histórico que parte de abril del año 2015 hasta marzo del año 2016. Estos datos están etiquetados, donde se describe si una transacción es legítima o si en realidad posee un fraude asociado.

Al estar estos datos etiquetados, el problema se aborda como un problema de aprendizaje supervisado.

1.1.1 Alcance

Para esta investigación, se analizan tarjetas de crédito cuyo método de entrada fuera manual, banda o comercio electrónico. En la siguiente figura se tiene un análisis por modo de entrada donde se puede ver la proporción de fraude sobre las transacciones legítimas. Las transacciones con chip fueron descartadas del análisis por tener una proporción de casi 40.000 legítimas por fraude.

Modo de entrada	Subtotal Trxs	Subtotal Fraude	Proporcion
Manual	3.265.154	19.494	167
Banda	11.340.082	47.869	237
Comercio Electrónico	9.628.631	20.171	477
Chip	104.458.658	2.668	39.152

Son consideradas sólo las operaciones aprobadas, ya que la transacciones denegadas no ofrecen daño a la entidad financiera. Una transacción denegada es una operación que no fue permitida por el autorizador de las transacciones. Además, se limita el análisis a las tarjetas con más de una transacción en promedio al mes. Esta consideración se debe a que los algoritmos que se utilizan para el clustering y clasificación, utilizan el histórico transaccional de las tarjetas.

1.1.2 Objetivo

El enfoque es generar una o varias agrupaciones de tarjetas para la clasificación, es decir, se generan clusters en base a características de las tarjetas, buscando generar perfiles de comportamiento y que luego estos perfiles sean utilizados para segmentar la clasificación por cada uno de ellos.

Chapter 2

Etapa Inicial

2.1 Limpieza de datos

En una transacción, existen atributos o características que pueden contener valores nulos por motivos. En bases de datos reales existen atributos que caen en desuso por lo que su valor por defecto es nulo. De los atributos activos existen mandatorios y no mandatorios, se estudia el porcentaje de nulos de los no mandatorios. Por último, el caso más complicado es la evaluación de limpieza de datos es la coherencia en la instancia en relación al atributo, por ejemplo, montos negativos.

Entonces, como etapa inicial de la limpieza, fueron quitados los atributos donde éstos contenían un alto porcentaje de valores nulos o era completamente nulos.

Como segunda y última etapa, se realizó la reconstrucción de las características que en tenían un porcentaje pequeño de valores nulos, si era posible la inferencia del valor que debía contener para cada transacción. Por ejemplo, existe una característica que es la hora de la transacción y otro que es la fecha y hora de la transacción. En algunas transacciones, el valor de la hora no estaba presente, pero fue reconstruido gracias a la otra característica.

2.2 Selección semántica de características

Existían características que no aportan para el problema de clasificación, o generan un sesgo al clasificador. Por ejemplo, características que contienen siempre el mismo valor, por lo que no iban a ayudar a la discriminación de fraude-legítima. Un ejemplo es el código de respuesta, donde identifica si una transacción fue denegada o aprobada. En este caso, eran todas transacciones aprobadas.

2.3 Generación de conjuntos

Se generan tres conjuntos, ellos son los conjuntos de entrenamiento, validación y prueba para clustering en base al conjunto total de tarjetas de crédito. El 60% de las tarjetas para training, el 20% para cross validation y el 20% restante para testing. Luego, en base a dichas tarjetas, se generan los archivos de entrenamiento, validación y prueba en base a las transacciones de las tarjetas de cada uno de los conjuntos.

Chapter 3

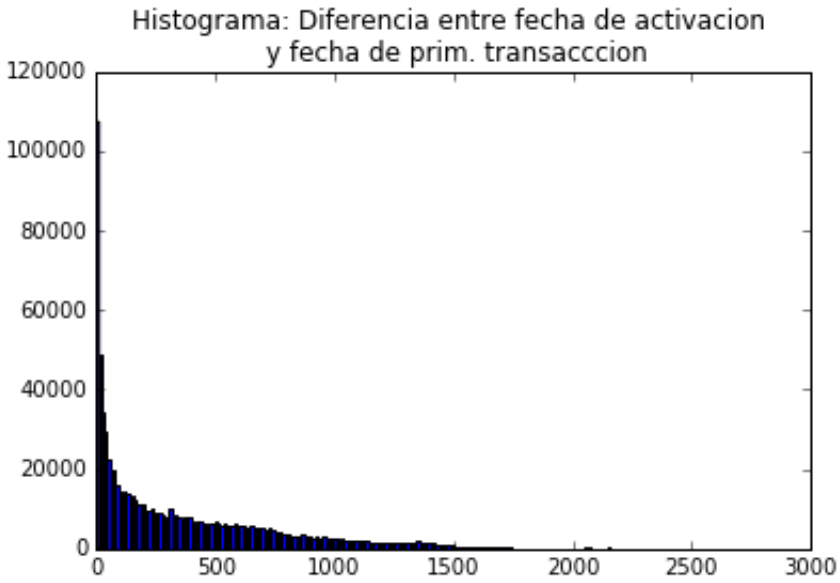
Clustering

3.1 Extracción de características

Se crean atributos para el clustering basado en el enfoque de tarjetas donde cada uno de los atributos identificaran a la tarjeta y no a la transacción.

- Producto
- Tarjeta
- Uso semanal
- Monto promedio
- Monto promedio por rubro del comercio
- Desviación standard del monto por rubro del comercio
- Diferencia entre fecha de activación de la tarjeta y fecha de primer transacción
- Diferencia entre fecha de activación de la tarjeta y fecha de emisión
- Rubro del comercio

Los atributos tales como uso semanal, monto promedio semanal y las diferencias entre fechas; tienen el siguiente comportamiento:



Para mitigar este tipo de distribuciones aplicamos dos métodos: generar variables por quantiles y aplicar logaritmo. Se generan quartiles y la media para las variables "Uso semanal", "Monto promedio". Para los quartiles se utilizó la definición:

$$Q_f(p) = \{x : P(X \leq x) = p\} \, \forall \, p \in \{0.25, 0.50, 0.75\} \tag{3.1}$$

Mientras que la transformación de logaritmo se le aplicó a todas las variables continuas.

3.2 Determinación del tipo de clustering

El tipo de clustering que se debe utilizar, es el basado en prototipo, ya que lo que es interesante es buscar conjuntos de tarjetas con comportamiento parecido. La tarjeta prototipo -o centroide-, será la que represente al conjunto, o cluster. En particular, se opta por utilizar el algoritmo de clustering K-Means donde la cantidad de clusters es un meta atributo.

3.3 Métrica utilizada

La métrica utilizada para el algoritmo de K-Means fue la métrica de Kullback–Leibler, que se define de la siguiente manera:

$$d(p, q) := \sum_i p_i \frac{p_i}{q_i} \tag{3.2}$$

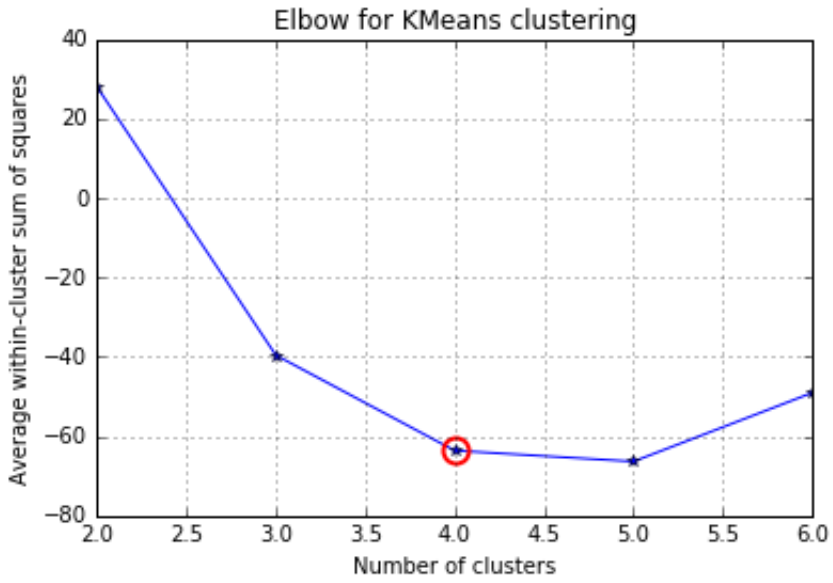
Al ser una métrica no simétrica, la simetrizamos de la siguiente manera:

$$d_s(p, q) := d(p, q) + d(q, p) \tag{3.3}$$

Se aplica esta métrica debido a que los atributos tienen distintas escalas. Por definición, esta métrica es una medida de información de cada una de las instancias de un atributo, esto es, un histograma. De esta forma, no son consideradas las escalas para medir las distancias sino solo la cantidad de instancias. Al introducir dicha métrica, introducimos el problema de las instancias con valor cero -ya que queda con distancia indefinida-, se definió para esto el suavizado de Dirichet.

3.4 Determinación del k óptimo

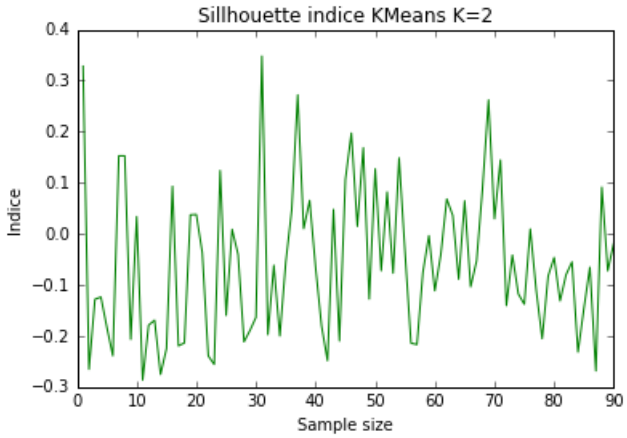
Se realiza la regla del Codo para determinar el k óptimo



Como expresa la gráfica, el k óptimo a utilizar es 4. Sin embargo, generando los conjuntos de validación y prueba a partir de entrenamiento, fue detectado que no existen tarjetas asociadas a más de dos clusters. Por lo tanto, se decide que la mejor solución para este punto es utilizar dos clusters, donde todos los clusters tienen tarjetas asociadas.

3.5 Evaluación del algoritmo

Se generaron dos clusters cuyo tamaño es para el cluster 0 de 995.162 muestras, mientras que el cluster 1 de 253.143 muestras. Como el agrupamiento es no supervisado, es necesario determinar si los clusters generados fueron los correctos. Para ello, se utiliza como herramienta de validación el índice de Silueta. Seleccionando submuestreos para el cálculo del índice obtenemos el siguiente resultado:



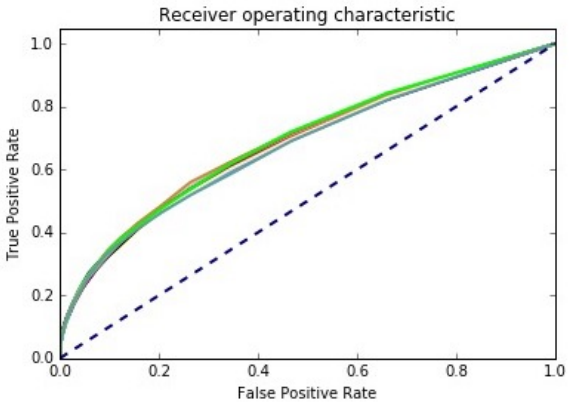
Presenta una alta variabilidad además de un índice en calidad de cluster aleatorio.

Chapter 4

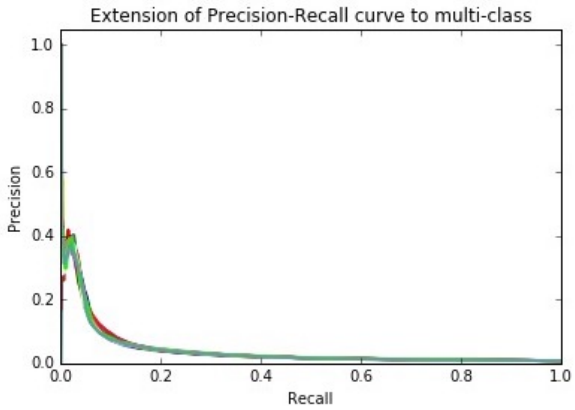
Clasificación

4.1 Modelo base

Para poder realizar una evaluación de si los procesos de extracción, selección de características y clustering han retornado mejoras en la construcción del modelo, es necesario poseer un punto de partida para poder contrastar los resultados encontrados. Este punto de partida es a lo que se llama Modelo base, donde participan las mismas transacciones de entrenamiento y testing que en los demás experimentos, pero con los atributos primarios, estos atributos primarios se componen solamente de los atributos propios de la transacción que no contengan datos nulos. En este modelo, no fue aplicado ningún tipo de clustering. Por último, el algortimo de clasificación para realizar la clasificación fue `RandomForest`. A continuación, se despliegan los resultados de la clasificación:



(a) PRC Modelo Base



(b) ROC Modelo Base

Table 4.2: Cluster 0: entrenamiento contra test

Recall	FP
0.100	132.819
0.824	107.470
0.692	90.328
0.588	78.966
0.520	68.229
0.462	59.231
0.413	51.491
0.369	44.963
0.333	39.019
0.294	34.604
0.261	30.524
0.223	27.752
0.200	24.129
0.174	21.408
0.153	18.805
0.131	16.879
0.115	14.845
0.98	13.25
0.85	11.568
0.75	10.144
0.67	8.898
0.59	7.772
0.54	6.585
0.49	5.620
0.47	4.658
0.44	4.116
0.41	3.6
0.36	3.483
0.33	3.233
0.31	3.134
0.27	2.893
0.25	2.950
0.23	2.781
0.22	2.707
0.20	2.666
0.17	2.763
0.16	2.843
0.14	2.946
0.12	3.020
0.11	3.066
0.10	3.095
0.9	3.105
0.8	3.151
0.7	3.206
0.7	2.785
0.6	2.652
0.5	2.578
0.4	2.6
0.3	2.636
0.2	2.222
0.1	1
0.0	1

La línea base tiene un promedio de la curva PRC de 0.04 mientras que de curva ROC tiene 0.68, en promedio.

4.2 Extracción y selección de características

4.2.1 GainRatio

Se utiliza la selección a través de `GainRatio` sobre cada uno de los clusters y se eliminan para cada uno de ellos, las características que no brindaban ninguna ganancia de información. Para el cluster 1, fueron 10 variables; mientras que para el cluster 0 fueron 6 variables.

4.2.2 Selección de características

Para los datos transacciones, se crea el atributo de pertenencia de cluster que proviene del análisis anterior.

Además, agregamos las siguientes variables booleanas:

- Supera el monto promedio más la desviación estándar
- Supera el monto promedio más la desviación estándar por rubro de comercio
- Alguna vez compró en este rubro de comercio
- Compró en los últimos 6 meses en este rubro de comercio
- Alguna vez compró con esa moneda.
- Compró con esa moneda en los últimos 6 meses
- Alguna vez tuvo una transacción de test.
- Cantidad de test que se le hicieron a la tarjeta
- Si es una nueva franja horaria
- Si es una nueva franja horaria en 6 meses

El atributo que describe si contiene transacciones de test significa si existieron operaciones menores a 10 reales y a la que luego se le efectuó un reverso.

4.2.3 `RandomForest`

Se utiliza para la clasificación un algoritmo de `RandomForest` debido a la gran cantidad de muestras a ser clasificadas y ya que las clases se presentan tan desbalanceadas. Se utilizaran `RandomForest` con 50 árboles.

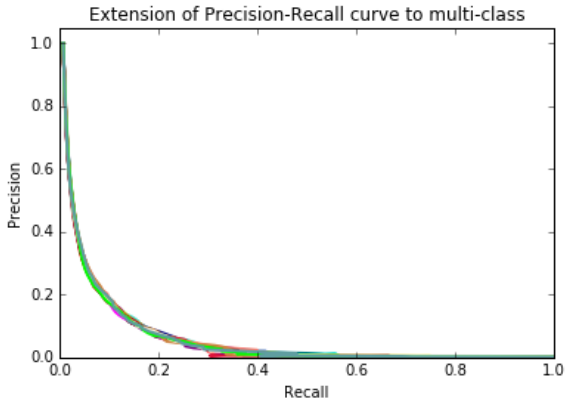
4.2.4 Evaluación

En esta investigación, se utilizan dos tipos de evaluación de performance para los modelos predictivos, ellos son la curva de ROC y la curva PRC. La curva de ROC evalúa el False Positive Rate contra el True Positive Rate, mientras que la curva PRC evalúa la Precisión del modelo, contra la Efectividad o Recall del modelo.

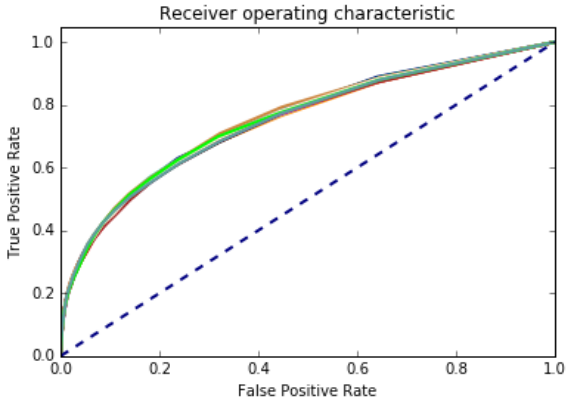
4.2.5 Validación

Debido a que los algoritmos de selección poseen una cuota de aleatoriedad, se realizan 20 ejecuciones para evaluar si la clasificación no depende de la aleatoriedad y verificar que los resultados sean fehacientes. Además se presentan los valores de recall y false positive rate para cada uno de los clusters con los sets de entrenamiento contra los sets de testing.

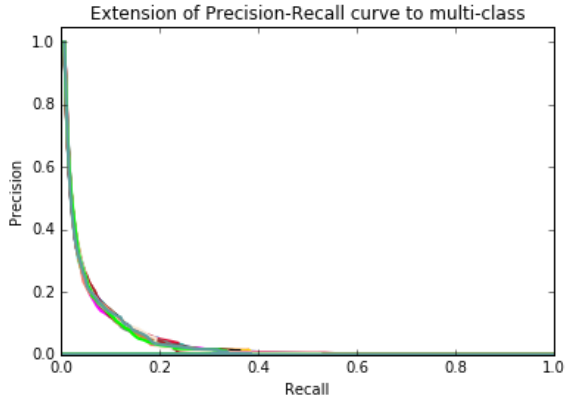
Para el cluster 0 se obtienen los siguientes resultados:



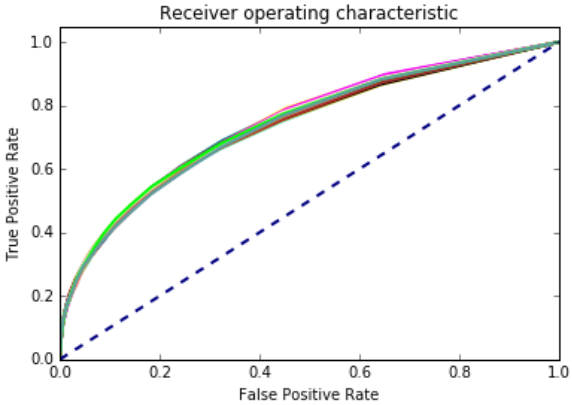
(a) PRC Datos de validación



(b) ROC Datos de validación



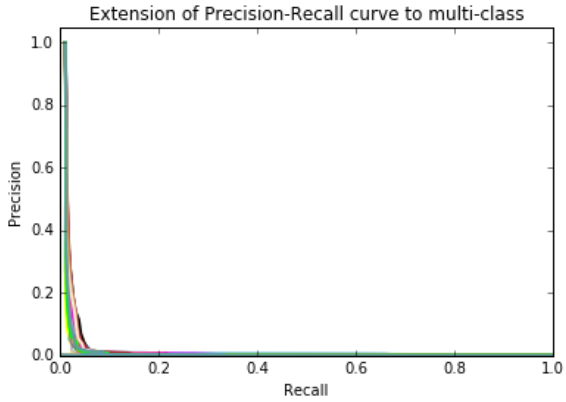
(a) PRC Datos de testing



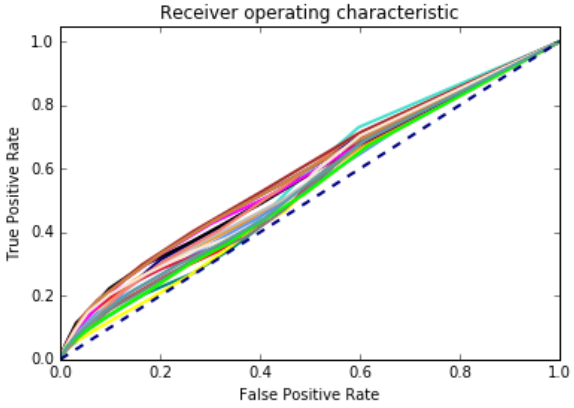
(b) ROC Datos de testing

Para el cluster 0, el valor promedio del área de PRC para los datos de validación es de 0.06, mientras que el área promedio para los mismos datos de ROC es de 0.749.Mientras que para los datos de prueba el área promedio de PRC es de 0.047, mientras que para el área ROC 0.7395, en promedio.

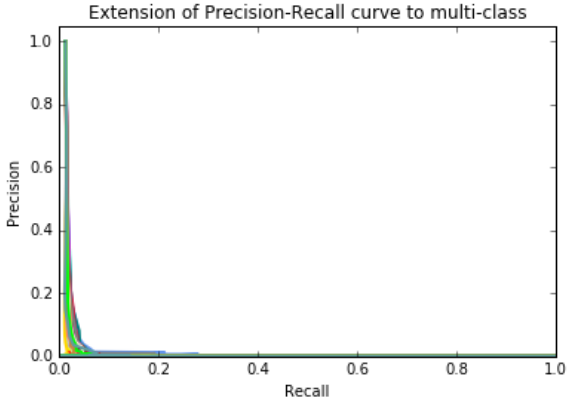
Mientras que para el cluster 1:



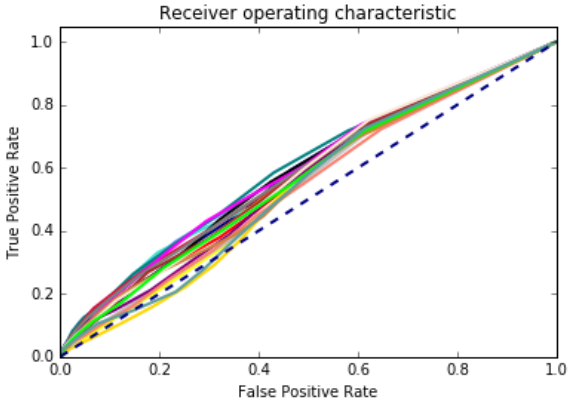
(a) PRC Datos de validación



(b) ROC Datos de validación



(a) PRC Datos de testing



(b) ROC Datos de testing

El valor promedio del área de PRC para los datos de validación es de 0.017, mientras que el área promedio para los mismos datos de ROC es de 0.555. Mientras que para los datos de prueba el área promedio de PRC es de 0.018, mientras que para el área ROC 0.569, en promedio.

Table 4.7: Cluster 0: entrenamiento contra validación

Recall	FP
1.00	146.472
0.885	107.161
0.777	84.772
0.689	69.190
0.614	57.819
0.562	48.211
0.515	40.655
0.464	35.364
0.425	30.459
0.393	26.378
0.360	23.217
0.323	21.038
0.298	18.820
0.271	17.171
0.250	15.531
0.235	13.929
0.221	12.728
0.210	11.616
0.197	10.702
0.188	9.895
0.177	9.240
0.167	8.649
0.154	8.207
0.142	7.729
0.133	7.350
0.123	6.917
0.115	6.448
0.104	6.198
0.94	5.965
0.87	5.561
0.81	5.024
0.74	4.559
0.67	4.165
0.59	3.861
0.49	3.657
0.42	3.446
0.37	3.185
0.31	2.947
0.25	2.807
0.21	2.666
0.15	2.688
0.10	2.966
0.7	2.857
0.4	3.272
0.2	3.857
0.2	2.714
0.1	1.75
0.0	3
0.0	2
0.0	1
0.1	1
0.0	1

Table 4.8: Cluster 1: entrenamiento contra validación

Recall	FP
1.00	92.836
0.665	84.116
0.544	83.971
0.456	81.096
0.378	73.274
0.287	69.641
0.223	61.410
0.163	54.742
0.121	46.330
0.77	43.259
0.54	35.912
0.32	34.705
0.19	31.75
0.13	24.428
0.9	19
0.5	14.2
0.4	7.25
0.4	4.5
0.2	5

Table 4.9: Cluster 0: entrenamiento contra test

Recall	FP
1.000	151.144
0.901	110.170
0.794	87.064
0.693	72.593
0.613	61.459
0.539	53.273
0.477	46.564
0.429	40.578
0.388	35.565
0.354	31.183
0.327	27.132
0.298	24.205
0.277	21.273
0.256	18.869
0.234	17.292
0.213	16.075
0.196	14.795
0.182	13.639
0.171	12.529
0.159	11.756
0.147	11.172
0.137	10.349
0.127	9.854
0.117	9.300
0.110	8.662
0.100	8.190
0.091	7.676
0.083	7.263
0.078	6.607
0.070	6.171
0.060	6.039
0.054	5.549
0.048	5.171
0.043	4.719
0.037	4.342
0.030	4.275
0.026	3.910
0.022	3.538
0.017	3.500
0.013	3.474
0.011	3.152
0.008	3.000
0.007	2.619
0.005	2.375
0.003	2.600
0.002	2.833
0.001	4.000
0.001	2.000
0.001	1.500
0.000	1.000

Table 4.10: Cluster 1: entrenamiento contra test

Recall	FP
1.000	79.743
0.693	69.047
0.569	67.901
0.491	64.898
0.393	60.996
0.286	56.403
0.176	56.852
0.118	49.757
0.068	49.063
0.043	42.540
0.026	39.433
0.017	31.850
0.010	28.083
0.008	18.111
0.003	23.000
0.002	24.500
0.000	1.000

Chapter 5

Conclusiones

5.1 Clustering

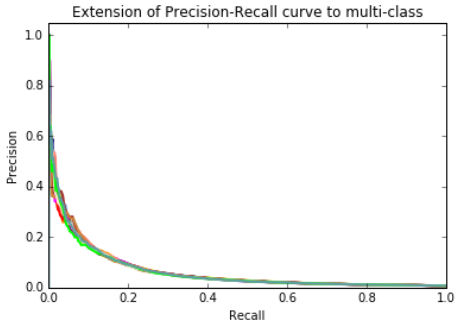
La regla del codo para nuestros set de datos extrajo como k óptimo 4, sin embargo, se concluyó que el k a utilizar sería 2. Se concluye que en este paso se puede haber introducido error a la hora de generar el cluster óptimo donde su variabilidad no es óptima. Este error se puede estar arrastrando en el resto del flujo de trabajo.

5.2 Clasificación

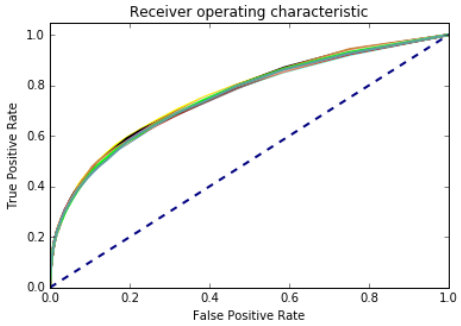
Consecuentemente con el punto anterior, el cluster 1, que es el minoritario, debe ser estudiado ya que los resultados de su clasificación son pobres con respecto al cluster 0.

A partir de las ejecuciones efectuadas se puede visualizar que no existe variabilidad en 20 ejecuciones, el cluster 0 se encuentra estable mientras que el cluster 1 presenta variablidad.

Se puede ver que en las figuras siguientes que con 90 árboles y 20 iteraciones para el cluster 0 los resultados son análogos:



(a) PRC Datos de validación



(b) ROC Datos de validación

El promedio de PRC es 0.62, mientras que el promedio del área ROC es .

5.3 Trabajos futuros

El clustering es una herramienta de agrupamiento, donde para poder dividir en perfiles diferentes de tarjetas, será necesario generar más atributos propios de la tarjeta que puedan ayudar a fomentar dicha división.

Se entiende que el algoritmo de clustering es dependiente de los atributos generados y al no ser supervisado es de mucha importancia la etapa de extracción de características.

La métrica utilizada es Kullback–Leibler debido a que los atributos se encuentran en distintas escalas. Es necesario rever la métrica o el algoritmo de clustering en función de mejorar el desempeño.

Se entiende que para clasificación es también vital la creación de variables relevantes que puedan generar un diferencial.

Bibliografía

[1] Richard O. Duda, Peter E. Hart, David G. Stork. *Pattern Classification*. John Wiley and Sons, 2012

[2] Ian H. Witten, Eibe Frank, Mark A. Hall. *Data Mining: Practical Machine Learning Tools and Techniques*. Morgan Kaufmann, 2011

[3] Alejandro Correa Bahnsen, Djamila Aouada, Aleksandar Stojanovic, Björn Ottersten, "Feature engineering strategies for credit card fraud detection", *ELSEVIER*, Volume 51, 1 June 2016, Pages 134–142

[4] Ulrike von Luxburg, Robert C. Williamson, Isabelle Guyon, *Clustering: Science or Art?*, `http://www.jmlr.org/proceedings/papers/v27/luxburg12a/luxburg12a.pdf`