



Universidad de San Carlos de Guatemala
Facultad de Ingeniería
Escuela de Ingeniería en Ciencias y Sistemas

**DISEÑO DE UN SISTEMA DE APOYO PARA LA TOMA DE DECISIONES EN
UN SISTEMA CRM UTILIZANDO LÓGICA DIFUSA**

Raúl Antonio Mijangos Castro

Asesorado por: Inga. Marcela Elizabeth Velásquez Miranda

Guatemala, febrero de 2007

UNIVERSIDAD DE SAN CARLOS DE GUATEMALA



FACULTAD DE INGENIERÍA

**DISEÑO DE UN SISTEMA DE APOYO PARA LA TOMA DE DECISIONES EN
UN SISTEMA CRM UTILIZANDO LÓGICA DIFUSA**

TRABAJO DE GRADUACIÓN

PRESENTADO A LA JUNTA DIRECTIVA DE LA
FACULTAD DE INGENIERÍA

POR

RAÚL ANTONIO MIJANGOS CASTRO

ASESORADO POR: INGA. MARCELA ELIZABETH VELÁSQUEZ MIRANDA
AL CONFERÍRSELE EL TÍTULO DE
INGENIERO EN CIENCIAS Y SISTEMAS

GUATEMALA, FEBRERO DE 2007

UNIVERSIDAD DE SAN CARLOS DE GUATEMALA
FACULTAD DE INGENIERÍA



NÓMINA DE JUNTA DIRECTIVA

DECANO	Ing. Murphy Olympo Paiz Recinos
VOCAL I	Inga. Glenda Patricia García Soria
VOCAL II	Ing. Alba Guerrero de López
VOCAL III	Ing. Miguel Ángel Dávila Calderón
VOCAL IV	Br. Kenneth Issur Estrada Ruiz
VOCAL V	Br. Elisa Yazminda Vides Leiva
SECRETARIA	Inga. Marcia Ivonne Véliz Vargas

TRIBUNAL QUE PRACTICÓ EL EXAMEN GENERAL PRIVADO

DECANO	Ing. Murphy Olympo Paiz Recinos
EXAMINADORA	Inga. Virginia Victoria Tala Ayerdi
EXAMINADOR	Ing. Javier Gramajo López
EXAMINADOR	Ing. César Augusto Fernández Cáceres
SECRETARIA	Inga. Marcia Ivonne Véliz Vargas

HONORABLE TRIBUNAL EXAMINADOR

Cumpliendo con los preceptos que establece la ley de la Universidad de San Carlos de Guatemala, presento a su consideración mi trabajo de graduación titulado:

**DISEÑO DE UN SISTEMA DE APOYO PARA LA TOMA DE DECISIONES EN
UN SISTEMA CRM UTILIZANDO LÓGICA DIFUSA,**

tema que me fuera asignado por la Dirección de la Escuela de Ciencias y Sistemas, de la Facultad de Ingeniería, con fecha marzo de 2005.

Raúl Antonio Mijangos Castro

ACTO QUE DEDICO A:

Dios,	porque sin su voluntad nada es posible.
Mis padres	por su cariño, empeño y apoyo en todo sentido.
Mis hermanas	por su cariño y apoyo moral.
Angélica	porque quiero ser una motivación en su vida.
Gladys	por su cariño, ejemplo y enseñanzas.
Los lectores	quienes le dan sentido a este trabajo.

“Toda máquina es tan inteligente como la persona que la programa, configura o enseña, nunca más que su creador.”

Raúl Mijangos

ÍNDICE GENERAL

ÍNDICE DE ILUSTRACIONES.....	V
LISTA DE SÍMBOLOS.....	IX
GLOSARIO.....	XI
RESUMEN.....	XIII
OBJETIVOS.....	XV
INTRODUCCIÓN.....	XVII

1. RELACIÓN ENTRE CRM Y LÓGICA DIFUSA

1.1. Técnicas de inteligencia artificial.....	1
1.1.1. Lógica difusa.....	2
1.1.1.1. Dimensión difusa.....	3
1.1.1.2. Atributos difusos.....	3
1.1.1.3. Pertenencia.....	4
1.1.1.4. Funciones de pertenencia.....	4
1.1.1.5. Operadores difusos.....	6
1.1.1.6. Conjuntos difusos.....	6
1.1.1.7. Reglas difusas.....	7
1.1.1.8. Grado de soporte.....	8
1.1.1.9. Determinación de conjuntos difusos.....	8
1.1.1.10. Compensación.....	9
1.1.1.11. Conversión de valores difusos.....	10
1.1.2. Otras técnicas de Inteligencia Artificial.....	11
1.1.2.1. Redes neuronales.....	11
1.1.2.2. Algoritmos genéticos.....	11
1.2. Metodología CRM.....	12

1.2.1.	Descripción y enfoque.....	13
1.2.2.	Elementos de CRM.....	15
1.2.2.1.	Centro de llamadas.....	15
1.2.2.2.	Centro de contactos.....	15
1.2.2.3.	Almacén de datos.....	15
1.2.2.4.	Fuerza de ventas.....	16
1.2.2.5.	Gestión del conocimiento.....	16
1.2.2.6.	Gestión de la información para ejecutivos.....	16
1.3.	Clasificación de los clientes.....	16
1.3.1.	Madurez de los clientes.....	17
1.3.1.1.	Nivel inicial.....	17
1.3.1.2.	Nivel adolescente.....	17
1.3.1.3.	Nivel adulto.....	18
1.3.1.4.	Nivel de vejez.....	18
1.3.2.	Segmentación.....	18
1.3.3.	Pirámides de clientes.....	19
1.3.3.1.	Población general.....	20
1.3.3.2.	Clientes probables.....	21
1.3.3.3.	Clientes potenciales.....	21
1.3.3.4.	Clientes inactivos.....	21
1.3.3.5.	Clientes activos.....	21
1.4.	¿Por qué técnicas de inteligencia artificial?.....	22
1.4.1.	Métodos clásicos de apoyo en la toma de decisiones.....	22
1.4.2.	Técnicas de inteligencia artificial.....	23

2. VARIABLES A UTILIZAR EN EL MODELO

2.1.	Variables de entrada.....	25
2.1.1.	Variables de clientes.....	26
2.1.2.	Variables de productos y promociones.....	29

2.2.	Variables de salida.....	29
2.2.1.	Variables de clientes.....	30
2.2.2.	Variables de promociones.....	30
2.3.	Almacenamiento de las variables de entrada.....	30
2.4.	Funciones de pertenencia.....	38
3.	DETERMINACIÓN DE REGLAS	
3.1.	Definición de variables de salida.....	43
3.2.	Detalle de reglas.....	44
3.2.1.	Asignación de atributos de salida.....	47
3.2.2.	Asignación de grados de soporte.....	48
4.	DETERMINACIÓN DE LOS CONJUNTOS DIFUSOS	
4.1.	Determinación de pertenencia.....	51
4.2.	Agregación.....	59
4.3.	Compensación.....	64
4.4.	Composición.....	70
5.	ANÁLISIS Y PRESENTACIÓN DE RESULTADOS	
5.1.	Conversión de conjuntos difusos a valores numéricos.....	77
5.2.	Interpretación de resultados.....	78
6.	CONSIDERACIONES PARA LA IMPLEMENTACIÓN	
6.1.	Sistemas sugeridos.....	81
6.2.	Relación con otros sistemas.....	82
6.3.	Interfaz de entrada.....	82
6.4.	Interfaz de salida.....	84
6.5.	Consideraciones finales.....	85

CONCLUSIONES.....87
RECOMENDACIONES.....89
BIBLIOGRAFÍA.....91

ÍNDICE DE ILUSTRACIONES

FIGURAS

1	Diagrama conceptual de conjuntos difusos	7
2	Diagrama conceptual de las reglas difusas	8
3	Diagrama conceptual del proceso difuso	10
4	Ejemplo de pirámide de clientes	20
5	Modelo de base de datos para el sistema	31
6	Funciones de pertenencia de los atributos de la dimensión SUELDO	41
7	Funciones de pertenencia de los atributos de la dimensión EDAD	42
8	Definición de dimensiones	83
9	Definición de atributos por dimensión	83
10	Definición de reglas	84

TABLAS

I	Datos de la tabla CLIENTE – parte 1	28
II	Datos de la tabla CLIENTE – parte 2	28
III	Datos de las dimensiones ejemplo	34
IV	Datos de la regla ejemplo	34
V	Asociación entre las dimensiones y la regla	35
VI	Código para la carga de datos	35

VII	Selección de dimensiones que están en la regla solicitada	36
VIII	Carga de datos de las dimensiones de entrada	37
IX	Datos de las dimensiones cargados en la tabla de procesamiento	38
X	Funciones de pertenencia expresadas por medio de fórmulas	39
XI	Tipos de función usados en el sistema	40
XII	Datos de los atributos de las dimensiones ejemplo	40
XIII	Código para determinar el detalle de reglas	45
XIV	Código a agregar cuando hay más de dos dimensiones	46
XV	Atributos de entrada combinados	46
XVI	Valores de salida correspondientes a cada pareja de los datos de entrada	47
XVII	Detalle de reglas con grado de soporte	49
XVIII	Datos de la tabla DETALLE_REGLA	50
XIX	Datos de la tabla DETALLE_REGLA_ATRIBUTO	50
XX	Código de consulta de nombres para los atributos de la dimensión SUELDO	52
XXI	Nombres para los atributos de la dimensión SUELDO	52
XXII	Código para agregar los campos de los atributos de la dimensión SUELDO	53
XXIII	Código de consulta de nombres para los atributos de la dimensión EDAD	53
XXIV	Nombres para los atributos de la dimensión EDAD	54
XXV	Código para agregar los campos de los atributos de la dimensión EDAD	54
XXVI	Estructura de la tabla luego de agregarle las columnas de los atributos de entrada	55
XXVII	Inserción de los datos de la columna SUELDO_ALTO	56
XXVIII	Inserción de los datos de la columna SUELDO_MEDIO	56

XXIX	Inserción de los datos de la columna SUELDO_BAJO	57
XXX	Inserción de los datos de la columna EDAD_JOVEN	57
XXXI	Inserción de los datos de la columna EDAD_MEDIA	58
XXXII	Inserción de los datos de la columna EDAD_AVANZADO	58
XXXIII	Valores de los grados de pertenencia insertados	59
XXXIV	Creación y llenado de la tabla DETALLE_REGLA_DOC	60
XXXV	Declaración de variables del proceso de agregación	61
XXXVI	Código de consulta y asignación de registros	61
XXXVII	Validación de carga de registros	62
XXXVIII	Llenado de la tabla temporal	63
XXXIX	Datos obtenidos con el proceso de agregación	64
XL	Declaración de variables del proceso de compensación	65
XLI	Código de consulta de información y asignación de registros	66
XLII	Validación de carga de registros	66
XLIII	Código del proceso de compensación	68
XLIV	Código de la función de compensación	69
XLV	Datos obtenidos con el proceso de compensación	70
XLVI	Código para agregar los campos para el proceso de composición	71
XLVII	Estructura de la tabla luego de agregar los campos para la composición	71
XLVIII	Declaración de variables del proceso de composición	72
XLIX	Código de consulta de información y asignación de registros	73
L	Validación de carga de registros	73
LI	Código del proceso de composición	74
LII	Datos obtenidos con el proceso de composición	75
LIII	Actualización de valores resultado	78
LIV	Datos obtenidos con la actualización	78
LV	Código para evaluar la aceptación	79

LISTA DE SÍMBOLOS

L	Representación del tipo de función de pertenencia, que tiene su valor significativo más a la derecha.
Γ	Representación del tipo de función de pertenencia, que tiene su valor significativo más a la izquierda.
Λ	Representación del tipo de función de pertenencia, que tiene su valor significativo definido en un sólo punto, y tiene los valores de otros atributos tanto a la derecha como a la izquierda.
Π	Representación del tipo de función de pertenencia, que tiene su valor significativo definido sobre un intervalo, y tiene los valores de otros atributos tanto a la derecha como a la izquierda. Representación de la serie multiplicativa desde un límite inferior a otro superior, de los valores que se encuentran con él.
γ	Representa el grado de compensación entre dos o más atributos, que se elige arbitrariamente para ser utilizado en el proceso de compensación.
μ	Representación del grado de pertenencia, de un valor determinado a un atributo difuso.

GLOSARIO

Algoritmo	Conjunto ordenado y finito de operaciones, que permite hallar la solución de un problema.
Compensación	Acción de igualar en opuesto sentido, el efecto de una cosa con el de otra.
Decisión	Determinación, resolución que se toma o se da en una cosa dudosa.
Inteligencia artificial	Desarrollo y utilización de ordenadores con los que se intenta reproducir los procesos de la inteligencia humana.
<i>Marketing</i>	Conjunto de principios y prácticas que buscan el aumento del comercio, especialmente de la demanda.
Metodología	Conjunto de métodos que se siguen en una investigación científica, o en una exposición doctrinal.
Pertenencia	Hecho o circunstancia de formar parte de un conjunto, como una clase, un grupo, una comunidad, una institución, etc.
Regla	Conjunto de operaciones que deben llevarse a cabo para realizar una inferencia o deducción correcta.

RESUMEN

Una de las metodologías utilizadas, desde hace ya varios años, para el *marketing* relacional ha sido CRM, pero con el avance de la tecnología ha podido tener una mayor utilización, especialmente gracias a los avances en el área de informática.

Se puede decir que CRM es conocer al cliente, entender sus gustos y prever sus necesidades, es por ello que siendo el cliente el corazón de esta metodología, se busca darle apoyo a la toma de decisiones en cuanto a clientes y promociones de productos.

Mientras tanto, por medio de las técnicas de inteligencia artificial, se han desarrollado aplicaciones que permiten imitar el razonamiento humano, aportando además la velocidad en el procesamiento de datos y la persistencia de la información. Entre estas técnicas se encuentra la lógica difusa, llamada también lógica borrosa, la cual permite definir criterios de selección que van más allá de la lógica clásica, la cual tiene sólo dos valores posibles, falso y verdadero. La lógica difusa permite asignarle valores de verdad o pertenencia a determinada categoría, para cada elemento a evaluar.

En el presente trabajo se diseña un sistema que funciona utilizando la lógica difusa, el cual consiste en definir variables lingüísticas, y para cada una un conjunto, de por lo menos tres y no más de siete términos, los que equivalen a las categorías antes mencionadas; para cada término se define una función de pertenencia, y luego se encuentran los valores de pertenencia para cada

valor según estas funciones. Cuando las entradas ya tienen su valor de pertenencia, se usan operaciones difusas para determinar el valor de las reglas, previamente definidas, con ello se obtiene el valor de verdad o pertenencia de la regla. De ser necesario se deben resolver los conflictos entre las reglas.

Finalmente, los valores encontrados para las reglas deben ser transformados a valores numéricos para ser mostrados al usuario, y que éste pueda interpretarlos.

El diseño de un sistema de apoyo a la toma de decisiones sobre la metodología CRM usando lógica difusa, es la base para la implementación de una aplicación que dará al usuario la facilidad de manejar grandes cantidades de información y minimizar el tiempo de análisis, utilizando un criterio no tan cerrado como el de la lógica clásica.

OBJETIVOS

GENERAL

Diseñar un sistema de apoyo para la toma de decisiones en un sistema CRM, utilizando lógica difusa.

ESPECÍFICOS

1. Describir el área de CRM que se busca apoyar en este proyecto.
2. Determinar clasificaciones de clientes y su utilidad en los negocios.
3. Describir las variables a utilizar en el modelo, tanto las de entrada como las de salida, indicando sus funciones de pertenencia.
4. Determinar las reglas del modelo, explicando las variables de salida en función de las de entrada.
5. Realizar el proceso de determinación de conjuntos difusos, para las variables de entrada del modelo.
6. Presentar los resultados del proceso de conversión de conjuntos difusos a valores numéricos.

INTRODUCCIÓN

El ser humano se ha enfrentado desde siempre, en todos los momentos de su vida, a situaciones en las cuales ha tenido que tomar decisiones, ya sea para decidir en qué lugar establecerse como cuando era nómada, para elegir el mejor punto de ataque durante la guerra, para escoger a qué caballo apostarle en las carreras, para elegir su carrera profesional, o para escoger a la persona con la que pasará el resto de su vida. En todos los casos debía seleccionar entre varias opciones (tomar una decisión), y para ello hizo uso de la experiencia, la intuición, algún método definido, el apoyo de terceros, o en algunos casos la combinación de varias de estas formas de tomar decisiones.

Actualmente, la mayoría de decisiones empresariales van dirigidas al cliente, tal como lo dice el eslogan “el cliente es lo primero”. Ello significa enfocar los esfuerzos de la empresa sobre el cliente, no sobre el producto. A raíz de ello se han desarrollado metodologías orientadas a la gestión de clientes, una de éstas metodologías es la Administración de relaciones con el cliente; CRM por sus siglas en inglés (Customer Relationship Management), la cual fue planteada hace ya varios años, pero que ha tenido un resurgimiento gracias a su combinación con la tecnología, ya que ésta ha facilitado a las empresas el manejo de grandes volúmenes de información en relación a los clientes.

Con el avance tecnológico se han desarrollado nuevas técnicas y herramientas para facilitar las tareas manuales de los usuarios, al punto de crear sistemas que tratan de imitar el razonamiento humano. Estos sistemas utilizan las denominadas técnicas de inteligencia artificial (IA), entre las que cabe mencionar, los algoritmos genéticos, las redes neuronales, y la lógica difusa.

Para poder dar un apoyo en la toma de decisiones, específicamente en la metodología CRM, en la parte de clasificación de clientes y promoción de productos, se propone el diseño de un sistema que pueda realizar estas clasificaciones, usando una técnica de IA: la lógica difusa.

1. RELACIÓN ENTRE CRM Y LÓGICA DIFUSA

1.1. Técnicas de inteligencia artificial

La inteligencia artificial surgió por la inquietud de hacer que las máquinas puedan tener un razonamiento inteligente, pero la pregunta es ¿Qué significa razonamiento inteligente?, ya que para todos existe un concepto diferente de inteligencia, en ese contexto, podemos definir que la inteligencia artificial es la forma de programar una máquina de manera que realice operaciones definidas por el usuario, con una capacidad de tomar decisiones, y en algunos casos de generar información a partir de los hechos que reciba como entrada, es decir, que sea capaz de retroalimentarse.

Desde el año 1956 se han venido proponiendo modelos y técnicas para poder llevar a cabo esta idea, hoy en día existen varias técnicas, las cuales han sido aplicadas ya en varios sistemas, no limitándose estos a la ingeniería en ciencias de la computación, sino además en áreas de procesos industriales, y en el área médica, entre otros.

A continuación se describe una de estas técnicas, la lógica difusa, y luego se mencionan brevemente algunas otras técnicas de las más utilizadas.

1.1.1. Lógica difusa

Varios de los procesos lógicos utilizados en la lógica Booleana consisten en clasificar los elementos de determinado conjunto en falso o verdadero, o si pertenece o no pertenece a cierta categoría o clasificación, con ello existía un criterio cerrado de clasificación, ya que para determinar si la temperatura de un objeto es alta, alguien podría definir un criterio basado en un valor específico, y las comparaciones se realizarían en torno a ese valor, con dos únicas clasificaciones “tiene” o “no tiene” temperatura alta.

Por ejemplo se puede decir que “un objeto tiene una alta temperatura si tiene 65° C o más”, entonces clasificándolo por medio de la lógica clásica un objeto con 67° tiene temperatura alta, mientras que un objeto con 64° no la tiene, a pesar de estar a tan solo un grado de diferencia respecto del límite propuesto para clasificar la temperatura alta, con ello se está perdiendo un dato que podría ser crítico, por ejemplo si se tratara de componentes industriales que tienen un límite de resistencia al calor, y que deben ser atendidos, para evitar su deterioro.

Tratando de evitar ésta pérdida de datos, alrededor de la década de 1960, el Dr. Lofti Zadeh de la Universidad de California en Berkeley, propuso un modelo de incertidumbre del lenguaje natural, en el que proponía la teoría de conjuntos borrosos. Los conjuntos borrosos son la base para los fundamentos de la lógica difusa, la cual se puede decir que es una generalización de la lógica Booleana, ya que admite una serie de valores que van degradados desde los dos extremos de la lógica clásica, o sea desde pertenecer totalmente a la categoría hasta no pertenecer a la misma. La lógica difusa tiene varios elementos para su manejo, los cuales se explicarán a continuación.

1.1.1.1. Dimensión difusa

Es llamada también variable lingüística, se refiere al campo o área sobre el que se desea trabajar. Cada dimensión debe tener varias posibles clasificaciones o categorías en las que se pueda dividir. Como ejemplo de dimensiones difusas podemos mencionar: la temperatura de un objeto, la estatura de una persona, la velocidad al conducir un automóvil, entre muchas otras.

1.1.1.2. Atributos difusos

Son los distintos valores de clasificación o categorías en las que se divide cada dimensión difusa, son llamados también términos o valores lingüísticos. Por lo general una dimensión difusa debe tener por lo menos tres atributos, y en la mayoría de los casos son un número impar, para poder tener una categoría central, y varias degradaciones, en igual número hacia ambos lados.

Como ejemplo de atributos difusos y para complementar la explicación de las dimensiones difusas, tenemos que los posibles atributos para la dimensión temperatura de un objeto podrían ser: muy fría, fría, media, caliente, y muy caliente; en el caso de la estatura de una persona podrían proponerse como atributos a muy alto, alto, bajo, y muy bajo; mientras que para la velocidad al manejar un automóvil, se pueden considerar a rápido, regular y lento como sus atributos difusos. Puede notarse que no en todos los casos existen una cantidad impar de elementos, ya que esta no es una regla, y que los atributos quedan a criterio de cada persona.

1.1.1.3. Pertenencia

La pertenencia indica el grado o porcentaje en que un valor o característica, del sujeto que se esté evaluando, pertenece o corresponde a un atributo difuso, por cada atributo difuso se tiene un grado de pertenencia.

Es importante entender que un determinado valor puede pertenecer en cierto grado a más de una clasificación o categoría. Por ejemplo, para la dimensión difusa estatura, si proponemos los atributos alto, medio y bajo, y evaluamos la estatura de determinada persona, por ejemplo 1.78 metros, entonces podríamos decir que esta persona pertenece en cierto porcentaje al grupo medio, y también se puede decir que en otro porcentaje pertenece al grupo alto, estos porcentajes son la pertenencia que tiene el valor 1.78 a los atributos medio y bajo. Cuando un valor no pertenece a un atributo difuso se dice que tiene un grado de pertenencia igual a cero, como en el caso del valor 1.78 respecto del atributo difuso bajo.

1.1.1.4. Funciones de pertenencia

Son funciones matemáticas que expresan, para cada atributo difuso de una dimensión difusa, el grado de pertenencia dado un valor de entrada, colocando en el dominio los valores posibles a evaluar, y en el contra dominio el grado de correspondencia para ese valor.

Existen varios tipos de funciones de pertenencia, entre los mas conocidos se tienen las funciones L, Γ , Λ y Π . Las dos primeras son utilizadas para determinar la pertenencia a atributos difusos de los extremos de la dimensión difusa; el tercer y cuarto tipo de función son similares entre sí, con la diferencia que el último sirve para determinar la pertenencia a atributos que están definidos en un intervalo, y no de forma puntual.

Para determinar una función tipo L, se le coloca un grado de pertenencia de uno a los valores menores o igual al mas significativo del atributo que se encuentra más a la izquierda, y se le da un valor de cero a los valores mayores o iguales al más significativo del atributo que está inmediatamente a la derecha del primer atributo mencionado.

Para determinar una función tipo Γ , se le coloca un grado de pertenencia de uno a los valores mayores o igual al más significativo del atributo que se encuentra más a la derecha, y se le da un valor de cero a los valores menores o iguales al más significativo del atributo que esta inmediatamente a la izquierda del primer atributo mencionado.

Para determinar una función tipo Λ , se hará con los atributos que no son extremos, y que su valor significativo no esté definido en un intervalo. Se le coloca un grado de pertenencia de uno al valor más significativo del atributo, y un grado de pertenencia de cero a los valores más significativos de los atributos que se encuentran inmediatamente a la derecha y a la izquierda del atributo. Como se menciono antes, la función Π se determina igual que la función Λ , pero esta será utilizada cuando el valor más significativo esté definido en un intervalo.

1.1.1.5. Operadores difusos

Los operadores difusos sirven para evaluar las relaciones que los atributos difusos de determinada dimensión difusa tienen unos con otros. Un operador difuso evalúa dos o más grados de pertenencia, y da como respuesta otro grado de pertenencia. Entre los operadores de uso más frecuente tenemos a los operadores mínimo, máximo, complemento, producto, e implicación.

El operador mínimo se aplica cuando existe una relación lógica de intersección, dada por el operador lógico y. Este operador evalúa los grados de pertenencia y da como respuesta el valor del más pequeño de los datos evaluados.

El operador máximo es aplicado cuando existe una relación lógica de unión, dada por el operador lógico o inclusivo. Este operador evalúa todas sus entradas, y da como resultado el valor más grande de los datos evaluados.

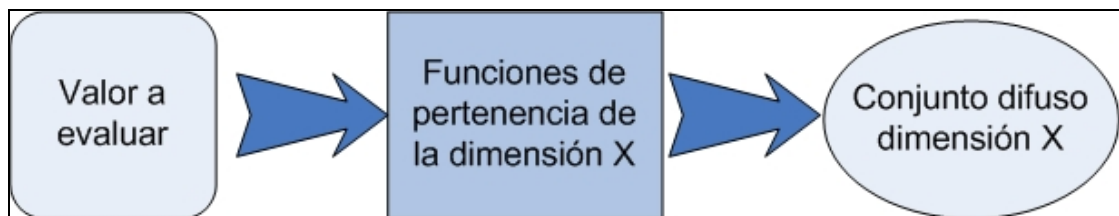
El operador producto funciona igual que una multiplicación entre dos números, que en este caso es la multiplicación de dos grados de pertenencia. La implicación sirve para denotar la consecuencia de que se cumplan las operaciones indicadas a la izquierda del operador.

1.1.1.6. Conjuntos difusos

Un conjunto difuso está formado por parejas de elementos, cada pareja está formada por un atributo difuso, y el grado de pertenencia de determinado valor a dicho atributo. Se tiene un conjunto difuso por cada valor que se esté evaluando, y se tienen tantas parejas en el conjunto, como atributos difusos tenga la dimensión difusa en cuestión.

Por ejemplo, usando el mismo ejemplo de la sección 1.1.1.3, con el valor 1.78 metros, podemos definir el conjunto difuso (alto, x), (medio, $1-x$), (bajo, 0). Debe notarse que la suma de todos los grados de pertenencia debe sumar uno. En la figura 1 se muestra el diagrama conceptual de los conjuntos difusos.

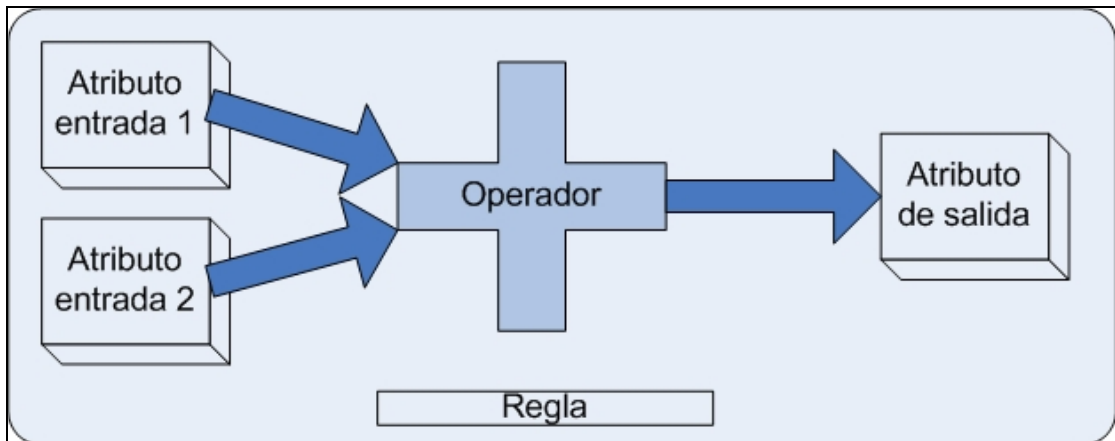
Figura 1. Diagrama conceptual de conjuntos difusos



1.1.1.7. Reglas difusas

Cuando se relacionan dos o más atributos por medio de los operadores difusos, se forma una expresión, la cual, si es seguida del operador de implicación se convierte en una condición, y la parte que está a la derecha del operador de implicación representa el resultado que se obtendrá si la condición se cumple. Una regla difusa sirve para expresar un atributo en función de las operaciones entre dos o más atributos que forman una condición. Los atributos usados para formar la expresión son llamados atributos de entrada, y el atributo resultante es llamado atributo de salida. En la figura 2 se muestra el diagrama conceptual de las reglas difusas.

Figura 2. Diagrama conceptual de las reglas difusas



1.1.1.8. Grado de soporte

El grado de soporte expresa en forma de porcentaje el peso o prioridad que tiene una regla respecto de las demás. El valor por defecto del grado de soporte es uno, el cual representa el máximo peso que una regla puede tener.

1.1.1.9. Determinación de conjuntos difusos

Para determinar los conjuntos difusos, a los valores a evaluar se les asigna un grado de pertenencia por cada atributo difuso, haciendo uso de las funciones de pertenencia, las cuales para cada valor del dominio que se les ingrese devolverán un valor entre cero y uno que representará el grado de pertenencia.

Para determinar los conjuntos difusos de los atributos de salida, se deben evaluar las reglas difusas, al proceso de evaluación de la parte condicional de las reglas se le conoce como agregación.

Luego de obtener el resultado de la agregación se debe aplicar el operador producto entre el resultado de la regla y su respectivo grado de soporte, para darle su verdadero valor a cada una, a este proceso se le conoce como composición.

1.1.1.10. Compensación

Cuando los atributos de la parte condicional de la regla difusa pueden nivelarse entre si, cuando la ausencia de uno puede ser sustituida por el exceso del otro, se aplica el proceso difuso conocido como compensación. Este se utiliza para evaluar las reglas difusas utilizando una constante arbitraria de compensación, que indica en que porcentaje uno de los atributos difusos puede nivelar o compensar al otro, y da como resultado el valor de la regla.

No en todos los casos se puede utilizar la compensación, por ejemplo si se tratara de los atributos intensidad del sol, y cantidad de bronceador, no se puede nivelar la ausencia del sol con una gran cantidad de bronceador, ni se podría compensar el no tener bloqueador a cambio de un sol intenso. Por ello debe evaluarse las consecuencias de aplicar el proceso de compensación antes de llevarlo a cabo. La función matemática que se utiliza para el proceso de compensación es:

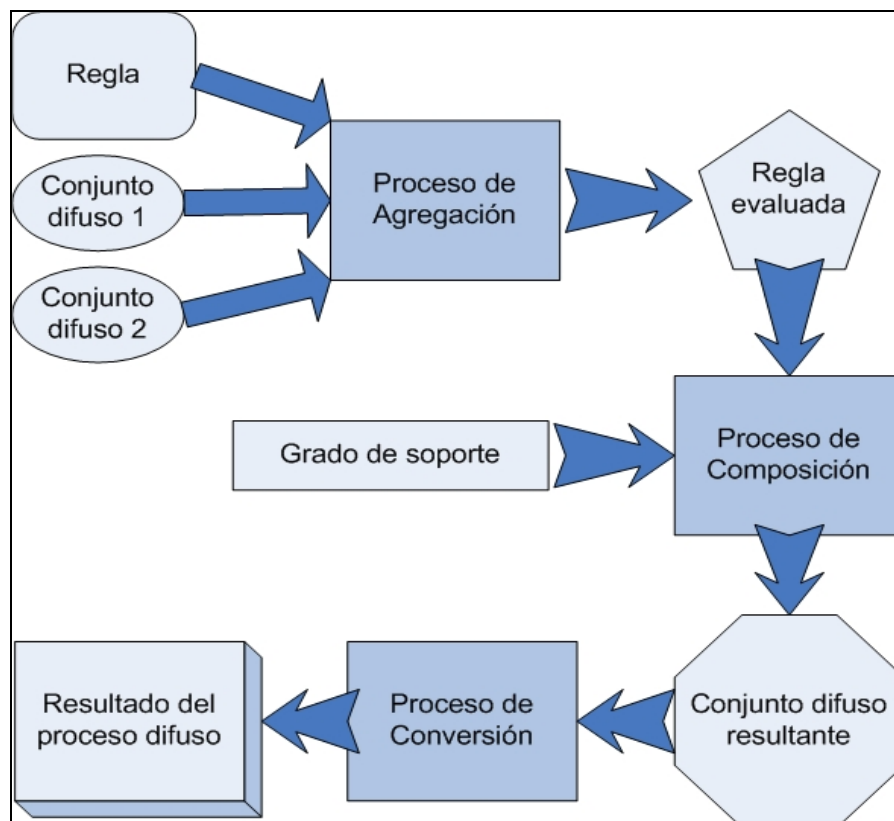
$$\mu = \left(\prod_i \mu_i \right)^{(1-\gamma)} \left(\prod_i (1 - \mu_i) \right)^\gamma$$

donde γ es el grado de compensación entre los atributos difusos, y μ es el grado de pertenencia de los valores que se están evaluando.

1.1.1.11. Conversión de valores difusos

Los valores difusos obtenidos de la evaluación de las reglas deben ser transformados a valores numéricos para su interpretación y presentación, ya que si se deja el proceso difuso hasta este punto lo único que se tiene es un conjunto difuso que no es de utilidad. Existen varios procesos para convertir un conjunto difuso a números, de los cuales el que se mencionará es el método del centro de gravedad, el cual consiste en realizar una sumatoria del producto de las parejas del conjunto difuso resultante, y luego dividirlo entre la sumatoria de los grados de pertenencia del conjunto difuso. En la figura 3 se muestra un diagrama con el flujo completo del proceso difuso.

Figura 3. Diagrama conceptual del proceso difuso



1.1.2. Otras técnicas de Inteligencia Artificial

Existen otras técnicas de inteligencia artificial, cada una con sus ventajas y características propias, que, según sea la aplicación que se realizará, así será el tipo de técnica a utilizar.

1.1.2.1. Redes neuronales

Es un sistema compuesto por un gran número de elementos básicos, agrupados en capas y que se encuentran altamente interconectados; Esta estructura posee varias entradas y salidas, las cuales serán entrenadas para reaccionar con ciertos valores de salida, ante los estímulos de entrada.

Estos sistemas tratan de emular al cerebro humano. Requieren aprender a comportarse y alguien debe encargarse de enseñarles o entrenarles, en base a un conocimiento previo del entorno del problema, luego conforme el sistema esté operando podrá aprender automáticamente, generando sus propios conocimientos en base a los existentes y las entradas que esté recibiendo.

Entre las aplicaciones de las redes neuronales se pueden mencionar las variaciones en la bolsa de valores, los riesgos en préstamos, el clima local, el reconocimiento de patrones y la minería de datos.

1.1.2.2. Algoritmos genéticos

Esta técnica se basa en la simulación de los de los procesos conocidos de la evolución natural, usando estos conceptos para describir el comportamiento y relaciones en el algoritmo. Fue propuesta por John Holland en la década de 1970. Los algoritmos genéticos son flexibles y robustos en problemas combinatorios complejos. No necesitan tener un conocimiento específico previo,

a diferencia de las redes neuronales, solo requieren conocer las reglas de como evaluar una solución propuesta.

A continuación se mencionan los conceptos de la evolución natural sobre los que se desarrolla ésta técnica y su interpretación en la técnica del algoritmo.

Población inicial, se debe definir un conjunto inicial de soluciones aleatorias al problema.

Selección natural, se debe evaluar cada una de las soluciones, y las mejores son seleccionadas.

Reproducción, las soluciones seleccionadas se combinan entre sí para generar una nueva población.

Mutación, algunos cambios aleatorios entre las soluciones son simulados.

Generaciones, después de las mutaciones se tiene una nueva población, a ésta también se le aplican los procesos de selección natural, reproducción y mutación dando lugar a una nueva generación, hasta que una de ellas dé la solución al problema.

1.2. Metodología CRM

CRM son las siglas en ingles de *Customer Relationship Management*, que podrían traducirse como administración de las relaciones con el cliente. La finalidad de esta metodología es, en base a la información que se tiene sobre el cliente, orientar el negocio hacia él y no enfocar al cliente hacia el negocio.

1.2.1. Descripción y enfoque

Al hablar de CRM lo primero que hay que tener claro es que no es un paquete de software, ni que haya surgido con el avance de la tecnología, sino que es una metodología orientada a la administración del negocio, o dicho de otra manera CRM es un concepto, el cual ha sido aplicado desde hace ya varios años, muchas veces de una forma empírica.

Para explicar de forma resumida cuál es el corazón del CRM, podemos decir que es conocer al cliente y sus preferencias, y en base a ello ofrecerle lo que le gusta, y orientar las características de los productos a cumplir con esas preferencias. Si la idea fundamental es conocer al cliente no es difícil pensar que cualquier persona que tenga un negocio, aunque no conozca esta metodología, puede utilizar este concepto para mantener contentos a sus clientes y lograr con ello su fidelidad y el fortalecimiento de sus relaciones, ya que es más fácil conservar un cliente que adquirir uno nuevo.

Con el avance tecnológico se ha logrado tener un mejor manejo de la información que las empresas tienen acerca del gran número de clientes que hay en su cartera, cosa que no sería posible llevar solo de forma mental o por medios manuales, como si fuera un negocio con cien clientes, por ejemplo.

CRM es la unión entre la tecnología y la teoría del marketing relacional, el cual se puede definir como una estrategia de negocios enfocada en conocer, satisfacer y anticipar los deseos de los clientes. Toda empresa que desee incorporar la metodología CRM a ella, debe cambiar el enfoque del marketing tradicional por el del marketing relacional.

El marketing relacional cambia la idea de centrar la atención sobre el

producto, y la enfoca hacia el cliente, ya que se necesita tener información sobre sus gustos para poder desarrollar los productos de acuerdo a lo que el esperaría; para lograr esto debe existir un proceso de comunicación de dos vías en el que es el cliente el que decide cuándo empieza y cuando termina, además ésta comunicación no debe ser de tipo colectivo, sino mas bien una promoción personalizada para cada cliente con información de los productos que poseen las características que le interesan.

El objetivo general de la metodología CRM es conservar a los clientes existentes, enfocando todos los esfuerzos hacia él y lo convierte en el centro de la empresa, algunos de los objetivos específicos de la metodología son:

1. Llevar a cabo un incremento de las ventas con lo clientes actuales.
2. Obtener la mayor cantidad de información sobre el cliente.
3. Mejorar el servicio al cliente.
4. Personalizar y optimizar la atención al cliente.
5. Mejorar las ofertas. y promociones
6. Identificar clientes con potencial de crecimiento dentro de la empresa.
7. Aumentar las tasas de retención de clientes.

La puesta en marcha de la metodología, ya sea por medios manuales o con algún paquete de software, no garantiza el éxito de la misma, ya que ésta es solo parte de los elementos clave en el éxito de una empresa. Algunos factores claves al poner en marcha la metodología son:

1. Tener comunicación con los proveedores, para que estos puedan colaborar con cumplir con las características requeridas por el cliente desde el origen del producto.
2. Determinación de la estrategia publicitaria, desde el contenido hasta

la forma en que se hará toda la publicidad.

3. Rápida administración y atención de las peticiones de los clientes, por medio de centros de atención y un equipo humano dispuesto para tal efecto.

1.2.2. Elementos de CRM

Muchos autores han propuesto varios componentes o elementos para esta metodología según el punto de vista de cada uno. A continuación se muestran algunos de ellos, y una breve descripción de los mismos.

1.2.2.1. Centro de llamadas

Es el encargado de recibir todas las peticiones de los clientes, y de ser posible colaborar con ellos para resolver sus problemas. Este elemento es quien da la cara al cliente de parte de la empresa, por lo que se debe procurar que el cliente se lleve una buena impresión del mismo.

1.2.2.2. Centro de contactos

Es el responsable de hacer llegar la información hasta los clientes, y de decidir por qué medio hacerlo, por ejemplo vía correo electrónico, por el correo tradicional, o por medio de volantes, entre otros.

1.2.2.3. Almacén de datos

Es la parte donde se almacena toda la información de clientes, productos, promociones, competencia y las características de los mismos.

1.2.2.4. Fuerza de Ventas

Es el equipo encargado de impulsar los productos, tanto tradicionales como los nuevos, es responsable de la distribución, a la vez que es la cara de la empresa ante los clientes.

1.2.2.5. Gestión del conocimiento

Este equipo es el encargado de llevar a cabo el análisis y presentación de los datos de los clientes luego de haberlos analizado.

1.2.2.6. Gestión de la información para ejecutivos

Este elemento prepara los informes y reportes sobre el estado de la empresa con una orientación hacia el alto mando de la empresa, para que este pueda interpretarlos y tomar decisiones apoyados en ellos.

1.3. Clasificación de los clientes

El ser humano en todo momento clasifica las cosas que tiene a su alrededor, con la finalidad de organizarlas o entenderlas mejor, y darles determinado trato según sus características. Por ejemplo, un automóvil puede ser feo, regular o bonito; la comida de cierto restaurante puede ser buena o mala; un ambiente puede ser húmedo, frío o caluroso.

Cuando se trata de una empresa que posee una amplia variedad de clientes, lo ideal sería tratarlos a cada uno de forma individual y personalizada, actividad que implicaría un incremento de costos y tiempo, ambos recursos también podrían ser utilizados para atraer nuevos clientes y aumentar las

ventas, en ambos casos resulta más económico trabajar con grupos de clientes que con cada uno de ellos. Es por ello que la cartera de clientes también debe ser objeto de clasificación. El tipo de clasificación que se realice dependerá del criterio de cada uno; en las siguientes secciones se mencionan algunos de ellos.

1.3.1. Madurez de los clientes

Similar a la madurez del ser humano que describe cuánto tiempo tiene de vida y cuán estable es, la madurez de los clientes mide qué tanto tiempo tienen de estar en la cartera y qué tan estables son en la misma.

El objetivo de esta clasificación es determinar qué tan estable es la cartera en base a la cantidad de clientes que posee cada nivel de madurez, ya que cada nivel tiene cierta tendencia que describe el comportamiento de los clientes que hay en ella.

1.3.1.1. Nivel inicial

En este nivel se encuentran los clientes que no tienen más de dos años de haber iniciado las actividades, generalmente son inestables, es decir, pueden en cualquier momento decidir romper la relación con la empresa, ya que no confían aún en ella. En esta etapa se requiere cuidar mucho al cliente para que no se vaya.

1.3.1.2. Nivel adolescente

Los clientes que se encuentran en este nivel tienen más de dos años de haber hecho su primer contacto con la empresa, se puede decir que en esta

etapa los clientes son más impetuosos, cosa que podría ser aprovechada para hacerle muchas propuestas de productos, ya que él seguramente aceptará dado que aún no tiene una posición definida en cuanto a los productos que desea. Puede decirse que en esta fase es cuando el cliente se define en el mercado.

1.3.1.3. Nivel adulto

En esta fase se encuentra la mayor cantidad de clientes de la cartera, acá se encuentran todos los clientes que ya se estabilizaron, y que a pesar de que se les hagan propuestas de productos nuevos, difícilmente aceptarán ya que tienen definidos sus gustos.

Dada su estabilidad, en esta etapa los clientes no requieren tanta atención, no se les debe abandonar, pero permiten darles más atención a los clientes de las etapas anteriores.

1.3.1.4. Nivel de vejez

Lamentablemente cuando un cliente llega a esta parte, es por que está abandonando ya la cartera, ante ello ya no se puede hacer mucho, ya que podría ser por muchas razones, y como ya tiene sus gustos definidos, no se les puede ofrecer algo nuevo para tratar de conservarlos.

1.3.2. Segmentación

El concepto de segmentación consiste en agrupar a los clientes de acuerdo a sus preferencias o de acuerdo a sus características, con la finalidad de obtener una clasificación de toda la cartera de clientes. Por muy variadas

que fuesen estas preferencias, toda la cartera podrá pertenecer a un segmento de los definidos por la empresa.

El objetivo de segmentar es poder manejar a toda la cartera de una mejor manera tratándolos a todos como grupos con características similares entre los miembros, pero a la vez diferentes entre los grupos, para orientar, ya sea una nueva campaña publicitaria, o una promoción o un nuevo producto, a un grupo o segmento determinado de acuerdo a esas características.

Algunas de las clasificaciones que se dan al segmentar se definen de acuerdo a las características del cliente, por ejemplo:

1. Por área geográfica: zonas de la capital, zona urbana y rural.
2. Por demografía: edad, género, estado civil, número de hijos
3. Por nivel socioeconómico: ocupación, ingresos, gastos

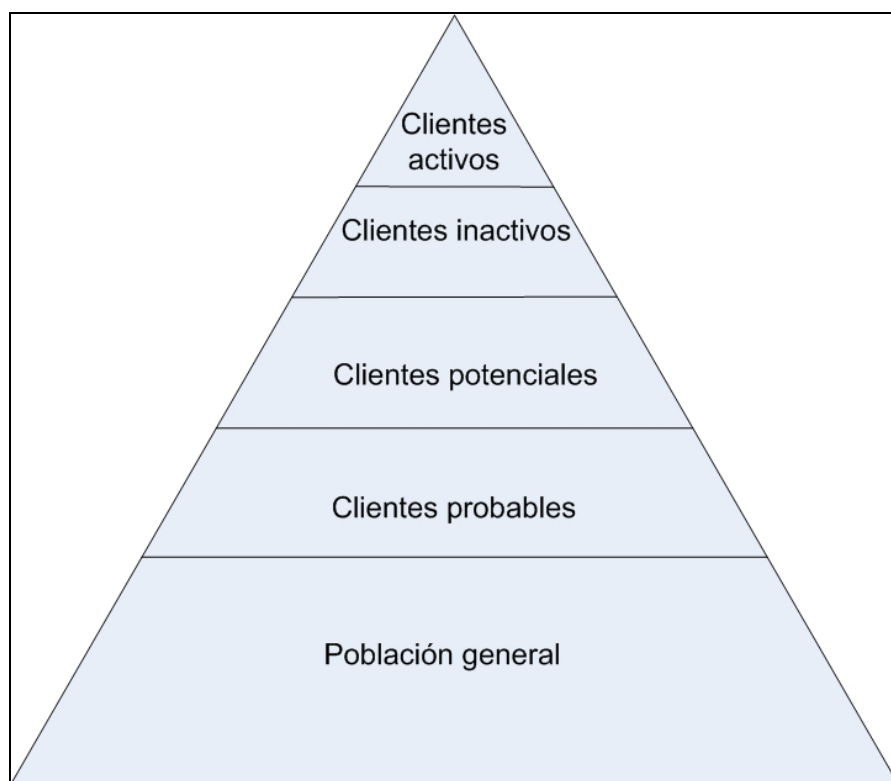
1.3.3. Pirámides de clientes

Las pirámides de clientes son clasificaciones de clientes en las que se busca darles una posición simbólica dentro de la cartera. La pirámide debe dividirse en niveles horizontales desde su base, asignándole una categoría a cada nivel.

El concepto de pirámide se aplica al expresar que, en la medida en que se asciende en ella, así va disminuyendo la cantidad de personas que corresponden a cada nivel de los que se han definido. Los grupos o clasificaciones de una pirámide de clientes pueden variar de acuerdo al criterio de cada persona, además de que cada uno puede o no subdividirse en más niveles.

En la figura 4 se muestra un ejemplo de la distribución conceptual de una pirámide de clientes. La división de estos niveles considera a todos, sean o no clientes, algunos de los niveles más comunes se detallan a continuación.

Figura 4. Ejemplo de pirámide de clientes



1.3.3.1. Población general

A este grupo corresponden todas las personas que nunca han requerido de la empresa, y que es muy probable que nunca lo hagan, ni siquiera se pueden considerar como candidatos a subir por la pirámide, sin embargo es bueno tomarlos en cuenta para ver que tan grande es la cantidad de personas que no está en la cartera de la empresa.

1.3.3.2. Clientes probables

Este grupo tampoco ha tenido algún tipo de contacto con la empresa, la diferencia radica en que, por las características que tiene, podría llegar en algún momento a requerir de la empresa, por lo que se le debe considerar como una posibilidad.

1.3.3.3. Clientes potenciales

Se diferencia de los anteriores en que, a pesar de no haber comprado en la empresa, ya existe un contacto con ellos, lo cual da la posibilidad de que, según la capacidad del contacto, podría convencer al grupo de dejar de ser un cliente potencial y subir por la pirámide.

1.3.3.4. Clientes inactivos

En este grupo se encuentran los clientes que en algún momento han realizado transacciones con la empresa, pero que en un período de tiempo, por ejemplo diez meses, no han vuelto a tener contacto; es importante enfocar algunas estrategias de promociones hacia este grupo de manera que se puedan recuperar las relaciones con él.

1.3.3.5. Clientes activos

Son la cumbre de la pirámide y el grupo más pequeño de la misma, acá se encuentran los clientes reales, es decir, los que han realizado transacciones con la empresa y que actualmente mantienen contacto frecuente. Dentro de este nivel se pueden realizar subdivisiones para clasificar a todo el grupo activo, algunas de las divisiones posibles pueden ser:

1. Por nivel de compra que realicen
2. Por la cantidad de contactos o visitas que se realiza con ellos
3. Por la cantidad de productos diferentes que compran
4. Por la cantidad de compras consecutivas que realizan

1.4. ¿Por qué técnicas de inteligencia artificial?

Dado el avance tecnológico, y la facilidad que éste brinda para el procesamiento de datos, para el ser humano sería ideal que las computadoras pudieran hacer el mismo tipo de análisis y razonamiento que él, pero si se utilizan las estructuras convencionales de la computación, por ejemplo listas, arreglos, colas, y otras, no es posible representar el pensamiento, ni el procesamiento que tiene la mente del ser humano, es por ello que se utilizan representaciones de lo que se le ha denominado inteligencia artificial, en donde sí es posible darle información a la computadora y las estructuras para que esta lo procese, simulando el razonamiento humano.

1.4.1. Métodos clásicos de apoyo en la toma de decisiones

Existen varios métodos que se han utilizado para poder apoyarse en la toma de decisiones, algunos de ellos haciendo uso de la tecnología para hacer determinados cálculos y poder procesar grandes cantidades de datos. Entre los más conocidos tenemos.

1. Experiencia y apoyo de terceros: el inconveniente que ofrece es que se necesita que ya haya ocurrido por lo menos una vez el hecho, para tener referencias acerca del mismo.

2. La intuición: no es un método formal, no es confiable, no es recomendable, sin embargo cuando no se cuenta con ningún otro es el único que se puede utilizar.
3. Métodos formales: son todos aquellos que siguen una serie de pasos definidos para poder encontrar un resultado en base a los datos que se tengan, algunos de los más utilizados son los métodos estadísticos.

1.4.2. Técnicas de inteligencia artificial

Se han aplicado en la toma de decisiones en las áreas gerenciales y financieras, a manera de ejemplo se puede citar "...Kingdon y Feldman han usado algoritmos genéticos para predecir la bancarrota de una corporación..... Dutta y Shekhar describen el uso de una red neuronal para predecir la clasificación de los bonos de una empresa..... El Fuji Bank, en Tokio, usa un sistema que maneja lógica difusa para efectuar transacciones con bonos a corto plazo..."¹

¹

Castillo Tapia, Guadalupe y A. Coello Coello. Uso de **Técnicas de Inteligencia Artificial para Aplicaciones Financieras**. <http://www.lania.mx/biblioteca/newsletters/1998-otono-invierno/tecnicas.html>. (2-Ago-2004)

2. VARIABLES A UTILIZAR EN EL MODELO

2.1. Variables de entrada

Las variables de entrada son todas aquellas que contienen información que el sistema utilizará para realizar todas las operaciones y en base a ellas determinar las soluciones a los problemas que sean planteados. De los valores de las variables de entrada es de donde se obtendrá la información que reflejará el resultado del proceso realizado por el sistema.

Todos los valores a utilizar en el sistema dependerán de lo que requieran las reglas. Existen muchos métodos de obtener los valores de las variables, entre ellos podemos mencionar:

1. Los ingresados por el usuario directamente, ofrece la desventaja de ser un medio lento, y en algunas ocasiones aburrido, su ventaja es que no requiere ningún proceso previo para ingresar la información y el usuario puede ver el resultado del ingreso inmediatamente.
2. Los que son cargados de un archivo de texto por medio de un proceso por lotes, la ventaja es que se pueden cargar varios registros a la vez, y el usuario puede dedicarse a hacer otras cosas mientras el proceso se realiza, la desventaja es que requiere de un proceso previo que genere el archivo con la información, y si este contiene errores, puede que el proceso de carga no se realice correctamente.

3. Los que son cargados de una base de datos a otra, este proceso es un tanto más delicado que los anteriores, ya que se está manipulando la información de la base de datos directamente, sin embargo ofrece la ventaja de realizarse rápidamente, ya que no debe esperar un proceso de generación de archivos.

Las variables de entrada deben ser precisas según la información que se necesita obtener, es decir, no se deben colocar variables que no serán de utilidad, ya que estas solo ocuparán un espacio de memoria al ser cargadas, o un espacio de disco si son guardadas temporalmente.

2.1.1. Variables de clientes

Estas variables contienen información sobre las características del cliente, tanto la información propia de la persona, como la información del cliente, es decir, la información referente a las transacciones que realiza con la empresa.

Entre las variables que contienen información referente a las características propias de la persona se pueden mencionar:

1. Edad: se calcula a partir de la fecha de nacimiento de la persona restándole de la fecha actual
2. Género: masculino o femenino
3. Ingresos: Indica cuánto es el ingreso promedio estimado de la persona en un período de tiempo específico
4. Número de hijos: pueden ser cero o más
5. Lugar de residencia: indica si la persona vive en un área rural o en un área urbana, puede también clasificarse acorde a zonas, o si está en la capital, municipios o departamentos.

6. Empleado: es un valor lógico que indica si la persona labora actualmente.
7. Profesión: indica a qué se dedica la persona.

En cuanto a las variables con información sobre la persona como cliente, pueden mencionarse:

1. Antigüedad: se refiere a la cantidad de tiempo que una persona tiene de estar en la cartera de clientes, se calcula a partir de la fecha de su primera transacción restándole de la fecha actual.
2. Frecuencia de visitas: mide cada cuanto tiempo un cliente es visitado por el contacto de la empresa, se obtiene de la diferencia promedio entre las visitas que se le ha realizado. Las visitas se refieren tanto a las hechas de forma personal, como a las que se hagan por correo, o por correo electrónico.
3. Madurez: indica en qué nivel o etapa de madurez se encuentra el cliente de acuerdo a los mencionados en el capítulo 1.3.1
4. Nivel en la pirámide de clientes: muestra qué clasificación tiene el cliente según la presentada en la sección 1.3.3.

A manera de ejemplo se muestra en las tablas I y II los datos de la tabla CLIENTE, la cual contiene la información de diez clientes ficticios, los cuales se utilizarán en el resto de los capítulos. En la tabla CLIENTE se pueden observar datos como:

1. Código del cliente
2. Nombre
3. Apellido
4. Dirección

5. Teléfono
6. Correo electrónico
7. Fecha de nacimiento
8. Estado civil
9. Número de hijos
10. Género
11. Sueldo

Tabla I. Datos de la tabla CLIENTE – parte 1

Cliente	Nombre	Apellido	Dirección	Teléfono
1	Raúl Antonio	Mijangos Castro	10 avenida 11-44 zona 18 colonia Atlántida	22555708
2	Luis	Pineda	zona 3	32563256
3	Juan	Perez	zona 6	28822828
4	José	Arreola	zona 1	21547889
5	Horacio	Gomez	zona 9	25568965
6	Elvira	Alvarez	Zona 17	54566655
7	Carlos	Diaz	Zona 1	21546598
8	Victor	Castillo	Zona 4	23156632
9	Clara	Acevedo	Zona 10	21568974
10	Mynor	Acajabón	Zona 1, Quetzaltenango	74569856

Tabla II. Datos de la tabla CLIENTE – parte 2

Email	Fecha_nacimiento	Estado_civil	numero_hijos	genero	sueldo
ramijangos@yahoo.es	10-09-80	0	1	0	3500
lpineda@lp.com	26-01-81	0	0	0	3500
jperez@jp.com	01-01-80	1	0	0	8000
jarreola@ja.com	02-05-78	0	0	0	12000
hgomez@gen.com.gt	06-06-72	1	2	0	30000
ealvarez@gen.com.gt	07-05-81	1	2	1	7500
cdiaz@dot.com	23-12-68	3	3	0	6100
vcastillo@hotmail.com	04-12-62	2	1	0	6500
clara.gomez@gmail.com	02-05-74	1	2	1	14000
mynor@chapinmail.com.gt	01-02-79	0	0	0	1200

2.1.2. Variables de productos y promociones

Las variables de productos se refieren a las características que posee un producto físicamente, y que permiten describirlo. Dado que éstas características son tan variadas, según sea el negocio del que se trate, así también será la definición de las mismas, las que son comunes a todos los productos independientemente del negocio son el precio, el tamaño y la cantidad.

Las variables de promociones describen las características que tiene una promoción, por ejemplo la duración, dada por la fecha de inicio y la fecha de finalización, el segmento de mercado al que está dirigido, algunas de las variables de clientes para indicar a que tipo de personas está dirigida.

Se debe aclarar que todas las variables antes mencionadas dependen de los datos que contenga el sistema general utilizado actualmente para el manejo de los clientes.

2.2. Variables de salida

Las variables de salida son todas aquellas por medio de las cuales el sistema mostrará los resultados obtenidos del procesamiento de los datos ingresados por medio de las variables de entrada, las variables de salida son totalmente dependientes del procesamiento que se haga con los datos de entrada. Estas variables son las que se utilizarán para interpretar los resultados y servirán de apoyo en la toma de decisiones.

2.2.1. Variables de clientes

Estas variables servirán como apoyo para tomar decisiones acerca de los clientes de la cartera, por ejemplo, indicando si un cliente es apto para recibir o no una campaña, para determinar si un cliente puede recibir un descuento, o si puede pasar a alguna categoría de clientes que esté definida en las empresa, por ejemplo, pasa a cliente dorado o permanece como cliente ordinario.

2.2.2. Variables de promociones

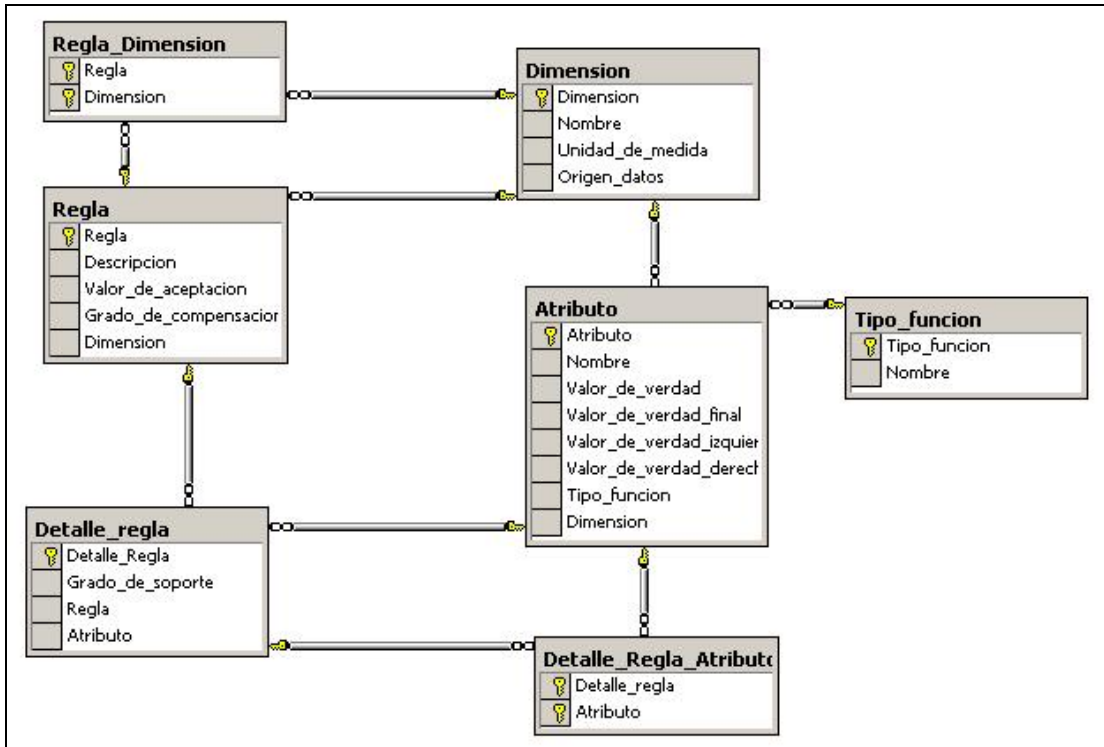
Las variables de promociones son las que se usarán como apoyo en la toma de decisiones sobre las promociones, al igual que con las variables de clientes, los resultados que éstas reciban dependerán de los datos de entrada que sean ingresados y luego procesados por el sistema, algunas de las variables que pueden ser usadas como resultado son el costo de ejecución de la promoción y la prioridad.

2.3. Almacenamiento de las variables de entrada

Para el manejo de los componentes de lógica difusa a utilizar en el sistema, estos deberán guardarse en la base de datos, con el fin de que los datos sean de fácil acceso, mantengan una relación entre sí y puedan mantener la integridad de las entidades como la integridad referencial.

El modelo de base de datos a utilizar por el sistema, mostrado en la figura 5, será exclusivamente para el manejo de los componentes de lógica difusa, es decir, no incluye las entidades utilizadas para el procesamiento de los datos.

Figura 5. Modelo de base de datos para el sistema



Toda la información que se almacene en estas entidades tiene un tiempo de vida relativamente largo, ya que luego de realizar el proceso de análisis de los datos, estos permanecerán hasta que el usuario decida borrarla o actualizarla.

Las entidades mostradas en el modelo entidad relación de la figura 1 son las siguientes:

1. DIMENSION: almacena un código identificador, un nombre descriptivo, la unidad de medida y un espacio para almacenar el código SQL para la obtención de los valores desde la base de datos actual.

2. TIPO_FUNCION: almacenará un código identificador y el nombre de la función de pertenencia de las que se usarán en el sistema.
3. ATRIBUTO: almacenará las características de los atributos a utilizar en el sistema, sus campos se detallan a continuación:
 - a. Un código identificador.
 - b. Un nombre descriptivo.
 - c. En el campo VALOR_DE_VERDAD se guardará el valor máximo que representa a ese atributo.
 - d. En el campo VALOR_DE_VERDAD_DERECHO se guardará el valor más significativo del atributo que se encuentra a la derecha del atributo en cuestión, este campo no aplica cuando el atributo tiene un tipo de función Gamma.
 - e. En el campo VALOR_DE_VERDAD_IZQUIERDO se guardará el valor más significativo del atributo que se encuentra a la izquierda del atributo en cuestión, este campo no aplica cuando el atributo tiene un tipo de función L.
 - f. En los casos donde el atributo tenga una función de tipo Pi el campo VALOR_DE_VERDAD_FINAL será quien guardará el valor superior del intervalo donde el atributo es más significativo
 - g. Los campos TIPO_FUNCION y DIMENSION hacen referencia a las entidades del mismo nombre, indicando cuál es la relación que se tiene con ellas.
4. REGLA: además de un código de identificación y una descripción, esta entidad guardará:
 - a. El valor de aceptación de la regla, el cual servirá como referencia para comparar con el resultado del registro evaluado.
 - b. El grado de compensación de la regla, el cual se usará para evaluarla por medio del método de compensación, y no por el de agregación.

- c. La referencia al código de la dimensión que la regla tiene configurada como dimensión difusa de salida.
5. REGLA_DIMENSION: esta es una entidad de tipo asociativo, en la cual se guardará el código de la regla, y la dimensión difusa que forma parte de las entradas de la regla.
6. DETALLE_REGLA: en esta entidad se especificará como está detallada la regla, por ello tiene un campo que hace referencia a esta, también hace una referencia al atributo que tiene configurado como salida, y el campo GRADO_DE_SOPORTE, almacenará un valor entre 0 y 1, que representa el peso que tiene ese detalle de regla respecto de los demás.
7. DETALLE_REGLA_ATRIBUTO: al igual que la entidad REGLA_DIMENSION, esta también es una entidad de tipo asociativo, la cual almacena el código de la entidad DETALLE_REGLA asociado al código del atributo que forma parte de las entradas del detalle de la regla.

Para obtener las variables de entrada se hará por medio de la obtención de datos directa de la base de datos, el código mostrado corresponde a sentencias de SQL procesadas desde el analizador de consultas de Microsoft versión 8.00.194.

Los datos de entrada serán almacenados en una tabla que será utilizada solo para el procesamiento de la información, y que, si el usuario lo desea, puede ser borrada inmediatamente después de haber sido utilizada. El nombre de la tabla para procesamiento es DATOS_ORIGEN_CLIENTE.

La tabla CLIENTE, mostrada en las tablas I y II representará la base de datos existente. Los datos mostrados en la tabla III contienen información sobre tres dimensiones difusas, las dimensiones 1 y 2 serán usadas como dimensiones de entrada, mientras que la dimensión 5 será usada como dimensión de salida.

Tabla III. Datos de las dimensiones ejemplo

Dimension	Nombre	Unidad_de_medida	Origen_datos
1	sueldo	Quetzales	SELECT CLI.SUELDO FROM CLIENTE CLI WHERE CLI.CLIENTE = DOC_ID
2	edad	Años	SELECT DATEDIFF (dd CLI.FECHA_NACIMIENTO getDate())/365 FROM CLIENTE CLI WHERE CLI.CLIENTE = DOC_ID
5	descuento	Porcentaje	--

Los datos de la tabla IV muestran la información sobre una regla, cuyo valor de aceptación es 65, es decir, que los valores iguales o superiores a este valor serán aceptados, y que su dimensión de salida o resultado es la número 5, la cual corresponde a la dimensión DESCUENTO, esta regla no tiene grado de compensación.

Tabla IV. Datos de la regla ejemplo

Regla	Descripción	Valor aceptación	Grado_de_compensacion	Dimension
1	determinación de descuentos	65	NULL	5

En la tabla V se muestran los registros de la entidad asociativa REGLA_DIMENSION que muestra que las dimensiones 1 y 2, o sea SUELDO y EDAD, están asociadas a la regla 1 como dimensiones de entrada. Las tablas I, II, III, IV y V servirán para explicar la carga de datos a la tabla de procesamiento.

Tabla V. Asociación entre las dimensiones y la regla

Regla	Dimension
1	1
1	2

El primer paso para la carga de datos será la creación de la tabla de procesamiento DATOS_ORIGEN_CLIENTE, en donde se declara un único campo llamado DOC_ID, el cual será la llave primaria de esta tabla, luego serán cargados los códigos de los clientes en la tabla de procesamiento, para tener sincronizados los identificadores de ambas tablas. El código de estas operaciones se muestra en la tabla VI.

Tabla VI. Código para la carga de datos

```
--CREA TABLA DE DATOS DE PROCESAMIENTO
CREATE TABLE Datos_Origen_cliente
(   DOC_ID int NOT NULL PRIMARY KEY   )
GO
--OBTIENE DATOS LLAVE DEL CLIENTE
INSERT INTO DATOS_ORIGEN_CLIENTE(DOC_ID)
SELECT CLIENTE FROM CLIENTE
```

El próximo paso es elegir la regla que se debe procesar, para efectos del ejemplo se hará con la regla 1, luego se deberá determinar cuáles son las dimensiones de entrada que tiene asociadas la regla que se está evaluando. El código a utilizar para realizar esta búsqueda se muestra en la tabla VII.

Tabla VII. Selección de dimensiones que están en la regla solicitada

```
--SELECCION DE DIMENSIONES QUE ESTAN EN LA REGLA SOLICITADA
SELECT      DIM.DIMENSION, DIM.NOMBRE 'DIMENSION_NOM'
FROM DIMENSION DIM
WHERE DIM.DIMENSION IN (SELECT DIMENSION
                        FROM  REGLA_DIMENSION
                        WHERE REGLA = 1)
```

El siguiente paso es agregar a la tabla de procesamiento los campos que corresponden a las dimensiones encontradas en el paso anterior, en este caso debemos agregar los campos SUELDO y EDAD, ya que estos son los nombres de las dimensiones asociadas como entrada a la regla 1.

Después se deberá cargar los datos en estos campos con la información de la tabla CLIENTE, tomando como referencia el campo ORIGEN_DATOS de la tabla DIMENSION, el cual especifica una consulta de cómo deben obtenerse dichos datos, el código de estas operaciones se puede observar en la tabla VIII.

Tabla VIII. Carga de datos de las dimensiones de entrada

```
--AGREGA DIMENSION SUELDO
ALTER TABLE DATOS_ORIGEN_CLIENTE ADD SUELDO money null

--INSERTA DATOS EN CAMPO SUELDO
UPDATE      DATOS_ORIGEN_CLIENTE
SET SUELDO = (   SELECT      CLI.SUELDO
                  FROM  CLIENTE CLI
                  WHERE CLI.CLIENTE = DOC_ID)

--AGREGA DIMENSION EDAD
ALTER TABLE DATOS_ORIGEN_CLIENTE ADD EDAD SMALLINT NULL

--INSERTA DATOS EN EL CAMPO EDAD
UPDATE      DATOS_ORIGEN_CLIENTE
SET EDAD = (   SELECT
                  DATEDIFF(dd, CLI.FECHA_NACIMIENTO, getDate())/365
                  FROM  CLIENTE CLI
                  WHERE CLI.CLIENTE = DOC_ID)
```

Hasta este momento se tiene la tabla de procesamiento de datos con tres campos: el identificador de cada registro, que corresponde al identificador de registro de la tabla CLIENTE, el campo SUELDO y el campo EDAD, ambos cargados con los valores que corresponden, obtenidos de los datos de la tabla CLIENTE. La apariencia de la tabla se muestra en la tabla IX.

Tabla IX. Datos de las dimensiones cargados en la tabla de procesamiento

DOC_ID	SUELDO	EDAD
1	3500.00	24
2	3500.00	24
3	8000.00	25
4	12000.00	26
5	30000.00	32
6	7500.00	23
7	6100.00	36
8	6500.00	42
9	14000.00	30
10	1200.00	26

2.4. Funciones de pertenencia

Las funciones de pertenencia son las que indican el grado de pertenencia de determinado valor según sea el tipo de atributo, es decir que, para cada atributo, el valor evaluado obtendrá, por medio de la función de pertenencia asociada al atributo, un valor entre 0 y 1 el cual representa en qué medida el valor pertenece a ese atributo. Al sumar los grados de pertenencia de un registro a los atributos de una misma dimensión, el total debe ser igual a uno.

Para poder determinar cuál es la forma de la función de pertenencia, se deberá hacer en base al tipo de función que tenga el atributo, y a los valores de verdad asociados a éste.

Utilizando la fórmula de punto y pendiente para elaborar gráficas de rectas, se realizó una adaptación de la misma a los valores de verdad que tienen los atributos, y luego por medio de la representación de funciones escalonadas, se determinó cuál debía ser la función de pertenencia según el tipo de función que se tratara.

En la tabla X se describen los valores de la pendiente y el intercepto de cada uno de los tipos de función utilizados en el sistema, además de mostrar la función en su representación escalonada, la cual también está en función de los valores de verdad del atributo.

Tabla X. Funciones de pertenencia expresadas por medio de fórmulas

Tipo función	Valores de m y b en f(x)=mx+b	f(x)=mx+b
L	$m = \frac{1}{VV - VVD} \quad b = -VVD * m$	$f(x) = \begin{cases} 1 & x \leq VV \\ mx + b & VV < x < VVD \\ 0 & x \geq VVD \end{cases}$
Λ	$m_1 = \frac{1}{VV - VVI} \quad b_1 = -VVI * m_1$ $m_2 = \frac{1}{VV - VVD} \quad b_2 = -VVD * m_2$	$f(x) = \begin{cases} 0 & x \leq VVI \\ m_1x + b_1 & VVI < x < VV \\ m_2x + b_2 & VV < x < VVD \\ 0 & x \geq VVD \end{cases}$
Π	$m_1 = \frac{1}{VV - VVI} \quad b_1 = -VVI * m_1$ $m_2 = \frac{1}{VVF - VVD} \quad b_2 = -VVD * m_2$	$f(x) = \begin{cases} 0 & x \leq VVI \\ m_1x + b_1 & VVI < x < VV \\ 1 & VV \leq x \leq VVF \\ m_2x + b_2 & VVF < x < VVD \\ 0 & x \geq VVD \end{cases}$
Γ	$m = \frac{1}{VV - VVI} \quad b = -VVI * m$	$f(x) = \begin{cases} 1 & x \geq VV \\ mx + b & VVI < x < VV \\ 0 & x \leq VVI \end{cases}$
VV = Valor de verdad VVF = Valor de verdad final VVD = Valor de verdad de la derecha VVI = Valor de verdad de la izquierda		

En la tabla XI se muestra el contenido de la tabla TIPO_FUNCION, en donde se puede ver el código asociado a cada nombre del tipo de función a utilizarse en el sistema, y que es referenciado por los registros de la tabla atributo.

Tabla XI. Tipos de función usados en el sistema

Tipo_funcion	Nombre
1	Ele
2	Lambda
3	PI
4	Gamma

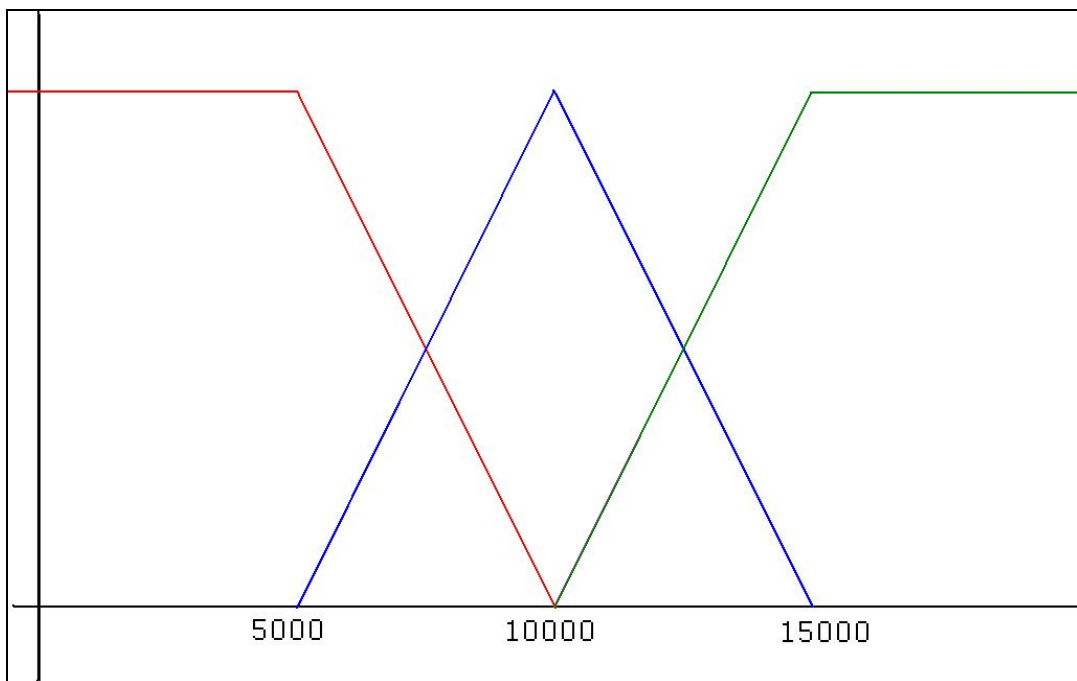
Para efectos de ejemplo se usarán los atributos asociados a las dimensiones de entrada mostrados en la tabla III, para poder elaborar las gráficas de las funciones asociadas a dichos atributos. Los datos de los atributos se muestran en la tabla XII.

Tabla XII. Datos de los atributos de las dimensiones ejemplo

Atributo	Nombre	Valor_de_verdad	Valor_de_verdad_final	Valor_de_verdad_izquierdo	Valor_de_verdad_derecho	Tipo_funcion	Dimension
1	alto	15000	NULL	10000		4	1
2	medio	10000	NULL	5000	15000	2	1
3	bajo	5000	NULL	NULL	10000	1	1
4	joven	18	NULL	NULL	30	1	2
5	media	30	40	18	55	3	2
6	avanzado	55	NULL	40	NULL	4	2
7	regular	50	NULL	25	75	2	5
8	mucho	75	NULL	50	NULL	4	5
9	poco	25	NULL	NULL	50	1	5

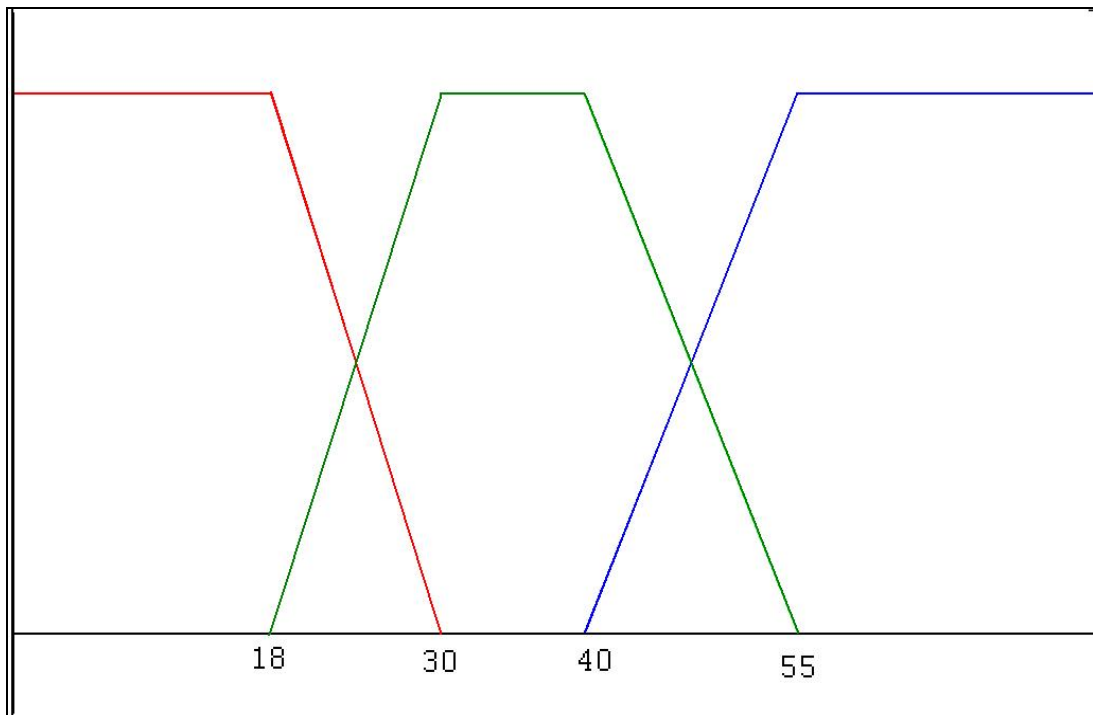
En la figura 6 se muestra la representación gráfica de las funciones de pertenencia asociadas a los atributos de la dimensión SUELDO, las cuales vistas de izquierda a derecha corresponden a los atributos BAJO, MEDIO, y ALTO respectivamente.

Figura 6. Funciones de pertenencia de los atributos de la dimensión SUELDO



En la figura 7 se muestran las gráficas de las funciones de pertenencia asociadas a los atributos de la dimensión EDAD, las cuales vistas de izquierda a derecha corresponden a los atributos JOVEN, MEDIA, y AVANZADO respectivamente.

Figura 7. Funciones de pertenencia de los atributos de la dimensión EDAD



Como puede observarse entre la figura 6 y la figura 7, existe una marcada diferencia entre la forma de la función del atributo MEDIA de la dimensión SUELDO, y la función del atributo MEDIA de la dimensión EDAD, esto es debido a que el atributo MEDIA de la dimensión EDAD tiene definido su máximo valor de verdad en un intervalo, mientras que el atributo MEDIA de la dimensión SUELDO tiene definido su máximo valor de verdad en un solo punto.

3. DETERMINACIÓN DE REGLAS

3.1. Definición de variables de salida

Las variables de salida son las dimensiones difusas que surgen de los resultados de las reglas, cada regla se define como la combinación de por lo menos dos variables o dimensiones difusas de entrada en la parte izquierda de la implicación de la regla, y una variable o dimensión difusa en la parte derecha, que es la variable de salida en cuestión.

Para seguir con el ejemplo, en la sección 2.3, en la tabla IV se muestran los datos de la regla 1 la cual está definida de la siguiente forma:

$$SUELDO Y EDAD \Rightarrow DESCUENTO$$

Esto quiere decir que se aplicará cierto descuento si la persona cumple con los requisitos de sueldo y edad, estos dos últimos son parte de las variables de entrada que se cargaron en el capítulo anterior, mientras que la dimensión descuento depende de los valores ingresados por sueldo y descuento y de cómo sea procesada la información por el sistema, para cada registro que se evalúe.

Las variables de salida definidas por medio de reglas, deben ser también dimensiones difusas, con sus respectivos atributos difusos, ya que la forma en que los resultados son devueltos también es por medio de conjuntos difusos, o sea, atributos y grados de pertenencia, entonces no se puede introducir una variable de salida simple, o sea no difusa, ya que no soportaría los resultados al llegar al punto de la conversión de valores, si no tienen los atributos difusos asociados.

Cuando se están definiendo las reglas como la combinación de varias dimensiones, y da como resultado otra dimensión, lo que se forma es la plantilla de la regla, o en términos de objetos lo que se tiene es la definición de la clase regla, es decir sabemos que el descuento depende del sueldo y de la edad, pero ¿de qué forma? Hasta este punto es algo que no se puede responder, ya que la regla fue definida solo de forma general, y hace falta detallarla, explicando qué atributos de la dimensión sueldo, combinados con los atributos de la dimensión edad resultan en determinados atributos de la dimensión descuento, en términos de objetos esto es hacer las instancias de la regla.

3.2. Detalle de reglas

Las reglas para ser detalladas deben estar definidas en función de los atributos de las dimensiones de entrada, y como salida deberán tener los atributos de la dimensión de salida. El número de detalles o instancias de la regla será igual al producto entre la cantidad de atributos de las dimensiones de entrada, o sea si una dimensión de entrada tiene tres atributos, y la otra dimensión de entrada tiene cuatro atributos se pueden formar doce instancias o detalles de la regla.

Para poder determinar los posibles detalles de las reglas se realiza un producto cartesiano entre los atributos de cada una de las dimensiones difusas asociadas a la regla que se esté evaluando.

A manera de ejemplo, utilizando la regla 1, y las dimensiones SUELDO y EDAD, se puede verificar que ambas tienen un total de tres atributos difusos cada una, lo cual nos indica que la regla 1 tendrá 9 detalles o instancias. El código para determinar los detalles de la regla 1 se muestra en la tabla XIII.

Tabla XIII. Código para determinar el detalle de reglas

```
--PARA CREAR EL DETALLE DE REGLAS
SELECT      a.*, b.*
FROM ( SELECT ATR.ATRIBUTO,  ATR.NOMBRE 'ATRIBUTO_NOM'
        FROM ATRIBUTO ATR, DIMENSION DIM
        WHERE ATR.DIMENSION = DIM.DIMENSION
        AND   DIM.DIMENSION = 1)a,
      ( SELECT ATR.ATRIBUTO,  ATR.NOMBRE 'ATRIBUTO_NOM'
        FROM ATRIBUTO ATR, DIMENSION DIM
        WHERE ATR.DIMENSION = DIM.DIMENSION
        AND   DIM.DIMENSION = 2)b
ORDER BY a.ATRIBUTO1 DESC, b.ATRIBUTO2
```

Si la regla a evaluar tuviera más de dos dimensiones de entrada, por cada una se debe agregar en la parte del FROM el código mostrado en la tabla XIV, haciendo referencia a la dimensión de la cual se desea obtener los atributos, en el SELECT una referencia al nuevo alias definido en el FROM, y en la cláusula ORDER BY el orden en el que se desea que aparezcan las combinaciones.

Tabla XIV. Código a agregar cuando hay más de dos dimensiones

```
--Agregado para el FROM cuando hay mas de dos dimensiones
( SELECT ATR.ATRIBUTO, ATR.NOMBRE 'ATRIBUTO_NOM'
FROM ATRIBUTO ATR, DIMENSION DIM
WHERE ATR.DIMENSION = DIM.DIMENSION
AND DIM.DIMENSION = <codigoDimension><alias>
```

El resultado de la búsqueda de detalles de la regla 1 se muestra en la tabla XV, en ella se puede apreciar cómo todos los atributos de las dimensiones SUELDO y EDAD se combinan entre sí.

Tabla XV. Atributos de entrada combinados

ATRIBUTO1	ATRIBUTO_NOM1	ATRIBUTO2	ATRIBUTO_NOM2
3	Bajo	4	joven
3	Bajo	5	media
3	Bajo	6	avanzado
2	Medio	4	joven
2	Medio	5	media
2	Medio	6	avanzado
1	Alto	4	joven
1	Alto	5	media
1	Alto	6	avanzado

A partir de la tabla XV se hará la inserción de los registros de la tabla DETALLE_REGLA_ATRIBUTO. Esta tabla, como se indicó en la sección 2.3 es de tipo asociativo, entre la tabla DETALLE_REGLA y la tabla ATRIBUTO, pero para poder llenarlas hace falta definir cuáles serán los atributos que tendrá como salida cada uno de los detalles de regla, y cuál será su grado de soporte.

3.2.1. Asignación de atributos de salida

Para asignar los atributos de salida se tomará como referencia la tabla XV, específicamente las columnas ATRIBUTO_NOM1 y ATRIBUTO_NOM2, las cuales vamos a copiar a una hoja electrónica, para poder manipularla manualmente, y así realizar pasos que dependen del criterio del usuario. Luego de tener las columnas mencionadas se deben numerar los resultados, en este caso de 1 a 9.

El siguiente paso es colocar en una cuarta columna un atributo perteneciente a la dimensión de salida, el cual corresponderá a la salida de cada detalle de la regla, tal como se muestran los datos en la tabla XVI luego de haberle ingresado los valores de salida a cada detalle de regla.

Tabla XVI. Valores de salida correspondientes a cada pareja de los datos de entrada

Número	Entrada1	Entrada2	Salida
1	Bajo	joven	mucho
2	Bajo	media	regular
3	Bajo	avanzado	poco
4	medio	joven	regular
5	medio	media	regular
6	medio	avanzado	poco
7	Alto	joven	regular
8	alto	media	mucho
9	alto	avanzado	mucho

3.2.2. Asignación de grados de soporte

Los detalles de reglas la mayoría de las veces repiten el atributo de salida, es decir, varios detalles de reglas pueden tener el mismo atributo de la dimensión de salida como resultado de su evaluación, situación que es muy normal. Por ejemplo, haciendo referencia a la tabla XVI, se puede observar que los detalles de regla número 1, 8 y 9, todos tienen como salida al atributo MUCHO; los detalles de regla número 2, 4, 5 y 7 tienen como salida al atributo REGULAR; y los detalles de regla número 3 y 6 tienen como salida al atributo POCO.

Según el criterio del usuario que esté configurando los detalles de reglas puede asignarles un peso o prioridad a cada una, donde el máximo valor a asignarle a cada una es 1, y el mínimo es cero, aunque éste último anularía el detalle de regla puede ser usado para representar que ese detalle no debe ser tomado en cuenta.

Para poder darle prioridad a un detalle de regla se requiere asignarle un grado de soporte, el cual representa un porcentaje del valor resultante del detalle de la regla. El grado de soporte predeterminado de cada detalle de regla es uno o sea del cien por ciento; en la medida en que el grado de soporte disminuya su valor, entonces el resultado del detalle de la regla también va a disminuir cuando se aplique el proceso de composición el detalle de regla y su grado de soporte.

En la tabla XVII se muestran los datos de la hoja electrónica luego de haberle agregado una quinta columna en donde manualmente, y a criterio del usuario, se le asignó un grado de soporte a cada uno de los detalles que se habían propuesto.

Tabla XVII. Detalle de reglas con grado de soporte

Número	Entrada1	Entrada2	Salida	Grado de soporte
1	bajo	joven	mucho	1
2	bajo	media	regular	0.9
3	bajo	avanzado	poco	1
4	medio	joven	regular	1
5	medio	media	regular	1
6	medio	avanzado	poco	1
7	alto	joven	regular	1
8	alto	media	mucho	0.9
9	alto	avanzado	mucho	1

Lo que los grados de soporte representan según los datos de la tabla XVII, es que cuando se evalúen los detalles de regla que tienen como salida al atributo REGULAR, los detalles de regla 4, 5 y 7, tendrán prioridad o mas peso sobre el detalle de regla número 2, ya que éste último tendrá solo el noventa por ciento del valor del resultado de la evaluación, mientras que el resto tendrá un cien por ciento. De manera similar con la evaluación de los detalles de regla que tienen como salida al atributo MUCHO, los detalles de regla número 1 y 9, tendrán mas peso que el detalle de regla número 8 al momento de la evaluación entre sí.

Las anotaciones realizadas en la hoja electrónica podrían haberse realizado con papel y lápiz. Esto ya queda a criterio de cada usuario como prefiera organizar la información, pues al final la misma deberá ser ingresada independientemente de donde provenga.

Finalmente con los datos de la tabla XVII, se pueden insertar los registros a las tablas DETALLE_REGLA y DETALLE_REGLA_ATRIBUTO, las cuales se muestran en las tablas XVIII y XIX respectivamente, luego de haberse realizado la inserción.

Tabla XVIII. Datos de la tabla DETALLE_REGLA

Detalle_regla	Grado_de_soporte	Regla	Atributo
1	1	1	8
2	0.9	1	7
3	1	1	9
4	1	1	7
5	1	1	7
6	1	1	9
7	1	1	7
8	0.9	1	8
9	1	1	8

Tabla XIX. Datos de la tabla DETALLE_REGLA_ATRIBUTO

Detalle_regla	Atributo
1	3
1	4
2	3
2	5
3	3
3	6
4	2
4	4
5	2
5	5
6	2
6	6
7	1
7	4
8	1
8	5
9	1
9	6

Al terminar este capítulo, tenemos los datos organizados y almacenados con la información sobre la definición de las reglas. En el siguiente capítulo se determinarán los conjuntos difusos, y el proceso de evaluación de reglas.

4. DETERMINACIÓN DE LOS CONJUNTOS DIFUSOS

4.1. Determinación de pertenencia

El grado o porcentaje de pertenencia representa qué tanto pertenece un valor a determinado atributo, a cada valor estará asociado cada uno de los atributos de la dimensión a la que corresponde el valor, con su respectivo grado de pertenencia.

En la tabla de procesamiento de datos `DATOS_ORIGEN_CLIENTE` se cuenta con tres campos, el identificador de cada registro, el campo `SUELDO` y el campo `EDAD`, tal como fue mostrado en la tabla IX. Para poder determinar cuáles son los grados de pertenencia de determinado valor, a la tabla `DATOS_ORIGEN_CLIENTE` se le agregará una columna por cada atributo de las dimensiones que corresponden a los valores que actualmente se encuentran en la tabla.

El nombre de las columnas a agregar estará formado por el nombre de la dimensión y por el nombre del atributo, esto se debe a que dos dimensiones pueden tener uno ó varios atributos con el mismo nombre, entonces en este caso la dimensión servirá para distinguir de cual se está hablando. El código mostrado en la tabla XX devuelve los nombres de las columnas que se deben agregar a la tabla, el tipo de función que el atributo tiene asociado y los valores de verdad del atributo, para los atributos de la dimensión `SUELDO`, la cual tiene asociado el código 1. En la tabla XXI se muestra el resultado de la ejecución de esta consulta.

Tabla XX. Código de consulta de nombres para los atributos de la dimensión SUELDO

```

SELECT  DIM.NOMBRE + '_' + ATR.NOMBRE 'NOMBRE_COLUMNA',
        ATR.TIPO_FUNCION,
        ATR.VALOR_DE_VERDAD 'V_VERDAD',
        ATR.VALOR_DE_VERDAD_FINAL 'V_VERDAD_FINAL',
        ATR.VALOR_DE_VERDAD_IZQUIERDO 'V_VERDAD_IZQ',
        ATR.VALOR_DE_VERDAD_DERECHO 'V_VERDAD_DER'
FROM ATRIBUTO ATR, DIMENSION DIM
WHERE ATR.DIMENSION = DIM.DIMENSION
AND DIM.DIMENSION = 1
    
```

Tabla XXI. Nombres para los atributos de la dimensión SUELDO

NOMBRE_COLUMNA	TIPO_FUNCION	V_VERDAD	V_VERDAD_FINAL	V_VERDAD_IZQ	V_VERDAD_DER
sueldo_alto	4	15000	NULL	10000	NULL
sueldo_medio	2	10000	NULL	5000	15000
sueldo_bajo	1	5000	NULL	NULL	10000

Una vez se obtienen estos datos se debe proceder a agregar estos campos a la tabla DATOS_ORIGEN_CLIENTE, con un tipo de dato decimal de una máximo de 3 posiciones enteras y 2 decimales, y con opción de dejarlos en blanco. El código utilizado para agregar los tres campos que corresponden a atributos de la dimensión SUELDO se muestra en la tabla XXII.

Tabla XXII. Código para agregar los campos de los atributos de la dimensión SUELDO

```
ALTER TABLE DATOS_ORIGEN_CLIENTE ADD SUELDO_ALTO decimal(5,2) null
ALTER TABLE DATOS_ORIGEN_CLIENTE ADD SUELDO_MEDIO decimal(5,2) null
ALTER TABLE DATOS_ORIGEN_CLIENTE ADD SUELDO_BAJO decimal(5,2) null
```

Para poder agregar las columnas correspondientes a los atributos asociados a la dimensión EDAD, la cual tiene el código 2, se muestra en la tabla XXIII el código que devuelve los nombres que se deben agregar a la tabla, el tipo de función que el atributo tiene asociado y los valores de verdad del atributo. En la tabla XXIV se muestra el resultado de la ejecución de esta consulta, y la tabla XXV muestra el código para agregar los campos correspondientes.

Tabla XXIII. Código de consulta de nombres para los atributos de la dimensión EDAD

```
SELECT DIM.NOMBRE + '_' + ATR.NOMBRE 'NOMBRE_COLUMNA',
       ATR.TIPO_FUNCION,
       ATR.VALOR_DE_VERDAD 'V_VERDAD',
       ATR.VALOR_DE_VERDAD_FINAL 'V_VERDAD_FINAL',
       ATR.VALOR_DE_VERDAD_IZQUIERDO 'V_VERDAD_IZQ',
       ATR.VALOR_DE_VERDAD_DERECHO 'V_VERDAD_DER'
FROM ATRIBUTO ATR, DIMENSION DIM
WHERE ATR.DIMENSION = DIM.DIMENSION
AND DIM.DIMENSION = 2
```

Tabla XXIV. Nombres para los atributos de la dimensión EDAD

NOMBRE_COLUMNA	TIPO_FUNCIÓN	V_VERDAD	V_VERDAD_FINAL	V_VERDAD_IZQ	V_VERDAD_DER
edad_joven	1	18	NULL	NULL	30
edad_media	3	30	40	18	55
edad_avanzado	4	55	NULL	40	NULL

Tabla XXV. Código para agregar los campos de los atributos de la dimensión EDAD

```
ALTER TABLE DATOS_ORIGEN_CLIENTE ADD EDAD_JOVEN decimal(5,2) null
ALTER TABLE DATOS_ORIGEN_CLIENTE ADD EDAD_MEDIA decimal(5,2) null
ALTER TABLE DATOS_ORIGEN_CLIENTE ADD EDAD_AVANZADO decimal(5,2) null
```

Hasta este punto se han agregado seis columnas a la tabla DATOS_ORIGEN_CLIENTE, las cuales corresponden a los seis atributos asociados a las dimensiones de entrada de la regla que se está evaluando. Lo que se actualizará por cada registro es el grado de pertenencia que tienen los valores que se están evaluando en cada uno de los atributos. En la tabla XXVI se muestra la estructura de la tabla luego de agregarle las columnas arriba mencionadas.

Tabla XXVI. Estructura de la tabla luego de agregarle las columnas de los atributos de entrada

DOC_ID	SUELDO	EDAD	SUELDO_ ALTO	SUELDO_ MEDIO	SUELDO_ BAJO	EDAD_ JOVEN	EDAD_ MEDIA	EDAD_ AVANZADA
1	3500	24						
2	3500	24						
3	8000	25						
4	12000	26						
5	30000	32						
6	7500	23						
7	6100	36						
8	6500	42						
9	14000	30						
10	1200	26						

Para poder determinar cual es el grado de pertenencia de un valor se debe utilizar una función de pertenencia definida para cada atributo que se esté evaluando, la función a utilizar dependerá del tipo de función y de los valores de verdad que tiene el atributo. Esta información se puede obtener de las tablas XXI y XXIV, correspondientes a los atributos SUELDO y EDAD respectivamente. Al combinar esta información con las fórmulas mostradas en la tabla X, se puede escribir un código que inserte el grado de pertenencia en la columna de la tabla DATOS_CLIENTE_ORIGEN que le corresponde. Se debe realizar una inserción por cada columna. El código mostrado en la tabla XXVII, corresponde a la inserción de los datos de la columna SUELDO_ALTO, la cual tiene un tipo de función gamma. En la tabla XXVIII se muestra el código de inserción de los datos de la columna SUELDO_MEDIO, la cual tiene un tipo de función lambda.

Tabla XXVII. Inserción de los datos de la columna SUELDO_ALTO

```
--INSERCIÓN DE VALORES PARA UN TIPO DE FUNCION 4 (gamma)
UPDATE DATOS_ORIGEN_CLIENTE
SET SUELDO_ALTO = a.VALOR
FROM ( SELECT CASE
          WHEN SUELDO<=10000 THEN
            0
          WHEN SUELDO>=15000 THEN
            1
          WHEN SUELDO>10000 AND SUELDO<15000 THEN
            (SUELDO-10000)/(15000-10000)
          END 'VALOR', DOC_ID
        FROM DATOS_ORIGEN_CLIENTE DOC
      ) a
WHERE a.DOC_ID = DATOS_Origen_CLIENTE.DOC_ID
```

Tabla XXVIII. Inserción de los datos de la columna SUELDO_MEDIO

```
--ACTUALIZACION PARA UN TIPO DE FUNCION 2 (lambda)
UPDATE DATOS_ORIGEN_CLIENTE
SET SUELDO_MEDIO = a.VALOR
FROM ( SELECT CASE
          WHEN SUELDO<=5000 THEN
            0
          WHEN SUELDO>5000 AND SUELDO<=10000 THEN
            (SUELDO-5000)/(10000-5000)
          WHEN SUELDO>=10000 AND SUELDO<15000 THEN
            (SUELDO-15000)/(10000-15000)
          WHEN SUELDO>=15000 THEN
            0
          END 'VALOR', DOC_ID
        FROM DATOS_ORIGEN_CLIENTE DOC
      ) a
WHERE a.DOC_ID = DATOS_Origen_CLIENTE.DOC_ID
```

Finalmente en la tabla XXIX se muestra el código de inserción de los datos de la columna SUELDO_BAJO, la cual tiene un tipo de función L.

Tabla XXIX. Inserción de los datos de la columna SUELDO_BAJO

```
--ACTUALIZACION PARA UN TIPO DE FUNCION 1 (ELE)
UPDATE      DATOS_ORIGEN_CLIENTE
SET SUELDO_BAJO = a.VALOR
FROM (      SELECT      CASE
                WHEN SUELDO>=10000 THEN
                0
                WHEN SUELDO>5000 AND SUELDO<10000 THEN
                (SUELDO-10000)/(5000-10000)
                WHEN SUELDO<=5000 THEN
                1
                END 'VALOR', DOC_ID
        FROM  DATOS_ORIGEN_CLIENTE DOC
    ) a
WHERE a.DOC_ID = DATOS_Origen_CLIENTE.DOC_ID
```

De manera similar con las columnas EDAD_JOVEN, EDAD_MEDIA y EDAD_AVANZADO, las cuales tienen los tipos de función L, PI y Gamma respectivamente, se muestra el código para la inserción de los valores de pertenencia en las tablas XXX, XXXI y XXXII.

Tabla XXX. Inserción de los datos de la columna EDAD_JOVEN

```
--ACTUALIZACION PARA UN TIPO DE FUNCION 1 (ELE)
UPDATE      DATOS_ORIGEN_CLIENTE
SET EDAD_JOVEN = a.VALOR
FROM (      SELECT      CASE
                WHEN EDAD<=18 THEN
                1
                WHEN EDAD>=30 THEN
                0
                WHEN EDAD>18 AND EDAD<30 THEN
                (CAST(EDAD AS DECIMAL(5,2))-30)/(18-30)
                ELSE
                -1
                END 'VALOR', DOC_ID, EDAD
        FROM  DATOS_ORIGEN_CLIENTE DOC
    ) a
```



```
WHERE a.DOC_ID = DATOS_Origen_CLIENTE.DOC_ID
```

Tabla XXXI. Inserción de los datos de la columna EDAD_MEDIA

```
--ACTUALIZACION PARA UN TIPO DE FUNCION 3 (pi)
UPDATE      DATOS_ORIGEN_CLIENTE
SET edad_media = a.VALOR
FROM (      SELECT      CASE
                WHEN EDAD<=18 THEN
                    0
                WHEN EDAD>18 AND EDAD<30 THEN
                    (CAST(EDAD AS DECIMAL(5,2))-18)/(30-18)
                WHEN EDAD>=30 AND EDAD<=40 THEN
                    1
                WHEN EDAD>40 AND EDAD<55 THEN
                    (CAST(EDAD AS DECIMAL(5,2))-55)/(40-55)
                WHEN EDAD>=55 THEN
                    0
                END 'VALOR', DOC_ID
        FROM  DATOS_ORIGEN_CLIENTE DOC
    ) a
WHERE a.DOC_ID = DATOS_Origen_CLIENTE.DOC_ID
```

Tabla XXXII. Inserción de los datos de la columna EDAD_AVANZADO

```
--ACTUALIZACION PARA UN TIPO DE FUNCION 4 (gamma)
UPDATE      DATOS_ORIGEN_CLIENTE
SET EDAD_AVANZADO = a.VALOR
FROM (      SELECT      CASE
                WHEN EDAD>=55 THEN
                    1
                WHEN EDAD>40 AND EDAD<55 THEN
                    (CAST(EDAD AS DECIMAL(5,2))-40)/(55-40)
                WHEN EDAD<=40 THEN
                    0
                END 'VALOR', DOC_ID, EDAD
        FROM  DATOS_ORIGEN_CLIENTE DOC
    ) a
WHERE a.DOC_ID = DATOS_Origen_CLIENTE.DOC_ID
```

En la tabla XXXIII se muestra el contenido de la tabla DATOS_ORIGEN_CLIENTE luego de haber insertado los valores correspondientes al grado de pertenencia.

Tabla XXXIII. Valores de los grados de pertenencia insertados

DOC_ID	SUELDO	EDAD	SUELDO_ ALTO	SUELDO_ MEDIO	SUELDO_ BAJO	EDAD_ JOVEN	EDAD_ MEDIA	EDAD_ AVANZADA
1	3500	24	0	0	1	0.5	0.5	0
2	3500	24	0	0	1	0.5	0.5	0
3	8000	25	0	0.6	0.4	0.42	0.58	0
4	12000	26	0.4	0.6	0	0.33	0.67	0
5	30000	32	1	0	0	0	1	0
6	7500	23	0	0.5	0.5	0.58	0.42	0
7	6100	36	0	0.22	0.78	0	1	0
8	6500	42	0	0.3	0.7	0	0.87	0.13
9	14000	30	0.8	0.2	0	0	1	0
10	1200	26	0	0	1	0.33	0.67	0

4.2. Agregación

Este es el proceso mediante el cual se evalúa cada uno de los detalles de regla en cada uno de los registros, aplicando el operador mínimo a los valores de pertenencia que indique el detalle de regla, es decir, se deben obtener los resultados de todos los detalles de regla para cada registro. Para el ejemplo que se está utilizando, por cada registro de la tabla DATOS_ORIGEN_CLIENTE se evaluarán los nueve detalles de regla que se definieron en la tabla XVII, estos nueve valores por registro son el resultado del proceso de agregación.

La forma en que se realizará este proceso es distinto de como se han venido haciendo los pasos anteriores, ya que éste no se hará desde el manejador de la base de datos directamente, dado que implica el manejo de

dos conjuntos de valores, por un lado los registros de la tabla de procesamiento de datos, y por el otro los detalles de la regla que se está evaluando.

En este proceso los resultados no serán insertados en la tabla DATOS_ORIGEN_CLIENTE, ya que los mismos deben pasar posteriormente por el proceso de composición, así que los datos serán manejados sobre una tabla temporal que guardará el código de cada cliente y los resultados de cada uno de los detalles de regla, el nombre de la tabla es DETALLE_REGLA_DOC.

En las tablas siguientes se muestra el código utilizado para llevar a cabo el proceso de agregación, acompañado de la explicación del flujo del proceso. El código para la creación y llenado inicial de la tabla DETALLE_REGLA_DOC se muestra en la tabla XXXIV

Tabla XXXIV. Creación y llenado de la tabla DETALLE_REGLA_DOC

```
--CREA TABLA TEMPORAL
CREATE TABLE Detalle_regla_DOC
(ID integer not null primary key,
 _1 decimal(5,2) null,
 _2 decimal(5,2) null,
 _3 decimal(5,2) null,
 _4 decimal(5,2) null,
 _5 decimal(5,2) null,
 _6 decimal(5,2) null,
 _7 decimal(5,2) null,
 _8 decimal(5,2) null,
 _9 decimal(5,2) null)

--OBTIENE DATOS LLAVE DE LA TABLA DATOS_ORIGEN_CLIENTE
INSERT INTO DETALLE_REGLA_DOC(ID)
SELECT DOC_ID FROM DATOS_ORIGEN_CLIENTE DOC
```

En el código mostrado en la tabla XXXV se realiza la declaración de variables que se utilizarán en el proceso.

Tabla XXXV. Declaración de variables del proceso de agregación

```
'Declaración de variables
Dim strSQL As String
Dim rsRegistros As ADOR.Recordset
Dim rsDetalleRegla As ADOR.Recordset
Dim blnRegistros As Boolean
Dim blnDetalleRegla As Boolean
Dim intActual As Integer
Dim douMinimo As Double
Dim strNombreCampo As String
Dim ID As Long
```

En el código de la tabla XXXVI se inicializan las variables lógicas de validación de carga de datos blnRegistros y blnDetalleRegla con el valor falso; se cargan los registros a las variables rsRegistros y rsDetalleRegla en base a las consultas asignadas a la variable strSQL, por medio de la función ExecuteSQL, la cual hace una conexión a la base de datos para ejecutar la sentencia enviada como parámetro.

Tabla XXXVI. Código de consulta y asignación de registros

```
'Asignaciones iniciales
blnRegistros = False
blnDetalleRegla = False
douMinimo = 1

'Carga los registros de datos de cliente
strSQL = "SELECT * FROM DATOS_ORIGEN_CLIENTE"
Set rsRegistros = ExecuteSQL(strSQL)

'Carga los detalles de la regla 1
strSQL = " SELECT DET.*, DIM.NOMBRE + '_' + ATR.NOMBRE 'CAMPO' " & _
" FROM DETALLE_REGLA DET, " & _
" DETALLE_REGLA_ATRIBUTO DRA, " & _
" ATRIBUTO ATR, " & _
" DIMENSION DIM " & _
```

```

" WHERE DET.DETALLE_REGLA = DRA.DETALLE_REGLA " & _
" AND ATR.ATRIBUTO = DRA.ATRIBUTO " & _
" AND DIM.DIMENSION = ATR.DIMENSION " & _
" AND DET.REGLA = 1 " & _
" ORDER BY DET.ATRIBUTO "
Set rsDetalleRegla = ExecuteSQL(strSQL)

```

En el código mostrado en la tabla XXXVII se realiza la evaluación de validez de la carga de los registros.

Tabla XXXVII. Validación de carga de registros

```

'Evalua validez de los registros
If Not rsRegistros Is Nothing Then
    If Not rsRegistros.EOF Then
        blnRegistros = True
    End If
End If

If Not rsDetalleRegla Is Nothing Then
    If Not rsDetalleRegla.EOF Then
        blnDetalleRegla = True
    End If
End If

```

El siguiente paso se lleva a cabo si ambos conjuntos de datos fueron cargados correctamente, esto es, si las dos variables lógicas de la tabla XXXVII tienen un valor verdadero.

A continuación se explica el código mostrado en la tabla XXXVIII, el cual es el que realiza el llenado de la tabla temporal. Por cada registro de la tabla DATOS_ORIGEN_CLIENTE se recorren todos los detalles de regla; por cada detalle de regla se debe validar si la regla actual es igual a la evaluada en el ciclo anterior, de ser así se mide cuál es el resultado menor y se asigna a la variable douMinimo; si el registro actual no es igual al anterior, y no es el primero que se prueba, entonces se manda a actualizar la tabla con los últimos datos que tengan las variables douMinimo e intActual, y luego se asigna el siguiente registro a estas variables. Al terminar todos los detalles de reglas el

último queda sin procesar, es por ello que se hace una actualización con los valores resultado del último detalle regla al terminar el ciclo.

Tabla XXXVIII. Llenado de la tabla temporal

```
Do While Not rsRegistros.EOF
    intActual = 0
    rsDetalleRegla.MoveFirst
    Do While Not rsDetalleRegla.EOF
        strNombreCampo = rsDetalleRegla.Fields("CAMPO").Value
        If rsDetalleRegla.Fields("DETALLE_REGLA").Value = intActual
            If douMinimo > rsRegistros.Fields(strNombreCampo).Value
                douMinimo = rsRegistros.Fields(strNombreCampo).Value
            End If
        Else
            If intActual <> 0 'Actualiza en la tabla
                ID = rsRegistros.Fields("DOC_ID").Value
                Call Actualiza(ID, douMinimo, intActual)
            End If

            intActual = rsDetalleRegla.Fields("DETALLE_REGLA").Value
            douMinimo = rsRegistros.Fields(strNombreCampo).Value
        End If

        rsDetalleRegla.MoveNext
        'Actualiza en la tabla
        If rsDetalleRegla.EOF
            ID = rsRegistros.Fields("DOC_ID").Value
            Call Actualiza(ID, douMinimo, intActual)
        End If
    Loop
    rsRegistros.MoveNext
Loop
```

En la tabla XXXIX se muestra como queda la tabla DETALLE_REGLA_DOC, luego de haberse llevado a cabo el proceso de agregación.

Tabla XXXIX. Datos obtenidos con el proceso de agregación

ID	_1	_2	_3	_4	_5	_6	_7	_8	_9
1	0.50	0.50	0.00	0.00	0.00	0.00	0.00	0.00	0.00
2	0.50	0.50	0.00	0.00	0.00	0.00	0.00	0.00	0.00
3	0.40	0.40	0.00	0.42	0.58	0.00	0.00	0.00	0.00
4	0.00	0.00	0.00	0.33	0.60	0.00	0.33	0.40	0.00
5	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	0.00
6	0.50	0.42	0.00	0.50	0.42	0.00	0.00	0.00	0.00
7	0.00	0.78	0.00	0.00	0.22	0.00	0.00	0.00	0.00
8	0.00	0.70	0.13	0.00	0.30	0.13	0.00	0.00	0.00
9	0.00	0.00	0.00	0.00	0.20	0.00	0.00	0.80	0.00
10	0.33	0.67	0.00	0.00	0.00	0.00	0.00	0.00	0.00

La forma como debe interpretarse la tabla es la siguiente: el campo ID indica cuál es el registro de la tabla DATOS_ORIGEN_CLIENTE al cual se está haciendo referencia, y cada uno de los otros campos tienen el resultado de haber evaluado cada detalle de regla con los datos referenciados. Los valores resultado representan el grado de pertenencia que los datos evaluados tienen al atributo de salida que está asociado a cada detalle de regla.

4.3. Compensación

Este proceso es la evaluación alternativa de la agregación, y al igual que ésta se trabajará con la tabla temporal, y devolverá los grados de pertenencia que tienen los datos evaluados al atributo de salida de cada detalle de regla. Se hace la aclaración que cuando se hace agregación no se hace compensación, y

viceversa. Para efectos de ejemplo se mostrará como quedan los datos si se aplica el proceso de compensación con un grado de compensación arbitrario.

En las siguientes tablas se muestra el código a utilizar para llevar a cabo el proceso de compensación, acompañadas de la explicación de cada una. En el código mostrado en la tabla XL se realiza la declaración de variables que se utilizarán en el proceso de compensación.

Las variables a utilizar son las mismas, pero además se utilizará una variable donde se guardarán los grados de pertenencia que serán evaluados por la función de compensación y otra variable que guardará el grado de compensación, el cual debe ser obtenido de la tabla REGLA.

Tabla XL. Declaración de variables del proceso de compensación

```
'Declaración de variables
Dim strSQL As String
Dim rsRegistros As ADOR.Recordset
Dim rsDetalleRegla As ADOR.Recordset
Dim blnRegistros As Boolean
Dim blnDetalleRegla As Boolean
Dim intActual As Integer
Dim douMinimo As Double
Dim strNombreCampo As String
Dim ID As Long
Dim strGradosPert As String
Dim douGradoComp As Double
```

El código mostrado en la tabla XLI hace exactamente lo mismo que el de la tabla XXXVI, asigna valores iniciales a las variables lógicas que indicarán si los registros han sido cargados correctamente y cargan los registros en base a

las consultas definidas en la variable strSQL. Además de asignar un valor a la variable grado de compensación, para efectos del ejemplo se utiliza el valor 0.3, pero este valor debe obtenerse del campo GRADO_DE_COMPENSACION, de la tabla REGLA.

Tabla XLI. Código de consulta de información y asignación de registros

```
'Asignaciones iniciales
  blnRegistros = False
  blnDetalleRegla = False
  douGradoComp = 0.3

'Carga los registros de datos de cliente
  strSQL = "SELECT * FROM DATOS_ORIGEN_CLIENTE"
  Set rsRegistros = ExecuteSQL(strSQL)

'Carga los detalles de la regla 1
  strSQL = " SELECT DET.*, DIM.NOMBRE + '_' + ATR.NOMBRE 'CAMPO' " & _
          " FROM DETALLE_REGLA DET, " & _
          " DETALLE_REGLA_ATRIBUTO DRA, " & _
          " ATRIBUTO ATR, " & _
          " DIMENSION DIM " & _
          " WHERE DET.DETALLE_REGLA = DRA.DETALLE_REGLA " & _
          " AND ATR.ATRIBUTO = DRA.ATRIBUTO " & _
          " AND DIM.DIMENSION = ATR.DIMENSION " & _
          " AND DET.REGLA = 1 " & _
          " ORDER BY DET.ATRIBUTO "
  Set rsDetalleRegla = ExecuteSQL(strSQL)
```

En el código mostrado en la tabla XLII se realiza la evaluación de validez de la carga de los registros.

Tabla XLII. Validación de carga de registros

```
'Evalua validez de los registros
  If Not rsRegistros Is Nothing Then
    If Not rsRegistros.EOF Then
      blnRegistros = True
    End If
  End If
```

```
If Not rsDetalleRegla Is Nothing Then
    If Not rsDetalleRegla.EOF Then
        blnDetalleRegla = True
    End If
End If
```

Si los dos conjuntos de registros fueron cargados correctamente, se lleva a cabo el siguiente paso, que es donde en realidad, se hace el proceso.

Por cada registro de la tabla DATOS_ORIGEN_CLIENTE se recorren todos los detalles de regla; por cada detalle de regla se debe validar si la regla actual es igual a la evaluada en el ciclo anterior, de ser así, se asigna ese grado de pertenencia a la variable strGradoPert, y se le agrega una coma; si el registro actual no es igual al anterior, y no es el primero que se prueba, entonces la variable douMinimo recibe el valor de la función de compensación cuando se le envían los parámetros strGradoPert y douGradoComp, luego se manda a actualizar la tabla con los valores de las variables douMinimo e intActual.

A continuación, en la tabla XLIII, se muestra el código explicado arriba.

Tabla XLIII. Código del proceso de compensación

```
Do While Not rsRegistros.EOF
    intActual = 0
    rsDetalleRegla.MoveFirst
    strGradosPert = ""
    Do While Not rsDetalleRegla.EOF
        strNombreCampo = rsDetalleRegla.Fields("CAMPO").Value
        If rsDetalleRegla.Fields("DETALLE_REGLA").Value = intActual
            strGradosPert = strGradosPert &
                CStr(rsRegistros.Fields(strNombreCampo).Value) & ", "
        Else
            If intActual <> 0 Then 'Actualiza en la tabla
                ID = rsRegistros.Fields("DOC_ID").Value
                douMinimo = FunComp(strGradosPert, douGradoComp)
                Call Actualiza(ID, douMinimo, intActual)
                strGradosPert = ""
            End If

            intActual = rsDetalleRegla.Fields("DETALLE_REGLA").Value
            strGradosPert = strGradosPert &
                CStr(rsRegistros.Fields(strNombreCampo).Value) & ", "
        End If

        rsDetalleRegla.MoveNext
        'Actualiza en la tabla
        If rsDetalleRegla.EOF Then
            ID = rsRegistros.Fields("DOC_ID").Value
            douMinimo = FunComp(strGradosPert, douGradoComp)
            Call Actualiza(ID, douMinimo, intActual)
        End If
    Loop
    rsRegistros.MoveNext
Loop
```

La función de compensación está basada en la fórmula mostrada en la sección 1.1.1.10, y su estructura se muestra en la tabla XLIV. La función recibe dos parámetros, uno es una cadena de grados de pertenencia separados por

coma, y el otro es el grado de compensación. Luego de procesar los datos lo que devuelve es un grado de pertenencia, es decir, un valor entre 0 y 1.

Tabla XLIV. Código de la función de compensación

```
Public Function FunComp(ByVal strGradosPert As String, _
                        ByVal douGrComp As Double) As Double
'Declaración de variables
    Dim douResultado As Double
    Dim arrGradosPert As Variant
    Dim intCuenta As Integer
    Dim douProducto1 As Double
    Dim douProducto2 As Double

'Seperación de grados de pertenencia
    arrGradosPert = Split(strGradosPert, ",")

    douProducto1 = 1
    For intCuenta = LBound(arrGradosPert) To UBound(arrGradosPert)
        douProducto1 = douProducto1 * CDb1(arrGradosPert(intCuenta))
    Next

    douResultado = Potencia(douProducto1, 1 - douGrComp)

    douProducto2 = 1
    For intCuenta = LBound(arrGradosPert) To UBound(arrGradosPert)
        douProducto2 = douProducto2 * (1 - arrGradosPert(intCuenta))
    Next

    douResultado = douResultado * Potencia(1 - douProducto2, douGrComp)

    FunComp = douResultado
End Function
```

En la tabla XLV se muestra como queda la tabla DETALLE_REGLA_DOC, luego de haberse llevado a cabo el proceso de compensación, si se usa un grado de compensación de 0.3.

Tabla XLV. Datos obtenidos con el proceso de compensación

ID	_1	_2	_3	_4	_5	_6	_7	_8	_9
1	0.62	0.62	0.00	0.00	0.00	0.00	0.00	0.00	0.00
2	0.62	0.62	0.00	0.00	0.00	0.00	0.00	0.00	0.00
3	0.25	0.33	0.00	0.35	0.45	0.00	0.00	0.00	0.00
4	0.00	0.00	0.00	0.29	0.51	0.00	0.21	0.37	0.00
5	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	0.00
6	0.39	0.30	0.00	0.39	0.30	0.00	0.00	0.00	0.00
7	0.00	0.84	0.00	0.00	0.35	0.00	0.00	0.00	0.00
8	0.00	0.70	0.17	0.00	0.38	0.08	0.00	0.00	0.00
9	0.00	0.00	0.00	0.00	0.32	0.00	0.00	0.86	0.00
10	0.46	0.76	0.00	0.00	0.00	0.00	0.00	0.00	0.00

La forma como debe interpretarse la tabla es la misma como cuando se hizo el proceso de agregación, o sea, los valores obtenidos son grados de pertenencia de los registros evaluados a los atributos de salida asociados a cada regla. Ya sea que se haga la agregación o la compensación, después debe realizarse la composición en base a los datos de la tabla DETALLE_REGLA_DOC.

4.4. Composición

Este es el proceso mediante el cual se determina cuál es el grado de pertenencia que tienen los datos evaluados, a los atributos de la dimensión de

salida de la regla evaluada. El proceso de composición consiste en evaluar mediante el operador PRODUCTO el resultado de cada uno de los detalles de regla, y luego entre los que tienen el mismo atributo de salida aplicar el operador MAXIMO, esto es, tomar el grado de pertenencia del detalle de regla más grande.

Previo a realizar el proceso de composición se debe preparar la tabla DATOS_ORIGEN_CLIENTE, en la cual se agregarán los campos correspondientes a los atributos de la dimensión de salida de la regla evaluada. El código para agregar estos campos se muestra en la tabla XLVI, y en la tabla XLVII se muestra la estructura de la tabla con los nuevos campos.

Tabla XLVI. Código para agregar los campos para el proceso de composición

```
ALTER TABLE DATOS_ORIGEN_CLIENTE
ADD DESCUENTO_REGULAR decimal(5,2) null

ALTER TABLE DATOS_ORIGEN_CLIENTE
ADD DESCUENTO_MUCHO decimal(5,2) null

ALTER TABLE DATOS_ORIGEN_CLIENTE
ADD DESCUENTO_POCO decimal(5,2) null
```

Tabla XLVII. Estructura de la tabla luego de agregar los campos para la composición

DOC_ID	SUELDO	EDAD	DESCUENTO_ REGULAR	DESCUENTO_ MUCHO	DESCUENTO_ POCO
1	3500	24			
2	3500	24			
3	8000	25			
4	12000	26			
5	30000	32			

6	7500	23			
7	6100	36			
8	6500	42			
9	14000	30			
10	1200	26			

En la tabla XLVIII se muestra la declaración de las variables a utilizar en este proceso, mientras que en la XLIX se muestran las asignaciones iniciales a las variables lógicas y de conteo, así como la carga de los datos de la tabla temporal para obtener los grados de pertenencia determinados por medio de la agregación o de la compensación, según sea el caso, y la carga de los datos de los detalles de regla con su respectivo atributo de salida y grado de soporte.

Tabla XLVIII. Declaración de variables del proceso de composición

```
'Declaración de variables
Dim strSQL As String
Dim rsTemporal As ADOR.Recordset
Dim rsDetalleRegla As ADOR.Recordset
Dim blnTemporal As Boolean
Dim blnDetalleRegla As Boolean
Dim intActual As Integer
Dim douMaximo As Double
Dim strNombreCampo As String
Dim ID As Long
Dim strCampoActualizar As String
```

Tabla XLIX. Código de consulta de información y asignación de registros

```
'Asignaciones iniciales
  blnTemporal = False
  blnDetalleRegla = False
  douMaximo = 0

'Carga los registros de la tabla temporal
  strSQL = "SELECT * FROM DETALLE_REGLA_DOC"
  Set rsTemporal = ExecuteSQL(strSQL)

'Carga los atributos de salida de la regla 1
  strSQL = " SELECT DET.ATRIBUTO, DET.DETALLE_REGLA, " & _
          "          DET.GRADO_DE_SOPORTE, " & _
          "          DIM.NOMBRE +'_' +ATR.NOMBRE 'ATRIBUTO_NOM' " & _
          " FROM    ATRIBUTO ATR, " & _
          "          DIMENSION DIM, " & _
          "          DETALLE_REGLA DET " & _
          " WHERE   DIM.DIMENSION = ATR.DIMENSION " & _
          " AND     DET.ATRIBUTO = ATR.ATRIBUTO " & _
          " AND     DET.REGLA = 1 " & _
          " ORDER BY DET.ATRIBUTO "
  Set rsDetalleRegla = ExecuteSQL(strSQL)
```

Al igual que en los dos procesos anteriores, se debe validar que la carga de los registros haya sido exitosa, esta validación se realiza con el código mostrado en la tabla L.

Tabla L. Validación de carga de registros

```
'Evalua validez de los registros
  If Not rsTemporal Is Nothing Then
    If Not rsTemporal.EOF Then
      blnTemporal = True
```



```

        End If
    End If

    If Not rsDetalleRegla Is Nothing Then
        If Not rsDetalleRegla.EOF Then
            blnDetalleRegla = True
        End If
    End If

```

Si ambas variables indican que los registros fueron cargados satisfactoriamente se procede a ejecutar el código mostrado en la tabla LI, el cual se explica a continuación.

Tabla LI. Código del proceso de composición

```

Do While Not rsTemporal.EOF
    intActual = 0
    rsDetalleRegla.MoveFirst
    Do While Not rsDetalleRegla.EOF
        strNombreCampo = "_" & _
            rsDetalleRegla.Fields("DETALLE_REGLA").Value
        If rsDetalleRegla.Fields("ATRIBUTO").Value = intActual Then
            'es del mismo atributo salida
            If douMaximo <
                rsTemporal.Fields(strNombreCampo).Value *
                rsDetalleRegla.Fields("GRADO_DE_SOPORTE").Value

                douMaximo = rsTemporal.Fields(strNombreCampo).Value *
                    rsDetalleRegla.Fields("GRADO_DE_SOPORTE").Value
            End If
        Else
            If intActual <> 0 Then 'Actualiza en la tabla
                ID = rsTemporal.Fields("ID").Value
                Call ActualizaDOC(ID, douMaximo, strCampoActualizar)
            End If

            intActual = rsDetalleRegla.Fields("ATRIBUTO").Value
            douMaximo = rsTemporal.Fields(strNombreCampo).Value *
                rsDetalleRegla.Fields("GRADO_DE_SOPORTE").Value
            strCampoActualizar =
                rsDetalleRegla.Fields("ATRIBUTO_NOM").Value
        End If

        rsDetalleRegla.MoveNext
        'Actualiza en la tabla
        If rsDetalleRegla.EOF Then

```

```

        ID = rsTemporal.Fields("ID").Value
        Call ActualizaDOC(ID, douMaximo, strCampoActualizar)
    End If
Loop
rsTemporal.MoveNext
Loop

```

Para cada registro ingresado en la tabla temporal, se evalúa por cada detalle de regla cuál de los valores de la tabla temporal, con el mismo atributo de salida, tiene el mayor valor luego de aplicarle el grado de soporte. El grado de pertenencia que tenga el mayor valor será insertado en la tabla DATOS_ORIGEN_CLIENTE en la columna que corresponde al atributo evaluado, en el registro correspondiente al de la tabla temporal. En la tabla LII se muestra parte del contenido de la tabla DATOS_ORIGEN_CLIENTE luego de haber realizado el proceso de composición, con lo cual se obtiene el conjunto difuso de salida correspondiente a la dimensión descuento.

Tabla LII. Datos obtenidos con el proceso de composición

DOC_ID	SUELDO	EDAD	DESCUENTO_ REGULAR	DESCUENTO_ MUCHO	DESCUENTO_ POCO
1	3500	24	0.45	0.50	0.00
2	3500	24	0.45	0.50	0.00
3	8000	25	0.58	0.40	0.00
4	12000	26	0.60	0.36	0.00
5	30000	32	0.00	0.90	0.00
6	7500	23	0.50	0.50	0.00
7	6100	36	0.70	0.00	0.00
8	6500	42	0.63	0.00	0.13
9	14000	30	0.20	0.72	0.00
10	1200	26	0.60	0.33	0.00

5. ANÁLISIS Y PRESENTACIÓN DE RESULTADOS

5.1. Conversión de conjuntos difusos a valores numéricos

El proceso de conversión de los conjuntos difusos de salida a valores numéricos es la parte final del proceso difuso, ya que con este obtenemos el valor numérico único comparable con el valor de aceptación definido en cada regla. Para realizar el proceso de conversión se requiere que la tabla DATOS_ORIGEN_CLIENTE tenga una columna donde recibirá los valores numéricos, resultado de la conversión.

Luego se necesita saber cuál es el valor de verdad más significativo para cada uno de los atributos de la dimensión de salida de la regla evaluada. Estos valores se deberán evaluar junto con los grados de pertenencia de cada uno de los registros de la tabla DATOS_ORIGEN_CLIENTE, por medio del método de conversión conocido como centro de gravedad.

El código para agregar la columna resultado, obtener los valores de verdad de cada atributo de salida y la inserción de valores en la tabla DATOS_ORIGEN_CLIENTE es mostrado en la tabla LIII. En la tabla LIV se muestra el código identificador de cada registro y el valor resultado del proceso difuso.

Tabla LIII. Actualización de valores resultado

```
--Agrega columna de resultado a la tabla DATOS_ORIGEN_CLIENTE
ALTER TABLE DATOS_ORIGEN_CLIENTE
ADD VALOR decimal(5,2) null

--CONSULTA VALORES DE VERDAD DE CADA ATRIBUTO DE LA DIMENSION DE SALIDA
SELECT VALOR_DE_VERDAD, ATRIBUTO FROM ATRIBUTO WHERE DIMENSION = 5

/* REGULAR=50 MUCHO=75 POCO=25 */

--ACTUALIZA LA TABLA
UPDATE DATOS_ORIGEN_CLIENTE
SET VALOR = (DESCUENTO_REGULAR*50+DESCUENTO_MUCHO*75+DESCUENTO_POCO*25)
            / (DESCUENTO_REGULAR+DESCUENTO_MUCHO+DESCUENTO_POCO)
```

Tabla LIV. Datos obtenidos con la actualización

DOC_ID	VALOR
1	63.16
2	63.16
3	60.20
4	59.38
5	75.00
6	62.50
7	50.00
8	45.72
9	69.57
10	58.87

5.2. Interpretación de resultados

Luego de haber terminado con el proceso de conversión de valores ya se pueden interpretar los resultados, comparando cada uno de ellos con el valor de aceptación definido en la regla.

Para los datos utilizados en el ejemplo, el valor de aceptación definido en la regla 1 es de 65, es decir, que el descuento se les dará a aquellos clientes cuyo valor de la variable Descuento es mayor o igual a 65.

En la tabla LV, se muestra el código para determinar si el cliente es aceptado o rechazado en la propuesta de descuentos, basada en su sueldo y su edad. El resultado de esta consulta se muestra en la tabla LVI.

Tabla LV. Código para evaluar la aceptación

```
SELECT DOC_ID, VALOR,
       CASE
         WHEN VALOR >= 65 THEN 'ACEPTADO'
         WHEN VALOR < 65 THEN 'RECHAZADO'
       END 'RESULTADO'
FROM DATOS_ORIGEN_CLIENTE
```

Tabla LVI. Resultado de la evaluación de aceptación

DOC_ID	VALOR	RESULTADO
1	63.16	RECHAZADO
2	63.16	RECHAZADO
3	60.20	RECHAZADO
4	59.38	RECHAZADO
5	75.00	ACEPTADO
6	62.50	RECHAZADO
7	50.00	RECHAZADO
8	45.72	RECHAZADO
9	69.57	ACEPTADO
10	58.87	RECHAZADO

Como se puede verificar los únicos seleccionados a recibir el descuento son los clientes 5 y 9, de un total de 10 clientes evaluados, debido a que el resultado de su proceso difuso fue mayor de 65.

6. CONSIDERACIONES PARA LA IMPLEMENTACIÓN

6.1. Sistemas sugeridos

Para la implementación de este sistema de apoyo en la toma de decisiones, existen varias opciones, en las distintas capas de desarrollo. A continuación se mencionan algunas propuestas sobre los sistemas sobre los que se puede llevar a cabo la implementación.

- | | | |
|----|-----------------------------|-------------------------------------|
| 1. | Sistema operativo: | Microsoft Windows |
| | Manejador de base de datos: | Microsoft SQL Server 2000 |
| | Entorno de programación: | Microsoft Visual Basic 6.0 |
| 2. | Sistema operativo: | Microsoft Windows |
| | Manejador de base de datos: | Microsoft SQL Server 2000 |
| | Entorno de programación: | Procedimientos almacenados
MSSQL |
| 3. | Sistema operativo: | Linux |
| | Manejador de base de datos: | MySQL |
| | Entorno de programación: | Java 2 |

6.2. Relación con otros sistemas

El sistema de apoyo en la toma de decisiones, tendrá relación con los sistemas donde se almacena la información sobre los clientes, productos y promociones, ya que éstos son los que lo proveerán de la información necesaria para realizar su tarea.

Las formas en que se relacionará, pueden ser por levantado de datos por medio de archivos, o bien relación directa con las tablas del sistema.

6.3. Interfaz de entrada

Para dar al usuario la confianza en el uso del sistema, lo primero debe ser que se sienta a gusto con él, por lo que se debe proveer de una interfaz amigable y de fácil uso. Si el usuario puede configurar fácilmente el manejo de reglas y dimensiones, podrá entender de mejor manera los resultados devueltos por el sistema.

Las entradas que el usuario tendrá a su disposición y que deben ser prácticas para él son:

1. Definición de dimensiones
2. Configuración de obtención de datos
3. Definición de atributos por dimensión
4. Definición de reglas
5. Configuración de las instancias de las reglas
6. Configuración de los atributos de salida

En las figuras 8, 9 y 10 se muestran algunos prototipos de ejemplo para las entradas al sistema.

Figura 8. Definición de dimensiones

Definición de dimensiones

Nombre: Sueldo

Unidad de medida: quetzales

Origen de datos: SELECT CLI.SUELDO
FROM CLIENTE CLI
WHERE

Validar origen

Aceptar Cancelar

Figura 9. Definición de atributos por dimensión

Definición de atributos por dimensión

Dimensión: Sueldo

Unidad de medida: Quetzales

Nombre: Medio

Valor de verdad: 10000 Valor de verdad final:

Valor de verdad izquierdo: 5000 Valor de verdad derecho: 15000

Tipo de función: Pi
L (ele)
Gamma
Lambda

Aceptar Cancelar

Figura 10. Definición de reglas

The image shows a software dialog box titled "Definición de reglas". It is divided into two main sections. On the left, under the heading "Dimensiones de entrada", there is a button labeled "Agregar dimensión". Below this, there are two dropdown menus: "Dimensión 1" with the value "Sueldo" and "Dimensión 2" with the value "Edad". On the right side of the dialog, there is a dropdown menu labeled "Dimensión de salida" with the value "Descuento". At the bottom right, there are two buttons: "Aceptar" and "Cancelar".

6.4. Interfaz de salida

Finalmente, se debe proveer al usuario de los datos resultado, ya que éstos son los que le interesan, por lo que los mismos deben presentarse de la manera más fácil de entender, dado que el usuario está usando el sistema para apoyarse, no para tardarse más que si no lo tuviera.

Las salidas que el sistema puede tener son:

1. Datos resultado a pantalla
2. Datos resultado a archivo
3. Detalle del proceso difuso a pantalla
4. Detalle del proceso difuso a archivo
5. Resultados a base de datos de otro sistema

6.5. Consideraciones finales

El diseño del sistema permite obtener datos de cualquier base de datos por medio de consultas a la base de datos de clientes, las mismas deberán construirse de forma dinámica, para poder dar flexibilidad en cuanto a la cantidad de variables que se deseen manejar en tiempo de ejecución.

Al momento de configurar el sistema se deberán definir todos los componentes difusos que éste usará, los resultados que del procesamiento de los mismos sean obtenidos será el reflejo de la lógica usada por la persona que lo configure, por lo que se recomienda que la persona encargada de configurar el sistema tenga conocimientos amplios sobre los criterios que utilice para determinar las reglas y condiciones.

CONCLUSIONES

1. A través de la segmentación de clientes y las distintas formas de clasificación, se pueden analizar y entender de mejor manera las preferencias del cliente, y con ello saber cómo satisfacerlo.
2. Los resultados de las reglas que se modelen dependerán de cómo esté estructurada la regla y cuáles sean sus valores de entrada.
3. Los resultados obtenidos según el presente diseño, podrán interpretarse tanto como valores puntuales en forma de porcentaje, así como valores de referencia para un rango determinado.
4. Las técnicas de inteligencia artificial, facilitan el manejo de información que sería más difícil trabajarla con lenguajes estructurados.

RECOMENDACIONES

1. Que las empresas enfoquen su *marketing* hacia el cliente, para comenzar a cambiar la cultura organizacional hacia el enfoque al cliente, de forma que cuando ésta se haya cambiado pueda pensarse en la implementación de la metodología CRM.
2. Tomar el presente trabajo como base para utilizar la lógica difusa en otras aplicaciones.
3. El presente trabajo puede ser usado como introducción de la metodología CRM, y algunas clasificaciones de clientes.

BIBLIOGRAFÍA

TEXTOS

1. Curry, Jay y Adam Curry. **CRM Cómo implementar y beneficiarse de la gestión de las relaciones con los clientes (Customer Relationship Management)**.tr.Ganzinelli, Carlos. España: Ediciones Gestión2000 S.A., 2002.280pp
2. Py, Pascal. **Métodos y técnicas para gestionar a los clientes**.tr.Gil, Esther.España: Éditions d'Organisation, 2002. 184pp

TESIS

3. Domínguez Alvarado, Elizabeth. Aplicación de la lógica fuzzy a los modelos económico-financieros.Tesis Ing. En Ciencias y Sistemas. Guatemala.Universidad de San Carlos de Guatemala, Facultad de Ingeniería.2001.208pp
4. Jolón Reyes, Santos Giovanni. Aplicación de redes neuronales en la predicción de la respuesta de los clientes de una institución bancaria ante un plan de *marketing*. Tesis Ing. en sistemas. Guatemala. Universidad Rafael Landivar, Facultad de Ingeniería.2004.

REFERENCIAS ELECTRÓNICAS

5. Área de marketing de Improven-Consultores. ¿QUÉ ES CRM?.
<http://www.gestiopolis.com/canales/demarketing/articulos/37/crmcliente.htm>. (13-Ago-2004)

6. Castillo Tapia, Guadalupe y A. Coello Coello. **Uso de Técnicas de Inteligencia Artificial para Aplicaciones Financieras.**
<http://www.lania.mx/biblioteca/newsletters/1998-otono-invierno/tecnicas.html>. (2-Ago-2004)
7. Corzo, Yuliana. **LÓGICA DIFUSA.**
<http://personales.ya.com/casanchi/mat/difusa01.htm>.(5-Ago-2004)
8. Gajardo Ugas, Sergio. **¿QUÉ ES CRM Y CUÁL ES EL VERDADERO SIGNIFICADO?.**
<http://www.gestiopolis.com/canales/demarketing/articulos/43/crmmba.htm>.(15-Ago-2004)
9. Microsoft. **EI CRM como estrategia de negocio.**
http://www.microsoft.com/spain/empresas/soluciones/20040115_guia_crm.aspx. (11-Ago-2004)
10. **A Comparison of the Interpretation Methods for Fuzzy Inference.**
School of Artificial Intelligence Division of Informatics, University of Edinburgh. Lucas Dixon.