ESCUELA TÉCNICA SUPERIOR DE INGENIERÍA INFORMÁTICA

DEPARTAMENTO DE LENGUAJES Y CIENCIAS DE LA COMPUTACIÓN

Procedimientos de explotación de información para la identificación de datos faltantes con ruido e inconsistentes.

TESIS DOCTORAL

AUTOR
HORACIO DANIEL KUNA

DIRECTOR
Francisco R. Villatoro Machuca

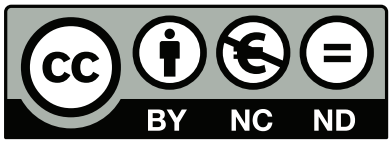
Co-Director
Ramón García Martínez

UNIVERSIDAD DE MÁLAGA

MARZO 2014

Publicaciones y Divulgación Científica
UNIVERSIDAD DE MÁLAGA

AUTOR: Horacio Daniel Kuna
EDITA: Publicaciones y Divulgación Científica. Universidad de Málaga



Dr. Francisco Villatoro Machuca, Profesor Titular de la Universidad de Málaga, como director de la Tesis Doctoral de D^{n}. Horacio Daniel Kuna

Dr. Ramón García Martínez. Profesor Titular de la Universidad Nacional de Lanús (Argentina), como co-director de la Tesis Doctoral de D^{n}. Horacio Daniel Kuna

CERTIFICAN:

Que D^{n}. Horacio Daniel Kuna, Licenciado en Sistemas, ha realizado en el Departamento de Lenguajes y Ciencias de la Computación de la E.T.S. de Ingeniería Informática de la Universidad de Málaga, bajo nuestra dirección, el trabajo de investigación correspondiente a su Tesis Doctoral titulado:

Procedimientos de explotación de información para la identificación de datos faltantes con ruido e inconsistentes.

Revisado el presente trabajo, estimo que puede ser presentado al Tribunal que ha de juzgarlo. Y para que conste a efectos de lo establecido en la legislación vigente, autorizo la presentación de este trabajo en la Universidad de Málaga.

Málaga, de de 2014.

Fdo. Dr. Ramón García Martínez

Fdo. Dr. Francisco Villatoro Machuca

Agradecimientos

La realización de esta tesis doctoral ha sido el fruto de muchos años de estudio e investigación y fue posible gracias a muchas personas e instituciones que han contribuido durante todos estos años.
Las primeras palabras de agradecimiento son para mí director y co-director Doctores Francisco Villatoro Machuca y Ramón García Martínez, el apoyo, orientación y acompañamiento brindado por ambos fue de fundamental importancia.
Quiero agradecer a los docentes y autoridades del doctorado.
Dedico un especial agradecimiento a mis colegas del programa de investigación en computación de la Facultad de Ciencias Exactas Químicas y Naturales de la Universidad Nacional de Misiones.
Al capítulo Buenos Aires de ISACA por haber auspiciado esta tesis.
A mi mujer Graciela, a mis hijas Paula y Andrea, por su amor, comprensión y apoyo.

Índice

1 Introducción. 17
1.1 Motivación. 17
1.2 El Problema. 18
1.3 Contenido de la tesis. 20
2. La minería de datos en la auditoría de sistemas. 21
2.1 La auditoría de sistemas. 21
2.1.1 Definición de la auditoría de Sistemas. 21
2.1.2 Objetivos de la auditoría de sistemas. 22
2.1.3 Rol de los auditores de sistemas. 23
2.1.4 Técnicas utilizadas en la auditoría de sistemas 23
2.1.5 Auditoría asistida por computadora. 24
2.1.6 Estándares, normativas, leyes utilizados en la Auditoría de Sistemas. 27
2.1.6.1 COBIT (*Control Objectives for Information and related Technology).* 28
2.1.6.2 Normas ISO. 29
2.1.6.3 ISACA. 31
2.1.6.4 Otras normas, leyes, estándares y buenas prácticas. 34
2.2 La minería de datos. 36
2.2.1 Introducción a la minería de datos. 36
2.2.2 Tipos de modelos de minería de datos. 37
2.2.3 Proceso de explotación de información 38
2.2.4 Técnicas de la minería de datos. 40
2.3 Minería de datos y auditoría de sistemas. 43
2.3.1 Auditoria continúa y minería de datos 43
2.3.2 Minería de Datos para la detección de fraudes en el área de finanzas y contabilidad. 46
2.3.2.1 Herramientas DM utilizadas en el área de finanzas y contabilidad. 51
2.3.2.2 Algoritmos para la detección de fraudes producidos por la alta gerencia. 55
2.3.3 Minería de Datos para la detección de intrusos en redes de telecomunicaciones. 58
2.3.3.1 Algoritmos utilizados en la detección de intrusos en redes de telecomunicaciones. 60
2.3.4 Minería de Datos para la detección de terroristas. 62
2.3.4.1 Data Mining Investigativo. 63
2.3.4.2 Sistemas de Detección de Terroristas. 64
2.4 Minería de datos y detección de outliers. 65
2.4.1 Enfoques y métodos para abordar el problema de la detección de outliers 72
2.4.2 Criterios para la elección de métodos de detección de outliers. 75
2.4.3 Taxonomía de metodologías de detección de outliers. 76
2.4.3.1 Métodos univariantes y los métodos multivariantes. 76
2.4.3.2 Métodos paramétricos y métodos no paramétricos. 77
2.4.3.3 Métodos basados en la estadística, mét. bas.en la distancia y mét. bas.en la densidad 77
2.4.3.4 Métodos basados en técnicas de clustering. 78
2.4.3.5 Métodos basados en redes neuronales 80
2.4.3.6 Otros métodos de detección de outliers 82
2.4.3.7 Resumen de métodos de detección de outliers. 86
2.4.4 Comparación de métodos de detección de outliers. 87

2.5 Discusión estado del arte y problema detectado. 89
3 Materiales y Métodos. Algoritmos y bases de datos utilizadas. 91
3.1 Algoritmo basado en la densidad. LOF (Local Outlier Factor). 92
3.2 Algoritmo K-Means. 94
3.3 Algoritmo de inducción C4.5. 95
3.4 Teoría de la información. 96
3.5 Redes bayesianas. 97
3.6 DBSCAN. 99
3.7 PRISM. 101
3.8 Bases de datos utilizadas en la experimentacón 101
4. Solución. 103
4.1 Consideraciones generales. 103
4.2 Procedimientos para la detección de outliers en bases de datos numéricas. 105
4.2.1 Procedimiento 1. 106
4.2.2 Procedimiento 2. 109
4.2.3 Experimentación Procedimientos 1 y 2. 110
4.2.3.1 Determinación del valor óptimo de LOF y sus parámetros. 112
4.2.3.2 Determinación de outliers aplicando métodos estadísticos. 113
4.2.3.3 Pruebas realizadas sobre Bases de Datos basadas en la distribución normal. 114
4.2.3.4 Experimentación en una base de datos real de los procedimientos 1 y 2. 120
4.2.3.4.1 Experimentación con el procedimiento 1. 121
4.2.3.4.2 Experimentación con el procedimiento 2. 122
4.2.3.5 Resultados y disc. de la aplicación de los procedimientos 1 y 2 para BD numéricas. 124
4.3 Procedimiento 3. Orientado a bases de datos alfanuméricas con un atributo target. 126
4.3.1 Experimentación. 129
4.3.2 Resultados de la aplicación del procedimiento desarrollado para BD alfanuméricas. 130
4.3.3 Discusión del procedimiento 3 desarrollado para la base de datos alfanumérica. 132
4.4 Procedimiento 4. Orientado a bases de datos alfanuméricas sin un atributo target. 133
4.4.1 Algoritmos seleccionados. 134
4.4.2 Algoritmos específicos para la detección de outliers considerados. 134
4.4.3 Algoritmos de clasificación considerados. 135
4.4.4 Proceso de selección de algoritmos. 136
4.4.5 Selección de algoritmos diseñados específicamente para detectar outliers. 137
4.4.5.1 Unión de los resultados de aplicar LOF y DBSCAN. 138
4.4.5.2 Reglas de determinación de outliers para algoritmos LOF y DBSCAN. 139
4.4.5.3 Combinación de LOF y DBSCAN. 140
4.4.6 Selección de algoritmos de clasificación. 141
4.4.6.1 Combinación de algoritmos de clasificación. 142
4.4.6.2 Reglas de determinación de outliers para algoritmos de clasificación. 143
4.4.6.3 Resultados de ejecutar los alg. C4.5, redes Bayesianas y PRISM en forma individual. 144
4.4.7 Diseño del procedimiento propuesto. 145
4.4.8 Experimentación sobre una base de datos real. 148
4.4.8.1 Resultados obtenidos con el procedimiento 4. 150
4.4.8.2 Discusión de la experimentación realizada con el procedimiento 4. 150
4.5 Discusión de las soluciones propuestas. 151
5. Conclusiones y futuras líneas de investigación. 155

5.1 Aportaciones de la tesis. 155
5.2 Futuras líneas de investigación. 158

Índice de figuras

Figura 2.1. Proceso de KDD 37
Figura 2.2. Relación Auditoría Continua y Minería de datos 46
Figura 2.3. Regla empírica para detectar outliers en una distribución normal 66
Figura 2.4. Grafico de bigote para definir outliers. 67
Figura 2.5. Representación grafica de outliers 69
Figura 2.6. Componentes de las técnicas de detección de outliers 71
Figura 2.7. Outliers de tipo 1. Data set relacionado con la actividad vitivinícola. 73
Figura 2.8. Outliers de tipo 2. Data set relacionado con la actividad vitivinícola. 74
Figura 4.1. Procedimiento 1 para BD numéricas 108
Figura 4.2. Procedimiento 2 para BD numéricas. 111
Figura 4.3. Histograma base de Base 2000 registros 115
Figura 4.4. Ejemplo de clusterización de la columna 12 122
Figura 4.5. Distancia del centroide de la columna 12. 123
Figura 4.6. Criterio de certeza 125
Figura 4.7. Resumen de outliers detectado 126
Figura 4.8. Procedimiento para la detección de outliers en bases de datos alfanuméricas 128
Figura 4.9. Unión de algoritmos 139
Figura 4.10. Unión de los algoritmos C4.5, RB y PRISM 142
Figura 4.11. Procedimiento propuesto. 147

Índice de tablas

Tabla 2.1. Relación entre técnicas y métodos de minería de datos 42
Tabla 2.2. Publicaciones de artículos relacionados con fraudes reales 51
Tabla 2.3. Herramientas testeadas 52
Tabla 2.4. Hardware y software utilizado en las pruebas 53
Tabla 2.5. Algoritmos de cada herramienta 53
Tabla 2. 6. Resultados de la comparación entre categorías y productos software de minería de datos 54
Tabla 2.7. Resultados con los modelos sin entrenar 57
Tabla 2.8. Resultados con los modelos entrenados 58
Tabla 2.9. Data Mining tradicional comparado con IDM 64
Tabla 2.10. Enfoques por metodología 87
Tabla 4.1. Valores Obtenidos en las pruebas realizadas aplicando el procedimiento propuesto sobre la BD generada, con valores de Límite de LOF igual 1.5 116
Tabla 4.2. Valores Obtenidos en las pruebas realizadas aplicando el procedimiento propuesto sobre la BD generada, con 200 registros variando los valores de *MinPtsMin, MinPtsMax* y LOF 117
Tabla 4.3. Valores obtenidos en las pruebas realizadas aplicando el procedimiento propuesto sobre la BD generada, con 400 registros variando los valores de *MinPtsMin, MinPtsMax* y LOF 118
Tabla 4.4. Valores Obtenidos en las pruebas realizadas aplicando el procedimiento propuesto sobre la BD generada, con 2000 registros variando los valores de MinPtsMin, MinPtsMax y LOF 119
Tabla 4.5. Numero de tupla de la Base de Datos con valores de LOF > 1.5 121
Tabla 4.6. Outliers detectados por el procedimiento 1 121
Tabla 4.7. Resumen outliers por cluster 2 123
Tabla 4.8. Outliers detectados por el procedimiento 2 124
Tabla 4.9. Atributos de la Base de Datos de Hongos 129
Tabla 4.10. Outliers detectados por cada atributo de la base de datos de Hongos 132
Tabla 4.11. Características BD artificial 137
Tabla 4.12. Resultados de los algoritmos para la detección de outliers 138
Tabla 4.13. Comparación resultados 141
Tabla 4.14. Resultado de aplicar los algoritmos de uso general 142
Tabla 4.15. Resultados de la aplicación de algoritmos de clasificación 145
Tabla 4.16. Tablas de logs seleccionadas para el análisis 148
Tabla 4.17. Resultados obtenidos 150
Tabla 4.18. Resumen enfoque de los algoritmos desarrollados 153

"La imaginación es más importante que el conocimiento. El conocimiento es limitado, mientras que la imaginación no"

Albert Einstein

1 Introducción.

1.1 Motivación.

En la actualidad los sistemas de información son cada vez más complejos, integrados y relacionados, garantizar la gobernanza de la tecnología de la información es una instancia crítica y fundamental para lograr que las empresas y organismos públicos y privados sean competitivos, eficaces y eficientes en el cumplimiento de los objetivos trazados. La dependencia que tienen las organizaciones de la información y de los sistemas informáticos que la gestionan, convierte a los mismos en factores críticos de éxito.

La tecnología de la información es una herramienta fundamental que debe dar soporte y estar alineada con los objetivos estratégicos de una empresa. Para garantizar esto es necesario implementar sistemas de control interno que protejan todos los bienes vinculados con la tecnología de la información, el personal, las instalaciones, la tecnología, los sistemas de aplicación y los datos.

Las empresas y organismos de todos los tamaños deben garantizar el cumplimiento de las normas y procedimientos definidos en relación con la administración de la tecnología de la información, la auditoría de sistemas entendida como un proceso formal que realiza una revisión y control sobre todos los aspectos relacionados con la tecnología de la información pasa a tener un rol central en el objetivo relacionado con tener información de calidad y segura. La auditoría aplica distintas técnicas en su proceso, una de ellas es la implementación de técnicas de auditoría asistida por computadora, una de las técnicas más utilizadas para asistir al auditor en su tarea es la minería de datos entendida como un proceso de descubrimiento de patrones de comportamiento de los datos de manera automática.

De acuerdo a la información relevada se detecta que un alto porcentaje de pequeñas y medianas empresas no tiene un proceso formal de auditoría de sistemas, centrándose este tipo de actividad en las grandes empresas, en las empresas internacionales y en las empresas vinculadas con el sector bancario

y financiero. En particular la aplicación de técnicas de auditoría asistida por computadora tiene una aplicación aún más limitada, en general relacionada con el desconocimiento por parte de los auditores de las herramientas existentes, su uso, aplicabilidad y beneficios.

Siendo entonces la información hoy la columna vertebral de cualquier empresa u organización, garantizar la calidad de la misma es un requerimiento central que debe ser tenido en cuenta dentro de las empresas y organizaciones. Los datos se encuentran almacenados en distintos formatos, en general están en bases de datos relacionales, por diferentes razones existen datos que son considerados anómalos, es decir que son diferentes al resto de los datos, esto puede deberse a algún tipo de error o por un acto malintencionado, detectar este tipo de dato es de fundamental importancia para garantizar que la información tenga características de seguridad, legalidad y calidad. El auditor de sistemas es un recurso humano escaso, altamente calificado y en muchos casos caro para organizaciones y empresas de mediano y pequeño tamaño, algunas empresas utilizan auditores sin la suficiente formación y experiencia para realizar tareas de evaluación y control de todo lo relacionado con la tecnología de la información. Contar con procedimientos formales que colaboren en la detección de campos anómalos en bases de datos ayudaría de manera fundamental en la aplicación de las mejores herramientas, métodos y algoritmos en la detección de aquellos datos considerados anómalos.

Motiva la presente tesis la inexistencia de algoritmos o procedimientos que permitan la detección de campos con valores anómalos en bases de datos, siendo esta detección de fundamental importancia para la tarea que realiza el auditor de sistemas.

1.2 El problema.

Existe una gran cantidad de algoritmos relacionados con la minería de datos y la estadística (Hawkings, 1980; Chandola et al., 2009; Hodge & Austin, 2004; Mansur et al., 2005) que detectan las tuplas que pueden considerarse

anómalas en bases de datos, el problema identificado es que estos algoritmos no detectan específicamente que campos dentro de esas tuplas son los que tiene valores que de alguna manera implican un ruido o anomalía dentro de la base de datos. En el caso de bases de datos complejas con gran cantidad de campos en sus filas se hace difícil para un auditor detectar específicamente cual es el campo que tiene sospecha de haber sido creado de manera distinta al resto de los campos, realizar esta tarea requiere por parte del auditor de sistemas una gran cantidad de tiempo, y es requisito que el mismo cuente con la suficiente experiencia para poder abordar esta búsqueda.

Se observa en la bibliografía que bajo escenarios específicos existen numerosos algoritmos que permiten la detección de filas consideradas anómalas, en general las bases de datos relacionadas con los sistemas de gestión no responden a una distribución predefinida en sus datos y por lo tanto se desconoce previamente cuales son los campos considerados anómalos, la validación empírica de la calidad de los algoritmos dificulta en la mayoría de los casos el proceso de generalización de la aplicación de cada algoritmo a otro tipo de entorno. Es necesario entender que cada algoritmo desarrollado es el mejor bajo determinadas condiciones de los escenarios relacionados con el set de datos a analizar.

El auditor de sistemas es un profesional que requiere tener un alto nivel de capacitación siendo recurso humano escaso y caro, por lo tanto desarrollar procedimientos de explotación de información para detectar campos considerados anómalos es un verdadero aporte que posibilita sistematizar la búsqueda de ruido en las bases de datos agregándole objetividad, eficacia y eficiencia a la tarea.

La presente tesis tiene como objetivos:

- Establecer una taxonomía relacionada con los métodos, técnicas y algoritmos de detección de valores anómalos en bases de datos, analizando las ventajas y desventajas de cada una de ellos.
- Diseñar y validar procedimientos de explotación de información que combinados entre sí permitan detectar los campos que tienen valores atípicos en bases de datos, combinando distintas técnicas de

minería de datos, entre ellas específicamente las de clustering, algoritmos de inducción, redes bayesianas, principios de la teoría de la Información, entre otras, para lograr mejorar de esa manera la calidad de los datos.

1.3 Contenido de la tesis.

Para obtener los objetivos mencionados esta memoria está organizada de la siguiente manera:

Capítulo 2: Realiza un introducción a la auditoría de sistemas; describiendo las distintas técnicas, estándares, normativas y leyes que se utilizan. También se desarrollan los conceptos fundamentales relacionados con la minería de datos y su aplicación en el proceso de auditoría de sistemas. Se realiza una taxonomía de las principales aplicaciones de la minería de datos en el proceso de auditoría de sistemas, presentándose los principales métodos y enfoques de la minería de datos en la detección de valores anómalos, realizándose un resumen de los métodos de detección de valores anómalos detectados.

Capítulo 3: Este capítulo realiza una revisión de los algoritmos utilizados en la propuesta de solución presentada que se relaciona con la detección de campos considerados anómalos, así como los tipos de bases de datos utilizadas para validar la experimentación.

Capítulo 4. Se presenta la solución propuesta, basada en el diseño de cuatro procedimientos, dos de ellos destinados a detectar valores anómalos en bases de datos numéricas, uno para bases de datos alfanuméricas con un atributo target, uno para detectar campos considerados anómalos en bases de datos alfanuméricas sin un atributo target.

Capítulo 5: este capítulo presenta las principales conclusiones que surgen de la tesis desarrollada y las futuras líneas de investigación.

2. La minería de datos en la auditoría de sistemas.

En este capítulo se realiza una introducción a la auditoría de sistemas (sección 2.1), se describen las características fundamentales de la minería de datos (sección 2.2), se realiza un análisis de las principales aplicaciones de la minería de datos en el área de auditoría de sistemas (sección 2.3), se describen los principales enfoques, algoritmos y técnicas utilizados por la minería de datos en la detección de datos anómalos (sección 2.4), presentándose en la misma sección una taxonomía de métodos utilizados en la detección de datos anómalos, finalmente se realiza una discusión sobre el estado del arte descripto en este capítulo (sección 2.5).

2.1 La auditoría de sistemas.

En esta sub-sección se desarrolla en el punto 2.1.1 la definición de la auditoría de sistemas, en el punto 2.1.2 se explican los principales objetivos de la auditoría de sistemas, en el punto 2.1.3 se describe el rol del auditor de sistemas, en el punto 2.1.4 se explican las principales técnicas utilizadas en la auditoría de sistemas, en el punto 2.1.5 se desarrollan los conceptos fundamentales relacionados con la auditoria asistida por computadora y en el punto 2.1.6 se presentan los estándares utilizados en la auditoría de sistemas.

2.1.1 Definición de la auditoría de sistemas.

Según Rivas (Rivas, 1998) la auditoría de sistemas *"es el conjunto de técnicas, actividades y procedimientos, destinados a analizar, evaluar, verificar y recomendar en asuntos relativos a la planificación, control, eficacia, seguridad y adecuación de los sistemas de información en la empresa"*.

Es función de los auditores de sistemas de información (ISACA, 2013) examinar y evaluar todos los aspectos relacionados con el desarrollo, la implementación, el mantenimiento y la operación del conjunto de componentes

de los sistemas informáticos y todas las interfaces que los mismos poseen con sistemas externos tanto manuales como automatizados.

2.1.2 Objetivos de la auditoría de sistemas.

La auditoría de los sistemas informáticos tiene por objeto examinar y evaluar la calidad y suficiencia de los controles establecidos por la empresa u organismo para lograr su mejor funcionamiento.

El auditor deberá formarse opinión e informar acerca de la razonabilidad de tales controles, dando cuenta de los hallazgos donde se detecta que se producen desviaciones con respecto al sistema de control interno vigente, y recomendando las propuestas para su mejora.

Se debe analizar y comprobar el funcionamiento del sistema de información, teniendo en cuenta los objetivos del control interno, según SIGEN, (2002) los mismos son:

- *"La emisión de información financiera y operativa confiable, íntegra, oportuna y útil para la toma de decisiones."*
- *"El cumplimiento de las leyes y normas aplicables."*
- *"La protección de los activos y demás recursos, incluyendo actividades para la disuasión de fraudes y otras irregularidades."*
- *"El conocimiento, por parte de la dirección superior, del grado de consecución de los objetivos operacionales, sobre la base de la aplicación de criterios de eficacia, eficiencia y economía."*

La tarea del auditor de sistemas tiene por objetivo el determinar si el sistema de control interno que está implementado tiene una estructura que brinde un razonable grado de seguridad que garantice el cumplimiento de los objetivos de la empresa.

El auditor debe informar a la alta dirección todos los aspectos que se relacionen con la tecnología de la información donde se evalúe que los mismos no tengan el funcionamiento esperado y que, de manera inmediata, deberían ser subsanados por las autoridades pertinentes, los auditores deberán

especificar los hallazgos y realizar las recomendaciones necesarias para poder corregir las deficiencias detectadas.

Es necesario conocer que bienes se deben proteger para iniciar cualquier proceso de auditoría, estos bienes son (COBIT, 2013):

- Datos, en todos sus formatos.
- Aplicaciones, que son el conjunto de sistemas informáticos.
- Tecnología, es el conjunto de hardware y software de base utilizado para operar las aplicaciones.
- Instalaciones, son todos los recursos necesarios para alojar a los sistemas de información.
- Recursos Humanos, es el bien más importante a proteger, se trata de todo el personal que se relaciona en forma directa con el desarrollo y producción de los sistemas de información.

2.1.3 Rol de los auditores de sistemas.

Se establecen tres tipos de funciones para los auditores de sistemas (Piattini, 2003):

- Participar en la revisión del diseño, programación, implantación y explotación de las aplicaciones informáticas.
- Revisar y evaluar los controles implementados en los sistemas informáticos.
- Revisar y evaluar la eficacia, eficiencia, utilidad, fiabilidad y seguridad del total de equipos informáticos que se utilizan así como la información que se ingresa y genera.

2.1.4 Técnicas utilizadas en la auditoría de sistemas.

Diversas técnicas son utilizadas en la auditoría informática, estas técnicas son utilizadas a lo largo de todo el ciclo de vida del sistema, algunas de ellas son:

- *Revisión de documentos y datos*: se trata del análisis y evaluación de la documentación y datos generados dentro de la empresa, tanto la información relacionada con la operatoria cotidiana, la información orientada a la toma de decisiones, las bases de datos, los planes desarrollados por la empresa, las auditorías anteriores y toda otra documentación que sea de interés para la auditoría.
- *Entrevistas:* las entrevistas son un conjunto de preguntas que se realizan en forma personal a un determinado usuario con el objetivo de recabar información de utilidad dentro del proceso de auditoría, las entrevistas pueden ser estructuradas o no estructuradas, con preguntas abiertas o cerradas y con una estructura inductiva o deductiva.
- *Observación:* en esta técnica el auditor observa en forma pasiva como se realizan las tareas cotidianas dentro de una empresa.
- *Análisis por muestreo:* el auditor evalúa un conjunto parcial del total de la información disponible.

2.1.5 Auditoría asistida por computadora.

Diferentes técnicas se utilizan en el proceso de auditoría, muchas de ellas relacionadas con el uso del ordenador dentro del proceso. Algunas normas internacionales se refieren a este tema, la norma SAP 1009 (SAP, 2012), denominada *Computer Assisted Audit Techniques* (CAATs)[1] o Técnicas de Auditoría Asistidas por Computador (TAACs), remarca la importancia del ordenador en el proceso de auditoría de sistemas, esta norma define a las CAATs como el conjunto de programas de ordenador y el conjunto de datos que el auditor utiliza durante el proceso de auditoría, tanto para recolectar como para analizar datos e información.

Las CAATs, de acuerdo a la norma SAP 1009, pueden ser usadas en:

[1] Algunos autores remplazan el término *Assisted* por *Aided.* En alguna bibliografía se utiliza como *Computer Assisted Audit Tools & Techniques* (CAATTs)

- Pruebas que se realizan sobre un determinado set de datos para analizar y evaluar el detalle de las transacciones y el balanceo de las mismas.
- Procedimientos analíticos, por ejemplo, identificación de inconsistencias o fluctuaciones significativas. En este caso las herramientas automatizan la búsqueda de información que puede considerarse anormal, por ejemplo valores altos de venta, números repetidos o faltantes de determinados comprobantes, etc.
- Herramientas que permiten realizar pruebas de controles generales, tales como configuraciones en sistemas operativos, análisis de software instalado en los ordenadores, procedimientos de acceso al sistema, comparación de códigos fuentes y ejecutables, comparaciones de contenidos y estructuras de bases de datos, etc.
- Programas que automatizan la extracción de datos.
- Pruebas de control en aplicaciones.
- Programas que posibilitan el re-calculo de determinadas operaciones que realizan las aplicaciones para evaluar si los resultados generados por los sistemas son adecuados.

Es importante destacar los cuidados que deben tener los auditores a la hora de usar este tipo de herramientas, su aplicación nunca debe significar una alteración del contenido de los datos que se encuentran en producción ni una alteración al normal funcionamiento de los sistemas.

Existen diferentes tipos de software relacionados con las CAATs:

Paquete de Auditoría. Son programas de uso comercial específicamente diseñados para asistir al auditor en su tarea, como por ejemplo para analizar el contenido de bases de datos, seleccionar información, realización de cálculos para verificar transacciones, crear archivos de datos para su posterior análisis, imprimir informes en un formato especificado, verificar el cumplimiento de buenas prácticas, etc. Estos paquetes son usados en las bases de datos para el control de secuencias, búsquedas de registros, detección de duplicaciones, detección de gaps, selección de datos, revisión de

operaciones lógicas y muestreo, algunos de ellos son el IDEA[2], ACL[3], etc. También existen paquetes que tienen incorporados estándares de auditoría con el objetivo de verificar su cumplimiento, como es el caso de los productos de MEYCOR[4] , muchos de ellos tienen incorporado el estándar COBIT.

Software para un propósito específico o diseñado a medida. Son programas de ordenador diseñados para desempeñar tareas de auditoría en determinados procesos específicos de la auditoría. Estos programas pueden ser desarrollados por el auditor, por la entidad que está siendo auditada, también estas herramientas pueden ser desarrolladas por un programador externo. En algunos casos cuando el software es desarrollado por la empresa los programas pueden ser usados en su formato original o para garantizar mayor independencia y eficiencia pueden ser modificados a pedido del equipo de auditoría. Dentro de esta categoría pueden incluirse por ejemplo programas que permitan generar check-list adaptados a las características de la empresa y de los objetivos específicos de la auditoría.

Los programas de utilería. Son usados por la organización auditada para desarrollar funciones comunes de procesamiento de datos, como clasificación, creación e impresión de archivos. Como por ejemplo, planillas de cálculo, procesadores de texto, etc. El uso de este tipo de programas debe ser complementario a la implementación de software específicamente diseñado para la auditoría.

Los programas de administración del sistema. Son herramientas software que normalmente son parte de los sistemas operativos sofisticados, sistemas de gestión de bases de datos y en algunos casos también de aplicaciones, por ejemplo, software para recuperación de datos, comparación de códigos fuentes y ejecutables, comparaciones de contenidos y estructuras de bases de datos, etc. Como en el caso anterior estas herramientas no son específicamente diseñadas para usos de auditoría. Existen en el mercado una gran variedad de este tipo de herramientas software como por ejemplo, los que

[2] IDEA. Disponible en www.**ideas**oftware.com/ (Consultado el 08/07/2013)
[3] ACL. Disponible en www.**acl**.com/ (Consultado el 08/07/2013)
[4] MEYCOR. Disponible en http://www.meycor-soft.com/en/meycor-cobit-csa-control-self-assessment (consultado el 26/09/2013)

permiten controlar las versiones de un sistema, p.ej. *Subversion* (Subversion, 2013).

Rutinas de Auditoría embebidas en Programas de aplicación. Módulos especiales de recolección de información incluidos en la aplicación y diseñados con fines específicos. Se trata de módulos del sistema de aplicación que posibilitan el registro en lo que se denominan *logs de auditoría* de las operaciones que se realizan dentro del sistema. Dependiendo del diseño en este tipo de rutinas se pueden registrar también (aparte de la operación, la fecha, la hora) el valor anterior de la transacción y el valor después de la modificación realizada.

Sistemas expertos. Se trata de sistemas basados en inteligencia artificial que simulan la tarea que puede realizar un experto en auditoría, entre otras aplicaciones utilizados en la gestión de riesgos, en el seguimiento de incidentes, etc.

2.1.6 Estándares, normativas, leyes utilizados en la Auditoría de Sistemas.

El uso de estándares, normativas, leyes y buenas prácticas se ha convertido en un elemento central para ayudar a garantizar que la información sirva para obtener los objetivos estratégicos, tácticos y operativos de las empresas.

A nivel internacional existen diferentes normas que intentan estandarizar el proceso de la auditoría de sistemas, algunas de ellas se describen en las siguientes sub-secciones, en 2.1.6.1 se explica COBIT, en el punto 2.1.6.2 se desarrollan las características básicas de las normas ISO relacionadas con el proceso de auditoría de sistemas, en el punto 2.1.6.3 se describen los estándares propuestos por ISACA y en el punto 2.1.6.4 se presentan otras leyes, estándares y buenas prácticas utilizadas durante la auditoría de sistemas.

2.1.6.1 COBIT *(*Control Objectives for Information and related Technology).

La misión y objetivo de COBIT (COBIT, 2013) *"es investigar, desarrollar, publicar y promover un conjunto de objetivos de control en tecnología de la información (TI), que los mismos tengan autoridad, estén actualizados, que sean de carácter internacional y aceptados generalmente para el uso cotidiano de gerentes de empresas y auditores, brindando un marco de negocios para el gobierno y la gestión de la Tecnología de la Información en las empresas."*

La *Information Systems Audit and Control Foundation* (ISACA)[5] y los patrocinadores de COBIT, diseñaron este conjunto de buenas prácticas con el objetivo que las mismas sean una fuente de guía e instrucción para los auditores de sistemas, en la actualidad la versión vigente es la 5.

El objetivo de COBIT es que el mismo sirva como un estándar que permita mejorar el conjunto de prácticas de control y seguridad de la TI, brindando también un marco de referencia para los directivos de las empresas, los usuarios y los propios auditores.

COBIT propone un marco de trabajo integral que debe ser una ayuda para que las empresas alcancen los objetivos para el gobierno y la gestión de las TI. COBIT posibilita que las Tecnologías de la Información sean gestionadas de un modo holístico, es un principio de este estándar el abordar en todos los niveles y estructuras de una empresa los problemas relacionados con la gobernanza de la TI. COBIT fue creado para ser aplicado a empresas u organismos de cualquier tamaño, sean comerciales, del sector público o entidades sin fines de lucro.

COBIT tiene los siguientes principios:

- Satisfacer las necesidades de las partes interesadas, tanto sea del sector operativo, gerencial o el relacionado con la dirección de la empresa.
- Cubrir la empresa en forma integral en todos los aspectos relacionados con garantizar la legalidad, seguridad y calidad de la información que se

[5] ISACA: https://www.isaca.org/Pages/default.aspx (visitado el 23/09/2013)

produce y genera.

- Contar y poder aplicar un marco de referencia único e integrado, evitando de esta manera tener que trabajar con muchas y diferentes normas, ya que en algunos casos existen diferentes criterios entre las normas y su aplicación se vuelve compleja.
- Hacer posible un enfoque de integración global a la hora de abordar temas relacionados con la gobernanza de TI.
- Separar el gobierno de TI de la gestión de TI dentro de las empresas u organismos.

COBIT propone una cascada de metas y métricas, se trata de un mecanismo para traducir las necesidades de las partes interesadas en metas corporativas, metas relacionadas con la TI y metas catalizadoras específicas, útiles y a medida. La cascada de metas es importante ya que ayuda a la definición de prioridades de implementación, posibilitando lograr una mejora del gobierno de la TI de la empresa, este gobierno de la TI se debe basar en las metas estratégicas de la empresa u organismo. COBIT presenta una matriz llamada *"RACI"* que brinda una sugerencia de los niveles de responsabilidad que se debe tener dentro de los procesos que se relacionan con el sistema de control interno.

2.1.6.2 Normas ISO.

La International Organization for Standardization[6] (ISO) es una organización que desarrolla y publica estándares internacionales, muchos de ellos relacionados con la seguridad de la información y aplicados en el proceso de auditoría de sistemas. Por ejemplo ha publicado entre otras las siguientes normas:

- *ISO 20000*. Es el estándar internacionalmente reconocido en gestión de servicios de tecnología de la información.
- *ISO 27001*. Fue publicada el 15 de Octubre de 2005. Se trata de la principal norma de la serie y contiene el conjunto de requisitos del

[6] ISO www.iso.org (visitado el 27/08/2013)

sistema de gestión de seguridad de la información, se origina en la norma BS 7799-2:2002. Se trata de la norma que se utiliza para certificar a los auditores externos en el *"Sistema de Gestión de la Seguridad de la Información"* (SGSI) de las organizaciones.

- *ISO 27002*. Esta norma fue publicada el 1 de Julio de 2007, es el nuevo nombre de la anterior norma ISO 17799:2005, contiene una guía de buenas prácticas que describen los objetivos de control y los controles recomendables en cuanto a la seguridad de la información. Esta norma no es certificable, la misma contiene 39 objetivos de control y 133 controles, agrupados en 11 dominios.
- *ISO 27003*. Publicada en 2008, consiste en una guía de implementación de SGSI brindando información acerca del uso del modelo "*plan–do–check–act or plan–do–check–adjust*" (PDCA) y de los requerimientos de sus diferentes fases.
- *ISO 27005*. Fue publicada el 4 de Junio de 2008. Establece las directrices relacionadas con la gestión del riesgo en la seguridad de la información. Apoya los conceptos generales especificados en la norma ISO/IEC 27001.
- *ISO 27006*. Esta norma fue publicada el 13 de Febrero de 2007, define los requisitos para la acreditación de entidades de auditoría y certificación de sistemas de gestión de seguridad de la información.
- *ISO 27007*. Consiste en una guía de auditoría de un SGSI.
- *ISO 27011*. Se relaciona con una guía de gestión de seguridad de la información específica para telecomunicaciones, elaborada conjuntamente con la ITU (Unión Internacional de Telecomunicaciones).
- ISO 27031. Se trata de una guía de continuidad del negocio en cuanto a tecnologías de la información y comunicaciones se refiere.
- *ISO 27032*. Es una guía relacionada a la ciberseguridad.
- *ISO 27033*. Se trata de una norma que tiene 7 partes: gestión de seguridad de redes, arquitectura de seguridad de redes, escenarios de redes de referencia, aseguramiento de las comunicaciones entre

redes mediante gateways, acceso remoto, aseguramiento de comunicaciones en redes mediante VPNs y diseño e implementación de seguridad en redes.

- *ISO 27034*. Es una guía de seguridad en aplicaciones.
- *ISO 27799*. Es un estándar de gestión de seguridad de la información en el sector sanitario aplicando ISO 17799 (actual ISO 27002).
- *ISO 31000*. Es una norma relacionada con la gestión de riesgos
- *ISO 38500*. El objetivo de esta norma es brindar un marco de referencia para que la dirección de los organismos y empresas la utilicen al dirigir, evaluar y controlar el uso de las tecnologías de la información.

2.1.6.3 ISACA.

ISACA (Information Systems Audit and Control Association) (ADACSI, 2013) estableció un conjunto de:

- ***Estándares:*** Los mismos son de aplicación obligatoria en el proceso de auditoría de sistemas y la generación de informes. Estos estándares son:
  - ***"510. Alcance.***
    - ***510.010 Responsabilidad, autoridad y control.***

    La responsabilidad, autoridad y control de las funciones de control de los sistemas de información deben ser adecuadamente documentadas y aprobadas por el nivel de gerencia apropiado.
  - ***520. Independencia***
    - ***520.010 Independencia profesional.***

    En todos los temas relativos al control de sistemas informáticos, el profesional de control de sistemas informáticos debe ser independiente en actitud y apariencia.
    - ***520.020 Relación organizacional.***

La función de control de sistemas informáticos debe ser lo suficientemente independiente del área a controlar para la objetividad en el desempeño de las tareas del profesional de control de sistemas informáticos.

- ***530. Ética profesional y estándares.***
  - ***530.010 Código de ética profesional****.*

El profesional de control de sistemas informáticos debe adherir al código de ética profesional propuesto por ISACA.

  - ***530.020 Debido cuidado profesional.***

En todos los aspectos del trabajo del profesional de control de sistemas de información debe ejercerse el debido cuidado profesional y la observancia de los estándares aplicables.

- ***540. Competencia.***
  - ***540.010 Experiencia y conocimientos.***

El profesional de control de sistemas de información debe ser técnicamente competente, teniendo la experiencia y conocimientos necesarios para el trabajo de control profesional.

  - ***540.020 Continuidad de la educación profesional.***

El profesional de control de sistemas de información debe mantener su competencia a través de la apropiada continuidad en su capacitación profesional. El profesional de control de sistemas de información debe mantener su competencia a través de la apropiada continuidad en su capacitación profesional.

- ***550. Planeamiento***
  - ***550.010 Planeamiento del control****.*

El profesional de control de sistemas de información debe utilizar evaluación de riesgos y otras herramientas apropiadas para el planeamiento y la priorización del

trabajo de control de sistemas informáticos para asegurar los objetivos de control.

- ***560. Desarrollo del trabajo***
  - ***560.010 Supervisión.***

Los profesionales de control de sistemas de información deben ser apropiadamente supervisados y coordinados para proveer seguridad de que se respetan los objetivos de control y se aplican los estándares profesionales.

  - ***560.020 Evidencia.***

Los profesionales de control de sistemas deben recolectar evidencia suficiente, relevante, real y útil de las tareas y actividades que se realicen para alcanzar los objetivos de control. La evaluación del control será respaldada por el apropiado análisis e interpretación de dicha evidencia. Los profesionales de control de sistemas deben recolectar evidencia suficiente, relevante, real y útil de las tareas y actividades que se realicen para alcanzar los objetivos de control. La evaluación del control será respaldada por el apropiado análisis e interpretación de dicha evidencia.

  - ***560.030 Efectividad.***

Los profesionales de control de sistemas de información deben establecer mediciones apropiadas de efectividad en el desempeño de sus tareas para asegurar los objetivos de su rol y los objetivos definidos en el alcance. Los profesionales de control de sistemas de información deben establecer mediciones apropiadas de efectividad en el desempeño de sus tareas para asegurar los objetivos de su rol y los objetivos definidos en el alcance.

- ***570. Reporte***
  - ***570.010 Reporte periódico.***

Los profesionales de control de sistemas de información deben reportar periódicamente a un nivel apropiado de administración sobre los objetivos de control establecidos.

- ***580. Seguimiento de las actividades.***
  - ***580.010 Seguimiento.***

  Los profesionales de control de sistemas de información deben monitorear el funcionamiento de los procedimientos de control y revisar la efectividad y eficiencia de las actividades de control asegurando que sean tomadas las acciones correctivas que sean necesarias."

- ***Guías o lineamientos***: facilitan una guía para la aplicación de los estándares de auditoría de sistemas. El auditor de sistemas debería tenerlos en consideración al implementar los estándares pero no son de uso obligatorio.
- ***Procedimientos:*** brindan un conjunto de ejemplos de procedimientos que el auditor de sistemas puede utilizar durante el proceso de revisión de la información.

2.1.6.4 Otras normas, leyes, estándares y buenas prácticas.

Algunas de las normas, leyes, estándares y buenas prácticas utilizadas a nivel internacional en la auditoría de sistemas son las siguientes:

- ***Ley Sarvanes-Oxley (SOX)***[7]. Se trata de una ley de transparencia y control, emitida por el gobierno de los Estados Unidos de América, el 30 de julio del 2002, como resultado de una serie de escándalos corporativos que afectaron a ciertas empresas estadounidenses a finales del 2001, producto de quiebras, fraudes y otros manejos administrativos no apropiados.
- ***Informe COSO***[8]. El nombre COSO proviene del *"Committe of Sponsoring Organizations of the Treadway Commission"*, se trata de

[7] Ley SOX. www.soxlaw.com (Visitado el 27/08/2013)
[8] Informe COSO. www.coso.org (Visitado el 27/08/2013)

una iniciativa privada auspiciada por las más importantes asociaciones profesionales de los EEUU (Instituto Americano de Contadores Públicos, Instituto de Auditores internos, Asociación Americana de Contabilidad, Instituto de Contadores de Gestión y el Instituto de Ejecutivos Financieros).

- ***Informe COSO II***[6]: El *"Commitee of Sponsoring Organizations of the Treadway Commission"* determinó la necesidad de desarrollar un marco específico para la gestión de riesgos. En enero de 2001 se inició el proyecto que tenía por objetivo el desarrollar un marco global para gestionar los riesgos.
- ***ITIL (Information Technology Infraestructure Library)***[9]. Es un marco de trabajo (framework) para la administración de procesos de TI. Es un estándar de facto para servicios de TI. Fue desarrollado a fines de la década del 80. Originalmente este framework fue creado por la *"Central Computer and Telecommunications Agency"* que es una agencia del Gobierno del Reino Unido.
- ***Comité de Basilea***[10]. Fue creado a fines de 1974 por los representantes de los bancos centrales del G-10 (Bélgica, Canadá, Francia, Alemania, Italia, Japón, Holanda, Suiza, Suecia, Reino Unido y USA). El objetivo del Comité fue crear regulaciones bancarias y prácticas de supervisión. Su primer objetivo era regular los problemas en las monedas internacionales y mercados bancarios.
- **The Management of the Control of data Information Technology**[11]. Este modelo fue creado por el Instituto Canadiense de Contadores Certificados, determinando roles y responsabilidades vinculadas con la seguridad y los correspondientes controles relacionados con la tecnología de la información. Se definen siete grupos de roles: administración general, gerentes de sistemas, dueños, agentes, usuarios

[9] ITIL. http://www.itil-officialsite.com/Qualifications/ITILQualificationLevels/ITILFoundation.aspx (visitado el 27/08/2013)
[10] Comité de Basilea. http://www.bis.org/bcbs/ (visitado el 27/08/2013)
[11] The Management of the Control of data Information Technology http://www.fin.gov.bc.ca/ocg/fmb/manuals/CPM/12_Info_Mgmt_and_Info_Tech.htm (Visitado el 30/08/2013)

de sistemas de información, proveedores de servicios, desarrollo y operaciones de servicios y soporte de sistemas.

2.2 La minería de datos.

En esta sub-sección se desarrolla en el punto 2.2.1 la definición de la minería de datos, en el punto 2.2.2 se desarrollan los tipos de modelos utilizados en la minería de datos, en el punto 2.2.3 se explica el proceso de explotación de información, en el punto 2.2.4 se presentan las técnicas de minería de datos.

2.2.1 Introducción a la minería de datos.

Se define a la minería de datos (Schiefer et al., 2004; Clark, 2000) como el proceso de extracción de conocimiento no trivial que se encuentra de manera implícita en los sets de datos disponibles, considerando que estos datos pueden provenir de diferentes y variadas fuentes; este conocimiento es previamente desconocido y debe ser de utilidad. Es decir que la minería de datos plantea dos desafíos, por un lado trabajar con grandes bases de datos y por el otro aplicar técnicas que conviertan en forma automática estos datos en conocimiento.

La minería de datos es un elemento dentro de un proceso más amplio que tiene como objetivo el descubrimiento de conocimiento en grandes bases de datos (Fayyad et al., 1996), en inglés "*Knowledge Discovery in Databases*" (KDD), este proceso, como lo muestra la figura 2.1, tiene una primera etapa de preparación de datos, luego el proceso de minería de datos, la obtención de patrones de comportamiento, y la evaluación e interpretación de los patrones descubiertos.

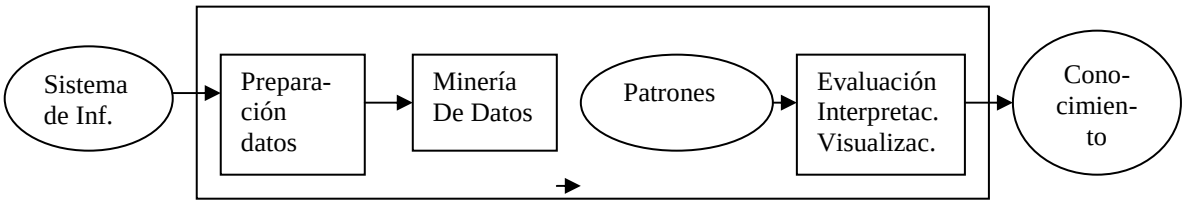


Figura 2.1. Proceso de KDD

2.2.2 Tipos de modelos de minería de datos.

Los modelos de minería de datos pueden ser de dos tipos, predictivos o descriptivos.

Dentro del modelo predictivo se encuentra la clasificación y la regresión. En el modelo descriptivo se encuentra el agrupamiento o clustering, las reglas de asociación, las reglas de asociación secuenciales.

- *Modelos predictivos*: Este tipo de modelo tiene como objetivo la estimación de valores desconocidos de variables de interés.

  Fundamentalmente en los modelos predictivos se utiliza la clasificación, donde dada una base de datos se indica mediante un valor en cada tupla a que clase pertenece, el objetivo es predecir a que clase pertenece una nueva instancia, considerando que los atributos pueden asumir valores discretos.

  También ser utiliza la regresión, en este caso el valor a predecir es numérico.

- *Modelos Descriptivos*: Estos modelos exploran las propiedades de los datos examinados con el objetivo de generar etiquetas o agrupaciones.

  Se utiliza el clustering, se trata de analizar datos para generar etiquetas.

  La correlación se utiliza para determinar el grado de similitud de los valores de dos variables numéricas.

  También se utilizan las reglas de asociación, que son similares a la correlación y tienen como objetivo encontrar relaciones no explicitas entre atributos, se aplican típicamente en el análisis del contenido de un carrito de compra.

Las reglas de asociación secuencial se utilizan para determinar los patrones secuenciales en los datos basados en el tiempo.

2.2.3 Proceso de explotación de información.

La minería de datos es considerada como una etapa dentro de un proceso de explotación de información (Larose, 2005), este proceso consta de las siguientes etapas (Britos et al., 2005):

- *Selección de datos (data selection).* Extracción de datos relevantes para obtener el objetivo planteado, estos datos pueden encontrarse en diferentes formatos y provenir de distintas fuentes.
- *Integración de datos (data integration).* El objetivo de esta etapa es crear una única fuente, para lograr esta meta se deben integrar todas las fuentes de datos.
- *Limpieza de datos (data cleaning).* Se deben identificar, procesar y de ser necesario eliminar de datos anómalos, erróneos, faltantes, ruidosos o sin relevancia.
- *Transformación de datos (data tranformation).* Los datos son transformados para optimizar el posterior proceso de minería de datos, algunas de las técnicas que generalmente se aplican son:
    - *Agregación,* con esta técnica se aplican funciones de resumen de los datos.
    - *Suavización,* el objetivo es eliminar el ruido de los datos.
    - *Construcción de atributos,* con el objetivo de mejorar el proceso de obtención de conocimiento se pueden crear nuevos atributos.
    - *Normalización,* con esta técnica se pretende crear categorías en determinados atributos y modificar el contenido de esas columnas en función de las categorías definidas.
- *Minería de datos (Data Mining).* Obtención de patrones de comportamiento de los datos en forma automática, que son previamente desconocidos y que son de utilidad.

- *Evaluación de patrones (Pattern Evaluation).* Identificación y análisis de patrones interesantes, esta tarea debe ser realizada por el experto en el dominio junto al especialista en minería de datos.
- *Presentación del conocimiento (Knowledge presentation).* Visualización y representación de los conocimientos obtenidos.

Existen varias metodologías para implementar procesos de explotación de información, las más difundidas son CRISP-DM, SEMMA y P3TQ:

- CRISP-DM (Chapman et al., 2000) se encuentra definida en función a un modelo jerárquico de procesos, donde se define un ciclo de vida de los proyectos de explotación de información cuyas fases son: entendimiento de negocios, entendimiento de los datos, preparación de los datos, modelado, evaluación y despliegue. Se trata de una metodología de uso público y es la más difundida y utilizada.
- SEMMA (Britos et al., 2008) toma su nombre de las etapas que esta metodología define para procesos de explotación de información, estas etapas son: muestreo (sample), exploración (explore), modificación (modify), modelado (model) y valoración (assess). La metodología fue desarrollada por SAS Institute Inc.[12], uno de las mayores organizaciones relacionadas con el desarrollo de software de inteligencia de negocios. SEMMA está desarrollada para aplicarla sobre la herramienta de minería de datos "SAS Enterprise Miner". Esta metodología es aplicada fundamentalmente por quienes usan esta específica herramienta software y su aplicación es limitada.
- P3TQ (Britos et al., 2008), su nombre proviene de los principales elementos que considera la metodología: producto (product), lugar (place), precio (price), tiempo (time) y cantidad (quantity). La metodología propone desarrollar el modelo de negocios y el modelo de explotación de información. Su aplicación aún es incipiente.

[12] SAS. http://www.sas.com/. (consultado el 08/08/2013)

2.2.4 Técnicas de la minería de datos.

Existen diferentes paradigmas detrás de las técnicas que se aplican en el proceso de minería de datos. La minería de datos utiliza técnicas basadas en el análisis estadístico y técnicas basadas en los sistemas inteligentes. En relación al análisis estadístico (García Martínez et al., 2004) algunas de las tecnologías más utilizadas son:

- *Análisis de agrupamiento:* en base a la característica común de una determinada cantidad de grupos, este tipo de análisis se utiliza para la clasificación de individuos.
- *Análisis de varianza:* es utilizado para la búsqueda de diferencias entre las medias de una cantidad determinada de variables de tipo continuas.
- *Análisis discriminante:* con este tipo de análisis es posible clasificar individuos en grupos establecidos previamente, para después encontrar la regla de clasificación para los elementos de cada grupo y por último identificar aquellas variables que definen la inclusión en un grupo.
- *Prueba de Chi-cuadrado:* Prueba que permite medir la probabilidad de que la frecuencia observada de una muestra sea debida sólo a la variación de dicha muestra.
- *Regresión:* es utilizada para definir relaciones entre un conjunto de variables y un conjunto asociado de otras variables que son utilizadas para la predicción de las primeras.
- *Series de tiempo:* utilizadas para el estudio de la evolución de una variable a lo largo del tiempo y para poder realizar predicciones.

Se pueden mencionar entre otras a las siguientes tecnologías basadas en sistemas inteligentes (García Martínez et al., 2003):

- *Algoritmos de Inducción*. TDIDT (Top Induction Decision Trees), donde a partir de ejemplos pre-clasificados es posible clasificar nuevos ejemplos.
- *Perceptron Multicapa*. Es un algoritmo basado en redes de neuronas artificiales de backpropagation.

- *Maquinas de vector soporte*. Se trata de un conjunto de algoritmos de aprendizaje supervisado.
- *Algoritmos a priori.* Que se basa en el conocimiento previo de los datos.
- *Redes neuronales SOM*. Conocidas como mapa auto organizado de Kohonen.
- *Algoritmos genéticos*. Que se basan en la evolución biológica.
- *Redes bayesianas*. Mediante la estimación de las probabilidades que utilizan el teorema de Bayes.
- *Algoritmo del vecino más próximo*. Utilizado en la solución del problema del viajante.
- Existen otras técnicas como las estocásticas, relacionales, declarativas, difusas, etc. y una variedad de técnicas hibridas.

La tabla 2.1 muestra la relación entre los modelos y las técnicas relacionadas con la minería de datos.

Existe una importante cantidad de productos software que posibilitan el proceso de minería de datos, algunos de los más conocidos son:

- CART[13] de la empresa Salford Systems.
- Clementine[14] de la empresa SPSS/Integral Solutions Limited (ISL).
- Darwin[15] de la empresa Oracle.
- Data Surveyor[16] de la empresa Data Distilleries.
- Enterprise Miner[17] de la empresa SAS.
- GainSmarts[18] de la empresa Urban Science.
- Intelligent Miner[19] de la empresa IBM.
- Knowledge Seeker[20] de la empresa Angoss.
- Microstrategy[21] de la empresa Microstrategy.
- Polyanalyst[22] de la empresa Megaputer.

[13] CART. Disponible en www.salford-systems.com/ (consultado el 08/08/2013)
[14] Clementine. Disponible en www.spss.com/clementine/ (Consultado el 08/08/2013)
[15] Darwin. Disponible en www.oracle.com/technology/documentation/darwin.html (Consultado el 07/07/2008)
[16] Data Surveyor. Disponible en http://www.ndparking.com/ddi.nl (Consultado el 08/08/2013)
[17] Enterprise Miner. Disponible en www.sas.com/technologies/analytics/datamining/miner/ (Consultado el 08/08/2013)
[18] GainSmarts. Disponible en www.urbanscience.com/GainSmarts.html (Consultado el 08/08/2013)
[19] Intelligent Miner. Disponible en www-306.ibm.com/software/data/iminer/ (Consultado el 08/08/2013)
[20] Knowledge Seeker. Disponible en www.angoss.com/ (Consultado el 08/08/2013)
[21] Microstrategy. Disponible en www.microstrategy.com/ (Consultado el 08/08/2013)
[22] Polyanalyst. Disponible en www.megaputer.com/polyanalyst.php (Consultado el 08/08/2013)

- Rapid Miner[23] Gratuito basado en la filosofía open source.
- SGI MineSet[24] de la empresa Silicon Graphics.
- WEKA[25] Gratuito basado en la filosofía open source.
- Wizsoft/Wizwhy[26] de la empresa Wizsoft.
- Pattern Recognition Workbench (PRW)[27].
- Orange[28]. Gratuito basado en la filosofía open source.
- R[29]. Gratuito basado en la filosofía open source.

Técnicas	Predictivo		Descriptivo		
	Clasificación	Regresión	Agrupamiento	Reglas de asociación	Correlaciones / Factorizaciones
Redes neuronales	■	■	■		
Árboles de decisión ID3, C4.5, C5.0	■				
Árboles de decisión CART	■	■			
Otros árboles de decisión	■	■	■	■	
Redes de Kohonen			■		
Regresión lineal		■			■
Regresión logística	■			■	
Kmeans			■		
A priori				■	
Naive Bayes	■				
Vecinos más próximos	■	■	■		
Algoritmos genéticos y evolutivos	■	■	■	■	■
Máquinas de vectores soporte	■	■	■		
Análisis discriminante multivalente	■				

Tabla 2.1. Relación entre técnicas y métodos de minería de datos

[23] Rapid Miner. Disponible en www.rapidminer.com/ (Consultado el 08/08/2013)
[24] SGI MineSet . Disponible en http://www.sgi.com/ (Consultado el 08/08/2013)
[25] WEKA. Disponible en www.weka.net.nz/ . (Consultado el 08/08/2013)
[26] Wizsoft/Wizwhy. Disponible en http://www.wizsoft.com/ (Consultado el 08/08/2013)
[27] Pattern Recognotiom Workbench (PRW). Disponible en http://www.unica.com/ (Consultado el 08/08/2013)
[28] Orange. Disponible en http://orange.biolab.si/ (consultado 08/08/2013)
[29] R. Disponible en http://www.r-project.org/ (consultado el 08/08/2013)

Cada vez son más los productos basados en la filosofía open source que existen en el mercado, esto ha permitido disminuir enormemente los costos de implementación de la minería de datos, en particular en las pequeñas y medianas empresas.

2.3 Minería de datos y auditoría de sistemas.

El mayor desarrollo en la aplicación de la minería de datos en el proceso de una auditoría de sistemas se relaciona con la detección de intrusos en redes de telecomunicaciones.

En las siguientes sub-secciones se presentan las principales aplicaciones relacionadas con el uso de la minería de datos en el proceso de auditoría de sistemas y los algoritmos que se utilizan, en el punto 2.3.1 se explica el concepto de auditoría continua y su relación con la minería de datos, en 2.3.2 se desarrolla las aplicaciones de la minería de datos en la detección de fraudes en el área de finanzas y contabilidad, en el punto 2.3.3 se explican las aplicaciones de la minería de datos en la detección de intrusos en redes de telecomunicaciones y en el punto 2.3.4 se presentan las aplicaciones de minería de datos en la detección de terroristas.

2.3.1 Auditoria continua y minería de datos.

Las opiniones expost que brindan las tradicionales auditorias anuales se relacionan fundamentalmente con la era pre-digital. Fundamentalmente el comercio electrónico entre otros factores ha modificado profundamente el concepto de la auditoría tradicional (Rezaee et al., 2002). El concepto de auditoría continua se relaciona con poder evaluar la calidad, legalidad y seguridad de la información de manera simultánea a la ocurrencia de los procesos que son controlados. La evaluación en tiempo real de la información procesada agrega enorme calidad en una auditoría de sistemas y en algunos casos este tipo de auditoría puede remplazar los tradicionales informes anuales.

En el proceso de auditoría continua el auditor puede acceder, evaluar y controlar en tiempo real los de datos de las transacciones procesadas por la empresa. El objetivo fundamental de la auditoría continua es disminuir el tiempo de latencia entre el momento en que se efectúan las operaciones y el control posterior que realiza la auditoría. Este tipo de proceso se vincula en forma directa con el uso de las nuevas tecnologías de la información y comunicación en la auditoría.

La auditoría continua tiene tres actividades principales:

- Planificación
- Dirección de la auditoría
- Generación de informes

La aparición del XML (eXtensible Markup Language) y XBRL (eXtensible Business Reporting Language) ha facilitado el proceso de auditoría continua (Garrity et al., 2006) ya que estas tecnologías minimizan los riesgos del uso de las CAATs, al no impactar en forma directa el análisis que se realiza sobre los datos en las bases de datos de los sistemas en producción.

Es necesario realizar un análisis de riesgo para iniciar un proceso de auditoría continua, de manera de identificar aquellos activos que tienen más posibilidades de sufrir algún daño. La minería de datos puede ser usada de manera eficiente en la identificación de transacciones erróneas, la posibilidad de realizar búsquedas no paramétricas a través de las redes neuronales o los árboles de decisión que han potenciado sustancialmente su uso.

Para aplicar la minería de datos en el proceso de auditoría continua se deben evaluar los siguientes elementos del algoritmo a aplicar:

- Escalabilidad, es la manera en que se comporta el algoritmo en grandes bases de datos.
- Exactitud, es la medida en que la información obtenida como resultado de aplicar el algoritmo permanece estable y constante más allá de las muestras obtenidas.
- Robustez, se espera que el algoritmo se comporte de igual manera en una variedad de dominios.
- Interpretación, el algoritmo debe proveer información comprensible.

Es muy difícil que un algoritmo de minería de datos se destaque en todos los puntos descriptos, no ha sido suficiente la experimentación realizada en el uso de la minería de datos en grandes bases de datos, siendo estos dos puntos la debilidad fundamental detectada en la identificación de anomalías en bases de datos.

La auditoría continua necesita contar con la tecnología adecuada para extraer y analizar datos de distintas fuentes y plataformas en tiempo real, existen tres alternativas tecnológicas para implementar la auditoría continua, a través de:

- Módulos de auditoría embebidos
- Uso del Datawarehousing para detectar patrones inusuales en los almacenes de datos.
- Uso de agentes inteligentes para obtener patrones en forma automática.

A continuación en la figura 2.2 se muestra en forma esquemática la relación entre la auditoría continua y la minería de datos. La utilización de XML y XBRL permiten aislar el proceso de producción de los sistemas informáticos del proceso de auditoría, garantizando de esta manera la integridad de la información con la que operan los sistemas y potenciando el uso de la CAATs. Es importante destacar como paso previo al uso de agentes inteligentes, la generación de *Data Marts* (Inmon, 1996)[30], que permiten por una lado la búsqueda "manual" de patrones, y por el otro son la entrada de los agentes inteligentes que permiten la búsqueda automática de valores anómalos en los datos.

Si bien aún es incipiente la implementación de procesos relacionados con la auditoría continua, el uso de XML brinda el soporte tecnológico para evitar los riesgos inherentes propios del análisis de información en tiempo real, mejorando en forma sustancial el proceso de auditoría continua. Siendo mucho el camino a recorrer, como por ejemplo, en la auditoría continua a la WEB, a la información textual, a las grandes bases de datos vinculadas al concepto de big data.

[30] Un Data Mart es una versión particular de un Datawarehousing, se trata de un subconjunto específico de datos orientado a un tema en particular.

La auditoría continua tiene entre otras ventajas la detección en tiempo real de posibles datos anómalos, requiriendo esta detección específicos algoritmos y procedimientos.

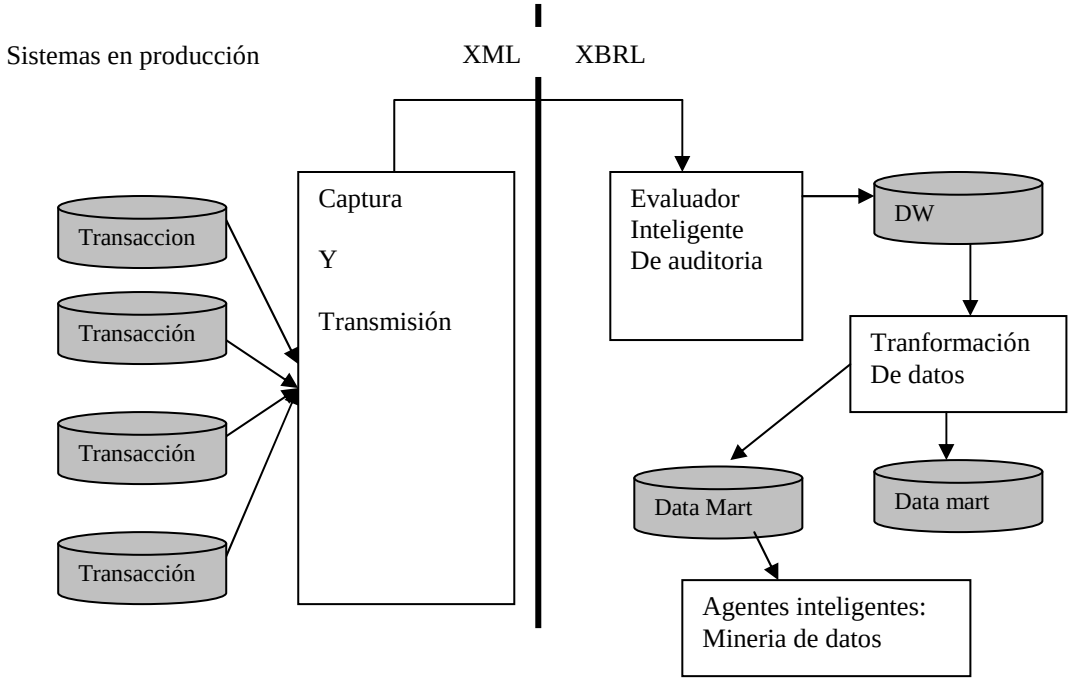


Figura 2.2. Relación Auditoría Continua y Minería de datos

2.3.2 Minería de datos para la detección de fraudes en el área de finanzas y contabilidad.

La aplicación de la minería de datos en el ámbito de las finanzas está reconocida por el "*American Institute of Chartered Public Accountants*"[31]. El *"Institute of Internal Auditors"*[32] toma como una de sus prioridades a la minería de datos (Koh & Low, 2004).

La detección de datos anómalos en bases de datos relacionadas con el área de finanzas y contabilidad brindan pistas de auditoría que pueden implicar algún tipo de fraude, por lo tanto su detección y evaluación temprana son de fundamental importancia.

[31] http://www.aicpa.org/Pages/default.aspx (Visitado el 22/09/2013)
[32] https://na.theiia.org/Pages/IIAHome.aspx (Visitado el 22/09/2013

En las siguientes revistas científicas se encuentra información sobre la relación entre minería de datos y el área de finanzas y contabilidad:

- Decision Support Systems[33].
- European Journal of Operational Research[34].
- Expert Systems with Applications[35].
- Intelligent Systems in Accounting, Finance & Management[36].
- International Journal of Accounting Information Systems[37].
- Journal of Forecasting[38].
- Knowledge Based Systems[39].
- Management Decision[40].
- Managerial Auditing Journal[41].
- Managerial Finance[42].
- Neural Networks[43].
- The International Journal of Management Science[44].

En el campo de la auditoría, los métodos y técnicas de minería de datos se desarrollan como una contribución prometedora. Los recientes eventos demuestran los problemas considerables en el proceso de auditoría. Los fracasos de Enron y Arthur Andersen[45] entre otros demuestran los problemas en detectar fraudes en general y en particular los perpetrados por los propios directivos de las compañías.

Existen normas internacionales que determinan la necesidad de controlar los datos grabados, como por ejemplo, la norma SAS 56 (Statement of Auditing Standards)[46], en el año 1997 *Auditing Standard Board* publicó la norma SAS 82 (Holm, 2007) que se relaciona con los fraudes en los estados financieros, donde se requiere a los auditores evalúen los riesgos relacionados con este

[33] http://www.journals.elsevier.com/decision-support-systems/ (Visitado el 02/09/2013)
[34] http://www.journals.elsevier.com/european-journal-of-operational-research/ (Visitado el 02/09/2013)
[35] http://www.journals.elsevier.com/expert-systems-with-applications/ (Visitado el 02/09/2013)
[36] http://onlinelibrary.wiley.com/journal/10.1002/(ISSN)1099-1174 (Visitado el 02/09/2013)
[37] http://www.journals.elsevier.com/international-journal-of-accounting-information-systems/ (Visitado el 02/09/2013)
[38] http://onlinelibrary.wiley.com/journal/10.1002/(ISSN)1099-131X/ (Visitado el 02/09/2013)
[39] http://www.elsevier.com/wps/product/cws_home/525448/description (Visitado el 02/09/2013)
[40] http://www.emeraldinsight.com/journals.htm?issn=0025-1747 (Visitado el 02/09/2013)
[41] http://www.emeraldinsight.com/journals.htm?issn=0268-6902 (Visitado el 02/09/2013)
[42] http://www.emeraldinsight.com/journals.htm?issn=0307-4358 (Visitado el 02/09/2013)
[43] http://www.elsevier.com/wps/product/cws_home/841/description (Visitado el 02/09/2013)
[44] http://www.omegajournal.org/ (Visitado el 02/09/2013)
[45] http://www.nytimes.com/2002/01/16/business/enron-s-collapse-overview-arthur-andersen-fires-executive-for-enron-orders.html?pagewanted=all&src=pm (Visitado el 22/09/2013)
[46] http://www.aicpa.org/Research/Standards/AuditAttest/Pages/SAS.aspx (Visitado el 02/09/2013)

tipo de fraudes, esta tarea es muy compleja para realizarla en forma manual, no sólo por la subjetividad sino por el volumen. Existen otras leyes de aplicación en muchos casos global, tal vez la más conocida sea la ley promulgada por el gobierno de Estados Unidos conocida como Sarvanes Oxley[47]entre otras (ver sección 2.1.6.4), en el caso de esta ley, conocida como SOX, es de aplicación obligatoria para todas las empresas estadounidenses que tienen sucursales o subsidiarias en otros países.

Algunas de las aplicaciones de la minería de datos en el ámbito de las finanzas fueron desarrolladas por Kirkos & Manolopoulos, (2004), quienes realizan una breve descripción bibliográfica de este tipo de aplicaciones:

- *Predicción de quiebras*, este es el uso más difundido. Se han utilizado técnicas de discriminante múltiple y técnicas estadísticas.

  En el año 2001 (Lin & McClean, 2001) se desarrollaron pronósticos relacionados con quiebras, donde se utilizaron cuatro modelos (dos estadísticos y dos de aprendizaje automático), los mejores resultados se obtuvieron con las redes neuronales, a partir de estos resultados se desarrolló un algoritmo híbrido con el objetivo de mejorar los mismos.

  En el año 2002 (Shin & Lee, 2002) se desarrolló un modelo basado en algoritmos genéticos, evaluándose que este tipo de modelos produce reglas más entendibles que los modelos basados en las redes neuronales.

  En el año 2003 (Kim & Han, 2003) desarrolló un modelo cualitativo basado en algoritmos genéticos.

  En el año 2004, (Tung et al., 2004) se desarrolla un modelo híbrido utilizando las redes neuronales y los algoritmos genéticos, considerándose una de las principales ventajas de este modelo el uso de modelos lingüísticos.

- *Empresa en marcha y penuria financiera*, la norma SAS 59[48] requiere al auditor que evalúe si existe duda sobre la capacidad de la entidad de mantener el negocio en marcha por lo menos un año después de

[47] Ley SOX. Ley SOX. www.soxlaw.com (Visitado el 27/08/2013)
[48] http://www.aicpa.org/Research/Standards/AuditAttest/Pages/SAS.aspx#SAS43 (visitado el 22/09/2013)

la fecha de los estados financieros auditados, en esta línea se han desarrollado los siguientes trabajos relacionados con el uso de la minería de datos para conseguir este objetivo:

En el año 2001 (Tan & Dihardjo, 2001) aplico las redes neuronales con el objetivo de detectar problemas financieros en empresas, mejorando el método al incorporar el concepto de "detector temprano".

En el año 2004 (Koh & Low, 2004) propuso el uso de métodos estadísticos y de aprendizaje automático, en su trabajo realizó comparaciones entre modelos basados en redes neuronales, árboles de decisión y regresión lógica, siendo los árboles de decisión quienes brindaron los mejores resultados.

- *Dirección fraudulenta*, se trata de directivos de empresas que realizan actividades maliciosas en forma intencional.

  En el año 2002 (Spathis, 2002) desarrolló dos modelos con el objetivo de detectar una dirección fraudulenta en una empresa, el método utilizado fue la regresión lógica.

- *Pronósticos de rendimiento cooperativo,* las líneas de trabajo encontradas fueron:

  En el año 2001 (Black et al., 2001) se desarrollaron dos modelos para agrupar compañías de acuerdo a su rendimiento.

  En el año 2003 (Lam, 2003) se aplicaron las redes neuronales y el algoritmo de Glare (Spencer et al., 1995).

En relación con la detección de fraudes en el área de finanzas y contabilidad se destaca el futuro de los modelos híbridos, estos modelos combinan algoritmos de distintos tipos. Existen muchas líneas de investigación abiertas, como por ejemplo, la interpretación de los patrones de la toma de decisiones en las redes neuronales; la comparación de los modelos basados en la inteligencia artificial con los modelos estadísticos; el perfeccionamiento de los métodos de visualización de manera de hacer más comprensible la información obtenida; la posibilidad de embeber en los ERP las herramientas de minería de datos con el objetivo de potenciar su uso; la incorporación de

datos macroeconómicos, información cualitativa como por ejemplo, la opinión de los auditores en informes anteriores al proceso de minería de datos; la optimización de las técnicas aplicadas a grandes bases de datos.

Diversos autores (Ngai et al., 2011) han especificado una clasificación de los fraudes financieros, los mismos pueden ser:

- Fraudes bancarios, donde se incluyen fraudes con tarjetas de crédito, blanqueo de dinero y el fraude hipotecario.
- Fraude relacionado con los seguros, este tipo de fraudes puede producirse en muchos puntos del proceso, particularmente en los reclamos.
- Otros tipos de fraude, esta categoría incluye los fraudes de tipo financiero no incluidos en las dos categorías anteriores, como los fraudes corporativos (falsificación de información financiera, tratamiento de la información corporativa, y obstrucción de la justicia) o los fraudes relacionados con el marketing masivo, por ejemplo telemarketing, mails masivos, internet, etc.

El fraude bancario relacionado con las tarjetas de crédito y los fraudes con seguros de automotores (Foster & Stine, 2004) son los tipos de delitos que han recibido la mayor atención por parte de los investigadores. La tabla 2.2 (Ngai et al., 2011) muestra una clasificación de artículos relacionados con fraudes financieros reales publicados entre el año 2000 y 2008 en las siguientes bases de datos:

- ABI/INFORM Database
- Academic Search Premier
- ACM
- Business Source Premier
- Emerald Full text
- IEEE Transactions
- Science Direct
- Springer-Link Journals
- World Scientific Net

El bajo número de publicaciones posiblemente se deba a la dificultad por parte de los investigadores para obtener los datos ya que la información requiere confidencialidad al tratarse de datos sensibles, esta confidencialidad es uno de los factores que mayor impacto tienen a la hora de obtener bases de datos reales para poder evaluar diferentes procedimientos en la búsqueda de valores anómalos.

En el punto 2.3.2.1 se presentan las herramientas de minería de datos utilizadas en el área de finanzas, en el punto 2.3.2.2 se hace una descripción de los algoritmos usados en la detección de fraudes provocados por la alta gerencia.

Categoría	Actividad	2000	2001	2002	2003	2004	2005	2006	2007	2008	total
Fraude bancario	Fraude con tarjetas de crédito							2		3	5
	Lavado de dinero								1		
Fraude en seguros	Fraudes en seguros para cosechas						1	1			2
	Fraudes en seguros de salud		1	1		1	1				4
	Fraudes en seguros de automóviles	1		5		1	3		2	1	13
Otros tipos de fraude	Fraudes corporativos	1		3	1			1	1	3	10
TOTAL		2	1	9	1	2	5	4	4	7	35

Tabla 2.2. Publicaciones de artículos relacionados con fraudes reales. (Ngai et al., 2011)

2.3.2.1 Herramientas DM utilizadas en el área de finanzas y contabilidad.

Se han realizado evaluaciones de herramientas de minería de datos para la detección de fraudes uno de los más interesantes fue realizado por Dean Abbott (Abbott et al., 1998), a continuación se presenta un resumen de este trabajo.

Las propiedades evaluadas fueron:

- Cumplimiento de la arquitectura Cliente Servidor.

- Documentación.
- Capacidad de los algoritmos puestos en práctica.
- Facilidad de uso.
- Exactitud sobre los datos de prueba en la detección de fraudes.

Finalmente se presentan las conclusiones del trabajo de Dean Abbott.

La tabla 2.3 muestra las herramientas que fueron comparadas.

Empresa	Producto	Versión
Integral Solutions, Ltd. (ISL)	Clementine	4.0
Thinking Machines (TMC)	Darwin	3.0.1.
IBM	Intelligent Miner for Data (IM)	2.0.
SAS Institute	Enterprise Miner (EM)	Beta
Unica Technologies, Inc.	Pattern Recognition Workbench (PRW)	2.5

Tabla 2.3. Herramientas testeadas. (Abbott et al., 1998)

- *Proceso Cliente Servidor.*

Contar con herramientas que incorporen la arquitectura cliente servidor ayuda a mejorar el rendimiento cuando es necesario procesar grandes volúmenes de información. La tabla 2.4 muestra el hardware y software utilizado en las pruebas realizadas por Dean Abbott (Abbott et al., 1998), considerando que se trabajó con una base de datos Oracle.

El resultado de las pruebas fue que:

- *Darwin* implementa mejor el paradigma cliente servidor.
- La performance de *Clementine* sobre un MODEM fue muy lenta.
- La prueba sobre *PRW* fue sobre una única estación de trabajo, con una muy buena performance.
- *Intelligent Miner* utiliza un cliente Java, corriendo más lentamente que otro GUI, aunque no fue muy significativa esta característica.
- *Enterprise Miner* fue probado sobre Windows NT requiriendo un hardware más potente que otras herramientas.

Producto	Empresa	Servidor	Cliente
Clementine	Integral Solutions, Ltd. (ISL)	Solaris 2.X	X Windows
Darwin	Thinking Machines (TMC)	Solaris 2.X	Windows NT
Enterprise Miner	IBM	Solaris 2.X	Windows NT
Intelligent Miner	SAS Institute	IBM AIX	Windows NT
PRW	Unica Technologies, Inc.	Data only	Windows NT

Tabla 2.4. Hardware y software utilizado en las pruebas. (Abbott et al., 1998)

- *Documentación.*

La documentación que generan las herramientas software es importante ya que posibilita contar con información que facilite el mantenimiento posterior de los productos que genera, las cinco herramientas testeadas brindaron documentación suficiente.

- *Algoritmos utilizados en la prueba*

La tabla 2.5 muestra las técnicas que implementa cada una de las herramientas.

Técnicas	IBM	ISL	SAS	TMC	UNICA
Arboles de decisión	X	X	X	X	
Redes Neuronales	X	X	X	X	X
Regresión	X	X	X		X
Funciones de base radiales	X				X
Vecino más cercano			X	X	X
Redes autoorganizadas de Kohonen		X	X		X
Clustering	X	X			X
Reglas de asociación	X	X			

Tabla 2.5. Algoritmos de cada herramienta. (Abbott et al., 1998)

- *Facilidad de uso.*

Para realizar esta prueba se definieron cuatro categorías, cada una de ellas con varias sub-categorías, varios usuarios hicieron las pruebas calculándose el promedio, siendo 5 el valor máximo. En la tabla 2.6 se encuentra el resultado obtenido.

	IBM	ISL	SAS	TMC	UNICA
Carga de datos y manipulación	3.1	3.7	3.7	3.1	3.9
Construcción del modelo	3.1	4.6	3.9	3.2	4.8
Entendimiento del modelo	3.2	4.2	2.6	3.8	3.8
Apoyo técnico	3.0	4.0	2.8	3.2	4.7
Resultado	3.1	4.1	3.1	3.4	4.2

Tabla 2. 6. Resultados de la comparación entre categorías y productos software de minería de datos. (Abbott et al., 1998)

- *Exactitud.*

El objetivo planteado en el trabajo de Dean Abbott (Abbott et al., 1998) fue evaluar la exactitud de las herramientas en encontrar transacciones fraudulentas sobre un conjunto de transacciones sin incurrir en falsos positivos.

Se utilizaron más de veinte modelos, las redes neuronales y los arboles de decisión dieron los mejores resultados, siendo los últimos los que mostraron mayor exactitud. *Clementine* mostró mejores resultados utilizando árboles de decisión en la detección de transacciones fraudulentas y *PRW* el mejor resultado utilizando redes neuronales.

- *Resultados obtenidos por Dean Abbott.*

Existe una enorme dependencia del entorno donde se deben aplicar las herramientas software de minería de datos para determinar el mejor producto, es muy difícil afirmar apriorísticamente que producto de minería de datos es el mejor para determinado entorno y set de datos, en el caso particular del estudio analizado se observa que en general los cinco productos mostraron buenos resultados, *Intelligent Miner* es el líder del mercado mundial, *Clementine* se destaca por el soporte y la facilidad de uso, *Enterprise Miner* es una herramienta especial para aquellos usuarios familiarizados con SAS, *TMC* mostró muy buena performance en grandes bases de datos, *Unica* es una muy buena herramienta cuando el algoritmo a utilizar no es obvio. Se debe destacar que los cinco productos analizados son comerciales. De todo este análisis se desprende que la utilización de determinada herramienta estará ligada al entorno en el que el auditor debe realizar su tarea y de los controles que se deben realizar.

2.3.2.2 Algoritmos para la detección de fraudes producidos por la alta gerencia.

En los últimos años los reclamos de fraude han aumentado considerablemente (Koskivaara, 2004). Desde el punto de vista económico, el fraude financiero cometido por la alta gerencia se está convirtiendo en un grave problema. Por ejemplo el fraude realizado por el ex presidente de Nasdaq[49], Bernard Madoff, que provocó una pérdida de aproximadamente U$S 50.000.000.000.- Otro ejemplo es el de José Hirko, el ex co-presidente ejecutivo de Enron Broadband Servicios (EBS), que ha declarado que la compañía debió restituir U$S 8.700.000.- a las víctimas de Enron a través de la Superintendencia de Valores EE.UU. y el Fondo Regular de Enron de la Comisión de Bolsa tras declararse culpable del fraude electrónico[49]. Otro ejemplo es el presentado en un informe de noticias de la BBC del año 2007[50], donde reclamos vinculados con actividades fraudulentas relacionadas con seguros tuvieron un costo para las aseguradoras del Reino Unido de un total de 1.600.000 libras al año. Las pérdidas causadas por fraudes internos son realmente incalculables, ya que en muchos casos las compañías evitan dar publicidad a este tipo de incidentes para evitar problemas relacionados con su imagen pública.

Se trata de fraudes cometidos por la alta gerencia, siendo en muchos casos muy compleja su detección, algunos de estos fraudes producto del proceso de globalización han conmocionado la economía mundial. Los riesgos de los fraudes cometidos por la alta gerencia son sustancialmente mayores que otro tipo de fraudes ya que en general las empresas no están preparadas para prevenirlos.

Las CAATs son de fundamental importancia en la detección temprana de fraudes ya que evitan analizar cada una de las transacciones en manera manual por parte de los auditores, realizar este análisis sin la ayuda de la CAATs en muchos casos, dado el enorme volumen de información, puede

[49] http://newyork.fbi.gov/dojpressrel/pressrel08/nyfo121108.htm (Visitado el 02/09/2013)
[50] http://news.bbc.co.uk/1/hi/business/6636005.stm (Visitado el 02/09/2013)

convertirse en una tarea imposible de concretar. La minería de datos tiene una ventaja teórica sobre las técnicas manuales en la búsqueda de evidencias al evitar la subjetividad del análisis y al optimizar de manera sustancial los tiempos requeridos para realizar las pruebas sustantivas.

Se desarrollaron modelos (Green & Choi, 1997) con redes neuronales que realizan una clasificación de este tipo de fraudes.

En (Eining et al., 1997) se presenta una investigación para determinar si el uso de los sistemas expertos había posibilitado una mejora en el trabajo de los auditores, determinando que el uso de este tipo de sistemas permitió a los auditores discriminar mejor entre situaciones con diferentes niveles de gestión de riesgo de fraude.

En (Fanning & Cogger, 1998) se utilizó una red neuronal para desarrollar un modelo de detección de fraudes internos. El vector de entrada estaba formado por ratios financieros y variables cualitativas, demostrando que su modelo era más eficaz en la detección de fraudes que los métodos estadísticos tradicionales.

En (Abbot et al., 2000) se utilizó la regresión estadística para evaluar si la existencia de un comité de auditoría independiente en las empresas servía para disminuir los riesgos de fraude interno. El resultado fue que aquellas empresas con comités de auditoría independientes que se reúnen por lo menos dos veces al año, tienen menos probabilidades de sufrir consecuencias relacionadas con la información fraudulenta.

En (Bell & Carcello, 2000) se desarrolló y testeó un modelo basado en técnicas de regresión logística para estimar la probabilidad de fraude relacionado con información financiera de un cliente de auditoría, este modelo estaba condicionado a la presencia o ausencia de varios factores de riesgo de fraude.

En (Spathis, 2002) se construyó un modelo basado en la regresión logística para detectar falsificación de estados financieros. Los resultados sugieren que existe un importante potencial en la implementación algoritmos relacionados con la minería de datos y la estadística en la detección de fraudes a través del análisis de los estados financieros publicados.

Existe una variedad de trabajos relacionados con la comparación de técnicas para la detección de fraudes. En (Kirkos et al., 2007) se desarrolló un trabajo donde se comparan los árboles de decisión, redes neuronales y redes bayesianas, los datos de esta comparación provienen de 76 compañías griegas relacionadas con la producción industrial. La identificación de fraudes relacionados con los estados financieros de las empresas es un típico problema de clasificación, donde se aplica un procedimiento que tiene dos etapas, en la primera se entrena al modelo a través de ejemplos, donde uno de los atributos es el atributo "clase" o "target" que contiene los valores de las clases posibles, en el segundo paso se valida el modelo intentando clasificar el set de datos no utilizado en el entrenamiento.

La tabla 2.7 muestra los resultados de las pruebas (Kirkos et al., 2007) realizadas en la etapa de entrenamiento del modelo, siendo las redes neuronales las más eficientes en discriminar compañías que sufrieron fraudes de las que no lo sufrieron, con los modelos sin entrenar.

Modelo	Fraudes	No fraudes	**Total**
ID3 [51]	92.1%	100.0%	**96.2%**
NN	100.0%	100.0%	**100.0%**
BBN	**97.4%**	**92.1%**	**94.7%**

Tabla 2.7. Resultados con los modelos sin entrenar. (Kirkos et al., 2007)

En la etapa de validación se produjeron algunas variaciones en los resultados que son expuestos en la tabla 2.8, las redes bayesianas fueron las que mostraron los mejores resultados en este caso. Como era previsible la precisión es menor en la etapa de validación que en la de entrenamiento, en el caso del algoritmo ID3 se produjo una disminución considerable en la etapa de validación ya que en el entrenamiento clasificó correctamente al 96% de los casos y en la validación al 73,60%, en el caso de la red neuronal se pasa de un 100% de acierto en el entrenamiento a un 80% en la validación, las redes bayesianas resultaron tener los mejores resultados en la validación con un 90,30% disminuyendo solamente un 4,40% desde la etapa de entrenamiento.

[51] ID3 es un algoritmo que se utiliza en la búsqueda de reglas o hipótesis dado un conjunto de ejemplos, para ello utiliza árboles de decisión, los elementos son, nodos, arcos y hojas.

Modelo	Fraudes	No fraudes	**Total**
ID3	75.0%	72.5%	**73.6%**
NN	82.5%	77.5%	**80.0%**
BBN	**91.7%**	**88.9%**	**90.3%**

Tabla 2.8. Resultados con los modelos entrenados. (Kirkos et al., 2007)

2.3.3 Minería de Datos para la detección de intrusos en redes de telecomunicaciones.

Una de las áreas de mayor desarrollo en lo relacionado con el uso de la minería de datos es la detección de intrusos en las redes de telecomunicaciones. Internet se ha convertido en una parte fundamental de la vida de las personas, siendo un soporte ineludible en muchas áreas como los negocios, la educación, el gobierno, etc., modificando de manera sustancial la manera en que la sociedad se comunica.

La seguridad en las telecomunicaciones se ha convertido en un tema central. Un intruso en una red de telecomunicaciones puede tener dos consecuencias con un impacto enorme, estas consecuencias son: el impedimento del flujo normal del tráfico de datos en la red o el robo de información. Algunos de los principales actos producidos por el accionar de los intrusos son (Chauhan et al., 2012):

- *Ataque de denegación de servicios (DoS)*[52], se trata de un ataque a un ordenador o a una red de ordenadores que intenta que un recurso sea inaccesible para los usuarios. En general provoca la pérdida o dificultad del funcionamiento de la red por el alto consumo del ancho de banda de la misma.
- *Ataques remotos de usuarios (R2L)*[53], se trata del acceso no autorizado desde una máquina remota a una cuenta de usuario (root) del sistema de destino. Explotando la vulnerabilidad de la maquina el atacante envía paquetes a través de una red para obtener ilegalmente el acceso local.

[52] Denial of Service
[53] Remote to Users (R2L)

- *Ataque usuario root (U2R)*[54], se trata del acceso a una cuenta de super-usuario local (root). El atacante accede a una cuenta normal de usuario y explotando las vulnerabilidades del servidor accede como root al sistema.
- *Sondeo (Proving)*, es el caso en que un atacante examina una red para obtener información o encontrar vulnerabilidades. El atacante busca vulnerabilidades a partir de los servicios que están activado en el servidor, en algunos casos accediendo en forma legal a un ordenador y en otras utilizando técnicas de ingeniería social.

En general las estadísticas demuestran que la mayoría de los ataques a los sistemas se producen desde dentro de las organizaciones, muchos de estos casos son producidos por ladrones de cuello blanco, basta recordar el caso de Enron producido el año 2001[55].

En general los sistemas dejan pistas de auditora embebidas en las bases de datos o diseñadas específicamente en cada sistema, algunos de los datos que poseen estos logs son el operador, la fecha, la operación, el numero de IP, la hora, el valor anterior del atributo, el nuevo valor, etc., existe mucha bibliografía relacionada con este tipo de aplicación (Van der Aalst & Van Dongen, 2002; Van der Aalst & Song, 2004; Van der Aalst & Medeiros, 2005; Agrawal et al., 1998).

La minería de datos es muy útil en este tipo de análisis ya que permite encontrar en forma automática los patrones de comportamiento de los procesos y de los operadores, para después poder comparar como se ejecutaron los procesos en la realidad con la forma en que teóricamente debían hacerlo.

Para enfrentar los ataques a las redes de telecomunicaciones se han desarrollado Sistemas de Detección de Intrusos (IDS), que están destinados a fortalecer los sistemas de comunicación e información (García-Teodoro et al., 2009; Zhang & Lee, 2000; Roesch, 1999; Hofmeyr et al., 1998; Warrender et al., 1999; Portnoy et al., 2001). Existen dos trabajos relevantes relacionados

[54] User to Root (U2R)
[55] http://news.bbc.co.uk/hi/spanish/news/newsid_1803000/1803224.stm (Visitado el 04/09/2013)

con el desarrollo de Sistemas de Detección de Intrusos (Denning, 1987; Staniford-Chen et al., 1998).

2.3.3.1 Algoritmos utilizados en la detección de intrusos en redes de telecomunicaciones.

Numerosos trabajos (Chauhan et al., 2012; García-Teodoro et al., 2009; Wu, & Banzhaf, 2010) han estudiado los algoritmos que mejores resultados han evidenciado en la búsqueda de intrusos redes de telecomunicaciones estos algoritmos pueden clasificarse en (García-Teodoro et al., 2009):

- Basados en la estadística. Donde no es requerido previamente el conocimiento relacionado con el funcionamiento normal de una red. En las técnicas basadas en la estadística se captura la actividad de la red y se crea un perfil que representa su comportamiento estocástico, este perfil está basado en distintas métricas, como la tasa de tráfico, el número de paquetes para cada protocolo, el número de diferentes direcciones IP, etc.
    - Univariante. Variables aleatorias Gaussianas independientes (Denning & Neumann, 1985).
    - Multivariante. Correlaciones entre varias métricas (Ye et al., 2002).
    - Modelos de series de tiempo. Intervalos de tiempo, contadores y algunos otros parámetros relacionados con el tiempo (Detecting hackers, 2013).
- Basados en el conocimiento. Son robustos, flexibles y escalables. Tienen un grado de dificultad y consumen mucho tiempo. Están disponibles para datos de alta calidad. Este enfoque es uno de los más utilizados en los Sistemas de Detección de Intrusos. Los sistemas expertos tienen como objetivo la clasificación de los datos de auditoría de acuerdo con un conjunto de reglas, que implican la implementación de tres pasos, en el primer paso los diferentes atributos y clases se identifican a partir de los datos de

entrenamiento. En segundo lugar, se genera un conjunto de reglas de clasificación, los parámetros y procedimientos. En tercer lugar, los datos de auditoría se clasifican de acuerdo a las reglas obtenidas previamente. Algunos de los trabajos realizados utilizan las siguientes técnicas:

- Maquinas de estados finitos. Estados y transiciones (Estevez-Tapiador et al., 2003).
- Lenguajes descriptivos. UML, N-gramas, LOTOS, etc. (García-Teodoro et al., 2009).
- Sistemas expertos. Clasificación basada en reglas (Hayes-Roth et al., 1984).

- Aprendizaje automático. Este tipo de aprendizaje es flexible y tiene un alto grado de adaptabilidad, posibilitando la captura de las interdependencias. Tienen una alta dependencia de la suposición sobre el comportamiento aceptado para el sistema y requieren un alto consumo de recursos. Las técnicas de aprendizaje automático están basadas en el establecimiento de un modelo explícito o implícito que posibilita categorizar a los patrones que fueron analizados. La aplicabilidad del aprendizaje automático en muchos casos coincide con la aplicación de técnicas estadísticas, el aprendizaje automático se sustenta en la construcción de un modelo que mejora su rendimiento basado de los resultados anteriores, teniendo la capacidad de cambiar su estrategia de ejecución al adquirir nueva información. Se han utilizado:
  - Redes bayesianas. Relaciones probabilísticas entre variables (Jensen, 1996).
  - Los modelos de Markov. Teoría estocástica de Markov) (Rabiner, 1989).
  - Las redes neuronales. Basadas en el funcionamiento del cerebro (Hagan et al., 1996).
  - La lógica difusa. Situaciones con alto grado de incertidumbre (Zadeh, 1988; Shah et al., 2003).

- Los algoritmos genéticos. Basados en la biología evolutiva (Goldberg & Holland1988).
- Clustización y detección de datos anómalos. Agrupamiento de datos basados en características comunes (Portnoy et al., 2001).

2.3.4 Minería de Datos para la detección de terroristas.

Cada vez más las organizaciones terroristas utilizan la infraestructura de Internet para comunicarse con sus seguidores, recaudar fondos, coordinar planes de acción, difundir mensajes propagandísticos, concretar acciones maliciosas en la red, etc. (Birnhack & Elkin-Koren, 2002).

Varias características de la Web (Goodman et al., 2007) posibilitan un entorno favorable para que grupos terroristas realicen sus actividades en Internet, por ejemplo el anonimato, la confidencialidad, la accesibilidad, el bajo costo, la facilidad de uso, la fuerza multiplicadora, etc.

Numerosos gobiernos están invirtiendo grandes esfuerzos en el desarrollo de nuevos métodos y tecnologías para la identificación de las actividades terroristas en la web.

Las técnicas de minería de datos están siendo cada vez más investigadas para detectar actividades que puedan estar relacionadas con el terrorismo (Thuraisingham, 2009; Memon et al., 2009; Johnson, 2012; Ozgul & Aksoy, 2007; Goodman et al., 2007; Memon et al., 2007; Last et al., 2006; Qin et al., 2007).

Muchas críticas ha tenido la minería de datos al ser utilizada en la detección de actividades terroristas en la WEB, fundamentalmente estas se relacionan con la baja precisión en los resultados y las graves violaciones a la privacidad detectadas[56]. Para enfrentar estos problemas Jensen et al., (2003) recomiendan generar clusters de tamaño fijo para obtener etiquetas de clases reales, de esta manera se reducen los falsos positivos después de la segunda vuelta de clasificación mientras que se mantienen las tasas reales positivas en

[56] http://www.theguardian.com/world/2013/jun/10/white-house-nsa-leaks-edward-snowden (Visitado el 09/07/2013)

la primer vuelta del algoritmo de clasificación. Para reducir los requerimientos de información se utiliza el 20% de los datos, siendo este punto una característica ponderada a la hora de considerar la confidencialidad de los datos.

La detección de valores atípicos en datos espaciales puede ser aplicada en el hallazgo de actividades terroristas, para preservar la privacidad en este tipo de datos Xue et al., (2008) utilizaron un algoritmo de minería de datos llamado *PPSLOF* (privacy-preserving spatial local outlier factor) logrando extraer de manera eficiente los datos anómalos y manteniendo la privacidad de la información manipulada.

Los arboles de decisión son muy utilizados en el proceso de clasificación con el objetivo de detectar actividades terroristas en la WEB, entre otros se utiliza el algoritmo C4.5 (Last et al., 2006) y el algoritmo PRISM[57]. También se utilizan algoritmos basados en la lógica difusa para clusterizar (Shah et al., 2003).

A partir de la necesidad de realizar una detección temprana de actividades terroristas en la web, se han desarrollado diversos métodos, técnicas y procedimientos, como por ejemplo Data Mining Investigativa desarrollado en el punto 2.3.4.1, o los Sistemas de Detección de Terroristas desarrollado en el punto 2.3.4.2.

2.3.4.1 Data Mining Investigativo.

Se puede definir al Data Mining Investigativo (Shaikh et al., 2007), IDM (Investigative Data Mining) según sus siglas en inglés, como la técnica que se utiliza para la visualización, clasificación, determinación de asociaciones y predicción de comportamientos relacionados con el terrorismo. Las actividades relacionadas el IDM están centradas en objetivos muy acotados dentro de una gran población de referencia y tienen como meta el identificar los vínculos y relaciones de una variedad mucho más amplia de actividades.

[57] http://www.theguardian.com/world/2013/jun/23/edward-snowden-nsa-files-timeline (Visitado el 01/07/2013)

Existen diferencias (Memon et al., 2009) entre el uso de la minería de datos tradicional y el IDM, en la tabla 2.9 se presentan las principales diferencias.

Data Mining Tradicional	Data Mining Investigativo
Descubre modelos comprensivos de bases de datos para desarrollar patrones estadísticamente válidos.	Detecta instancias particulares de la base de datos que están implicadas en patrones considerados raros.
No hay puntos de partida.	Puntos de partida conocidos o patrones previamente estimados por los expertos.
Aplica los modelos sobre el set de datos completo.	Reduce el espacio de búsqueda.
Tuplas independientes	Tuplas relacionadas
Es necesaria una consolidación mínima de los datos.	La consolidación de los datos es un factor clave de éxito.
Atributos concentrados	Atributos dispersos
Datos relativamente homogéneos	Datos relativamente heterogéneos
Políticas de privacidad en general uniformes.	Políticas de privacidad en general no uniformes.

Tabla 2.9. Data Mining tradicional comparado con IDM. (Memon et al., 2009)

El objetivo principal de esta técnica es encontrar actores importantes, relaciones, subgrupos, roles, etc., con el fin de identificar actividades terroristas. Se diferencian tres niveles de análisis:

- El elemento. En el nivel del elemento el interés se relaciona con las propiedades de los actores individuales, enlaces, o incidencias
- El grupo. En el nivel de grupo, el interés es clasificar los elementos de una red y las propiedades de las redes secundarias.
- La red. En el nivel de red, El interés se centra en las propiedades de la red general, como la conectividad o el equilibrio.

2.3.4.2 Sistemas de Detección de Terroristas.

Se trata de un nuevo tipo de sistema que tiene entre otros el objetivo de detectar usuarios que están vinculados con actividades relacionadas al terrorismo (Elovici el at., 2004). El sistema se basa en el seguimiento en tiempo

real del tráfico de internet de un grupo definido de usuarios, grupo que debe tener sospechas de estar integrado por terroristas. En general este tipo de sistemas trabaja con el contenido textual de la WEB y está formado por dos módulos:

- *Módulo de entrenamiento*. Este módulo trabaja en forma batch. La entrada es un conjunto de páginas cuyos contenidos se relacionan con la actividad terrorista, se aplican algoritmos de clusterización al contenido textual de esas páginas, obteniéndose un conjunto de vectores que representan eficientemente típicas áreas de interés de los terroristas.
- *Módulo de detección*. En este caso el modulo opera en tiempo real y realiza un seguimiento en línea del tráfico entre los usuarios que se controlan y las páginas WEB a las que acceden. Construyéndose un vector que representa el perfil del usuario. Este perfil construido se mantiene durante un período junto con el número de transacciones definidas por los parámetros del sistema operativo. La similitud se mide entre cada perfil de usuario y las áreas de interés típicamente relacionadas con el terrorismo. Cuando se detecta una sospechosa relación entre un usuario específico y contenidos relacionados con el terrorismo se produce un alerta sobre ese usuario, a través de la dirección IP que debe ser solicitada al proveedor de internet, se identifica físicamente el ordenador, desde el cual ese usuario accede a Internet.

2.4 Minería de datos y detección de outliers.

Los datos y la información son en la actualidad de fundamental importancia en las empresas y organismos, de la calidad de los datos depende la calidad de la información que se utiliza tanto en la operatoria cotidiana como en la toma de decisiones.

De manera intuitiva se puede afirmar que la calidad de una observación dentro de un set de datos se refleja por la relación que los mismos tienen con

otras observaciones del mismo set de datos que se obtuvieron bajo similares condiciones. Es común encontrarse con datos que parecen ser distintos que el resto de los datos, por tener valores más pequeños o más grandes, por tener características distintas que el resto de la muestra. Estos datos han sido denominados de diferentes maneras, datos anómalos, datos atípicos, valores extremos, datos sucios, outliers, etc.

Formalmente puede definirse un outliers como un dato que por ser muy diferente a los demás pertenecientes a un mismo conjunto de datos, como por ejemplo a una base de datos, puede considerarse que fue creado por un mecanismo diferente (Hawkins, 1980).

Desde el punto de vista empírico (Arnold & Salinas, 2012; Johnson & Kuby 2008) se afirma que si la distribución de los datos es simétrica y unimodal, o sea que se trata de una distribución normal, entonces alrededor de un 68% de los datos están incluidos dentro de ±1 desviaciones estándar de la media, un 95% dentro de ±2 desviaciones y 99.7% dentro de ±3 desviaciones estándar de la media. La figura 2.3 muestra una grafica con esta regla empírica.

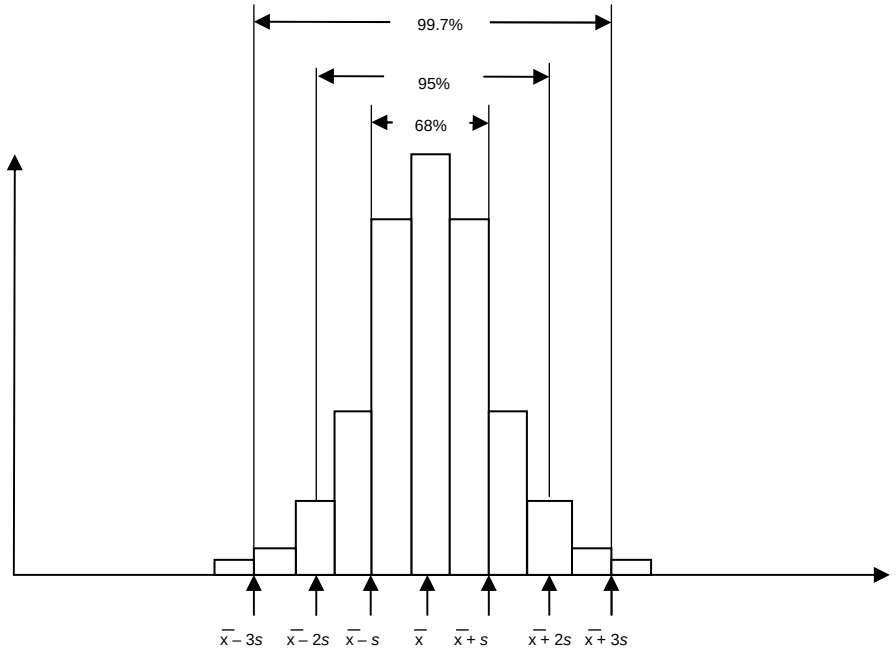


Figura 2.3. Regla empírica para detectar outliers en una distribución normal

Desde el punto de vista estadístico se puede identificar a los datos anómalos en una grafico de caja o bigote cuando se dan alguna de las dos siguientes condiciones (Reimann et al., 2005):

• Cuando un dato es < Q1 – 1.5 (Q3-Q1). Valor mínimo.

• Cuando un dato es > Q3 + 1.5 (Q3-Q1). Valor máximo.

Un grafico de caja o bigote permite representar los valores máximos, mínimos, los valores atípicos que se encuentran por debajo y por encima de los valores máximos y mínimos definidos, así como la distribución de un determinado set de datos, en la figura 2.4 se observa un grafico de bigote para el siguiente set de datos: 50/16/11/13/8/9/13, claramente “50” es un outlier, donde:

- Q1, denominado cuartil de orden 1, es tal que el 25% de los valores es inferior a él.
- Q2, denominado cuartil de orden 2, representa la mediana de la muestra y es tal que el 50% de los valores es inferior a él.
- Q3, denominado cuartil de orden 3, es tal que el 75% de los valores es inferior a él.

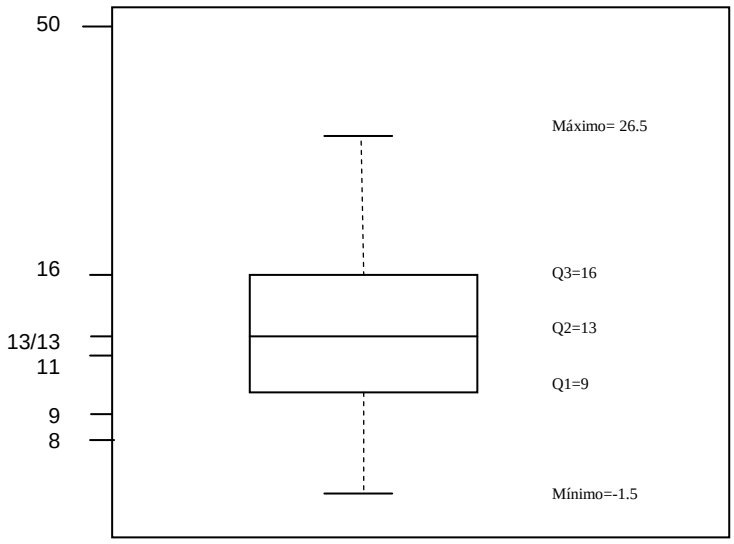


Figura 2.4. Grafico de bigote para definir outliers.

Las razones por las cuales existen datos anómalos pueden deberse a la carga incorrecta de los datos, a errores en el sistema software, a que el dato proviene de una población distinta, a algún tipo de fraude, entre otras.

En la tarea que realizan los auditores donde se trabaja en muchos casos con datos categóricos o donde no se trabaja con una distribución estándar de los datos, la identificación de datos anómalos tiene una enorme dificultad.

La presencia de datos anómalos en una base de datos puede crear distorsiones al realizar cualquier tipo de análisis sobre la misma. Sin embargo son menos frecuentes los estudios sobre la calidad de los datos, considerando a los outliers como posibles datos anómalos, comparándolos con el resto de aplicaciones de la minería de datos. La distorsión que producen este tipo de datos puede tener consecuencias graves dependiendo del campo de aplicación, por ejemplo datos anómalos en una resonancia magnética (Spence et al., 2001) pueden indicar la presencia de tumores malignos cuando en realidad se trata de un caso de falso positivo; datos anómalos en transacciones con tarjetas de crédito (Aleskerov et al., 1997) pueden significar algún tipo de fraude, o un patrón de trafico anómalo en una red de telecomunicaciones (Kumar, 2005) puede significar que un ordenador hackeado este enviando datos sensibles a un destino no autorizado. La figura 2.5 muestra las anomalías de un conjunto de datos en dos dimensiones, los datos tienen dos regiones normales (N_1 y N_2) donde se encuentran la mayoría de los datos, los puntos O_1 y O_2 así como el conglomerado O_3 representan datos anómalos en este caso por estar alejados de los datos normales.

Buscar anomalías realizando consultas manuales o formalizar un análisis de tipo secuencial sobre los datos, requiere conocer previamente los posibles valores anómalos, dado el tamaño de las bases de datos en muchos casos esta búsqueda es inviable. Para tareas de auditoría es relevante tener mecanismos que permitan automatizar estas prácticas, entre las cuales la aplicación de la minería de datos resulta interesante debido a su capacidad para detectar patrones y relaciones entre los datos que no son evidentes. La búsqueda de valores anómalos no es nueva, desde hace mas de 200 años

(Boscovich, 1757) se han descartado los datos que parecen discordantes del resto de la muestra.

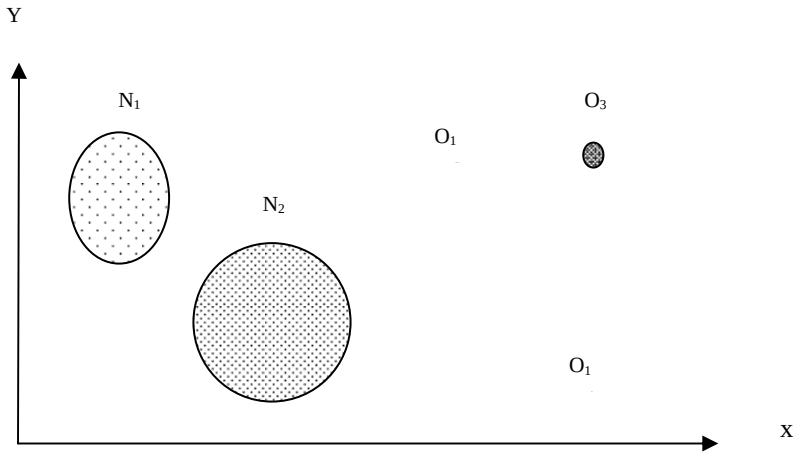


Figura 2.5. Representación grafica de outliers

La búsqueda de outliers tiene dificultades que pueden resumirse en los siguientes puntos:

- Es muy difícil definir que son datos normales cuando no se está trabajando con una distribución definida, en el caso de los sistemas de aplicación al no tener en general una distribución definida de sus datos la identificación de outliers se ve dificultada.
- La frontera entre datos normales y anormales en muchos casos es imprecisa, y aunque la aplicación de procesos de minería de datos determine tuplas consideradas outliers, es siempre necesario un análisis y evaluación posterior por parte del auditor ya que puede tratarse de datos extremos que pueden ser normales en la operatoria de una empresa, por ejemplo una venta extraordinaria.
- El concepto de datos anómalos en muchos casos depende del dominio de la aplicación, este es uno de los problemas fundamentales a resolver ya que muchas de las pruebas que se realizan para evaluar determinados algoritmos dependen del set de datos con que se hagan y el proceso de generalización es en muchos casos riesgoso.

- En muchos dominios el concepto de normalidad evoluciona imposibilitando la estabilidad del concepto de datos anómalos en ese dominio, en particular esta situación se da en empresas en crecimiento donde mes a mes se van desplazando los valores considerados extremos.
- Cuando los outliers son el resultado de acciones maliciosas, quienes realizan este tipo de acciones intentan de alguna manera disimular estos datos como normales y su identificación se ve dificultadas por la habilidad de la persona que realiza el fraude.

A las dificultades descriptas relacionadas con la búsqueda de outliers, es necesario incorporar el concepto de inliers, se trata de observaciones detectadas como atípicas pero que no tienen el comportamiento de un verdadero outlier, o sea se comportan de manera similar que el resto de los datos de nuestro análisis considerados como normales, es decir que una vez detectados los outliers es necesario descartar que no se trate de un caso de inliers. Existen incipientes investigaciones en la detección de inliers (Jouan-Rimbaud et al., 1999; Fernández Pierna et al., 2003; Anbaroglu, 2008; Fleck & Duric, 2009; Yoursi, 2010).

Considerando los problemas mencionados, la resolución general del problema de la detección de datos anómalos es muy compleja de implementar, la mayoría de los algoritmos utilizados actualmente resuelven un tipo especifico de problema, es decir que la solución depende directamente del dominio del problema a resolver. Se han adoptado conceptos de distintas disciplinas, como la estadística, la minería de datos, el aprendizaje automático, etc., para resolver problemas específicos relacionados con la detección de datos anómalos, la figura 2.6 (Chandola et al., 2009) muestra los componentes principales asociados con cualquier técnica utilizada en la detección de outliers.

Existen trabajos que definen una taxonomía de las anomalías detectadas en la búsqueda de outliers (Chandola et al., 2009), donde se mencionan estudios realizados en diferentes contextos como detección de fraude tanto en tarjetas de crédito (Bolton & Hand, 2001; Teng, Chen & Lu, 1990), como en teléfonos celulares (Fawcett & Provost, 1999), entre otros.

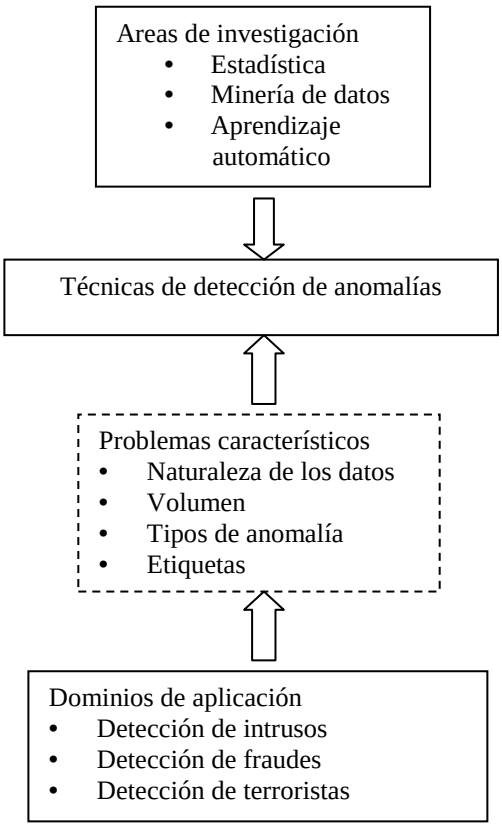


Figura 2.6. Componentes de las técnicas de detección de outliers. (Chandola et al., 2009)

La siguiente lista (Hodge & Austin, 2004) resume las aplicaciones que utilizan la detección de outliers:

- Detección de fraudes con tarjetas de crédito.
- Detección de intrusos en redes u ordenadores.
- Detección de fraudes en solicitudes de préstamos.
- Supervisión del rendimiento de una red.
- Diagnostico de fallos en redes.
- Análisis de imágenes satelitales.
- Detección de defectos estructurales.
- Detecciones sospechosas en imágenes.
- Aplicaciones criticas de seguridad industrial.
- Monitoreo de condiciones médicas de un paciente.

- Investigación farmacéutica.
- Diagnósticos médicos.
- Detección de actividades terroristas.
- Detección de anomalías en la WEB.

En la sub-sección 2.4.1 se presentan los diferentes enfoques y métodos para abordar la detección de outliers, en 2.4.2 se explican los criterios para la elección de métodos para detectar datos anómalos, en 2.4.3 se realiza una clasificación de las metodologías utilizadas en la detección de outliers, en 2.4.4 se hace una comparación de métodos de detección de outliers.

2.4.1 Enfoques y métodos para abordar el problema de la detección de outliers.

En (Hodge & Austin, 2004; Chandola et al., 2009) se proponen tres enfoques para abordar la problemática relacionada con la detección de outliers, los mismos son:

- *Tipo 1. Detección de outliers con aprendizaje no supervisado*.
  En esta tipología se determinan los valores atípicos sin contar con el conocimiento previo de los datos. Se trata fundamentalmente de un enfoque de aprendizaje no supervisado. Se procesan los datos como una distribución estática, señalándose los puntos más remotos del set de datos y se señala a los mismos como potenciales datos con valores atípicos. Se supone que los datos anómalos están separados de los datos considerados normales. En la Figura 2.7, se observa el grupo principal de datos y alejado del mismo están los puntos V, W, X, Y y Z a los que se puede señalar como posibles valores atípicos.

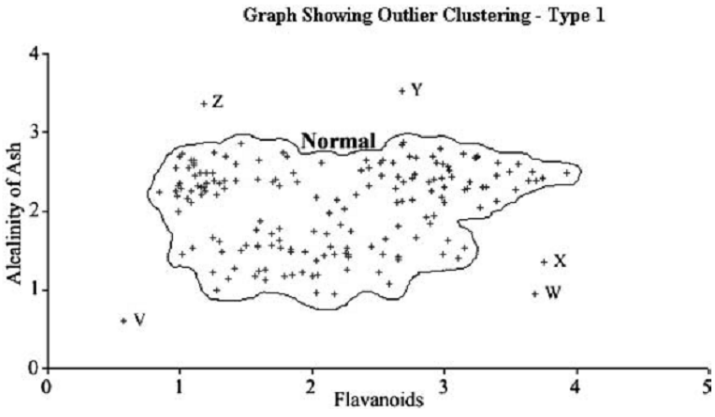


Figura 2.7. Outliers de tipo 1. Data set relacionado con la actividad vitivinícola. (Blake & Merz, 1998)

Se requiere que todos los datos estén disponibles antes de procesarlos, pero una vez que el sistema tenga una base de datos con la suficiente cobertura de casos es posible comparar los nuevos registros con los datos existentes y determinar si se trata o no de outliers.

Comúnmente se emplean dos sub-técnicas, el diagnóstico y el alojamiento (Rousseeuw & Leroy, 1996). Un enfoque diagnóstico para la detección de valores atípicos señala y destaca los puntos que poseen posibles valores atípicos, una vez detectados estos puntos el sistema los elimina. La técnica de alojamiento incorpora los valores atípicos en el modelo de distribución de los datos que generan y emplea un método robusto de clasificación. Estos métodos robustos de clasificación pueden soportar los valores atípicos en los datos.

- *Tipo 2. Detección de outliers con aprendizaje supervisado.*

  En este caso se trata de la aplicación de un proceso de clasificación con aprendizaje supervisado, los datos deben estar previamente clasificados en normales y anormales. En la figura 2.8 se muestran tres subclases de datos normales y valores anormales alejados pre-etiquetados como tales, este tipo de enfoque puede ser utilizado para

la clasificación en línea donde el algoritmo aprende del modelo para poder clasificar luego nuevos ejemplos donde se desconoce cuál es el tipo de etiqueta que tienen. Los algoritmos de clasificación requieren de una buena distribución de los datos normales y anormales.

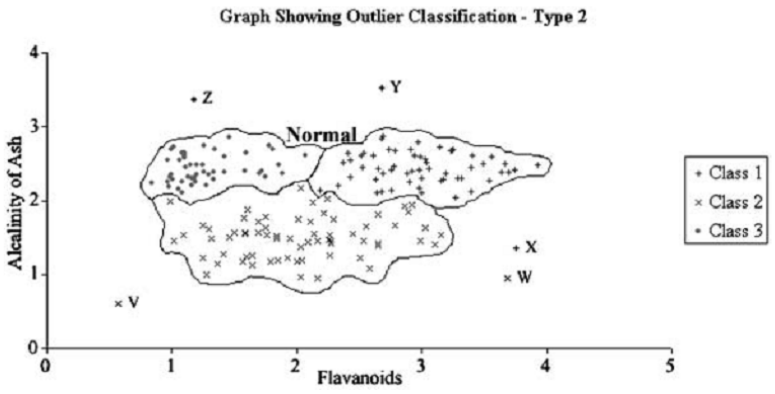


Figura 2.8. Outliers de tipo 2. Data set relacionado con la actividad vitivinícola. (Blake & Merz, 1998)

- *Tipo 3. Detección de Outliers con Aprendizaje semi-supervisado.*
  Numerosos autores (Japkowicz et al., 1995; Fawcett & Provost, 1999) llaman a esta técnica como detección de novedades o reconocimiento de novedades, se trata de una técnica de clasificación semi-estrucurada, donde se entrena el algoritmo para reconocer la clase normal de datos, pero el algoritmo aprende a reconocer los datos anómalos. Es adecuado para datos tanto estáticos como dinámicos, ya que el algoritmo se entrena sólo para reconocer los datos normales, el objetivo final es definir un límite de normalidad.

  En la figura 2.7 se observa que solo se aprendió la clase con datos normales y se estableció un límite de la normalidad. Si los puntos V, W, X, Y y Z de la figura 2.7 se comparan con el detector de novedades, estos puntos se etiquetarán como anormales ya que se encuentran fuera de los límites de normalidad, un algoritmo especifico podría detectar el grado en que ese punto es un outlier.

Este enfoque requiere tener un set de datos que cubra todo el espectro de normalidad que los mismos puedan tener, al no requerir de datos anómalos para su entrenamiento es muy útil para los casos donde es muy costoso obtener datos anormales. También es muy útil cuando existe volatilidad en el concepto de datos normales dentro de un determinado entorno, donde se vuelve a entrenar al algoritmo y se corren los límites de normalidad establecido, detectando por ejemplo de esta manera un nuevo tipo de fraude.

2.4.2 Criterios para la elección de métodos de detección de outliers.

El objetivo de los enfoques descriptos en el punto 2.4.1 es en definitiva mapear datos en vectores para poder aislar en una clase a los valores atípicos. Esta clase puede tener atributos numéricos y simbólicos, atributos discretos (ordinales), atributos categóricos (sin orden numérico), en general los métodos estadísticos y las redes neuronales requieren atributos de tipo numéricos, las técnicas de aprendizaje automático son capaces de clasificar también atributos simbólicos. Los valores anómalos se determinaron a partir de la "cercanía" de otros valores anómalos.

Deben considerarse dos elementos para seleccionar el método adecuado para detectar outliers:

- Selección de un algoritmo que puede modelar con precisión la distribución de los datos y que el mismo pueda reconocer los puntos periféricos del modelo. El algoritmo debe tener escalabilidad para poder procesar el set real de datos.
- Selección de una vecindad adecuada para los valores atípicos. Algunos algoritmos definen límites de normalidad, en muchos casos es el usuario quien debe definir esos parámetros como por ejemplo el límite de la vecindad o la cantidad de elementos mínimos a considerar en el vecindario.

2.4.3 Taxonomía de metodologías de detección de outliers.

Se han desarrollado muchas clasificaciones de métodos de detección de outliers, en la sub-sección 2.4.3.1 se presentan los métodos univariantes y multivariantes; en 2.4.3.2 se desarrollan los métodos paramétricos y no paramétricos; en 2.4.3.3 se explican los métodos basados en la estadística, en la distancia y en la densidad; en 2.4.3.4 se presentan los métodos basados en técnicas de clustering; en 2.4.3.5 se explican los métodos basados en redes neuronales; en 2.4.3.6 se desarrollan otros métodos de detección de outliers y en 2.4.3.7 se realiza un resumen de los métodos descriptos.

2.4.3.1 Métodos univariantes y los métodos multivariantes.

En el caso de los primeros se trata de métodos que trabajan con una sola variable para describir un determinado objeto, en cambio los multivariantes utilizan varias variables para cada individuo u objeto que es objeto de estudio.

Inicialmente se utilizaron los métodos univariantes para la detección de valores atípicos, los mismos están basados en el conocimiento previo de una distribución de los datos (Barnett & Lewis, 1994),

En el caso de los métodos multivariantes se observa que es muy complejo detectar un valor atípico cuando cada variable es considerada en forma independiente, es necesario considerar la relación entre las variables para detectar valores anómalos.

Cuando se trabaja con muchos valores atípicos en una muestra con características multivariantes los outliers están sujetos a un efecto de enmascaramiento de sus características que los hacen definir como posibles valores anómalos (Acuna & Rodriguez, 2004).

La detección de outliers con aprendizaje supervisado y semi-supervisado (enfoques de tipo 2 y 3 definidos en la sección 2.4.1) es la que mejor se adapta a los métodos univariantes, en cambio para los métodos multivariantes la detección de outliers tipo 1 de aprendizaje no supervisado es la que mejores resultados da.

2.4.3.2 Métodos paramétricos y métodos no paramétricos.

Los métodos paramétricos están basados en la estadística y suponen una distribución de los datos que esta subyacente (Hawkins, 1980; Rousseeuw & Leory, 1987; Barnett & Lewis, 1994), o se basan en estimaciones estadísticas de los parámetros de distribución desconocidos (Hadi, 1992; Caussinus & Roiz, 1990). Los métodos paramétricos en general no son los más adecuados para un set de datos que tenga alta dimesionalidad y para el conjunto de datos donde no hay conocimiento previo de la distribución de los mismos (Papadimitriou et al., 2003).

Los métodos no paramétricos son modelos libres que no se basan en una distribución predefinida de los datos (Williams et al., 2002), dentro de esta categoría se destacan los métodos de minería de datos, este tipo de método también se lo conoce como basados en la distancia, utilizando en general medidas de distancia locales (Knorr & Ng, 1997; Knorr & Ng, 1998; Knorr et al., 2000; Fawcett & Provost, 1997; Breunig et al., 2000; Hawkins, 1980).

La detección de outliers con aprendizaje supervisado y semi-supervisado es la que mejor se adapta a los métodos paramétricos, la detección de outliers tipo 1 de aprendizaje no supervisado es la que mejor resultado da en los métodos no-paramétricos.

2.4.3.3 Métodos basados en la estadística, métodos basados en la distancia y métodos basados en la densidad.

Algunos autores definen una taxonomía basada en la estadística, la distancia y la densidad (Niu et al., 2011).

- Los métodos estadísticos se basan en la premisa que la distribución de los datos es conocida. Como se estableció anteriormente, este tipo de método tiene el grave problema que en muchas situaciones no es conocida la distribución de los datos.
- Los métodos basados en la distancia se relacionan con la medida de distancia entre los objetos, cuanto mayor sea la distancia de un

objeto en relación al resto de la muestra, éste objeto es considerado un outlier.

- Los métodos basados en la densidad (Kuna et al., 2011) están fundamentados en la estimación de densidad de cada elemento del set de datos, los datos que se encuentran en regiones de baja densidad y que son relativamente distantes de sus vecinos se consideran outliers. En general este tipo de método utiliza el aprendizaje no supervisado, dado que no se conoce previamente un atributo "clase" que determine las tuplas normales y las tuplas anómalas.

La detección de outliers con aprendizaje supervisado y semi-supervisado es la que mejor se adapta a los métodos estadísticos, la detección de outliers tipo 1 (descripto en la sección 2.4.1) de aprendizaje no supervisado es la que mejor resultado da en los métodos basados en la distancia y la densidad.

2.4.3.4 Métodos basados en técnicas de clustering.

Se trata de un método que utiliza el aprendizaje no supervisado, donde la agrupación de los datos se relaciona con características comunes de los mismos. Este tipo de método es muy utilizado para descubrir conocimiento oculto, patrones de comportamiento y valores extremos (Kuna et al., 2010b), los métodos basados en técnicas de clustering se utilizan en bases de datos donde no existe ningún conocimiento previo de su distribución.

Analizando la distancia entre los elementos de un set de datos el criterio es que cuanto mayor es la distancia entre un objeto de una base de datos y el resto de la muestra, mayor es la posibilidad de considerar al objeto como un valor anómalo. Existen varios métodos que posibilitan medir la distancia entre los objetos:

- Distancia Euclídea, la misma se calcula para t variables de la siguiente manera:

$$d_{ij} = \sqrt{\sum_{k=1}^{t} (X_{ik} - X_{jk})^2} \quad (1)$$

- La distancia de Manhattan, también conocida como función de la

distancia absoluta o City Block, que para t variables se calcula de acuerdo a la siguiente fórmula:

$$d_{ij} = \sum_{k=1}^{t} \left| X_{ik} - X_{jk} \right| \quad (2)$$

- La distancia de Mahalanobis. Cuya fórmula es:

$$d_{ij} = \left(X_i - X_j\right)' \sum^{-1} \left(X_i - X_j\right) \quad (3)$$

Xi y Xj son matrices fila $(1 \times p)$ de observaciones para cada sujeto y Σ es la matriz de varianzas - covarianzas de las variables consideradas.

Las técnicas de agrupación o clustering se pueden clasificar de la siguiente manera:

- Agrupamiento jerárquico, se produce una descomposición jerárquica del conjunto de datos analizado, creándose un gráfico conocido como *dendograma* que representa la manera en que los datos se agrupan en clases y la distancia entre ellos. Los datos pueden agruparse en dos categorías, agrupación por divisiones (top-down) o agrupación por aglomeración (bottom-up), la distancia entre clusters se puede medir entre los centroides, entre los vecinos más cercanos, entre los vecinos más lejanos, entre otros métodos. Algunos de los algoritmos jerárquicos son: ROCK (Guha et al., 2000), CHAMALEON (Karypis et al., 1999), CURE (Guha et al., 1998), BIRCH (Zhang et al., 1997).
- Métodos basados en particiones, en los que se realizan divisiones sucesivas del conjunto de datos. Los objetos se organizan en k grupos, de modo que la desviación de cada objeto debe reducirse al mínimo en relación con el centro de la agrupación. Este método es ideal cuando se trabaja con una gran cantidad de datos, su desventaja es que el valor de k debe ser definido anteriormente y la selección del centro de cada grupo es arbitraria. Algunos de los algoritmos que utilizan este método son: K-MEANS (Huang, 1998), CLARA (Kaufman & Rousseeuw 2009), CLARANS (Ng & Han, 2002).
- Métodos basados en la densidad, donde cada cluster se relaciona con una medida vinculada con este parámetro. Aquí los objetos situados en regiones con baja concentración son considerados

anómalos. Entre los algoritmos basados en la densidad se destacan: LOF (Breunig et al., 1996), DBSCAN (Tan, 2007), Optica (Ankerst et al., 1999).

El método basado en clusters ya que utiliza el aprendizaje no supervisado se relaciona con el enfoque de tipo 1 descripto en la sección 2.4.1.

2.4.3.5 Métodos basados en redes neuronales

Las redes neuronales son otro importante tipo de tecnología a utilizar en la detección de datos anómalos (Hodge & Austin, 2004). Se trata de un modelo que permite la generalización y tiene capacidad de aprender. Son técnicas que imitan el comportamiento de las neuronas humanas, aprendiendo un modelo a través del entrenamiento de los pesos que conectan a las neuronas entre sí. Es necesario el entrenamiento para optimizar el funcionamiento de la red. En muchos casos las redes neuronales están impactadas por problemas relacionados con la alta dimensionalidad de los datos. Las redes neuronales en general se clasifican de acuerdo a los algoritmos o métodos de entrenamiento:

- *Redes con aprendizaje supervisado*.

  El aprendizaje se realiza mediante un entrenamiento que está controlado por un elemento externo denominado supervisor que establece cual es la respuesta que debe dar la red dada una determinada entrada. Es función del supervisor evaluar la salida y de ser necesario modificar los pesos de las conexiones entre las neuronas para mejorar los resultados. Existen tres formas de aprendizaje:

  - Por refuerzo, se caracteriza por no contar con un ejemplo completo del comportamiento deseado de la red, la tarea en este caso del supervisor es indicar mediante una señal de refuerzo si la salida obtenida en la red se ajusta a la salida esperada y de acuerdo a los resultados se ajustan los pesos sinápticos utilizando un mecanismo de probabilidades. Ejemplos de algoritmos de este tipo de redes son el Linear Reward-Penalty

(Narendra & Thathachar, 1974) y Adaptative Heuristic Critic (Barto, 1984).

- Aprendizaje por corrección de error: El aprendizaje se basa en ajustar los pesos sinápticos de las conexiones de la red de acuerdo a las diferencias entre los valores esperados y los obtenidos después de entrenar a la red. Ejemplos de algoritmos de este tipo de redes son el Perceptron (Rosenblatt, 1958), Perceptron Multicapa (Bourlard & Kamp, 1988), Adaline (Nguyen & Widrow, 1990) entre otros.
- Aprendizaje estocástico: Se efectúan modificaciones aleatorias en los valores de los pesos sinápticos de las conexiones entre neuronas y se analiza su efecto en función del objetivo deseado y de distribuciones de probabilidad. Ejemplos de este tipo de redes son la red Boltzman Machine (Hinton & Sejnowski, 1986) y la red Cauchy Machine (Scheff & Szu, 1987).

Varios trabajos se han desarrollado utilizando este tipo de redes en la detección de valores anómalos en particular aplicando el Perceptron Multicapa (Nairac et al., 1999; Bishop, 1995; Hawkins et al., 2002).

El método basado en redes neuronales con aprendizaje supervisado se relaciona con el enfoque de tipo 2 y 3 descriptos en la sección 2.4.1.

- *Redes con aprendizaje no supervisado*.

Este tipo de red requiere una base de datos preclasificada para permitir el proceso de aprendizaje, este tipo de redes no recibe información del entorno que le señale si dada una determinada entrada la salida es la correcta, son redes que tienen la capacidad de autoorganizarse y deben estar en condiciones de encontrar regularidades, características, categorías que se pueden establecer en los datos de la entrada. En muchos casos estas redes generan clusters y aquel cluster más lejano es sospechado de contener datos anómalos. Las redes con aprendizaje no supervisado suelen ser de dos tipos:

- Aprendizaje hebbiano: el objetivo es obtener características de los datos de entrada, se ajustan los pesos de las conexiones en relación

a los valores de activación de las dos neuronas que están conectadas entre sí. Algunas de la redes con este tipo de aprendizaje son: la red de Hopfied (Hopfield, 1982), Additive Grossberg (Grossberg, 1976), Bidirectional Associative Memory (Kosko, 1988).

- Aprendizaje competitivo y cooperativo: en este tipo de aprendizaje las neuronas compiten entre sí con el objetivo de activarse, los objetos similares del set de datos se clasifican dentro de una misma categoría, activando la misma neurona de salida. Algunas de las redes con este tipo de aprendizaje son las redes autoorganizadas de Kohonen o redes SOM (Self-Organizing Map) (Kohonen, 1990) que crean un mapa topológico con regiones que tienen datos similares, las redes SOM son un tipo de red no supervisada y competitiva, que utilizan los mapas topológicos para generar mapas bidimensionales. Aplica vectores con valores de los datos que modifican los pesos sinápticos de diversas regiones de la red de manera tal que se van identificando y diferenciando los grupos a los que pertenecen las instancias clasificadas.

Numerosos trabajos desarrollan el uso de redes neuronales con aprendizaje no supervisado en la detección de outliers, en particular utilizando redes SOM (Muñoz & Muruzábal, 1998; Rustum & Adeloye 2007; Marsland, 2003).

Las redes neuronales con aprendizaje no supervisado en general se relacionan con el enfoque de tipo 1 descripto en la sección 2.4.1, aunque algunos trabajos donde se aplican redes neuronales se relacionan con el enfoque de tipo 3 (Caudell & Newman, 1993).

2.4.3.6 Otros métodos de detección de outliers

Existe otra variada cantidad de métodos de detección de outliers por ejemplo:

- *Métodos basados en el aprendizaje automático.*

  El aprendizaje automático (Michalski et al., 1998) es una rama de la inteligencia artificial que tiene como objetivo desarrollar técnicas que posibiliten a los ordenadores aprender a partir de ejemplos. Se trata de un proceso de inducción de conocimiento. El objetivo es crear programas que sean capaces de generalizar el comportamiento a partir de información que no se encuentra estructurada y que está dada en forma de ejemplos.

  El aprendizaje automático está centrado en el estudio de la complejidad computacional de los problemas. Existen muchos tipos de aprendizaje, por ejemplo: aprendizaje repetitivo, por instrucción, por analogía, por ejemplos, por observaciones y descubrimiento.

  Se basan en algoritmos como "divide y vencerás" armando una representación en base a reglas, donde se generan particiones en forma recursiva. Dentro de este tipo de métodos es posible mencionar los algoritmos: ID3 (Quinlan, 1986), C4.5 (Quinlan, 1993) y C5 (Quinlan, 1999). Pueden usarse en problemas de clasificación, regresión y agrupamiento.

  En varios trabajos (John, 1995; Zhou & Jiang, 2003; Abe et al., 2006) han utilizado el árbol de decisión C4.5 para detectar valores anómalos en datos categóricos. No es necesario ningún conocimiento previo de los datos para aplicar arboles de decisión, los mismos son sólidos, escalables, pueden ejecutarse en grandes bases de datos, soportan la alta dimensionalidad de los datos y fundamentalmente tienen un buen desempeño con datos con ruido.

  El algoritmo C4.5 se relaciona con el enfoque de tipo 2 descriptos en la sección 2.4.1.

- *Métodos difusos.*

  Las técnicas de clasificación tradicionales utilizan clasificadores "duros", donde cada objeto tiene una única categoría y muchos problemas tienen a la incertidumbre como uno de los factores a considerar a la hora de clasificar a sus elementos. Para solucionar

este problema se utilizan métodos de clusterización basados en la lógica difusa (Diaz et al., 2006). La aplicación de métodos de clasificación difusos permite brindar flexibilidad a la clusterización ya que cada objeto puede pertenecer a cualquier clase, definiéndose en los métodos difusos un grado de pertenencia, de esta manera los clusters son considerados conjuntos difusos. Los métodos de agrupamiento difuso se relacionan con el enfoque de tipo 1 descripto en la sección 2.4.1.

- *Métodos basados en la teoría de la información.*

  Se han desarrollado algunas experiencias (Ni et al., 2008) aplicando la teoría de la información (Shannon, 2001) en la detección de valores anómalos en bases de datos, ya que esta teoría permite establecer un marco que posibilita medir el grado de dependencia entre variables, proporciona una medida del grado de las características extraídas de cada elemento en relación a la clase a la que pertenece, de manera que midiendo la entropía entre los datos es posible detectar ruido.

  Los métodos que utilizan la teoría de la información en la detección de outliers se relacionan con el enfoque de tipo 1 descripto en la sección 2.4.1.

- *Métodos basados en algoritmos genéticos.*

  Este tipo de algoritmo simula la evolución de una población de individuos (Goldberg, 1989; Sivanandam & Deepa, 2007) aplicando un proceso iterativo sobre un conjunto de estructuras, cada una de ellas representan las características que posee en individuo en su entorno. Las poblaciones evolucionan mediante la selección, recombinación y mutación. En la medida en que las estructuras que contienen a las poblaciones tienen una aptitud relativamente buena sobreviven a cada generación. Desde la óptica de la resolución de problemas, un individuo representa una posible solución a un problema dado, el objetivo final de los algoritmos genéticos es buscar la mejor solución a un problema.

Los algoritmos genéticos han sido utilizados en la detección de outliers (Leardi, 1994; Crawford & Wainwright, 1995), se ha demostrado que los mismos son un eficiente método para resolver problemas y la detección de valores anómalos es un tipo de problema que estos algoritmos pueden resolver.

Los métodos que utilizan los algoritmos genéticos en la detección de outliers se relacionan con el enfoque de tipo 1 descripto en la sección 2.4.1.

- *Métodos híbridos.*

  Los métodos más recientes estudiados en la detección de outliers son los denominados híbridos, los mismos incorporan algoritmos de al menos dos categorías distintas de detección de datos anómalos, se utiliza este método para superar las deficiencias de un algoritmo particular potenciando los puntos fuertes de cada algoritmo y minimizando los débiles.

  Varios autores han utilizado este tipo de método con muy buenos resultados combinando por ejemplo:

  - El Perceptrón Multicapa con un Parzen Windows, que es una técnica no paramétrica de estimación de densidad (Bishop, 1995).
  - El Perceptron multicapa con el modelo oculto de Markov o HMM (Hidden Markov Model) que es considerada como la red bayesiana dinámica más simple (Smyth, 1994).
  - El sistema JAM (Java agents for meta-learning) integra cinco técnicas de aprendizaje automático, el árbol de decisión ID3 (Quinlan, 1986), CART (Breiman, 1993), C4.5 (Quinlan, 1993), Ripper (Cohen, 1995) y un clasificador basado en redes bayesianas.
  - Se combinaron tres clasificadores (Brodley & Friedl, 1996), un árbol de decisiones, el algoritmo de aprendizaje supervisado "*Maquina de Vector Soporte*" y un algoritmo para solucionar el

problema del viajante. El objetivo era identificar casos mal clasificados de píxeles de imágenes satelitales.

- Se combinaron (Deng & Mei, 2009) los algoritmos SOM y K-means con el objetivo de detectar fraudes en cien empresas financieras de China entre 1999 y 2006.
- Se realizó un estudio (Penny & Jolliffe, 2001) comparando seis métodos para detectar valores atípicos en set de datos multivariantes y la conclusión del trabajo sugiere utilizar una batería de métodos para optimizar la detección de outliers.

Se observa que esta es una de las metodologías con mayor futuro ya que aumenta de manera sustantiva las alternativas de solución a determinados problemas combinando algoritmos y logrando a partir de esa combinación disminuir las debilidades de cada algoritmo y potenciar sus fortalezas.

2.4.3.7 Resumen de métodos de detección de outliers.

La tabla 2.10 muestra un resumen de las principales metodologías de detección de outliers y el enfoque que cada una de ellas tiene, se destaca el enfoque de tipo 1 (no supervisado) ya que el 75% de los métodos relevados corresponde al tipo 1, un 37.5% al tipo 2 y un 37.5% al tipo 3, es importante destacar que algunos métodos soportan algoritmos de más de un tipo, y que en el caso particular de los métodos híbridos son los únicos que pueden implementar en forma simultánea más de un enfoque ya que justamente su característica es que aplican algoritmos de distintas metodologías en forma combinada, esta característica le da una enorme potencialidad a este tipo de métodos.

Enfoque / Metodología	1	2	3
Univariante		X	X
Multivariante	X		
Paramétrico		X	X
No paramétrico	X		
Estadístico		x	X
Distancia	X		
Densidad	X		
Cluster Jerárquico	X		
Cluster particiones	X		
Cluster densidad	X		
RRNN Aprendizaje Supervisado		X	X
RRNN Aprendizaje no Supervisado	X		X
Aprendizaje automático		X	
Lógica difusa	X		
Entropía	X		
Algoritmos genéticos	X		
Híbridos	X	X	X

Tabla 2.10. Enfoques por metodología

2.4.4 Comparación de métodos de detección de outliers.

Dado que los métodos y algoritmos de detección de outliers se relacionan con diferentes hipótesis (Ben-Gal, 2005), utilizan distintos sets de datos, fueron diseñados para solucionar problemas específicos, es muy complejo realizar una comparación directa entre los métodos desarrollados. Sin embargo hay algunos trabajos que comparan varios de los métodos de detección de outliers.

En (Penny & Jolliffe, 2001) se compararon seis métodos de detección de valores atípicos multivariantes. La conclusión del estudio se relaciona con una recomendación para utilizar una "*batería de métodos multivariantes*" en el conjunto de datos con el fin de detectar posibles valores atípicos, brindando una serie de recomendaciones para aplicar este tipo de solución, como por ejemplo el tamaño del set de datos, limitaciones de tiempo, dimensionalidad del set de datos, etc. La solución planteada por los autores se acerca a la metodología hibrida descripta en el punto 2.4.3.6.

En (Williams et al., 2002), se sugiere un método basado en redes neuronales, comparando las redes neuronales con dos métodos paramétricos basados en la estadística y un método no paramétrico especifico de la minería de datos. La conclusión de la comparación es que las redes neuronales tienen un funcionamiento mejor que el resto de los métodos analizados, en particular cuando se trata de grandes bases de datos. Afirmando que en algunos casos los métodos estadísticos tienen un buen rendimiento para grandes sets de datos.

En (Lu et al., 2003) se realiza una comparación de tres algoritmos para la detección de outliers espaciales. Se trata de dos algoritmos secuenciales y un algoritmo basado en la mediana, concluyen que los algoritmos utilizados reducen la cantidad de falsos positivos.

En la sección 2.3.2.1 se describe un trabajo realizado por Dean Abbott (Aboot et al., 1998) donde compara distintas herramientas y algoritmos utilizados en la detección de fraudes.

En la sección 2.3.2.2. se describe un trabajo realizado por Kirkos et al., (2007) donde se comparan diferentes algoritmos en la detección de fraudes internos.

Como se afirmó previamente son reducidos los trabajos que realizan comparaciones de métodos de detección de outliers, estas comparaciones para ser válidas deberían comparar algoritmos que resuelvan un mismo tipo de problema con idéntico set de datos, siendo muy difícil generalizar sus resultados.

2.5 Discusión estado del arte y problema detectado.

De acuerdo a lo presentado en este capítulo los principales temas relacionados con la aplicación de técnicas de minería de datos en el proceso de auditoría de sistemas se relacionan con la detección de fraudes en áreas de finanzas y contabilidad, detección de intrusos en redes y detección de terroristas. En la detección de outliers en bases de datos si bien existen antecedentes relacionados con este tipo de aplicación los mismos siempre se relacionan con la detección de tuplas consideradas anómalas.

La detección de outliers tiene muchas dificultades relacionadas con el tipo de dato, la dimensionalidad, el dominio, el volumen, etc., junto con estos problemas se encuentra la determinación una vez detectado un outlier que el mismo en realidad no sea un inlier. Por lo que la generalización de un algoritmo para resolver cualquier tipo de búsqueda de valores anómalos se convierte en inviable.

Existen muchos métodos para detectar los valores anómalos, los mismos pueden resumirse en tres enfoques, la detección de outliers basada en el aprendizaje no supervisado, basada en el aprendizaje supervisado y la basada en el aprendizaje semi-supervisado. La mayoría de los algoritmos analizados se relacionan con el aprendizaje no supervisado.

Dentro de los métodos analizados uno de los mas emergentes es el híbrido, en este método se propone aplicar más de un algoritmo en la detección de outliers, ya que se considera que un solo algoritmo difícilmente pueda abordar las distintas aplicaciones, dimensionalidades, volúmenes, tipos de datos, etc., con las que un auditor debe enfrentarse en forma cotidiana.

En la actualidad las bases de datos son cada vez más complejas y más grandes, es habitual que un auditor tenga que analizar y evaluar bases de datos con millones de tuplas y cientos de atributos en cada tabla, este trabajo resulta imposible abordarlo en forma manual, pero también se dificulta enormemente cuando se debe analizar una base de datos con gran cantidad de atributos, ya que es, en la actualidad, posible detectar las tuplas que son sospechosas de incluir valores en atributos anómalos pero no es posible

detectar que atributo es el que debe ser específicamente analizado, la individualización del campo especifico que tiene valores anómalos es compleja en grandes bases de datos, insume tiempo y esfuerzo del auditor y tiene una carga subjetiva que depende de los antecedentes del profesional, por esto el objetivo de la tesis que es justamente la detección de los campos considerados anómalos se convierte en un aporte fundamental para automatizar todo el proceso de detección de outliers, permitiendo que sea realizado en menos tiempo, de manera más objetiva y sin necesidad de contar con un auditor con mucha experiencia en la actividad.

3 Materiales y métodos. Algoritmos y bases de datos utilizados.

Para abordar la solución al problema de la detección de campos considerados outliers en bases de datos se observa que no existe un algoritmo que realice esta tarea, los modelos implementados hasta el momento permiten detectar si una tupla tiene valores anómalos, pero no se encuentran experiencias en la detección de campos con valores anómalos.

Para dar una solución al problema detectado se propone implementar procedimientos que se basen en el modelo híbrido (ver sección 2.4.3.6). Este tipo de modelo incorpora algoritmos de al menos dos categorías de detección de outliers distintas, para de esta manera poder superar las deficiencias de un algoritmo particular potenciando los puntos fuertes de cada algoritmo y minimizando los débiles. Dada la diversidad de situaciones donde se debe aplicar la búsqueda de campos outliers, con diferentes sets de datos, diversa dimensionalidad, diferentes tipos de datos, etc., la integración de los tres enfoques descriptos en la sección 2.4.1 permitirá abordar el problema con una batería de métodos y algoritmos, complementando de esta manera la potencialidad de cada uno de estos algoritmos.

En el capítulo 3 se desarrollan los algoritmos utilizados en los procedimientos diseñados en el capítulo 4, los mismos son: LOF, K-means, C4.5, DBSCAN, Redes bayesianas, PRISM, y la Teoría de la Información. El algoritmo LOF se describe en la sección 3.1, el algoritmo K-means se explica en la sección 3.2, el algoritmo C4.5 es presentado en la sección 3.3, se realiza una introducción a la Teoría de la Información en la sección 3.4, las Redes bayesianas son explicadas en la sección 3.5, el algoritmo DBSCAN es presentado en la sección 3.6, en la sección 3.7 se desarrolla el algoritmo PRISM y finalmente en la sección 3.8 se explican las bases de datos utilizadas en la experimentación. Es de destacar que todos estos algoritmos así como la Teoría de la Información fueron aplicados en su formato original desarrollado por los autores.

3.1 Algoritmo basado en la densidad. LOF (Local Outlier Factor).

En general los algoritmos de agrupamiento utilizados para detectar valores atípicos no fueron diseñados para ese propósito. El algoritmo LOF, desarrollado por Breunig et al., (2000) fue creado específicamente para detectar outliers, difiere de la mayoría de los otros algoritmos de agrupamiento, donde la detección de valores atípicos es un beneficio secundario.

El valor de LOF en un objeto p representa el grado en el que p es un valor atípico. A continuación se presenta una explicación del algoritmo LOF original, desarrollado por sus autores:

Definición 1: *(DB(pct, dmin)-Outlier)*
Un objeto p en un set de datos D es un outlier si el cardinal (o sea la cantidad de elementos) del conjunto $\{q \in D \mid d(p, q) \leq dmin\}$ (donde d es la distancia, p y q son elementos del set de datos D que están a una distancia menor que $dmin$ de p) es menor o igual que el $(100 - pct)\%$ del tamaño del set de datos D.

Definición 2. *K-distancia 'p':*
Para cualquier entero positivo k, la k-distancia del objeto p, denominado como la *k-distancia*(p), es definido como la distancia $d(p,o)$ entre p *y un* objecto $o \in D$ tal que:
1) Para por lo menos k objetos $o' \in D \setminus \{p\}$ se cumple que $d(p,o') \leq d(p,o)$, y
2) Para a lo sumo $k-1$ objetos $o' \in D \setminus \{p\}$ se cumple que $d(p,o') < d(p,o)$.

Definición 3. Grado de vecindad, *K-distancia de 'p'*
Dada la *k-distancia* de p, la *k-distancia del vecindario de* p contiene todos los objetos cuya distancia de p no es mayor que la *k-distancia*, por ejemplo: $Nk\text{-}distancia(p)(p) = \{ q \in D\setminus\{p\} \mid d(p, q) \leq kdistancia(p) \}$.
Estos objetos q se llaman *k-vecinos* más cercanos de p.

Definición 4: Distancia de accesibilidad de un objeto *p* respecto a un objeto *o*. Donde *k* es un número natural. La *Distancia de accesibilidad* de *p* respecto al objeto *o* se define como $reach\text{-}dist_k(p, o) = max\{ k\text{-}distancia(o), d(p, o) \}$.

Hay dos parámetros que definen la noción de densidad, uno de ellos es el parámetro *MinPts* que especifica el mínimo número de objetos, el otro es el volumen. Estos dos parámetros determinan un umbral de densidad utilizados por los algoritmos de agrupamiento para operar. O sea, los elementos del set de datos o regiones están conectados si sus densidades vecinales superan el umbral de densidad dada. Para detectar valores atípicos basados en la densidad, es necesario comparar las densidades de los diferentes conjuntos de objetos, lo que significa que hay que determinar la densidad del conjunto de objetos de manera dinámicamente. Por lo tanto, se mantiene a *MinPts* como único parámetro y el uso de los valores de $reach\text{-}dist_{MinPts}(p, o)$, para $o \in N_{MinPts}(p)$ como una medida del volumen para determinar la densidad en la vecindad de un objeto *p*.

Definición 5. Densidad de accesibilidad local *(lrd)* de *p* (local reachability density). Se define como:

$$LRD_{MinPts}(p) = 1 \Big/ \left(\frac{\sum_{o \in N_{MinPts}(p)} reach\text{-}dist_{MinPts}(p, o)}{|N_{MinPts}(p)|} \right) \quad (4)$$

Este valor es la densidad de accesibilidad local del objeto *p*, *lrd(p)* se calcula mediante la inversa de la distancia media de *reach-dist (p,o)* sobre la base de los vecinos más cercanos de *MinPts-p*. Hay que considerar la densidad local puede ser ∞ si todas las distancias de accesibilidad al sumarlas son 0.

Definición 6. Local outlier factor (*LOF*) de *p* (Factor local de outliers).

$$LOF_{MinPts}(p) = \frac{\sum_{o \in N_{MinPts}(p)} LRD_{MinPts}(o) / LRD_{MinPts}(p)}{|N_{MinPts}(p)|} \quad (5)$$

El valor de LOF de un objeto p representa el grado en que se considera a p como un valor atípico, es la media de la relación de y sus $p's$ vecinos más cercanos de $MinPts$. Cuanto menor es $lrd(p)$ y cuanto mayor sea la densidad de accesibilidad local de los vecinos más cercanos de los valores de $p's$ de $MinPts$**,** mayor es el valor de LOF de p**.**

3.2 Algoritmo K-Means.

El algoritmo K-Means fue creado por MacQueen (1967), es uno de los más utilizados para realizar clusterización por su simpleza y eficacia, permite clasificar un conjunto de objetos en un número K de clusters, siendo K un número determinado a priori. Cada cluster se representa por su centroide (media ponderada) que se encuentra en el medio de los elementos que componen el cluster. La idea básica del algoritmo es una vez definidos los centroides ubicar el resto de los puntos en la clase de su centroide más cercano, a posteriori se recalculan los centroides y se reubican cada uno de los puntos en cada conglomerado, este proceso se repite hasta que no se produzcan cambios en la distribución de los puntos en una nueva iteración.

El algoritmo se realiza en 4 pasos siendo *O* un conjunto de objetos $Dn = (x1, x2, x3\ldots, xn)$ para todo i, xi reales y $k, v1,$ los centro de los clusters, los pasos son:

- Paso 1: Determinar el valor de K y elegir en forma aleatoria K objetos para formar los clusters iniciales, para cada cluster K el valor inicial del centroide es $= x_i$, siendo estos los únicos objetos de D_n que pertenecen al cluster.
- Paso 2: Reasignar los objetos del cluster. Según una medida de distancia para cada elemento *x* el prototipo que se le asigna es el que está más próximo al objeto
- Paso 3. Recalcular los centroides de cada cluster una vez que todos los objetos son asignados.

- Paso 4. Repetir los pasos 2 y 3 hasta que no se produzcan más reasignaciones.

Algunos de los inconvenientes de este algoritmo son:

- Solo puede aplicarse con atributos numéricos ya que es necesario calcular el punto medio.
- El no conocer a priori el valor de *K,* esto puede hacerlo poco eficaz, aunque existen varias métricas que permiten validar el valor de K.
- El algoritmo es sensible a los valores anómalos.

3.3 Algoritmo de inducción C4.5.

Inicialmente fue desarrollado para crear un árbol de decisión para determinar los campos significativos. Para ello el algoritmo C4.5 realiza particiones en forma recursiva utilizando la estrategia “primero en profundidad” (*depthfirst*) (Quinlan, 1993). En las pruebas sucesivas realizadas donde se comparan los ejemplos del set de datos, el algoritmo busca aquellos ejemplos que tienen la mayor ganancia de información. En el caso de atributos discretos la prueba considera una cantidad de resultados teniendo en cuenta el número de valores posibles que puede tomar el atributo.

La detección de los atributos significativos dentro de la base de datos aplicando algoritmos de inducción como el algoritmo C4.5 permite reducir el espacio de búsqueda de datos anómalos a solo aquellos campos que son relevantes en la base de datos, es decir, aquellos atributos que representan dentro del grupo de datos a los que aportan más información para clasificar al atributo *target* objetivo, y poder de esta manera optimizar la performance y funcionamiento del procedimiento que detecta los campos anómalos.

El pseudocódigo original del algoritmo C4.5 es el siguiente:

(R=Conjunto de atributos no clasificados, C=atributo clasificador, S=conjunto de entrenamiento)

Comienzo

Si S = vacío,

Devolver un único nodo con Valor Falla;

Si todos los registros de S tienen el mismo valor para C,
Devolver un único valor con el valor más frecuente de C en los registros de S;
Si R = vacío,
D ← atributo con mayor proporción de ganancia (D;S) entre los atributos de R;
Sean $\{d_j \mid j=1,2,\ldots, m\}$ los valores del atributo D;
Sean $\{d_j \mid j=1,2,\ldots, m\}$ los subconjuntos de S correspondientes a los valores de d_j respectivamente;
Devolver un árbol con la raíz nombrada como D y con los arcos nombrados $d_1,d_2,\ldots,d_m$, que van respectivamente a los árboles
C4.5 (R-{D}, C, S1), C4.5 (R-{D}, C, S2), C4.5 (R-{D}, C, Sm);
Fin.

Existen tres tipos de pruebas posibles y el sistema debe decidir cuál de ellas ejecutar en cada nodo para dividir los datos, estas pruebas pueden ser:

- Una prueba estándar para las variables discretas, con un resultado y una rama para cada valor posible de la variable.
- Una prueba más compleja, que se basa en una variable discreta, donde los valores posibles se asignan a un número variable de grupos con un resultado posible para cada uno de ellos, en lugar de asignarlos para cada valor.
- En el caso que una variable A que tiene valores numéricos continuos, se realiza una prueba binaria con resultados $A <= Z$ y $A > Z$, para lograr esto se debe determinar el valor límite de Z.

3.4 Teoría de la información.

La teoría de la información nace como un modelo matemático enunciado en 1948 por Claude Shannon (Shannon, 2001). El sistema de comunicación propuesto posee una fuente que determina los mensajes a ser transmitidos, un transmisor que codifica el mensaje convirtiéndolo en una señal que se propaga por medio de un canal de transmisión. La señal arriba al decodificador que la convierte nuevamente en el mensaje para el destinatario. Este mensaje puede

ser idéntico al generado en el emisor, o similar, en el caso que se encuentre sometido el canal de transmisión a una fuente de ruido durante la transmisión del mensaje. La información se mide mediante la entropía, que es un término de la termodinámica que mide el nivel de desorden de un sistema, la siguiente formula representa el cálculo de la entropía:

$$H = \sum_{k=1}^{m} p_k \log \frac{1}{p_k} \qquad (6)$$

La teoría de la información se refiere a la cantidad de información promedio que contienen los símbolos usados. Es decir cuanto menor probabilidad de aparición tiene un símbolo mayor es la cantidad de información que el mismo aporta.

Esta teoría aplicada en proceso de minería de datos (Ferreyra, 2007) indica la posibilidad de trabajar los datos desde binomios del tipo “mensaje de entrada” *(E)* y “mensaje de salida” *(S)* para detectar los outliers en cada atributo. Si consideramos el ejemplo de arrojar un dado *(E)*, sobre esta acción resultante *(S)* debería existir un número entre 1 y 6. Si esto no es así estaríamos en presencia de un outlier. De esta manera cuanto menor sea la probabilidad del par *(E) – (S)* analizado más posibilidades habrá que se trate de una inconsistencia.

Teniendo en cuenta lo anteriormente mencionado se utiliza LOF para analizar la entrada *(E)* tomando siempre como referencia los valores existentes en las salida *(S)* que corresponde al atributo definido como objetivo o target.

3.5 Redes bayesianas.

Se define a una red bayesiana como un grafo acíclico dirigido, donde cada nodo representa a las variables aleatorias y las flechas muestran las influencias causales entre las variables que pueden ser continuas o discretas, si un nodo es padre de otro significa que es causa directa del mismo. Se trata de un modelo que tiene un buen funcionamiento para representar el

conocimiento cuando existe un alto grado de incertidumbre. A través de la representación gráfica de las dependencias e interdependencias entre las variables que son parte del dominio permiten representar a un modelo causal (Pearl, 1988).

Las redes bayesianas aparte de modelar de manera cualitativa el conocimiento permiten expresar en forma numérica la intensidad de la relación entre las variables.

Una red bayesiana es una tupla $B=(G,\Theta)$, G es al grafo y Θ representa al conjunto de distribuciones de probabilidades $P(X_i|Pa(X_i))$ para cada una de las variables desde i=1 hasta i=n, dado el grafo G $Pa(X_i)$ representa los padres de la variable X_i.

Es posible representar a una red bayesiana de dos maneras:

- Como una base de reglas donde cada arco está representando a un conjunto de reglas que permiten asociar a las variables involucradas, donde las probabilidades cuantifican dichas reglas.
- Como una distribución de la probabilidad conjunta de las variables representadas en la red bayesiana.

Una distribución de probabilidades puede ser considerada como un modelo de dependencias aplicando la siguiente relación:

$$I(X,Y|Z) \Leftrightarrow P(X|YZ)=P(X|Z) \qquad (7)$$

donde X, Y, Z son subconjuntos de variables del modelo y $I(X,Y|Z)$ es una relación de independencia condicional. La probabilidad conjunta de n variables esta especificada por el producto de las probabilidades dado sus padres, como lo muestra la siguiente fórmula:

$$P(X_1, X_2, X_3,\ldots, X_N=\prod_{i=1}^{n}(PX_i)|Pa(X_i)) \qquad (8)$$

Se plantea el problema de optimización y lo que se desea es obtener la estructura en forma de árbol que más se aproxime a la distribución “real”. Para ello se utiliza una medida de la diferencia de información entre la distribución real (P) y la aproximada (P*):

$$I(P,P^*)= \sum_X P(x)log(P(X)/P^*(X)) \quad (9)$$

Para minimizar el valor de I se define una diferencia considerando la información mutua entre pares de variables, como lo muestra la siguiente fórmula:

$$I(X_i,X_j)= \sum_X P(X_i,X_j)log(P(X_i,X_j)/P(X_i)P(Xj)) \quad (10)$$

Chow & Liu, (1968) demostraron que la diferencia de información es una función del negativo de la suma de los pesos de todos los pares de variables que constituyen el árbol. Por lo que encontrar el árbol más próximo equivale a encontrar el árbol con mayor peso. Basado en lo anterior, el algoritmo original para determinar la red bayesiana óptima a partir de datos es el siguiente:

1. *Calcular la información mutua entre todos los pares de variables (hay exactamente n (n - 1) / 2 pares).*
2. *Ordenar las informaciones mutuas de mayor a menor.*
3. *Seleccionar la rama de mayor valor como árbol inicial.*
4. *Agregar la siguiente rama mientras no forme ciclo, si es así, desechar.*
5. *Repetir (4) hasta que se cubran todas las variables (n - 1 ramas).*

3.6 DBSCAN.

Se trata del primer algoritmo basado en la densidad (Ester et al., 1996), donde se definen los conceptos de punto central, borde y ruido. Los puntos centrales poseen un área de vecindad para un determinado radio que tiene por lo menos un número mínimo de puntos, o sea que su área de vecindad excede un determinado umbral. Un punto es central si el número de puntos en el área de vecindad que se define según el radio *Eps* excede un cierto umbral conocido, *MinPts*. Es decir, si se cumple:

$$|N_{Eps}(p)| \geq MinPts \quad (11)$$

El área de vecindad de un punto p que pertenece a una base de datos D esta dado por un radio Eps que se define como:

$$N_{EPS}(p) = \{q \in D | dist(p,q)\} \leq Eps \qquad (12)$$

Se trata de un algoritmo muy sencillo de implementar, donde la densidad de los puntos depende del radio del área de vecindad que se ha especificado. De manera que si el radio es lo suficientemente grande todos los puntos considerados deberán tener una densidad igual al número de puntos total del conjunto de datos. Por el contrario, si el radio es muy pequeño no todos los puntos tendrán una densidad igual a 1, es decir, el punto se encontrará aislado.

Los algoritmos basados en la densidad localizan regiones de alta concentración de puntos que se encuentran separadas entre sí por regiones con menor densidad.

Inicialmente el algoritmo comienza seleccionando un punto p arbitrario, si p es un punto central, se inicia la construcción de un grupo y se incorporan en dicho grupo todos los objetos denso-alcanzables desde p. En el caso que p no sea un punto central se lee otro elemento del conjunto de datos. Este proceso continúa hasta que todos los objetos han sido procesados. Los puntos que quedan fuera de los grupos formados se llaman puntos ruido, los puntos que no son ni ruido ni centrales se llaman puntos borde.

Esta es la manera en que DBSCAN crea los grupos, donde sus puntos son centrales o borde, se debe tener en cuenta que un grupo puede tener más de un punto central.

La idea principal del algoritmo DBSCAN es encontrar todos los puntos centrales, considerando que los puntos centrales de un grupo son aquellos que tienen un área de vecindad que contiene un número mínimo de puntos para un radio determinado. La forma del área de vecindad está determinada por la elección que se haga de la medida de la distancia entre dos puntos. En el caso de utilizar la distancia de Manhattan, el algoritmo tiende a formar grupos con forma rectangular.

3.7 PRISM.

Se trata de un algoritmo de aprendizaje basado en reglas, que como característica particular asume que el set de datos no posee ruido, esto tiene como ventaja que crea reglas que cubren la mayor parte de los elementos del set de datos, aislando las instancias para analizarlas por separado (Cendrowska, 1987).

Este algoritmo tiene la siguiente estructura:

Para cada clase C
Sea E = ejemplo de entrenamiento
Mientras E tenga ejemplos de la clase C
Crea una regla R con LHS vacío y clase C
Until R es perfecta do
Para cada atributo A no incluido en R y cada valor v,
Considera añadir la condición A = v al LHS de R
Selecciona el par A = v que maximice p/t
Añade A = v a R
Elimina de E los ejemplos cubiertos por R

El algoritmo funciona de la siguiente manera:

- Siendo t el numero de ejemplos que están cubiertos por la regla y sea p el número de ejemplos positivos que cubre la regla.
- El algoritmo PRISM añade condiciones a reglas que maximicen la relación p/t.
- Como se van eliminando los ejemplos que va cubriendo cada regla, las reglas que se construyen deben interpretarse en orden.
- Las reglas que dependen del orden de su interpretación se conocen como listas de decisión.
- Con varias clasificaciones es posible seleccionar la regla que cubra más ejemplos, y en el caso que no se tenga una clasificación se escoge la clase mayoritaria.

3.8 Bases de datos utilizadas en la experimentación.

Para realizar la experimentación y poder validar la misma se utilizaron 4 tipos de bases de datos:

- Bases de datos de laboratorio con fines experimentales, estas bases de datos fueron creadas de acuerdo a la distribución normal. La determinación de los valores anómalos para validar la experimentación fue establecida con métodos estadísticos.
- Bases de datos de laboratorio con fines experimentales, estas bases de datos fueron creadas sin responder a una determinada distribución. Fueron incorporados un conjunto de campos anómalos en forma aleatoria, estos campos con ruido fueron identificados para validar el resultado de las pruebas experimentales realizadas.
- Bases de datos reales, sin distribución conocida y con un atributo clase, siendo el contenido de este atributo el que determina si se trata de una tupla outlier o una tupla con datos normales.
- Bases de datos reales, sin distribución conocida y sin un atributo clase, en este último caso se requirió la asistencia de un experto en el dominio para determinar de manera manual cuales eran los campos considerados outliers.

4. Solución.

En este capítulo se presentan cuatro procedimientos que permiten detectar campos considerados anómalos para bases de datos relacionadas con sistemas de gestión. En la sección 4.1 se presentan las consideraciones generales de la solución propuesta. La sección 4.2 desarrolla dos procedimientos para detectar outliers en bases de datos numéricas. La sección 4.3 presenta un procedimiento para detectar outliers en bases de datos alfanuméricas con un atributo target. La sección 4.4 desarrolla un procedimiento para detectar ruido en bases de datos alfanuméricas sin un atributo target. Por último la sección 4.5 presenta una discusión general sobre las soluciones propuestas en este capítulo.

4.1 Consideraciones generales.

Se han desarrollado muchos algoritmos para detectar outliers en bases de datos, ninguno de ellos permite identificar en forma automática los campos que son considerados anómalos, todos los algoritmos existentes permiten detectar que fila o tupla es considerada anómala, pero no que campo especifico dentro de esa tupla es anómalo.

La evaluación de los algoritmos utilizados para la detección de outliers son validados por sus resultados empíricos, utilizándose en general el repositorio de la UCI (University of California - IRVINE) como una de las principales fuentes donde se obtienen los sets de datos. Demostrándose a través de numerosos trabajos publicados que un algoritmo funciona correctamente con un determinado set de datos y para determinado tipo de problema, el proceso de generalización se hace por lo menos muy cuestionable, en definitiva, el éxito de este proceso de generalización que se basa en pruebas empíricas se relaciona siempre con la selección del tipo de problema a resolver y el set de datos utilizado.

En esta tesis se propone la integración de un variado número de algoritmos de minería de datos, el objetivo es generar distintos procedimientos

que sean lo más eficiente y eficaces posible dada la naturaleza de la actividad para la cual se diseñaron, es decir detectar campos anómalos en bases de datos relacionadas con sistemas de gestión. En trabajos previos se determinó (Schaffer, 1994; Bishop, 1995; Smyth, 1994; Brodley & Friedl, 1996; Penny & Jolliffe, 2001; Deng & Mei, 2009) que un único algoritmo no es suficiente para obtener la calidad de resultados que exige una actividad como es la auditoría de sistemas. Este enfoque que implica la combinación e integración de diferentes algoritmos de minería de datos es utilizada en esta tesis dado que la integración de los mismos permite obtener un subconjunto de datos anómalos que implican una mayor seguridad en la calidad de los outliers detectados corrigiendo posibles errores que uno de los algoritmos pueda cometer ante escenarios tan diversos e inciertos.

Dado que el problema que se quiere resolver se relaciona con la dificultad que tienen los auditores de sistemas para detectar que campo específicamente de la base de datos tiene valores considerados anómalos, por lo tanto se seleccionaron algunas características del entorno a auditar donde se quiere desarrollar diferentes procedimientos de explotación de información para solucionar el problema planteado, las mismas se relacionan con las situaciones fácticas con las que los auditores de sistemas se enfrentan en su tarea cotidiana, estas características son:

- Detectar outliers en campos numéricos.
- Detectar outliers en campos alfanuméricos.
- Detectar outliers en bases de datos que poseen un atributo target.
- Detectar outliers en bases de datos que no poseen un atributo target.
- Reducir el espacio de búsqueda dentro de la base de datos para optimizar el funcionamiento del procedimiento.

En esta tesis se presentan 4 procedimientos para la detección de outliers, 2 orientados a datos numéricos y 2 procedimientos a alfanuméricos, que combinados permiten obtener los resultados esperados con un mayor nivel de calidad y seguridad. La métrica de calidad mínima para evaluar los

procedimientos se definió en el 60% de outliers detectados y menos del 10% de falsos positivos.

De acuerdo a lo propuesto por diversos trabajos (Penny & Jolliffe, 2001) se propone desarrollar procedimientos basados en una metodología hibrida (ver sección 2.4.3.6) implementando los enfoques propuestos por Hodge & Austin, (2004); Chandola et al., (2009), es decir el enfoque de detección de outliers con aprendizaje supervisado, no supervisado y semisupervisado (ver sección 2.4.1)

Se diseñaron entonces distintos tipos de procedimientos para distintos tipos de problemática con el objetivo de demostrar la necesidad de adaptar los procedimientos que se utilicen en la detección de outliers a dominios específicos:

- Procedimientos 1 y 2, el primero basado en los metadatos del set de datos considerado normal que fueron detectados por el algoritmo LOF, este procedimiento está basado en el enfoque semisupervisado; y el segundo procedimiento que combina los algoritmos LOF y K-means. La intersección de los outliers detectados por ambos procedimientos es utilizada en la detección de anomalías de los campos en bases de datos numéricas.
- Procedimiento 3 orientado a bases de datos alfanuméricas que contienen un atributo target, donde se combinan algoritmos de inducción, con LOF y con principios de la teoría de la información
- Procedimiento 4 orientado a bases de datos alfanuméricas sin atributo target, donde se combinan los algoritmos LOF, DBSCAN, redes bayesianas, C4.5 y PRISM, junto con K-means.

4.2 Procedimientos para la detección de outliers en bases de datos numéricas.

El procedimiento 1 basa la detección de outliers en la comparación de los metadatos de los campo considerados como normales con los metadatos de las tuplas consideradas por el algoritmo LOF como anómalas, en los casos

donde no hay coincidencia con las características de los metadatos considerados normales se los considera como campos outliers.

El procedimiento 2 reduce el espacio de búsqueda de valores atípicos eliminando las filas y columnas que se evalúan con valores normales y clusterizando cada columna del set de dato cuyos valores se consideran outliers, separándolos en dos grupos, aquel que es más lejano del centroide de este sub-set de datos permite aislar los campos anómalos.

Como se observa los criterios de ambos procedimientos para la obtención de datos anómalos son disimiles y su combinación garantiza mayor eficiencia y seguridad en los resultados obtenidos.

En la sub-sección 4.2.1 se presenta el procedimiento 1, en la 4.2.2 se desarrolla el procedimiento 2 y en la 4.2.3 se realiza la experimentación de ambos procedimientos.

4.2.1 Procedimiento 1.

En este proceso se aplica el algoritmo LOF con el fin de crear dos clusters, uno con datos limpios y el otro con los datos anómalos. Los datos que no son detectados por LOF como anómalos son utilizados para obtener sus metadatos, es decir las características de los mismos como su media, valores mínimos y máximos y tipo de datos.

El procedimiento compara los metadatos de cada atributo obtenidos de la base de datos considerada “limpia” con la base de datos cuyos valores de LOF permiten considerar a cada fila con sospecha de contener valores anómalos.

Este procedimiento integra el enfoque de tipo 3 semi-supervisado (ver sección 2.4.1) al detectar los metadatos de la base de datos considerada limpia, con el enfoque de tipo 1 no-supervisado que utiliza el algoritmo LOF, este algoritmo pertenece a los métodos relacionados con técnicas de clustering y está basado en la densidad (ver sección 2.4.3.4).

Este procedimiento tiene los siguientes pasos como se observa en la figura 4.1:

- *Entrada: base de datos*
    - *Aplicar el algoritmo LOF a una base de datos, como resultado de ese proceso se agrega una columna con el Factor de Outlier de cada tupla.*
        - *Identificar las tuplas con valor de LOF ≥ n y con valores de LOF <n (siendo n un valor a determinar experimentalmente), se considera que las tuplas con valores de LOF <n no contienen campos anómalos y las tuplas con valores de LOF ≥ n tienen alguno de sus campos que se consideran outliers.*
        - *Determinar los metadatos en las filas con valores de LOF < n, estos datos son: nombre del atributo, tipo de valor, y el rango (valores máximos y mínimos).*
            - *Desarrollar un script que realiza las siguientes funciones: recorre todas las filas y columnas y compara los metadatos identificados en el punto anterior con los campos de cada tupla que tienen valores de LOF ≥ n, si el valor del campo es mayor o menor que los valores "normales" se identifica ese campo como posible outlier.*
                - *Aplicar el script sobre las tuplas donde se sospecha que se trata de valores anómalos o sea donde el valor de LOF de la tupla representa un posible outlier (LOF ≥ n), como resultado se identifican los campos que posiblemente sean valores extremos, obteniéndose de esta manera el objetivo esperado.*
- *Salida: base de datos con los campos anómalos detectados*

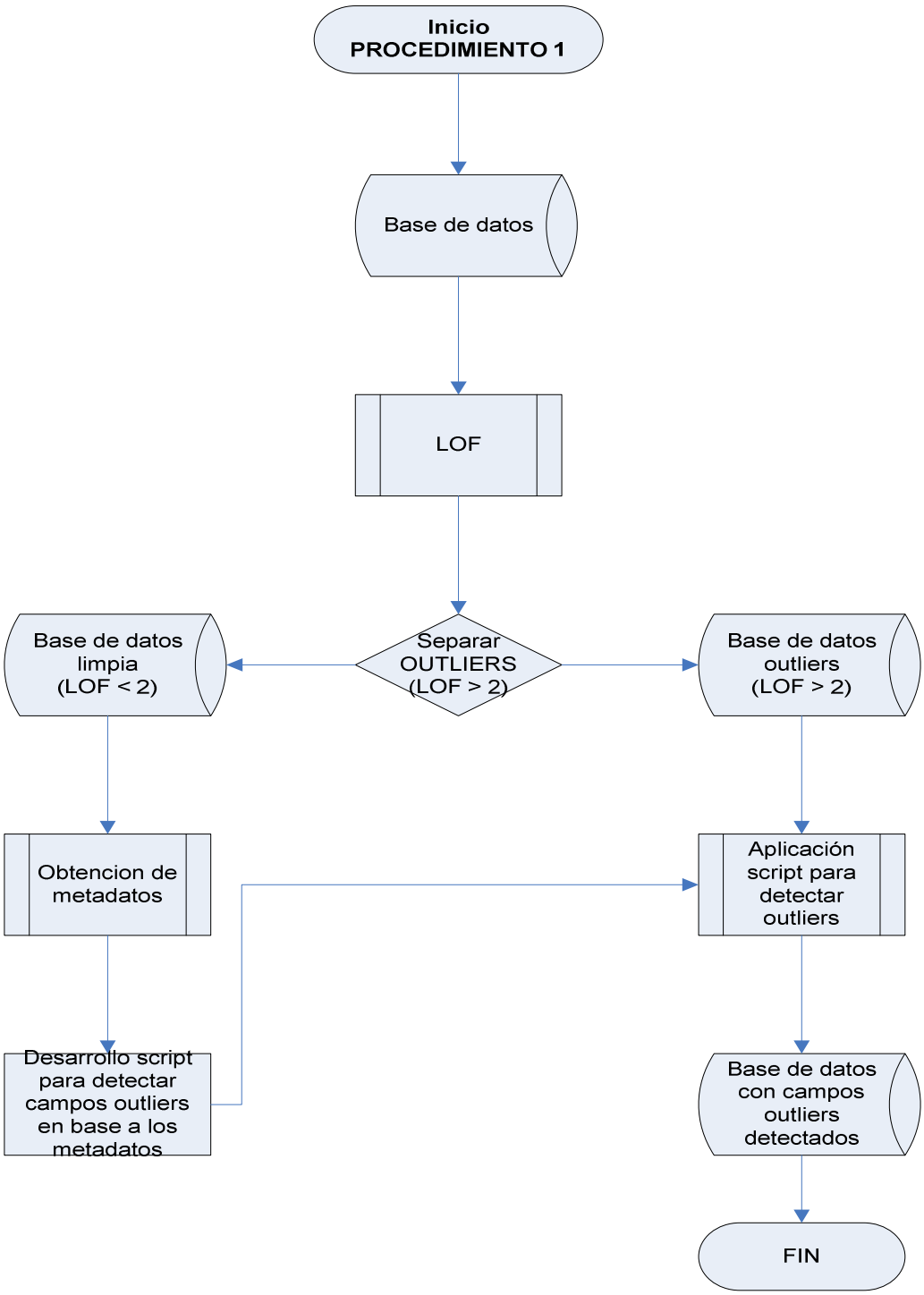


Figura. 4.1. Procedimiento 1 para BD numéricas

4.2.2 Procedimiento 2.

Este proceso aplica LOF en cada tupla y cada columna, se crea una nueva base de datos con sólo las filas que LOF evalúa que incluyen valores atípicos, el objetivo de este paso es hacer más eficiente el algoritmo al reducir el espacio de búsqueda, ya que se supone que debe ser aplicado a grandes bases de datos; a continuación se aplica el algoritmo K-means en cada columna de la nueva base de datos, con un valor de $K = 2$, se mide la distancia entre el centroide del set de datos y los centroides de cada clusters, evaluándose que el cluster más lejano es aquel que contiene cada campo de esa columna que se considera outlier.

El procedimiento 2 tiene un enfoque de tipo 1 con aprendizaje no supervisado y utiliza métodos de clustering basados en la densidad (LOF) y en particiones (K-means).

Este procedimiento tiene los siguientes pasos como se visualiza en la figura 4.2:

- *Entrada: base de datos*
    - *Aplicar LOF a cada fila de la BD (base de datos), como resultado de ese proceso se agrega una columna con el Factor de Outlier de cada tupla.*
    - *Invertir filas y columnas en la base de datos*
    - *Aplicar el algoritmo LOF a la base de datos invertida, como resultado de ese proceso se agrega columna una tupla con el Factor de Outlier de cada tupla de la base de datos invertida.*
    - *Volver a invertir filas y columnas para retornar al formato original de la base de datos*
        - *Seleccionar solo las filas cuyo valor de LOF es >n (siendo n un valor a determinar experimentalmente), y crear una nueva BD solo con tuplas que se considera que tienen atributos outliers (LOF >n), el objetivo de este paso es optimizar el funcionamiento del procedimiento en grandes*

bases de datos procesando solo las tuplas que tienen atributos considerados anómalos.

- *Crear una BD por cada columna cuyo valor de LOF es >n*
  - *Clusterizar la primer BD (primera columna) creada que tiene un valor de LOF >n con K-MEANS con K=2*
  - *Calcular la distancia entre los centroides de los clusters creados, el cluster que está más lejano del centroide es el que contiene los campos considerados outliers. De esta manera se logra identificar los atributos específicos de esa columna que tienen sospecha de ser considerados outliers.*
  - *Repetir el procedimiento para cada columna de la base de datos que tiene BD que contiene un valor de LOF >n (siendo n un valor a determinar experimentalmente) se considera a estas columnas como sospechosas de contener valores anómalos.*

- *Salida: base de datos con los campos anómalos detectados.*

4.2.3 Experimentación Procedimientos 1 y 2.

La experimentación tuvo dos pasos:

- El primero consistió en la aplicación de los procedimientos desarrollados inicialmente sobre un conjunto de datos creada en forma aleatoria considerando la distribución normal, dado que esta distribución permite desde el punto de vista estadístico determinar los valores atípicos. Esta prueba inicial se realizó para establecer los mejores valores de los parámetros de LOF.

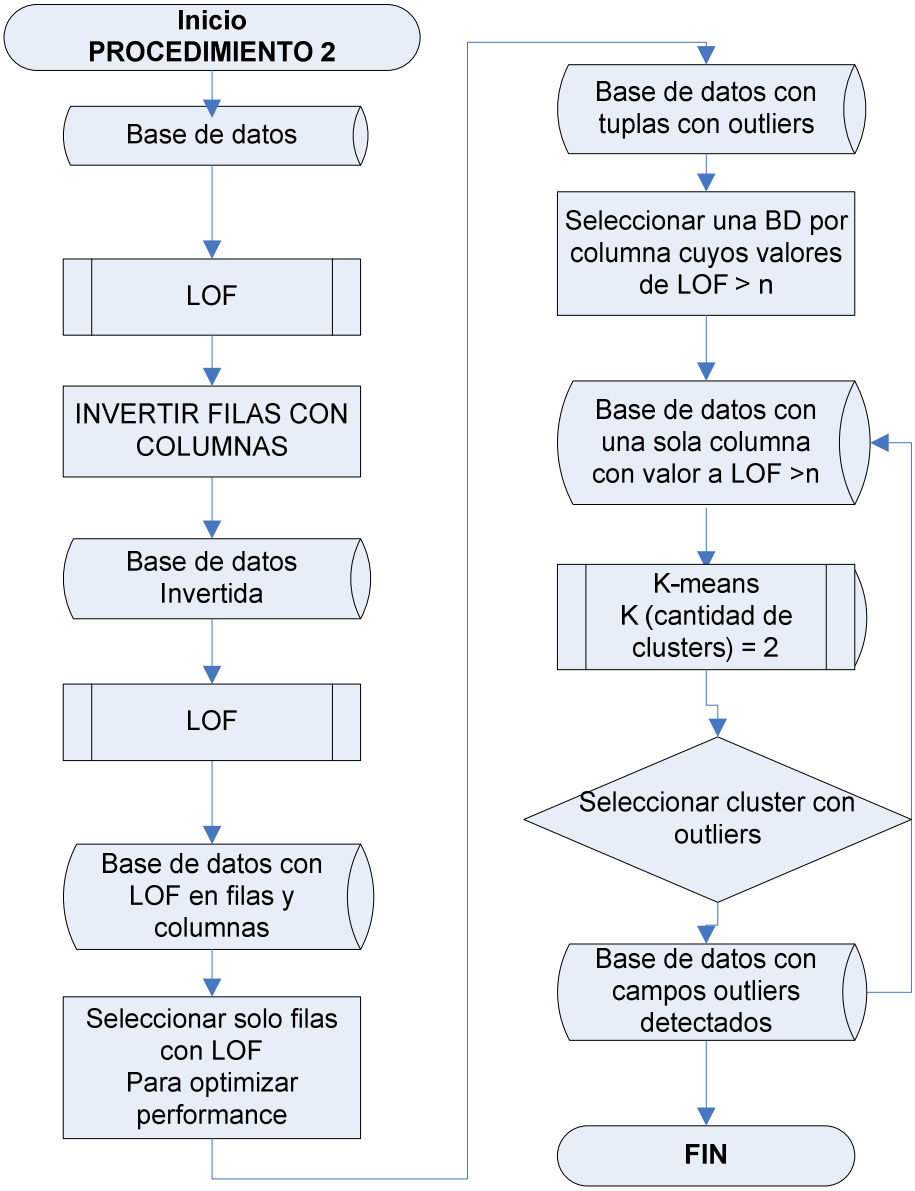


Figura 4.2. Procedimiento 2 para BD numéricas.

- El segundo paso en la prueba de los procedimientos con los valores óptimos de los parámetros de LOF obtenidos en el punto anterior, se realiza sobre una base de datos del mundo real. Esta estrategia es utilizada en repetidas oportunidades (Aggarwal & Philip, 2005; Johnson et al., 1998; Williams et al., 2002).

En el punto 4.2.3.1 se determinan los valores óptimos del algoritmo LOF y los de sus parámetros, en 4.2.3.2 se identifican los outliers aplicando métodos estadísticos, en 4.2.3.3 se realizan las pruebas sobre bases de datos

basadas en la distribución normal, en 4.2.3.4 se formaliza la experimentación sobre bases de datos reales y por último en 4.2.3.5 se discuten los resultados obtenidos al aplicar los procedimientos 1 y 2.

4.2.3.1 Determinación del valor Óptimo de LOF y sus parámetros.

Se utilizó el procedimiento 1 para determinar el valor optimo de LOF a partir del cual se debe considerar como outliers una tupla, así como los valores de los parámetro de dicho algoritmo, los mismos son:

- *MinPtsMin* – Debe ser un valor entero positivos únicamente.
- *MinPtsMax* – Debe ser un valor entero positivos únicamente.
- *Kind of distance* – valor por defecto: euclidean. Los otros tipos de medición disponibles son: squarted, cosine, inverted cosine y angle radiant.

Los 2 primeros parámetros son utilizados para definir el vecindario que el algoritmo formará alrededor de cada tupla a la hora de su análisis particular. El *MinPtsMin* es el límite de la cantidad mínima de tuplas con las cuales se debe realizar el cálculo del valor de LOF, mientras que el *MinPtsMax* marca el límite máximo de tuplas que se van a emplear para la misma tarea; por esto es que se dice que definen el 'vecindario' de tuplas contra las que cada una se va a comparar para determinar su valor de outlier.

Como se expreso en el punto anterior para realizar esta etapa de la experimentación se generó una base de datos con números aleatorios que responden a una distribución normal, definida como:

$$M = random(\text{‘}Normal\text{’},mu,sigma,m,n) \quad (13)$$

Donde

- *Normal* es el nombre de la distribución.
- *mu* es la media.
- *sigma* es la desviación estándar.
- *m* es la cantidad de registros que se van a generar.
- *n* es el número de columnas.

- M es una matriz *m-por-n*.

Con el set de datos creado se aplicó el procedimiento 1. Este proceso sirvió entonces para dos objetivos:

- Determinar los valores óptimos de LOF y sus parámetros
- Determinar los valores outliers y contrastarlos con los obtenidos a través del análisis estadístico de los datos.

Para realizar la experimentación se utilizo la distancia euclidea (Breunig et al., 2000), siendo su fórmula:

$$d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \qquad (14)$$

4.2.3.2 Determinación de outliers aplicando métodos estadísticos.

Con el objetivo de validar los resultados de la aplicación del procedimiento 1, se procedió a determinar por métodos estadísticos (Peña, 2002) los valores extremos o atípicos.

Sobre los distintos set de datos creados considerando la distribución normal se realizaron los siguientes pasos:

- Se calcula el valor medio del conjunto de datos (columnas, o campos de los registros).
  Para lograr este objetivo inicialmente se calculó la media aritmética, donde dados los *n* números $\{x_1 , x_2 , x_3 , \ldots . , x_n\}$, la media aritmética se define como:

$$m = \frac{\sum_{i=1}^{n} x_i}{n} \qquad (15)$$

- Se calcula el desvío estándar de este conjunto de datos.
  El desvío estándar muestral se calculó según la siguiente fórmula:

$$s = \sqrt{\frac{\sum_{i=1}^{n}\left(x_i - \bar{x}\right)^2}{n-1}} \quad (16)$$

Donde x_i es cada uno de los elementos de la muestra y n es el tamaño de la muestra.

Donde x es la media de muestra o promedio *(número1; número2;…n)* y n es el tamaño de la muestra.

Se determinan los rangos de valores máximos y mínimos aceptables sumando y restando el doble del valor del desvío estándar al valor de la media calculada según lo establecido en el área de estadística. En este paso se identifican los valores atípicos, que son aquellos que están fuera del rango establecido en el punto anterior, la figura 2.3 de la sección 2.4 muestra el criterio que se adoptó para definir outliers en una distribución normal.

Cabe destacar que la finalidad del método estadístico es determinar los outliers para luego compararlos con los obtenidos a través de la ejecución del procedimiento 1 y de esta manera evaluar la eficacia del procedimiento propuesto.

4.2.3.3 Pruebas realizadas sobre Bases de Datos basadas en la distribución normal.

Se crearon tres BD (bases de datos) con la distribución normal, un set de datos con 200 tuplas, otro con 400 y un último set de datos de 2000 registros.

Como ejemplo en la figura 4.3 se muestra el histograma de la base de datos con 2000 registros.

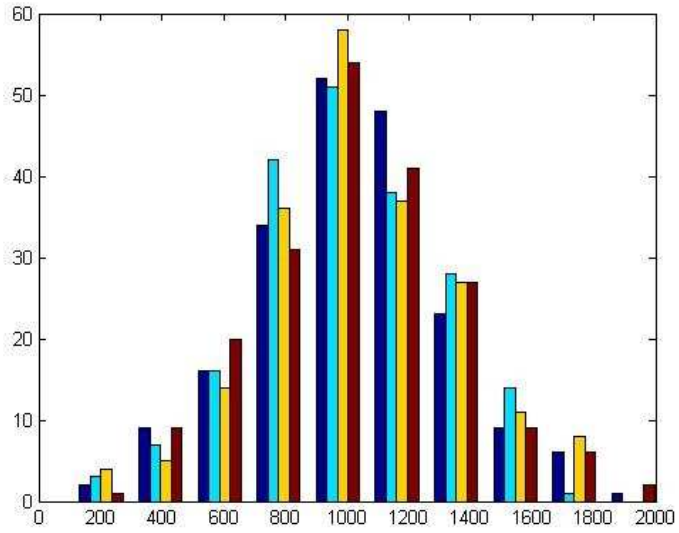


Figura 4.3. Histograma base de Base 2000 registros

Se realizaron pruebas para cada una de las bases de datos primero utilizando el análisis estadístico, para determinar los outliers y luego se aplicó el procedimiento 1 propuesto.

Sobre las 3 bases de datos creadas de acuerdo a la distribución normal se testearon diferentes valores de las variables *MinPtsMin* y *MinPtsMax* y limites de LOF tomando como criterios de validación y efectividad los casos que mayor acierto en el descubrimientos de outliers y menores casos de falsos positivos tenía.

Los valores de las variables utilizadas en cada prueba fueron:

- Valores límites de LOF para considerar a la tupla como outlier: 1.3, 1.5, 1.7, 1.9 y 2.
- *MinPtsMin*: 1, 5, 10, 20 y 50.
- *MinPtsMax*: 2, 10, 15, 20, 40 y 100
- Sets de datos: 200, 400 y 2000 registros.

Los mejores resultados se obtuvieron con valores siguientes:

Límite de LOF = 1.5

***MinPtsMin* = 10**

***MinPtsMax* = 20**

La tabla 4.1 presenta los resultados obtenidos con los diferentes grupos de datos (200, 400 y 2000 registros). Estos valores se tomaron como parámetros para realizar las pruebas con el procedimiento diseñado que incluye aplicar clustering sobre la base de datos de cáncer de mama.

La efectividad se calculó teniendo en cuenta el valor de aciertos (valor porcentual de outliers detectados sobre el total existente) menos el valor de yerro (valor porcentual de los falsos positivos sobre el total existente). Tomando como mejor efectividad los valores más altos positivos y como una mala efectividad los valores más bajos negativos. Siendo en definitiva:

efectividad = porcentaje de outliers detectados – porcentaje falsos positivos

200 rows

LOF limit value:	1,5					
MinPtsMin value:	1	5	10	5	20	50
MinPtsMax value:	2	15	20	10	40	100
Total outlier quantity:	35	35	35	35	35	35
Outliers detected in the procedure:	12	35	26	26	0	0
False positives:	25	165	16	16	0	0
Effectiveness:	-37,14	-371,43	28,57	28,57	0	0

400 rows

LOF limit value:						1,5
MinPtsMin value:	1	5	10	5	20	50
MinPtsMax value:	2	15	20	10	40	100
Total outlier quantity:	65	65	65	65	65	65
Outliers detected in the procedure:	17	65	40	65	0	0
False positives:	53	335	25	335	0	0
Effectiveness:	-55,38	-415,38	23,08	-415,38	0	0

2000 rows

LOF limit value:						1,5
MinPtsMin value:	1	5	10	5	20	50
MinPtsMax value:	2	15	20	10	40	100
Total outlier quantity:	350	350	350	350	350	350
Outliers detected in the procedure:	90	349	135	349	1	0
False positives:	222	1642	59	1642	0	0
Effectiveness:	-37,71	-369,43	21,71	-369,43	0,29	0

Tabla 4.1. Valores Obtenidos en las pruebas realizadas aplicando el procedimiento propuesto sobre la BD generada, con valores de Límite de LOF igual 1.5

Para obtener estos valores óptimos se ejecutaron sucesivas pruebas determinando la efectividad obtenida según el criterio mencionado anteriormente. Los datos para los diferentes valores *MinPtsMin*, *MinPtsMax* y límite de LOF para 200 registros se observa en la tabla 4.2.

BD DE 200 Reg						
LOF limit value:	1,3					
MinPtsMin value:	1	5	10	5	20	50
MinPtsMax value:	2	15	20	10	40	100
Total outlier quantity:	35	35	35	35	35	35
Outliers detected in the procedure:	17	35	34	35	0	0
False positives:	38	165	38	165	0	0
Effectiveness:	-60	-371,428571	-11,4285714	-371,428571	0	0
LOF limit value:	1,5					
MinPtsMin value:	1	5	10	5	20	50
MinPtsMax value:	2	15	20	10	40	100
Total outlier quantity:	35	35	35	35	35	35
Outliers detected in the procedure:	12	35	26	26	0	0
False positives:	25	165	16	16	0	0
Effectiveness:	-37,1428571	-371,428571	28,5714286	28,5714286	0	0
LOF limit value:	1,7					
MinPtsMin value:	1	5	10	5	20	50
MinPtsMax value:	2	15	20	10	40	100
Total outlier quantity:	35	35	35	35	35	35
Outliers detected in the procedure:	9	35	13	35	0	0
False positives:	16	150	8	150	0	0
Effectiveness:	-20	-328,571429	14,2857143	-328,571429	0	0
LOF limit value:	1,9					
MinPtsMin value:	1	5	10	5	20	50
MinPtsMax value:	2	15	20	10	40	100
Total outlier quantity:	35	35	35	35	35	35
Outliers detected in the procedure:	8	35	11	35	0	0
False positives:	0	123	0	150	0	0
Effectiveness:	22,8571429	-251,428571	31,4285714	-328,571429	0	0
LOF limit value:	2					
MinPtsMin value:	1	5	10	5	20	50
MinPtsMax value:	2	15	20	10	40	100
Total outlier quantity:	35	35	35	35	35	35
Outliers detected in the procedure:	6	35	8	35	0	0
False positives:	12	108	0	108	0	0
Effectiveness:	-17,1428571	-208,571429	22,8571429	-208,571429	0	0

Tabla 4.2. Valores Obtenidos en las pruebas realizadas aplicando el procedimiento propuesto sobre la BD generada, con 200 registros variando los valores de *MinPtsMin*, *MinPtsMax* y LOF

Los datos para los diferentes valores *MinPtsMin*, *MinPtsMax* y límite de LOF para 400 registros se observa en la tabla 4.3.

BD DE 400 Reg						
LOF limit value:						
MinPtsMin value:	1	5	10	5	20	50
MinPtsMax value:	2	15	20	10	40	100
Total outlier quantity:	65	65	65	65	65	65
Outliers detected in the procedure:	21	65	55	65	0	0
False positives:	87	335	82	335	0	0
Effectiveness:	-101,538462	-415,384615	-41,5384615	-415,384615	0	0

LOF limit value:						
MinPtsMin value:	1	5	10	5	20	50
MinPtsMax value:	2	15	20	10	40	100
Total outlier quantity:	65	65	65	65	65	65
Outliers detected in the procedure:	17	65	40	65	0	0
False positives:	53	335	25	335	0	0
Effectiveness:	-55,3846154	-415,384615	23,0769231	-415,384615	0	0

LOF limit value:						
MinPtsMin value:	1	5	10	5	20	50
MinPtsMax value:	2	15	20	10	40	100
Total outlier quantity:	65	65	65	65	65	65
Outliers detected in the procedure:	12	65	26	65	0	0
False positives:	33	335	13	321	0	0
Effectiveness:	-32,3076923	-415,384615	20	-393,846154	0	0

LOF limit value:						
MinPtsMin value:	1	5	10	5	20	50
MinPtsMax value:	2	15	20	10	40	100
Total outlier quantity:	65	65	65	65	65	65
Outliers detected in the procedure:	6	64	15	64	0	0
False positives:	21	263	4	263	0	0
Effectiveness:	-23,0769231	-306,153846	16,9230769	-306,153846	0	0

LOF limit value:						
MinPtsMin value:	1	5	10	5	20	50
MinPtsMax value:	2	15	20	10	40	100
Total outlier quantity:	65	65	65	65	65	65
Outliers detected in the procedure:	6	65	12	65	0	0
False positives:	18	262	1	227	0	0
Effectiveness:	-18,4615385	-303,076923	16,9230769	-249,230769	0	0

Tabla 4.3. Valores obtenidos en las pruebas realizadas aplicando el procedimiento propuesto sobre la BD generada, con 400 registros variando los valores de *MinPtsMin*, *MinPtsMax* y LOF

Los datos para los diferentes valores *MinPtsMin*, *MinPtsMax* y límite de LOF para 2000 registros se observa en la tabla 4.4.

BD DE 2000 Reg						
LOF limit value:						
MinPtsMin value:	1	5	10	5	20	50
MinPtsMax value:	2	15	20	10	40	100
Total outlier quantity:	350	350	350	350	350	350
Outliers detected in the procedure:	121	350	219	350	4	0
False positives:	351	1648	227	1648	0	0
Effectiveness:	-65,7142857	-370,857143	-2,28571429	-370,857143	1,14285714	0

LOF limit value:						
MinPtsMin value:	1	5	10	5	20	50
MinPtsMax value:	2	15	20	10	40	100
Total outlier quantity:	350	350	350	350	350	350
Outliers detected in the procedure:	90	349	135	349	1	0
False positives:	222	1642	59	1642	0	0
Effectiveness:	-37,7142857	-369,428571	21,7142857	-369,428571	0,28571429	0

LOF limit value:						
MinPtsMin value:	1	5	10	5	20	50
MinPtsMax value:	2	15	20	10	40	100
Total outlier quantity:	350	350	350	350	350	350
Outliers detected in the procedure:	61	347	87	347	1	0
False positives:	131	1550	12	1550	0	0
Effectiveness:	-20	-343,714286	21,4285714	-343,714286	0,28571429	0

LOF limit value:						
MinPtsMin value:	1	5	10	5	20	50
MinPtsMax value:	2	15	20	10	40	100
Total outlier quantity:	350	350	350	350	350	350
Outliers detected in the procedure:	39	332	46	332	0	0
False positives:	89	1254	2	1254	0	0
Effectiveness:	-14,2857143	-263,428571	12,5714286	-263,428571	0	0

LOF limit value:						
MinPtsMin value:	1	5	10	5	20	50
MinPtsMax value:	2	15	20	10	40	100
Total outlier quantity:	350	350	350	350	350	350
Outliers detected in the procedure:	34	311	35	311	0	0
False positives:	73	1041	1	1041	0	0
Effectiveness:	-11,1428571	-208,571429	9,71428571	-208,571429	0	0

Tabla 4.4. Valores Obtenidos en las pruebas realizadas aplicando el procedimiento propuesto sobre la BD generada, con 2000 registros variando los valores de MinPtsMin, MinPtsMax y LOF

De esta manera se puede observar como los valores óptimos para los parámetros se encuentran con *límite de LOF* = 1,5 *MinPtsMin* = 10 y *MinPtsMax* = 20.

4.2.3.4 Experimentación en una base de datos real de los procedimientos 1 y 2.

A partir de los resultados obtenidos en la base de datos creada de acuerdo a la distribución normal y obtenidos los valores óptimos LOF, *MinPtsMin* y *MinPtsMax,* se realizó la experimentación sobre un set de datos real.

Existen dificultades a la hora de seleccionar una base de datos real ya que es muy complejo determinar a priori cuales son las tuplas consideradas anómalas. En el estudio realizado en *"A New Local Distance-Based Outlier Detection Approach for Scattered Real-World Data"* [58] (Zhang et al., 2009) se utiliza la base de datos de cáncer de mama de *Wisconsin Diagnostic Breast Cancer* para detectar tuplas outliers. En esta base de datos aparecen clasificados tipos de cáncer de mamas con diferentes características que responden a casos de cáncer maligno y benigno. Esta base de datos tiene 569 registros con 30 atributos, 357 benignos y 212 malignos.

Tomando como referencia este estudio, el cual considera el set de datos que indican cáncer benigno como datos "normales" y el conjunto de datos que representan casos de cáncer malignos como los datos "anómalos", se separaron los datos correspondientes a cáncer maligno, y con el objetivo de optimizar el funcionamiento del procedimiento se tomaron 10 de las 212 tuplas con solamente con el atributo "*tipo de cáncer" = "maligno"*.

Este set de datos permite identificar entonces de manera univoca las tuplas consideradas outliers, no así específicamente que campos son los que tienen valores anómalos, para eso entonces se ejecutaron los procedimientos desarrollados.

En la sub-sección 4.2.3.4.1 se realiza la experimentación con el procedimiento 1 y en 4.2.3.4.2 se formaliza la experimentación sobre el procedimiento 2.

[58] WDBC Dataset is UCI ML Repository: http:// archive.ics.uci.edu/ml

4.2.3.4.1 Experimentación con el procedimiento 1.

Tomando el valor de *LOF = 1.5* y los valores de *MinPtsMin* = 10 y *MinPtsMax* = 20, se ejecutó el procedimiento1, la tabla 4.5 muestra el numero de tupla y los valores de LOF mayores a 1.5, estos resultados coinciden en forma exacta con las 10 tuplas con el atributo *"tipo de cáncer"= "maligno"*

id	LOF
id_20	5,2557
id_23	5,6504
id_30	4,9437
id_41	6,0198
id_83	4,5806
id_174	6,3284
id_217	5,1842
id_298	3,6028
id_303	6,4412
id_313	8.6087

Tabla 4.5. Numero de tupla de la Base de Datos con valores de LOF > 1.5

Se detectaron 16 columnas y 118 campos con valores anómalos, la tabla 4.6 muestra los campos detectados como outliers después de ejecutar el procedimiento 1.

Colum 1	*Colum 3*	*Colum 4*	*Colum 6*	*Colum 7*	*Colum 8*	*Colum 11*	*Colum 12*	*Colum 13*
23,51	155,1	1747	0,1283	0,2308	0,141	1009	0,9245	6462
28,11	188,5	2499	0,1516	0,3201	0,1595	2873	1476	21,98
23,27	152,1	1686	0,1145	0,1324	0,097	0,6642	0,8561	4603
27,42	186,9	2501	0,1988	0,3635	0,1689	2547	1306	18,65
27,22	182,11	2250	0,1914	0,2871	0,1878	0,8361	1481	5,82
25,73	174,2	2010	0,2363	0,3368	0,1913	0,9948	0,8509	7222
23,21	153,5	1670	0,1682	0,195	0,1237	1058	0,9635	7247
24,63	165,5	1841	0,2106	0,231	0,1471	0,9915	0,9004	7,05
25,22	171,5	1878	0,2665	0,3339	0,1845	0,8973	1474	7382
24,25	166,2	1761	0,2867	0,4268	0,2012	1509	3,12	9807
Colum 14	***Colum 15***	***Colum 21***	***Colum 23***	***Colum 24***	***Colum 26***	***Colum 28***	***id***	
164,1	0,0063	30,67	202,4	2906	0,2678	0,2089	Id_20	
525,6	0,0135	28,11	188,5	2499	0,1516	0,1595	Id_23	
97,85	0,0049	28,01	184,2	2403	0,3583	0,2346	Id_30	
542,2	0,0077	36,04	251,2	4254	0,4256	0,2625	Id_41	
128,7	0,0046	33,12	220,8	3216	0,4034	0,2688	Id_83	
153,1	0,0064	33,13	229,3	3234	0,5937	0,2756	Id_174	
155,8	0,0064	31,01	206	2944	0,4126	0,2593	Id_217	
139,9	0,005	29,92	205,7	2642	0,4188	0,2475	Id_298	
120	0,0082	30	211,7	2562	0,6076	0,2867	Id_303	
233	0,0233	26,02	180,9	2073	0,4244	0,2248	Id_313	

Tabla 4.6. Outliers detectados por el procedimiento 1

4.2.3.4.2 Experimentación con el procedimiento 2.

Tomando el valor de $LOF = 1.5$ y los valores de $MinPtsMin = 10$ y $MinPtsMax = 20$, se ejecutó el procedimiento 2, el set de datos se redujo a 10 filas y 14 columnas como resultado de aplicar el algoritmo LOF, después de clusterizar con K-means cada columna con $k = 2$ y al medir las distancias entre centroides se obtuvieron 2 clusters por cada columna, en la figura 4.4 se observa un ejemplo de la clusterización de la columna 12.

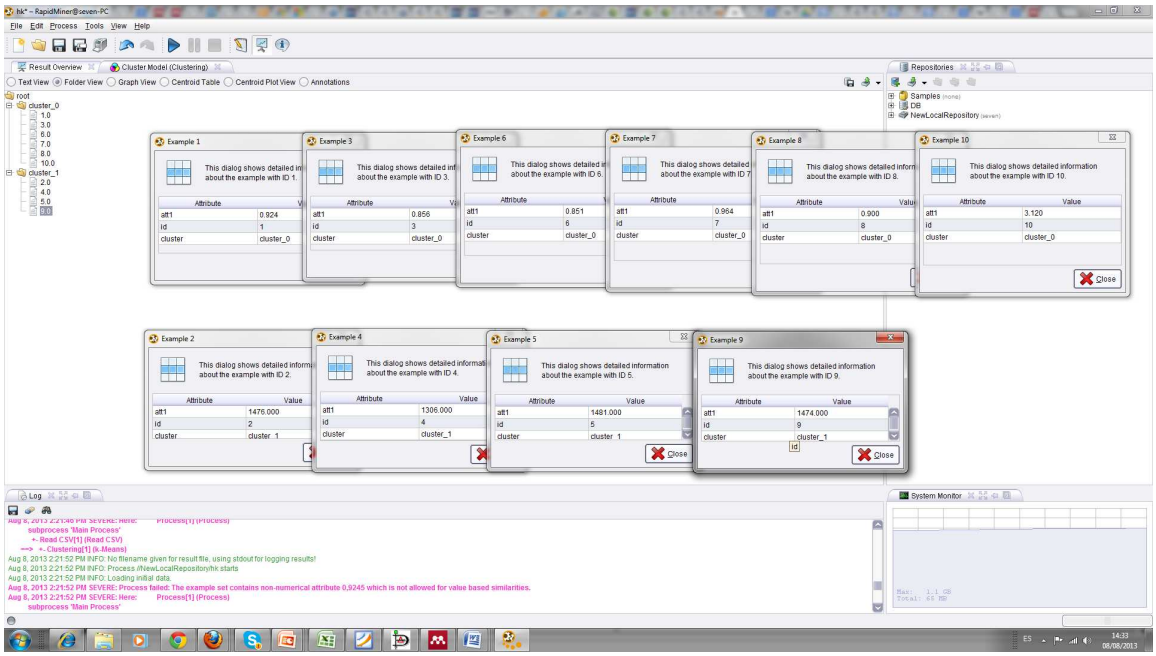


Figura 4.4. Ejemplo de clusterización de la columna 12

La figura 4.5 muestra en el caso de la columna 12 las distancias de cada cluster al centroide del set de datos, observándose claramente que el cluster 1 es el más lejano, por lo tanto es el que contiene los campos los campos sospechosos de ser outliers.

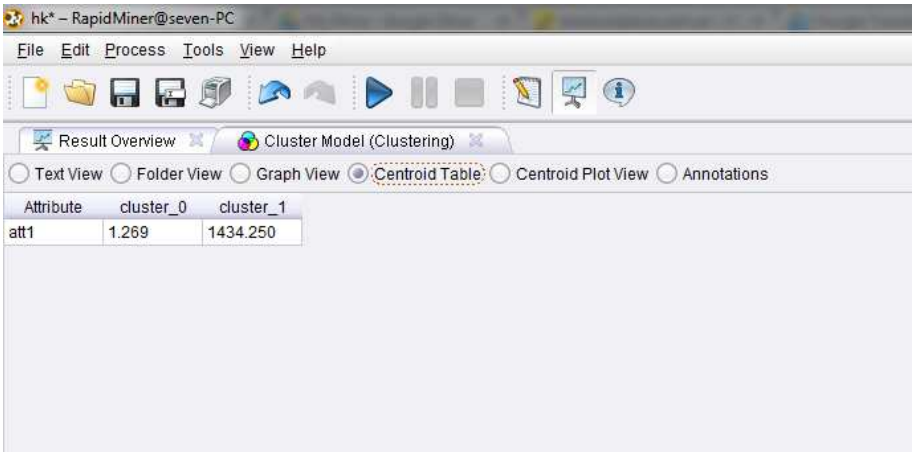


Figura 4.5. Distancia del centroide de la columna 12.

La Tabla 4.7 contiene un resumen por columna de los resultados obtenidos.

Numero de Columna	Cantidad de elementos Cluster 0 (limpios)	Cantidad de elementos Cluster 1 (outliers)
1	0	10
2	0	10
3	0	10
4	0	10
11	6	4
12	6	4
13	4	6
14	8	2
21	0	10
22	0	10
23	0	10
24	0	10
26	8	2
28	1	9

Tabla 4.7. Resumen outliers por cluster 2

La tabla 4.8 muestra resaltados los 107 campos detectados como outliers.

Colum 1	*Colum 2*	*Colum 3*	*Colum 4*	*Colum 11*	*Colum 12*	*Colum 13*
23,51	24,27	155,1	1747	1009	0,9245	6462
28,11	18,47	188,5	2499	2873	1476	21,98
23,27	22,04	152,1	1686	0,6642	0,8561	4606
27,42	26,27	186,9	2501	2547	1306	18,65
27,22	21,87	182,1	2250	0,8361	1481	5,82
25,73	17,46	174,2	2010	0,9948	0,8509	7222
23,21	26,97	153,5	1670	1058	0,9635	7247
24,63	21,6	165,5	1841	0,9915	0,9004	7,05
25,22	24,91	171,5	1878	0,8973	1474	7382
24,25	20,2	166,2	1761	1509	3,12	9807
Colum 14	***Colum 21***	***Colum 22***	***Colum 23***	***Colum 24***	***Colum 26***	***Colum 28***
164,1	30,67	30,73	202,4	2906	0,2678	0,2089
525,6	28,11	18,47	188,5	2499	0,1516	0,1595
97,85	28,01	28,22	184,2	2403	0,3583	0,2346
542,2	36,04	31,37	251,2	4254	0,4256	0,2625
128,7	33,12	32,85	220,8	3216	0,4034	0,2688
153,1	33,13	23,58	229,3	3234	0,5937	0,2756
155,8	31,01	34,51	206	2944	0,4126	0,2593
139,9	29,92	26,93	205,7	2642	0,4188	0,2475
120	30	33,62	211,7	2562	0,6076	0,2867
233	26,02	23,99	180,9	2073	0,4244	0,2248

Tabla 4.8. Outliers detectados por el procedimiento 2

4.2.3.5 Resultados y discusión de la aplicación de los procedimientos 1 y 2 para BD numéricas.

La ejecución de los procedimientos 1 y 2 dio como resultado un nuevo set de datos con sospecha de ser datos anómalos, se establece un criterio de certeza de datos outliers en la base de datos relacionado con la intersección del conjunto de datos detectados por el procedimiento 1 y 2, como lo muestra la figura 4.6. Se consideró que la intersección del conjunto de datos detectados como anómalos por cada uno de los procedimientos diseñados permite obtener un subconjunto de datos que al ser detectados como outliers por ambos procedimientos brindan un nivel de confianza mayor potenciando la calidad del proceso de detección realizado. Los datos que no están incluidos en esa intersección y que fueron detectados por uno u otro procedimiento requerirán un análisis posterior para determinar su calidad.

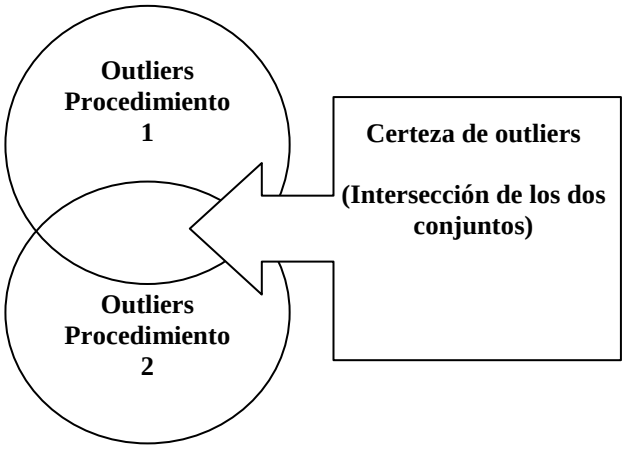


Figura 4.6. Criterio de certeza

Del análisis del resultado de la ejecución de los 2 procedimientos surgen los siguientes resultados:

- 83 campos fueron detectados en forma coincidente por los dos procedimientos.
- 35 campos fueron detectados por el procedimiento 1 y no fueron detectados por el procedimiento 2.
- 20 campos fueron detectados por el procedimiento 2 y no fueron detectados por el procedimiento 1.
- La aplicación de ambos procedimientos no implicó la detección de falsos positivos.
- Del total de posibles outliers detectados por los procedimientos 1 y 2 hay coincidencia en un 60,15%, considerándose este porcentaje representa el conjunto de campos donde hay una sería sospecha que se tratan de campos anómalos.

En la figura 4.7 se observa un resumen de los outliers detectados.

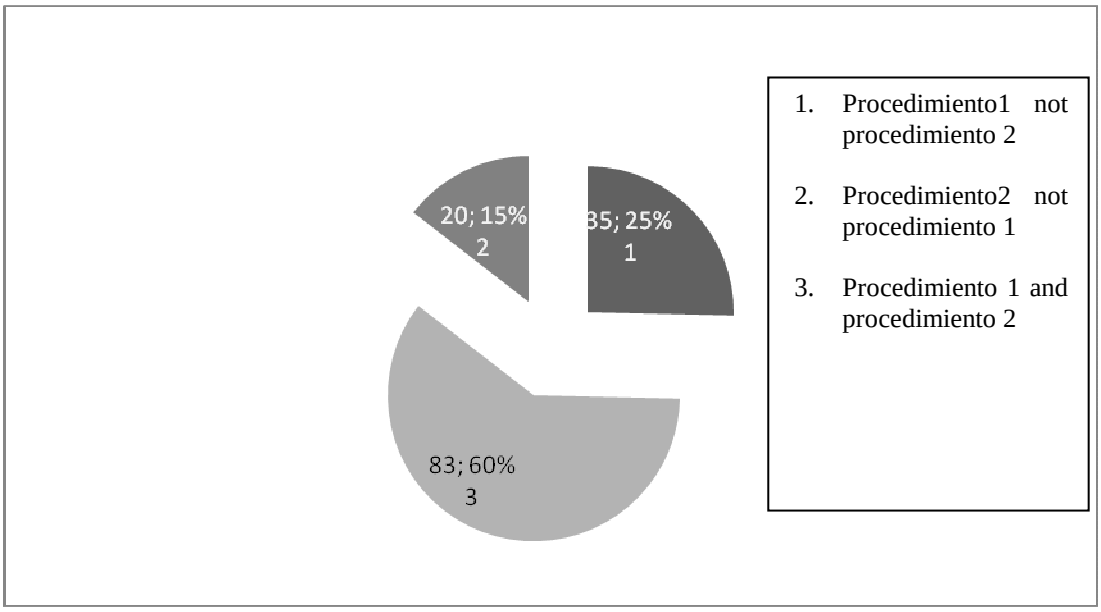


Figura 4.7. Resumen de outliers detectado

4.3 Procedimiento 3. Orientado a bases de datos alfanuméricas con un atributo target.

El procedimiento propuesto tiene como objetivo detectar campos considerados outliers en bases de datos alfanuméricas que contienen un atributo target, es de destacar que no en todos los casos se cuenta con un atributo de estas características.

El procedimiento implementa inicialmente un algoritmos de inducción considerando al atributo target u objetivo definido previamente, aplicar este tipo de algoritmos tiene el objetivo de reducir el set de datos a aquellos atributos que son los más representativos de la muestra y de esta manera hacer más eficiente el procedimiento. Una vez identificados el conjunto de atributos representativos del set de datos, teniendo en cuenta los conceptos de teoría de información (ver sección 2.4.3.6), se forman pares entradas salidas (E+S) con cada atributo y el atributo target identificado. Para evitar inconvenientes con los valores nulos son reemplazados por una etiqueta “nulo”, y se aplica LOF para identificar a aquellos elementos que se encuentran en regiones de baja

densidad para considerarlos como sospechosos de contener valores anómalos. El procedimiento 3 tiene un enfoque de tipo 1 con aprendizaje no supervisado al utilizar la teoría de la información y un algoritmo de clusterización basado en la densidad (LOF), este procedimiento híbrido también tiene un enfoque de tipo 2 al utilizar algoritmos relacionados con el aprendizaje automático (C4.5).

A continuación se detallan cada una de las etapas que involucran al procedimiento descripto anteriormente, los mismos pueden verse en el diagrama de la figura 4.8.

- *Entrada: base de datos*
  - *Determinar atributo target*
  - *Aplicar algoritmo C4.5*
  - *Para cada atributo significativo realizar lo siguiente*
    - *arma un binconjunto de datos (E)+(S) como se explico anteriormente, tomando como entrada (E) el atributo seleccionado y como salida (S) el atributo target u objetivo.*
    - *Los datos que componen al atributo entrada son analizados en busca de valores nulos y son reemplazados por una etiqueta para su posterior proceso.*
    - *se aplica LOF al bin (E)+(S) dando como resultado la generación de un atributo outlier*
  - *Se filtran aquellos pares de datos bin (E)-(S) cuyo valor de outlier, al ser distinto de cero (0), nos indica la presencia de un dato anómalo que no aporta información y corresponde a ruido en referencia al target definido.*
- *Salida: base de datos con los campos anómalos detectados*

En la sub-sección 4.3.1 se formaliza la experimentación del procedimiento 3, en 4.3.2 se analizan los resultados obtenidos y en 4.3.3 se realiza una discusión sobre el procedimiento 3.

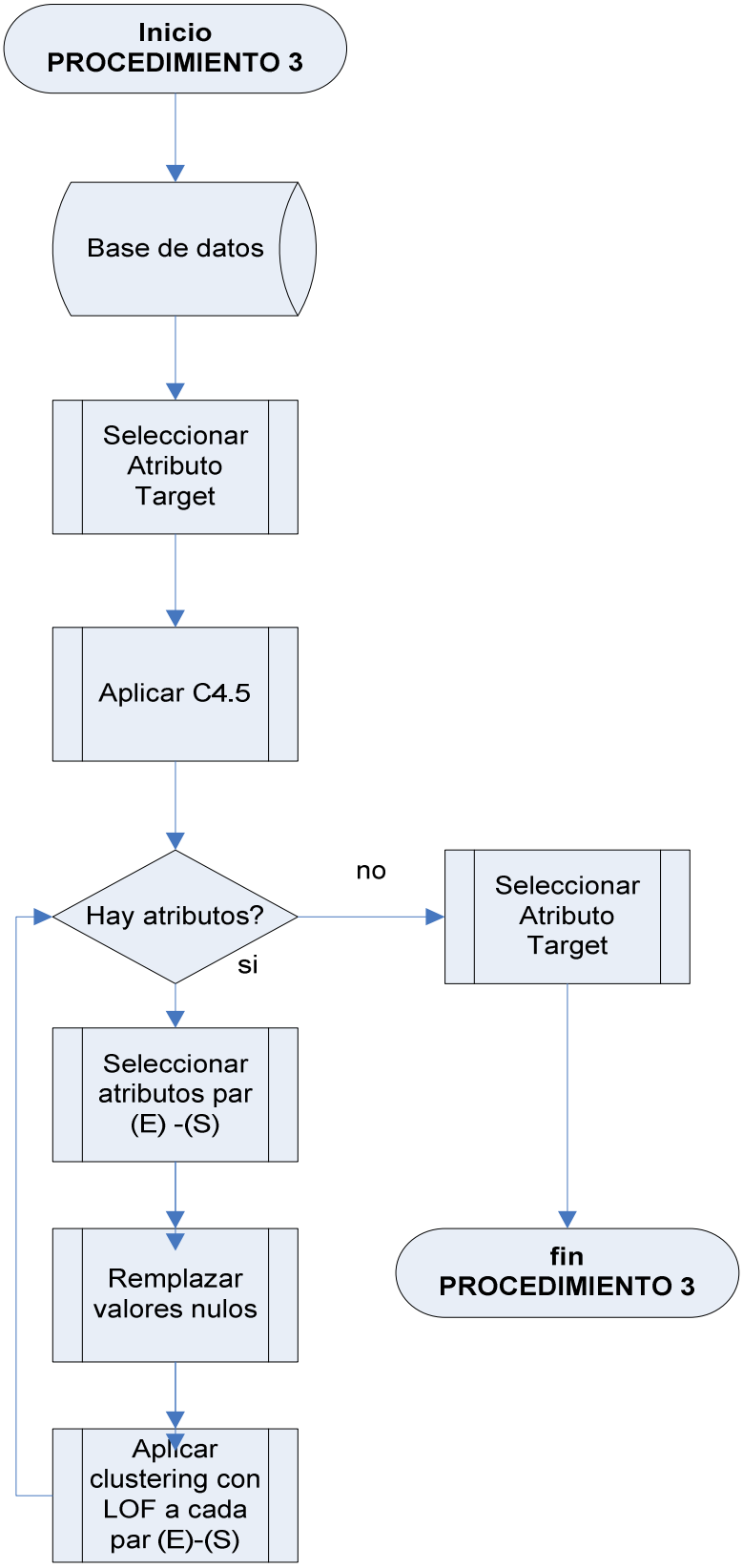


Figura 4.8. Procedimiento para la detección de outliers en bases de datos alfanuméricas

4.3.1 Experimentación.

La experimentación se realiza sobre una base de datos de hongos obtenida del repositorio "Machine Learning Repository" de la UCI (University of California - IRVINE) (Frank & Asuncion, 2012). Es una base de datos con valores nominales que incluye un atributo objetivo o target, que es el campo: *"clase"*, que clasifica a los hongos como venenosos o comestibles. La base de datos tiene 23 atributos y 8124 tuplas. En la tabla 4.9 se presentan los atributos de la base de datos.

Forma_sombrero	*Superficie_sombrero*	*Superficie_sombrero*	*Magulladuras*
Olor	*Tipo_membrana*	*Espaciado_ membrana*	*Tamaño_mem brana*
Color_membrana	*Forma_tronco*	*Raiz_tronco*	*Sup_tronco_arriba_ani llo*
Sup_tronco_debajo_ani llo	*ColorTronco_arriba_an illo*	*ColorTronco_debajo_an illo*	*Tipo_velo*
Color_velo	*Cantidad_anillos*	*Tipo_anillo*	*Color_esporas*
Poblacion	*Habitat*	***Clase (Target)***	

Tabla 4.9. Atributos de la Base de Datos de Hongos

Cuando se realiza la experimentación con bases de datos reales que no responden a un tipo de distribución específico, y que por lo tanto no es posible identificar por métodos estadísticos los valores anómalos, como es en la mayoría de los casos cuando se trabaja con sets de datos relacionados con sistemas de gestión, la identificación previa de los campos considerados anómalos se dificulta, por este motivo con el objetivo de verificar la calidad de los resultados obtenidos después de aplicar el procedimiento propuesto se analizó en forma manual la base de datos con un experto en hongos comestibles, quien detecto los posibles outliers existentes en la base de datos, el experto identificó un total de 59 campos sospechosos de ser anómalos, este conjunto de datos fue utilizado entonces para validar el procedimiento.

En el presente caso de estudio se ha remplazado en la entrada (E) los valores nulos por la etiqueta *"nulo"*, de esta manera se mantiene la

característica nominal de los datos, en el caso del algoritmo LOF los valores de configuración para *MinPts* se definen de acuerdo a los valores obtenidos a partir de la experimentación realizada con el procedimiento 1, los mismos son: límite inferior en 10 y límite superior en 20 y el cálculo de la función de la distancia que se utiliza para el cálculo de la distancia entre dos objetos es la distancia euclidiana.

Al ejecutar el flujo de minería, LOF detecta los outliers presentes en el atributo seleccionado como (E) incorporando el atributo outlier calculado y colocándole el valor ∞ (infinito) a aquellos que él algoritmo detecta como anómalos.

Como se mencionó anteriormente cuanto menor sea la probabilidad de aparición del par (E) – (S) analizado mayor es la posibilidad de que corresponda a una inconsistencia.

Se forman los "bin" (Ferreyra, 2007) con los atributos de entrada (E) y el atributo target (S) y el cálculo de LOF representa la relación existente entre (E) y el elemento (S). En la teoría la información cuando Shannon (Shannon, 2001) hace referencia a la cantidad de información que aporta el elemento, dicha cantidad de información se relaciona con la probabilidad de que ese elemento aparezca en la salida, siendo (E) es el mensaje emitido y (S) es el mensaje recibido. En el procedimiento diseñado el atributo de entrada (E) con baja densidad con respecto al atributo target de salida (S) representa una alta probabilidad de que represente ruido (elemento anómalo). Se utiliza LOF para identificar la densidad local de ese elemento de entrada en relación al atributo target de salida, si la densidad es muy baja indicaría que es probable que se trate de un valor anómalo que requiere atención por parte del auditor.

4.3.2 Resultados de la aplicación del procedimiento desarrollado para BD alfanuméricas.

El resultado de ejecutar el algoritmo C4.5 permitió definir 5 atributos significativos junto al atributo Target:

- *espaciado_membrana.*
- *forma_sombrero.*

- *forma_tronco.*
- *olor.*
- *tipo_membrana.*
- *clase.*

En la tabla 4.10 se presentan los resultados después de realizada la experimentación, la implementación del procedimiento propuesta aparte de identificar los campos considerados anómalos, permitió obtener conocimiento útil para el experto en la detección de hongos venenosos, algunos de los hallazgos aparecidos en la base de datos después de aplicar el procedimiento son:

- El atributo "*forma_sombrero"* tiene 4 registros con forma de sombrero "*cónica"* y "*clase" = "venenoso"* identificados como outliers, no se encuentra otra combinación con forma de sombrero cónica.
- La combinación del atributo "*forma_ sombrero"= "acampanado"* con *"clase" = "venoso"* se detecto en solo 4 registros y los mismos fueron identificados como outliers por el procedimiento, el resto de tuplas que tienen el atributo "*forma_ sombrero"= "acampanado"* se relacionan con "*forma_ sombrero"= "comestible"*.
- El árbol de decisiones generado por el procedimiento señaló la importancia que tiene el atributo "*olor*", sin embargo de acuerdo a lo señalado por el experto difícilmente un hongo que no tenga olor puede ser venenoso, los casos donde se observa que el atributo *"olor" = "no"* fueron identificados por el procedimiento como outliers.
- El atributo "*tipo membrana"* aparece en la base de datos en 18 tuplas con la combinación *"tipo membrana" = "adherida"* y *"clase" = "venenoso"*, el experto consideró esta combinación como anómala.
- En el caso del atributo "*forma tronco"* de los 12 campos detectados como outliers, 2 correspondían a datos nulos, los mismos corresponden a falsos positivos.

La efectividad tal cual el criterio utilizado en la Sección 4.2.3.3 se calcula de la siguiente manera:

efectividad = porcentaje de outliers detectados – porcentaje falsos positivos

Del total de 59 campos considerados outliers por el experto, el procedimiento detecto el 100% de estos campos, y se obtuvieron un total de 6 falsos positivos. La efectividad promedio fue del 90,01% superando de esta manera el objetivo originalmente establecido en cuanto a la calidad del proceso de detección de valores anómalos.

Atributo	outliers detectados	outliers existentes	Falsos positivos	datos nulos	Efectividad
espaciado_membrana	8	8	0	0	100
forma_sombrero	12	8	4	0	66.66
forma_tronco	27	15	0	12	100
Olor	12	10	2	0	83.40
tipo_membrana	18	18	0	0	100

Tabla 4.10. Outliers detectados por cada atributo de la base de datos de Hongos

4.3.3 Discusión del procedimiento 3 desarrollado para la base de datos alfanumérica.

El procedimiento especificado permite identificar que campos y de que tuplas presentan algún tipo de inconsistencia, en bases de datos reales donde los datos no responden a una distribución conocida la validación de la experimentación se ve dificultada, siendo necesario en muchos casos contar con la asistencia de un experto en el dominio para analizar los posibles outliers y descartar los falsos positivos.

La incorporación en el procedimiento del algoritmo C4.5 permite reducir el espacio de búsqueda de posibles outliers, haciendo más eficiente su aplicación en grandes bases de datos.

El porcentaje de acierto es del 100% en lo relacionado a la detección de outliers y la cantidad de falsos positivos es muy baja.

Este procedimiento no es aplicable a bases de datos que no tienen un campo clasificador o target ya que es necesario realizar las combinaciones de (E) – (S), donde (S) es el atributo target.

4.4 Procedimiento 4. Orientado a bases de datos alfanuméricas sin un atributo target.

El objetivo de este procedimiento es detectar outliers en bases de datos alfanuméricas sin un atributo clase o target. La experimentación se realizará inicialmente en una base de datos creada en forma artificial y a continuación se validarán los resultados en un log de auditoría de un sistema real de gestión académica con el objetivo de detectar valores atípicos.

En la sub-sección 4.4.1 se determinan los posibles algoritmos a utilizar en el procedimiento 4, en 4.4.2 se definen los algoritmos posibles a utilizar específicamente diseñados para detectar outliers, en 4.4.3 se determinan los posibles algoritmos de clasificación a aplicar, en 4.4.4 se explica el proceso de selección de algoritmos, en 4.4.5 se seleccionan los algoritmos específicamente diseñados para detectar outliers a utilizar en el procedimiento 4, en 4.4.6 se establecen los algoritmos de clasificación a utilizar en el procedimiento 4, en 4.4.7 se diseña el procedimiento 4 y en 4.4.8 se realiza la experimentación sobre una base de datos real.

4.4.1 Algoritmos seleccionados.

Para la selección de las técnicas y algoritmos se establecieron dos criterios, el primero relacionado con algoritmos específicamente diseñados para la detección de outliers que tuviesen la capacidad de trabajar sobre los datos de tipo alfanumérico sin un atributo target, este punto se relaciona con la naturaleza del problema a resolver. El segundo criterio se relaciona con la selección de algoritmos de clasificación de carácter general, es decir, no orientadas en forma explícita a la detección de outliers, pero capaces de complementarse con los algoritmos específicamente diseñados para detectar datos anómalos, validando los outliers detectados por el primer grupo de algoritmos. El objetivo de utilizar estos dos criterios es lograr que el

procedimiento cumpla con las exigencias de efectividad y eficacia que requiere un proceso de auditoría de sistemas, el objetivo de la inclusión del segundo grupo de técnicas fue el de obtener modelos que permitieran confirmar o corregir los resultados obtenidos por los algoritmos específicamente diseñados para detectar outliers. Para la determinación de las técnicas del segundo grupo de algoritmos, o sea los algoritmos de clasificación se consideraron las recomendaciones realizadas por Pyle (Pyle, 2003) donde se establece que debe considerarse el entorno donde se quiere aplicar el proceso de minería de datos para determinar el mejor algoritmo a aplicar.

Considerando el problema a resolver las técnicas recomendadas por la bibliografía (Pyle, 2003) son:

- Extracción de reglas
- Árboles de decisión.
- Redes bayesianas.
- Redes neuronales.

Una vez detectada la fila considerada outlier se procederá de acuerdo a lo realizado en el procedimiento 3, creando una base de datos por columna y clusterizando con K-means con $K = 2$, seleccionando aquellos clusters que contienen los campos outliers.

4.4.2 Algoritmos específicos para la detección de outliers considerados.

Todos los algoritmos seleccionados para la detección de tuplas outliers son métodos basados en la densidad:

- LOF: Este algoritmo (Breunig et al., 2000) considera la densidad de los datos para determinar un factor local de outlier (LOF, en inglés) especificando en qué medida una tupla es outlier. Este algoritmo no requiere tener un atributo clase.
- DBSCAN: Este algoritmo (Saad & Hewahi, 2009) aplica principios similares a los utilizados por LOF, se basa en agrupar aquellas filas que son definidas como outliers en un cluster, separándolas del resto de la filas de la base de datos, que se agrupan en otros clusters. En DBSCAN no es necesario contar con un atributo de clase, por lo

tanto es posible aplicarlo para resolver el problema planteado en el procedimiento 4.

- DB-Outliers: Este algoritmo (Knorr & Ng, 1998) se basa en la distancia existente entre las filas de la base de datos, las tuplas que se encuentren más distantes de una determinada cantidad de vecinos serán consideradas outliers.
- Existen otros algoritmos específicamente diseñados para detectar outliers, como por ejemplo COF que detecta outliers de clase (Tang et al., 2002), pero fueron descartados ya que requieren contar con un atributo clase.

4.4.3 Algoritmos de clasificación considerados.

Los algoritmos de clasificación considerados fueron los siguientes:

- C4.5: Se implementa un árbol (Quinlan, 1993) de la familia de algoritmos TDIDT (Top Down Induction Decision Trees) para realizar tareas de clasificación. El algoritmo crea un árbol de decisión en base a un atributo clase o clasificador previamente definido usando a los demás atributos de la base de datos y los valores que pueden tomar para determinar si una tupla es incluida en una u otra clase.
- Red Bayesiana: Es una técnica (Jensen, 1996) que a través de un modelo probabilístico representa las relaciones existentes entre los diferentes campos de una base de datos, proporcionando un peso o importancia a cada una de las relaciones detectadas.
- PRISM: es un algoritmo (Cendrowska, 1987) de extracción de reglas que busca caracterizar en forma exacta a todos los elementos de una base de datos, entonces como resultado se obtiene un listado de reglas a evaluar en un orden específico.
- PART: Este algoritmo (Cao & Wu, 2004) obtiene una lista de reglas a seguir para tomar la decisión de clasificar a las filas de una BD, utiliza el algoritmo C4.5 para crear árboles de decisión parciales y

selecciona la mejor rama del árbol para transformar esa rama en una regla.

- Red Perceptrón: Se basa en una red neuronal (Rosenblatt, 1958) que actúa como un clasificador binario que asigna su entrada a un valor de salida, permitiendo definir a que clase pertenece una tupla, sólo puede resolver problemas que sean linealmente separables.
- Red Perceptrón Multicapa: Se implementa una red neuronal (Gardner & Dorling, 1998) compuesta de varias capas, usualmente una de entrada, varias capas ocultas y una de salida, en este tipo de red neuronal se puede clasificar a las tuplas de la base de datos en más de una clase.

4.4.4 Proceso de selección de algoritmos.

Se realizó un análisis sobre los distintos algoritmos tanto diseñados para detectar outliers como los de aplicación general siguiendo los siguientes criterios:

- La efectividad en la detección de outliers (Breunig et al., 2000; Ester et al., 1996).
- La cantidad de errores que se obtengan en el proceso de clasificación (Knorr & Ng, 1998; Aggarwal & Philip, 2005).
- La capacidad del algoritmo que sus resultados puedan ser comprensibles al usuario final.
- La compatibilidad del algoritmo con los objetivos del procedimiento.
- La mejora conjunta de la efectividad entre las técnicas que sean empleadas (Schaffer, 1994).
- Que el algoritmo pudiera operar sobre datos alfanuméricos.
- Dada una de las características del problema a resolver que se relaciona con la posibilidad que un auditor de sistemas no experto en minería de datos pueda ejecutar el procedimiento diseñado, se consideró que los algoritmos no requieran una gran cantidad de parámetros y que la determinación de los mismos se pudiera realizar en forma automática.

- Para los algoritmos de clasificación se priorizó que pudieran utilizarse en forma complementaria con los algoritmos diseñados específicamente para detectar outliers, de manera de poder integrar ambos tipos de algoritmos en el procedimiento propuesto.

El análisis de los algoritmos para seleccionar aquellos que se utilizarán en el procedimiento a diseñar fue realizado sobre una base de datos creada en forma artificial, siguiendo la estrategia seguida por varios autores (Johnson et al., 1998; Williams et al., 2002; Aggarwal & Philip, 2005).

Se aplicó un procedimiento para generar sobre la base de datos artificial un conjunto de campos anómalos en forma aleatoria, comparando la base de datos antes de incluir outliers y la misma después de incluirlos, de esta manera se individualizaron todos los valores anómalos de la BD de prueba. El porcentaje de valores anómalos incorporados en la base de datos para realizar las pruebas fue del 5% siguiendo el criterio establecido por otros investigadores (Peña, 2003; Zhang, Hutter & Jin, 2009; Aggarwal & Yu, 2001). En la tabla 4.11 se muestran las características de la base de datos utilizada.

Características	Cantidad
Cantidad de tuplas	500
Cantidad de atributos por tupla	8
Cantidad de Valores por atributo	5
Porcentaje de tuplas con outliers	5%

Tabla 4.11. Características BD artificial

4.4.5 Selección de algoritmos diseñados específicamente para detectar outliers.

La tabla 4.12 muestra los resultados obtenidos, en dicha tabla se pueden observar dos medidas de rendimiento utilizadas en numerosos trabajos, el porcentaje de efectividad y los falsos positivos de cada algoritmo analizado (Eskin et al., 2002; Zhang & Zulkernine, 2006; Leung & Leckie, 2005). La efectividad se define como el número de outliers detectados correctamente

dividido por el número de outliers presentes en la base de datos, los falsos positivos representa el número de tuplas erróneamente clasificadas como outliers dividido por la cantidad de tuplas no outliers de la base de datos. Se trata de buscar de un equilibrio entre un alto margen de efectividad, 65% mínimo; con un margen reducido menor al 5%, de falsos positivos.

Algoritmo	Outliers existentes	Outliers detectados	Efectividad	Falsos Positivos
LOF	25	17	68%	2,2%
BDSCAN	25	12	48%	1,6%
DB-OUTLIERS	25	19	76%	1,2%

Tabla 4.12. Resultados de los algoritmos para la detección de outliers

Como se puede observar el algoritmo BD-OUTLIERS tuvo muy buenos resultados, pero fue descartado ya que es necesario establecer previamente un número fijo de outliers a detectar, dado que en situaciones reales no es posible conocer ese número ambas técnicas fueron descartadas, seleccionándose LOF y DBSCAN.

En la sub-sección 4.4.5.1 se presenta una propuesta de unificación de los resultados obtenidos por los algoritmos LOF y DBSCAN, en 4.4.5.2 se definen un conjunto de reglas para determinar outliers después de unir los resultados de LOF y DBSCAN, en 4.4.5.3 se muestran los resultados de combinar los algoritmos LOF y DBSCAN.

4.4.5.1 Unión de los resultados de aplicar LOF y DBSCAN.

Con el objetivo de optimizar los resultados se propone la unión de los dos algoritmos seleccionados como se muestra en la figura 4.9, considerando que después de aplicar cada algoritmo en forma individual se agrega un atributo binario que determina si la tupla es un outlier o no lo es para ese algoritmo, en el caso de LOF cuando el valor que se obtiene después de

aplicarlo es mayor de 1,5 se lo considera un outlier de acuerdo a lo determinado en la sección 4.2.3.3. En el caso de DBSCAN los outliers son agrupados en el cluster 0.

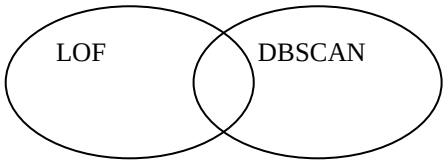


Figura 4.9. Unión de algoritmos

Concretamente se deben agregar cuatro columnas a la Tabla de auditoría evaluada, *"LOF"*, *"valor_LOF"*, *"valor_DBSCAN"* y *"tipo_outlier"*. Para cada tupla completar los valores de esos atributos con el siguiente criterio:

Aplicar LOF

Grabar en "LOF" el valor que se obtuvo después de aplicar este algoritmo

Si valor de "LOF" ≤ 1,5 entonces "valor_LOF" = "0"

Si valor de "LOF" > 1,5 entonces "valor_LOF" = "1"

Aplicar DBSCAN

Si la tupla pertenece a un clustes ≠ "0" entonces "valor_DBSCAN" = "0"

Si la tupla pertenece a un clustes = "0" entonces "valor_DBSCAN" = "1"

4.4.5.2 Reglas de determinación de outliers para algoritmos LOF y DBSCAN.

Inicialmente se asignan los siguientes valores al atributo *"tipo_outlier" = "outlier_doble"* cuando ambos algoritmos detectan a la tupla como outlier, *"outlier_simple"* cuando solo uno de los algoritmos detectan a la tupla como outlier, *"no_outlier"* cuando ninguno de los dos algoritmos detectan a la tupla como outlier, para ello se aplican las siguiendo las siguientes reglas sobre el set de datos:

Si "valor_LOF" = "1" y "valor_DBSCAN" = "1" entonces "tipo_outlier" = "outlier_doble"

Si "valor_LOF" = "1" y "valor_DBSCAN" = "0" entonces "tipo_outlier" = "outlier_simple"

Si "valor_LOF" = "0" y "valor_DBSCAN" = "1" entonces "tipo_outlier" = "outlier_simple"

Si "valor_LOF" = "0" y "valor_DBSCAN" = "0" entonces "tipo_outlier" = "no_outlier"

Con el objetivo de disminuir la cantidad de falsos positivos y optimizar el número de outliers detectados se realiza un ajuste siguiendo el siguiente criterio en cada una de las tuplas:

Si *"tipo_outliers" = "outlier_simple"* y *"valor_LOF" = "1" y "LOF" > 1.575 entonces "tipo_outlier" = "outlier_simple"*

Si *"tipo_outliers" = "outlier_simple"* y *"valor_LOF" = "1" y "LOF" ≤ 1.575 entonces "tipo_outlier" = "no_outlier"*

(El aumento en el valor límite del 5% del valor de LOF se debe a un incremento en la exigencia para la determinación de los outliers considerando los resultados de la técnica para evitar falsos positivos.)

Si *"tipo_outliers" = "outlier_simple" y "valor_DBSCAN" = "1" y "LOF" > 1.425 entonces "tipo_outlier" = "outlier_simple"*

Si *"tipo_outliers" = "outlier_simple" y "valor_DBSCAN" = "1" y "LOF" ≤ 1.425 entonces "tipo_outlier" = "no_outlier"*

(En este caso la disminución del 5% del valor de LOF se debe a un mayor grado de confianza en los resultados obtenidos por DBSCAN en base a la cantidad de errores registrados.)

4.4.5.3 Combinación de LOF y DBSCAN.

Como resultado del proceso descripto anteriormente se obtiene la base de datos resultante de la unión de aplicar LOF y DBSCAN con 4 nuevos atributos, *"LOF"*, *"valor_LOF"*, *"valor_DBSCAN"* y *"tipo_outlier"*.

La efectividad en la detección de outliers, ver tabla 4.13, tuvo una mejora del 4% con respecto a los resultados de aplicar LOF y del 24% con respecto a

DBSCAN, en lo relacionado con los falsos positivos la disminución fue de una unidad con respecto a LOF y de cuatro decimos con respecto a DBSCAN.

LOF		DBSCAN		Unión LOF y DBSCAN	
Efectividad	Falsos positivos	Efectividad	Falsos positivos	Efectividad	Falsos positivos
68%	2,2%	48%	1,6%	72%	1,2%

Tabla 4.13. Comparación resultados

4.4.6 Selección de algoritmos de clasificación.

La selección de algoritmos de clasificación se encuentra focalizada en la mejora de los resultados obtenidos al aplicar la unión de LOF y DBSCAN, el objetivo se relaciona entonces con mejorar la efectividad en la detección de outliers y disminuir la cantidad de falsos positivos en la base de datos después de aplicar los algoritmos específicamente desarrollados para detectar outliers a través de la aplicación de uno o más algoritmos de clasificación, es decir que estos algoritmos se aplican sobre la base de datos después de aplicar los algoritmos LOF y DBSCAN.

Los algoritmos de clasificación predicen el valor de un atributo denominado “clase” basándose en otros atributos del set de datos, como resultado de aplicar los algoritmos especialmente diseñados para detectar outliers se cuenta con un atributo de clase único (*“tipo_outlier”)* que indica el resultado de cada tupla después de aplicar los algoritmos LOF y DBSCAN, esto simplifica la experimentación con los algoritmos de minería de datos de clasificación ya que no es necesario modificar la base de datos para agregar un atributo target, al aplicar los algoritmos de clasificación lo que se logra es verificar bajo determinadas reglas si la tupla se confirma o no como outlier.

En la tabla 4.14 se muestran el resultado de aplicar cada uno de los algoritmos considerados en la sección 4.3.5 , las columnas “diferencia efectividad” y “diferencia falsos positivos” relacionan los resultados de cada algoritmo de uso general con la unión de LOF+DBSCAN, en relación con la

efectividad los algoritmos C4.5 y PRISM fueron lo que mantuvieron el porcentaje de detección de outliers similar al de los algoritmos LOF y DBSCAN, el resto de algoritmos representaron una disminución en el porcentaje de efectividad.

En la sub-sección 4.4.6.1 se combinan los algoritmos de clasificación, en 4.4.6.2 se establecen las reglas para aplicar los algoritmos de clasificación y en 4.4.6.3 se presentan los resultados de ejecutar en forma individual los algoritmos de clasificación seleccionados.

Algoritmo	Efectividad	Falsos Positivos	Diferencia Efectividad	Diferencia Falsos Positivos
C4.5	72%	1,2%	0%	0%
Red Bayesiana	4%	0%	-68%	-1,2%
PRISM	72%	1,2%	0%	0%
PART	66%	1%	-6%	-0,2%
Red Perceptrón	6%	0,4%	-64%	-0,8%
Red Perceptrón Multicapa	32%	0,4%	-40%	-0,8%

Tabla 4.14. Resultado de aplicar los algoritmos de uso general

4.4.6.1 Combinación de algoritmos de clasificación.

Como muestra la figura 4.10 se decide aplicar la unión de los algoritmos C4.5 + Red Bayesiana + PRISM.

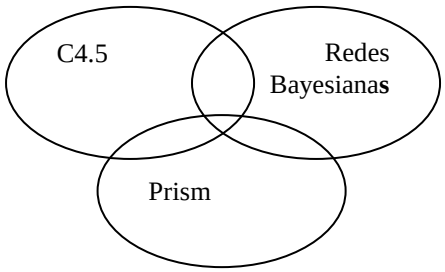


Figura 4.10. Unión de los algoritmos C4.5, RB y PRISM

Los algoritmos C4.5 y las Red Bayesiana (RB) son incluidos ya que la primera mantuvo la efectividad en la detección de outliers, pero no logró una variación en los falsos positivos; mientras que el resultado de aplicar las Redes bayesianas permitió eliminar todos los falsos positivos pero disminuyó drásticamente la efectividad en la detección de outliers. Con el objetivo de equilibrar esta situación se incluyó al algoritmo PRISM con el objeto de obtener los mejores resultados globales posibles a través de sus respectivos modelos.

4.4.6.2 Reglas de determinación de outliers para algoritmos de clasificación.

Se establece dentro del procedimiento, después de aplicar LOF+DBSCAN, un conjunto de reglas para optimizar la combinación de los resultados obtenidos por los algoritmos C4.5, PRISM y Redes bayesianas, dando como resultado dos valores posibles en cada tupla del atributo *"tipo_outlier"*, estos valores son *"outlier"* o *"limpio"*, los resultados de los valores obtenidos al aplicar LOF son tenidos en cuenta como un elemento de validación. Todas estas reglas surgen de la experimentación realizada.

El objetivo entonces de aplicar estos 3 algoritmos de clasificación es optimizar los resultados obtenidos por los algoritmos de detección de outliers siguiendo un conjunto de reglas sobre cada tupla para determinar el valor final del atributo *"tipo_outlier"* que será el atributo clase de la tupla y que como resultado final podrá ser *"outlier" o "limpio"*. Estas reglas son:

- *Si "tipo_outlier" = "outlier_simple" entonces "tipo_outlier" = "outlier_doble"* (el objetivo de esta operación es contar con solo dos valores en el atributo *"tipo_outlier"* de manera de simplificar el proceso de clasificación).
- *Clasificar cada tupla con los algoritmos C4.5, Redes bayesianas y PRISM.*
- *Para cada tupla Si para los 3 algoritmos de clasificación "tipo_outlier" = "outlier_doble" entonces "tipo_outlier" = "outlier".*

- *Si para los algoritmos Redes bayesianas y PRISM "tipo_outlier"= "outlier_doble" entonces"tipo_outlier" = "outlier".*
- *Si para el algoritmo de Redes bayesianas "tipo_outlier" = "outlier_doble" y para C4.5 y PRISM "tipo_outlier" = "no_outlier" y "valor_LOF" > 1.575 entonces"tipo_outlier" = "outlier".*
- *Si para los algoritmos C4.5 y PRISM "tipo_outlier" = "outlier_doble" y para Redes bayesianas "tipo_outlier" = "no_outlier" entonces "tipo_outlier" = "outlier".*
- *Si para el algoritmo C4.5 "tipo_outlier" = "outlier_doble" y para PRISM "tipo_outlier" = "no_outlier" y para Redes bayesianas "tipo_outlier" = "no_outlier" y "valor_LOF" > 1.65 entonces "tipo_outlier" = outlier.* (el 10% sobre el valor límite de 1,5 establecido originalmente para considerar un outlier surge de la experimentación realizada)
- *Si para el algoritmo PRISM "tipo_outlier" = "outlier_doble" y para C4.5 "tipo_outlier" = "no_outlier" y para Redes bayesianas "tipo_outlier" = "no_outlier" y "valor_LOF" > 1.65 entonces "tipo_outlier" = "outlier". (el 10% sobre el valor límite de 1,5 establecido originalmente para considerar un outlier surge de la experimentación realizada mejorando la detección de outliers)*
- *Cualquiera de las tuplas que no cumple con estas condiciones entonces "tipo_outlier" = "limpio".*

4.4.6.3 Resultados de ejecutar los algoritmos C4.5, redes bayesianas y PRISM en forma individual.

Como resultado de aplicar los 3 algoritmos de clasificación sobre la base de datos creada específicamente para realizar estas pruebas se observa que los falsos positivos han disminuido al plantear un enfoque combinatorio. La tabla 4.15 muestra los resultados obtenidos.

Resultados obtenidos		Diferencia C4.5 con la unión de LOF+DBSCAN		Diferencia RB con la unión de LOF+DBSCAN		Diferencia PRISM con la unión de LOF+DBSCAN	
Efectividad 68%	Falsos positivos 0.8%	Efectividad -4%	Falsos positivos -0.4%	Efectividad +64%	Falsos positivos 0%	Efectividad 0%	Falsos positivos 0%

Tabla 4.15. Resultados de la aplicación de algoritmos de clasificación

4.4.7 Diseño del procedimiento propuesto.

El procedimiento 4 tiene un enfoque de tipo 1 con aprendizaje no supervisado al utilizar dos algoritmos de clusterización basado en la densidad (LOF y DBSCAN), redes bayesianas, el algoritmo C4.5, PRISM y K-means, este procedimiento híbrido también tiene un enfoque de tipo 2 al utilizar algoritmos relacionados con el aprendizaje automático (C4.5).

Como lo muestra la figura 4.11 el procedimiento propuesto tiene las siguientes etapas:

- *Entrada: base de datos.*
  - *Leer Base de datos.*
    - *Aplicar LOF. Agregar atributo "valor_LOF" a cada tupla y grabar resultado de LOF de acuerdo a criterios desarrollados en la Sección 4.4.5.2 .*
    - *Aplicar DBSCAN. Agregar "valor_DBSCAN" a cada tupla y grabar resultado de acuerdo a criterios desarrollados en la Sección 4.4.5.2.*
  - *Unir resultados. Agregar atributo "tipo_outlier" a cada tupla. Grabar resultado de acuerdo a criterios desarrollados en la Sección 4.4.5.1.*
  - *Leer BD.*
    - *Aplicar C4.5 y determinar valor del atributo target "tipo_outlier" para este algoritmo y grabar en memoria.*

        - *Aplicar Redes bayesianas y determinar valor del atributo target "tipo_outlier" para este algoritmo y grabar en memoria.*
        - *Aplicar PRISM y determinar valor del atributo target "tipo_outlier" para este algoritmo y grabar en memoria.*
    - *Unir resultados.*
        - *Aplicar reglas para determinar tuplas outliers de acuerdo a criterios determinados en la Sección 4.4.6.3.*
        - *Grabar resultados finales en cada tupla del atributo target "tipo_outlier" con dos valores posibles: "limpio" o "outlier".*
        - *Crear una BD por cada columna cuyo valor "tipo_outlier" = "outlier".*
        - *Clusterizar la primer BD (primera columna) creada que tiene un valor "tipo_outlier" = "outlier" con K-MEANS con K=2.*
        - *Calcular la distancia entre los centroides de los clusters creados, el cluster que está más lejano del centroide es el que contiene los campos considerados outliers. De esta manera se logra identificar los atributos específicos de esa columna que tienen sospecha de ser considerados outliers.*
        - *Repetir el procedimiento para cada columna de la base de datos que tiene BD que contiene un valor de "tipo_outlier" = "outlier".*
- *Salida: base de datos con los campos anómalos detectados.*
    - *Fin procedimiento.*

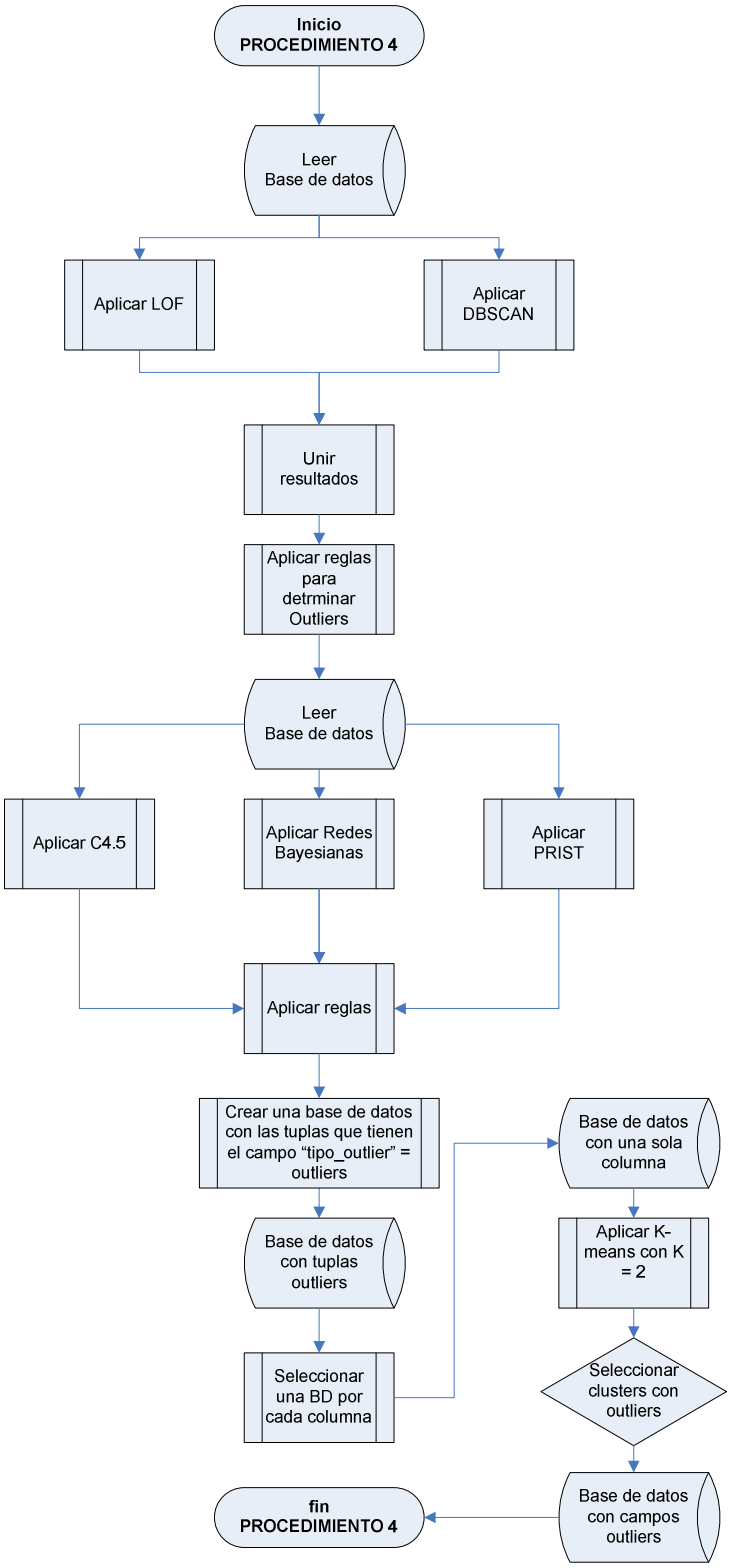


Figura 4.11. Procedimiento propuesto.

4.4.8 Experimentación sobre una base de datos real.

La experimentación se realizó sobre una base de datos real de un sistema de gestión de alumnos de una universidad, analizando los registros de auditoría de los módulos “Gestión de Exámenes”, “Gestión de Cursadas” y “Gestión de Matriculas”. La selección de los módulos y tablas para verificar el procedimiento propuesto se realizó en conjunto con el personal responsable de la administración del sistema.

La tabla 4.15 muestra las tablas que contienen los logs de auditoría de cada uno de los módulos analizados.

Módulos	Nombre tabla	Cantidad de filas	Cantidad de Columnas
Gestión de Exámenes	log_actas_examen	10805	28
	log_detalle_actas	84263	23
Gestión cursada	log_actas_cursada	18572	19
	log_actas_promocion	7403	22
	log_detalle_actas_cursada	17321	20
	log_detalle_actas_promocion	12776	19
Gestión de Matriculas	log_alumnos	5701	16
	log_carrera_aspira	9049	15

Tabla 4.16. Tablas de logs seleccionadas para el análisis

Los datos seleccionados corresponden a los años 1998 a 2001, esta selección inicial fue recomendada por los expertos en el dominio ya que en esos años se han producido operaciones consideradas anómalas dentro del sistema. Junto con los expertos se estableció un mínimo del 65% de efectividad como umbral de calidad del proceso automático de detección de outliers, el máximo porcentaje de falsos positivos aceptado fue del 1%. Los expertos son dos administradores del sistema de gestión de alumnos.

Los expertos en la operación del sistema analizaron en detalle y en forma manual los logs de auditoría del módulo “Gestión de Exámenes” y se

determinó las tuplas consideradas anómalas, este análisis manual se realizó con el objetivo de validar los resultados obtenidos después de aplicar el procedimiento propuesto. Se consideró para validar los resultados al modulo “Gestión de Exámenes” dado que para los administradores del sistema cumple una función crítica dentro del sistema.

Se realizó un preproceso de las tablas relacionadas con el Modulo de Exámenes, se eliminaron atributos de las tablas que no aportaban información o que tenían valores nulos.

Se unieron las tablas *“log_actas_examen”* y *“log_detalle_actas”* con el objetivo de aplicar el procedimiento sobre una única tabla de logs de auditoría del Modulo de Exámenes. Creándose la Base de datos “l*og_examenes”.* Se seleccionó para realizar la experimentación los datos correspondientes solamente al año 2000 ya que los expertos en el dominio evaluaron que en ese año se produjeron los mayores problemas en la operación del sistema.

A los efectos de optimizar el análisis y detección de outliers y siguiendo las recomendaciones de los expertos en el dominio se procedió a dividir esta base de datos unificada en 10 tablas, una para cada turno de examen, la tabla 4.16 muestra las tablas sobre las que se aplicó el procedimiento.

Nombre de la tabla	Cantidad filas
log_examen_primero_2000	7840
log_examen_segundo_2000	4468
log_examen_tercero_2000	13590
log_examen_cuarto_2000	7830
log_examen_quinto_2000	3985
log_examen_sexto_2000	1896
log_examen_septimo_2000	17009
log_examen_octavo_2000	7912
log_examen_noveno_2000	9663
log_examen_especial_2000	2236

Tabla 4.16. Tablas utilizadas para aplicar el procedimiento

En la sub-sección 4.4.8.1 se presentan los resultados después de aplicar el procedimiento 4 y en 4.4.8.2 se realiza una discusión sobre el procedimiento 4.

4.4.8.1 Resultados obtenidos con el procedimiento 4.

Los resultados obtenidos después de aplicar el procedimiento se muestran en la tabla 4.17, los resultados obtenidos por el procedimiento 4 fueron comparados con los outliers detectados en forma manual por los expertos en el dominio.

Nombre de la tabla	Cantidad de outliers a detectar	Cantidad de outliers detectados	Efectividad	Falsos positivos detectados
log_examen_primero_2000	94	70	65.95%	0.10%
log_examen_segundo_2000	55	53	76.36%	0.24%
log_examen_tercero_2000	93	93	69.89%	0.20%
log_examen_cuarto_2000	88	72	65.90%	0.17%
log_examen_quinto_2000	57	64	84.21%	0.40%
log_examen_sexto_2000	38	39	78.94%	0.47%
log_examen_septimo_2000	92	81	66.30%	0.11%
log_examen_octavo_2000	77	90	88.31%	0.27%
log_examen_noveno_2000	97	100	76.28%	0.26%
log_examen_especial_2000	60	69	90.00%	0.67%

Tabla 4.17. Resultados obtenidos

4.4.8.2 Discusión de la experimentación realizada con el procedimiento 4.

El promedio de la efectividad promedio obtenida fue del 76%, siendo el mínimo 65.90% y el máximo del 90%, la base del 65% previamente establecido de efectividad fue ampliamente superada por el procedimiento, por lo tanto se cumplieron los requerimientos de calidad previamente establecidos. En ningún caso el porcentaje de los falsos positivos fue superior al 1%, porcentaje establecido por los expertos como máximo valor aceptado.

4.5 Discusión de las soluciones propuestas.

Como se ha expresado en la sección 1.2, donde se explicó el problema a resolver, no se ha encontrado evidencias de la existencia de algoritmos que detecten campos anómalos, se observa la existencia de numerosos algoritmos que resuelven bajo determinados escenarios la detección de filas consideradas outliers, como se desarrolló en la sección 4.1 la validación empírica de la calidad de los algoritmos dificulta en la mayoría de los casos el proceso de generalización de la aplicación de los mismos, cada algoritmo es el mejor bajo determinados escenarios relacionados con el set de datos a analizar. La tarea del auditor de sistemas a la hora de analizar la calidad de los datos de una base de datos relacionada con sistemas de gestión se encuentra con la duda de cuál es el algoritmo que mejor aplica al escenario que debe abordar. Como se ha demostrado durante esta tesis no existen algoritmos que detecten campos considerados outliers, si se desarrollaron algoritmos que permiten detectar tuplas pero no específicamente que atributo de esa tupla contiene valores anómalos, para resolver este problema se desarrollaron 4 procedimientos que combinando distintas técnicas y para diversos escenarios resuelven este problema, los mismos son:

- Procedimientos 1 y 2: Permiten detectar outliers en bases de datos numéricas, los mismos se basan en el uso de los metadatos de los registros considerados normales, y en el algoritmo LOF que proporciona un valor local de outliers, que permite definir en qué medida una tupla es considerada anómala. Se utilizó una base de datos creada considerando la distribución normal para definir los mejores valores de las distintas variables de LOF utilizadas, este tipo de distribución permite definir de manera estadística los valores anómalos, problema de compleja resolución en bases de datos reales. Se realizaron más de 100 pruebas para determinar los valores óptimos de LOF, *MinPtsMin* y *MinPtsMax*, en estas pruebas realizadas con bases de datos construidas de acuerdo a la distribución normal se observó un elevado número de falsos positivos. Se utilizó una base de datos real, con datos relacionados con el cáncer donde los

dos procedimientos no detectaron falsos positivos, o sea su comportamiento fue mejor en la base de datos real que en la base de datos creada de acuerdo a la distribución normal. La intersección de los campos detectados por ambos procedimientos detectó el 60.15% de campos considerados outliers, considerándose que el resultado responde a la métrica de calidad mínima establecida.

- El procedimiento 3 fue diseñado para detectar campos outliers en bases de datos alfanuméricas con un atributo target, el mismo combina elementos de la teoría de la Información de Shannon, un algoritmo de inducción y LOF, se experimentó con una base de datos relacionada con hongos que contenía un atributo que define si el hongo es venenoso o no lo es. La efectividad promedio del procedimiento fue del 90% y el número de falsos positivos fue muy bajo.
- El procedimiento 4 también se relaciona con una base de datos alfanumérica pero sin un atributo target, el mismo combina inicialmente los algoritmos LOF y DBSCAN y luego se combinan los algoritmos C4.5, PRISM y las Redes Bayesiana, aplicando finalmente el algoritmo K-means, se desarrollaron reglas para optimizar la determinación de outliers.

  El algoritmo LOF demuestra mejores resultados en bases de datos numéricas, por esa razón se combinaron diferentes algoritmos sobre una base de datos alfanumérica para mejorar sus resultados, la batería de algoritmos implementados dentro del enfoque hibrido del procedimiento 4 aconseja su aplicación sobre bases de datos con función crítica ya que tiene un alto porcentaje de acierto y muy bajo de falsos positivos y la combinación propuesta de algoritmos garantiza buenos resultados, aunque no es aconsejable para grandes bases de datos ya que la combinación de algoritmos implica una baja en la performance del procedimiento propuesto.

  La experimentación se realizó con los logs de auditoría de una base de datos real relacionada con un sistema de gestión académica. La

efectividad promedio fue del 76% y en ningún caso los falsos positivos representaron más del 1% de los outliers detectados.

Como resultado del desarrollo y experimentación de los procedimientos propuestos es posible afirmar que la combinación de algoritmos de distinta naturaleza y también la combinación de procedimientos permite detectar outliers con un nivel de confianza mayor al 60%, entendiendo que para cada escenario específico es conveniente combinar distintos algoritmos para lograr mejores resultados en la detección de campos considerados outliers.

- Los métodos híbridos implementados en los procedimientos propuestos demostraron ser una interesante alternativa a la hora de abordar la búsqueda de campos outliers en diferentes escenarios.

La tabla 4.18 muestra un resumen de los enfoques de cada uno de los procedimientos propuestos.

Procedimiento	Entorno	Algoritmos/técnicas	Enfoques
1	Bases de datos numéricas con o sin atributo target	LOF Metadatos	1 y 3
2	Bases de datos numéricas con o sin atributo target	LOF K-means	1
3	bases de datos alfanuméricas que contienen un atributo Target	C4.5 Teoría de la información LOF	1 y 2
4	bases de datos alfanuméricas que no contienen un atributo Target	LOF DBSCAN C4.5 Redes bayesianas PRISM K-means	1 y 2

Tabla 4.18. Resumen enfoque de los algoritmos desarrollados

5. Conclusiones y futuras líneas de investigación.

El problema que resuelve esta tesis se relaciona con la detección automática de campos considerados outliers en bases de datos, esta detección es de suma importancia en la tarea del auditor de sistemas ya que automatiza el proceso de detección de campos anómalos; optimizando los tiempos, la eficacia y la eficiencia de esta actividad.

Existen antecedentes en la detección de tuplas anómalas, de la revisión bibliográfica realizada no se han detectado antecedentes en la detección de cual es específicamente el campo que contiene valores donde existe una sospecha que fueron creados por un mecanismo distinto al resto de campos de la base de datos, siendo esta detección el principal aporte de esta tesis.

En la sección 5.1 se desarrollan las principales aportaciones de la tesis, en la sección 5.2 se proponen las futuras líneas de investigación.

5.1 Aportaciones de la tesis.

En esta tesis doctoral se realiza una taxonomía de los métodos de detección de outliers, es importante destacar que existe una gran variedad de algoritmos y herramientas que utilizan a la minería de datos en la obtención de conocimiento previamente desconocido y que es de utilidad en bases de datos (Kuna et al., 2010a), muchos de estos algoritmos son utilizados en la detección de datos anómalos, en el capítulo 2 se realiza una clasificación de los métodos de detección de outliers, los mismos pueden resumirse en tres enfoques:

- Basado en el aprendizaje no supervisado
- Basado en el aprendizaje supervisado
- Basado en el aprendizaje semi-supervisado

considerando los tres enfoques descriptos se asignó a cada algoritmo a uno de los enfoques descriptos. Se analizó la relación entre los métodos y los enfoques utilizados en la detección de outliers y se detectó que la mayoría de los algoritmos utilizados se relacionan con el enfoque vinculado con el aprendizaje no supervisado, esto se debe a la imposibilidad, en muchos casos,

de definir previamente dentro de un set de datos real cuales son las tuplas consideradas outliers. Solo en casos muy específicos es factible utilizar el enfoque relacionado con el aprendizaje supervisado ya que en la mayoría de los casos no es posible conocer previamente en un set de datos particular cuales son efectivamente los campos outliers. El aprendizaje semi-supervisado, de casi nula aplicación en la búsqueda bibliográfica realizada, donde lo que se define son las características de los datos considerados normales como una manera de detectar los datos anómalos, aparece como una alternativa importante a la hora de definir estrategias en la búsqueda de datos que implican ruido dentro de la base de datos en la que no existe una distribución conocida de los datos. Para diseñar los procedimientos propuestos, cuyo objetivo era detectar campos anómalos, se combinaron algoritmos de los tres enfoques produciéndose los resultados esperados.

Como consecuencia de la confidencialidad que requieren empresas y organismos para tratar sus datos, está restringido el acceso a las bases de datos, esto produce dificultades para poder realizar la experimentación en la búsqueda y detección de outliers, en este contexto en esta tesis se validan los procedimientos creando bases de datos que responden a la distribución normal y se utilizan base de datos reales donde son previamente conocidos los datos considerados anómalos, en algunos casos por la existencia de un atributo target y en otros se debió recurrir a los servicios de expertos en el dominio que realizaron una evaluación manual de las bases de datos de manera de conocer los datos que se consideran outliers y de esta manera poder validar los resultados. Se considera que esta metodología es necesaria para validar cualquier procedimiento relacionado con la detección de outliers, la misma consiste entonces en validar los resultados utilizando dos bases de datos, una creada en forma aleatoria respetando la distribución normal y detectando los outliers por métodos estadísticos y otra base de datos real que contiene un atributo target que permite identificar las tuplas anómalas junto a la tarea manual del experto en el dominio que identifica los campos anómalos.

Existen obstáculos en generalizar la aplicación de los algoritmos que se utilizan ya que los mismos tienen una alta dependencia con el entorno donde

se aplican y se observa una fuerte relación entre las soluciones propuestas y el dominio del problema a resolver.

No es posible detectar campos anómalos utilizando un único algoritmo, ya que como se demostró a lo largo de la tesis este algoritmo no fue desarrollado y no se considera posible desarrollar un algoritmo que resuelva todos los entornos posibles a los que debe enfrentarse un auditor de sistemas.

Es necesario adaptar los procedimientos que se utilicen en la detección de campos anómalos a los dominios específicos donde se requiere realizar esta detección, considerando: las características de la base de datos a analizar, el tipo de campo a detectar, la dimensionalidad, entre otros puntos.

Fue necesario entonces desarrollar procedimientos que combinen diferentes algoritmos, aplicando de esta manera el método híbrido que surgen como la solución al problema que se quiere resolver. La integración de distintos algoritmos no solo permiten detectar los campos considerados outliers sino que minimiza los posibles errores que pueda tener un algoritmo ante tan diversos e inciertos escenarios, la utilización de este tipo de métodos permite superar las deficiencias de un algoritmo particular potenciando los puntos fuertes de cada algoritmo y minimizando los débiles, siendo esta la solución elegida para identificar los campos anómalos.

Un problema adicional a la detección de outliers que debe ser considerado son los inliers que son datos detectados como atípicos pero que no tiene el comportamiento de un verdadero outlier.

Ante la variedad de escenarios posibles se seleccionaron un conjunto de características del entorno a auditar, estas especificidades se relacionan con las situaciones fácticas con las que los auditores de sistemas debe enfrentarse: detección de campos numéricos, alfanuméricos, bases de datos con la presencia de un atributo target o sin ese atributo.

Para detectar campos considerados outliers se evaluó entonces que la mejor alternativa era aplicar métodos híbridos, utilizando los tres enfoques detectados, el supervisado, el no supervisado y el semi-supervisado. Se aplicaron algoritmos específicamente diseñados para detectar outliers (como LOF y DBSCAN), algoritmos de uso general (como K-means, redes

bayesianas, Prism, entre otros) y el algoritmo C4.5 (Kuna et al, 2009; Kuna et al., 2010a; Pautch et al., 2011) con el objetivo de detectar los atributos que mayor ganancia de información tienen para de esta manera reducir el espacio de búsqueda y poder optimizar la performance de los procedimientos. Como parte de la aportación de esta tesis se definieron reglas que permitieron combinar los distintos algoritmos dentro de los procedimientos.

Los procedimientos 1 y 2 (Kuna et al., 2012b; Kuna et al., 2013a) se combinan para detectar campos outliers en bases de datos numéricas.

El procedimiento 3 (Kuna et al., 2012c; Kuna et al., 2013b) está orientado a la detección de campos considerados outliers en bases de datos alfanuméricas que contienen un atributo Target.

El procedimiento 4 (Kuna et al. ,2014) detecta campos outliers en bases de datos alfanuméricas sin atributo Target.

5.2 Futuras líneas de investigación.

Se proponen como futuras líneas de investigación el análisis de la performance de los procedimientos a utilizar, evaluando en qué casos es conveniente utilizar a la computación de altas prestaciones como soporte para las grandes bases de datos.

Surge como una línea interesante de investigación a abordar la implementación de otros algoritmos como por ejemplo la lógica difusa para aquellas bases de datos con un alto grado de incertidumbre, o los algoritmos genéticos con el objetivo de ahorrar tiempo computacional.

La reducción de falsos positivos es un tema que requiere futuras investigaciones, ya que dependiendo de la criticidad de los datos que se analizan, esta disminución de falsos positivos puede ser fundamental en la calidad del trabajo que realiza el auditor de sistemas

La detección automática de inliers es un tema de importancia a abordar en el futuro ya que en determinados entornos la presencia de este tipo de datos puede requerir su identificación.

También es interesante investigar la posibilidad de embeber en los sistemas de gestión los procedimientos propuestos para de esta manera aportar al proceso de autoevaluación de controles por parte de las empresas y organismos.

Referencias

Abbott, D. W., Matkovsky, I. P., & Elder, J. F. (1998). An evaluation of high-end data mining tools for fraud detection. *Proceedings of IEEE International Conference on Systems, Man, and Cybernetics*, *Vol. 3*, 2836 -2841.

Abbot, J. L., Park, Y., & Parker, S. (2000). The effects of audit committee activity and independence on corporate fraud. *Managerial Finance, 26*(11), 55-67.

Abe, N., Zadrozny, B., & Langford, J. (2006). Outlier detection by active learning. *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining,* 504-509.

Acuna E. and Rodriguez C., (2004), A Meta analysis study of outlier detection methods in classification, Technical paper, Department of Mathematics, University of Puerto Rico at Mayaguez, available at academic.uprm.edu/~eacuna/paperout.pdf. *Proceedings IPSI*.

Asociación de Auditoría y Control de Sistemas de Información (ADACSI). (s.f.). Recuperado de http://www.adacsi.org.ar (recuperado el 10 de julio de 2013).

Agrawal, R., Gunopulos, D., & Leymann, F. (1998). Mining process models from workflow logs. *Springer Berlin Heidelberg*, 467-483.

Aggarwal, C. C., & Yu, P. S. (2001). Outlier detection for high dimensional data. *ACM Sigmod Record, Vol. 30*(2), 37-46.

Aggarwal, C. C., & Philip, S. Y. (2005). An effective and efficient algorithm for high-dimensional outlier detection. *The VLDB journal*, *14*(2), 211-221.

Aleskerov, E., B. Freisleben, B. Rao (1997). Cardwatch: A neural network based database mining system for credit card fraud detection. *Proceedings of the IEEE/IAFE, Conference on Computational Intelligence for Financial Engineering (CIFEr),* 220-226.

Anbaroglu, B. (2008). Considering spatio-temporal outliers in function approximation. *Computer and Information Sciences*, ISCIS'08. 23rd International Symposium, 1-6.

Ankerst, M., Breunig, M. M., Kriegel, H. P., & Sander, J. (1999). OPTICS: ordering points to identify the clustering structure. *ACM SIGMOD Record*, *28*(2), 49-60.

Arnold, B. C., & Salinas, H. S. (2012). A doubly skewed normal distribution. *X Congreso Latinoamericano de Sociedades de Estadística, Córdoba, Argentina.*

Arning, A., Agrawal, R., & Raghavan, P. (1996). A Linear Method for Deviation Detection in Large Databases. *KDD,* 164-169.

Back, B., Toivonen, J., Vanharanta, H., & Visa, A. (2001). Comparing numerical data and text information from annual reports using self-organizing maps. *International Journal of Accounting Information Systems, 2*(4), 249-269.

Barnett, V., & Lewis, T. (1994). Outliers in statistical data, *3*.

Barto, A. G. (1984). Simulation Experiments with Goal-Seeking Adaptive Elements. *Massachusetts Univ Amherst Dept of Computer and Information Science.*

Bell, T., & Carcello, J. (2000). A decision aid for assessing the likelihood of fraudulent financial reporting. *Auditing: A Journal of Practice & Theory*, *9*(1), 169-178.

Ben-Gal, I. (2005). Outlier detection. *Data Mining and Knowledge Discovery Handbook,* 131-146.

Birnhack, M. D., & Elkin-Koren, N. (2002). Fighting Terror On-Line: The Legal Ramifications of September 11. Internal Report, *The Law and Technology Center*, Haifa University.

Bishop, C. M. (1995). Neural networks for pattern recognition. *Oxford university press.*

Blake, C. L. & Merz, C. J. (1998). UCI Repository of Machine Learning Databases, University of California, Irvine, Department of Information and Computer Sciences. Recuperado de http://www.ics.uci.edu/mlearn/MLRepository.html.

Bolton, R. J., & Hand, D. J. (2001). Unsupervised profiling methods for fraud detection. *Credit Scoring and Credit Control, II*, 235-255.

Boscovich, R. J. (1757). De litteraria expeditione per pontificiam ditionem, et synopsis amplioris operis, ac habentur plura ejus ex exemplaria etiam sensorum impressa. *Bononiensi Scientarum et Artum Instituto Atque Academia Commentarii, 4,* 353-396.

Bourlard, H., & Kamp, Y. (1988). Auto-association by multilayer perceptrons and singular value decomposition. *Biological cybernetics*, *59*(4-5), 291-294.

Britos, P. Hossian, A., García-Martinez, R., & Sierra, E. (2005). Minería de Datos Basada en Sistemas Inteligentes*. Nueva Librería*.

Britos, P., Grosser, H., Rodríguez, D., & Garcia-Martinez, R. (2008). Detecting Unusual Changes of Users Consumption. En M. Bramer (Ed.), *Artificial Intelligence in Theory and Practice II, 276,* 297–306.

Breiman, L. (Ed.). (1993). *Classification and regression trees*. CRC press.

Breunig, M. M., Kriegel, H. P., Ng, R. T., & Sander, J. (1996). LOF: identifying density-based local outliers. *ACM Sigmod Record, 29*(2), 93-104.

Breunig, M. M., Kriegel, H. P., Ng, R. T., & Sander, J. (2000). LOF: identifying density-based local outliers. *ACM Sigmod Record, 29*(2), 93-104.

Brodley, C. E., & Friedl, M. A. (1996). Identifying and eliminating mislabeled training instances. In *AAAI/IAAI,* 1, 799-805.

Caudell, T., & Newman, D. (1993). An adaptive resonance architecture to define normality and detect novelties in time series and databases. *Portland, Oregon*, 166-176.

Cendrowska, J. (1987). PRISM: An algorithm for inducing modular rules. *International Journal of Man-Machine Studies*, *27*(4), 349-370.

Cao, Y., & Wu, J. (2004). Dynamics of projective adaptive resonance theory model: the foundation of PART algorithm. Neural Networks. *IEEE Transactions* on, *15*(2), 245-260.

Chandola, V., Banerjee, A., & Kumar, V. (2009). Anomaly detection: A survey. *ACM Computing Surveys (CSUR)*, *41*(3), 15.

Chapman, P., Clinton, J., Kerber, R., Khabaza, T., Reinartz, T., Shearer, C., & Wirth, R. (2000). *CRISP-DM 1.0 Step-by-step data mining guide.*

Chauhan, A., Mishra, G., & Kumar, G. (2012). Survey on Data mining Techniques in Intrusion *Detection. Lap Lambert Academic Publ.*

Chow, C., & Liu, C. (1968). Approximating discrete probability distributions with dependence trees. *Information Theory, IEEE Transactions* on, 14(3), 462-467.

Clark, P., & Boswell, R. (2000). Practical Machine Learning Tools and Techniques with Java Implementation. *Morgan Kaufmann Publisher.*

Control Objectives for Information and related Technology (COBIT). (2013). Recuperado de http://www.isaca.org/cobit/ (recuperado el 8 de junio de 2013).

Cohen, W. W. (1995). Fast effective rule induction. *ICML, 95*, 115-123.

Crawford, K. D., & Wainwright, R. L. (1995). Applying Genetic Algorithms to Outlier Detection. *ICGA,* 546-550.

Denning, D., & Neumann, P. G. (1985). Requirements and Model for IDES-a Real-time Intrusion-detection Expert System: Final Report. *SRI International*.

Denning, D. E. (1987). An intrusion-detection model. Software Engineering, *IEEE Transactions* on, *2*, 222-232.

Deng, Q., & Mei, G. (2009). Combining self-organizing map and K-means clustering for detecting fraudulent financial statements. *Granular Computing, 2009, GRC'09. IEEE International Conference on,* 126-131.

Detecting hackers. (2013). (analyzing network traffic) by Poisson model measure.Available from: http://www.ensc.sfu.ca/people/grad/pwangf/IPSW_report.pdf.

Diaz, B., Moniche, L., & Morillas, A. (2006). A fuzzy clustering approach to the key sectors of the Spanish economy. *Economic Systems Research*, *18*(3), 299-318.

Eining, M. M., Jones, D. R., & Loebbecke, J. K. (1997). Reliance on decision aids: an examination of auditors' assessment of management fraud. *Auditing: A Journal of Practice and Theory*, 16(2), 1-19.

Elovici, Y., Kandel, A., Last, M., Shapira, B., Zaafrany, O., Schneider, M., & Friedman, M. (2004). Terrorist detection system. Knowledge *Discovery in Databases: PKDD* 2004, 540-542.

Emilio, D. P. N., & Piattini, M. G. (2003). Auditoría Informática: Un enfoque Práctico. (2a. ed., ampliada y revisada*). Editorial RA-MA,* 28-30.

Eskin, E., Arnold, A., Prerau, M., Portnoy, L., & Stolfo, S. (2002). A geometric framework for unsupervised anomaly detection. *Applications of data mining in computer security,* 77-101.

Ester, M., Kriegel, H. P., Sander, J., & Xu, X. (1996). A density-based algorithm for discovering clusters in large spatial databases with noise. *KDD, 96*, 226-231.

Estevez-Tapiador, J. M., Garcia-Teodoro, P., & Diaz-Verdejo, J. E. (2003). Stochastic protocol modeling for anomaly based network intrusion detection. *Information Assurance, 2003. IWIAS 2003. Proceedings.* First IEEE International Workshop on, 3-12.

Fanning, K. M., & Cogger, K. O. (1998). Neural network detection of management fraud using published financial data*. International Journal*

of Intelligent Systems in Accounting, Finance & Management, 7(1), 21-41.

Fayyad, U. M., Piatetsky-Shapiro, G., Smyth, P., & Uthurusamy, R. (1996). Advances in knowledge discovery and data mining. *MIT Press.*

Fawcett, T., & Provost, F. (1999). Activity monitoring: Noticing interesting changes in behavior. *Proceedings of the fifth ACM SIGKDD international conference on Knowledge discovery and data mining,* 53-62.

Fernández Pierna, J. A., Jin, L., Daszykowski, M., Wahl, F., & Massart, D. L. (2003). A methodology to detect outliers/inliers in prediction with PLS. *Chemometrics and intelligent laboratory systems*, *68*(1), 17-28.

Ferreyra, M. (2007). Powerhouse: Data Mining usando Teoría de la información. *Recuperado* de http://web.austral.edu.ar/images/contenido/facultad-ingenieria/2-Data_Mining_basado_Teoria_Informacion_Marcelo_Ferreyra.pdf (última visita 29/05/2013).

Fleck, D., & Duric, Z. (2009). Affine invariant-based classification of inliers and outliers for image matching. *Image Analysis and Recognition,* 268-277.

Foster, D. & Stine, R. (2004). Variable Selection in Data Mining: Building a Predictive Model for Bankruptcy. *Journal of American Statistical Association, 99*, 303-313.

Foss, A., & Zaïane, O. R. (2002). A parameterless method for efficiently discovering clusters of arbitrary shape in large datasets. In *Data Mining, 2002. ICDM 2003. Proceedings. 2002 IEEE International Conference on,* 179-186.

Frank, A. & Asuncion, A. (2012). *UCI Machine Learning Repository. Irvine, CA: University of California, School of Information and Computer Science. Recuperado de http://archive.ics.uci.edu/ml (*última visita 20/05/2012).

Freeman, J., & Skapura, D. (1991). Neural Networks: Algorithms, Applications, and Programming Techniques. *Computation and neural systems series. Addison-Wesley.*

Garcia Martinez, R., Servente, M., & Pasquini, D. (2003). *Sistemas Inteligentes*. *Bs. As., Argentina: Editorial Nueva Librería.*

Garcia-Teodoro, P., Diaz-Verdejo, J., Maciá-Fernández, G., & Vázquez, E. (2009). Anomaly-based network intrusion detection: Techniques, systems and challenges. *Computers & security, 28*(1), 18-28.

Gardner, M. W., & Dorling, S. R. (1998). Artificial neural networks (the multilayer perceptron)--a *review of applications in the atmospheric sciences.Atmospheric environment,* 32(14-15), *2627-2636.*

Garrity Edward J., O'Donnell Joseph B., Sanders GL. (2006). Continuos Auditing and Data Mining. *Encyclopedia of Data Warehousing and Mining*, 217-222.

Goldberg, D. E., & Holland, J. H. (1988). Genetic algorithms and machine learning. *Machine learning*, *3*(2), 95-99.

Goldberg, D. E. (1989). Genetic Algorithms in Search, Optimization and Machine Learning (1st ed.). *Boston, MA, USA: Addison-Wesley Longman Publishing Co., Inc.*

Goodman, S. E., Kirk, J. C., & Kirk, M. H. (2007). Cyberspace as a medium for terrorists. *Technological Forecasting and Social Change*, *74*(2), 193-210.

Green, B. P., & Choi, J. H. (1997). Assessing the risk of management fraud through neural network technology. *Auditing-A Journal Of Practice & Theory*,*16*(1), 14-28.

Grossberg, S. (1976). Adaptive pattern classification and universal recoding: I. Parallel development and coding of neural feature detectors. *Biological cybernetics*, *23*(3), 121-134.

Grubbs, F. E. (1969). Procedures for detecting outlying observations in samples. *Technometrics*, *11*(1), 1-21.

Guha, S., Rastogi, R., & Shim, K. (1998). CURE: an efficient clustering algorithm for large databases. In *ACM SIGMOD Record, 27*(2), 73-84.

Guha, S., Rastogi, R., & Shim, K. (2000). ROCK: A robust clustering algorithm for categorical attributes. *Information systems*, *25*(5), 345-366.

Hagan, M. T., Demuth, H. B., & Beale, M. H. (1996). Neural network design. *Boston: Pws Pub.*, 2-14.

Hawkins, D. M. (1980). Identification of outliers. *London: Chapman and Hall.,* 11.

Hawkins, S., He, H., Williams, G., & Baxter, R. (2002). Outlier detection using replicator neural networks. *Data Warehousing and Knowledge Discovery,* 170-180.

Hayes-Roth, F., Waterman, D., & Lenat, D. (1984). Building expert systems. *Retrieved from http://www.osti.gov/scitech/servlets/purl/7151220.*

Hilera, J. R., & Martinez, V. J. (1995). Redes neuronales artificiales: fundamentos, modelos y aplicaciones. *Addison-Wesley Iberoamericana.*

Hinton, G. E., & Sejnowski, T. J. (1986). Learning and relearning in Boltzmann machines. *MIT Press, Cambridge, Mass*, *1*, 282-317.

Hodge, V. J., & Austin, J. (2004). A survey of outlier detection methodologies. *Artificial Intelligence Review*, *22*(2), 85-126.

Hofmeyr, S. A., Forrest, S., & Somayaji, A. (1998). Intrusion detection using sequences of system calls. *Journal of computer security*, *6*(3), 151-180.

Holm, C. (2007). An Examination of Actual Fraud Cases With a Focus on the Auditor's Responsibility*. In European Accounting Association Annual Congress.*

Hopfield, J. J. (1982). Neural networks and physical systems with emergent collective computational abilities. *Proceedings of the national academy of sciences*, *79*(8), 2554-2558.

Huang, Z. (1998). Extensions to the k-means algorithm for clustering large data sets with categorical values. *Data Mining and Knowledge Discovery*, *2*(3), 283-304.

Inmon, W. H. (1996). The data warehouse and data mining. *Communications of the ACM*, *39*(11), 49-50.

ISACA (2013). Recuperado de http://www.isaca.org (recuperado el 10 de mayo de 2013).

Japkowicz, N., Myers, C., & Gluck, M. (1995). A novelty detection approach to classification. *IJCAI,* 518-523.

Jensen, F. V. (1996*).* An introduction to Bayesian networks. *London: UCL press,* 210.

Jensen, D., Rattigan, M., & Blau, H. (2003). Information awareness: A prospective technical assessment. *Proceedings of the ninth ACM*

SIGKDD international conference on Knowledge discovery and data mining, 378-387.

John, G. H. (1995). Robust Decision Trees: Removing Outliers from Databases. *KDD*, 174-179.

Johnson, R. R., & Kuby, P. (2008). Estadística elemental = Just the Essentials of Elementary Statistics. Lo esencial = The Essentials. *Cengage Learning México.*

Johnson, T., Kwok, I., & Ng, R. T. (1998). Fast Computation of 2-Dimensional Depth Contours. *In KDD,* 224-228.

Johnson, J. R. B. (2012). Detecting Emergent Terrorism Events: Finding Needles in Information Haystacks. *Intelligence and Security Informatics Conference (EISIC),* 2012 European, 5-6.

Jouan-Rimbaud, D., Bouveresse, E., Massart, D. L., & De Noord, O. E. (1999). Detection of prediction outliers and inliers in multivariate calibration. *Analytica chimica acta*, *388*(3), 283-301.

Karypis, G., Han, E. H., & Kumar, V. (1999). Chameleon: Hierarchical clustering using dynamic modeling. *Computer*, *32*(8), 68-75.

Kaufman, L., & Rousseeuw, P. J. (2009*). Finding groups in data: an introduction to cluster analysis, 344*.

Kim, M. J., & Han, I. (2003). The discovery of experts' decision rules from qualitative bankruptcy data using genetic algorithms. *Expert Systems with Applications*, *25*(4), 637-646.

Kirkos, S., & Manolopoulos, Y. (2004). Data mining in finance and accounting: a review of current research trends. *Proceedings of the 1st international conference on enterprise systems and accounting* (ICESAcc), 63-78.

Kirkos, E., Spathis, C., & Manolopoulos, Y. (2007). Data mining techniques for the detection of fraudulent financial statements*. Expert Systems with Applications*, *32*(4), 995-1003.

Knorr, E. M., & Ng, R. T. (1997, November). A unified approach for mining outliers. *Proceedings of the 1997 conference of the Centre for Advanced Studies on Collaborative research,* 11.

Knorr, E. M., & Ng, R. T. (1998). Algorithms for mining distance-based outliers in large datasets. *Proceedings of the International Conference on Very Large Data Bases*.

Knorr, E. M., Ng, R. T., & Tucakov, V. (2000). Distance-based outliers: algorithms and applications. *The VLDB Journal—The International Journal on Very Large Data Bases*, *8*(3-4), 237-253.

Koh, H. C., & Low, C. K. (2004). Going concern prediction using data mining techniques. *Managerial Auditing Journal, 19*(3), 462-476.

Kohonen, T. (1990). The self-organizing map*. Proceedings of the IEEE, 78*(9), 1464-1480.

Kosko, B. (1988). Bidirectional associative memories. *Systems, Man and Cybernetics, IEEE Transactions on*, *18*(1), 49-60.

Koskivaara, E. (2004). Artificial neural networks in analytical review procedures. *Managerial Auditing Journal*, *19*(2), 191-223.

Kumar, V. (2005). Parallel and distributed computing for cybersecurity. *Distributed Systems Online, IEEE*, *6*(10).

Kuna, H., García Martínez, R., Villatoro, F. (2009). Identificación de Causales de Abandono de Estudios Universitarios. Uso de Procesos de Explotación de Información. *Revista Iberoamericana de Tecnología en Educación y Educación en* Tecnología, *5*, 39-44.

Kuna, H., García-Martínez, R. Villatoro, F. (2010a). Pattern Discovery in University Students Desertion Based on Data Mining. In *Advances and Applications in Statistical Sciences* Journal, *2*(2): 275-286.

Kuna, H., Caballero, S., Rambo, A., Meinl, E., Steinhilber, A., Pautch, G., García-Martínez, R., Villatoro, F. (2010b). Avances en procedimientos de la explotación de información para la identificación de datos faltantes, con ruido e inconsistentes. *Proceedings XII Workshop de Investigadores en Ciencias de la Computación,* 137-141.

Kuna, H., Caballero, S.. Rambo, A., Meinl, E., Steinhilber, A., Pautch, G., Rodríguez, D., García-Martínez, R., Villatoro, F. (2010c). Identification of Noisy Data in Databases by Means of a Clustering Process. *Ingeniería de Software e Ingeniería del Conocimiento: Tendencias de Investigación e Innovación Tecnológica en Iberoamérica,* 264-273.

Kuna, H., Pautsch, G., Rey, M., Cuba, C., Rambo, A., Caballero, S., Steinhilber, A., García-Martínez, R., Villatoro, F. (2011). Avances en procedimientos de la explotación de información con algoritmos basados en la densidad para la identificación de outliers en bases de datos. *Proceedings XIII Workshop de Investigadores en Ciencias de la Computación.* Artículo 3745.

Kuna, H., García-Martinez, R., Villatoro, F. (2012a). Automatic Outliers Fields Detection in Databases. In *Journal of Modelling and Simulation of Systems*, *3*(1), 14-20.

Kuna, H., Rambo, A., Caballero, S., Pautsch, G., Rey, M., Cuba, C., García-Martínez, R., Villatoro, F. (2012b). Procedimientos para la identificación de datos anómalos en bases de datos. In *Proceedings of CONISOFT,* 184-193.

Kuna, H. , Pautsch, G., Rey, M., Cuba, C., Rambo, A., Caballero, S., García-Martínez, R., Villatoro, F. (2012c). Comparación de la efectividad de procedimientos de la explotación de información para la identificación de outliers en bases de datos. *Proceedings del XIV Workshop de Investigadores en Ciencias de la Computación,* 296-300.

Kuna, H., Villatoro, F., García-Martinez, R. (2013a). Development and Comparison of Procedures for Outlier Detection in Databases. *Computers & Security*. (en evaluación).

Kuna, H., Pautsch, G., Rambo, A., Rey, M., Cortes, J., Rolón, S. (2013b). Procedimiento de explotación de información para la identificación de campos anómalos en base de datos alfanuméricas. R*evista Latinoamericana de Ingeniería de Software*, *1*(3): 102-106.

Kuna, H., García-Martínez, R., Villatoro, F. (2014). Outlier detection in audit logs for application systems. *Information Systems*, http://dx.doi.org/10.1016/j.is.2014.03.001i.

Lam, M. (2004). Neural network techniques for financial performance prediction: integrating fundamental and technical analysis. *Decision Support Systems, 37*(4), 567-581.

Larose, D. T. (2005). Discovering knowledge in data: an introduction to data mining. *Wiley. com.*

Last, M., Markov, A., & Kandel, A. (2006). Multi-lingual detection of terrorist content on the web. *In Intelligence and Security Informatics,* 16-30. Springer Berlin Heidelberg.

Lauría, E. J. M., & Duchessi, P. J. (2006). A Bayesian Belief Network for IT implementation decision support. *Decision Support Systems*, *42*(3), 1573–1588.

Leung, K., & Leckie, C. (2005). Unsupervised anomaly detection in network intrusion detection using clusters. In *Proceedings of the Twenty-eighth Australasian conference on Computer Science-Volume 38,* 333-342.

Leardi, R. (1994). Application of a genetic algorithm to feature selection under full validation conditions and to outlier detection. *Journal of Chemometrics*, *8*(1), 65-79.

Lin, F. Y., & McClean, S. (2001). A data mining approach to the prediction of corporate failure. K*nowledge-Based Systems, 14*(3), 189-195.

Lu, C. T., Chen, D., & Kou, Y. (2003). Algorithms for spatial outlier detection. In *Data Mining, 2003. ICDM 2003. Third IEEE International Conference on,* 597-600.

MacQueen, J. (1967). Some methods for classification and analysis of multivariate observations. In *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability, 1*(281-297),14.

Mansur, M. O., Sap, M., & Noor, M. (2005). Outlier Detection Technique in Data Mining: A Research Perspective. *Recuperado* de http://eprints.utm.my/3336/ (última visita 29/09/2013).

Marsland, S. (2003). Novelty detection in learning systems. *Neural computing surveys*, *3*(2), 157-195.

Memon, N., Hicks, D. L., & Larsen, H. L. (2007). How investigative data mining can help intelligence agencies to discover dependence of nodes in terrorist networks. *Advanced Data Mining and Applications,* 430-441.

Memon, N., Qureshi, A. R., Wiil, U. K., & Hicks, D. (2009). Notice of Violation of IEEE Publication Principles: Novel Algorithms for Subgroup Detection in Terrorist Networks. *Recuperado* de http://forskningsbasen.deff.dk/Share.external?sp=S748da970-d25f-11dd-a016-000ea68e967b&sp=Saau (última visita 29/08/2013).

Michalski, R. S., Bratko, I., & Bratko, A. (1998). Machine Learning and Data Mining; Methods and Applications*. John Wiley & Sons, Inc..*

Muñoz, A., & Muruzábal, J. (1998). Self-organizing maps for outlier detection. *Neurocomputing*, *18*(1), 33-60.

Nairac, A., Townsend, N., Carr, R., King, S., Cowley, P., & Tarassenko, L. (1999). A system for the analysis of jet engine vibration data. *Integrated Computer-Aided Engineering*, *6*(1), 53-66.

Narendra, K. S., & Thathachar, M. A. (1974). Learning automata-a survey.*Systems, Man and Cybernetics, IEEE Transactions on*, (4), 323-334.

Ng, R. T., & Han, J. (2002). CLARANS: A method for clustering objects for spatial data mining. *Knowledge and Data Engineering, IEEE Transactions on*,*14*(5), 1003-1016.

Ngai, E. W. T., Hu, Y., Wong, Y. H., Chen, Y., & Sun, X. (2011). The application of data mining techniques in financial fraud detection: A classification framework and an academic review of literature. *Decision Support Systems, 50*(3), 559-569.

Nguyen, D. H., & Widrow, B. (1990). Neural networks for self-learning control systems. *Control Systems Magazine*, IEEE, *10*(3), 18-23.

Ni, W., Chen, G., Lu, J., Wu, Y., & Sun, Z. (2008). Local entropy based weighted subspace outlier mining algorithm. *Journal of Computer Research and Development*, *45*(7), 1189-1192.

Niu, Z., Shi, S., Sun, J., & He, X. (2011). A survey of outlier detection methodologies and their applications. *Artificial Intelligence and Computational Intelligence,* 380-387.

Ozgul, F., Bondy, J., & Aksoy, H. (2007). Mining for offender group detection and story of a police operation. *Proceedings of the sixth Australasian conference on Data mining and analytics, 70*, 189-193.

Qin, J., Zhou, Y., Reid, E., Lai, G., & Chen, H. (2007). Analyzing terror campaigns on the internet: Technical sophistication, content richness, and Web interactivity. *International Journal of Human-Computer Studies, 65*(1), 71-84.

Quinlan, J. R. (1986). Induction of decision trees. *Machine Learning*, *1*, 81-106.

Quinlan, J. R. (1993). C4.5: programs for machine learning. *Morgan Kaufmann*.

Quinlan, J. R. (1999). Simplifying decision trees. *International Journal of Human-Computer Studies, 51*(2), 497-510.

Papadimitriou, S., Kitagawa, H., Gibbons, P. B., & Faloutsos, C. (2003). Loci: Fast outlier detection using the local correlation integral. In *Data Engineering, 2003. Proceedings. 19th International Conference on,* 315-326.

Pautsch, J., Kuna, H., & Godoy, A. E. (2011). Resultados preliminares del proceso de minería de datos aplicado al análisis de la deserción en carreras de informática utilizando herramientas open source. In *Proceedings of XVII Congreso Argentino de Ciencias de la Computación.*

Pearl, J. (1988). Probabilistic reasoning in intelligent systems: networks of plausible inference. *Morgan Kaufmann.*

Penny, K. I., & Jolliffe, I. T. (2001). A comparison of multivariate outlier detection methods for clinical laboratory safety data. *Journal of the Royal Statistical Society: Series D (The Statistician)*, *50*(3), 295-307.

Peña, D. (2002). *Análisis de datos multivariantes*. Madrid: *McGraw-Hill/Interamericana de España*.

Petrovskiy, M. (2003). A fuzzy kernel-based method for real-time network intrusion detection. In *Innovative Internet Community Systems,* 189-200.

Portnoy, L., Eskin, E., & Stolfo, S. (2001). Intrusion detection with unlabeled data using clustering. *In Proceedings of ACM CSS Workshop on Data Mining Applied to Security* (DMSA-2001).

Pyle, D. (2003). Business modeling and data mining. *Morgan Kaufmann.*

Rabiner, L. R. (1989). A tutorial on hidden Markov models and selected applications in speech recognition. *Proceedings of the IEEE*, *77*(2), 257-286.

Rebane, G., & Pearl, J. (2013). The recovery of causal poly-trees from statistical data. *arXiv preprint arXiv*:1304.2736.

Reimann, C., Filzmoser, P., & Garrett, R. G. (2005). Background and threshold: critical comparison of methods of determination. *Science of the Total Environment*, *346*(1), 1-16.

Rezaee, Z., Sharbatoghlie, A., Elam, R., & McMickle, P. L. (2002). Continuous auditing: Building automated auditing capability. *Auditing: A Journal of Practice & Theory*, *21*(1), 147-163.

Rivas, G. A. (1989). Auditoría informática. *Ediciones Díaz de Santos.*

Roesch, M. (1999, November). Snort: Lightweight Intrusion Detection for Networks. *LISA, 99*, 229-238.

Rosenblatt, F. (1958). The perceptron: a probabilistic model for information storage and organization in the brain. Psychological *review*, *65*(6), 386.

Rousseeuw, P. J., & Leroy, A. M. (2005). *Robust regression and outlier detection,* 589.

Rustum, R., & Adeloye, A. J. (2007). Replacing outliers and missing values from activated sludge data using Kohonen self-organizing map. *Journal of Environmental Engineering*, *133*(9), 909-916.

Statement of Auditing Practice (SAP). Recuperado de http://www.eisi.ues.edu.sv/jportillo/taac.doc (recuperado el 22 de junio de 2012).

Saad, M. K., & Hewahi, N. M. (2009). A comparative study of outlier mining and class outlier mining. *COMPUTER SCIENCE LETTERS*, *1*(1).

Shaikh, M. A., Wang, J., Liu, H., & Song, Y. (2007). Investigative data mining for counterterrorism. *Advances in Hybrid Information Technology,* 31-41.

Schaffer, C. (1994, July). A Conservation Law for Generalization Performance. In *ICML, 94*, 259-265.

Scheff, K., & Szu, H. (1987). 1-D optical Cauchy machine infinite film spectrum search. In *Proceedings of the IEEE First International Conference on Neural Networks, 52*, 673-680.

Schiefer, J., Jeng, J. J., Kapoor, S., & Chowdhary, P. (2004, July). Process information factory: a data management approach for enhancing business process intelligence. *In e-Commerce Technology, 2004. CEC 2004. Proceedings. IEEE International Conference* on, 162-169.

Schölkopf, B., Platt, J. C., Shawe-Taylor, J., Smola, A. J., & Williamson, R. C. (2001). Estimating the support of a high-dimensional distribution. *Neural computation*, *13*(7), 1443-1471.

Shah, H., Undercoffer, J., & Joshi, A. (2003, May). Fuzzy clustering for intrusion detection. In Fuzzy Systems, 2003. *FUZZ'03. The 12th IEEE International Conference* on, *2*, 1274-1278.

Shannon, C. E. (2001). A mathematical theory of communication. *ACM SIGMOBILE Mobile Computing and Communications Review*, *5*(1), 3-55.

Shin, K. S., & Lee, Y. J. (2002). A genetic algorithm application in bankruptcy prediction modeling*. Expert Systems with Applications*, *23*(3), 321-328.

SIGEN. (2002). Resolucion 152/2002-sgn. Retrieved from http://www.sigen.gov.ar/normativa/pdfs/r152_2002_SGN.pdf

Smyth, P. (1994). Markov monitoring with unknown states. *Selected Areas in Communications, IEEE Journal on*, *12*(9), 1600-1612.

Sykacek, P. (1997, April). Equivalent error bars for neural network classifiers trained by Bayesian inference. In *ESANN*.

Sivanandam, S. N., & Deepa, S. N. (2007). Introduction to Genetic Algorithms (1st ed.). *Springer Publishing Company, Incorporated.*

Spathis, C. T. (2002). Detecting false financial statements using published data: some evidence from Greece. *Managerial Auditing Journal*, *17*(4), 179-191.

Spencer, G., Shirley, P., Zimmerman, K., & Greenberg, D. P. (1995). Physically-based glare effects for digital images. *In Proceedings of the 22nd annual conference on Computer graphics and interactive techniques,* 325-334.

Spence, C., Parra, L., & Sajda, P. (2001). Detection, synthesis and compression in mammographic image analysis with a hierarchical image probability model. *In Mathematical Methods in Biomedical Image Analysis,* 2001. MMBIA 2001. IEEE Workshop on, 3-10).

Subversion (s.f.). Recuperado de http://subversion.tigris.org/ (recuperado el 22 de junio de 2008).

Staniford-Chen, S., Tung, B., & Schnackenberg, D. (1998). The common intrusion detection framework (CIDF). *Proceedings of the information survivability workshop*.

Tan, C. N., & Dihardjo, H. (2001). A study of using artificial neural networks to develop an early warning predictor for credit union financial distress with comparison to the probit model. *Managerial Finance*, *27*(4), 56-77.

Tan, P. N. (2007). Introduction to data mining. *Pearson Education India.*

Tang, J., Chen, Z., Fu, A. W. C., & Cheung, D. W. (2002). Enhancing effectiveness of outlier detections for low density patterns. *Advances in Knowledge Discovery and Data Mining,* 535-548.

Teng, H. S., Chen, K., & Lu, S. C. (1990). Adaptive real-time anomaly detection using inductively generated sequential patterns. *Research in Security and Privacy, 1990. Proceedings., 1990 IEEE Computer Society Symposium on,* 278-284.

Thuraisingham, B. (2009). Data mining for malicious code detection and security applications. *Web Intelligence and Intelligent Agent Technologies, 2009. WI-IAT'09. IEEE/WIC/ACM International Joint Conferences* on, *2*, 6-7).

Tung, W. L., Quek, C., & Cheng, P. (2004). GenSo-EWS: a novel neural-fuzzy based early warning system for predicting bank failures. *Neural Networks*,17(4), 567-587.

Van der Aalst, W. M., & Van Dongen, B. F. (2002). Discovering workflow performance models from timed logs. *In Engineering and Deployment of Cooperative Information Systems,* 45-63.

Van der Aalst, W. M., & Song, M. (2004). Mining Social Networks: Uncovering interaction patterns in business processes. *Business Process Management,* 244-260.

Van der Aalst, W. M., & de Medeiros, A. K. A. (2005). Process mining and security: Detecting anomalous process executions and checking process conformance*. Electronic Notes in Theoretical Computer Science*, *121*, 3-21.

Warrender, C., Forrest, S., & Pearlmutter, B. (1999). Detecting intrusions using system calls: Alternative data models. In Security and Privacy, 1999. *Proceedings of the 1999 IEEE Symposium* on, 133-145.

Williams, G., Baxter, R., He, H., Hawkins, S., & Gu, L. (2002). A comparative study of RNN for outlier detection in data mining. In *Data Mining, 2002. ICDM 2003. Proceedings. 2002 IEEE International Conference on,* 709-712.

Wu, S. X., & Banzhaf, W. (2010). The use of computational intelligence in intrusion detection systems: A review. *Applied Soft Computing*, *10*(1), 1-35.

Xue, A., Duan, X., Ma, H., Chen, W., & Ju, S. (2008). Privacy preserving spatial outlier detection. *In Young Computer Scientists, 2008. ICYCS 2008. The 9th International Conference,* 714-719).

Ye, N., Emran, S. M., Chen, Q., & Vilbert, S. (2002). Multivariate statistical analysis of audit trails for host-based intrusion detection. *Computers, IEEE Transactions* on, *51*(7), 810-820.

Yoursi, N. A. (2010). A validity index for outlier detection. *Intelligent Systems Design and Applications* (ISDA), 2010 10th International Conference on, 325-329.

Zadeh, L. A. (1988). Fuzzy logic. *Computer*, *21*(4), 83-93.

Zhang, T., Ramakrishnan, R., & Livny, M. (1997). BIRCH: A new data clustering algorithm and its applications. *Data Mining and Knowledge Discovery*, *1*(2), 141-182.

Zhang, Y., & Lee, W. (2000, August). Intrusion detection in wireless ad-hoc networks. *Proceedings of the 6th annual international conference on Mobile computing and networking,* 275-283.

Zhang, J., & Zulkernine, M. (2006). Anomaly based network intrusion detection with unsupervised outlier detection. *Communications, 2006. ICC'06. IEEE International Conference on, 5*, 2388-2393).

Zhang, Y., Meratnia, N., & Havinga, P. J. M. (2007). A taxonomy framework for unsupervised outlier detection techniques for multi-type data sets.

Zhang, K., Hutter, M., & Jin, H. (2009). A new local distance-based outlier detection approach for scattered real-world data. *Advances in Knowledge Discovery and Data Mining*, 813-822.

Zhou, Z. H., & Jiang, Y. (2003). Medical diagnosis with C4.5 rule preceded by artificial neural network ensemble. *Information Technology in Biomedicine, IEEE Transactions on*, *7*(1), 37-42.