RISI 10(1), 87 - 98 (2013)
REVISTA DE INVESTIGACIÓN DE SISTEMAS E INFORMÁTICA
FACULTAD DE INGENIERÍA DE SISTEMAS E INFORMÁTICA
UNIVERSIDAD NACIONAL MAYOR DE SAN MARCOS

ISSN 1815-0268 (versión impresa)
ISSN 1816-3823 (versión electrónica)

# Un Algoritmo genético para la detección de fraude electrónico en tarjetas de débito en el Perú

***A Genetic Algorithm For Fraud Detection Electronic Debit Cards In Peru***

Luis Enrique Lavado Napaico

Universidad Nacional Mayor de San Marcos
Facultad de Ingeniería de Sistemas e Informática

llavado@msn.com

## RESUMEN

La gran problemática de la banca peruana es detectar las transacciones fraudulentas que se encuentran dispersas con las transacciones genuinas. Las soluciones propuestas no son suficientes para detectar estas operaciones ilícitas con precisión porque están orientadas a mercados diferentes al peruano. Consecuentemente, se han revisado las principales aportaciones dentro de este ámbito, tales como teoría de Dempster-Shafer y aprendizaje bayesiano, sistema evolutivo Fuzzy Darwinian evolutionary system, etc. se presenta el modelado y diseño de un algoritmo genético para obtener las reglas más representativas de compra de los tarjetahabientes dentro del universo de datos transaccionales recopilados de un banco peruano. De las pruebas experimentales se obtuvo una precisión del 95.5% en el canal internet y 95.8% para el canal punto de venta. Finalmente, Se concluyó que el empleo de la estrategia de algoritmo evolutivo obtuvo una aceptable exactitud en la predicción.

**Palabras clave:** Comercio electrónico, algoritmos genéticos, fraude electrónico.

## ABSTRACT

The great problem of the bank is to detect fraudulent transactions which are scattered with genuine transactions. The proposed solutions are not sufficient to detect these illegal operations precisely because they are aimed at different Peruvian markets. Consequently, we reviewed the main contributions in this area, such as Dempster–Shafer theory and Bayesian learning, Fuzzy Darwinian, etc. presents the modeling and design of a genetic algorithm to obtain more representative rules cardholder purchase within the universe of transactional data collected from a Peruvian bank. From experimental evidence obtained an accuracy of 95.5% in the internet channel 95.8% for the point of sale. Finally, it was concluded that the use of evolutionary algorithms strategy got an acceptable accuracy in prediction.

**Keywords:** Electronic commerce, genetic algorithms, fraud electronic.

## 1. INTRODUCCIÓN

El impacto que las tecnologías de información tienen en el mundo actual es impresionante. Cada vez se crean nuevos paradigmas y la forma de realizar operaciones comerciales está cambiando. Las entidades comerciales y financieras en el mundo están apostando fuerte por la sustitución del papel moneda por el dinero electrónico, sin embargo, para expandir el comercio electrónico se requiere de un sistema de pago que se ajuste a las necesidades de compra de los usuarios, transmitiendo la seguridad y confianza en sus operaciones. El estudio presentado por Richardson Robert [1] director de Computer Security Institute (CSI) indica que entre los principales riesgos de seguridad, los ataques por fraude financiero y las vulnerabilidades relacionadas con aplicaciones Web de comercio electrónico están aumentando y que postulan como nuevas tendencias de riesgo. El informe realizado por CyberSource Corporation [2] revela que en el año 2011 los comercios informaron que perdieron un promedio de 1% de los ingresos totales por fraudes cometidos en transacciones de comercio electrónico. Los comercios informaron una reducción del 33% de los porcentajes de órdenes perdidas por fraude. Las operaciones fraudulentas o ilegales suelen seguir patrones característicos que permiten, con cierto grado de probabilidad, distinguirlas de las legítimas y desarrollar así mecanismos para tomar medidas rápidas frente a ellas.

### 1.1. Problemática

La seguridad se ha convertido en el principal problema e inquietud del comercio electrónico. Existe el temor de los consumidores y negocios debido a las grandes olas de estafas que el mundo está experimentando. El informe realizado por Frost & Sullivan [3] revela que el 22% de los usuarios en Latinoamérica han dejado de usar la banca en línea, y el 10% han cambiado de banco debido a incidentes de fraude. Un impresionante 95% de los usuarios de transacciones en línea creen que su banco debe implementar mayores y mejores soluciones de seguridad, a fin de minimizar los riesgos por fraude. La realidad del Banco de la Nación del Perú indica que los servicios de banca por internet y operaciones VISA ha sufrido una baja en los últimos años debido a la desconfianza y el temor del tarjetahabiente por el uso de estos servicios no tradicionales.

### 1.2. Propuesta

Se propone la utilización de un modelo heurístico basado en el comportamiento transaccional de los clientes y la determinación de los patrones de desviación que sean catalogadas como sospechosas, para ello se empleará técnicas basadas en algortimos genéticos. En la primera fase se recopilará datos históricos transaccionales de los canales: punto de venta (POS) y operaciones en línea internet de un periodo representativo (julio y agosto del 2011); posteriormente, se diseñará un algoritmo genético cuyo objetivo será maximizar la precisión de la predicción frente a los datos reales. Los objetivos alternos serán: la minimización de los falsos positivos que son las transacciones que el sistema deja pasar aún siendo fraudulentas y la minimización los falsos negativos que son aquellas que el sistema devuelve una transacción como sospechosa siendo una operación válida.

## 2. ESTADO DEL ARTE

La detección del fraude consiste en analizar el comportamiento de los usuarios cuando realiza una operación financiera para estimar, detectar o evitar las transacciones indeseables. Para una lucha eficaz contra el fraude de transacciones de tarjeta de débito es necesario comprender las estrategias empleadas en investigaciones científicas que derivaron en reglas con alto valor predictivo.

### 2.1. Paradigmas de detección de fraude

#### *2.1.1. Fusión teoría Dempster–Shafer y Bayesian learning*

Panigrahi S. et al. [4], proponen un modelo para la detección de fraudes de tarjetas crédito, que combina las evidencias actuales así como el comportamiento pasado almacenado en el historial de transacciones. La propuesta se basa en la integración de los enfoques: rule-based filtering, teoría Dempster–Shafer y Bayesian learning. En el modelo propuesto, un número de reglas se utilizan para analizar la desviación de cada transacción entrante de un perfil normal de tarjetahabiente. Una transacción entrante es primero manejada por el componente Rule-based. La creencia de los valores son combinados para tener una creencia inicial para la transacción mediante la aplicación de la teoría Dempster–Shafer. La creencia general es, además, reforzada o debilitada conforme a similares transacciones históricas de fraude u operaciones genuinas usando Bayesian learning. El modelo se representa en la figura 1.

La simulación del modelo obtuvo 98% verdadero positivo (TP) y menos que el 10% de Falso Positivo (FP). La utilización de la teoría de Dempster-Shafer ofrece una buena precisión, sobre todo en términos de verdaderos positivos, el aprendizaje bayesiano ayuda a mejorar la exactitud del sistema.

### *2.1.2. Fuzzy Darwinian*

Bentley Peter J. et al. [5], propone un modelo de detección Fuzzy Darwinian capaz de clasificar las transacciones de tarjetas de crédito en clases sospechosos y no sospechoso. El sistema se compone de un algoritmo de búsqueda en programación genética (GP) y un sistema experto difuso. El sistema híbrido genético difuso es un sistema adaptativo donde la estructura se puede construir y los parámetros se pueden ajustar por un algoritmo de aprendizaje. La Figura 2 provee una visión de este modelo.

Los resultados experimentales muestran que este método es capaz de alcanzar una buena precisión y los niveles de inteligibilidad de los datos reales. Se tiene una precisión muy alta y produce una baja falsa alarma, detectando el 100% de las sospechas verdadero positivo (TP) y un 5.79% de Falso Positivo (FP).

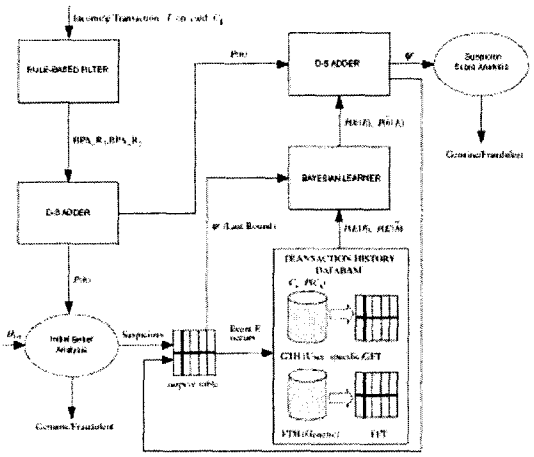



**Figura 1.** Propuesta de detección de fraude Dempster–Shafer theory and Bayesian learning (Panigrahi et al., 2009)

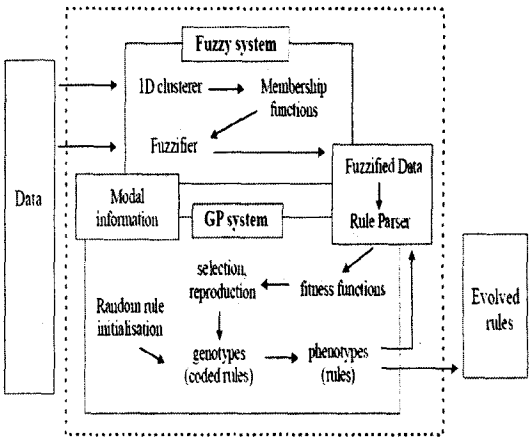



**Figura 2.** Arquitectura Evolutionary-Fuzzy System (Peter J. Bentley, 2000).

## 2.2. Comparación de métodos de detección de fraude

En las publicaciones Benson y Portia [6], se presenta un análisis comparativo de métodos detección de fraude, ver Tabla 1. Para la comparación de diferentes métodos se emplea la matriz de confusión. Los autores comparan las siguientes técnicas: Dempster-Shafer theory and Bayesian learning [4], BLAST-SSAHA Hybridization [7], Hidden Markov Model [8], Fuzzy Darwinian [5] y Bayesian and Neural Networks [9]. Los parámetros usados para la comparación de las técnicas de detección de fraude son la precisión, ratios en término de verdadero positivo (TP-denota el número de casos positivos que han sido predichos como tal) y falso positivo (FP-denota el número de casos negativos que han sido predichos como positivos). Los resultados muestran que la técnica de detección de fraude Fuzzy tienen alta ocurrencia, el ratio TP llegó al 100% en comparación de los otros y para el caso del ratio FP fue de 5.79% el menor de todos.

## 3. METODOLOGÍA DE LA INVESTIGACIÓN

### 3.1. Modelo Conceptual de la propuesta

La propuesta de extracción de reglas de comportamiento esbozado en la figura 3, consiste en recopilar los datos históricos transaccionales de los canales punto de venta (POS) y operaciones en internet. De los datos de muestreo el 70% servirá para el proceso de entrenamiento del algoritmo genético y la diferencia para el proceso de evaluación. Los datos de transacciones genuinas y fraudulentas se conjugan con las bases de datos demográficos del cliente, datos de las

cuentas de ahorros, ingresos y saldos del cliente y la base de datos de tarjetas y préstamos con el fin de obtener atributos relevantes para el análisis. Se aplica la metodología CRISP-DM como herramienta que guía el proceso de descubrimiento de conocimiento y permite la identificación de variables. En la fase de modelado del CRISP-DM se emplea una técnica basada en algoritmos genéticos, la resultante del proceso de entrenamiento del algoritmo genera reglas interesantes que identifiquen a las transacciones genuinas y fraudulentas. En la fase de evaluación se verifica la calidad de las reglas obtenidas del proceso de entrenamiento. La salida del esquema propuesto establece dos grupos de reglas de decisión, (1) las que son consideradas como comportamiento habitual y (2) las reglas que fueron catalogadas como fraudulentas.

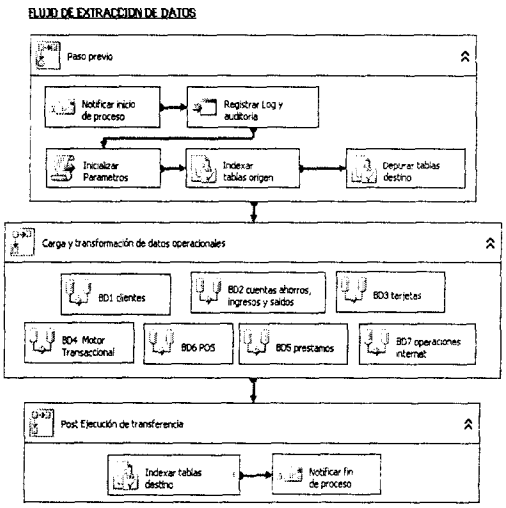
| Parametro | | Fusion of Dempster-Shafer theory and Bayesian Learning | Hybridization BLAST and SSAHA | HMM | Artificial Neuronal Networks and Bayesian Nueronal Network | | Fuzzy Darwinian Detection |
|---|---|---|---|---|---|---|---|
| | | | | | ANN | BNN | |
| Study | | Suvasini Panigrahi, Amlan Kundu, Shamik Sural a,A.K. Majumdar b (2009) | Amlan Kundu, Suvasini Panigrahi, Shamik Sural, Senior Member, IEEE, and Arun K. Majumdar, Senior Member, IEEE (2009) | Abhinav Srivastava, Amlan Kundu, Shamik Sural, Senior Member, IEEE, and Arun K. Majumdar, Senior Member, IEEE (2008) | Sam Maes, Karl Tuyls, Bram Vanschoenwinkel, Bernard Manderick (1993) | | Peter J. Bentley, Jungwon Kim, Gil-Ho Jung and Jong-Uk Choi (2000) |
| Method | | Machine Learning | sequence alignment | Hidden Markov Model | Artifical Inteligence, Machine Learning | Artifical Inteligence, Machine Learning | Genetic Programing, Fuzzy Logic |
| Fraud Detection | TP% | 98% | 86% | 70% | 70% | 74% | 100% |
| | FP% | 10% | 10% | 20% | 10% | 10% | 5.79% |
| Procesing Speed | | Medio | Muy Alto | Alto | Alto | Bajo | Bajo |
| Training Required | | Si | No | Si | Si | Si | Si |
| Superised Learning | | Supervisado | No supervisado | Supervisado | Supervisado | Supervisado | Supervisado |
| Costo | | Caro | Barato | bastante caro | Caro | Caro | muy Caro |
| Accuracy | | Alto | Alto | Medio | Medio | Medio | Muy alto |

**Figura 3.** Esquema del Modelo Conceptual de la propuesta.

## 3.2. Fases para la obtención de variables

### *3.1.1. Entendimiento de los datos*

En la Figura 4, se representa el flujo del proceso de extracción y transformación de los datos. En la etapa inicial del paquete el primer grupo de tareas se encargan del registro de eventos, inicialización de variables, re-indexación de tablas, y a la depuración de los datos en las tablas de destino. En la segunda etapa del paquete se procede a leer los datos de cada sistema origen y transferirlos a los nuevos esquemas; en este proceso se adicionó tareas de conversión y limpieza que permita verificar si los datos cumplen con la estructura solicitada y la calidad esperada. En la etapa final del paquete se procede a indexar las tablas de destino para un mejor rendimiento.



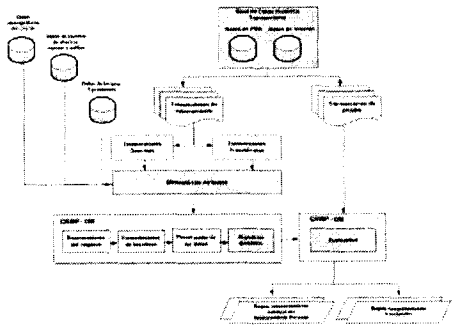


**Figura 4.** Flujo de extracción de los datos iniciales.

### *3.1.2. Preparación de los datos*

El empleo y representación de las variables dependen de los objetivos definidos en la fase "entendimiento del negocio" y de estudios previos de investigadores cuyo modelos han demostrado una propuesta de solución al fraude electrónico. En la tabla 2 se lista las variables utilizadas en el algortimo genético.

| # | Alias | Descripción | Canal |
|---|---|---|---|
| **Variables de entrada** | | | |
| 1 | HCHP | Horario de compra habitual del cliente dentro de las 24 horas | POS |
| 2 | HCHI | Horario de compra habitual del cliente dentro de las 24 horas | WWW |
| 3 | CSCP | Situación civil del cliente asociada a la compra | POS |
| 4 | CSCI | Situación civil del cliente asociada a la compra | WWW |
| 5 | CCEP | Categoría de establecimiento más concurrida por el cliente | POS |
| 6 | CCEI | Categoría de establecimiento más concurrida por el cliente | WWW |
| 7 | CREP | Rango de edad del cliente asociado al hábito de compra | POS |
| 8 | CREI | Rango de edad del cliente asociado al hábito de compra | WWW |
| 9 | TETP | Tiempo transcurrido (días) desde su enrolamiento de la tarjeta y su primera compra en el periodo de muestra | POS |
| 10 | TETI | Tiempo transcurrido (días) desde su enrolamiento en internet y su primera compra en el periodo de muestra (enrolamiento en la compra) | WWW |
| 11 | TUCP | Tiempo transcurrido (horas) desde la última compra a la penúltima compra con la misma tarjeta | POS |
| 12 | TUCI | Tiempo transcurrido (horas) desde la última compra a la penúltima compra con la misma tarjeta | WWW |
| 13 | MMXP | Monto máximo de compra | POS |
| 14 | MMXI | Monto máximo de compra | WWW |
| 15 | NMXP | Número máximo de transacciones efectivas realizadas con una misma tarjeta en el día | POS |
| 16 | NMXI | Número máximo de transacciones efectivas realizadas con una misma tarjeta en el día | WWW |
| 17 | NEVP | Número de establecimientos visitados en un día | POS |
| 18 | NEVI | Número de establecimientos visitados en un día | WWW |
| 19 | NMRP | Número máximo de rechazos detectados antes de ser efectiva la transacción de compra en el mismo establecimiento | POS |
| 20 | NMEI | Número máximo de intentos de enrolamiento antes de la primera compra | WWW |
| **Variable de salida** | | | |
| 1 | OUPT | Situación de la transacción | POS / WWW |

**Tabla 2.** Variables utilizadas en el algortimo genético.

## 4. INGENIERÍA DEL ARTEFACTO

### 4.1. Planteamiento del algoritmo genético propuesto

El algoritmo genético planteado sigue el enfoque Iterative Rule Learning (IRL). Según esta estrategia cada individuo de la población representa una única regla, sólo el mejor individuo es considerado como parte de la solución descartándose al resto de individuos de la población. Por lo tanto, cada ejecución de los algoritmos genéticos proporciona una solución parcial al problema de aprendizaje, ya que en cada iteración aportará una nueva regla a la solución problema [10].

Para la aplicación del paradigma de un algoritmo genético presenta dos factores críticos: la codificación de los individuos y la evaluación de éstos. Ambos factores, entre otros, influyen en la efectividad del algoritmo y su convergencia, siendo por tanto los aspectos donde se centra gran parte del esfuerzo.

### *4.1.1. Codificación del Cromosoma*

La forma de codificación propuesta sigue el enfoque IRL, en el que cada cromosoma representa una regla "Cromosoma = Regla", como SLAVE [11], MOGUL [12] y la propuesta de Carvalho y Freitas [13]. Cada individuo de la población representa una única regla de decisión que describe la relación entre los valores de los atributos y las etiquetas de clase.

$$SI\ a_1 \in \{v_1, v_2, \ldots v_i\}\ Y\ a_2 \in \{w_1, w_2, \ldots w_j\}\ \ldots\ ENTONCES\ Clase = E$$

Donde:
$a_1$, $a_2$ son atributos y $v_1 \ldots v_i$, $w_1 \ldots w_j$ valores asignados para cada atributo, y $Clase = E$ etiqueta de clase de clasificación.

Las reglas se representan normalmente mediante cadenas binarias de bit ***(0's y 1's)*** de una longitud determinada ***L*** que vendrá impuesta por el número de variables existentes en la solución y por el número de bits necesarios para codificarla, el espacio de búsqueda tendrá un tamaño de . El antecedente o descripción de la regla (parte izquierda) es una conjunción entre atributos y varias disyunciones sobre los valores de los atributos. El esquema de codificación binario, por cada atributo se almacena un bit para cada uno de los valores categóricos que puede tomar, de forma que si el bit correspondiente tiene el valor de 0 indica que no pertenece a la condición y si tiene el valor 1 que sí pertenece. Si en un individuo todos los bits correspondientes a un atributo tienen valor 1, indica que dicha variable no es relevante para la información aportada en la regla (cualquier valor de la variable verifica la condición de la regla), por lo que esta variable se ignora. El código 0 es omitido en el conjunto de valores codificados al carecer éste de sentido. En la tabla 3 se ejemplifica el diseño del antecedente del canal POS conformado por 10 variables. Para el caso del atributo ***HCHP*** los valores categóricos que participan se encuentran en la posición 2 y 4 que representa el horario de compra realizada entre 12:00 y 15:00 horas y el rango entre las 20:00 y 23:00 horas, existen 16 posibles combinaciones de ésta variable.

| VAR | Etiquetas Lingüísticas | | | | | | | | | Comentario | Comb |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | | |
| HCHP | 0 | 1 | 0 | 1 | | | | | | Compras realizadas entre las horas:<br>* 12:00 <= t <= 15:00<br>* 20:00 <= t <= 23:00 | $2^4 = 16$ |
| CSCP | 1 | 1 | 0 | 0 | 0 | | | | | Situación de los clientes que realizaron una compra:<br>* Soltero * Casado | $2^5 = 32$ |
| CCEP | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | Categoría del establecimiento donde se realizó la compra:<br>* Restaurantes | $2^9 = 512$ |
| CREP | 0 | 1 | 1 | 1 | 0 | | | | | Rango de edad de las personas con hábitos de compra:<br>* 33 <= e <= 40<br>* 41 <= e <= 49<br>* 50 <= e <= 60 | $2^5 = 32$ |
| TETP | 1 | 0 | 0 | 1 | 0 | | | | | Tiempo transcurrido en días desde su enrolamiento y su primera compra:<br>* t <= 690<br>* 1108 <= t <= 1444 | $2^5 = 32$ |
| TUCP | 1 | 1 | 1 | 0 | 0 | | | | | Tiempo transcurrido en horas desde la última compra a la penúltima:<br>* t <= 1<br>* 2 <= t <= 10<br>* 11 <= t <= 37 | $2^5 = 32$ |
| MMXP | 0 | 0 | 0 | 1 | 1 | | | | | Monto máximo de compra:<br>* 201 <= m <= 400<br>* 401 <= m | $2^5 = 32$ |
| NMXP | 1 | 1 | 0 | | | | | | | Número máximo de transacciones efectivas:<br>* n = 1 * n = 2 | $2^3 = 8$ |
| NEVP | 1 | 0 | 0 | | | | | | | Número de establecimientos visitados en un día:<br>* n = 1 | $2^3 = 8$ |
| NMRP | 1 | 1 | 0 | | | | | | | Número máximo de rechazos detectados:<br>* n = 1 * n = 2 | $2^3 = 8$ |

**Tabla 3.** Diseño del antecedente para el canal POS.

Respecto a la clase, es importante señalar que la operación se realiza sobre conjuntos de datos donde cada ejemplo es etiquetado con una única etiqueta de clase discreta ***Clase=E***, donde los valores posibles son: 0 de tratarse de una transacción genuina y 1 si es una transacción fraudulenta. Tanto para el canal POS e internet la clase es representada por el valor discreto de la variable de salida ***OUPT***.

Se ha seleccionado la Codificación Natural [14] como la forma de representación del individuo que reducirá el cardinal del conjunto de posibles soluciones, sin que ello produzca pérdida en la precisión en las mismas. Partiendo de la codificación binaria que asigna un bit a cada posible valor, denotando con 1 y 0 la presencia o ausencia del valor en la condición, respectivamente, la codificación natural transforma esa cadena binaria en su representación decimal. Así, un gen discreto es un número natural que identifica un conjunto de valores discretos y pertenece al intervalo , donde $|A|$ es el cardinal del conjunto de valores posibles del atributo. En la tabla 4 muestra un ejemplo de codificación natural de la variable ***HCHP***, conformada por 4 etiquetas lingüísticas. La codificación natural asociada a las compras realizadas entre 12:00 y 15:00 horas y el rango entre las 20:00 y 23:00 horas tiene el valor de 5, que equivale a su representación binaria {0 1 0 1}.

| Etiquetas Lingüísticas de la Variable HCHP | | | | Codificación Natural |
|---|---|---|---|---|
| *HCHP1* | *HCHP2* | *HCHP3* | *HCHP4* | |
| 0 | 0 | 0 | 1 | 1 |
| 0 | 0 | 1 | 0 | 2 |
| ... | ... | ... | ... | ... |
| **0** | **1** | **0** | **1** | 5 |
| 0 | 1 | 1 | 0 | 6 |
| 1 | 1 | 1 | 1 | 15 |

**Tabla 4.** Diseño del antecedente para el canal POS.

### 4..2. Operador genético

#### *4.2.1. Mutación del antecedente*

Partimos de un número natural cuyos bits en representación binaria denotan la presencia o ausencia de un valor en una condición. La mutación consiste en cambiar el valor del gen por otro símbolo del alfabeto que represente un conjunto de valores distinto al inicial donde simplemente se ha agregado o suprimido un valor. Esta mutación aplicada en la codificación binaria tiene una implementación consistente en seleccionar un bit al azar y cambiar su valor de 0 a 1 o al revés, según el caso. La tabla 5 ilustra la mutación del primer gen del atributo ***HCHP*** el valor original del gen ***HCHP1*** es reemplazado su valor de O a 1, generando el nuevo valor natural de 13.

**Atributo HCHP original.**

| VAR | Valor categórico | | | | Etiquetas Lingüística | | | | Cod. Nat. |
|---|---|---|---|---|---|---|---|---|---|
| | 1 | 2 | 3 | 4 | 1 | 2 | 3 | 4 | |
| HCHP | HCHP1 | HCHP2 | HCHP3 | HCHP4 | 0 | 1 | 0 | 1 | 5 |

**Atributo HCHP mutado en el primer gen.**

| Atributo | Valor categórico | | | | Etiquetas Lingüística | | | | Cod. Nat. |
|---|---|---|---|---|---|---|---|---|---|
| | 1 | 2 | 3 | 4 | 1 | 2 | 3 | 4 | |
| HCHP | HCHP1 | HCHP2 | HCHP3 | HCHP4 | 1 | 1 | 0 | 1 | 13 |

**Tabla 5.** Esquema de mutación del primer gen del atributo HCHP.

#### *4.2.2. Cruce del antecedente*

El cruce se basa en la mutación natural definida anteriormente. Cada gen que interviene en el cruce proporciona un conjunto de candidatos. Estos candidatos son el resultado de unir el conjunto de posibles mutaciones del gen con el propio gen. Así, la descendencia de dos genes se calcula como la intersección de los conjuntos de candidatos que cada gen aporta. Cuando los padres no ofrecen candidatos comunes, se calculan nuevos candidatos para cada padre mutando los conjuntos iniciales hasta que la intersección no esté vacía [14]. Se detalla un ejemplo de los operadores naturales para atributos discretos basados en ***HCHP***. En la tabla 6 el gen codificado con el número natural ***11*** tiene como código binario el ***1011***. El bloque a la derecha da las posibles mutaciones que este gen puede sufrir. Para el gen codificado con el valor natural ***8 (1000)*** es similar. Así, los conjuntos de posibles mutaciones de ***11*** y ***8*** son ***{10; 9; 15; 3}*** y ***{9; 10; 12; 0}***, respectivamente.

Gen 11 → { 1 0 1 1 } del atributo HCHP

| Valor natural Original | Etiquetas Lingüísticas 1 2 3 4 | Etiquetas Lingüísticas 1 2 3 4 | Mutación | Valor categórico 1 2 3 4 |
|---|---|---|---|---|
| 11 | 1 0 1 1 | 1 0 1 0 | 10 | HCHP1 HCHP2 HCHP3 HCHP4 |
| 11 | 1 0 1 1 | 1 0 0 1 | 9 | HCHP1 HCHP2 HCHP3 HCHP4 |
| 11 | 1 0 1 1 | 1 1 1 1 | 15 | HCHP1 HCHP2 HCHP3 HCHP4 |
| 11 | 1 0 1 1 | 0 0 1 1 | 3 | HCHP1 HCHP2 HCHP3 HCHP4 |

Gen 8 → { 1 0 0 0 } del atributo HCHP

| Valor natural Original | Etiquetas Lingüísticas 1 2 3 4 | Etiquetas Lingüísticas 1 2 3 4 | Mutación | Valor categórico 1 2 3 4 |
|---|---|---|---|---|
| 8 | 1 0 0 0 | 1 0 0 1 | 9 | HCHP1 HCHP2 HCHP3 HCHP4 |
| 8 | 1 0 0 0 | 1 0 1 0 | 10 | HCHP1 HCHP2 HCHP3 HCHP4 |
| 8 | 1 0 0 0 | 1 1 0 0 | 12 | HCHP1 HCHP2 HCHP3 HCHP4 |
| 8 | 1 0 0 0 | 0 0 0 0 | 0 | HCHP1 HCHP2 HCHP3 HCHP4 |

**Tabla 6.** Tabla de comprobación de mutaciones del gen discreto HCHP.

El cruce entre ambos genera el conjunto de genes ***{9; 10}***, pues es la intersección de los dos conjuntos anteriores.

Mut(11) = {10,9,15,3}

Mut(8) = {9,10,12,0}

Cruce(11,8) = { Mut(11) Mut(8) }

Cruce(11,8) = { {10,9,15,3} {9,10,12,0}}

Cruce(11,8) = {9,10}

### 4.3. Descripción del algoritmo genético

El algoritmo propuesto se divide en un procedimiento principal ***MainDetectFraud***, el cual construye el conjunto de reglas; y la función complementaria ***AlgoritmoEvolutivo***, que implementa el algoritmo genético propiamente dicho. Inicialmente, el conjunto de reglas ***R*** está vacío y en cada iteración se añade la regla que devuelve ***AlgoritmoEvolutivo***. El parámetro ***DE*** es el conjunto de datos de entrenamiento, el cual es representada con base a la propuesta de Codificación Natural [14], con el fin de reducir el espacio de búsqueda y acelerar la convergencia del algoritmo. La transformación da como resultado el conjunto de ejemplos codificados ***xDE*** que será usado durante toda la ejecución. La variable ***n*** almacena el número inicial de ejemplos de ***xDE***, ya que éste será reducido en cada iteración. Dicha reducción se produce eliminando aquellos ejemplos de ***xDE*** que son cubiertos por la regla ***r*** obtenida de ***AlgoritmoEvolutivo***. La regla obtenida r se adiciona al conjunto de reglas solución ***R***. El parámetro ***Poda*** (factor de poda de datos de entrenamiento) controla el número de ejemplos ***xDE*** que aún no han sido cubiertos durante el proceso. El proceso iterativo finaliza cuando el número de ejemplos que restan el conjunto de entrenamiento ***xDE*** no supera el factor establecido por ***Poda*** sobre el número inicial de ejemplos ***n***. El módulo ***AlgoritmoEvolutivo*** inicializa la población ***P*** y ejecuta la función evolucionar que devuelve la regla con la mejor fitness obtenida en el proceso evolutivo. Finalmente, se ejecuta la función ***reducirEscenario*** que elimina los datos de entrada que clasificaron con la regla seleccionada. El módulo ***Evolucionar*** lleva a cabo la evolución siempre y cuando el número de generaciones es menor igual al parámetro ***num_generaciones*** ingresada por el usuario. En cada iteración, el procedimiento ***Evaluacion*** asigna un valor de bondad a cada individuo de la población actual ***P***, la función de bondad está conformada por dos factores: la complejidad y completitud de la regla [15].

El procedimiento ***Reemplazo*** genera la nueva población mediante los operadores genéticos: selección, cruce y mutación de individuos. Cuando el número de generaciones preestablecido ***num_generaciones*** es alcanzado, la población final es evaluada para seleccionar el mejor individuo de ésta. Para relajar el modelo, en la fase de selección de los individuos para el cruce se empleará el método de la ruleta para aceptar soluciones malas y para salir del óptimo local. El individuo seleccionado en el procedimiento ***Evaluacion*** es devuelto al módulo principal para eliminar los ejemplos cubiertos ***xDE***, se incluirá en el conjunto de reglas ***R*** y se continuará el proceso.

## 5. EXPERIMIENTOS Y RESULTADOS

En las siguientes secciones se detalla los resultados de cada experimento realizados en forma independiente para el canal POS e internet.

## 5.1. Canal POS

### *5.1.1. Parámetros iniciales de entrenamiento*

El proceso de entrenamiento consiste en ejecutar el algoritmo genético para 4 escenarios, en los cuales se modificó el parámetro número de individuos de la población con el fin de obtener el conjunto de reglas con la mejor bondad. Para una muestra de 520 registros (480 transacciones genuinas y 45 transacciones fraudulentas), se inicializa con 10 individuos hasta llegar a 50 individuos. El factor de mutación se mantiene uniforme en 2% de la cantidad de individuos. El parámetro de completitud es mayor a la complejidad porque la búsqueda está orientada a la obtención de reglas con una alta capacidad predictiva (número de ejemplos cubiertos por la regla).

### *5.1.2. Resultados del entrenamiento*

En la tabla 7 se detalla los escenarios probados y la cantidad de reglas genuinas y fraudulentas obtenidas. En el escenario 1 se ejecutó el algoritmo genético con una población de 10 individuos obteniéndose 32 reglas, sucesivamente se aumentó el número de individuos.

| # | Parámetros Entrada | | | | Resultado entrenamiento | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Nuestra | | | | Reglas obtenidas | | | | | Ejemplos cubiertos | | | |
| | G | F | Total | Individuo | G | F | Total | | Cobertura por regla | G | F | Total | % |
| 1 | 480 | 45 | 525 | 10 | 20 | 12 | 32 | G | 35 19 13 5 12 12 11 8 7 5 5 3 2 2 2 2 2 2 1 1 1 | 147 | 43 | 190 | 36.2 % |
| | | | | | | | | F | 17 7 1 1 3 2 2 2 1 1 1 1 1 | | | | |
| 2 | 480 | 45 | 525 | 20 | 14 | 6 | 20 | G | 158 68 49 35 5 3 25 25 20 16 11 11 5 4 3 | 457 | 43 | 500 | 95.2 % |
| | | | | | | | | F | 25 5 3 2 2 2 | | | | |
| 3 | 480 | 45 | 525 | 30 | 6 | 1 | 6 | G | 257 17 9 5 1 1 12 | 466 | 45 | 511 | 97.3 % |
| | | | | | | | | F | 45 | | | | |
| 4 | 480 | 45 | 525 | 40 | 4 | 1 | 5 | G | 295 93 65 17 | 468 | 45 | 513 | 97.7 % |
| | | | | | | | | F | 25 | | | | |

**Tabla 7.** Resultados del entrenamiento canal POS.

En el escenario 4 se obtuvo 5 reglas (4 reglas para transacciones genuinas 1 una regla para transacciones fraudulentas) que han cubierto 513 ejemplos de las 525 muestras, haciendo un 97.7% de cobertura del total de la muestra. De las simulaciones realizadas se deduce que el aumento de individuos en el proceso genético se obtiene como resultado la reducción del número de reglas y un aumento en el número de ejemplos cubiertos.

### *5.1.3. Validación del entrenamiento*

En la tabla 8 se detalla los resultados obtenidos del proceso de verificación de las reglas generadas en el proceso de entrenamiento. El conjunto de datos de prueba está conformado de 240 transacciones (220 transacciones genuinas y 20 fraudulentas). En el escenario 1 el resultado de la precisión fue muy desalentador obteniéndose sólo un 31.3%. Al mejorarse la bondad de las reglas en el proceso de entrenamiento se obtuvo un mejor rendimiento en cada escenario. En el escenario 4 la precisión alcanzó un 95.8%, alcanzando un valor de TP de 211 de las 220 transacciones genuinas y un valor de TN de 19 de las 20 transacciones fraudulentas.

| # | Parámetros Entrada | | Reglas obtenidas | Nro. de TRX Validación | Validar algoritmo | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | Muestra | Individuo | | | TP | FP | FN | TN | Total | Precisión |
| 1 | 525 | 10 | 32 | 240 | 70 | 15 | 150 | 5 | 240 | 31.3% |
| 2 | 525 | 20 | 20 | 240 | 180 | 13 | 30 | 7 | 240 | 82.1% |
| 3 | 525 | 30 | 6 | 240 | 197 | 9 | 23 | 11 | 240 | 86.7% |
| 4 | 525 | 40 | 5 | 240 | 211 | 1 | 9 | 19 | 240 | 96.8% |

**Tabla 8.** Validación de reglas obtenidas canal POS.

## 5.2. Canal Internet

### *5.2.1. Parámetros iniciales de entrenamiento*

El proceso de entrenamiento ejecuta el algoritmo genético para 4 escenarios. Se tomó como muestra 418 transacciones por internet distribuidas en 346 transacciones genuinas y 72 transacciones fraudulentas. La población es inicializada con 10 individuos los cuales son incrementados aritméticamente en 10 unidades llegando a un tope de 40 individuos. El factor de mutación se mantiene uniforme en 2% de la cantidad de individuos. El porcentaje del parámetro completitud es del 98% y la diferencia es para el parámetro complejidad. El factor de poda que controla el número de ejemplos que aún no han sido cubiertos durante el proceso es del 5% del total de transacciones de muestra.

### 5.2.2. Resultados del entrenamiento

Las reglas obtenidas del proceso de entrenamiento se describen en la tabla 9. En la columna reglas obtenidas se cuantifica los individuos derivados de la ejecución del algoritmo genético, diferenciando las transacciones genuinas de las fraudulentas, además se detalla el nivel de cobertura de cada individuo. En la columna ejemplos cubiertos se totaliza las coincidencias encontradas, se elige el escenario que posea el menor número de reglas que cubran la mayor cantidad de transacciones de muestra. En el escenario 1 se ejecutó el algoritmo genético con una población de 10 individuos obteniéndose 28 reglas y un 73.4% de cobertura, sucesivamente se aumentó el número de individuos con el fin de mejorar estos resultados. En el escenario 4 se obtuvo 5 reglas (4 reglas para transacciones genuinas 1 una regla para transacciones fraudulentas) que han cubierto 417 ejemplos de las 418 muestras, haciendo un 99.8% de cobertura del total de la muestra.

| # | Parámetros Entrada | | | | Resultado entrenamiento | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Muestra | | | | Reglas obtenidas | | | | | Ejemplos cubiertos | | | |
| | G | F | Total | Individuo | G | F | Total | | Cobertura por reglas | G | F | Total | % |
| 1 | 346 | 72 | 418 | 10 | 21 | 7 | 28 | G | 58\|40\|35\|23\|15\|12\|11\|11\|9\|8\|8\|6\|6\|5\|4\|3\|2\|2\|1\|1\|1\| | 261 | 46 | 307 | 73.4 % |
| | | | | | | | | F | 18\|11\|7\|6\|3\|1\| | | | | |
| 2 | 346 | 72 | 418 | 20 | 10 | 5 | 15 | G | 71\|57\|47\|31\|25\|15\|12\|11\|10\|8\| | 287 | 53 | 340 | 81.3 % |
| | | | | | | | | F | 25\|19\|4\|3\|2\| | | | | |
| 3 | 346 | 72 | 418 | 30 | 7 | 4 | 11 | G | 180\|66\|36\|19\|20\|5\|5\| | 331 | 68 | 399 | 95.5 % |
| | | | | | | | | F | 35\|25\|5\|3\| | | | | |
| 4 | 346 | 72 | 418 | 40 | 4 | 1 | 5 | G | 261\|39\|29\|16\| | 345 | 72 | 417 | 99.8 % |
| | | | | | | | | F | 72\| | | | | |

**Tabla 9.** Resultados del entrenamiento canal Internet.

### 5.2.3. Validación del entrenamiento

Con el objetivo de determinar la precisión de las reglas obtenidas en el proceso de entrenamiento, se verificó los resultados con un conjunto de datos de prueba conformada por 202 transacciones (150 transacciones genuinas y 52 transacciones fraudulentas). En la tabla 10 se detalla los resultados del proceso de verificación, en función de los valores TP, FP, FN y TN. En el escenario 4 la precisión alcanzó un 95.5%, alcanzando un valor de TP de 142 de las 150 transacciones genuinas y un valor de TN de 51 de las 52 transacciones fraudulentas.

| # | Parámetros Entrada | | Reglas obtenidas | Nro. de TRX Validación | Validar algoritmo | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | Muestra | Individuo | | | TP | FP | FN | TN | Total | Precisión |
| 1 | 418 | 10 | 28 | 202 | 97 | 15 | 53 | 37 | 202 | 66.3% |
| 2 | 418 | 20 | 15 | 202 | 120 | 8 | 30 | 44 | 202 | 81.2% |
| 3 | 418 | 30 | 11 | 202 | 135 | 5 | 15 | 47 | 202 | 90.1% |
| 4 | 418 | 40 | 5 | 202 | 142 | 1 | 8 | 51 | 202 | 95.5% |

**Tabla 10.** Validación de reglas obtenidas canal Internet.

## 6. CONCLUSIONES

En este estudio se analizó la problemática del descubrimiento de amenazas de fraude electrónico en transacciones financieras que involucra los canales POS e Internet. Los resultados experimentales de la investigación son alentadores mostrando que el método empleado es capaz de alcanzar una buena precisión.

En la revisión de literaturas para la detección de fraude; los resultados de los autores muestran que las técnicas de detección de fraude basado en algoritmos evolutivos han alcanzado un mejor nivel de precisión, el ratio TP llegó a 100% a comparación de los otros modelos.

El algoritmo genético planteado sigue el enfoque Iterative Rule Learning (IRL) donde, la solución global está formada por las mejores reglas de una serie de ejecuciones sucesivas. En el diseño del algoritmo, se ha seleccionado la Codificación Natural como la forma de representación del individuo, permitiéndonos disminuir el cardinal del conjunto de posibles soluciones.

Se ha desarrollado un algortimo genético, el cual permite conocer el hábito de compra del tarjetahabiente peruano con base a la discretización de 20 variables continuas, de las cuales 10 son analizadas para el canal POS y 10 para el canal Internet. En el proceso de entrenamiento del algoritmo se observó que el aumento en el número de individuos de la población se obtuvo como resultado la reducción del número de reglas de decisión y un mejor nivel de cobertura. En el análisis del cuarto escenario del canal POS se obtuvo 5 reglas (4

reglas de transacciones genuinas y 1 regla de transacciones fraudulentas) con un porcentaje del 97.7% de cobertura de los 525 registros de muestra. Para el caso del canal Internet, el cuarto escenario obtuvo 5 reglas (4 reglas de transacciones genuinas y 1 regla de transacciones fraudulentas) con un porcentaje del 99.8% de cobertura de 418 registros.

En la verificación del modelo la precisión de la predicción aumentó progresivamente con base al incremento de los individuos en cada escenario. En la revisión del cuarto escenario del canal POS, la precisión de las variables TP y TN bordeó los 230 aciertos que equivale al 95.8% de un total de 240 registros; las transacciones que el sistema dejó pasar siendo fraudulentas (FP) y aquellas transacciones que no permitió realizar la operación (FN) de pago sumó 10 registros. Para el caso del canal Internet el cuarto escenario obtuvo la precisión de 95.5% que equivale a 193 aciertos de 202 operaciones, y fueron 9 transacciones no predichas correctamente por las reglas.

## 7. REFERENCIAS BIBLIOGRÁFICAS


[1] Richardson Robert, 2010/2011 Computer Crime and Security Survey [Encuesta]. 15ta edición anual de CSI - Computer Security Institute, 2011.

[2] CyberSource Corporation, Informe sobre fraude en comercio electrónico 2012 – Tendencias de fraude en pagos en línea, prácticas de los comercios y comparativos [Encuesta]. 13ra edición anual de CyberSource, 2012.

[3] Frost A. y White Sullivan, Retos Clave Contra el Fraude Electrónico en las Instituciones Bancarias y Financieras de Latinoamérica [Publicación]. Octubre del 2010 Frost & Sullivan.

[4] Panigrahi Suvasini, Kundu Amlan, Sural Shamik, Majumdar A.K., Credit card fraud detection: A fusion approach using Dempster–Shafer theory and Bayesian learning [Publicación]. Journal Information Fusion, octubre de 2009. Vol. 10 issue 4, págs 354 – 363.

[5] Bentley Peter J., Kim Jungwon, Jung Gil H. y Choi Jong U., Fuzzy Darwinian Detection of Credit Card Fraud [Publicación]. In 14th Annual Symposium of the Korean Information Processing Society, 2000.

[6] Benson Edwin Raj, Portia A. Annie, Analysis on Credit Card Fraud Detection Methods [Publicación]. International Conference on, Computer, Communication and Electrical Technology (ICCCET), marzo del 2011. Págs 152 – 156.

[7] Kundu Amlan, Panigrahi Suvasini, Sural Shamik y Majumdar Arun K., BLAST-SSAHA Hybridization for Credit Card Fraud Detection [Publicación], Journal IEEE Transactions on: dependable and secure computing, diciembre 2009. Vol. 6, issue 4, págs 309 – 315.

[8] Srivastava Abhinav, Kundu Amlan, Sural Shamik, Majumdar Arun K., Credit Card Fraud Detection Using Hidden Markov Model [Publicación]. IEEE transactions on, dependable and secure computing, enero de 2008. Vol. 5.

[9] Adeyiga J. A., Ezike J. O., Omotosho A. y Amakulor W., A Neural Network Based Model for Detecting Irregularities in e-Banking Transactions [Publicación]. African Journal of Computing y ICT, diciembre del 2011. Vol. 4, issue 2.

[10] Herrera Francisco y Magdalena Luis, Genetic Fuzzy Systems: A Tutorial [Publicación]. Tatra Mountains Mathematical Publications, junio 1997. Vol. 13, págs 93-121.

[11] González Antonio. and Pérez Raúl, SLAVE: a genetic learning system based on an iterative approach [Publicación]. Journal IEEE Transactions on: Fuzzy Systems, 1999. – Vol. 7, págs. 176 – 191.

[12] Cordón O., Herrera F., Lozano M. y M. J. del Jesus, MOGUL: A Methodology to Obtain Genetic fuzzy rule-based systems Under the iterative rule Learning approach [Publicación]. International Journal of Intelligent Systems, 1998. – Vol. 14.

[13] Carvalho Deborah R. y Freitas Alex A., A genetic algorithm for discovering small disjunct rules in data mining [Publicación]. Applied Soft Computing 2002.

[14] Giráldez R. Raúl, Mejoras en eficiencia y eficacia de algoritmos evolutivos para aprendizaje supervisado [Tesis]. Sevilla: Universidad de Sevilla, departamento de lenguajes y sistemas informáticos, setiembre 2003.

[15] Noda E., Freitas Alex A. y Lopes H. S., Discovering Interesting Prediction Rules with a Genetic Algorithm [Publicación]. Evolutionary Computation, 1999. CEC 99. Proceedings of the 1999 Congress on. – Vol. 2, ISBN: 0-7803-5536-9.