UNIVERSIDAD NACIONAL AGRARIA
LA MOLINA

FACULTAD DE ECONOMÍA Y PLANIFICACIÓN

"SELECCIÓN DE ATRIBUTOS POR MÉTODOS DE FILTRADO Y WRAPPER PARA PREDECIR LA SATISFACCIÓN DE USUARIOS DE SALUD"

TESIS PARA OPTAR TÍTULO DE
INGENIERO EN ESTADÍSTICA E INFORMÁTICA

EDGAR FERNANDO ANCAJIMA BOHÓRQUEZ

Lima – Perú

2022

Ouriginal by Turnitin

Document Information

Analyzed document	Ancajima_Tesis_2022_V03.docx (D147162467)
Submitted	10/21/2022 1:16:00 AM
Submitted by	César Higinio Menacho Chiok
Submitter email	cmenacho@lamolina.edu.pe
Similarity	10%
Analysis address	cmenacho.unalm@analysis.urkund.com

Sources included in the report

SA **Universidad Nacional Agraria La Molina / Trabajo TSP_VF.docx**
Document Trabajo TSP_VF.docx (D142755749)
Submitted by: cmenacho@lamolina.edu.pe
Receiver: cmenacho.unalm@analysis.urkund.com

34

Entire Document

UNIVERSIDAD NACIONAL AGRARIA LA MOLINA FACULTAD DE ECONOMÍA Y PLANIFICACIÓN “MÉTODOS DE FILTRADO PARA LA SELECCIÓN DE ATRIBUTOS EN EL APRENDIZAJE SUPERVISADO”
Presentada por: Edgar Fernando Ancajima Bohórquez
TESIS PARA OPTAR EL TÍTULO DE INGENIERO EN ESTADÍSTICA E INFORMÁTICA
Dr. César
Menacho
ChiokLima –
Perú 2022
UNIVERSIDAD NACIONAL AGRARIA LA MOLINA FACULTAD DE ECONOMÍA Y PLANIFICACIÓN “MÉTODOS DE FILTRADO PARA LA SELECCIÓN DE ATRIBUTOS EN EL APRENDIZAJE SUPERVISADO”
Presentada por: Edgar Fernando Ancajima Bohórquez
TESIS PARA OPTAR EL TÍTULO DE INGENIERO EN ESTADÍSTICA E INFORMÁTICA
_____ _____ Mg.Sc. Clodomiro Fernando Miranda Villagómez Dr. César Higinio Menacho Chiok PRESIDENTE PATROCINADOR
_____ _____ Ph.D Frida Rosa Coaquira Nina Mg.Sc. Ana Cecilia Vargas Paredes MIEMBRO MIEMBRO

UNIVERSIDAD NACIONAL AGRARIA
LA MOLINA

FACULTAD DE ECONOMÍA Y PLANIFICACIÓN

"SELECCIÓN DE ATRIBUTOS POR MÉTODOS DE FILTRADO Y WRAPPER PARA PREDECIR LA SATISFACCIÓN DE USUARIOS DE SALUD"

PRESENTADO POR

EDGAR FERNANDO ANCAJIMA BOHÓRQUEZ

TESIS PARA OPTAR TÍTULO DE
INGENIERO EN ESTADÍSTICA E INFORMÁTICA

SUSTENTADA Y APROBADA ANTE EL SIGUIENTE JURADO

Mg.Sc. Clodomiro Fernando Miranda Villagómez	Dr. César Higinio Menacho Chiok
PRESIDENTE	**PATROCINADOR**
Ph.D Frida Rosa Coaquira Nina	Mg.Sc. Ana Cecilia Vargas Paredes
MIEMBRO	**MIEMBRO**

Lima – Perú

2022

DEDICATORIA

La presente Tesis está dedicada a toda mi familia, principalmente a mis padres Edgar y Cleny quienes son mi motivación para crecer como persona y profesional, de ellos aprendí la perseverancia para cumplir mis metas.

Y a mi hermana Claudia quien me motivó e inspiró a cerrar este primer paso y objetivo.

AGRADECIMIENTO

A mi profesor,

Dr. César Menacho, por asesorarme y motivarme durante la elaboración del presente trabajo.

ÍNDICE GENERAL

ÍNDICE DE TABLAS

Pag.

ÍNDICE DE FIGURAS

Pag.

RESUMEN

Las técnicas de minería de datos (TMD) usadas para el aprendizaje supervisado, generalmente deben considerar un gran número de atributos en las bases de datos a ser analizadas, y muchos de estos atributos son irrelevantes y redundantes que pueden distorsionar el rendimiento y la funcionalidad de estas técnicas, y por lo tanto su capacidad predictiva. Las investigaciones sobre el tema de la selección de atributos, mencionan que, al seleccionar un número menor de atributos del conjunto total, puede traer una serie de ventajas: reducir la redundancia, eliminar el ruido, maximizar la relevancia de los atributos, disminuir costo computacional, aumentar la interpretación y mejorar la precisión del clasificador de aprendizaje supervisado. El objetivo es presentar los métodos de selección de atributos por filtrado y Wrapper que pueden ser aplicadas en las técnicas de minería de datos supervisadas para la tarea de clasificación, consiguiendo los mejores subconjuntos de atributos relevantes con las mayores tasas de precisión. Se aplican cuatro métricas para seleccionar los atributos por filtrado (Chi-Cuadrado, Ganancia de información, Razón de ganancia y Relief) y cuatro métodos por Wrapper (Best-First, Greedy forward, Greedy backward y Hill climbing) en la Encuesta Nacional de Satisfacción de Usuarios de Salud–2015. Los resultados aplicando cuatro TMD a cada uno de los diferentes subconjuntos de atributos seleccionados con los métodos de por filtrado y wrapper, mostraron con las mayores capacidades predictivas para predecir la satisfacción de los usuarios de la atención recibida de los servicios de salud, en el caso de la regresión logística binaria el método wrapper Best-First con 5 atributos y una precisión del 88,7%, el árbol de clasificación C5.0 con wrapper Greedy forward con 6 atributos y una precisión del 89,1%, la redes bayesianas Naive con wrapper Greedy backward con 16 atributos y una precisión del 88,3% y el multiclasificador random Forest con wrapper Greedy backard con 16 atributos y una precisión del 93,0%. Los mayores AUC para la regresión logística binaria fue con el método Greedy forward con 0,932, el árbol de clasificación C5.0 con Greedy forward con 0,891, la rede bayesianas Naive con wrapper Greedy forward con 0,9221 y el multiclasificador random Forest con Greedy backard con 0,941.

Palabras clave. Aprendizaje supervisado, selección de atributos, métodos de filtrado, métodos wrapper, capacidad predictiva.

ABSTRACT

Data mining techniques (DMT) used for supervised learning generally must consider the large number of attributes in the databases to be analyzed, and many of these attributes are irrelevant and redundant that can distort performance and functionality of these techniques, and therefore their predictive capacity. Research on the subject of feature selection mentions that by selecting a smaller number of features from the total set, it can bring a series of advantages: reduce redundancy, eliminate noise, maximize the relevance of features, reduce computational cost, increase the interpretation and improve the accuracy of the supervised learning classifier. The objective is to present the filtering and Wrapper attribute selection methods that can be applied in supervised data mining techniques for the classification task, obtaining the best subsets of relevant attributes with the highest accuracy rates. Four metrics are applied to select the attributes by filtering (Chi-Square, Information gain, Gain ratio and Relief) and four methods by Wrapper (Best-First, Greedy forward, Greedy backward and Hill climbing) in the National Satisfaction Survey of Health Users–2015. The results applying four TMD to each of the different subsets of attributes selected with the methods of filtering and wrapper, showed the greatest predictive capacities to predict the satisfaction of the users of the attention received from the health services, in the case of binary logistic regression, the Best-First wrapper method with 5 attributes and an accuracy of 88,7%, the C5.0 classification tree with Greedy forward wrapper with 6 attributes and an accuracy of 89,1%, the Naive Bayesian network with Greedy backward wrapper with 16 attributes and an accuracy of 88,3% and the random Forest multiclassifier with Greedy backard wrapper with 16 attributes and an accuracy of 93,0%. The highest AUC for binary logistic regression was with the Greedy forward method with 0,932, the C5.0 classification tree with Greedy forward with 0,891, the Naive Bayesian network with Greedy forward wrapper with 0,9221 and the random Forest multiclassifier with Greedy backard. with 0,941.

Keywords. Supervised learning, attribute selection, filter methods, wrapper methods, predictive capacity.

I. INTRODUCCIÓN

En las últimas décadas, a nivel mundial las organizaciones públicas de los diferentes sectores y empresas privadas de las diferentes áreas de servicios, comercialización o manufactura están aplicando las técnicas de minería de datos para el tratamiento, procesamiento y análisis a los grandes volúmenes de datos que se generan como producto de sus actividades o transacciones diarias. La Técnicas de Minería de Datos (TMD), es una metodología que aplica técnicas estadísticas, algoritmos de inteligencia artificial y aprendizaje de máquinas con la finalidad de extraer conocimientos relevantes de las bases de datos a partir de la búsqueda de patrones y tendencias que permitan apoyar la toma de decisiones de estas organizaciones. Existen una gran variedad de TMD clasificadas como supervisadas, porque se aplican cuando en el conjunto de datos existe una variable dependiente (atributo clase) que puede ser de naturaleza cualitativa (problema de clasificación) o cuantitativa (problema de predicción). Las TMD cuentan con el desarrollo de algoritmos y herramientas muy especializadas que son capaces de descubrir automáticamente conocimientos ocultos de las bases de datos, sin embargo, hay aspectos que tienen que ver con la eficiencia, la calidad y la precisión de los resultados obtenidos con estas técnicas. Uno de los problemas más comunes que debe enfrentar las TMD supervisadas, es el gran número de atributos que conforman las bases de datos. En muchos de los casos, gran parte de estos atributos son irrelevantes y redundantes, y que pueden distorsionar la funcionalidad y la calidad de dichas técnicas y por lo tanto su capacidad predictiva. Para enfrentar este problema se están llevando a cabo muchos estudios bajo la denominación de selección de atributos (Feature Selection) o selección de características, siendo actualmente un campo de gran investigación. Según Kumar et al. (2014) la selección de atributos, es un proceso de encontrar un subconjunto óptimo de atributos “x” desde un conjunto total de “X” para lo cual se requiere una larga búsqueda de subconjuntos de atributos, siendo el subconjunto óptimo de atributos evaluados por un criterio de medida.
Según Yuanhong et al. (2007) el objetivo principal de la selección de atributos es el reducir su número, removiendo los irrelevantes, redundantes y causantes de ruido; lo cual permite la reducción de la complejidad y el sobre ajuste de los métodos de aprendizaje supervisado e incrementando la velocidad computacional y teniendo como finalidad mejorar la precisión de la clasificación con la técnica de minería de datos supervisada.

Según Mani et al. (2016) la selección de atributos, es considerada como una fase esencial del pre procesamiento de datos, pudiendo mejorar la precisión, la calidad de los datos y

comprensión de los resultados obtenidos con las TMD para el aprendizaje supervisado. Así como también sirve para reducir la dimensionalidad y reducir los costos computacionales.
Kalousis et al. (2007) mencionan que la selección de atributos se ha convertido en una etapa muy importante en el proceso del aprendizaje automático. Mientras que Molina et al. (2002) mencionan que el proceso de selección de atributos, tienen un efecto inmediato en la aplicación de los algoritmos de minería de datos al mejorar la calidad de los datos y su rendimiento, redundando en el incremento de la precisión predictiva y la comprensión de los resultados

A la vez Lin (2003) no informa que, en el aprendizaje supervisado para la clasificación, la influencia que tienen los atributos sobre el atributo clase (atributo objetivo) juega un rol muy importante la selección de atributos. Es así, que un aspecto importante que debe ser considerado en el proceso de selección de atributos, es la relevancia y redundancia de los atributos sobre el atributo clase. La relevancia de atributos se clasifica en tres tipos: gran relevancia, poca relevancia e irrelevantes. Siempre es necesario tener un atributo con relevancia fuerte en el subconjunto óptimo y por lo cual no se puede eliminar. Se dice que un atributo es poco relevante, si es necesario para un subconjunto óptimo y sólo en ciertas condiciones. Un atributo irrelevante, es aquel que no aporta ninguna información al atributo clase, se comporta como un ruido para el clasificador y por lo tanto debe ser eliminado. Así mismo, se dice que un atributo es redundante, si toma similar comportamiento que otro. La eliminación de estos atributos irrelevantes y redundantes, reduce la dimensionalidad del espacio de atributos consiguiendo reducir el tiempo de ejecución, mejorar la calidad y aumentar la precisión del aprendizaje supervisado con los algoritmos de clasificación.

En la selección de atributos para el aprendizaje supervisado existen tres enfoques o métodos que dependen de la relación que se defina entre la TMD supervisada y el algoritmo de selección de atributos: 1) el enfoque basado en el filtrado, el proceso de selección de los atributos es independiente del algoritmo de minería de datos y usa métricas para medir la dependencia de cada atributo con el atributo clase proporcionando como resultado un ranking de atributos; 2) el enfoque de Wrapper, los métodos son dependientes del algoritmo de minería de datos y usa algoritmos de búsqueda para seleccionar el subconjuntos de atributos y 3) el enfoque híbrido que combina los dos métodos. Ladha et al. (2011) menciona que los métodos de filtrado, usan para la selección de atributos una métrica o medida (el coeficiente de correlación, la información mutua, la distancia euclidiana, ganancia de información, medida de Relief, etc.) que permite evaluar la relevancia de cada atributo con respecto al

atributo clase. Mientras que el método wrapper, se basa en aplicar algoritmos búsqueda para obtener un subconjunto de atributos.

Para mostrar la aplicación de los métodos para la selección de atributos, se usarán los datos de la Encuesta Nacional de Satisfacción de Usuarios de Salud 2015 que se encuentra en el portal del INEI. Se propone desarrollar el proceso para la selección de atributos a partir de generar los subconjuntos de atributos, la evaluación, criterio de detección y la evaluación de los subconjuntos formados. Con esta finalidad, se aplica cuatro métricas para la selección de atributos por filtrado (Chi-Cuadrado, Ganancia de información, Razón de ganancia y Relief) y cuatro métodos de Wrapper (Best-First, Greedy Forward, Greedy Backwar y Hill Climbing). Así mismo, con la finalidad de evaluar y comparar los métodos de selección de atributos, se propone usar las TMD de regresión logística binaria, árbol de clasificación C4.5, redes bayesinas Naive y random forest, y métricas a partir de la matriz de confusión como tasa de precisión curvas ROC y AUC que permitan evaluar su capacidad predictiva. Se usará las respectivas librerías del programa R para el procesamiento de los datos.

Objetivo general:

Presentar los métodos de selección de atributos por filtrado y wrapper en el aprendizaje supervisado, con la finalidad de identificar los atributos más relevantes para mejorar la capacidad predictiva de las técnicas de minería de datos para predecir la satisfacción de los usuarios de la atención recibida en la Encuesta Nacional de Satisfacción de Usuarios de Salud – 2015.

Objetivos específicos:

1. Presentar el proceso de los métodos de selección de atributos por filtrado y wrapper.
2. Identificar los atributos relevantes aplicando los métodos de selección de atributos por filtrado y wrapper para predecir la satisfacción de los usuarios de la atención recibida de los servicios de salud.
3. Evaluar y comparar la capacidad predictiva de los métodos de selección de atributos por filtrado y Wrapper usando las TMD de regresión logística binaria, árbol de decisión C5.0, redes bayesianas Naive y clasificador random Forest.

II. REVISIÓN DE LITERATURA

Según Bolón et al. (2013) existen una gran variedad de métodos para la selección de atributos en la literatura, y por consiguiente es necesario hacer estudios comparativos sobre su comportamiento en el aprendizaje supervisado. Conocer los atributos relevantes dentro de un conjunto de datos reales y la eficacia de los métodos de selección de atributos es muy difícil de determinar, ya que dichos conjuntos de datos pueden incluir muchos aspectos a considerar, como el gran número de atributos irrelevantes y redundantes, datos ruidosos y alta dimensionalidad en términos de atributos. Por lo tanto, el rendimiento de los algoritmos de aprendizaje supervisado depende en gran medida del manejo de la colección de atributos. La performance de los métodos para la selección de atributos, se basan en usar una variedad de medidas o métricas que evalúan el rendimiento del atributo clase con el conjunto de variables predictoras. Por lo tanto, los nuevos métodos y estrategias para la selección de atributos están aumentando constantemente para abordar el problema del tratamiento de la alta dimensionalidad.

2.1 La selección de atributos en el aprendizaje supervisado

La selección de atributos surgió como un campo de investigación y desarrollo en los años 70 en las áreas de la estadística, minería de datos y máquinas de aprendizaje. La selección de atributos es considerada un problema fundamental en el análisis de datos, donde se han propuesto muchos métodos, técnicas, estrategias y enfoques, y que es aplicada en muchas áreas de investigación, tales como reconocimiento de patrones, clasificación de textos, metadatos, bioinformática, predicción de defectos de software, análisis de sentimientos, recuperación de imágenes, análisis de stock de mercados, etc.

Según Yun et al. (2007) la selección de atributos en el aprendizaje automático y reconocimiento de patrones es un área importante, dónde muchas investigaciones se están realizando. Se realiza un estudio experimental con algunos algoritmos de selección de atributos y se analiza su desempeño usando varios conjuntos de datos de dominio público. Los resultados indican que el número de atributos reducidos con métodos propuestos para la selección de características mejora del rendimiento de aprendizaje. Se usaron siete conjuntos de datos (5 de UCI Machine Learning atributos discretos y 2 micro arreglos de cáncer y leucemia atributos continuos) para medir el rendimiento de los métodos de selección de atributos se usó Weka evaluando los subconjuntos seleccionados con dos algoritmos de aprendizaje Naive Bayes (NB) y Support Vector Machines (SVMs) y utilizando la validación

cruzada de 10 folds para los de datos UCI y Leave-One-Out (LOOCV) para los dos conjuntos de datos de micro arreglos.

Novakovic et al. (2011) presenta una comparación entre varios métodos de filtrado (6 métodos de ranking y 4 algoritmos de aprendizaje supervisado: IB1, Naive Bayes, C4.5 y Función base radial). Los resultados aplicados a dos conjuntos de datos reales indicaron que los métodos de ranking son importantes para la precisión de la clasificación. Schiezaro et al. (2013) menciona que se investigó un método de selección de características basado en el algoritmo artificial de colonia de abejas para la clasificación de diferentes conjuntos de datos. Los resultados muestran que un número reducido de características puede alcanzar una mayor precisión de clasificación en comparación con el uso del conjunto completo de características. ElAlami (2009), se presenta un nuevo algoritmo de selección de características que es entrenado por la red neuronal utilizando un algoritmo genético que está destinado a encontrar las características relevantes óptima que maximizan la función de salida de la red neuronal artificial entrenada.

Yu et al. (2003), proponen un filtro de correlación rápida (FCBF) método que puede identificar la relevancia y redundancia entre las características pertinentes utilizando el concepto de correlación predominante. El resultado muestra que el método FCBF maneja eficientemente la reducción de atributos cuando hay un alto grado de la dimensionalidad. Así mismo Yin et al. (2013) ilustran los problemas de selección de características y la ineficiencia del método tradicional de selección (coeficiente de correlación, información mutua y criterio fisher). Se proponen dos enfoques diferentes (i) la descomposición de grandes clases en pequeñas subclases con un tamaño relativamente uniforme y luego calcular la bondad de las características con los nuevos datos descompuestos. (ii) el método de selección de funciones basado en la distancia de Hellinger. Los resultados muestran que los dos enfoques propuestos superan los métodos tradicionales de selección de características. Tu et al. (2007) proponen la optimización de enjambre de partículas para el rendimiento de la selección de características y la máquina de vector de soporte vectorial como método que sirve hallar un valor de la aptitud de PSO (Particle Swarm Optimization) para el problema de la clasificación. Los resultados del método propuesto muestran la optimización del proceso de selección de características y el aumento de la precisión de clasificación en comparación con otros métodos de selección de características existentes. Según Brahim et al. (2016) proponen un nuevo método de selección de características híbridas basado en el aprendizaje de instancias. Su tarea principal es cambiar el problema del tamaño pequeño de la muestra a una herramienta que permite

seleccionar algunos subconjuntos de características para analizar en una etapa de filtro. A continuación, se propone una búsqueda de subconjunto cooperativo con un algoritmo clasificador como sistema de evaluación de Wrapper. El método propuesto supera a otros métodos en términos de precisión y estabilidad de la selección de características.

Según Dasgupta et al. (2007) el problema de los algoritmos para la clasificación automatizada de textos surge por la alta dimensionalidad, que tienen el desafío de seleccionar una estructura de datos apropiada para representar los documentos y por lo cual debe elegirse una función objetivo para optimizar, para evitar la sobrecarga y obtener buena precisión.
Forman (2003), se presenta una comparación empírica de doce métodos de selección de características (Gain de información, Información mutua, etc.) evaluados en un índice de referencia de 229 ejemplos de problemas de clasificación de texto recopilados por Reuters y medidas de precisión. Los resultados revelan que una nueva métrica de selección de características que llamamos 'Separación Bi-Normal' (BNS), superó a los otros por un margen sustancial en la mayoría de las situaciones.

2.2 Fundamentos de la selección de atributos

2.2.1 Relevancia de atributos

El conjunto de atributos óptimo es un subconjunto de todos los atributos relevantes. En la literatura, los atributos se clasifican por su relevancia en tres categorías: irrelevante, débilmente relevante y relevante.

Kumar y Minz (2014) definen la relevancia de atributos como *S* el conjunto de datos compuesto por |*S*| instancias (observaciones) y se considera como el resultado del muestreo. El dominio de los atributos es el conjunto $X=\{x_1, x_2, \ldots, x_n\}$ y el espacio de la instancia es definido como $I=I_1xI_2x\ldots xI_m$. Sea *P* la distribución de probabilidad de *I*. La función objetivo $C:I \rightarrow L$ según su atributo de relevancia, donde *L* es un espacio de etiquetas.

Definición 1: Relevancia con respecto a un objetivo. Un atributo es relevante para un concepto objetivo *C*, si existe un par de ejemplos *A y B* y en el espacio de instancias tal que A y B difieren sólo en su asignación para el atributo x_i y $C(A)\neq C(B)$.

Definición 2. Relevancia alta con respecto a una muestra/distribución. Un atributo $x_i \varepsilon$ X es altamente relevante para la muestra S, si existe un par de ejemplos A, B ε S que sólo difieren en su asignación a x_i y $C(A)\neq C(B)$. Un atributo $x_i \varepsilon$ X es fuertemente relevante para

un objetivo C en la distribución P, si existe un par de ejemplos A, B ε I con P(A)≠0 y P(B)≠0 que solamente difieren en su asignación a x_i y *C(A)≠C(B)*.

Definición 3: Relevancia débil con respecto a una muestra/distribución. Un atributo x_i ε X es poco relevante para la muestra S, si existe al menos X'⊂X(x_i ε X') donde x_i es fuertemente relevante con respecto a S. Un atributo x_i ε X es débilmente relevante al objetivo C en la distribución P si existe por lo menos un apropiado X'⊂X(x_i ε X') donde x_i es fuertemente relevante con respecto a P.

Definición 4: Relevancia como medida de complejidad. Dada una muestra de datos S y un conjunto de conceptos C, sea r(S,C) el número de atributos relevantes que utilizan la Definición 1 para un concepto en C que fuera de entre todos aquellos cuyo error sobre S es menor, teniendo el menor número de características relevantes. El rendimiento óptimo sobre S con el concepto C, utilizando el menor número de características. Los conceptos anteriores de relevancia son independientes del algoritmo de aprendizaje específico. Esto significa que no es necesario que una característica relevante dada sea adecuada para el algoritmo de aprendizaje.

Definición 5: Utilidad incremental. Dada una muestra de datos S, un algoritmo de aprendizaje L y un subconjunto de atributos X', el atributo x_i es incrementalmente útil a L con respecto a X' si la precisión de la hipótesis que L produce utilizando el conjunto de atributos {x_i } es mejor que la exactitud lograda utilizando sólo el subconjunto de atributos X'.

Definición 6: Entropía Relevancia. Denotando la información mutua I(x;y)=H(x)-H(x/y) con entropía de Shannon (x), la entropía relevancia de x para y es definido como r(x;y)=I(x;y)/H(y). Sea C el objetivo visto como un atributo y X es el conjunto original de atributos, un subconjunto X'⊂X es suficiente si I(X',C)=I(X,C). Para un subconjunto suficiente, debe satisfacer r(X',C)=r(X,C). Por lo tanto, r(X',C) y r(X,C) y son maximizadas conjuntamente.

2.2.2 Medidas para la selección de atributos

Los métodos para la selección de atributos se basan en medidas o métricas para evaluar la relevancia de atributos. Entonces se establecen métricas para identificar los rangos o subconjuntos de los mejores atributos y evaluar el desempeño de los algoritmos para la selección de atributos por filtrado o por Wapprer. Según Molina et al. (2002) un procedimiento es evaluar una medida que valore la precisión de los algoritmos tomando en

cuenta la relevancia, la irrelevancia y la redundancia en los conjuntos de datos y obtener un subconjunto de atributos con un mejor grado de rendimiento.

Los métodos de filtrado aplicados en la selección de atributos, tienen como resultado el ranking de los atributos que permite seleccionar un sub conjunto de atributos, para lo cual utiliza alguna métrica para medir y evaluar el grado de dependencia o asociación de cada atributo con el atributo clase. Entre las principales medidas o métricas que se utilizan en los métodos de filtrado para la selección de atributos en los algoritmos de aprendizaje supervisado, se tiene:

1. Ganancia de información (GI).

Se basa en la medida de entropía para determinar la relevancia entre un atributo y el atributo clase. La GI entre el atributo X y el atributo clase Y es calculada por la expresión:

$$GI(X,Y)=H(X)-H(X/Y).$$

Donde:

H(X) es la entropía de X. La entropía de X puede ser calculado: $H(X)=-\sum_i P(x_i)\log_2 P(x_i)$.

H(X/Y) Es la entropía de Y después de observar X. La entropía de Y después de observar X se puede calcular: $H(X/Y)=-\sum_j P(y_j)\sum P(x_i/y_j)\log_2 P(x_i/y_j)$

El método de selección de atributos basado en la GI, calcula la GI para cada atributo separadamente y selecciona los *m* atributos más relevantes entre los n atributos $\{x_1, x_2, \ldots, x_n\}$ con los más altos valores de la GI. Se busca seleccionar los atributos con altos valores de GI. La GI está en el rango [0, 1]. Si GI=1, indica que el conocimiento de X predice completamente Y y si GI=0, indica que X e Y están incorrelacionados. La GI no puede manejar características redundantes puesto que la selección es de manera univariada.

2. Razón de Ganancia (RG).

Se basa en la medida de entropía para determinar la relevancia entre un atributo X y el atributo clase Y. La RG, permite eliminar el sesgo que se presenta con la GI. La RG se calcula por: $RG=\frac{GI}{H(X)}$

Así, para predecir el atributo clase Y, es necesario normalizar la medida dividiendo GI por la entropía de X. Se busca atributos con menores valores de RG.

3. Información Mutua (IM).

Es un método ponderado simple y supervisado para seleccionar atributos. La IM se calcula entre cada atributo X y el atributo clase Y, de tal manera que con sus valores se obtiene un ranking ordenado de mayor a menor de los n atributos. La IM se calcula por:

$$IM(x_i, y) = \sum_i \sum P(x_i, y_j) \log_2 \left(\frac{P(x_i, y_j)}{P(x_i) \times P(y_j)} \right).$$

La IM provee una buena medida para evaluar la importancia de un atributo. El atributo X, es importante si la IM(X,Y) con el atributo clase Y es alta.

4. Estadístico Chi (X^2).

La estadística del chi cuadrado evalúa la asociación entre dos variables y determina si son independiente o relacionadas. La prueba estadística se obtiene como:

$$X^2 = \sum_{i=1}^{r} \sum_{j=1}^{c} \frac{(O_{i,j} - E_{i,j})^2}{E_{i,j}}$$

5. Incertidumbre simétrico (SU).

En la métrica de la ganancia de información modificada para lo cual la GI se divide por la suma de las entropías del atributo X y atributo clase Y. Se calcula por:

$$SU = 2 \times \frac{GI(X,Y)}{H(X) + H(Y)}$$

La SU puede tomar valores entre el rango [0, 1]. Si SU=1, especifica que el conocimiento de un atributo predice completamente al otro. Si SU=0, indica que X e Y no están correlacionados. Este método selecciona atributos con los menores valores de SU.

6. Relief

El algoritmo Relief fue desarrollado por Kira y Rendell (1992) como un enfoque simple, rápido y efectivo para el pesaje de atributos. El resultado del algoritmo es un peso entre -1 y 1 para cada atributo, con pesos más positivos que indican atributos más predictivos. El pseudo código Relief es el siguiente:

```
set W[a] = 0 for each attribute a
for i = 1 to n do
        select sample si from data at random
        find nearest hit sh and nearest miss sm
        for each attribute a do
            ΔWi[a] = diff(a, si, sm) - diff(a,sj, sh)
              W[a] = W[a] + ΔWi[a]
         end for
end for
for each attribute a do
        W[a] = W[a] / n
end for
where diff(a, si, sj) = 0, if si[a] = sj[a]
                      = 1, if si[a] ≠ sj[a]
```

2.2.3 Métodos de selección de atributos

Mitra et al. (2002) mencionan que la selección de atributos tiene varios beneficios en los modelos de minería de datos; tales como, reducir la complejidad computacional, mejorar la interpretación del modelo y reducir el sobre ajuste del modelo. Según Ladha (2011) la finalidad principal de la selección de atributos en el aprendizaje supervisado, es seleccionar un sub conjunto de atributos de entrada basados en la eliminación de atributos con poca o nada información predictiva. Los métodos para la selección de atributos pueden considerarse en tres categorías: métodos de filtrado, métodos de Wrapper y métodos híbridos. A continuación, se describen las tres categorías.

1. Métodos de filtrado

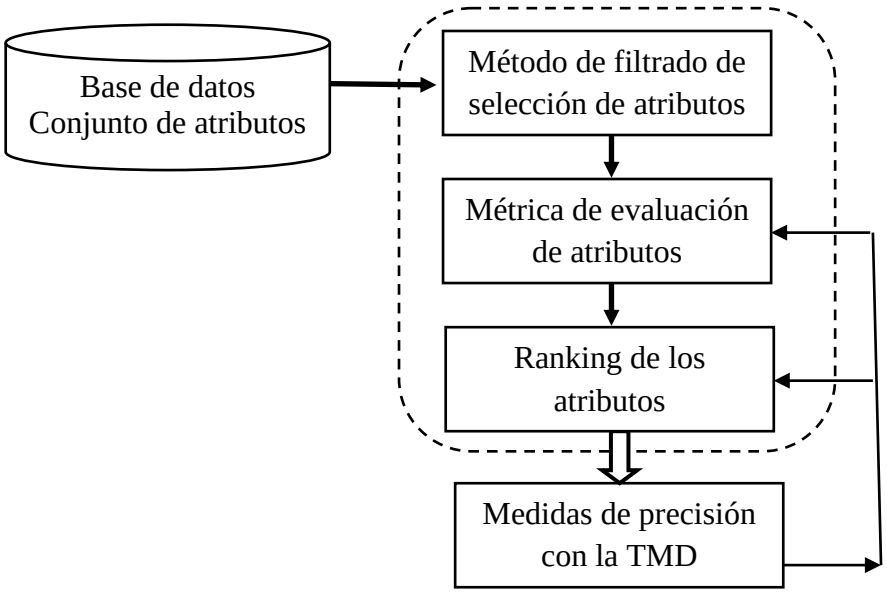
Son métodos que se basan en un proceso de filtrado para seleccionar los atributos relevantes en el aprendizaje supervisado y son independientes del algoritmo de clasificación (TMD) que se está utilizando. Para la selección de los atributos relevantes se usan medidas o métricas que evalúan la importancia de cada atributo con el atributo clase, tales como: Ganancia de

información, Razón de ganancia, Chi-cuadrado, Relief, etc. El uso de la métrica depende del tipo de atributos (discretos o continuos) que se están evaluando. El resultado de la aplicación de un método de filtrado, es la colección de conjunto de atributos relevantes con un ranking (ordenado) de todos los atributos según la métrica utilizada.

En la Figura 1, se muestra el proceso del método de filtrado. Se inicia con la base de datos con todos sus atributos. Se aplica el método de filtrado para la selección de atributos, luego se selecciona alguna métrica para evaluar la relevancia de los atributos con el atributo clase, el resultado es la colección del ranking de todos los atributos ordenados por la métrica empleada. Con el resultado del ranking se selecciona un subconjunto de atributos y se le aplica la TMD supervisada propuesta, luego se calcula alguna medida (Tasa de buena clasificación, Matriz de confusión, curva ROC, AUC, etc.), para evaluar su precisión de la TMD. Si los valores de la medida no satisfacen la precisión deseada entonces se vuelve a elegir del ranking otro subconjunto de atributos o en otro caso se elige otra métrica para obtener un nuevo ranking de atributos.

Figura 1

Método de filtrado para la selección de atributos



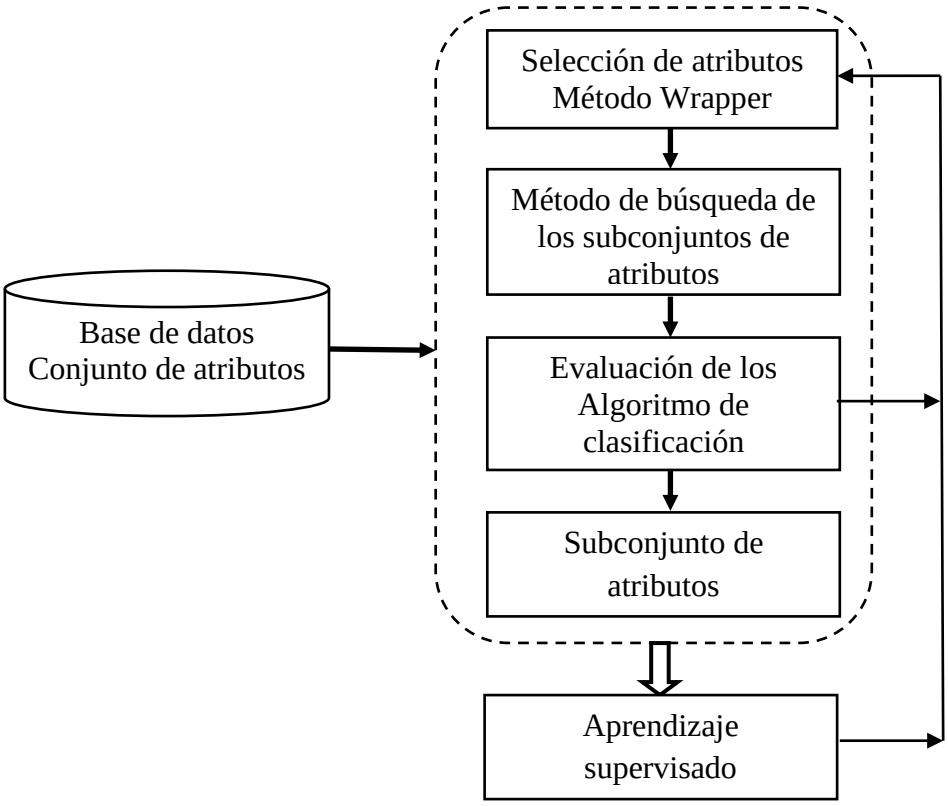
2. Métodos envolventes (Wrappers)

Estos métodos son dependientes de la TMD utilizada. La selección de los atributos está en función de la precisión del mismo algoritmo de clasificación que se está aplicando. Estos métodos seleccionan un subconjunto de atributos relevantes, a través de algoritmos que permiten su búsqueda y evaluación sobre la precisión con el atributo clase. Generalmente se usa la matriz de confusión, curvas ROC y medida de precisión la tasa de buena clasificación. El resultado de aplicar el método wrapper, es un subconjunto de atributos relevantes.

En la Figura 2, se muestra el proceso del método de wrapper. Se tiene la base de datos con todos los sus atributos. La selección de los subconjuntos de atributos se inicia con aplicar algún método de búsqueda (forward, backward, etc.) y criterio para definir la forma de ser evaluados (best-firt, beam-search, hill climbing, greedy, etc.). La evaluación de los subconjuntos de atributos se realiza aplicando TMD como el algoritmo de clasificación definido en el aprendizaje supervisado evaluando su precisión. Si no se consigue una precisión de la clasificación buena, se vuelve a considerar otro subconjunto de atributos

Figura 2

Método de Wrapper para la selección de atributos



3. Métodos híbridos

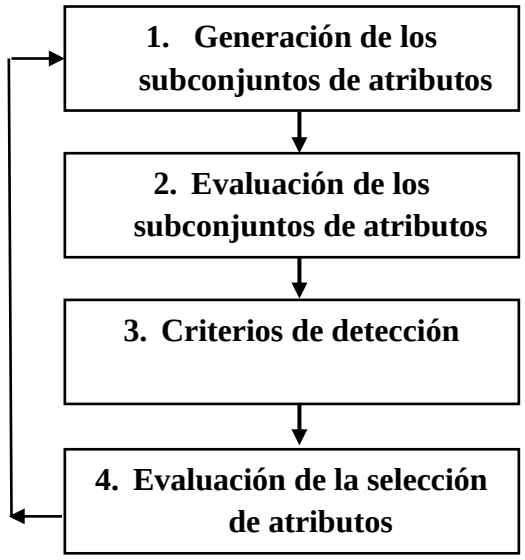
Combina los métodos de filtrado y de wrapper, para la selección del subconjunto de atributos relevantes.

2.3 Proceso para la selección de atributos

En el aprendizaje supervisado, existen dos enfoques principales para la selección de atributos. El primero es la evaluación individual (método de filtrado) y el segundo es la evaluación de subconjuntos (método wrapper). En la evaluación individual, la clasificación o ranking se basa en hallar el peso o importancia de una característica individual de acuerdo a su grado de relevancia con el atributo clase usando una métrica. En la evaluación de subconjuntos, estos se construyen utilizando una estrategia de búsqueda y la evaluación de los subconjuntos relevantes se miden con la matriz de confusión aplicando la TMD de clasificación propuesto para el aprendizaje supervisado. En la Figura 3, se presenta el procedimiento propuesto para la selección de atributos se puede realizar en cuatro pasos:

Figura 3

Proceso para la selección de atributos



1. Generación de subconjuntos.

Es una búsqueda heurística en la que cada estado especifica un subconjunto candidato para la evaluación en el espacio de búsqueda. Dos cuestiones básicas determinan la naturaleza del proceso de generación de subconjuntos. En primer lugar, la generación sucesora decide el

punto de partida de búsqueda, que influye en la dirección de búsqueda. Para decidir los puntos de partida de búsqueda en cada estado, se pueden considerar los métodos de avance, retroceso, compuesto, ponderación y aleatorios. En segundo lugar, la organización de búsqueda es responsable del proceso de selección de características con una estrategia específica, como búsqueda secuencial, búsqueda exponencial o búsqueda aleatoria.

Estrategias de búsqueda

La generación de los subconjuntos de atributos involucra usar una estrategia de búsqueda. Para el total de atributos existen en una base de datos, se tiene C^{N}_{2} de posibles subconjuntos de atributos que se pueden formar, por lo que se requiere de una buena estrategia de búsqueda que se realiza por la ejecución de una serie de pasos recursivos. La estrategia de búsqueda, requiere especificar en primer lugar el criterio de la conformación de los subconjuntos de atributos (punto inicial de búsqueda). Se puede considerar varias estrategias:

- **La selección hacía adelante (forward selection).** Consiste en ir incorporando los atributos en forma progresiva en cada paso. Se empieza con un conjunto de atributos vacío y luego se van conformando los subconjuntos añadiendo los atributos relevantes. Es computacionalmente más eficiente, pero tiende a producir peores subconjuntos, ya que se toman decisiones locales.

- **La selección hacía atrás (backward selection).** Consiste en ir eliminando progresivamente los atributos en cada paso. Se empieza con un conjunto de todos los atributos seleccionados y luego se van conformando los subconjuntos eliminando los atributos irrelevantes. Es computacionalmente más caro, pero puede considerar atributos débiles individualmente, pero fuertes cuando se consideran en conjunto.

- **La selección de la combinación hacía adelante y atrás (bi-direccional).** Consiste en ir añadiendo o eliminando en cada paso atributos partiendo de un subconjunto inicial. que se añaden y quitan atributos y la selección aleatoria para conformar los subconjuntos de atributos. Una alternativa es añadir (o quitar) p atributos en cada paso y eliminar (añadir) q atributos en el siguiente paso ($p > q$).

- **La selección aleatoria (random selection).** Esta estrategia consiste en generar inicialmente en forma aleatoria un subconjunto de atributos y luego continuar y con una

estrategia de búsqueda (greedy, best-first, hill climbing, etc.) repitiendo el proceso varias veces se puede introducir aleatoriedad a la búsqueda.

2. Evaluación de los subconjuntos de atributos.

Los subconjuntos de atributos generados deben ser evaluados por un cierto criterio de evaluación. Por lo tanto, en la literatura se han propuesto muchos criterios de evaluación para determinar la bondad del subconjunto candidato de las características. Basándose en su dependencia de los algoritmos de minería de datos, los criterios de evaluación pueden clasificarse en dos grupos: criterios independientes y dependientes.

- **Los criterios independientes (métodos de filtrado).** Evalúan las bondades de los subconjuntos de atributos de los datos de entrenamiento sin involucrar ningún algoritmo de minería de datos. En este caso se usan métricas o medidas, con la finalidad de valuar el rendimiento de cada atributo en particular con el atributo clase. Las métricas pueden estar basadas en: medida de distancia, medida de información, medida de dependencia o medida de consistencia. Son rápidos y producen una lista ordenada de atributos de acuerdo a su medida de evaluación o *ranking.* Una vez producida la lista, se tiene que especificar donde cortar (o hasta qué atributo considerar) y para esto existen diferentes opciones. Estos métodos no pueden capturar combinaciones que podrían dar buenos resultados. Por lo que una variable que aparentemente no sirve por sí sola, puede dar resultados muy buenos en combinación con otras. Inclusive dos variables que no aportan nada por separado pueden ser útiles juntas. Los que evalúan un atributo a la vez pueden eliminar atributos irrelevantes, pero no los redundantes, ya que tienen una evaluación parecida a otros.

- **Los criterios dependientes (métodos wrapper).** Implican usar algoritmos de minería de datos predeterminados para evaluar la bondad del subconjunto de los atributos seleccionados, basados en el rendimiento con algoritmo de clasificación.

3. Criterios de detención.

Para detener el proceso de selección de atributos, se debe determinar un criterio de detención. El proceso de selección de atributos se detiene en el procedimiento de validación. No es la parte del proceso de selección de atributos, pero el método de selección de características debe validarse realizando diferentes pruebas y comparaciones con resultados previamente

establecidos o comparándolos con los resultados de métodos competitivos usando conjuntos de datos artificiales, conjuntos de datos del mundo real o ambos.

4. Evaluación de la selección de atributos

Consiste en evaluar la selección de atributos. Para los métodos de filtrado la evaluación de la selección de atributos se realiza con la métrica o medida utilizada para la discriminación y la formación del ranking de los atributos. Para los métodos wrapper, la evaluación de los subconjuntos de atributos seleccionados se realiza con alguna métrica o criterio que se genera a partir de aplicar el algoritmo de aprendizaje y que permita evaluar el rendimiento o precisión de los subconjuntos de atributos.

Evaluación para los métodos de filtrado

Existe una gran cantidad de medidas para evaluar la relevancia de los atributos. Estas métricas se usan para evaluar individualmente el rendimiento de cada atributo con el atributo clase, aunque muchas de ellas se pueden extender para evaluar todo un subconjunto. Se obtienen una lista de atributos ordenados o rankeados por la métrica utilizada. La finalidad es seleccionar el mejor subconjunto de atributos. Se puede clasificar las medidas en los siguientes tipos:

- **Medidas de distancia.** Se basan en evaluar la cercanía (similitud) a través de medir la distancia entre cada atributo X y el atributo clase C. Entre las medidas para medir la distancia se tiene: Euclideana, Mahalanobic, Manhathan, etc.
- **Medidas de información.** La ganancia de información del atributo *X* la podemos definir como la diferencia entre la incertidumbre previa y la posterior al usar *X*. El atributo *X* es mejor que *Y*, si *X* tiene más ganancia de información que *Y*.
- **Medidas de dependencia**. Son métricas que miden la dependencia o correlación entre atributos. Si la correlación del atributo *X* con la clase *C* es mayor que la del atributo *Y* con la clase *C*, entonces *X* es mejor que *Y*. Esta medida también permite determinar redundancia de un atributo, en el sentido que evalúa que tan correlacionado está un atributo con otros para.
- **Medidas de consistencia.** Miden la consistencia de las hipótesis con respecto a un grupo de atributos, buscando el mínimo conjunto de atributos que genere hipótesis consistentes.

Evaluación para los métodos de wrapper

La evaluación de la selección de los atributos en los métodos wrapper, se basa el medir el grado de precisión que se consigue con el subconjunto de atributos aplicando el algoritmo de aprendizaje supervisado. Las métricas más usadas para evaluar los subconjuntos de atributos seleccionados, son aquellas que se calculan a partir de la matriz de confusión.

- **Matriz de confusión**

La tabla de clasificación o matriz de confusión, se usa para evaluar y comparar las técnicas de minería de datos aplicadas a la tarea de la clasificación (aprendizaje supervisado), cuando la variable clase es categórica. La tabla de clasificación, es una tabla de contingencia que muestra la distribución del número de observaciones de la correcta e incorrecta clasificación, con respecto a la clasificación observada y la predicha, para las distintas categorías de la variable clase. A partir de esta tabla se estima los porcentajes de la correcta e incorrecta clasificación (la tasa de aciertos y error) que realiza el clasificador con el conjunto de datos. En el Tabla 1, se muestra la tabla de clasificación para el caso de dos clases: Positiva (Clase 0) y Negativa (Clase 1).

Tabla 1

Matriz de confusión de clasificación para dos clases

Clasificación observada	**Clasificación predecida**		**Total Observado**
	Positiva (Clase 0)	Negativa (Clase 1)	
Positiva (Clase 0)	VP	FN	VP + FN
Negativa (Clase 1)	FP	VN	FP + VN
Total Predecido	VP + FP	FN + VN	N

Nota: Donde $N = VP + VN + FP + FN$

- **El VP (verdaderos positivos)**, es el número de observaciones que predice correctamente el clasificador como la clase positiva. **El FP (falsos positivos)**, es el número de observaciones que se predice incorrectamente como la clase positiva siendo de la clase negativa. **El VN (verdaderos negativos)**, es el número de observaciones que predice correctamente el clasificador como la clase negativa. **El FN (falsos negativos)**, es el número de observaciones que se predice incorrectamente como la clase negativa siendo de la clase positiva.

- **Los totales de las filas**, representan el número de observaciones de la clasificación observada (real) para las clases positiva (VP+FN) y negativa (FP+VN). **Los totales de las columnas**, representan el número de observaciones de la clasificación predecida (por el clasificador) para las clases positiva (VP+FP) y negativa (FN+VN).

- Se puede determinar la proporción observada (probabilidad a priori) de la clase positiva: $P(Clase\,0) = \frac{VP + FN}{N}$ y negativa: $P(Clase\,1) = \frac{FP + VN}{N}$. Además la proporción que el clasificador predice la clase positiva: $\hat{P}(Clase\,0) = \frac{VP + FP}{N}$ y la clase negativa: $\hat{P}(Clase\,1) = \frac{FN + VN}{N}$.

A partir de la tabla de clasificación se pueden definir una serie de métricas con la finalidad de evaluar y comparar la performance de los clasificadores. En la Tabla 2 se muestra un conjunto de métricas usadas para evaluar clasificadores, expresadas generalmente en porcentajes.

Tabla 2

Métricas para evaluar clasificadores

Clase	TVP Tasa de verdaderos	TFP Tasa de falsos	Precisión	F-Medida
Clase 0	TVP_0	TFP_0	P_0	F_0
Clase 1	TVP_1	TFP_1	P_1	F_1

Exactitud. Es la métrica más usada para medir la bondad de los clasificadores. La exactitud de un clasificador, se mide a través de la tasa de aciertos.

- **La tasa de acierto.** Se define como la proporción de observaciones que el clasificador predice correctamente (buena clasificación) las clase positiva y negativa, con respecto al total de los datos: $TA = \frac{VP + VN}{N}$. **La tasa de error.** Es la proporción de observaciones que el clasificador predice incorrectamente (mala clasificación): $TE = \frac{FN + FP}{N} = 1 - TA$.

- **La tasa de verdaderos positivos y negativos son respectivamente:** $TVP_0 = \frac{VP}{VP + FN}$ y $TVP_1 = \frac{VN}{FP + VN}$, indican la proporción de observaciones que se predice correctamente la

clase positiva o negativa con respecto a su total observado (totales de filas). Los valores de TVP_0 y TVP_1, son conocidos como la Sensibilidad y la Especificidad respectivamente.

- **La tasa de falsos positivos y negativos son respectivamente:** $TFP_0 = \frac{FN}{VP + FN}$ y $TFP_1 = \frac{FP}{FP + VN}$, indican la proporción de observaciones que se predice incorrectamente la clase positiva o negativa con respecto a su total observado (totales de fila).

Precisión. Es la proporción de observaciones que el clasificador predice correctamente la clase positiva o negativa con respecto a su total predecido (totales de columna). Así, $P_0 = \frac{VP}{VP + FP}$ y $P_1 = \frac{VN}{FN + VN}$, son respectivamente la proporción de observaciones que predice correctamente el clasificador la clase positiva o negativa. Además, $1 - P_0 = \frac{FP}{VP + FP}$ y $1 - P_1 = \frac{FN}{FN + VN}$, son la proporción de observaciones que predice incorrectamente el clasificador la clase positiva y negativa respectivamente.

F-Score. Se define como la media armónica entre la precisión y recall: $F - Measure = \frac{(1 + \beta^2) \times \text{Pr}ecision \times \text{Re}call}{(\beta^2 \times \text{Pr}ecision) + \text{Re}call}$. Donde $\beta \varepsilon [0,1]$, determina la influencia relativa de ambas métricas. En la mayoría de los casos se usa $\beta = 1$. Por lo cual resulta: $F - Measure = \frac{2 \times \text{Pr}ecision \times \text{Re}call}{\text{Pr}ecision) + \text{Re}call} = \frac{2\,VP}{2\,VP + FP + FN}$

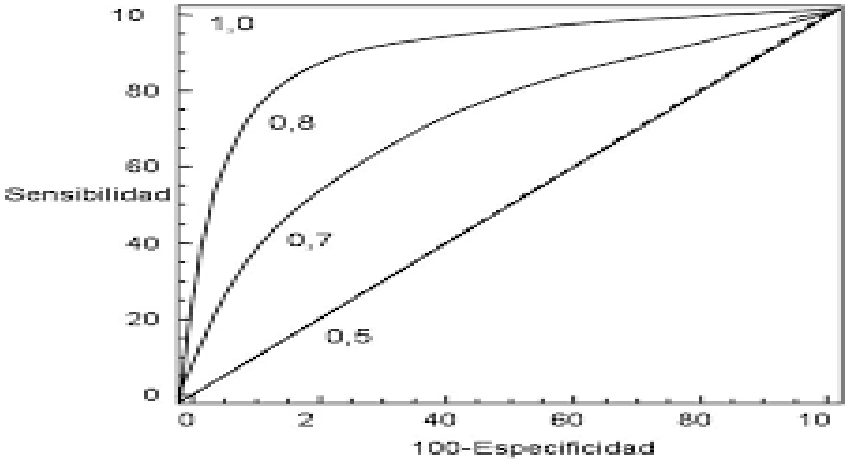
- **Análisis de las Curvas ROC.**

El análisis ROC (Receiver Operating Characteristic), es usado para comparar y valuar si un clasificador es mejor que otro, seleccionando aquel que tenga el mejor porcentaje de buena clasificación. Wang (2005) menciona que son curvas que muestran la habilidad del clasificador para posicionar las instancias verdaderas respecto a las falsas. En una definición más acertada se puede decir que las Curvas ROC son las que miden la relación de la tasa de verdaderos positivos (predicciones acertadas) versus la tasa de falsos positivos (predicciones erradas). La curva ROC se obtiene representando para cada posible elección del valor de corte, graficando en el eje X la Tasa de falsos positivos (**1- *especificidad)*** y en el eje Y la Tasa de verdaderos positivos (***sensibilidad)***. La curva ROC, es creciente lo cual refleja que, al

modificar el valor de corte para obtener mayor sensibilidad, sólo se puede hacer a expensas de disminuir al mismo tiempo la especificidad. Si la prueba no discrimina entre las clases, la curva ROC sería la diagonal que une los vértices inferior izquierdo y superior derecho. La exactitud de la prueba aumenta a medida que la curva se desplaza desde la diagonal hacia el vértice superior izquierdo. Si la discriminación fuera perfecta (100% de sensibilidad y 100% de especificidad) la curva pasaría por dicho punto. Por lo tanto, lo que se espera encontrar son puntos sobre una curva por encima de la diagonal y entre más cercana se encuentre de la esquina superior izquierda, mejor será su predicción.

Figura 4

Ejemplo de una curva ROC



- **Análisis de las AUC**

El área bajo la curva ROC se puede usar como un índice conveniente de la exactitud global de la prueba, llamada AUC (Área bajo la curva ROC) como un índice de la performance del clasificador (la exactitud máxima correspondería a un valor del área bajo la curva de 1 y la mínima a un valor de 0.5). Según Fayyad et al. (1996) menciona que el AUC de un clasificador indica la probabilidad que el clasificador posicionará una instancia aleatoria positiva mejor que una instancia aleatoria negativa. La AUC se calcula con la siguiente expresión:

$$AUC = \frac{1 + TVP - TFP}{2}, \qquad 0 \le AUC \le 1$$

- **Coeficiente de concordancia de Kappa (k).**

Es un coeficiente estadístico propuesto originalmente por Cohen (1960) que permite evaluar la concordancia entre los resultados de dos o más variables cualitativas. El índice k, aplicado a la tabla de confusión permite evaluar si la clasificación observada es similar (concordante) con la clasificación predicha por el clasificador. Para el caso de dos categorías, el coeficiente de Kappa se calcula:

$$k = \frac{P_0 - P_e}{1 - P_e}, \qquad 0 \le k \le 1 \qquad con: \quad P_0 = \frac{VP + VN}{N} \; y \; P_e = \frac{a * c + b * d}{N^2}$$

$$siendo: a = VP + FP, b = FN + VN, c = VP + FN, d = FP + VN$$

Donde: P_o, es la proporción de aciertos. P_e, es la proporción de aciertos esperados bajo la hipótesis de independencia entre las dos variables. Para la valoración del valor de k, se utiliza la escala propuesta por Landis et al. (1977). En el Tabla 3 se muestra el grado de concordancia que se define según el valor del índice de Kappa.

Tabla 3

Índice por grado de concordancia

Índice de Kappa	Grado de concordancia
0,00	sin acuerdo
< 0,00 - 0,20]	insignificante
[0,21 - 0,40]	Discreto
[0,41 - 0,60]	Moderado
[0,61 - 0,80]	Sustancial
[0,81 - 1,00]	casi perfecto

2.4 Técnicas de minería de datos

Las técnicas de minería de datos son clasificadas en dos grandes grupos: supervisadas y no supervisadas. Las técnicas supervisadas se aplican cuando el conjunto de entrenamiento existe una variable dependiente; que en el caso sea cualitativa correspondería a un problema de clasificación y si es cuantitativa a un problema de predicción. Las técnicas no supervisadas, se aplican cuando en el conjunto de datos todas las variables son independientes o predictoras, por lo que no existe variable dependiente y en este caso se asocia a los problemas de agrupación y asociación.

2.4.1 Técnicas de minería de datos supervisadas

La mayoría de las TMD supervisadas que existen son aplicadas para la tarea o problema de la clasificación. El aprendizaje supervisado hace referencia que las observaciones del conjunto de entrenamiento están previamente agrupadas según el atributo clase y un conjunto de atributos predictores. Las técnicas de clasificación pueden resolver muchos problemas en diferentes campos: medicina, industria, negocio, educación, ciencias, etc. Kiruthika et al. (2015) indican que la clasificación es el proceso de encontrar un modelo o función que describa datos etiquetados con alguna clase, con el propósito de ser utilizado para predecir datos nuevos de una clase que es desconocida.

Formulación de las técnicas supervisadas de clasificación

Una base de datos (D), con p atributos y n observaciones, puede ser expresada como elementos de una matriz de datos X_{ij}; $i=1,2,\ldots,n$ y $j=1,2,\ldots,p$, que representa a la i-ésima observación (instancia) y j-ésima atributo (variable). Además, se tiene un conjunto de m clases. Sea $X_j^k; k=1,2,\ldots,q$ el j-ésimo atributo cualitativo independiente con su k-ésimo valor y con "q" posibles valores. La variable dependiente para un problema de clasificación, se define como cualitativa Y^g; $g=1,2,\ldots,m$, que corresponde la g-ésima clase. La variable dependiente Y^g, se denomina variable o atributo clase, conteniendo m clase o categorías. El problema de clasificación es encontrar una función (clasificador) $f:D\rightarrow C$, tal que cada observación sea asignada a una clase C_g.

Métodos para la validación de clasificadores

Son métodos que permiten evaluar la performance o rendimiento de las TMD supervisada para la clasificación, con la finalidad de realizar una evaluación honesta sobre su bondad de ajuste al conjunto de datos de entrenamiento. Los métodos consisten en dividir el conjunto total de observaciones en tres subconjuntos: conjunto de entrenamiento (usado para el proceso de aprendizaje o estimación del clasificador), conjunto de validación (usado para validar el clasificador) y conjunto de prueba (usado para la inferencia de nuevas observaciones); el último conjunto es opcional y generalmente se usa datos no incluidos en la base de datos. Existen varios métodos para validación de los clasificadores:

- **Método de Validación Cruzada (Croos-Validation).** Es el más utilizado en el aprendizaje supervisado, consiste en dividir aleatoriamente el conjunto de entrenamiento *D* en k subconjuntos (k-folds) mutuamente excluyentes $\{D_1, D_2,\ldots,D_k\}$ de similar tamaño.

El proceso de validación cruzada es repetido durante k iteraciones, de tal manera que en cada iteración el clasificador usa un subconjunto para la validación D_v y es entrenado con los k-1 subconjuntos $(D-D_V)$, el error de clasificación se calcula como la media aritmética de los errores de cada iteración. Un caso particular de la validación cruzada dejar-uno-afuera (Leave-one-out), implica que en cada iteración se tenga un solo dato de prueba y el resto para entrenamiento, el error se calcula como el promedio de los errores cometidos.

- **Método de retención (Holt-Out).** Este método particiona aleatoriamente el conjunto de datos *D* en dos conjuntos mutuamente excluyentes: conjunto de entrenamiento (D_E) y conjunto de validación (D_V). El tamaño del D_E generalmente es mayor al D_V en proporciones 2/3 y 1/3, 4/5 y 1/5, etc. respectivamente. Los elementos del D_E suelen obtenerse mediante muestreo sin reemplazo de todo el conjunto de datos, mientras que el conjunto D_V lo conforma las observaciones restantes que no pertenecen al D_E. Suele ser aplicado en un conjunto de datos grandes.

Las principales TMD para el aprendizaje supervisado para el problema de clasificación, se describen a continuación:

1. TMD de regresión logística

La Regresión Logística (RL), es una técnica que permite estudiar la dependencia funcional entre una variable dependiente categórica dicotómica (dos clases) o politómica (más de dos clases) y un conjunto de variables independientes o predictoras que pueden ser cuantitativas o categóricas. La RL como TMD, se aplica a los problemas de clasificación o predicción.

Formulación del modelo logístico

Los modelos de regresión logística se clasifican según el número de categorías o clases que tenga la variable dependiente; pudiendo ser binario (dos categorías) o politómicos (más de dos categorías). Así, los modelos logísticos se pueden clasificar en:

$$Modelos\ Logísti\cos\begin{cases}\text{Re}spuesta\ Di\cot ómica: Y=0\ ó\ 1\begin{cases}\text{Re}gresión\ Logística\ Binaria\\ (Logit\ Binario)\end{cases}\\ \text{Re}spuesta\ Politómica: Y=1,2,..J\begin{cases}\text{Re}gresión\ Logística\ No\min al\\ (Logit\ No\min al)\\ \text{Re}gresión\ Logística\ Ordinal\\ (Logit\ Ordinal)\end{cases}\end{cases}$$

- **Regresión logística binaria.** Son aquellos donde la variable dependiente Y puede tomar solo dos valores posibles 1 o 0 (Y=1: éxito; Y=0: fracaso); generalmente Y=1 es el ***evento de interés*** y por lo cual las observaciones están clasificadas en dos categorías o grupos. El modelo de regresión de respuesta binaria, trata de explicar la variable respuesta binaria Y en términos de que tan probable suceda el evento de interés (Y=1), en función de un conjunto de predictores $X_1, X_2, \ldots, X_p$. Modeliza la probabilidad Y=1 usando la función de enlace Logit para relacionarla con el predictor lineal. Entonces el modelo logístico binario se expresa por:

$$E[Y / X] = P(Y = 1) = \pi = \frac{\exp(\beta_0 + \beta_1 X_1 + \ldots + \beta_p X_p)}{1 + \exp(\beta_0 + \beta_1 X_1 + \ldots + \beta_p X_p)}$$

- **Modelo logístico politómico.** Son modelos donde la variable respuesta cualitativa presenta más de dos categorías, por lo cual se debe seleccionar una de ellas como categoría de referencia. El modelo general logístico politómico con J categorías se expresa por:

$$\pi_1 = P(Y = 1 / X_1, \ldots, X_p) = \frac{1}{1 + \sum_{j=2}^{J} \exp(x_j' \beta_j)} \quad (j = 1 \, categoría \, referencial)$$

$$\pi_j = P(Y_j = J / X_1, \ldots, X_p) = \frac{\exp(x_j' \beta_j)}{1 + \sum_{j=2}^{J} \exp(x_j' \beta_j)} \quad j = 2, \ldots, J$$

Con $\pi_1 + \pi_2 + \ldots + \pi_J = 1$

2. Árbol de decisión

Un Árbol de decisión (AD) es una técnica para el aprendizaje inductivo supervisado, fácil de implementar y a su vez de los más poderosos para problemas de clasificación. La estructura de un AD corresponde a un grafo acíclico dirigido, compuesto por un nodo llamado raíz, un conjunto de nodos internos que se les asocia una variable y cuyos arcos representan los diferentes valores que toma la variable brindando la interconexión entre los nodos internos y los nodos terminales, llamados hojas del árbol que están etiquetadas con algún valor de la variable clase. El conocimiento se representa en el árbol a través de un conjunto de reglas (condiciones) organizadas en su estructura jerárquica, y que son determinadas con el recorrido desde el nodo raíz hasta las hojas.

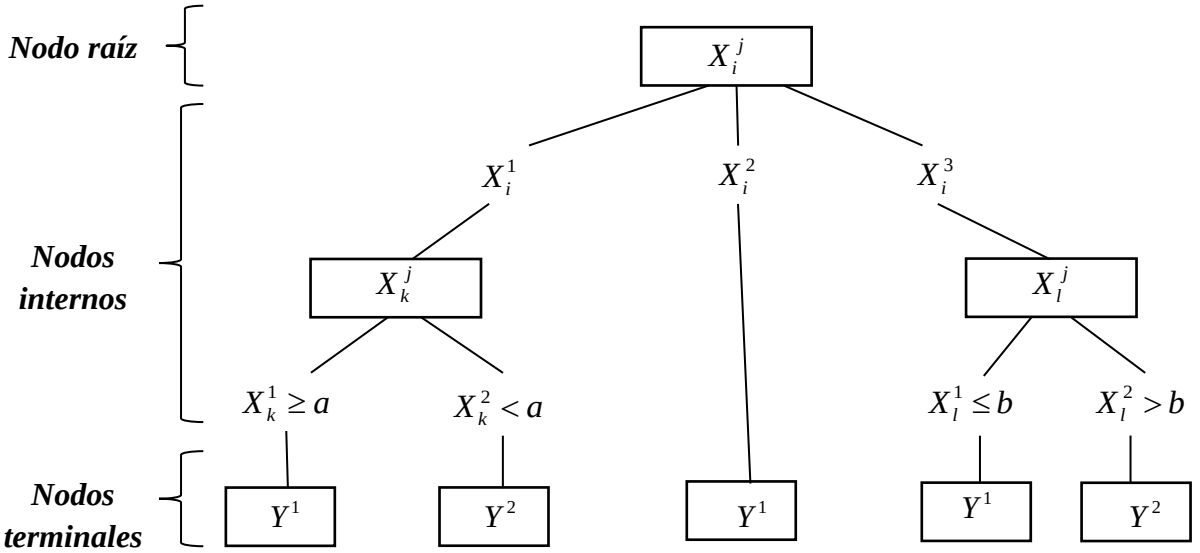
Formulación del árbol de clasificación

La mayoría de los algoritmos están agrupados en la llamada familia de **Algoritmos TDIDT** (Top-Down Induction on Decision Trees). Los algoritmos para construir los árboles de

decisión se fundamentan en la estrategia “divide y vencerás” y se realiza recursivamente en forma descendentemente. La estructura de un árbol de decisión, se basa en un árbol con una ***raíz***; donde cada ***nodo*** representa un atributo y sus ***ramas*** las divisiones del atributo de sus diferentes valores; las ***hojas*** son el punto terminal donde se alcanza una decisión (***Clase objetivo***). En la Figura 5 se presenta el grafo de un árbol de clasificación con las clases de nodos.

Figura 5

Estructura de un árbol de clasificación



El proceso de inferencia o el proceso de predicción del modelo para clasificar nuevas observaciones usando el árbol construido (inducido) para algún valor del atributo clase, consiste en ir corriendo de acuerdo a los valores de la nueva observación, desde el nodo raíz hasta una hoja (nodo terminal), donde se determina la pertenencia de la observación a alguna de las clases.

Para el aprendizaje supervisado de un AR, existe una variedad de métricas que se aplica según el algoritmo, cuya finalidad es seleccionar el atributo que se asignará para dividir un nodo. Entre las métricas usadas por los algoritmos se tiene:

- **Entropía.** Es una medida del grado de incertidumbre (información que se proporciona). La entropía se calcula respecto a los valores de la clase objetivo y se utiliza como una

medida para decidir que atributo se elige en la división de los nodos. La Entropía se expresa por:

$$H = -\sum_k p_k Log_2 p_k$$

$Donde$:

$p_k = \Pr obabilidad\, de\, que\, una\, observación\, esté\, en\, la\, clase\, objetivo\ k$

$Log_2 = Logaritmo\, en\, base\, 2$

Entropía para la variable con respecto a la clase objetivo: $H(X_j^k) = -\sum_k p_h^k Log_2 p_h^k$

Entropía media ponderada para una variable: $H_p(X_j^k) = -\sum_k p_h^k H(X_j^k)$,.

Entropía para la variable clase: $H(Y^g) = -\sum_g p^g Log_2 p^g$

- **Ganancia de Información (GI).** Mide la ganancia que se obtiene por usar la variable. Se escoge la variable con el valor más alto, indicando que reduce la incertidumbre de la clasificación de la clase objetivo. $GI(X_j^k) = H(Y^g) - H_p(X_j^k)$
- **La Razón de la Ganancia de información (RG).** Se define como el cociente entre la ganancia de información y su entropía, escogiendo la variable con la mayor ganancia de información. La RG se define por: $GR(X_j^k) = \frac{GI(X_j^k)}{H(X_j^k)}$
- **Índice de Gini.** Usado en el algoritmo CART para medir el grado de impureza

El proceso de aprendizaje con los árboles de clasificación, se deben tener en cuenta parámetros:

- Factor de Confianza (FC). Es un criterio para el podado del árbol. Generalmente se considera 25% (corresponde a un z=0.69) o menor. Cuando FC disminuye, se tiende a obtener un árbol más pequeño.
- Poda. Para evitar el sobre ajuste del árbol, se aplica la poda si es necesario.
- Tamaño del árbol. Número total de nodos
- Número de hojas. Número total de nodos hoja
- Número de reglas. Número de reglas que se desea obtener.

3. Redes bayesianas

Las Redes Bayesianas (RB), son modelos gráficos probabilísticos que representan la función de distribución conjunta de un conjunto de variables, se fundamentan en la teoría de la

probabilidad y el teorema de Bayes. Las RB se aplican como TMD para el problema de clasificación (aprendizaje supervisado). La RB es un grafo que representa las independencias y dependencias probabilísticas entre un conjunto de variables, compuesto por nodos que representan las variables aleatorias y arcos asociados a los valores de la variable, que representan las dependencias entre dichos nodos.

Formulación de las redes bayesianas

La estructura de una Red Bayesiana para la clasificación, está compuesta por un nodo raíz que corresponde a la variable dependiente Y^g (variable clase) y un conjunto de variables independientes $X=(X_1, X_2,\ldots,X_p)$ que están representadas en los nodos de la red. Un clasificador RB, asume que Y^g es el nodo padre del conjunto de variables explicativas. La inducción de un árbol, consiste en el proceso de su construcción a partir de un conjunto de entrenamiento. Considerando un conjunto de entrenamiento D, con N instancias (observaciones) agrupadas en m clases para la variable dependiente Y^g; g=1,,2…,m. Se define N^g como el número de instancias que corresponde a la clase g, N_h que contiene el nodo h, N en el nodo raíz y N_h^g para el nodo h y clase g, tal que $\sum N_h^g = N_h$ y para el nodo h y la clase g, por $p_h^g = \frac{N_h^g}{N_h}$.

El obtener una red bayesiana a partir de un conjunto de datos, es un proceso de aprendizaje bayesiano que se divide en dos etapas:

- **Aprendizaje estructural.** Consiste en obtener la estructura de la red bayesiana, que muestre las relaciones de dependencia e independencia entre las variables. Las técnicas para el aprendizaje estructural dependen del tipo de estructura de la red: árbol, poliárboles o redes multiconectadas.
- **Aprendizaje paramétrico.** Dada una estructura de la red, se requiere calcular las probabilidades a priori y condicionales requeridas en la red bayesiana. El aprendizaje con las RB se deben calcular las siguientes medidas de probabilidades:
  - **Probabilidad a priori.** Se asocian a los valores de la variable clase $(Y^g; g = 1,2,\ldots,m)$. Se define la probabilidad a priori para clase g: $P(Y^g = g) = p^g = \frac{N^g}{N}$.

- **Probabilidades condicionales**. Son las probabilidades condicionales para cada variable predictora j para su k–ésimo valor (X_j^k) dada la variable clase Y^g: $P(X_j^k / Y^g) = \frac{N_h^k}{N_h}$.

- **Probabilidad total.** Es la probabilidad que para la variable predictora X_j^k ocurra su k-ésimo valor: $P(X_j^k) = \sum_{k=1}^{m} P(Y^g) P(X_j^k / Y^g)$.

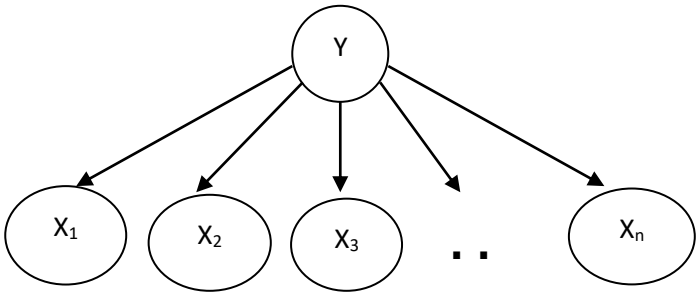
- **Probabilidad a posteriori.** Aplicando el Teorema de Bayes, se quiere predecir la clasificación de una nueva observación X_0^k a algún valor de la variable clase C^g. La probabilidad a posteriori : $P(C^g / X_0^k) = \prod_{j=1}^{p} P(Y^g) P(X_j^k / C_g); \quad g = 1,2,...,m$.

Existen una variedad de algoritmos para construir redes bayesianas, que dependen de la estructura de la red bayesiana.

- **Clasificador Naive Bayes (NB).** El Clasificador Naive Bayes es el más simple para la clasificación supervisada con Redes Bayesianas y tiene una estructura de red fija. El fundamento principal de clasificador NB es la suposición de la independencia condicional de las variables explicativas dada la variable clase. Esta suposición de independencia entre las variables es representada por una estructura de la red fija y el grafo acíclico dirigido, contiene un único nodo raíz (variable clase) y en la que todos los atributos (variables explicativas) son nodos hijos. La representación de la estructura de una Red Bayesiana Naive Bayes se muestra en la Figura 6.

Figura 6

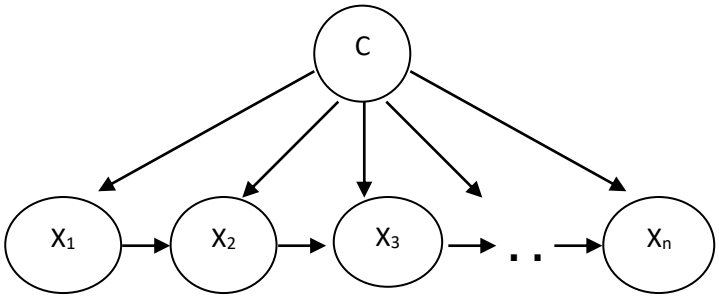
Estructura del clasificador de una Red Bayesiana Naive Bayes



- **Clasificador Naive Bayes Aumentado a Árbol (TAN).** El clasificador Naive Bayes supone una independencia entre las variables predictoras, suposición no tan realista. El clasificador de Naive Bayes Aumentado a Árbol, incluye la posibilidad de modelar las relaciones de dependencia entre las variables predictoras formando así una estructura de árbol. En la estructura de árbol aumentado un arco del nodo X_i hacía X_j, indica que la influencia de X_i en la asignación de la variable clase C también depende del valor de X_j. En la figura 4 se muestra la estructura de una Red Bayesiana TAN en la que la variable clase no tiene padres y las variables predictoras tienen como padre a la variable clase C y a lo más alguna otra variable (cada variable puede tener un arco dirigido hacía él). La representación de la estructura de una Red Bayesiana TAN se muestra en la Figura 7.

Figura 7

Estructura de una Red Bayesiana Naive Bayes aumentada a árbol (TAN)



2.4.2 Técnicas de minería de datos multiclasificadores

Según Segrera et al. (2006) mencionan que las técnicas de minería de datos multiclasificadores o métodos para multiclasificadores, permiten combinar o ensamblar una colección de clasificadores diferentes que se han ajustado a un conjunto de datos. Los métodos de ensamble se basan en combinar un conjunto de clasificadores para crear uno nuevo. Para que un multiclasificador sea efectivo, una condición necesaria y suficiente que se debe de cumplir es que los clasificadores individuales sean precisos y diversos. Un clasificador se considera preciso si su error es inferior a 0.5 y dos clasificadores individuales son diversos cuando sus errores de salida no son correlacionados.

Los métodos de multiclasificadores se dividen en dos grupos: los métodos de ensamble y los métodos híbridos. Así mismo, los métodos de multiclasificadores se diferencian en: el número de clasificadores individuales acoplados, el tipo de cada clasificador (redes neuronales,

árboles de decisión, vecino más cercano, etc.), las características de los subconjuntos usados por cada clasificador del conjunto, la agregación de las decisiones particulares (voto mayoritario, asignación de pesos, subespacio de mejor comportamiento, funciones de promedio, máximo, mínimo, producto, etc.) y el tamaño y la naturaleza de los conjuntos de datos de entrenamiento para los clasificadores. La evaluación y comparación entre los distintos multiclasificadores se establece a través de indicadores de rendimiento que incluyen el grado de generalización, aprendizaje, clasificación correcta e incorrecta y el tiempo real de ejecución.

Estrategias de combinación

Son las estrategias o técnicas que se aplican para fusionar las salidas individuales de cada clasificador en una sola. Entre las principales se tiene:

- **Voto mayoritario simple.** Esta técnica se basa en asignar un voto a cada clasificador, eligiendo el clasificador final el de mayor votación.
- **Voto mayoritario por peso.** En esta técnica el voto del clasificador se basa en asignarle un peso que influye en la decisión final. Una forma es entrenar a cada clasificador con un conjunto de prueba y asigna su peso de acuerdo a su rendimiento. La clase con mayor peso es la que se emite como resultado final.
- **Reglas basadas en el enfoque de Bayes.** La fusión se realiza como combinación de los clasificadores usando la estadística bayesiana.

Existen tres métodos para aplicar los multiclasificadores:

- **Método Bagging.** El método de ensamblaje Bagging fue propuesto por (Breiman, 1996) y está basado en los conceptos y beneficios de bootstrapping y agregación, de allí su nombre del método (Bootstrap AGGregatING). El método de Boostrapping permite la generación de muestras aleatorias uniformes con reemplazo (conjuntos de entrenamientos aleatorios) a partir del conjunto de entrenamientos; creando tantas muestras bootstrap como clasificadores existan de tal manera que cada clasificador se entrena con una réplica bootstrap. En el Bagging cada clasificador individual aprende con una muestra Bootstraping y la decisión final se utiliza el mayor voto mayoritario, para clasificar un ejemplo nuevo se predice y se selecciona la clase.

- **Método Boosting.** Freund y Schapire (1996) indican que el mecanismo se basa en la asignación de un peso a cada ejemplo del conjunto de entrenamiento. En cada iteración,

este método aprende un modelo que minimiza la suma de los pesos de los ejemplos clasificados erróneamente. Según Hernández et al. (2004), los errores de cada iteración se utilizan para actualizar los pesos de los ejemplos del conjunto de entrenamiento de manera que se incremente el peso de los ejemplos errados y se reduce el peso de los ejemplos acertados. El algoritmo AdaBoost es la variante de Boosting más conocida y usada. En un ciclo se aprende un modelo a través de la evidencia ponderada, se estima el error del modelo y dependiendo del valor del error se detiene el algoritmo o se continua el proceso repitiendo el ciclo, de ser así se actualiza los pesos de los ejemplos clasificados de forma acertada, se almacena el modelo y se efectúa la normalización del peso de todos los ejemplos (véase el algoritmo en la figura 11). A diferencia del método Bagging, este método considera como criterio de parada el valor del error. Es decir, mientras que en Bagging se predetermina el número de iteraciones para concluir el método, Boosting considera que si el error es igual o superior a 0.5 o no se puede disminuir (igual a 0), se detiene el algoritmo. Además, garantiza que se incrementen los pesos de los ejemplos mal clasificados para que en la siguiente iteración tengan mayor preferencia.

- **Métodos híbridos.** Los métodos híbridos a diferencia de los métodos de ensamble, combinan diferentes algoritmos de aprendizaje. Entre los más utilizados son: Acumulación (Stacking) y Castada (Cascading).

III. MATERIALES Y METODOS

La metodología propuesta para el desarrollo de la tesis es la siguiente:

3.1 Materiales

Los materiales y equipos que se usarán para realizar la presente tesis son los siguientes:

1. Una computadora personal Intel® Core ™ i7. CPU 3.5 GHz. RAM de 4.00 GB.
2. Una impresora inyectora HP.
3. El programa R. Ver. 4.0.2. En el Tabla 4, se presentan las librerías usadas:

Tabla 4

Librerías usadas del programa R

Librerías	Descripción
tidyverse, dplyr	Para el manejo de los datos.
ggplot2,gridExtra, factoextra, patchwork	Para elaborar gráficos.
ROSE	Para el balanceo de los datos.
FSelector	Para aplicar los métodos de selección de atributos por filtrado y Wrapper.
Caret, rpart, nnet, C50, bnclassify, randomForest	Para aplicar las técnicas de minería de datos (árbol de clasificación CART, regresión logística nominal, árbol de clasificación C5.0, redes bayesianas Naive Bayes y árbol de clasificación como multiclasificador Bagging).

3.2 Métodos

La metodología propuesta para la presente tesis es:

3.2.1 Tipo de investigación

La investigación es de tipo descriptivo y predictivo. Se describen los métodos de filtrado y Wrapper para la selección de atributos con las técnicas de minería de datos supervisadas, evaluando su capacidad predictiva de los atributos seleccionados con el atributo clase usando medidas tales como tasa de buena clasificación, curvas ROC, AUC, etc.

Diseño de la investigación

El diseño de la investigación es no experimental de corte transversal, porque no hay manipulación de variables independientes y se usa la Encuesta Nacional de Satisfacción de Usuarios de Salud (SUSALUD) realizada por el Instituto Nacional de Estadística e Informática se aplicó en un período de tiempo del 2015.

Hipótesis de la investigación

Los métodos de selección de atributos filtrado y Wrapper para el aprendizaje supervisado, aumentará el porcentaje de la tasa de buena clasificación cuando se realiza la selección de subconjuntos de atributos relevantes y la eliminación de atributos irrelevantes y redundantes.

3.2.2 Población y muestra

- **Población.** Todos los usuarios del sistema de salud en el 2015.
- **Muestra.** Encuesta Nacional de Satisfacción de Usuarios de Salud tomada por el INEI 2015.

3.2.3 Descripción de variables

En el estudio se usará los datos recolectados en la Encuesta Nacional de Satisfacción de Usuarios de Salud del 2015 realizada por el INEI en coordinación con SUSALUD (Superintendencia Nacional de Salud), cuyo objetivo es el de suministrar información estadística que permita evaluar el grado de satisfacción y las experiencias de los usuarios internos y externos de los servicios de salud para evaluar los servicios de atención de salud. Se considera del cuestionario el módulo: V. De la atención actual y las preguntas 25, 27 y 28 que se refiere a la calificación sobre el trato recibido por el personal: administrativo, no médico y médico, cada pregunta está referida a seis ítems de evaluación: Amabilidad y cortesía, Respeto, Interés/disposición por atender, Confianza y seguridad que le inspira, Vestuario y Claridad de la información y la pregunta 40 ¿Cómo calificaría su nivel de satisfacción?. En total se definen 18 atributos predictores (preguntas 25, 27 y 28 cada una con 6 ítems) y un atributo clase (pregunta 40).

Atributo clase (variable dependiente):

- Respecto a la atención recibida el día de hoy en este establecimiento:
  Y = ¿Cómo calificaría su nivel de satisfacción?.: Muy satisfecho/a (5), Satisfecho/a (4), Ni satisfecho/a / Ni insatisfecho/a (3), Insatisfecho/a (2) Muy insatisfecho/a (1).

Atributos predictores (variables independientes):

Todas las variables tienen las categorías: Muy bueno (5), Bueno (4), Ni Bueno/Ni Malo/a (3), Malo (2), Muy malos, No sabe/No responde (1).

- ¿Cómo calificaría Ud. el trato recibido por el personal administrativo, en cuanto a:?
  X1=Amabilidad y cortesía
  X2=Respeto
  X3=Interés/disposición por atender
  X4=Confianza y seguridad que le inspira
  X5=Vestuario (uniforme)
  X6=Claridad de la información
- ¿Cómo calificaría Ud. el trato que ha recibido por el personal no médico, en cuanto a:?
  X7=Amabilidad y cortesía
  X8=Respecto
  X9=Interés/disposición por atender
  X10=Confianza y seguridad que le inspira
  X11=Vestuario (uniforme)
  X12=Claridad de la información
- ¿Cómo calificaría Ud. el trato que ha recibido el día de hoy por el médico tratante, en cuanto a:?
  X13=Amabilidad y cortesía
  X14=Respecto
  X15=Interés/disposición por atender
  X16=Confianza y seguridad que le inspira
  X17=Vestuario (uniforme)
  X18=Claridad de la información

3.2.4 Procedimiento de análisis de datos

Para el realizar el análisis de datos se propone los siguientes pasos:

1. **Recopilación de datos.**

Se recopila los datos de la Encuesta Nacional de Satisfacción de Usuarios de Salud tomada por el INEI en el año 2015. Se toman del cuestionario las preguntas 25, 27, 28 (cada una con 6 ítems) y 40, siendo un total de 19 preguntas.

2. **Pre procesamiento de datos.**

Se aplicaron técnicas de pre procesamiento con la finalidad de hacer una limpieza de datos, se tomó en cuenta el manejo de los datos atípicos y faltantes en los conjuntos de datos y el balanceo de los datos. Así mismo, se recodifican las categorías de las variables.

- **Manejo de datos Atípicos.** Se aplican técnicas para la detección de datos atípicos u outlinear.
- **Manejo de datos faltantes.** Se aplican técnicas para identificar y manejar los datos missing. Tales como listwise o técnicas de imputación
- **Recodificación de datos.** Se aplica procedimientos para agrupar o recodificar las categorías no significativas de los atributos.
- **Balanceo de datos.** En el aprendizaje supervisado para la clasificación, el problema de desbalanceo de los datos es muy recurrente, dónde el atributo clase presenta una clase minoritaria y la otra mayoritaria; por lo cual se afecta las medidas de precisión de la TMD sesgando la clase mayoritaria. Se propone aplicar el enfoque de sobre muestreo, que consiste en aumentar con observaciones la clase minoritaria para conseguir una base de datos balanceada.

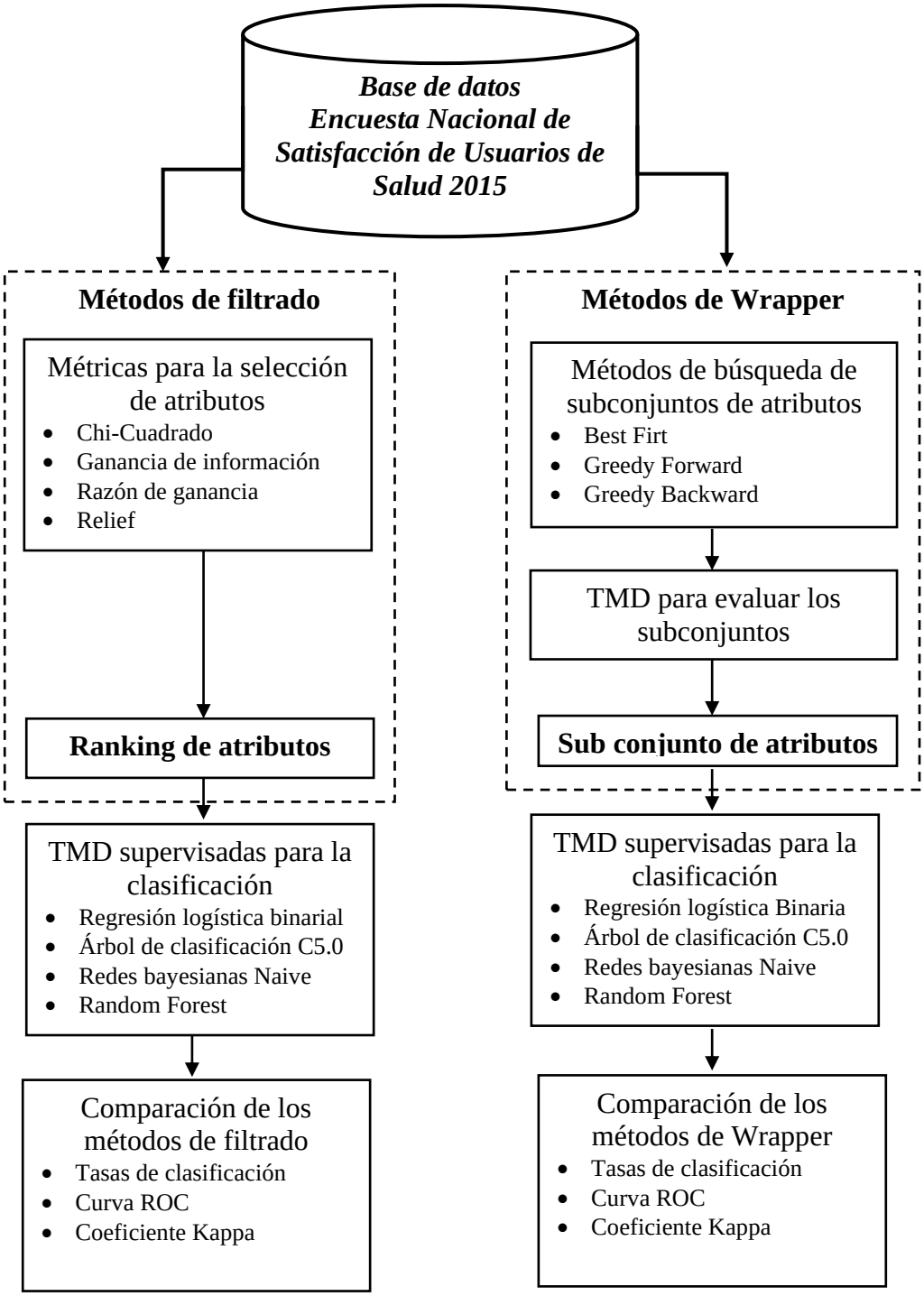
3. **Proceso del análisis estadístico.**

Se aplican los métodos de selección de atributos al conjunto de entrenamiento. En la Figura 8, se presenta el proceso propuesto para aplicar los dos enfoques para la selección de tributos en el aprendizaje supervisado para el problema de clasificación. La diferencia principal del método de filtrado es la independencia que tienen con la TMD a ser aplicada, mientras que el método Wrapper si es dependiente. Para los métodos de filtrado se propone aplicar cuatro métricas que permiten medir la relación con el atributo clase para realizar el filtrado, dando como resultado un ranking de atributos. Para los métodos Wrapper, se consideran dos métodos de búsqueda para seleccionar un sub conjunto de atributos. Con la finalidad de

validar la calidad de la selección de atributos, se aplican cuatro TMD y se calcularán medidas para evaluar su capacidad predictiva

Figura 8

Proceso para la aplicación de los métodos de selección de atributos



IV. RESULTADOS Y DISCUSIÓN

En la presente tesis se aplican los métodos de selección de atributos a la Encuesta Nacional de Satisfacción de Usuarios de Salud 2015. Se comparan los métodos de selección de atributos por filtrado y por Wrapper. Se consideran cuatro métricas para la selección de atributos por filtrado: Chi-Cuadrado, ganancia de información, razón de ganacia y Relief; y cuatro para el método de Wrapper: Best-Firt, Greedy Forward, Greedy Backward y Hill climbing. Las métricas de filtrado y Wrapper son evaluadas aplicando cuatro TMD: regresión logística nominal, árbol de decisión C4.5, Naive Bayes y el multiclasificador random Forest y a través de su capacidad predictiva. A continuación, se presenta los resultados siguiendo el proceso metodológico propuesto para la selección de atributos, con lo cual se espera satisfacer los objeticos de esta investigación.

4.1 Recopilación de los datos

Se accede al Portal del INEI para recopilar la base de datos correspondiente a la Encuesta Nacional de Satisfacción de Usuarios de Salud 2015. Los datos de la encuesta se almacenan en archivos en formato .sav (SPSS) que representan módulos de temas de información. Básicamente se usa el módulo: 447-Módulo550 (02_C1_CAPITULOS.sav). Además, con la finalidad de tener una mayor homogeneidad de los datos, se realizó un filtrado de los encuestados considerando sólo a los usuarios con resultado de la encuesta completa (C1RESULTFINAL=1), con la institución de ESSALUD (INSTITUCION=2) y que asistieron a la especialidad de Medicina General (C1CONSULTORIO=12). Así mismo, se eliminaron las observaciones con respuestas en los atributos como No sabe/No responde. Como resultado se obtuvo la base de datos inicialmente para el estudio con 2456 usuarios. Para el estudio se consideraron18 variables, las cuales están agrupadas por tres categorías según el personal que calificó el usuario: administrativo, no médico y médico.

Con la finalidad de homogeneizar los atributos se procedió a recategorizar las categorías que se presenta en la encuesta de satisfacción. Para definir el atributo clase, las categorías Insatisfecho y Muy insatisfecho en Insatisfecho; las categorías Satisfecho y Muy satisfecho en Satisfecho. Para el resto de atributos, se agruparon las categorías de la encuesta Malo y Muy malo en Malo, Ni malo/Ni bueno en Regular y Muy bueno y bueno en Bueno. En el Tabla 5, se presenta el resultado de la recategorización con los nuevos valores asignados para el atributo clase y los atributos predictores. En el atributo clase, se definen dos categorías**:**

Insatisfecho y Satisfecho, permite evaluar la satisfacción de la atención recibida de los usuarios por los tres servicios: administrativo, no médico y médico. En los atributos predictores se han considerado tres categorías: Malo, Regular y Bueno.

Tabla 5

Recategorización de los atributos clase y predictores

Atributo Clase (Y)	Atributos predictores (X)
Insatisfecho (Insatisfecho y Muy insatisfecho)	Malo (Malo y Muy malo)
Satisfecho (Muy satisfecho y Satisfecho)	Regular (Ni Malo/Ni Bueno)
	Bueno (Muy bueno y Bueno)

4.2 Pre procesamiento de datos

Se aplicaron técnicas para el pre procesamiento de datos conducentes a la limpieza (manejo de datos faltantes y datos atípicos) y el balanceo de datos, con la finalidad de conseguir una base de datos consistente y limpia para aplicar el proceso de selección de atributos.

4.2.1 Manejo de los datos faltantes

Por la naturaleza de variables cualitativas en el estudio, no se aplicaron ningún procedimiento para el manejo de datos atípicos. En el caso de los datos faltantes se aplicó el procedimiento de Listwise, que consistió en considerar sólo los datos completos, eliminando las observaciones dónde en alguna variable exista un dato faltante. En el Tabla 6, se presenta la distribución de los encuestados como consecuencia del manejo de los datos faltantes respecto a la calificación de la atención recibida en el servicio de salud. Se obtuvo finalmente para el estudio una base de datos que considera a 768 usuarios encuestados, distribuidos con respecto a la calificación de la atención recibida con 93Insatisfechos y 675 Satisfechos.

Tabla 6.

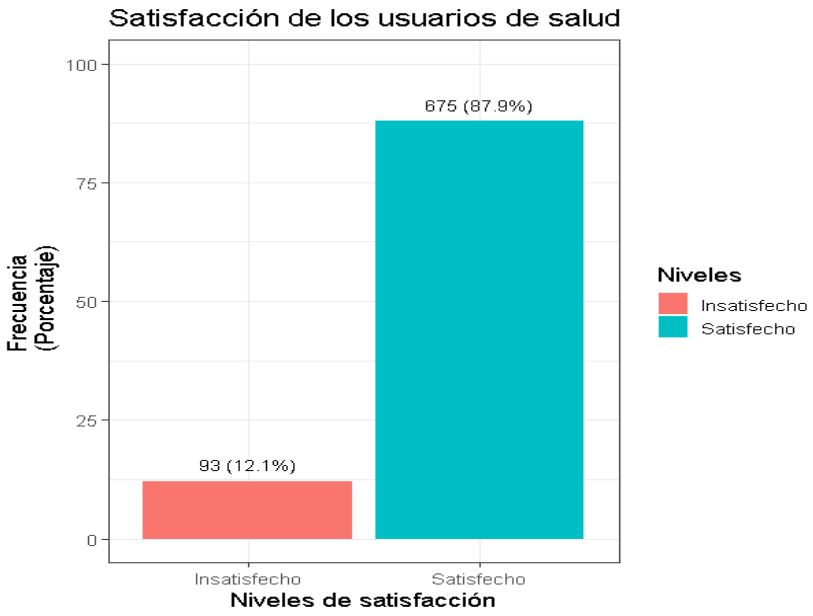
Distribución de la calificación de la atención recibida

Atributo clase	Número	Porcentaje
Insatisfecho	93	12,1
Satisfecho	675	87,9
Total	768	100,0

En la Figura 9, se muestra la distribución de los encuestados respecto al atributo clase. El 12,1% de los usuarios encuestados pertenecen a la clase Insatisfecha y el 87,9% a la clase Satisfecha. Esto indica que existe un desbalance en la base de datos.

Figura 9

Distribución de la calificación de la atención recibida en los servicios de salud



Nota: Los valores de la distribución son realizados con los datos originales

4.2.2 Balanceo de datos

Se observa que la base de datos está desbalanceada respecto a la calificación de la atención recibida de los usuarios, la clase minoritaria corresponde a los Insatisfechos (12,1%) y la mayoritaria a los Satisfechos (87,9%). Para dar solución al problema del desbalanceo, se aplica en enfoque híbrido con sub muestreo y sobre muestreo para conseguir una base de datos balanceada, antes de aplicar las TMD. Se usa el algoritmo ROSE con la respectiva librería en R. En el Tabla 7, se presenta la distribución de la calificación de la atención recibida como resultados de aplicar el algoritmo ROSE. Se muestra una base de datos balanceada para el atributo clase, con un 48,7% (374) de usuarios Insatisfechos y un 51,3% (394) de usuarios Satisfechos.

Tabla 7

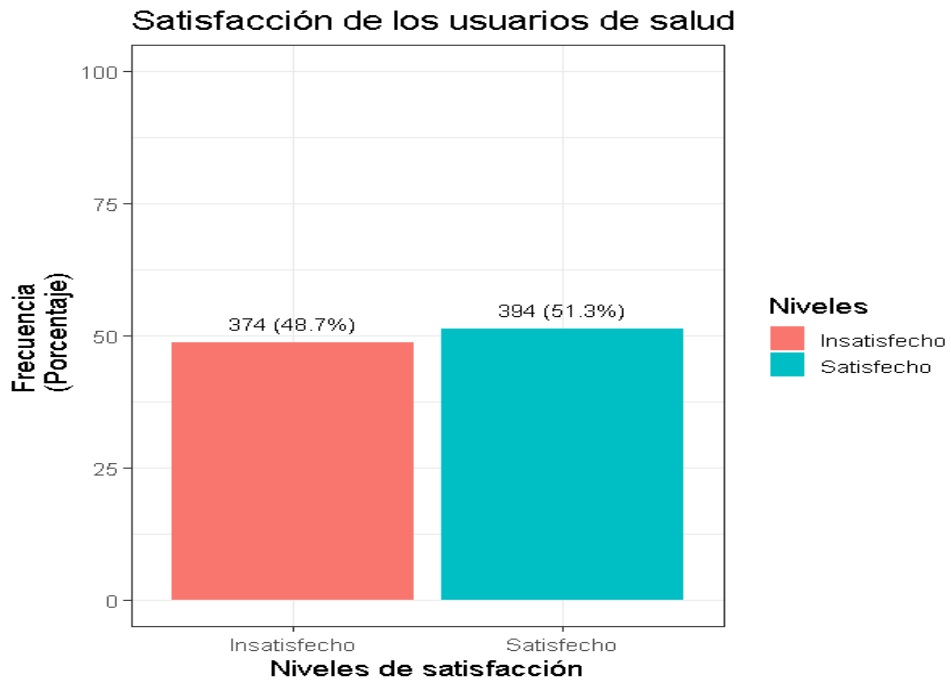
Distribución de la calificación de la atención recibida

Atributo clase	Número	Porcentaje
Insatisfecho	374	48,7
Satisfecho	394	51,3
Total	768	100,0

En la Figura 10, se muestra la distribución de los encuestados con una base de datos balanceada para el atributo clase. Se muestra un 48,7% de los usuarios con la clase Insatisfecha y un 51,3% con la clase Satisfecha. Por lo tanto, se tienen una base de datos balanceada.

Figura 10

Distribución de la calificación de la atención recibida en los servicios de salud



Nota: Los valores de la distribución son realizados con los datos balanceados

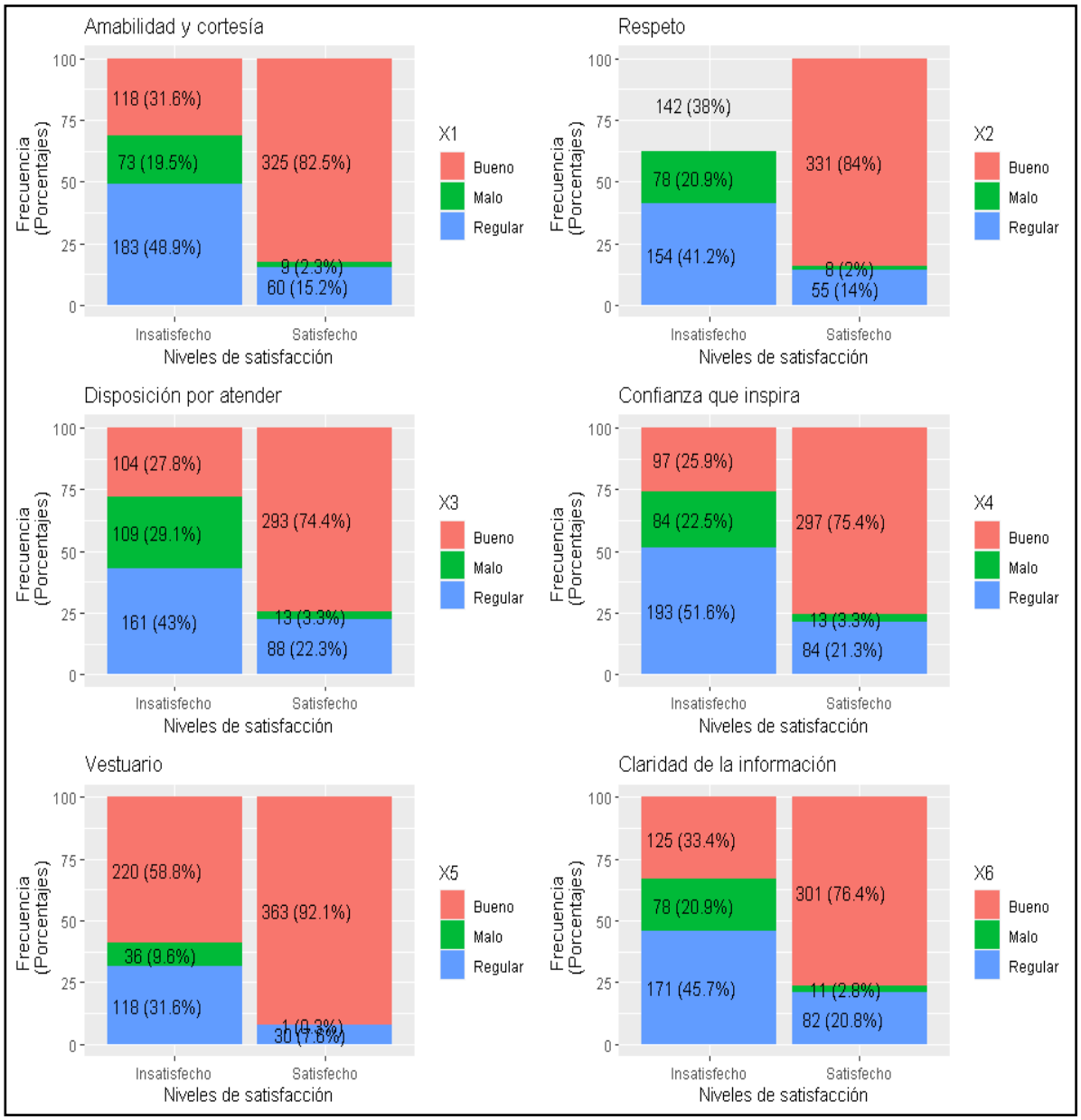
4.2.3 Análisis exploratorio de datos

Se obtienen gráficos para cada atributo predictor, mostrando la distribución de la calificación de los usuarios sobre la atención recibida en los servicios de salud por el personal, agrupados en tres servicios: administrativo, no médico y médico. En la Figura 11, se muestra la

distribución de la calificación de la atención recibida a los usuarios en los servicios de salud con respecto al personal administrativo en seis aspectos evaluados. Se observa que los usuarios que manifestaron estar insatisfechos, han califican la atención recibida con Regular en mayor porcentaje con respecto a la amabilidad, respecto, atención, confianza, vestuario e información, mientras que los usuarios que están satisfechos lo han calificado como Bueno con el mayor porcentaje.

Figura 11

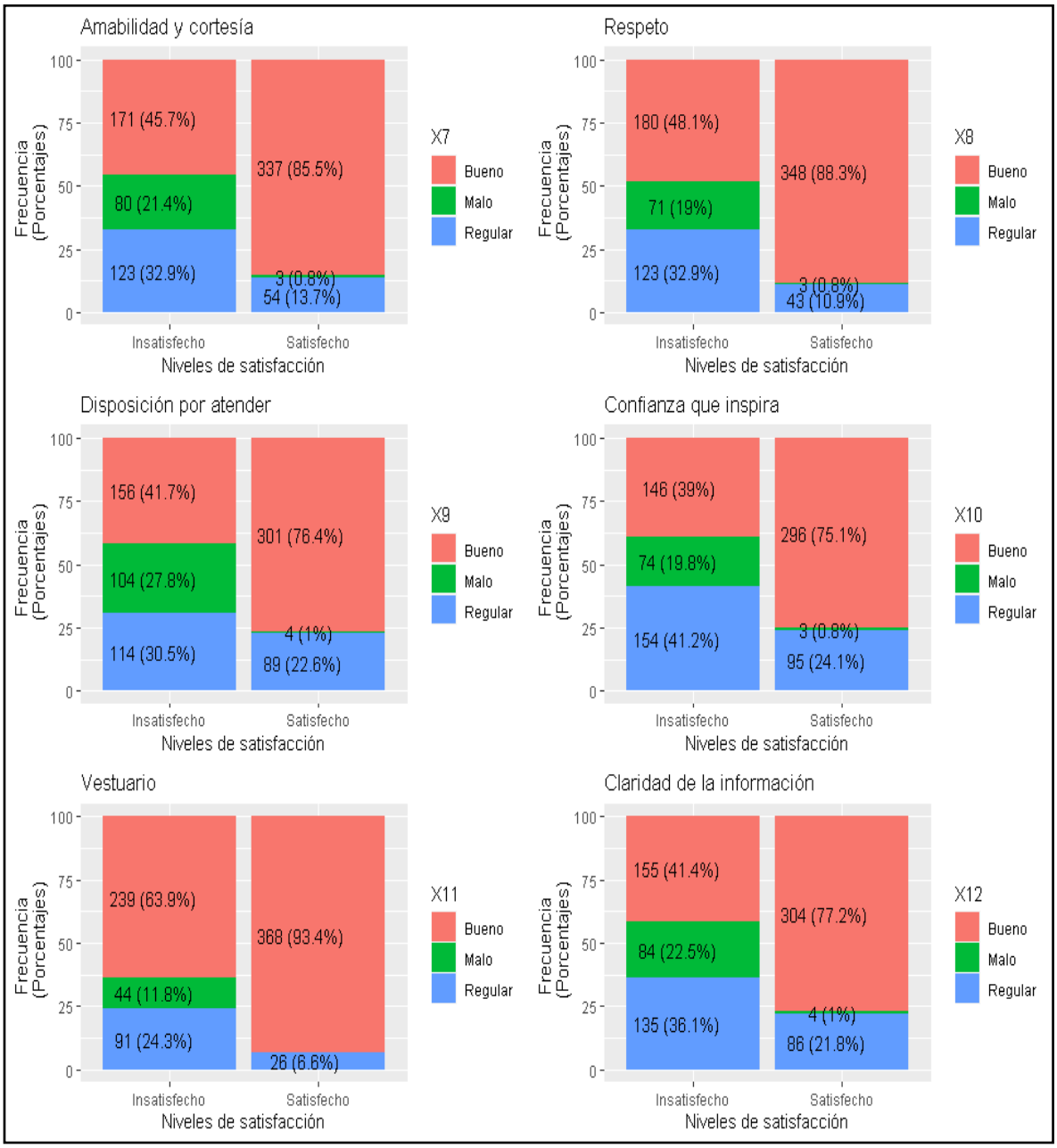
Distribución de la calificación de la atención recibida de parte del personal administrativo



En la Figura 12, se muestra la distribución de la calificación de la atención recibida a los usuarios en los servicios de salud con respecto al personal no médico en seis aspectos evaluados. Se observa que los usuarios que manifestaron estar insatisfechos, han califican la atención recibida con Bueno en mayor porcentaje con respecto a la amabilidad, respecto, atención, confianza, vestuario e información, mientras que los usuarios que están satisfechos lo han calificado como Bueno con el mayor porcentaje.

Figura 12

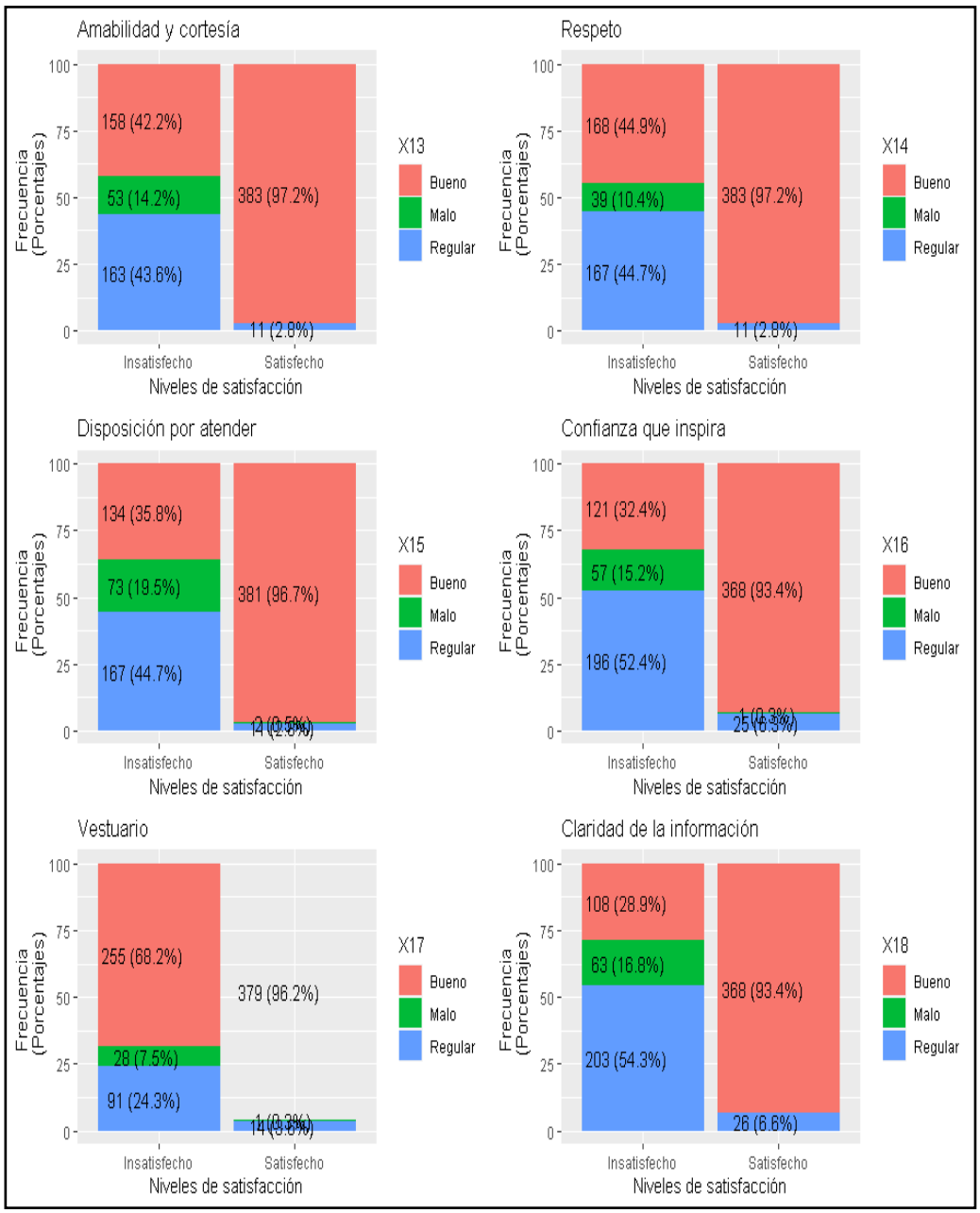
Distribución de la calificación de la atención recibida de parte del personal no médico



En la Figura 13, se muestra la distribución de la calificación de la atención recibida a los usuarios en los servicios de salud con respecto al personal no médico en seis aspectos evaluados. Se observa que los usuarios que manifestaron estar insatisfechos, han califican la atención recibida con Regular en mayor porcentaje con respecto a la amabilidad, respecto, atención, confianza, vestuario e información, mientras que los usuarios que están satisfechos lo han calificado como Bueno con el mayor porcentaje.

Figura 13

Distribución de la calificación de la atención recibida de parte del personal médico



4.3 Aplicación de los MTD supervisadas

Se aplican cuatro TMD supervisadas para la clasificación: regresión logística binaria, árbol de decisión C5.0, Naive Bayes y Random Forest. Para el aprendizaje se consideró una división del 70% para el conjunto de datos de entrenamiento y 30% para el conjunto de prueba. En el Tabla 8, se presenta los resultados de la aplicación de las técnicas de minería de datos considerando el total de todos los atributos de la base de datos. Los valores corresponden a los porcentajes de las tasas de precisión de los clasificadores que evalúa el porcentaje de la clasificación correcta, los valores de la AUC (curva bajo el área ROC) y el índice Kappa. Se observa que la mayor tasa de precisión se consigue con random Forest con un 85,0% de buena clasificación, luego Naive Bayes con 82,5%, árbol de clasificación C5.0 con 82,0% y regresión logística nominal con 81,5%. El AUC que permite medir la performance de la TMD para varios umbrales, se presenta con mayor valor el clasificador random Forest con 0,942, seguido de la regresión logística binaria con 0,928, Naive Bayes con 0,907 y árbol de clasificación C5.0 con 0,876. El índice Kappa que permite evaluar el grado de concordancia en la matriz de confusión entre la clasificación observada y la predicha para una TMD, se presentada con mayor valor el algoritmo random Forest con un valor de 0,852 indicando una concordancia de la matriz de confusión casi perfecto, luego los clasificadores Naive Bayes, árbol de clasificación C5.0 y regresión logística nominal con valores de 0,756, 0,746 y 0,694 respectivamente indicando una sustancial concordancia.

Tabla 8

Métricas con las TMD con el total de atributos

Técnica de minería de datos	Precisión	AUC	Kappa
Regresión logística binaria	81,5	0,928	0,694
Árbol de clasificación C5.0	82,0	0,876	0,746
Redes bayesianas Naive	82,5	0,907	0,756
Random Forest	85,0	0,942	0,852

En el Tabla 8, se presenta los valores para el coeficiente Kappa y curva ROC (AUC) para las cuatro TMD considerando todos los atributos. Los coeficientes de Kappa están entre el rango de valores de 0,446 a 0,503, lo cual corresponde según el criterio a un grado de concordancia moderado. Esto indica que la matriz de confusión en las cuatro TMD muestra una

concordancia moderada entre la clasificación observada y la predicha por las TMD. Los valores de AUC para las cuatro TMD también muestran valores aceptables.

4.4 Aplicación de los métodos de selección de atributos por filtrado

Ser aplican los métodos por filtrado con las métricas de Chi-Cuadrado, ganancia de información, razón de ganancia y Relief. En el Tabla 9, se presenta el ranking de los 10 mejores atributos y los respectivos valores para cada una de las métricas consideradas. Se observa que las cuatro métricas de filtrado han coincidido en la selección en seis atributos (X1, X2, X14, X15, X16 y X18). Las métricas de Chi-Cuadrado, Ganancia de información y Razón de ganancia coinciden en seleccionar a 8 atributos (X1, X2, X4, X13, X14, X15, X16 y X18).

Tabla 9

Selección de atributos por métrica de filtrado

Métrica de filtrado	Ranking de atributos									
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
Chi-Cuadrado	X18	X15	X16	X13	X14	X1	X4	X3	X2	X8
	0,667	0,648	0,636	0,603	0,581	0,521	0,505	0,493	0,483	0,448
Ganancia de información	X18	X15	X16	X13	X14	X1	X4	X3	X2	X7
	0,368	0,350	0,330	0,309	0,285	0,209	0,195	0,189	0,180	0,161
Razón de ganancia	X18	X15	X13	X14	X16	X1	X4	X17	X2	X8
	0,296	0,291	0,279	0,272	0,272	0,157	0,139	0,139	0,139	0,137
Relief	X9	X7	X15	X1	X2	X14	X16	X18	X12	X8
	0,340	0,320	0,280	0,240	0,240	0,220	0,200	0,200	0,180	0,160

Con la finalidad de evaluar cada métrica de filtrado, se aplica a cada una cuatro TMD (regresión logística binaria, árbol de clasificación C5.0, Naive Bayes y Random Forest) y se evalúan en cada una de ellas su capacidad predictiva a través de la tasa de precisión a partir de la matriz de confusión. En el Tabla 10, se presenta para cada una de las métricas propuestas para la selección de atributos por filtrado el porcentaje de la tasa de buena clasificación (precisión) obtenidas con cada una de las cuatro TMD. Se aprecia los mayores porcentajes de la tasa de buena clasificación para el clasificador Random Forest en las cuatro métricas de filtrado y en segundo lugar el árbol de clasificación C5.0. Por lo tanto, la TMD Random

Forest muestra la mejor capacidad predictiva para predecir el grado de satisfacción de la atención recibida por los usuarios de salud.

Tabla 10

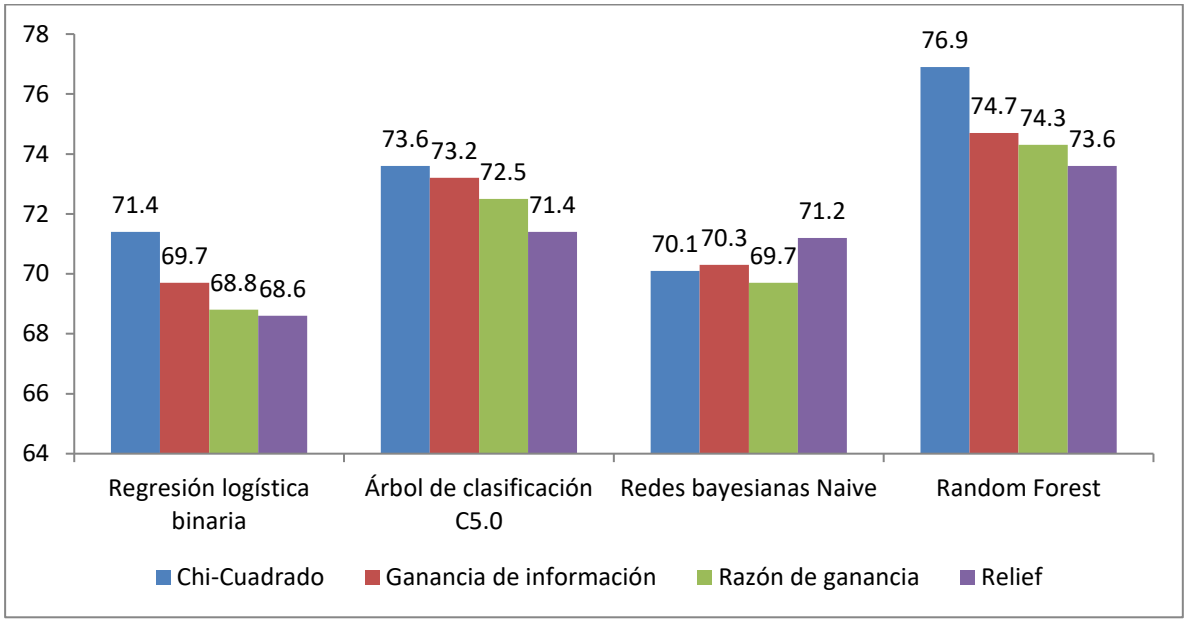
Porcentaje de buena clasificación de las TMD para cada métrica de filtrado

TMD	Métricas para el filtrado de atributos			
	Chi-Cuadrado	**Ganancia de información**	**Razón de ganancia**	**Relief**
Regresión logística binaria	71,4	69,7	68,8	68,6
Árbol de clasificación C5.0	73,6	73,2	72,5	71,4
Redes bayesianas Naive	70,1	70,3	69,7	71,2
Random Forest	76,9	74,7	74,3	73,6

En la Figura 14, se muestra la comparación de los porcentajes de las tasas de buena clasificación para cada uno de las TMD con cada una de las cuatro métricas de filtrado. Se puede mencionar que, en la regresión logística binaria, árbol de clasificación C5.0 y random Forest se consigue la mayor tasa de precisión con la métrica Chi-Cuadrado, seguido con la métrica ganancia de información, razón de ganancia y por último Relief. En el caso de Naive Bayes, las cuatro métricas muestran similar tasa de buena clasificación. En la TMD Random Forest, es la que presenta las mayores tasas de precisión, siendo los valores obtenidos para las métricas Chi-Cuadrado, ganancia de información, razón de ganancia y Relief 76,9%, 74,7%, 74,3% y 73,6% respectivamente.

Figura 14

Comparación de las tasas de precisión de las TMD con las métricas de filtrado



En el Tabla 11, se presenta los valores para los AUC de la curva ROC que permite medir la performance de los clasificadores evaluando varios umbrales de puntos de corte. Se presenta para las cuatro TMD y para cada una de las métricas de filtrado. En general, los mayores AUC para las cuatro métricas de filtrado se consigue con random Forest. Para la regresión binaria, el mayor AUC es con la métrica Chi-Cuadrado (0,926), para el árbol de clasificación C5.0 es con Chi-Cuadrado (0,872), para Naive Bayes es la razón de ganancia (0,764) y para random Forest es similarmente con las métricas Chi-Cuadrado, ganancia de información y razón de ganancia (0,834).

Tabla 11

AUC de las TMD para cada una de las métricas de filtrado

TMD	Métricas para el filtrado de atributos			
	Chi-Cuadrado	**Ganancia de información**	**Razón de ganancia**	**Relief**
Regresión logística binaria	0,926	0,764	0,764	0,839
Árbol de clasificación C5.0	0,872	0,747	0,695	0,747
Redes bayesianas Naive	0,738	0,738	0,764	0,747
Random Forest	0,834	0,834	0,834	0,817

En el Tabla 12, se presenta los valores del coeficiente Kappa para las cuatro TMD aplicando a cada una de las cuatro métricas de filtrado, que permite medir la concordancia de la matriz de confusión entre la clasificación observada y predicha con la TMD. Las TMD Naive Bayes y Randon Forest presentan valores entre el rango [0,81- 1,0] para las cuatro métricas de filtrado, indicando que tienen una concordancia casi perfecta la matriz de confusión, similar comportamiento se muestra para la regresión logística binaria y árbol de clasificación C5.0 pero sólo para las métricas de ganancia de información, razón de ganancia y Relief y una concordancia sustancial con la métrica Chi-Cuadrado.

Tabla 12

Coeficiente de concordancia Kappa con las TMD y las métricas de filtrado

TMD	Métricas para el filtrado de atributos			
	Chi-Cuadrado	**Ganancia de información**	**Razón de ganancia**	**Relief**
Regresión logística binaria	0,764	0,928	0,925	0,917
Árbol de clasificación C5.0	0,746	0,878	0,846	0,876
Redes bayesianas Naive	0,919	0,918	0,920	0,917
Random Forest	0,938	0,936	0,940	0,941

4.5 Aplicación de los métodos de selección de atributos por Wrapper

Se aplican el método Wrapper para seleccionar subconjuntos de atributos, considerando como método de búsqueda el Best-Firt, Greedy-Forwarf, Greedy-Backward y Hill-Climbing y usando como algoritmo para la evaluación el árbol de clasificación CART. En el Tabla 13, se presenta los subconjuntos de atributos seleccionados con cuatro métodos Wrapper, mostrando diferente número y atributos seleccionados.

Tabla 13

Selección de subconjuntos de atributos con el método Wrapper

Método Wrapper	Selección de atributos	Número
Best-first	X3,X5,X11,X14,X18	5
Greedy forward	X1,X3,X14,X16,X17,X18	6
Greedy backward	X3,X4,X5,X6,X7,X8,X9,X10,X11,X12, X13, X14,X15,X16,X17,X18	16
Hill climbing	X2,X3,X4,X5,X7,X8,X11,X16,X18	9

En el Tabla 14, se presenta los porcentajes de las tasas de buena clasificación (precisión) con las cuatro TMD y para cada uno de los métodos Wrapper. Se aprecia que los mayores porcentajes de la tasa de buena clasificación se consigue con el clasificador Random Forest en los cuatro métodos Wrapper, por lo tanto, es el que tienen mejor capacidad predictiva para predecir el grado de satisfacción de la atención recibida por los usuarios de salud. Para la regresión logística binaria se consigue las mayores tasas de buena clasificación con los métodos Best-Firts y Greedy Forward (77,7%), para el árbol de clasificación C5.0 el método Greedy Forward y para Naive Bayes el método Greedy Backward.

Tabla 14

Porcentaje de buena clasificación de las TMD para cada uno de los métodos Wrapper

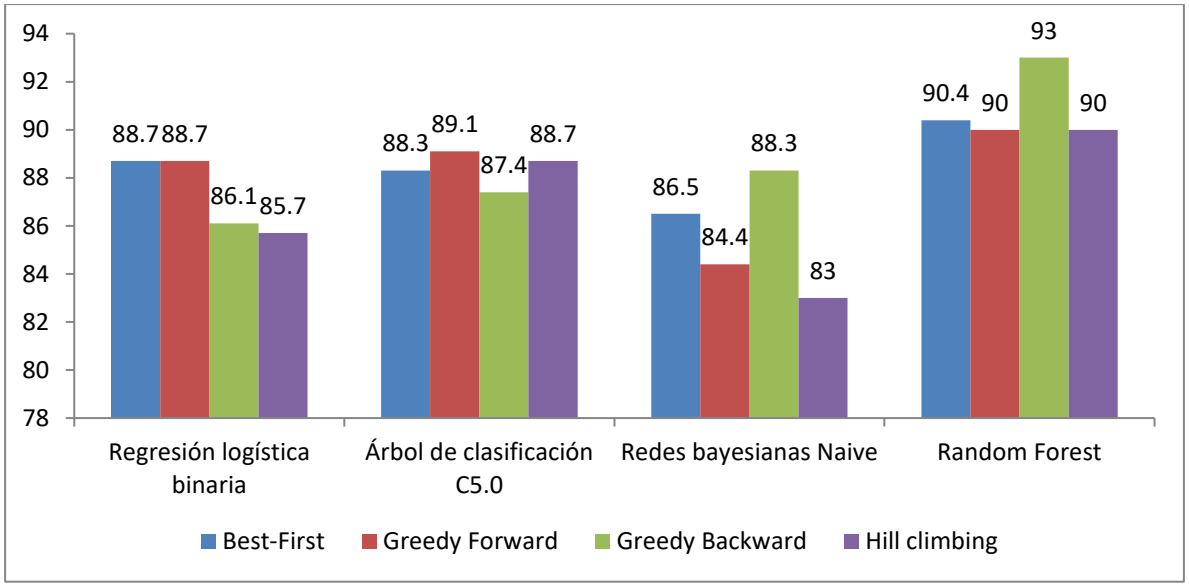
TMD	Métodos de Wrapper			
	Best-First	**Greedy Forward**	**Greedy Backward**	**Hill Climbing**
Regresión logística binaria	88,7	88,7	86,1	85,7
Árbol de clasificación C5.0	88,3	89,1	87,4	88,7
Redes bayesianas Naive	86,5	84,4	88,3	83,0
Random Forest	90,4	90,0	93,0	90,0

En la Figura 15, se muestra la comparación de los porcentajes de las tasas de buena clasificación para cada uno de las TMD con cada una de los métodos Wrapper. Se puede mencionar que en la regresión logística binaria la mayor tasa de precisión se tiene con el método Best-First y Greedy Forward (88,7%), el árbol de clasificación C5.0 con el método

Greedy Forward (89,1%), Naive Bayes con el método Greedy Backward (88,3%) y Random Forest con el método Greedy Backward (93,0%).

Figura 15

Comparación de las tasas de precisión de las TMD con los métodos Wrapper



En el Tabla 15, se presenta los valores para los AUC de la curva ROC que permite medir la performance de los clasificadores evaluando varios umbrales de puntos de corte. Se presenta para las cuatro TMD y para cada una de los métodos Wrapper. En general, los mayores AUC para los cuatro métodos Wrapper se consigue con Random Forest. Para la regresión logística binaria, árbol de clasificación C5.0 y Naive Bayes el mayor AUC es con el método Greedy Forward con valores 0,932, 0,891 y 0,922 respectivamente.

Tabla 15

AUC con las TMD para cada uno de los métodos Wrapper

Técnica de minería de datos	Métodos de Wrapper			
	Best-First	**Greedy Forward**	**Greedy Backward**	**Hill Climbing**
Regresión logística binaria	0,929	0,932	0,909	0,918
Árbol de clasificación C5.0	0,882	0,891	0,879	0,885
Redes bayesianas Naive	0,916	0,922	0,903	0,906
Random Forest	0,925	0,932	0,941	0,937

En el Tabla 16, se presenta los valores del coeficiente Kappa para las cuatro TMD aplicando a cada una de los cuatro métodos Wrapper, que permite medir la concordancia de la matriz de confusión entre la clasificación observada y predicha con la TMD. Las TMD regresión logística binaria, árbol de clasificación C5.0, Naive Bayes y Random Forest presentan valores entre el rango [0,61- 0,80] para los cuatro métodos, indicando que tienen una concordancia sustancial de la matriz de confusión.

Tabla 16

Coeficiente de concordancia Kappa con las TMD y los métodos Wrapper

Técnica de minería de datos	Métodos de Wrapper			
	Best-First	**Greedy Forward**	**Greedy Backward**	**Hill Climbing**
Regresión logística binaria	0,740	0,773	0,721	0,712
Árbol de clasificación C5.0	0,765	0,782	0,746	0,773
Redes bayesianas Naive	0,729	0,685	0,764	0,660
Random Forest	0,808	0,799	0,861	0,799

4.6 Comparación de las TMD con los métodos de selección de atributos

En el Tabla 17, se presenta la comparación de las tasas de buena clasificación obtenidas con los métodos de selección y sin selección de atributos y para cada una de las TMD analizadas. Se observa que los cuatro métodos de selección por subconjunto de atributos Wrapper presentan las más altas tasas de precisión, indicando que tienen la mejor capacidad predictiva para predecir la satisfacción de la atención recibida de los usuarios de salud.

Tabla 17

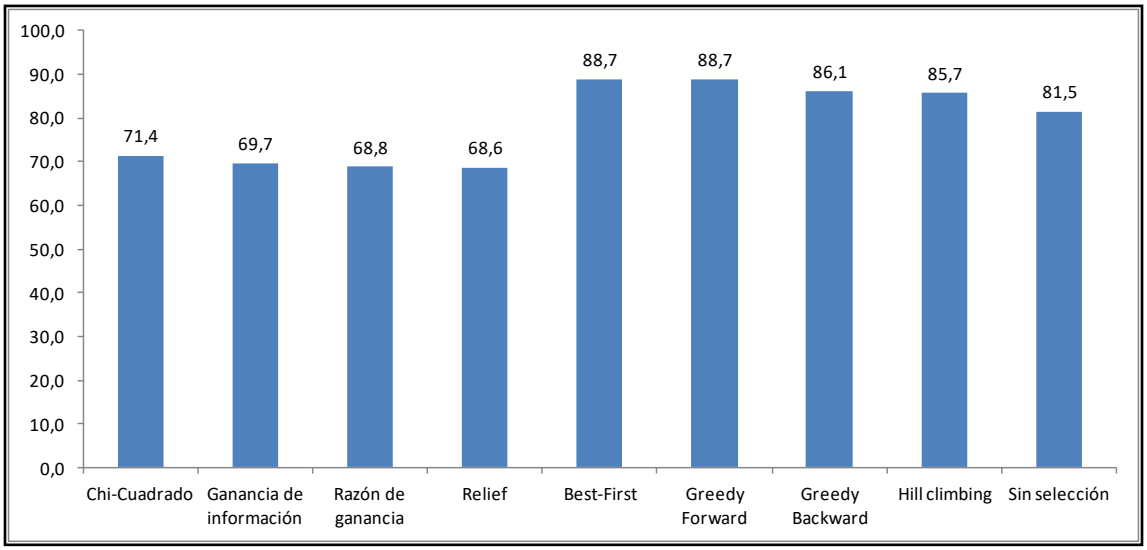
Comparación de las tasas de precisión de las TMD entre los métodos de selección de atributos

TMD	Métricas para el filtrado				Métodos de Wrapper				Sin selección
	Chi-Cuadrado	Ganancia de información	Razón de ganancia	Relief	Best-First	Greedy Forward	Greedy Backward	Hill climbing	
Regresión logística binaria	71,4	69,7	68,8	68,6	88,7	88,7	86,1	85,7	81,5
Árbol de clasificación C5.0	73,6	73,2	72,5	71,4	88,3	89,1	87,4	88,7	82,0
Naive Bayes	70,1	70,3	69,7	71,2	86,5	84,4	88,3	83,0	82,5
Randon Forest	76,9	74,7	74,3	73,6	90,4	94,0	93,0	90,0	85,0

En la Figura 16, se muestra la comparación de los porcentajes de precisión de la buena clasificación para los métodos de selección de atributos y sin selección aplicando la TMD de la regresión logística binaria. Las cuatro métricas de filtrado para la selección de atributos son las que muestran los menores porcentajes de buena clasificación. Los cuatro métodos de selección de atributos Wrapper son los que presentan los mayores porcentajes de las tasas de buena clasificación, siendo los mayores para el método Greedy-First (88,7%) y Greedy Forward (88,7%). Mientras que la tasa de buena clasificación sin la selección de atributos fue de 81,5%. Se afirmar que la regresión logística binaria, con el método de Wrapper BestFirt o Greedy Forward presentan la mayor capacidad predictiva para predecir la satisfacción de la atención de los usuarios de los servicios de salud de la encuesta 2015.

Figura 16

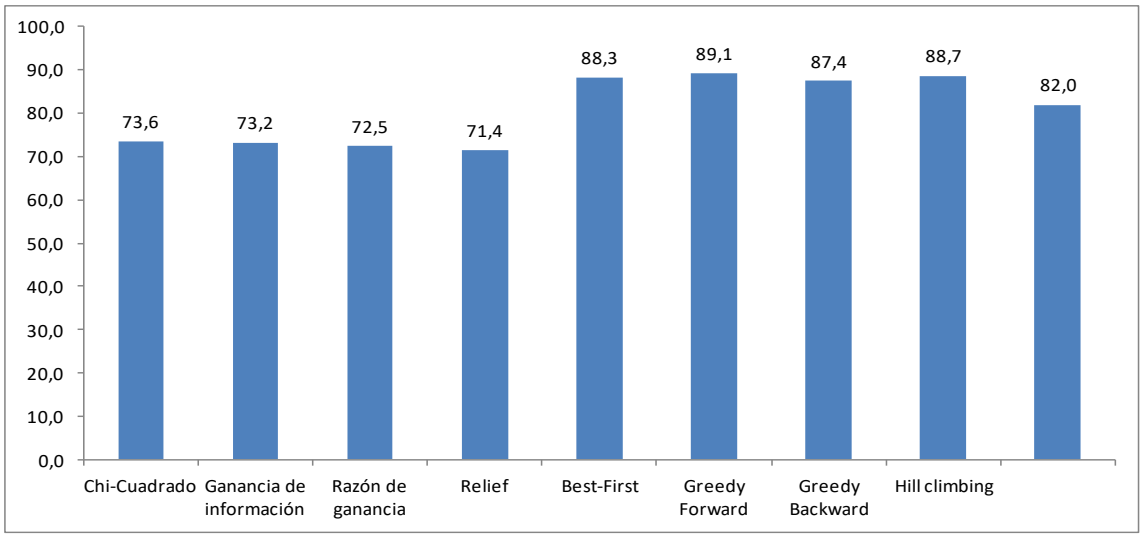
Porcentajes de precisión de métodos de selección de atributos con la regresión logística binaria



En la Figura 17, se muestra la comparación de los porcentajes de precisión de la buena clasificación para los métodos de selección de atributos y sin selección aplicando la TMD del árbol de clasificación C5.0. Las cuatro métricas de filtrado para la selección de atributos son las que muestran los menores porcentajes de buena clasificación. Los cuatro métodos de selección de atributos Wrapper son los que presentan los mayores porcentajes de las tasas de buena clasificación, siendo el mayor para el método Greedy Forward (89,1%). Mientras que la tasa de buena clasificación sin la selección de atributos fue de 82,0%. Se afirmar que el árbol de clasificación C5.0, con el método de Greedy Forward presenta la mayor capacidad predictiva para predecir la satisfacción de la atención de los usuarios de los servicios de salud de la encuesta 2015.

Figura 17

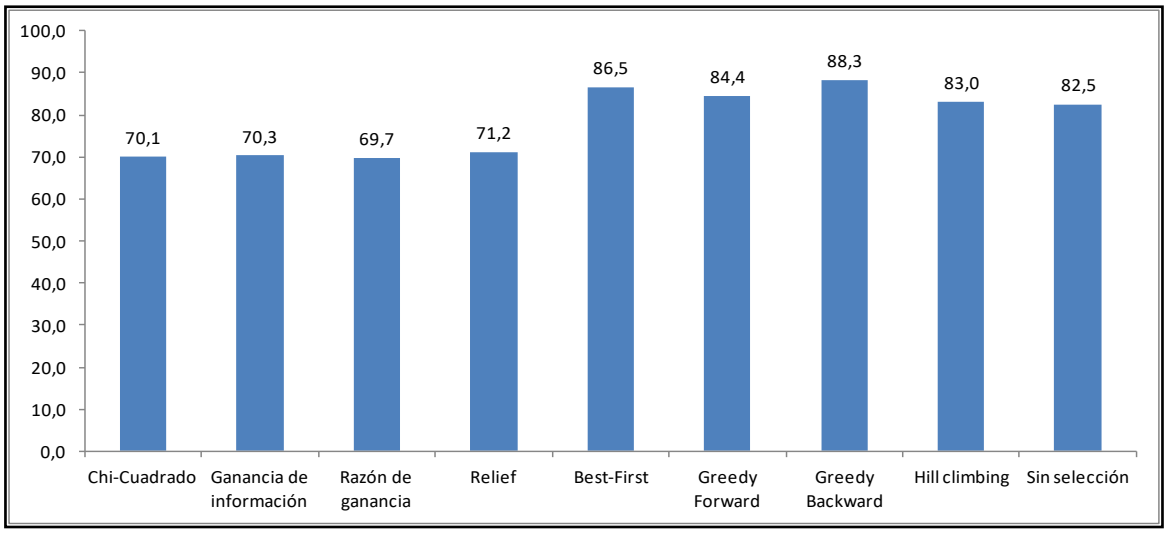
Porcentajes de precisión de métodos de selección de atributos con el árbol de clasificación C.5.0



En la Figura 18, se muestra la comparación de los porcentajes de precisión de la buena clasificación para los métodos de selección de atributos y sin selección aplicando la TMD de Naive Bayes. Las cuatro métricas de filtrado para la selección de atributos son las que muestran los menores porcentajes de buena clasificación. Los cuatro métodos de selección de atributos Wrapper son los que presentan los mayores porcentajes de las tasas de buena clasificación, siendo el mayor para el método Greedy Backward (88,3%). Mientras que la tasa de buena clasificación sin la selección de atributos fue de 82,5%. Se afirma que Naive Bayes, con el método de Greedy Backward presenta la mayor capacidad predictiva para predecir la satisfacción de la atención de los usuarios de los servicios de salud de la encuesta 2015.

Figura 18

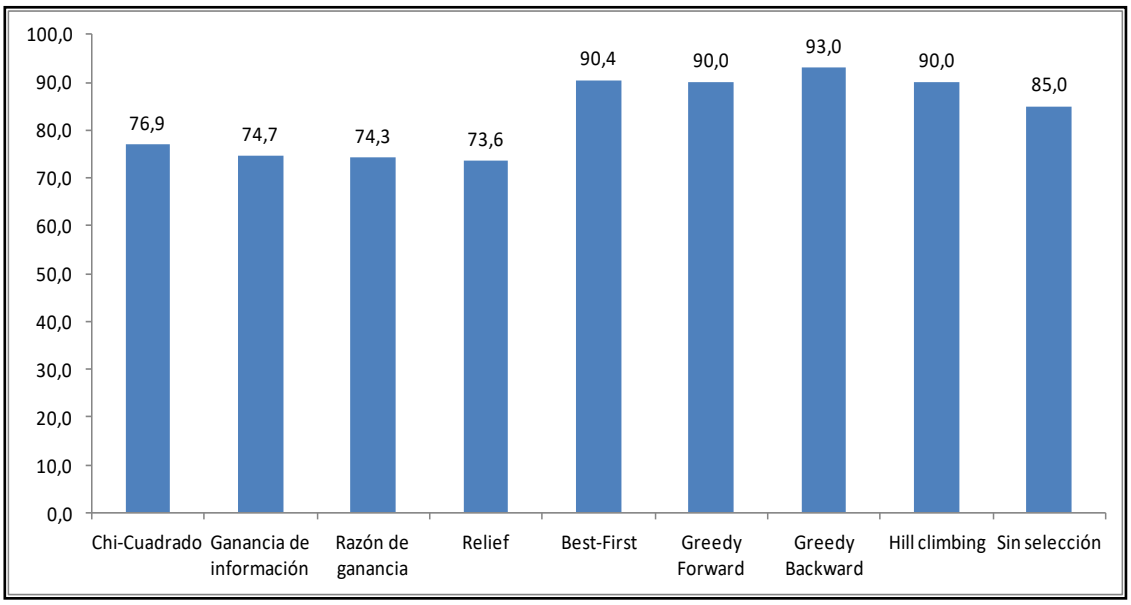
Porcentajes de precisión de métodos de selección de atributos con la red bayesiana Naive



En la Figura 19, se muestra la comparación de los porcentajes de precisión de la buena clasificación para los métodos de selección de atributos y sin selección aplicando la TMD de multiclasificador random Forest. Las cuatro métricas de filtrado para la selección de atributos son las que muestran los menores porcentajes de buena clasificación. Los cuatro métodos de selección de atributos Wrapper son los que presentan los mayores porcentajes de las tasas de buena clasificación, siendo el mayor para el método Greedy Backward (93,0%). Mientras que la tasa de buena clasificación sin la selección de atributos fue de 85,0%. Se afirmar que el multiclasificador Random Forest, con el método de Greedy Backward presenta la mayor capacidad predictiva para predecir la satisfacción de la atención de los usuarios de los servicios de salud de la encuesta 2015.

Figura 19

Comparación de los métodos de selección y sin selección de atributos con random Forest

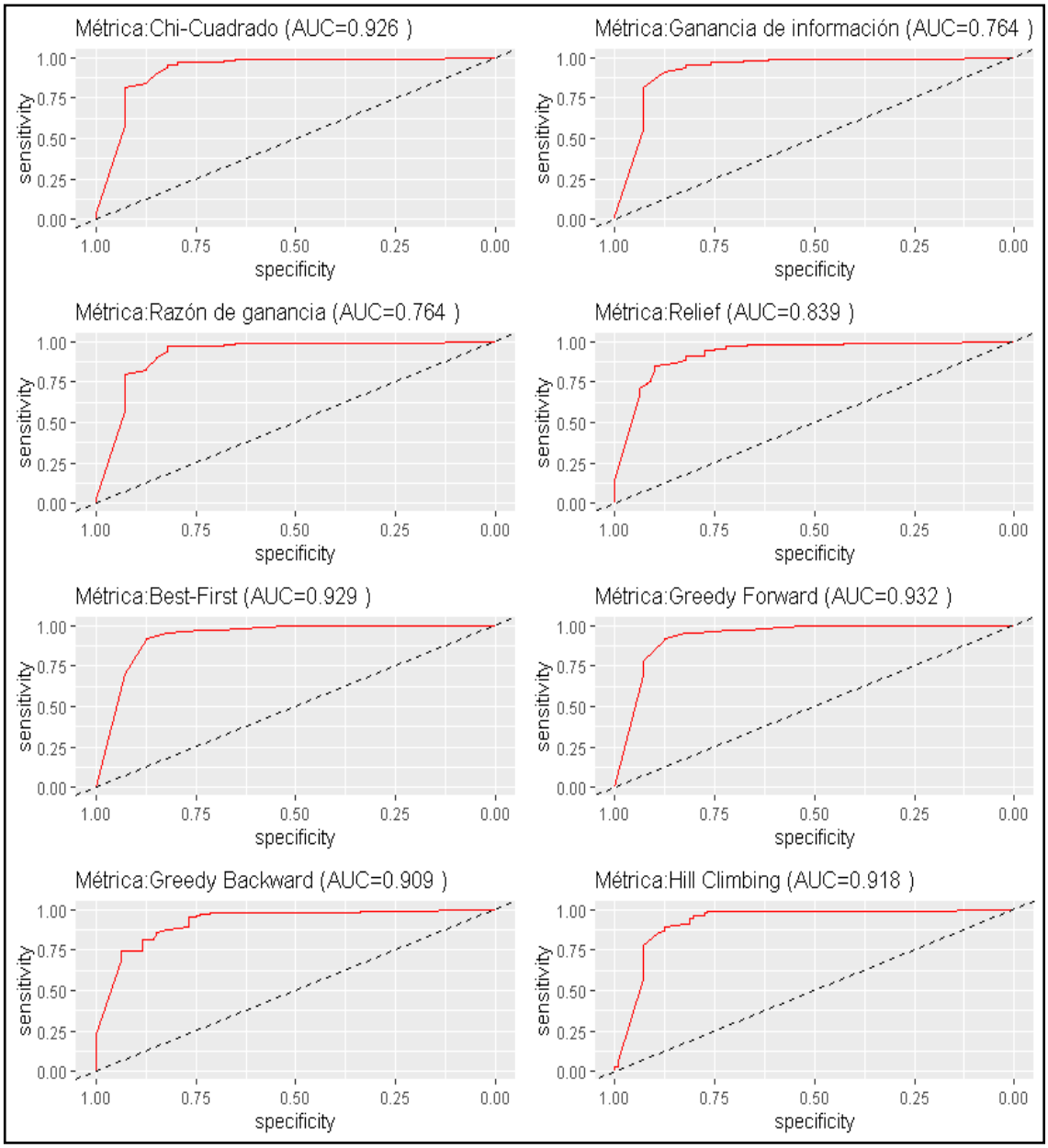


Por consiguiente, se observa que los cuatro métodos de selección de atributos Wrapper (Best-First, Greedy Forward, Greedy Backwar y Fill Climbing) son los que presentan los mayores porcentajes de buena clasificación con las cuatro técnicas de minería de datos (regresión logística binaria, árbol de clasificación C5.0, Naive Bayes y random Forest), superando a las tasas obtenidas sin selección de atributos.

En la Figura 20, se muestra la comparación de las curvas ROC y los valores de los AUC de los cuatro métodos de filtrado (Chi-Cuadrado, Ganancia de información, Razón de ganancia y Relief) y cuatro métodos de Wrapper (Best-First, Greedy Forward, Greedy Backward y Fill Climbing) aplicando la TMD de la regresión logística binaria. Se observa que casi todas las curvas ROC están cercanas al extremo superior izquierdo, indicando que los métodos de selección de atributos tienen un alto rendimiento de la capacidad predictiva para clasificar a los usuarios como satisfechos o insatisfechos en la satisfacción de la atención de los servicios de salud. También esto es mostrado por los altos valores de los AUC para los métodos de selección de atributos Wrapper.

Figura 20

Curvas ROC de los métodos de selección de atributos con la regresión logística binaria

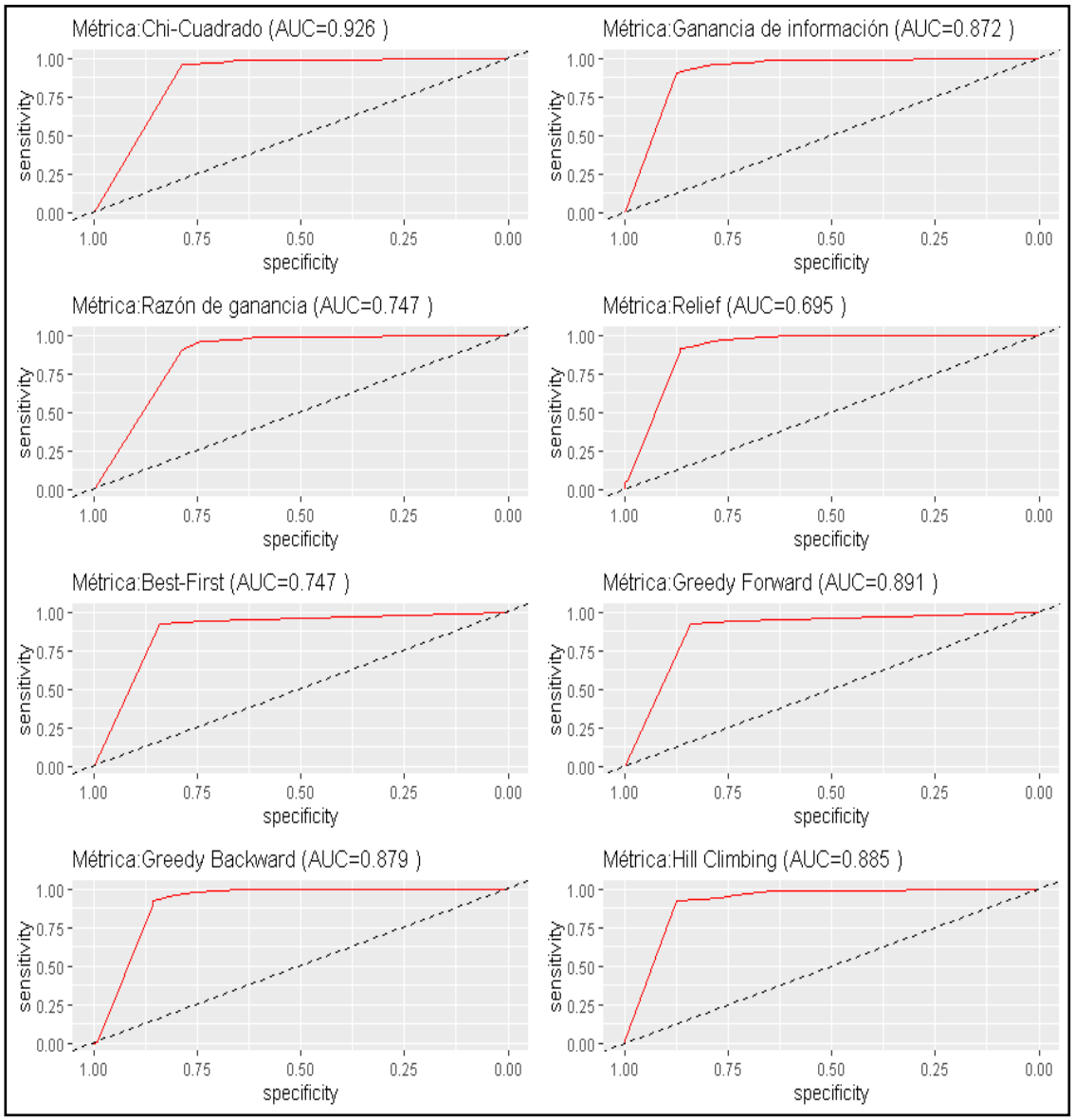


En la Figura 21, se muestra la comparación de las curvas ROC y los valores de los AUC de los cuatro métodos de filtrado (Chi-Cuadrado, Ganancia de información, Razón de ganancia y Relief) y cuatro métodos de Wrapper (Best-First, Greedy Forward, Greedy Backward y rand Fill Climbing) aplicando la TMD del árbol de clasificación C5.0. Se observa que casi todas las curvas ROC están cercanas al extremo superior izquierdo, indicando que los métodos de selección de atributos tienen un alto rendimiento de la capacidad predictiva para clasificar a los usuarios como satisfechos o insatisfechos en la satisfacción de la atención de los servicios de salud. También esto es mostrado por los altos valores de los AUC para los métodos de

selección de atributos de filtrado con la métrica Chi-Cuadrado (0.926) y Wrapper con el método Greedy Forward (0.891).

Figura 21

Curvas ROC de los métodos de selección de atributos con el árbol de clasificación C5.0

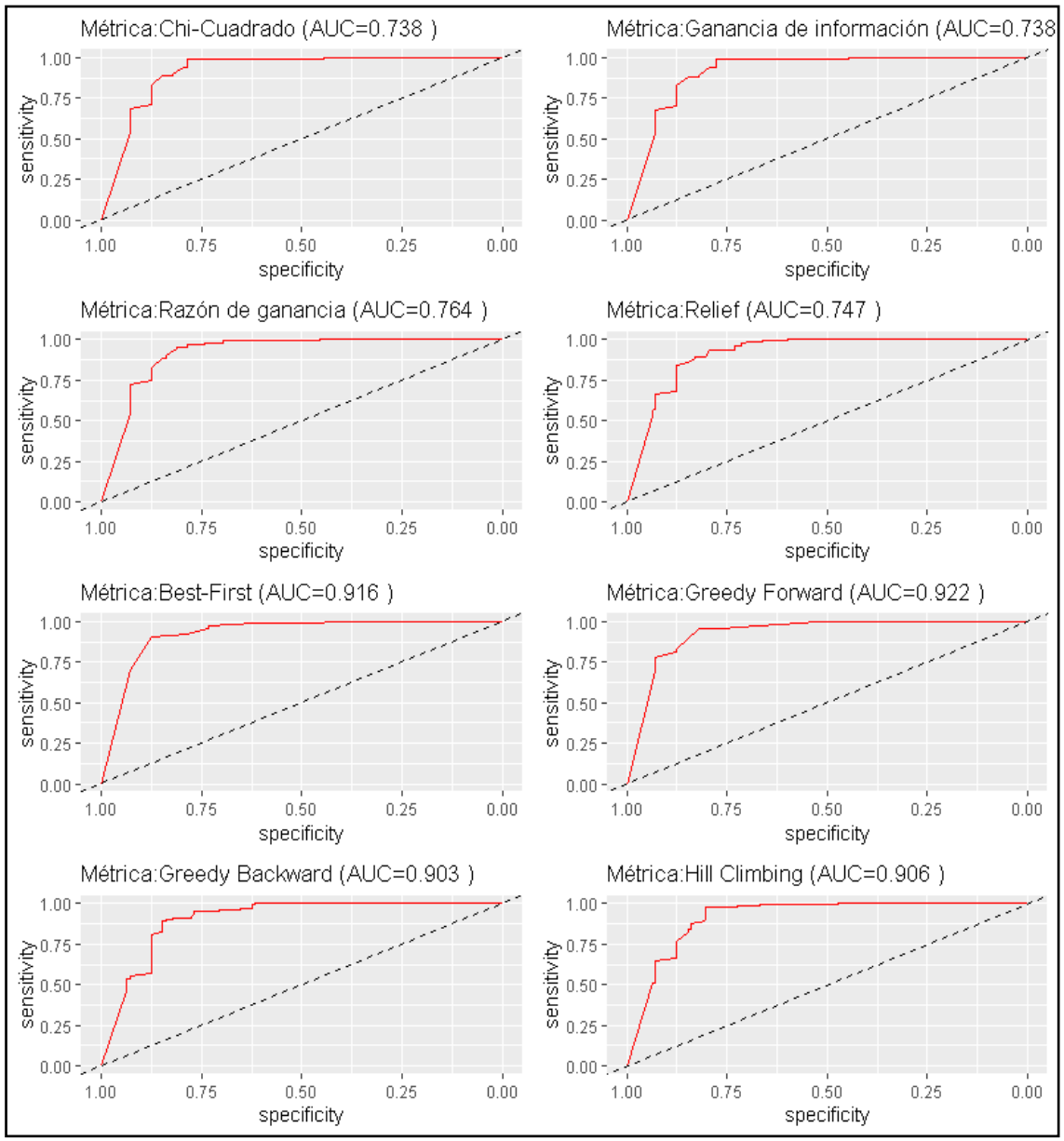


En la Figura 22, se muestra la comparación de las curvas ROC y los valores de los AUC de los cuatro métodos de filtrado (Chi-Cuadrado, Ganancia de información, Razón de ganancia y Relief) y cuatro métodos de Wrapper (Best-First, Greedy Forward, Greedy Backward y Fill Climbing) aplicando la TMD de la red bayesiana Naive. Se observa que los métodos Wrapper son los que presentan las curvas ROC más cercanas al extremo superior izquierdo, indicando que los métodos de selección de atributos tienen un alto rendimiento de la capacidad predictiva para clasificar a los usuarios como satisfechos o insatisfechos en la satisfacción de

la atención de los servicios de salud. También esto es mostrado por los altos valores de los AUC para los métodos de selección de atributos con Wrapper, sobre todo con el método Greedy Forward (0.922).

Figura 22

Curvas ROC de los métodos de selección de atributos con la red bayesiana Naive

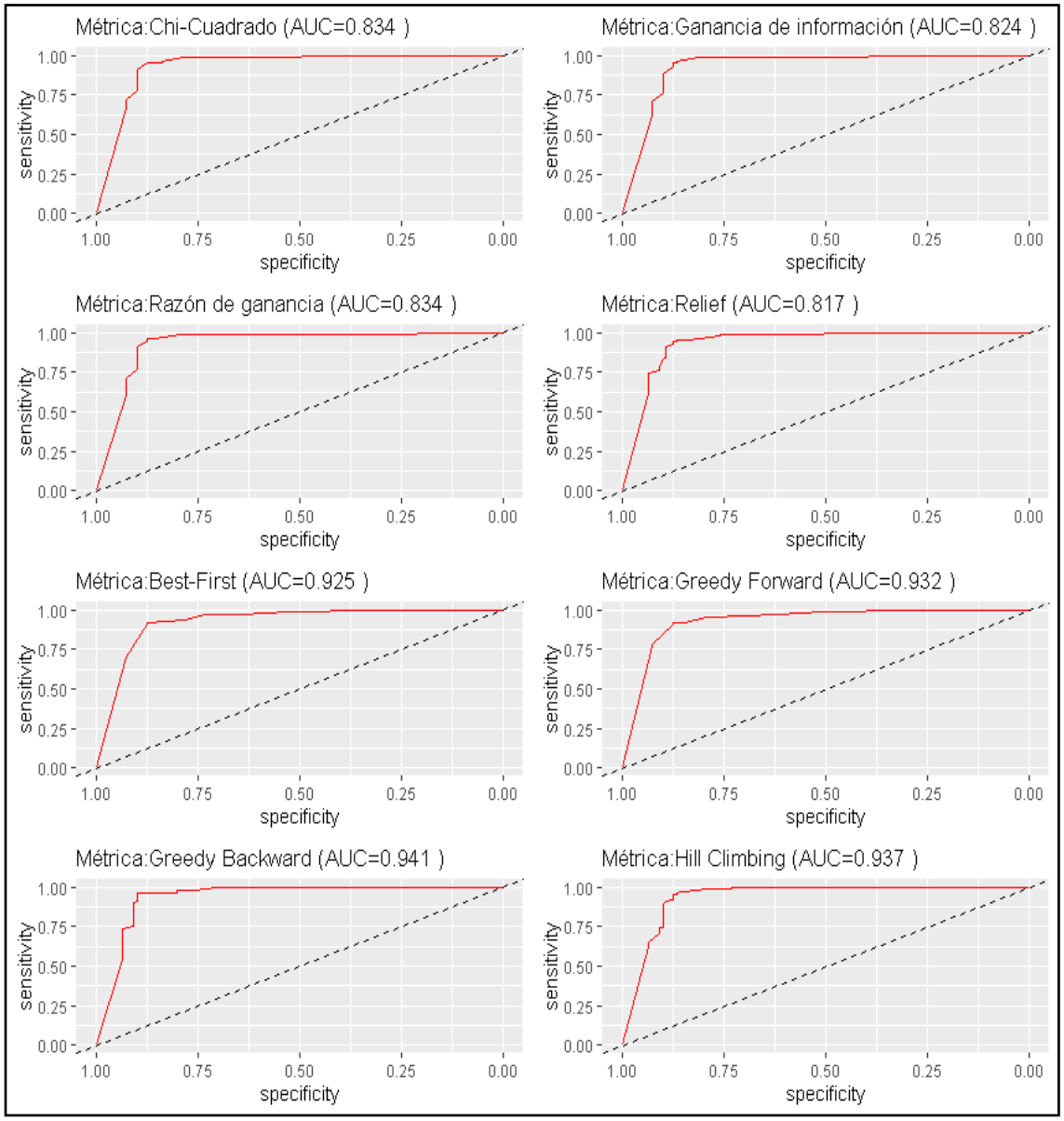


En la Figura 23, se muestra la comparación de las curvas ROC y los valores de los AUC de los cuatro métodos de filtrado (Chi-Cuadrado, Ganancia de información, Razón de ganancia y Relief) y cuatro métodos de Wrapper (Best-First, Greedy Forward, Greedy Backward y Fill Climbing) aplicando la TMD random Forest. Se observa que todos los métodos de selección

de atributos presentan las curvas ROC más cercanas al extremo superior izquierdo, indicando que los métodos de selección de atributos tienen un alto rendimiento de la capacidad predictiva para clasificar a los usuarios como satisfechos o insatisfechos en la satisfacción de la atención de los servicios de salud. También esto es mostrado por los altos valores de los AUC, siendo para los métodos de selección de atributos con filtrado con la métrica Chi-Cuadrado y Razón de ganancia (0.834) y Wrapper con Greedy Backward (0.941).

Figura 23

Curvas ROC de los métodos de selección de atributos con Random Forest



V. CONCLUSIONES

Las conclusiones de la presente investigación son:

1. Las TMD supervisada pueden hacer frente al problema de la alta dimensionalidad de atributos en las bases de datos, usando dos enfoques para la selección de atributos, el método por filtrado que se basa en usar una métrica para obtener como resultados un ranking de atributos y los métodos wrapper que dependiente de una TMD para evaluar y obtener como resultado un subconjunto de atributos.

2. El método de selección de atributos por filtrado con las métricas Chi-Cuadrado, ganancia de información, razón de ganancia y Relief coincidieron en identificar a 6 atributos relevantes (X1, X2, X14, X15, X16 y X18). Los métodos Wrapper con Best-First seleccionaron un subconjunto de 5 atributos relevantes (X3,X5,X11,X14,X18), con Greedy forward 6 (X1,X3,X14,X16,X17,X18), con Greedy backward 16 (X3,X4,X5,X6,X7,X8,X9, X10,X11,X12,X13,X14,X15,X16,X17,X18), con Hill climbing 9 (X2,X3,X4,X5,X7,X8, X11,X16,X18).

3. La regresión logística binaria obtuvo las mayores tasas de buena clasificación, aplicando la selección de atributos por filtrado con la métrica Chi-Cuadrado (71,4%) y con los métodos Wrapper de Best-First (88,7%) y Greedy Forward (88,7%), con la finalidad de predecir la satisfacción de los usuarios de la atención recibida de los servicios de salud.

4. El árbol de clasificación C5.0 obtuvo las mayores tasas de buena clasificación, aplicando la selección de atributos por filtrado con la métrica Chi-Cuadrado (73,6%) y con el método Wrapper de Greedy Forward (89,1%), con la finalidad de predecir la satisfacción de los usuarios de la atención recibida de los servicios de salud.

5. La Naive Bayes obtuvo las mayores tasas de buena clasificación, aplicando la selección de atributos por filtrado con la métrica Relief (71,2%) y con el método Wrapper de Greedy Backward (88,3%), con la finalidad de predecir la satisfacción de los usuarios de la atención recibida de los servicios de salud.

6. El multiclasificador random Forest obtuvo las mayores tasas de buena clasificación, aplicando la selección de atributos por filtrado con la métrica Chi-Cuadrado (76,9%) y con el método Wrapper de Greedy Backward (93,0%), con la finalidad de predecir la satisfacción de los usuarios de la atención recibida de los servicios de salud.

7. Los mayores AUC para la regresión logística binaria fue con el método Greedy forward con 0,932, el árbol de clasificación C5.0 con Greedy forward con 0,891, Naive Bayes con wrapper Greedy forward con 0,9221 y el multiclasificador Random Forest con Greedy backard con 0,941.

VI. RECOMENDACIONES

Las recomendaciones son las siguientes:

1. Aplicar a los métodos de selección de atributos por filtrado wrapper las técnicas de ensambladores Bagging y Boosting con la finalidad de mejorar la capacidad predictiva de los TMD.
2. Aplicar la selección de atributos, usando un método hibrido es decir realizar en dos etapas la selección, en la primera fase el método de filtrado luego aplicar los métodos de wrapper como alternativa al uso individual de cada método de selección.

VII. REFERENCIA BIBLIOGRAFÍA

Bolón-Canedo, V., & Sánchez-Maroño, N. (2013). A review of feature selection methods on synthetic data. *Knowledge and Information System, Vol.34, No. 3*, pp. 483-51.

Brahim, A., & Limam, M. (2016). A hybrid feature selection method based on instance learning and cooperative subset search. pp. 28-34.

Breiman, L. (1996). Bagging predictors. *Machine Learning, Kluwer Academic Publishers, Boston, Manufactured in The Netherlands, vol. 24, 2*, pp. 123-140.

Cohen, J. (1960). A coefficient of agreement for nominal scales. *Educational Psychol*, pp. 37-46.

Dasgupta, A., Drineas, P., Harb, B., Josifovski, V., & Mahonay, M. (2007). Feature Selection Methods for Text Classification.

ElAlami, M.E. (2009). Filter model for feature subset selection based on genetic algorithm. pp. 356-362.

Fayyad, U., Piatetsky-shapiro, G., & Smyth, P. (1996). From data mining to knowledge discovery in databases. *AI Magazine Volume 17 Number 3* .

Forman, G. (2003). An extensive empirical study of feature selection metrics for text classification. *Journal of Machine Learning Research, Vol. 3.*, pp. 1289-1305.

Freund, Y., & Schapire, R. E. (1996). Experiments with a new boosting algorithm. *Proceedings 13th International Conference on Machine Learning*, pp. 148-156.

Hernández-Orallo, J., Ramírez, M. J., & Ferri, C. (2004). Introducción a la minería de datos. *Pearson Educación, S.A. Madrid*.

Kalousis, A., Prados, J., & Hilario, M. (2007). Stability of Feature Selection Algorithms: a study on high dimensional spaces. *Knowledge and information System, vol. 12, No. 1*, pp. 95-116.

Kira, K & Rendell, L. A. (1992). The Feature Selection Problem: Traditional Methods and a New Algorithm. *Proceedings of the 10th National Conference on Artificial Intelligence.*

Kumar, A., & Elavarasan, N. (2014). A Survey on Dimensionality Reduction Technique. *International Journal of Emerging Trands & Technology in Computer Science (ITETICS). Vol.3 (6).*, pag. 36-41.

Kumar, V., & Minz, S. (2014). Feature Selection: A literature Review. *Smart Computing Revier, Vol. 4, No.3.*, Pag. 211-229.

Ladha, A. (2011). Feature Selection Methods and Algorithms. *International Journal on Computer Science and Engineering (IJCSE), Vol. 3 No. 5* , pp. 1787-1797.

Ladha, L., & Deepa, T. (2011). Feature selection methods and algorithms. *International Journal on Computer Science and Engineering (IJCSE). Vol. 3, N°. 5.*, pp. 1787-1797.

Landis, J. R., & Koch, G. G. (1977). The measurement of observer agreement for categorical data. *Biometrics*, pp. 159-174.

Lin, P. (2003). A Framework for Consistency Based Feature Selection. *Western Kentucky University, Graduate Studies and Research Master Thesis*.

Mani, K., & Kalpana, P. (2016). A Review on Filter Based Feature Selection. *International Journal of Innovative Research in Computer and Communication Engineering (An ISO 3297: 2007 Certified Organization) Vol. 4, Issue 5*.

Mitra, P., Murthy, C. A., & Pal, S. . (2002). Unsupervised Feature Selection using Feature Similarity. *IEEE Transactions on Pattern Analysis and Machine Intelligence", Vol 24, No 4*.

Molina, L., Belanche, LL., & Nebot, A. (2002). Feature Selection Algorithms: A Survey and Experimental Evaluation. pp. 306-313.

Novakovic, J., Strbac, P., & Bulatovic, D. (2011). Toward optimal feature selection using ranking methods and classification algorithms. *Yugoslav Journal of Operations Research 21* , pp. 119-135.

S., K., & Kiruthika, P. (2015). An Overview of Classification Algorithm in Data mining. *International Journal of Advanced Research in Computer and Communication Engineering Vol. 4, Issue 12*, pp. 255-257.

Schiezaro, M., & Pedrini, H. (2013). Data Feature selection based on Artificial Bee Colony algorithm. *Journal on Image and Video Processing, 47*.

Segrera, S., & Moreno, M. (2006). Multiclasificadores. Métodos y Arquitectura. *Informe técnico.*, pp. 1-41.

Tu, C.-J., Chuang, L., & Yang, Ch. (2007). Feature Selection using PSO-SVM.

Wang, L. (2005). Support Vector Machines: Theory and Applications (Studies in Fuzziness and Soft Computing). *Springer-Verlag*.

Yin, L., Ge, Y., Xiao, K., Wang, X., & Quan, X. (2013). Feature selection for high-dimensional imbalanced data. *Neurocomputing 105*, pp. 3-11.

Yu, L., & Liu, H. (2003). Feature Selection for High-Dimensional Data: A Fast Correlation-Based Filter Solution. *Proceedings of the Twentieth International Conference on Machine Learning*.

Yuanhong, L., Ming, D., & Jing, H. (2007). AGaussian Mixture Model To Detect Clusters Embedded In Feature Subspace. *Journal of Communications in Information and Systems, Vol.7, No.4*, pag. 337-352.

Yun, Ch., Shin, D., Jo, H., Yang, J., & Kim S. (2007). An Experimental Study on Feature Subset Selection Methods. *Seventh International Conference on Computer and Information Technology. IEEE*.

VIII. ANEXOS

```
#####################################################################
#  Programa R. Tesis_Fernando Ancajima                              #
#  Métodos de selección de atributos                                #
#####################################################################
#
# 1. Pre procesamiento de datos
library(tidyverse)
library(ggplot2)
library(factoextra)
library(gridExtra)
library(dplyr)
library(patchwork)
library(ROSE)
#
#  Lectura de datos
Datos=read.table("Ancajima_Tesis_Datos.csv",stringsAsFactors=TRUE,header=TRUE,sep=";")
attach(Datos)
n=dim(Datos)[1]; p=dim(Datos)[2]-1
f=table(Datos$Y); fr=round(prop.table(f)*100,1)
Tabla=as.data.frame(cbind(addmargins(f),addmargins(fr))); Tabla
Tabla_Y <- Datos %>% group_by(Y) %>% summarise(Total=n()) %>%
  dplyr::mutate(Porcentaje = round(Total/sum(Total)*100, 1))
G_Y<-ggplot(Tabla_Y, aes(x = Y, y=Porcentaje,fill=Y) ) +
  geom_bar(width = 0.9, stat="identity", position = position_dodge())+ylim(c(0,100))+
labs(x="Niveles de satisfacción", y= "Frecuencia \n (Porcentaje)") + labs(fill = "")+
geom_text(aes(label=paste0(Total," ", "", "(", Porcentaje, "%",")")), vjust=-0.9, color="black",
hjust=0.5, position = position_dodge(0.9), angle=0, size=4.0) +  scale_fill_discrete(name = "Niveles",
labels = c("Insatisfecho", "Satisfecho")) + theme(axis.text.x = element_text(angle = 45, vjust = 1,
hjust=1)) + theme_bw(base_size = 14) +  scale_x_discrete(labels=c("Insatisfecho", "Satisfecho")) +
labs(title="Satisfacción de los usuarios de salud")
G_Y
# 1.1 Balanceo de datos (Sobre muestreo)
Datos_B<-ROSE(Y~., data=Datos, seed=123)$data
# Guardar en un archivo los datos balanceados
write.table(Datos_B,  file ="Ancajima_Tesis_Datos_Balanceados.csv", sep = ";", eol = "\n", dec=
".", row.names = TRUE, col.names = TRUE)
Datos=read.table("Ancajima_Tesis_Datos_Balanceados.csv", stringsAsFactors=TRUE,
header=TRUE,sep=";")
attach(Datos)
f=table(Datos_B$Y); fr=round(prop.table(f)*100,1)
Tabla=as.data.frame(cbind(addmargins(f),addmargins(fr))); Tabla
Tabla_Y <- Datos %>% group_by(Y) %>% summarise(Total=n()) %>%
  dplyr::mutate(Porcentaje = round(Total/sum(Total)*100, 1))
G_Y<-ggplot(Tabla_Y, aes(x = Y, y=Porcentaje,fill=Y) ) +
  geom_bar(width = 0.9, stat="identity", position = position_dodge())+ylim(c(0,100))+
labs(x="Niveles de satisfacción", y= "Frecuencia \n (Porcentaje)") + labs(fill = "")+
geom_text(aes(label=paste0(Total," ", "", "(", Porcentaje, "%",")")), vjust=-0.9, color="black",
hjust=0.5, position = position_dodge(0.9), angle=0, size=4.0) +  scale_fill_discrete(name = "Niveles",
labels = c("Insatisfecho", "Satisfecho")) + theme(axis.text.x = element_text(angle = 45, vjust = 1,
hjust=1)) +  theme_bw(base_size = 14) +  scale_x_discrete(labels=c("Insatisfecho", "Satisfecho")) +
labs(title="Satisfacción de los usuarios de salud")
G_Y
#  1.2 Análisis exploratorio de datos
# Detección datos faltantes
any(!complete.cases(Datos))
```

```
map_dbl(Datos, .f = function(x){sum(is.na(x))})
Datos %>% map_lgl(.f = function(x){any(!is.na(x) & x == "")})
#  Gráfico de las variables cualitativas
Tabla_X1<- Datos %>% group_by(Y, X1) %>% summarise(Total=n()) %>%
  dplyr::mutate(Porcentaje = round(Total/sum(Total)*100, 1))
G_X1<-ggplot(data=Tabla_X1, aes(x=Y, y=Porcentaje, fill=X1)) +  geom_bar(width = 0.9,
stat="identity")+ ylim(c(0,100))+ labs(x="Niveles de satisfacción", y= "Frecuencia \n (Porcentajes)")
+ labs(fill = "")+ scale_x_discrete(labels=c("Insatisfecho", "Satisfecho")) +
  geom_text(aes(label=paste0(Total," ", "", "(", Porcentaje, "%",")")), vjust=0.5, color="black",
hjust=0.7, position = position_stack(vjust =0.5), angle=0, size=4.0) + scale_fill_discrete(name = "X1",
labels = c("Bueno", "Malo", "Regular")) +labs(title="Amabilidad y cortesía")
Tabla_X2<- Datos %>% group_by(Y, X2) %>% summarise(Total=n()) %>%
  dplyr::mutate(Porcentaje = round(Total/sum(Total)*100, 1))
G_X2<-ggplot(data=Tabla_X2, aes(x=Y, y=Porcentaje, fill=X2)) +
  geom_bar(width = 0.9, stat="identity")+ ylim(c(0,100))+
  labs(x="Niveles de satisfacción", y= "Frecuencia \n (Porcentajes)") + labs(fill = "")+
scale_x_discrete(labels=c("Insatisfecho", "Satisfecho")) +
  geom_text(aes(label=paste0(Total," ", "", "(", Porcentaje, "%",")")), vjust=0.5, color="black",
hjust=0.7, position = position_stack(vjust =0.5), angle=0, size=4.0) + scale_fill_discrete(name = "X2",
labels = c("Bueno", "Malo", "Regular")) + labs(title="Respeto")
Tabla_X3<- Datos %>% group_by(Y, X3) %>% summarise(Total=n()) %>%
  dplyr::mutate(Porcentaje = round(Total/sum(Total)*100, 1))
G_X3<-ggplot(data=Tabla_X3, aes(x=Y, y=Porcentaje, fill=X3)) +
  geom_bar(width = 0.9, stat="identity")+ ylim(c(0,100))+
  labs(x="Niveles de satisfacción", y= "Frecuencia \n (Porcentajes)") + labs(fill = "")+
scale_x_discrete(labels=c("Insatisfecho", "Satisfecho")) +
  geom_text(aes(label=paste0(Total," ", "", "(", Porcentaje, "%",")")), vjust=0.5, color="black",
hjust=0.7, position = position_stack(vjust =0.5), angle=0, size=4.0) + scale_fill_discrete(name = "X3",
labels = c("Bueno", "Malo", "Regular")) +labs(title="Disposición por atender")
Tabla_X4<- Datos %>% group_by(Y, X4) %>% summarise(Total=n()) %>%
  dplyr::mutate(Porcentaje = round(Total/sum(Total)*100, 1))
G_X4<-ggplot(data=Tabla_X4, aes(x=Y, y=Porcentaje, fill=X4)) +
  geom_bar(width = 0.9, stat="identity")+ ylim(c(0,100))+
  labs(x="Niveles de satisfacción", y= "Frecuencia \n (Porcentajes)") + labs(fill = "")+
scale_x_discrete(labels=c("Insatisfecho", "Satisfecho")) +
  geom_text(aes(label=paste0(Total," ", "", "(", Porcentaje, "%",")")), vjust=0.5, color="black",
hjust=0.7, position = position_stack(vjust =0.5), angle=0, size=4.0) + scale_fill_discrete(name = "X4",
labels = c("Bueno", "Malo", "Regular")) +labs(title="Confianza que inspira")
Tabla_X5<- Datos %>% group_by(Y, X5) %>% summarise(Total=n()) %>%
  dplyr::mutate(Porcentaje = round(Total/sum(Total)*100, 1))
G_X5<-ggplot(data=Tabla_X5, aes(x=Y, y=Porcentaje, fill=X5)) +
  geom_bar(width = 0.9, stat="identity")+ ylim(c(0,100))+
  labs(x="Niveles de satisfacción", y= "Frecuencia \n (Porcentajes)") + labs(fill = "")+
scale_x_discrete(labels=c("Insatisfecho", "Satisfecho")) +
  geom_text(aes(label=paste0(Total," ", "", "(", Porcentaje, "%",")")), vjust=0.5, color="black",
hjust=0.7, position = position_stack(vjust =0.5), angle=0, size=4.0) + scale_fill_discrete(name = "X5",
labels = c("Bueno", "Malo", "Regular")) +labs(title="Vestuario")
Tabla_X6<- Datos %>% group_by(Y, X6) %>% summarise(Total=n()) %>%
  dplyr::mutate(Porcentaje = round(Total/sum(Total)*100, 1))
G_X6<-ggplot(data=Tabla_X6, aes(x=Y, y=Porcentaje, fill=X6)) +
  geom_bar(width = 0.9, stat="identity")+ ylim(c(0,100))+
  labs(x="Niveles de satisfacción", y= "Frecuencia \n (Porcentajes)") + labs(fill = "")+
scale_x_discrete(labels=c("Insatisfecho", "Satisfecho")) +
  geom_text(aes(label=paste0(Total," ", "", "(", Porcentaje, "%",")")), vjust=0.5, color="black",
hjust=0.7, position = position_stack(vjust =0.5), angle=0, size=4.0) + scale_fill_discrete(name = "X6",
labels = c("Bueno", "Malo", "Regular")) +labs(title="Claridad de la información")
G_1 <- grid.arrange(G_X1, G_X2, G_X3, G_X4, G_X5, G_X6); G_1
```

```
Tabla_X7<- Datos %>% group_by(Y, X7) %>% summarise(Total=n()) %>%
  dplyr::mutate(Porcentaje = round(Total/sum(Total)*100, 1))
G_X7<-ggplot(data=Tabla_X7, aes(x=Y, y=Porcentaje, fill=X7)) +
  geom_bar(width = 0.9, stat="identity")+ ylim(c(0,100))+
  labs(x="Niveles de satisfacción", y= "Frecuencia \n (Porcentajes)") + labs(fill = "")+
scale_x_discrete(labels=c("Insatisfecho", "Satisfecho")) +
  geom_text(aes(label=paste0(Total," ", "", "(", Porcentaje, "%",")")), vjust=0.5, color="black",
hjust=0.7, position = position_stack(vjust =0.5), angle=0, size=4.0) +scale_fill_discrete(name = "X7",
labels = c("Bueno", "Malo", "Regular")) +labs(title="Amabilidad y cortesía")
Tabla_X8<- Datos %>% group_by(Y, X8) %>% summarise(Total=n()) %>%
  dplyr::mutate(Porcentaje = round(Total/sum(Total)*100, 1))
G_X8<-ggplot(data=Tabla_X8, aes(x=Y, y=Porcentaje, fill=X8)) +
  geom_bar(width = 0.9, stat="identity")+ ylim(c(0,100))+
  labs(x="Niveles de satisfacción", y= "Frecuencia \n (Porcentajes)") + labs(fill = "")+
scale_x_discrete(labels=c("Insatisfecho", "Satisfecho")) +
  geom_text(aes(label=paste0(Total," ", "", "(", Porcentaje, "%",")")), vjust=0.5, color="black",
hjust=0.7, position = position_stack(vjust =0.5), angle=0, size=4.0) + scale_fill_discrete(name = "X8",
labels = c("Bueno", "Malo", "Regular")) +labs(title="Respeto")
Tabla_X9<- Datos %>% group_by(Y, X9) %>% summarise(Total=n()) %>%
  dplyr::mutate(Porcentaje = round(Total/sum(Total)*100, 1))
G_X9<-ggplot(data=Tabla_X9, aes(x=Y, y=Porcentaje, fill=X9)) +
  geom_bar(width = 0.9, stat="identity")+ ylim(c(0,100))+
  labs(x="Niveles de satisfacción", y= "Frecuencia \n (Porcentajes)") + labs(fill = "")+
scale_x_discrete(labels=c("Insatisfecho", "Satisfecho")) +
  geom_text(aes(label=paste0(Total," ", "", "(", Porcentaje, "%",")")), vjust=0.5, color="black",
hjust=0.7, position = position_stack(vjust =0.5), angle=0, size=4.0) + scale_fill_discrete(name = "X9",
labels = c("Bueno", "Malo", "Regular")) +labs(title="Disposición por atender")
Tabla_X10<- Datos %>% group_by(Y, X10) %>% summarise(Total=n()) %>%
dplyr::mutate(Porcentaje = round(Total/sum(Total)*100, 1))
G_X10<-ggplot(data=Tabla_X10, aes(x=Y, y=Porcentaje, fill=X10)) +
  geom_bar(width = 0.9, stat="identity")+ ylim(c(0,100))+
  labs(x="Niveles de satisfacción", y= "Frecuencia \n (Porcentajes)") + labs(fill = "")+
scale_x_discrete(labels=c("Insatisfecho", "Satisfecho")) +
  geom_text(aes(label=paste0(Total," ", "", "(", Porcentaje, "%",")")), vjust=0.5, color="black",
hjust=0.7, position = position_stack(vjust =0.5), angle=0, size=4.0) +scale_fill_discrete(name =
"X10", labels = c("Bueno", "Malo", "Regular")) +labs(title="Confianza que inspira")
Tabla_X11<- Datos %>% group_by(Y, X11) %>% summarise(Total=n()) %>%
dplyr::mutate(Porcentaje = round(Total/sum(Total)*100, 1))
G_X11<-ggplot(data=Tabla_X11, aes(x=Y, y=Porcentaje, fill=X11)) +
  geom_bar(width = 0.9, stat="identity")+ ylim(c(0,100))+
  labs(x="Niveles de satisfacción", y= "Frecuencia \n (Porcentajes)") + labs(fill = "")+
scale_x_discrete(labels=c("Insatisfecho", "Satisfecho")) +
  geom_text(aes(label=paste0(Total," ", "", "(", Porcentaje, "%",")")), vjust=0.5, color="black",
hjust=0.7, position = position_stack(vjust =0.5), angle=0, size=4.0) +scale_fill_discrete(name =
"X11", labels = c("Bueno", "Malo", "Regular")) +labs(title="Vestuario")
Tabla_X12<- Datos %>% group_by(Y, X12) %>% summarise(Total=n()) %>%
dplyr::mutate(Porcentaje = round(Total/sum(Total)*100, 1))
G_X12<-ggplot(data=Tabla_X12, aes(x=Y, y=Porcentaje, fill=X12)) +
  geom_bar(width = 0.9, stat="identity")+ ylim(c(0,100))+
  labs(x="Niveles de satisfacción", y= "Frecuencia \n (Porcentajes)") + labs(fill = "")+
scale_x_discrete(labels=c("Insatisfecho", "Satisfecho")) +
  geom_text(aes(label=paste0(Total," ", "", "(", Porcentaje, "%",")")), vjust=0.5, color="black",
hjust=0.7, position = position_stack(vjust =0.5), angle=0, size=4.0) +scale_fill_discrete(name =
"X12", labels = c("Bueno", "Malo", "Regular")) +labs(title="Claridad de la información")
G_2 <- grid.arrange(G_X7, G_X8, G_X9, G_X10, G_X11, G_X12); G_2
Tabla_X13<- Datos %>% group_by(Y, X13) %>% summarise(Total=n()) %>%
dplyr::mutate(Porcentaje = round(Total/sum(Total)*100, 1))
```

```
G_X13<-ggplot(data=Tabla_X13, aes(x=Y, y=Porcentaje, fill=X13)) +
  geom_bar(width = 0.9, stat="identity")+ ylim(c(0,100))+
  labs(x="Niveles de satisfacción", y= "Frecuencia \n (Porcentajes)") + labs(fill = "")+
scale_x_discrete(labels=c("Insatisfecho", "Satisfecho")) +
  geom_text(aes(label=paste0(Total," ", "", "(", Porcentaje, "%",")")), vjust=0.5, color="black",
hjust=0.7, position = position_stack(vjust =0.5), angle=0, size=4.0) +scale_fill_discrete(name =
"X13", labels = c("Bueno", "Malo", "Regular")) +labs(title="Amabilidad y cortesía")
Tabla_X14<- Datos %>% group_by(Y, X14) %>% summarise(Total=n()) %>%
dplyr::mutate(Porcentaje = round(Total/sum(Total)*100, 1))
G_X14<-ggplot(data=Tabla_X14, aes(x=Y, y=Porcentaje, fill=X14)) +
  geom_bar(width = 0.9, stat="identity")+ ylim(c(0,100))+
  labs(x="Niveles de satisfacción", y= "Frecuencia \n (Porcentajes)") + labs(fill = "")+
scale_x_discrete(labels=c("Insatisfecho", "Satisfecho")) +
  geom_text(aes(label=paste0(Total," ", "", "(", Porcentaje, "%",")")), vjust=0.5, color="black",
hjust=0.7, position = position_stack(vjust =0.5), angle=0, size=4.0) +scale_fill_discrete(name =
"X14", labels = c("Bueno", "Malo", "Regular")) +labs(title="Respeto")
Tabla_X15<- Datos %>% group_by(Y, X15) %>% summarise(Total=n()) %>%
dplyr::mutate(Porcentaje = round(Total/sum(Total)*100, 1))
G_X15<-ggplot(data=Tabla_X15, aes(x=Y, y=Porcentaje, fill=X15)) +
  geom_bar(width = 0.9, stat="identity")+ ylim(c(0,100))+
  labs(x="Niveles de satisfacción", y= "Frecuencia \n (Porcentajes)") + labs(fill = "")+
scale_x_discrete(labels=c("Insatisfecho", "Satisfecho")) +
  geom_text(aes(label=paste0(Total," ", "", "(", Porcentaje, "%",")")), vjust=0.5, color="black",
hjust=0.7, position = position_stack(vjust =0.5), angle=0, size=4.0) + scale_fill_discrete(name =
"X15", labels = c("Bueno", "Malo", "Regular")) +labs(title="Disposición por atender")
Tabla_X16<- Datos %>% group_by(Y, X16) %>% summarise(Total=n()) %>%
dplyr::mutate(Porcentaje = round(Total/sum(Total)*100, 1))
G_X16<-ggplot(data=Tabla_X16, aes(x=Y, y=Porcentaje, fill=X16)) +
  geom_bar(width = 0.9, stat="identity")+ ylim(c(0,100))+
  labs(x="Niveles de satisfacción", y= "Frecuencia \n (Porcentajes)") + labs(fill =
"")+scale_x_discrete(labels=c("Insatisfecho", "Satisfecho")) +
  geom_text(aes(label=paste0(Total," ", "", "(", Porcentaje, "%",")")), vjust=0.5, color="black",
hjust=0.7, position = position_stack(vjust =0.5), angle=0, size=4.0) +scale_fill_discrete(name =
"X16", labels = c("Bueno", "Malo", "Regular")) +labs(title="Confianza que inspira")
Tabla_X17<- Datos %>% group_by(Y, X17) %>% summarise(Total=n()) %>%
dplyr::mutate(Porcentaje = round(Total/sum(Total)*100, 1))
G_X17<-ggplot(data=Tabla_X17, aes(x=Y, y=Porcentaje, fill=X17)) +
  geom_bar(width = 0.9, stat="identity")+ ylim(c(0,100))+
  labs(x="Niveles de satisfacción", y= "Frecuencia \n (Porcentajes)") + labs(fill = "")+
scale_x_discrete(labels=c("Insatisfecho", "Satisfecho")) +
  geom_text(aes(label=paste0(Total," ", "", "(", Porcentaje, "%",")")), vjust=0.5, color="black",
hjust=0.7, position = position_stack(vjust =0.5), angle=0, size=4.0) +scale_fill_discrete(name =
"X17", labels = c("Bueno", "Malo", "Regular")) +labs(title="Vestuario")
Tabla_X18<- Datos %>% group_by(Y, X18) %>% summarise(Total=n()) %>%
dplyr::mutate(Porcentaje = round(Total/sum(Total)*100, 1))
G_X18<-ggplot(data=Tabla_X18, aes(x=Y, y=Porcentaje, fill=X18)) +
  geom_bar(width = 0.9, stat="identity")+ ylim(c(0,100))+
  labs(x="Niveles de satisfacción", y= "Frecuencia \n (Porcentajes)") + labs(fill = "")+
scale_x_discrete(labels=c("Insatisfecho", "Satisfecho")) +
  geom_text(aes(label=paste0(Total," ", "", "(", Porcentaje, "%",")")), vjust=0.5, color="black",
hjust=0.7, position = position_stack(vjust =0.5), angle=0, size=4.0) +scale_fill_discrete(name =
"X18", labels = c("Bueno", "Malo", "Regular")) +labs(title="Claridad de la información")
G_3 <- grid.arrange(G_X13, G_X14, G_X15, G_X16, G_X17, G_X18); G_3
#  2. Métodos para la Selección de atributos (variables)
library("FSelector")
#  Métodos de filtrado
#  1. Método filtrado por Chi-Cuadrado (V. cualitativas)
```

```
Pesos <- chi.squared(Y~., Datos); print(Pesos)
Subset_1 <- cutoff.k(Pesos, 10)
F<-as.simple.formula(Subset_1, "Y"); print(F)
#  2. Basados en entropía: Ganancia de información
Pesos <- information.gain(Y~., Datos, unit="log2"); print(Pesos)
Subset_2 <- cutoff.k(Pesos, 10)
F <- as.simple.formula(Subset_2, "Y"); print(F)
#  3. Basados en entropía: Razón de ganancia
Pesos <- gain.ratio(Y~., Datos); print(Pesos)
Subset_3 <- cutoff.k(Pesos, 10)
F <- as.simple.formula(Subset_3, "Y"); print(F)
#  4. Método filtrado Relief
Pesos <- relief(Y~., Datos, neighbours.count =5, sample.size=10); print(Pesos)
Subset_4 <- cutoff.k(Pesos, 10)
F <- as.simple.formula(Subset_4, "Y"); print(F)
#  Métodos de Wrapper
#  Definición de algoritmo de evaluación
library(rpart)
evaluator <- function(subset) {
k <- 10 #k-fold cross validation
splits <- runif(nrow(Datos))
results = sapply(1:k, function(i) {
test.idx <- (splits >= (i - 1) / k) & (splits < i / k)
train.idx <- !test.idx
test <- Datos[test.idx, , drop=FALSE]
train <- Datos[train.idx, , drop=FALSE]
tree <- rpart(as.simple.formula(subset, "Y"), train)
error.rate = sum(test$Y != predict(tree, test, type="c")) / nrow(test)
return(1 - error.rate)
})
print(subset); print(mean(results)); return(mean(results))
}
#  5. Best-first search
Subset_5 <- best.first.search(names(Datos)[-19], evaluator)
F <- as.simple.formula(Subset_5, "Y"); print(F)
#  6. Greedy search: Forward
Subset_6 <- forward.search(names(Datos)[-19], evaluator)
F <- as.simple.formula(Subset_6, "Y"); print(F)
#  7. Greedy search: Backward
Subset_7 <- backward.search(names(Datos)[-19], evaluator)
F <- as.simple.formula(Subset_7, "Y"); print(F)
#  8. Hill climbing search
Subset_8 <- hill.climbing.search(names(Datos)[-19], evaluator)
F <- as.simple.formula(Subset_8, "Y"); print(F)
#  3. Técnicas de Minería de Datos
library(caret)
library(nnet)
library(C50)
library(bnclassify)
library(randomForest)
library(pROC)
#  Selección del conjunto de entrenamiento y de prueba
set.seed(123)
Indice<- createDataPartition(Datos$Y, p = 0.70, list = FALSE)
Datos_E <- Datos[Indice,]
Datos_P <- Datos[-Indice,]
f=table(Datos_E$Y); fr=round(prop.table(f)*100,1)
```

```
Tabla=as.data.frame(cbind(addmargins(f),addmargins(fr))); Tabla
f=table(Datos_P$Y); fr=round(prop.table(f)*100,1)
Tabla=as.data.frame(cbind(addmargins(f),addmargins(fr))); Tabla
#  Seleccionar las variables por cada método de filtrado y Wrapper
#  Datos_S=Datos_E   # Considerando todas las variables
Datos_S<-Datos_E[,c(Subset_8, "Y")]  #Considerando variables selecccionadas
str(Datos_S)
#  1. Regresión logística binaria
Modelo<-multinom(Y~., data=Datos_S)
Prediccion_C <-predict(Modelo, Datos_P, type="class")
Prediccion_P <-predict(Modelo, Datos_P, type="prob")
confusionMatrix(Prediccion_C, Datos_P$Y)
ROC <- roc(Datos_P$Y, Prediccion_P)
AUC <- auc(ROC)
G1=ggroc (ROC, color = 'red', size =0) + geom_abline(slope = 1, intercept = 1, linetype='dashed')+
theme_bw()+labs(title='Curvas ROC')+ ggtitle ( paste0 (' Curva ROC ', ' (AUC = ', round(AUC,3), ' )
'))+ theme_minimal ()
#  2. Árbol de clasificación C5.0
Arbol <- C5.0(Y~ ., data = Datos_S, rules=TRUE)
Prediccion_C <-predict(Arbol, Datos_P, type = "class")
Prediccion_P <-predict(Arbol, Datos_P, type="prob")
confusionMatrix(Prediccion_C, Datos_P$Y)
ROC <- roc(Datos_P$Y, Prediccion_P[,2])
AUC <- auc(ROC)
G1=ggroc (ROC, color = 'red', size =0) + geom_abline(slope = 1, intercept = 1, linetype='dashed')+
theme_bw()+labs(title='Curvas ROC')+ ggtitle ( paste0 (' Curva ROC ', ' (AUC = ', round(AUC,3), ' )
'))+ theme_minimal ()
#  3. Red bayesiana Naive de Bayes
Red <- bnc('nb', 'Y', Datos_S, smooth=1)
cv(Red, Datos_E, k=10)
Prediccion_C<- predict(Red, Datos_P, prob=FALSE)
Prediccion_P <-predict(Red, Datos_P, prob=TRUE)
confusionMatrix(Prediccion_C, Datos_P$Y)
ROC <- roc(Datos_P$Y, Prediccion_P[,2])
AUC <- auc(ROC); AUC
G1=ggroc (ROC, color = 'red', size =0) + geom_abline(slope = 1, intercept = 1, linetype='dashed')+
theme_bw()+labs(title='Curvas ROC')+ ggtitle ( paste0 (' Curva ROC ', ' (AUC = ', round(AUC,3), ' )
'))+ theme_minimal ()
#  4. Random Forest (Multiclasificador)
Parámetros<- data.frame(ntree=5000, mtry=6, max.depth=12)
Modelo <- randomForest(Y~ ., data=Datos_S, parame=Parámetros)
Prediccion_C <- predict(Modelo, Datos_P, type="class")
Prediccion_P <-predict(Modelo, Datos_P, type="prob")
confusionMatrix(Prediccion_C, Datos_P$Y)
ROC <- roc(Datos_P$Y, Prediccion_P[,2])
AUC <- auc(ROC)
G1=ggroc (ROC, color = 'red', size =0) + geom_abline(slope = 1, intercept = 1, linetype='dashed')+
theme_bw()+labs(title='Curvas ROC')+ ggtitle ( paste0 (' Curva ROC ', ' (AUC = ', round(AUC,3), ' )
'))+ theme_minimal ()
```