Benemérita Universidad Autónoma de Puebla

Facultad de Ciencias de la Computación

Clasificador Bayesiano Ingenuo en RapidMiner

Tesis

Que para obtener el Título de:

Maestra en Ciencias de la Computación

Presenta:

Imelda Hernández Baez

Asesora:

Dra. María de Lourdes Sandoval Solís

Puebla, Puebla. Septiembre, 2016

*A mi hija **Imelda***

Gracias por tus enseñanzas…

Agradecimientos

Gracias a **Dios** por darme la oportunidad de nacer y vivir… Gracias por bendecirme todos los días de mi vida y regalarme la mejor familia:

A mis padres **Lázara** y **Florencio**, por amarme, aceptarme y darme todo su paciencia y apoyo a lo largo de mi formación personal y profesional. Gracias por hacer de mí una mujer respetuosa y respetable. ¡Los quiero mucho!

A mis hermanos: **Luis**, **Roque**, **Elvia**, **Cecilia**, **Mariano** y **Guadalupe**, por su amor y apoyo incondicionales, porque no imagino mi vida sin mis seis hermanos… Gracias por darme la oportunidad de estudiar una carrera y lograr ahora esta meta, que parecía inalcanzable.

A mis sobrinos: **Alma**, **Luis**, **Jesús**, **Susana**, **Ana**, **Cecilia**, **Gerardo**, **Ricardo**, **Pilar**, **Ernesto**, **Belén**, **Rocío**, **Regina** y **Sandra**, compañeros de juego, de aprendizaje y de grandes aventuras; son una motivación para seguir preparándome. ¡Gracias por ser parte de mi vida!

A mi esposo **Jesús**, por todo su amor y los momentos inolvidables que compartimos en lo profesional y familiar… ¡Gracias por estar conmigo siempre, en todo!

A mi hija **Imelda**, Gracias por ser mi compañera, amiga, cómplice y confidente… ¡Gracias por enseñarme a valorar los grandes momentos y las grandes cosas que nos ofrece la vida!

Con admiración y respeto, a mi asesora, **Dra**. **María de Lourdes Sandoval Solís**, por su infinita paciencia, tiempo y dedicación para dirigirme.

Admiro su entrega y pasión por el trabajo y la forma de inspirar a sus alumnos para seguir aprendiendo y superándose cada día.

¡Gracias por todo!

A mis revisores de tesis: **Dra. Maya Carrillo Ruíz** y **Dr. Pedro García Juárez** por sus valiosas aportaciones y observaciones, que fueron de suma importancia para completar este trabajo. Gracias por todo su tiempo y dedicación.

Contenido

INTRODUCCIÓN **2**
CAPÍTULO 1 ..... **4**
CLASIFICACIÓN **4**
1.1 CLASIFICACIÓN BAYESIANA **6**
1.2 CLASIFICADOR BAYESIANO INGENUO **7**
1.3 CLASIFICADOR BAYESIANO GAUSSIANO **9**
1.4 CLASIFICADOR BAYESIANO INGENUO - KERNEL **12**
CAPÍTULO 2 ..... **15**
RAPIDMINER **15**
2.1 CLASIFICADOR BAYESIANO INGENUO EN RAPIDMINER **18**
2.2 CLASIFICADOR BAYESIANO INGENUO-KERNEL EN RAPIDMINER **23**
CAPÍTULO 3 ..... **27**
PRUEBAS **27**
Ejemplo 1. Clasificación de Sexo **29**
Ejemplo 2. Predicción sexo. CBI- Kernel. **31**
Ejemplo 3. PIMA Diabetes. **39**
Ejemplo 4. Clasificación de Vehículos **41**
Ejemplo 5: Base de datos Zoo. **45**
CONCLUSIONES **47**
BIBLIOGRAFÍA **49**

INTRODUCCIÓN

Clasificar cosas es parte de la vida desde que se es pequeño. Este proceso está implícito en muchas de nuestras actividades cotidianas, se clasifica la fruta como *verde* o *madura*, un auto como *último model*o o *clásico*, el médico clasifica a los pacientes con base en ciertos estudios o valoraciones físicas como *apto* o *no apto* para realizar una cirugía, etc. Catalogar objetos en distintas clases, a partir de un criterio determinado, es sumamente común, y muchas veces necesario.

Hoy en día, con las tecnologías de la información relacionadas casi a todos los aspectos de la vida diaria, se puede tener acceso a grandes cantidades de información que guardan las características más comunes para clasificar objetos. Por ejemplo, una universidad puede saber a qué categoría pertenece un alumno con base en algunos atributos especiales que le solicita, y/o que va observando y almacenando a lo largo de su vida estudiantil. Un alumno puede ser apto para otorgarle una beca o puede ser candidato para estudiar un posgrado, o en definitiva se sabe que no estará en la universidad en el próximo periodo.

Debido a esto, se necesitan herramientas que faciliten el proceso de clasificación de grandes cantidades de información, y que sea relativamente fácil catalogar personas u objetos con base en ciertos criterios. Una propuesta es el Clasificador Bayesiano Ingenuo (CBI), conocido también como *Naive Bayes*, que toma las características de cada objeto y supone que todas ellas son independientes entre sí y no afectan en la clasificación, además sólo requiere una pequeña cantidad de datos de entrenamiento para lograr un resultado exitoso.

Existen algunas aplicaciones para utilizar el CBI, una alternativa es *RapidMiner*, una aplicación de software libre, que tiene una interfaz sencilla y ofrece una gran cantidad de operadores no sólo para clasificación, sino para otras técnicas de análisis y minería de datos.

El **objetivo** de este trabajo es realizar un análisis del desempeño del Clasificador Bayesiano Ingenuo en el software de Minería de datos Rapidminer.

En el capítulo 1 se presenta el CBI, sus principales características, parámetros, así como su sencillez y eficiencia en la clasificación supervisada. Se analiza también una variante del clasificador, llamado CBI-Kernel, que mantiene las ventajas del CBI pero además se puede usar en situaciones donde el comportamiento de los datos no sigue una distribución normal. Ambos ofrecen una alternativa más en *RapidMiner* para lograr una clasificación exitosa.

En el capítulo 2 se presenta una breve descripción de *RapidMiner*, un poco de historia, interfaz, operadores, procesos, parámetros, etc. Así como la forma de ejecutar el CBI y CBI-Kernel, y analizar e interpretar el resultado.

En el capítulo 3 se realizan pruebas con CBI y CBI-Kernel para evaluar su eficiencia, utilizando varios ejemplos. Se usan bases de datos reales y académicas, variando el tamaño y el contenido de éstas. El objetivo es entrenar al clasificador con la base de datos completa y/o con diferentes tamaños de muestra, para luego probar su eficiencia al realizar la clasificación con nuevos datos. Se utilizan las opciones disponibles en *RapidMiner* para CBI y CBI-Kernel y se realiza una combinación con los diferentes parámetros para cada caso.

Finalmente se presentan las conclusiones al realizar este trabajo. Mostrando la eficiencia del CBI en *RapidMiner*, y resaltando las ventajas que se encuentran al usar este software para realizar la clasificación y obtener un resultado e interpretación muy simple, aún para usuarios no expertos en el tema.

CAPÍTULO 1

CLASIFICACIÓN

Clasificar consiste en asignar un objeto (instancia, dato) a una clase (categoría).
Por ejemplo, se puede clasificar una imagen como paisaje, retrato, urbana, etc. Otro ejemplo es asignar palabras a categorías gramaticales: sustantivo, verbo, adjetivo, etc.
El poder clasificar lo que se percibe con los sentidos es algo natural en el ser humano; esto permite abstraer la información, llevándola a una representación más adecuada para la toma de decisiones. [1]

La clasificación es también muy importante en el desarrollo de sistemas computacionales para muchas aplicaciones, por ejemplo:

- Diagnóstico médico: Dados algunos síntomas o características, saber si una persona es candidata a desarrollar ciertas enfermedades.
- Control de calidad en la industria: clasificar una pieza o producto como correcta o defectuosa.
- Sistemas de seguridad: identificar, por ejemplo, si una persona tiene acceso o no a cierto lugar.
- Lectores de correo electrónico: filtrar mensajes que sean "basura" (spam).
- Análisis de imágenes médicas: detectar tumores en rayos-X.
- Sistemas biométricos: asignar una imagen de una huella a la persona correspondiente.

Por lo tanto, es importante diseñar clasificadores que puedan ayudar a resolver dichos problemas.

Desde un punto de vista matemático, el proceso de clasificación consiste en asignar una clase, *c*, de un conjunto de clases, *C*, a cierta instancia, representada por un vector de características o atributos, $X = (X_1, X_2, \dots, X_M)$.

Hay dos tipos básicos de clasificadores:

No supervisado o **agrupamiento**: en este caso las clases son desconocidas, y el problema consiste en dividir un conjunto de *n* objetos en *k* clases, de forma que a objetos similares se les asigna la misma clase.
Supervisado: las clases se conocen a priori, y el problema consiste en encontrar una función que asigne a cada objeto su clase correspondiente.

Este trabajo se enfoca en clasificación supervisada. Entonces, el problema consiste en encontrar una función que realice un mapeo de los atributos del objeto a su clase correspondiente, esto es: $c = f(X)$. En general, es difícil construir dicha función, por lo que se utilizan técnicas de aprendizaje computacional para obtener la función a partir de datos –ejemplos de objetos en que se especifican sus características y la clase correspondiente. Este conjunto de datos, *D*, se compone de *n* ejemplos, cada uno a su vez compuesto de un vector de atributos y la clase correspondiente: $(X_1; c_1), \dots, (X_n;\ c_n)$.

Hay varios criterios en base a los cuales se evalúa un clasificador: [1]

- Exactitud: proporción de clasificaciones correctas.
- Rapidez: tiempo que toma hacer la clasificación.
- Claridad: qué tan comprensible es para los humanos.
- Tiempo de aprendizaje: tiempo para entrenar o ajustar el clasificador a partir de datos.

Las redes bayesianas, junto con los árboles de decisión y las redes neuronales artificiales, han sido los tres métodos más usados en aprendizaje automático durante los últimos años en tareas como la clasificación de documentos o filtros de mensajes de correo electrónico. Las redes bayesianas son un método importante no sólo porque ofrecen un análisis cualitativo de los atributos y valores que pueden intervenir en el problema, sino porque dan cuenta también de la importancia cuantitativa de esos atributos. En el aspecto cualitativo podemos representar cómo se relacionan esos atributos ya sea en una forma causal, o señalando simplemente la correlación que existe entre esas variables (o atributos). Cuantitativamente, da una medida probabilística de la importancia de esas variables en el problema (y por lo tanto una probabilidad explícita de las hipótesis que se formulan). [3]

1.1 CLASIFICACIÓN BAYESIANA

Desde un enfoque bayesiano, el problema de clasificación supervisada consiste en asignar a un objeto descrito por un conjunto de atributos o características, $X_1, X_2, \dots, X_n$, a una de m clases posibles, $c_1, c_2, \dots, c_m$, tal que la probabilidad de la clase dados los atributos se maximiza: [1]

$$Arg_C[MaxP(C|X_1, X_2, \dots, X_n)] \quad (1)$$

Si se denota al conjunto de atributos como: $X = \{X_1, X_2, \dots, X_n\}$, la expresión (1) se puede escribir como: $ArgC[MaxP(C|X)]$. La formulación del clasificador bayesiano se basa en utilizar la regla de Bayes para calcular la probabilidad posterior de la clase dados los atributos:

$$P(C|X_1, X_2, \dots, X_n) = \frac{P(C)P(X_1, X_2, \dots, X_n|C)}{P(X_1, X_2, \dots, X_n)} \quad (2)$$

Que se puede escribir de la forma:

$$P(C|X) = \frac{P(C)P(X|C)}{P(X)} \quad (3)$$

Entonces el problema de clasificación basado en la ecuación (3) se puede expresar como:

$$Arg_C\left[Max\left[P(C|X) = \frac{P(C)P(X|C)}{P(X)}\right]\right] \quad (4)$$

El denominador $P(X)$, no varía para las diferentes clases, por lo que se puede considerar como una constante si lo que interesa es maximizar la probabilidad de la clase:

$$Arg_C\big[Max[P(C|X) = \alpha P(C)P(X|C)]\big] \quad (5)$$

Para resolver un problema de clasificación bajo el enfoque bayesiano, se requiere la probabilidad *a priori* de cada clase, $P(C)$, y la probabilidad de los atributos dada la clase, $P(X|C)$, conocida como *verosimilitud*; para obtener la probabilidad *posterior* $P(C|A)$. En términos comunes, la ecuación se puede expresar como:

$$posterior = \frac{a\ priori * verosimilitud}{evidencia}$$

Entonces, para que este clasificador aprenda de un conjunto de datos, se requiere estimar estas probabilidades, a priori y verosimilitud, a partir de los datos, conocidos como los parámetros del clasificador.

La aplicación directa de la ecuación (5), resulta en un sistema muy complejo al implementarlo en una computadora, ya que el término $P(X_1, X_2, \dots, X_n|C)$, incrementa exponencialmente de tamaño en función del número de atributos; resultando en un requerimiento muy alto de memoria para almacenarlo, y también el número de operaciones para calcular la probabilidad crece significativamente. Una alternativa es considerar relaciones de independencia mediante lo que se conoce como el clasificador bayesiano simple, también conocido como Clasificador Bayesiano Ingenuo.

1.2 CLASIFICADOR BAYESIANO INGENUO

El Clasificador Bayesiano Ingenuo (CBI) se basa en la suposición de que todos los atributos son independientes dada la clase; esto es, cada atributo X_i es condicionalmente *independiente* de los demás atributos dada la clase: $P(X_1|X_j, C) = P(X_i|C), \forall j \neq i$. Considerando esto, la ecuación (2) se puede escribir como:

$$P(C|X_1, X_2, \dots, X_3) = \frac{P(C)P(X_1|C)P(X_2|C) \dots P(X_n|C)}{P(X)} \quad (6)$$

Donde $P(X)$ se puede considerar como una constante de normalización.

El CBI reduce significativamente la complejidad del clasificador bayesiano en espacio y tiempo de cálculo. En cuanto a espacio de memoria, se requiere la probabilidad previa de las *m* clases (vector de $1 \times m$), y las *n* probabilidades condicionales de cada atributo dada la clase (si suponemos que los atributos son discretos con *k* posibles valores, esto implica *n* matrices de $m \times k$). Básicamente el espacio requerido aumenta linealmente con el número de atributos. También el cálculo de la probabilidad posterior se vuelve muy eficiente, ya que se requieren del orden de *n* multiplicaciones para calcular la probabilidad posterior de cada clase dados los atributos (complejidad lineal).
Se puede representar gráficamente la estructura de un clasificador bayesiano simple utilizando los principios de los modelos gráficos probabilistas, donde las independencias condicionales entre las variables se representan mediante un grafo. El CBI tiene una estructura de estrella, con la clase en el medio y arcos dirigidos de la clase a cada atributo. Esto expresa que los atributos dependen de la clase y son independientes entre sí dada la clase (no hay arcos directos entre los atributos). Observe la figura 1.1.

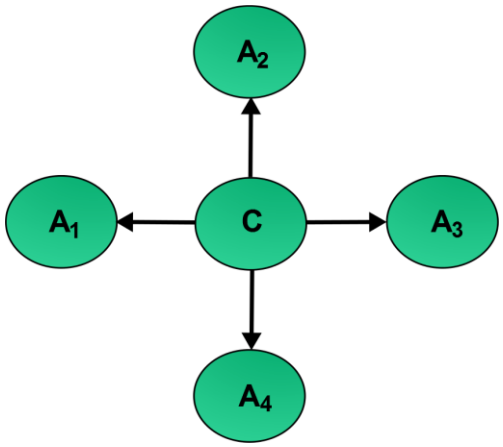


Figura 1.1. Representación gráfica de un CBI, una variable clase *C* y 4 atributos, A_1,.., A_4.

Para que un CBI aprenda se requiere la probabilidad previa de cada clase, $P(C)$, y la probabilidad condicional de cada atributo dada la clase, $P(X_i|C)$. Estas probabilidades se pueden obtener mediante estimados subjetivos de expertos en el área, o a partir de datos mediante máxima verosimilitud (consiste en que las probabilidades se aproximan por las estadísticas de los datos).

ESTIMACIÓN DE PARÁMETROS Y MODELOS DE EVENTOS

Todos los parámetros del modelo se pueden aproximar con frecuencias relativas del conjunto de entrenamiento. Estas son las estimaciones de máxima verosimilitud de las probabilidades. Una clase a priori se puede calcular asumiendo clases equiprobables (es decir, a priori = 1/ (número de clases)), o mediante el cálculo de una estimación de la probabilidad de clase del conjunto de entrenamiento (es decir, (el a priori de una clase dada) = (número de muestras en la clase) / (número total de muestras)). Para la estimación de los parámetros de la distribución de una característica, se debe asumir una distribución o generar modelos de estadística no paramétrica de las características del conjunto de entrenamiento.

Las hipótesis sobre las distribuciones de las características son llamadas el **Modelo de Eventos** del CBI. [2] La distribución *Multinomial* y la distribución de *Bernoulli* son populares para características discretas como las encontradas en la clasificación de documentos (incluyendo el filtrado de spam).

1.3 CLASIFICADOR BAYESIANO GAUSSIANO

Cuando se trata con los datos continuos, una hipótesis típica es que los valores continuos asociados con cada clase se distribuyen según una **Distribución Normal o Gaussiana**.
Por ejemplo, supongamos que los datos de entrenamiento contienen un atributo continuo, *x*. En primer lugar, se segmentan los datos por la clase, y a continuación, se calcula la media y la varianza de *x* en cada clase. Sea μ_c la media de *x* asociada a la clase *c*, y σ_c^2 la varianza de *x* asociada a la clase *c*. Entonces, la densidad de probabilidad de un cierto valor dada una clase, $P(x = v|c)$, se puede calcular agregando *v* en la ecuación de una distribución Normal con parámetros μ_c y σ_c^2. Es decir: [2]

$$P(x = v|c) = \frac{1}{\sqrt{2\pi\sigma_c^2}} e^{-\frac{(v-\mu_c)^2}{2\sigma_c^2}} \quad (7)$$

Se presenta a continuación un ejemplo para ilustrar el método.

EJEMPLO 1 – CLASIFICACIÓN DE SEXO

Para este ejemplo, se considera un grupo de personas originarias de la ciudad de Puebla, a las que se les solicitaron ciertas medidas sobre partes de su cuerpo. Las edades varían desde los 8 hasta los 88 años y se considera que la muestra es equilibrada.

El problema consiste en: Clasificar a una persona en hombre o mujer, basándose en las siguientes características: estatura, tamaño de pie, largo de brazo, ancho de espalda, perímetro de cráneo (medidas en centímetros), edad (entero) y peso (Kg). Se muestran a continuación los 12 datos iniciales de entrenamiento en la tabla 1.1.

Sexo	Estatura	Edad	Pie	Largo de brazo	Ancho de espalda	Perímetro de craneo	Peso
hombre	161	47	26	67	50	57	82
mujer	156	45	23	62	49	55	77
mujer	158	24	23	65	44	56	60
mujer	129	8	21	52	32	52	29
hombre	165	22	27	67	52	57	85
mujer	159	41	24	62	46	56	54.5
mujer	153	49	21.5	55	41	53.5	60
mujer	151	23	22.5	49	35	52.5	48
mujer	155	22	22	53	34	52	46
hombre	172	42	26.5	66	53	58.7	106
hombre	170	23	29	56	43	59	65
hombre	161	23	25	51.5	38	54.2	56

Tabla 1.1. Datos iniciales de entrenamiento para clasificación de sexo.

La probabilidad de que sea hombre es de $\frac{5}{12} = 0.417$ y la de que sea mujer es de $\frac{7}{12} = 0.583$.
Se calcula la media y la varianza de cada característica, obteniendo los resultados mostrados en la tabla 1.2:

	Estatura		Edad		Pie		Brazo		Espalda		Cráneo		Peso	
Sexo	Media	Varianza	Media	Varianza	Media	Varianza	Media	Varianza	Media	Varianza	Media	Varianza	Media	Varianza
hombre	165.8	25.7	31.4	146.3	26.7	2.2	61.5	52.8	47.2	41.7	57.2	3.6	78.8	374.7
mujer	151.6	106.6	30.3	223.2	22.4	1.0	56.9	37.1	40.1	43.1	53.9	3.2	53.5	220.8

Tabla 1.2. Medias y varianzas para los datos de entrenamiento.

Ahora se analiza el siguiente dato muestra para ser clasificado como hombre o mujer:

Sexo	Estatura	Edad	Pie	Largo de brazo	Ancho de espalda	Perímetro de craneo	Peso
muestra	153	48	24	49.5	41	53	69

Tabla 1.3. Dato muestra para clasificar como hombre o mujer.

Se calcula la probabilidad a posteriori para ambos casos, según la ecuación (6):

$$P(C|X_1, X_2, \dots, X_n) = \frac{P(C)P(X_1|C)P(X_2|C)\dots P(X_n|C)}{P(X)}$$

Para HOMBRE:

$$P(hombre|estatura, edad, \dots, peso) = \frac{P(hombre)P(estatura|hombre)P(edad|hombre)\dots P(peso|hombre)}{Evidencia}$$

Sabemos que $P(hombre) = 0.417$, para el primer caso *v*= 153, estatura del dato muestra, y se conocen la media y la varianza para cada característica, (observar la tabla 2.2), entonces se obtendrán todas las probabilidades para obtener el numerador a posteriori:

$$P(estatura|hombre) = \frac{1}{\sqrt{2\pi\sigma_c^2}} e^{-\frac{(153-\mu_c)^2}{2\sigma_c^2}}$$

$$P(estatura|hombre) = \frac{1}{\sqrt{2\pi * 25.7}} e^{-\frac{(153-165.8)^2}{2*25.7}}$$

$$P(\boldsymbol{estatura|hombre}) = 0.003247$$

Ahora *v*=48, edad del dato muestra:

$$P(edad|hombre) = \frac{1}{\sqrt{2\pi\sigma_c^2}} e^{-\frac{(48-\mu_c)^2}{2\sigma_c^2}}$$

$$P(edad|hombre) = \frac{1}{\sqrt{2\pi * 146.3}} e^{-\frac{(48-31.4)^2}{2*146.3}}$$

$$P(\boldsymbol{edad|hombre}) = 0.012861$$

Para MUJER:

$$P(estatura|mujer) = \frac{1}{\sqrt{2\pi\sigma_c^2}} e^{-\frac{(153-\mu_c)^2}{2\sigma_c^2}}$$

$$P(estatura|mujer) = \frac{1}{\sqrt{2\pi * 106.6}} e^{-\frac{(153-151.6)^2}{2*106.6}}$$

$$P(\boldsymbol{estatura}|\boldsymbol{mujer}) = 0.038268$$

$$P(edad|mujer) = \frac{1}{\sqrt{2\pi\sigma_c^2}} e^{-\frac{(48-\mu_c)^2}{2\sigma_c^2}}$$

$$P(edad|mujer) = \frac{1}{\sqrt{2\pi * 223.2}} e^{-\frac{(48-30.3)^2}{2*223.2}}$$

$$P(\boldsymbol{edad}|\boldsymbol{mujer}) = 0.013222$$

Se realizan los mismos cálculos para todas las características de hombres y mujeres, y se calcula el producto entre ellas, obteniendo los siguientes datos, mostrados en la figura 1.4:

	Estatura	Edad	Pie	Brazo	Espalda	Cráneo	Peso		Producto
hombre	0.003247945	0.012861	0.0513	0.01403	0.018988	0.052532	0.0181		5.4373E-13
mujer	0.038268039	0.013222	0.119	0.03159	0.060222	0.052092	0.0156		9.29777E-11

Tabla 1.4. Probabilidades de cada característica para ambos sexos.

Se observa que el numerador a posteriori más grande es el de la mujer, **9.2977E-11**, por lo tanto, de acuerdo a la ecuación (5), se determina que el dato muestra corresponde a una mujer. Lo cual es correcto, el dato muestra corresponde al sexo femenino.

1.4 CLASIFICADOR BAYESIANO INGENUO - KERNEL

Una variante del CBI es el Clasificador Bayesiano Ingenuo Kernel (CBI-Kernel), que mantiene las ventajas del CBI y además se puede aplicar en situaciones donde los datos no siguen una distribución normal.

Aunque en la bibliografía no existe una fecha exacta en la que se hayan utilizado los kernel, la mayoría de las referencias datan de los años noventa. Se han aplicado kernels en distintas áreas y en diferentes contextos, como Máquinas de Soporte Vectorial, Análisis de Correlaciones Canónicas, Análisis de patrones, etc. [14, 15] En este trabajo, un kernel es una función que se describirá a continuación.

Un **kernel** es una función de peso usada en técnicas de estimación no paramétrica. Los kernels son usados en Estimación de Densidad de Kernel (KDE, por sus siglas en inglés), para la estimación de la función de densidad de una variable aleatoria, o en regresión kernel, para estimar el valor esperado de una variable aleatoria. [7]

Los estimadores de densidad de kernel pertenecen a la clase de estimadores de densidad no paramétricos. A diferencia de los modelos paramétricos, que fijan completamente la distribución, excepto por el valor de uno o varios parámetros reales que deben ser estimados. El modelo paramétrico más utilizado es el normal. Sin embargo, hay muchas situaciones prácticas en que un sencillo análisis de los datos muestra claramente que la suposición de normalidad es inadecuada. [8]

Los estimadores de tipo kernel fueron diseñados para superar las dificultades al utilizar técnicas paramétricas en situaciones donde el comportamiento de los datos no sigue una distribución normal. Son los más utilizados en estimación no paramétrica.

Definición: **Estimador de densidad de kernel**. [9] Sea $(x_1, x_2, \dots, x_n)$ una muestra independiente e idénticamente distribuida trazada desde alguna distribución con una densidad no conocida f. Se está interesado en estimar la forma de esa función f. El estimador de densidad de kernel es:

$$\hat{f}_h(x) = \frac{1}{n}\sum_{i=1}^{n} K_h(x - x_i) = \frac{1}{nh}\sum_{i=1}^{n} K\left(\frac{x - x_i}{h}\right) \tag{8}$$

Donde $K(.)$ es el kernel – una función no negativa que se integra a uno y tiene media igual a cero– y $h > 0$ es un parámetro de suavizado llamado *ancho de banda*. Un kernel con subíndice h es llamado kernel escalado y se define como $K_h(x) = \frac{1}{h}K\left(\frac{x}{h}\right)$. Intuitivamente, se desea elegir h lo más pequeña posible como los datos lo permitan ($h \to 0$), para poder asegurar que $\hat{f}_h$ tiende a la verdadera densidad f de las variables x_i.

Las propiedades más importantes de estos estimadores no se ven afectadas por la función kernel que se elija. Frecuentemente se toma K como la función de densidad de la distribución normal estándar.

La elección correcta del parámetro h es el problema más difícil en la estimación no paramétrica. Si se elige demasiado pequeño, el estimador aparece "infrasuavizado", e incorpora demasiado "ruido", reflejado en la presencia de muchas modas (máximos relativos) que no aparecen en la densidad que se desea estimar. Por el contrario, si h se elige demasiado grande, se da el fenómeno contrario de "sobresuavizado" y el estimador es casi insensible a los datos. [8]

Existen varias formas que permiten asignar h de manera óptima. El criterio más común para seleccionar este parámetro es la función de riesgo L_2, también conocida como *Mean Integrated Squared Error (MISE)*:

$$MISE(h) = E\int (\hat{f}_h(x) - f(x))^2 dx \tag{9}$$

Si se usan funciones Gaussianas para aproximar datos univariados, y la densidad subyacente es Gaussiana, la elección óptima de h (que minimiza la MISE) es:

$$h = \left(\frac{4\hat{\sigma}^5}{3n}\right)^{\frac{1}{5}} \approx 1.06\hat{\sigma}n^{-\frac{1}{5}} \tag{10}$$

Donde $\hat{\sigma}$ es la desviación estándar de la muestra. Esta se denomina *aproximación Gaussiana* o la *Regla de Thumb de Silverman*. [9, 10]

A continuación se muestra gráficamente que la elección de *h* es de suma importancia para lograr la correcta densidad de los datos. Observe la figura 1.5, la línea gris corresponde a la distribución normal estándar, mientras que las líneas azul, verde y roja representan diferentes anchos de banda. En este caso, el valor de *h* que mejor representa la densidad de los datos es 0.3 y 0.1, mientras que el valor de 0.05 no es adecuado, ya que presenta el fenómeno de "infrasuavizado".

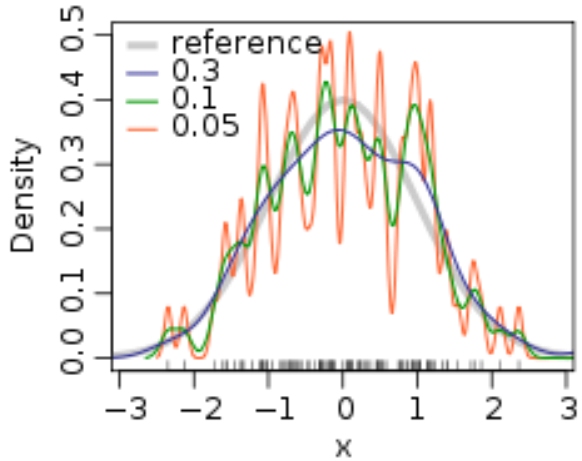


Figura 1.5 Distintos anchos de banda para estimación de densidad del kernel

Algunas funciones kernel de uso común [11, 12] se muestran en la tabla 1.6

	Funciones Kernel, $K(u)$	$\int u^2 K(u) du$	$\int K(u)^2 du$
Uniforme	$K(u) = \frac{1}{2} 1_{\{\lvert u \rvert \leq 1\}}$	$\frac{1}{3}$	$\frac{1}{2}$
Triangular	$K(u) = (1 - \lvert u \rvert) 1_{\{\lvert u \rvert \leq 1\}}$	$\frac{1}{6}$	$\frac{2}{3}$
Epanechnikov	$K(u) = \frac{3}{4}(1 - u^2) 1_{\{\lvert u \rvert \leq 1\}}$	$\frac{1}{5}$	$\frac{3}{5}$
Gaussiana	$K(u) = \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}u^2}$	1	$\frac{1}{2\sqrt{\pi}}$
Coseno	$K(u) = \frac{\pi}{4} \cos\left(\frac{\pi}{2} u\right) 1_{\{\lvert u \rvert \leq 1\}}$	$1 - \frac{8}{\pi^2}$	$\frac{\pi^2}{16}$

Tabla 1.6 Funciones Kernel más comunes.

CAPÍTULO 2

RAPIDMINER

RapidMiner es un programa informático para el análisis y minería de datos. Permite el desarrollo de procesos de análisis de datos mediante el encadenamiento de operadores a través de un entorno gráfico. Se usa en investigación y en aplicaciones empresariales. [7]

El desarrollo de *RapidMiner* inició bajo el nombre de "Yet Another Learning Environment" (YALE) en el departamento de Inteligencia Artificial de la Universidad de Dortmund, Alemania, bajo la dirección de la Dra. Katharina Morik. El software se volvió más y más robusto conforme pasó el tiempo, más de un millón y medio de descargas se han registrado desde que inició su desarrollo, en 2001. [7]
RapidMiner proporciona más de 500 operadores orientados al análisis de datos, incluyendo los necesarios para realizar operaciones de entrada y salida, pre procesamiento de datos y visualización. Puede ser descargado desde el sitio: **http://www.rapidminer.com**

La interfaz gráfica de usuario tiene como objetivo simplificar el uso para las tareas complejas de esta área. A continuación se presentan sus **características** más relevantes:

- Desarrollado en Java
- Multiplataforma
- Representación interna de los procesos de análisis de datos en archivos XML
- Permite el desarrollo de programas a través de un lenguaje script
- Puede usarse de diversas maneras:
    - A través de un GUI
    - En línea de comandos
    - Desde otros programas a través de llamadas a sus bibliotecas
- Extensible
- Incluye gráficos y herramientas de visualización de datos
- Dispone de un módulo de integración con R

RapidMiner es el líder mundial de código abierto para la minería de datos debido a la combinación de su tecnología de primera calidad y su rango de funcionalidad. [7] Cubre un amplio rango de conceptos de minería de datos, además de ser una herramienta flexible para aprender y explorar.

PREPARACIÓN DE LOS DATOS

Los datos de entrada en RapidMiner deben almacenarse en la carpeta *Local Repository*, para que el sistema les dé el formato adecuado para su procesamiento. Se pueden importar datos en diferentes formatos: Excel, CSV, XML, binarios; así como tablas de bases de datos (*MySQL*, *PostgreSQL*, *Ingress*, *Oracle*, etc.)
En este trabajo, los datos se manejarán en formato *Excel*.

LA INTERFAZ DE *RapidMiner* 5.3

RapiMiner proporciona una interfaz gráfica de usuario muy sencilla. Si se desea crear un nuevo proceso, basta con seleccionar el icono *New Process*, y se abrirá la ventana de diseño. Se describirán a continuación sus principales elementos.

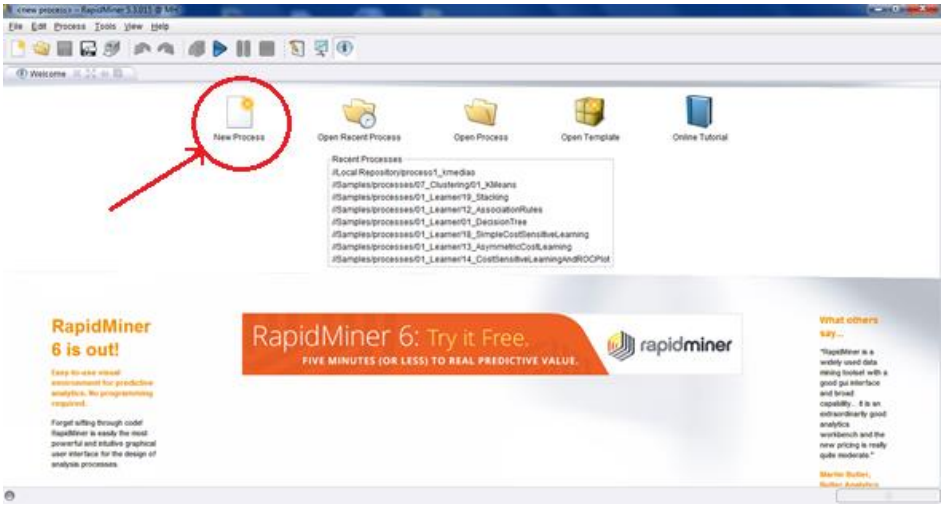


Figura 2.1. La interfaz principal de RapidMiner

En la ventana para crear nuevos procesos, se pueden observar diferentes pestañas: Operadores, Repositorios, Procesos, Parámetros, Ayuda, Comentarios, etc. (Ver figura 2.2)
En esta ventana se llevará a cabo el diseño de los procesos de minería de datos, desde la preparación de los datos hasta la selección y configuración de la técnica que se desee usar.
El diseño se lleva a cabo en la ventana principal, llamada "*Main Process*"

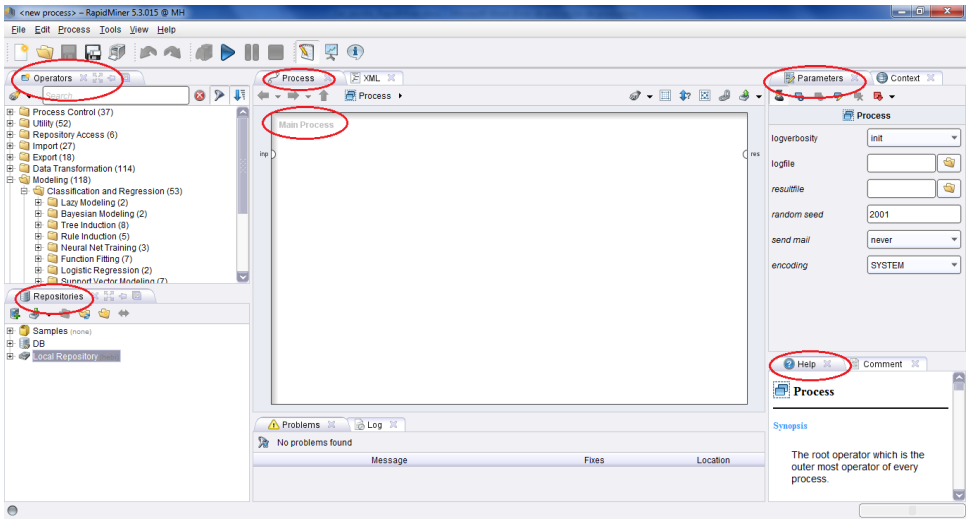


Figura 2.2 Principales componentes de RapidMiner

Para iniciar un proceso, se necesita "abrir" la base de datos, esto se hace en la ventana *Operators*, dentro de la carpeta llamada *Repository Access*, se elige el operador *Retrieve* y se arrastra a la ventana *Main Process*.

Una vez que se tiene el operador *Retrieve* en la ventana de diseño, se procede a localizar el archivo que contiene los datos, dando *click* en la carpeta *Repository entry*, ubicada en el panel derecho de la ventana principal. (Ver figura 2.3)

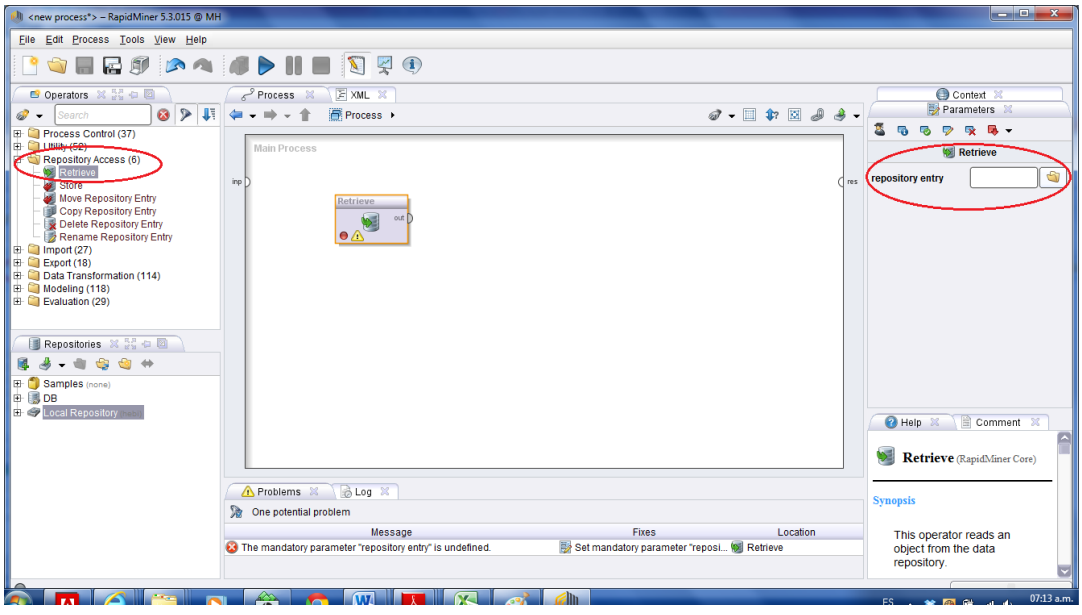


Figura 2.3 El operador *Retrieve*

A partir de este paso, ya se puede elegir el operador que se desee para realizar minería de datos.

2.1 CLASIFICADOR BAYESIANO INGENUO EN RAPIDMINER

Ahora se describirá el funcionamiento del operador *Naive Bayes*, correspondiente al Clasificador Bayesiano Ingenuo (CBI) en *RapidMiner*, usando el ejemplo 1 de datos reales de personas para clasificarlos como hombre o mujer, analizado anteriormente (página 10).
Los datos de entrada en *Rapidminer* se pueden importar de diferentes aplicaciones. Particularmente, se trabajará con datos importados desde Excel. En la opción *File* de la barra de menús, seleccionar la opción *Import Data* → *Import Excel Sheet*; seleccionar el archivo en formato Excel y seguir los pasos para completar la importación. Es muy importante guardar el archivo resultante en la carpeta *Local Repository*, ya que RapidMiner sólo considera como archivos de entrada listos para usarse aquellos que están alojados en esta ubicación
Una vez que se tiene en la carpeta *Local Repository* el archivo con el que se va a trabajar, se realizan los siguientes pasos: [7]

1. Agregar el operador **Repository Access** → **Retrieve** a la zona de trabajo y localizar el archivo */Local Repository/12sexos_reales* con el navegador del parámetro *repository entry*.
2. Agregar el operador **Data Transformation** → **Name and Role Modification** → **Set Role** y conectar la salida del operador **Retrieve** a la entrada *exa* del operador **Set Role**. Una vez que se tienen conectados, hay que definir el atributo que se usará para realizar la predicción. En este caso, el parámetro *attribute name* es *clase* y en la opción *target rol* se seleccionará *label*.
3. Agregar el operador **Modeling** → **Classification and Regression** → **Bayesian Modeling** → **Naive Bayes**. El parámetro *laplace correction* indica si se debe usar la corrección de Laplace para prevenir la alta influencia de probabilidades cero. Conectar la salida *exa* del operador **Set Role** a la entrada *tra* de este operador y las salidas *mod* y *exa* a los conectores respectivos **res** del panel.
4. Ejecutar el proceso dando click en el icono  de la barra de herramientas.

En las figuras 2.4 a), b) y c) se muestra el proceso.

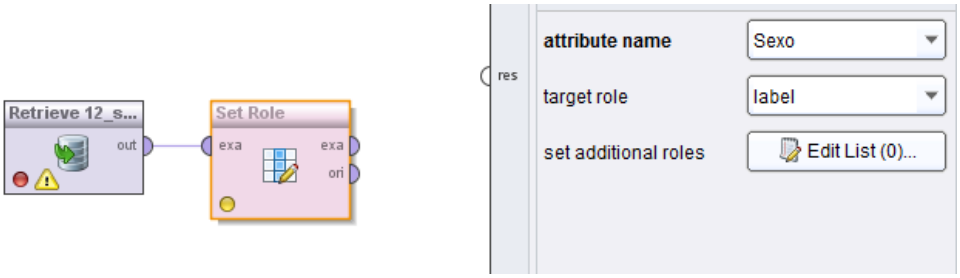


Figura 2.4 a) El operador Set Role y sus atributos

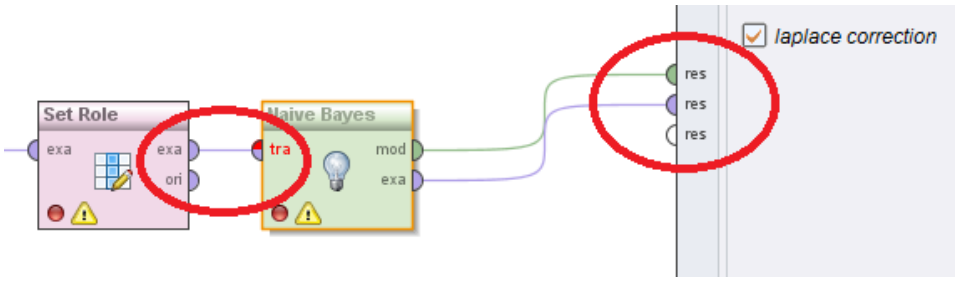


Figura 2.4 b) El operador Naive Bayes: sus conectores y atributos

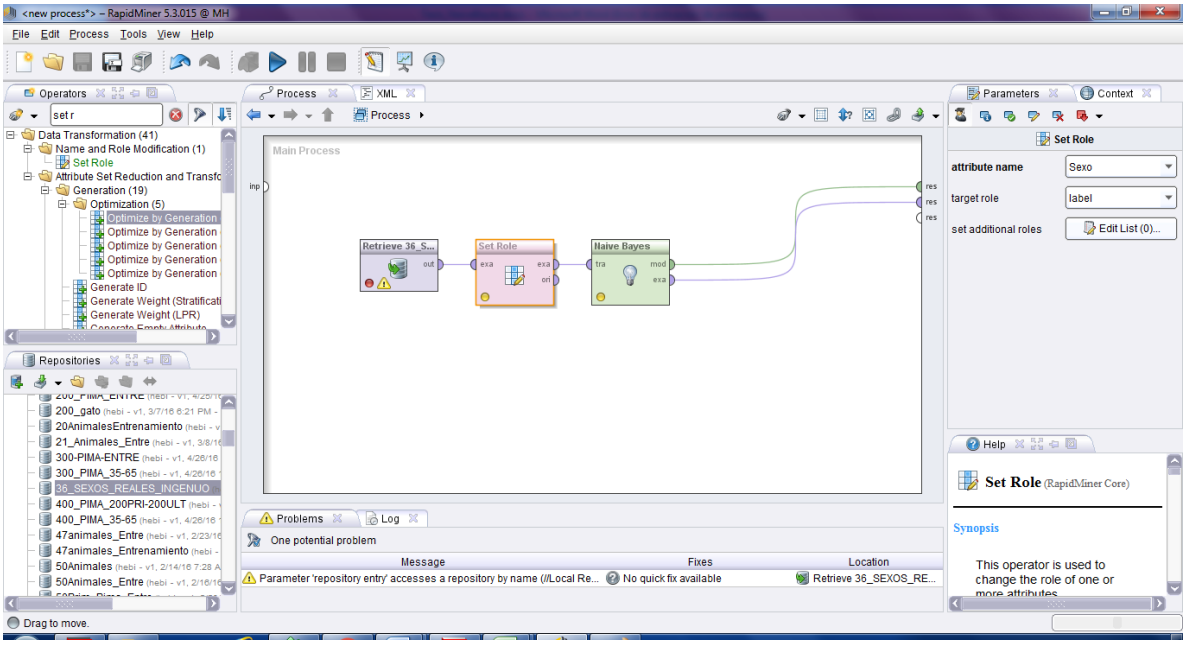


Figura 2.4 c) El Clasificador Bayesiano Ingenuo en *RapidMiner*

Como salida de este operador se obtendrá la distribución por clase *sexo*, la tabla de distribuciones, y las gráficas de densidad correspondientes para cada atributo. Las figuras 2.5 a), b) y c) muestran el resultado.

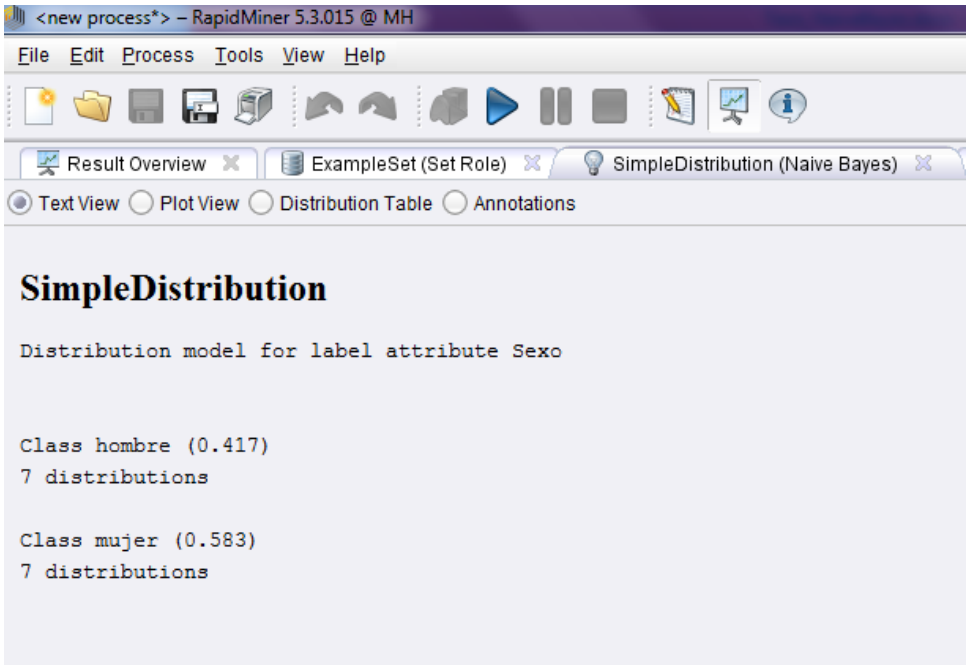
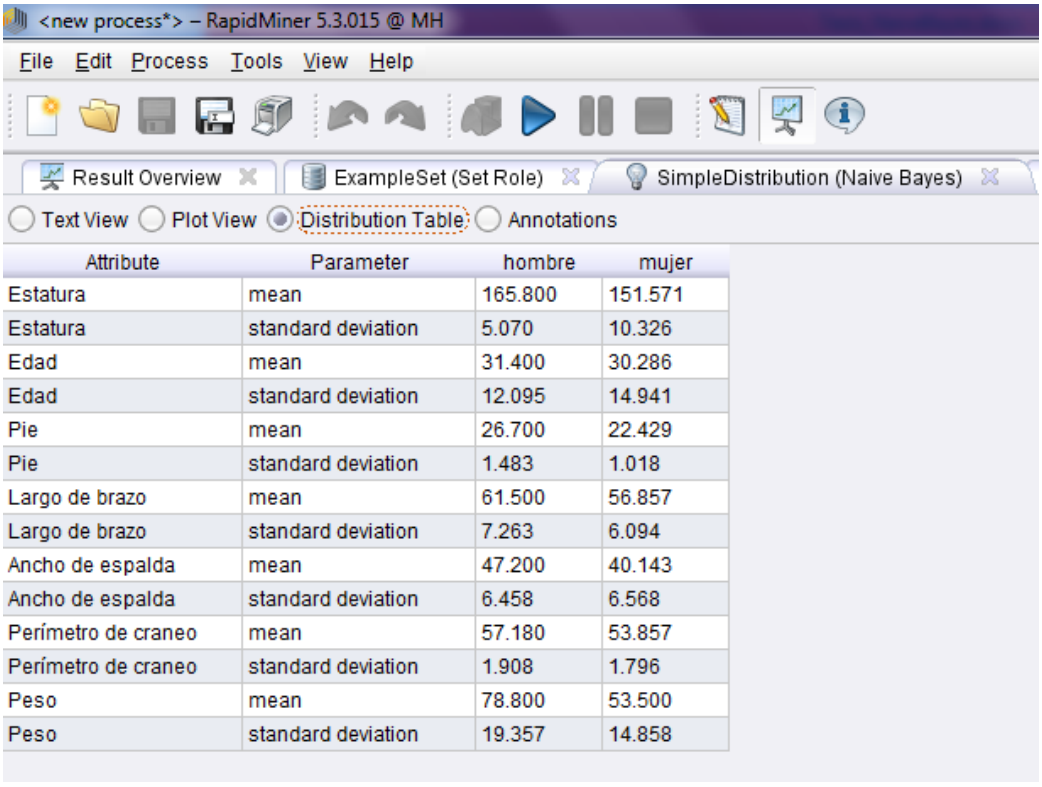


Figura 2.5 a) Distribución por clase *sexo*



Attribute	Parameter	hombre	mujer
Estatura	mean	165.800	151.571
Estatura	standard deviation	5.070	10.326
Edad	mean	31.400	30.286
Edad	standard deviation	12.095	14.941
Pie	mean	26.700	22.429
Pie	standard deviation	1.483	1.018
Largo de brazo	mean	61.500	56.857
Largo de brazo	standard deviation	7.263	6.094
Ancho de espalda	mean	47.200	40.143
Ancho de espalda	standard deviation	6.458	6.568
Perímetro de craneo	mean	57.180	53.857
Perímetro de craneo	standard deviation	1.908	1.796
Peso	mean	78.800	53.500
Peso	standard deviation	19.357	14.858

Figura 2.5 b) Tabla de distribución para cada atributo

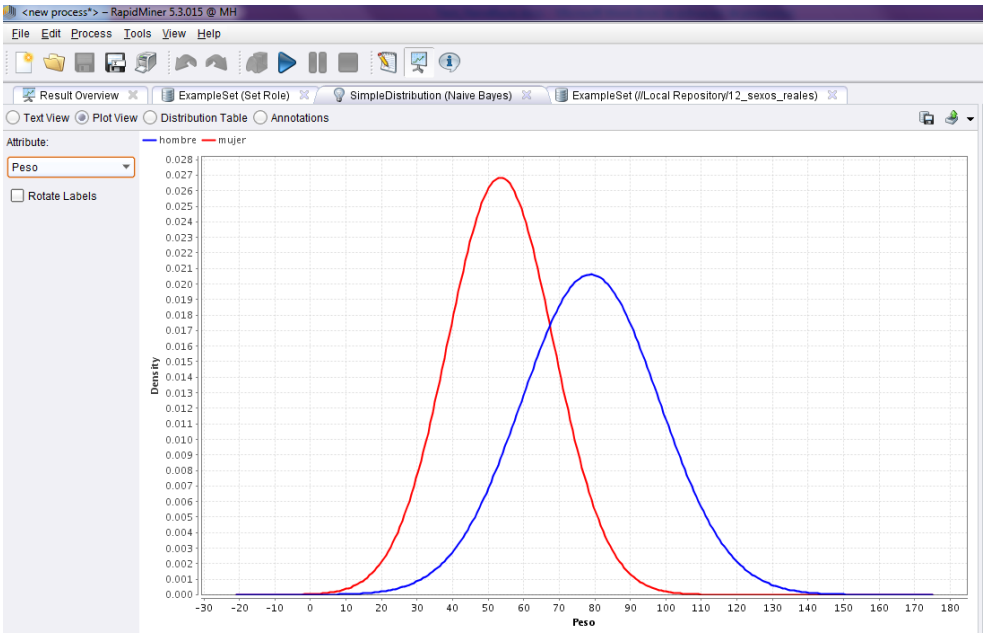


Figura 2.5 c) Gráfica de densidad para el atributo "peso"

Una vez que se realizan estos pasos, el CBI ya ha sido entrenado y está listo para probarse con una base de datos muestra, y saber si la clasificación obtenida es correcta. A continuación se describen los pasos a seguir para este procedimiento.

Una vez que se siguieron los pasos 1 a 4 (mostrados en la figura 2.4), se procede ahora a agregar la base de datos muestra para conectarla con el CBI y aplicar el modelo con los nuevos datos. Para esto, se deben seguir los siguientes pasos:

1. Agregar el operador **Repository Access → Retrieve** a la zona de trabajo y localizar el archivo */Local Repository/3_sexos_muestra* con el navegador del parámetro *repository entry*.
2. Agregar el operador **Modeling → Model Application → Apply Model**.
3. Conectar la salida del operador Retrieve con la entrada *uni* del operador **Apply Model**; y la salida *mod* del operador **Naive Bayes** a la entrada *mod* del operador **Apply Model**.
4. Finalmente, conectar la salida *exa* del operador **Naive Bayes** a la salida *res* del panel y las salidas *lab* y *mod* del operador **Apply Model** a las respectivas salidas *res* del panel.
5. Ejecutar el modelo dando click en el icono de la barra de herramientas.

La figura 2.6 muestra el proceso completo.

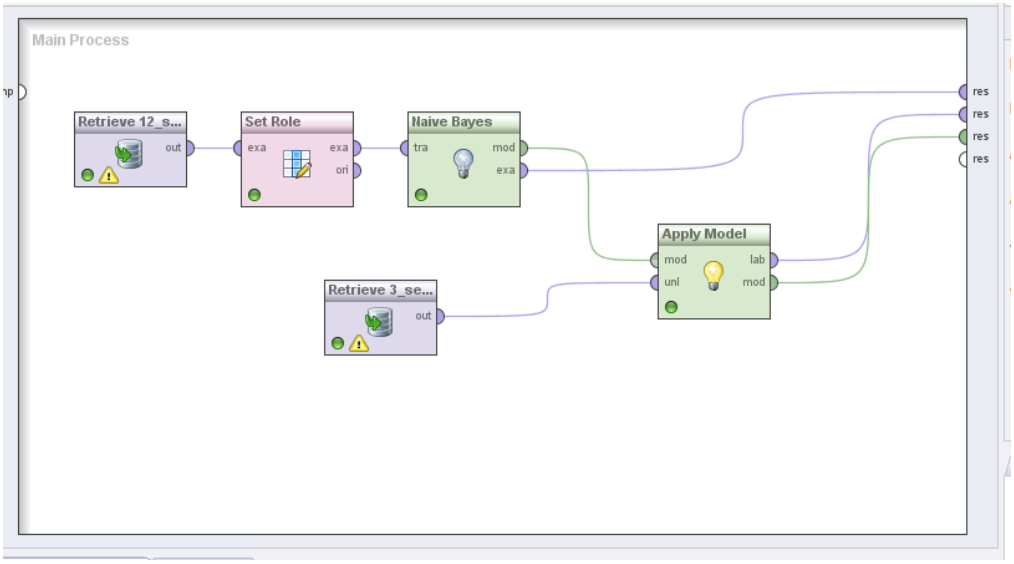


Figura 2.6 Proceso completo de entrenamiento del CBI

Como resultado se obtiene la tabla de predicción para cada elemento muestra, además de la distribución por clase *sexo*, la tabla de distribuciones, y las gráficas de densidad correspondientes para cada atributo; como en el primer modelo. La figura 2.7 muestra el resultado.

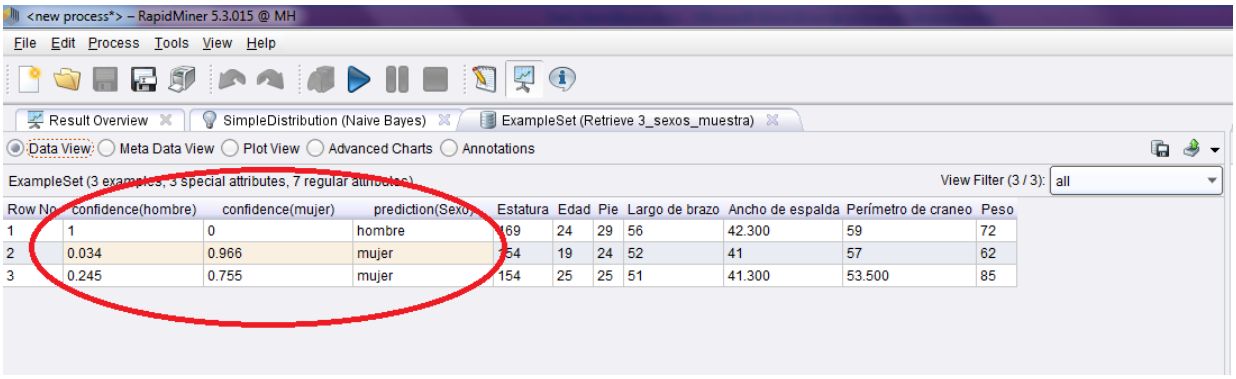


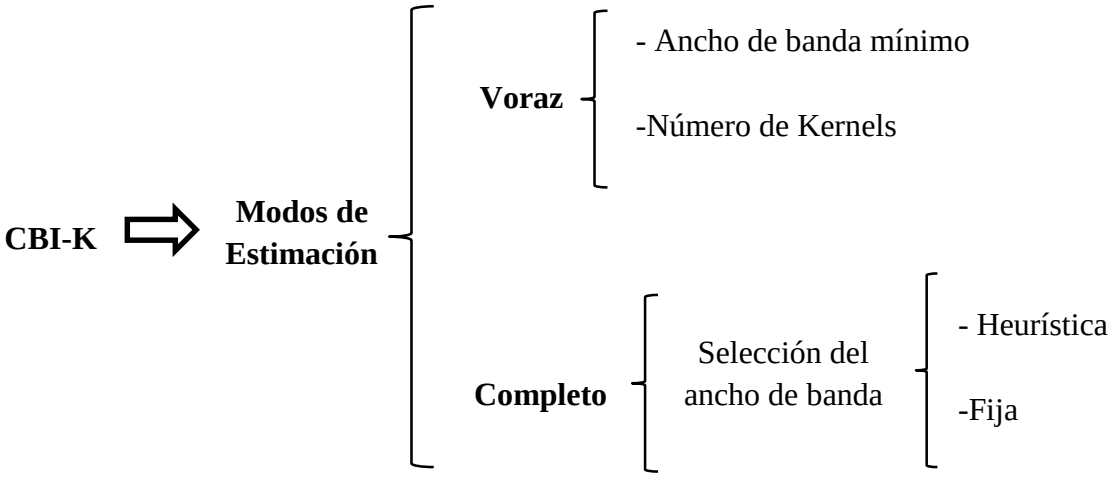
Figura 2.7 Tabla de predicción del CBI para cada elemento muestra.

Como se puede observar en la figura 2.7, la tabla de predicciones muestra la "confidencia" de cada elemento de la muestra para ser clasificado como *hombre* o *mujer*, en este caso, el resultado obtenido es correcto, los datos muestra corresponden a un hombre y dos mujeres.

2.2CLASIFICADOR BAYESIANO INGENUO-KERNEL EN RAPIDMINER

En este apartado se presenta de manera general el Clasificador Bayesiano Ingenuo-Kernel (CBI-K), cuyo operador en *RapidMiner* se denomina como *Naive Bayes Kernel*, se describen sus parámetros de entrada, datos de salida y la forma de construcción del modelo. Se tomará como referencia el ejemplo de clasificación de sexos, con una variante, ahora la base de datos completa consta de 36 datos iniciales.

A diferencia del CBI, que sólo necesita la corrección de Laplace como parámetro de entrada, el CBI-K ofrece 2 formas de estimación: *greedy* (voraz) y *full* (completa).
Para cada forma, varían los parámetros necesarios, el cuadro C1 muestra un resumen de las dos formas de estimación y sus requerimientos.



Cuadro C1. Opciones de estimación para el CBI-K en *RapidMiner*

Para ejecutar el operador CBI-K, se seguirán los siguientes pasos:

i) Para el modo de estimación **Voraz**:

1. Agregar el operador **Repository Access → Retrieve** a la zona de trabajo y localizar el archivo */Local Repository/36_sexos_reales* con el navegador del parámetro *repository entry*.
2. Agregar el operador **Data Transformation → Name and Role Modification → Set Role** y conectar la salida del operador **Retrieve** a la entrada *exa* del operador **Set Role**. Una vez que se tienen conectados, hay que definir el atributo que se usará para realizar la predicción. En este caso, el parámetro *attribute name* es *clase* y en la opción *target rol* se seleccionará *label*.
3. Agregar el operador **Modeling → Classification and Regression → Bayesian Modeling → Naive Bayes Kernel**. Seleccionar el modo de estimación voraz y elegir el mínimo ancho de banda y el número de kernels. RapidMiner tiene valores por omisión para estos

parámetros, para este ejemplo, se usarán los valores: ancho de banda=5 y número de kernels=10.

4. Conectar la salida del operador **Set Role** a la entrada *tra* del operador **Naive Bayes Kernel**.
5. Agregar el operador **Repository Access → Retrieve** a la zona de trabajo y localizar el archivo */Local Repository/3_sexos_muestra* con el navegador del parámetro *repository entry*.
6. Agregar el operador **Modeling → Model Application → Apply Model**.
7. Conectar la salida del operador **Retrieve**/*3_sexos_muestra* con la entrada *uni* del operador **Apply Model**; y la salida *mod* del operador **Naive Bayes Kernel** a la entrada *mod* del operador **Apply Model**.
8. Finalmente, conectar la salida *exa* del operador **Naive Bayes** a la salida *res* del panel y las salidas *lab* y *mod* del operador **Apply Model** a las respectivas salidas *res* del panel.
9. Ejecutar el proceso dando *click* en el icono  de la barra de herramientas.

El proceso completo se muestra en la figura 2.8

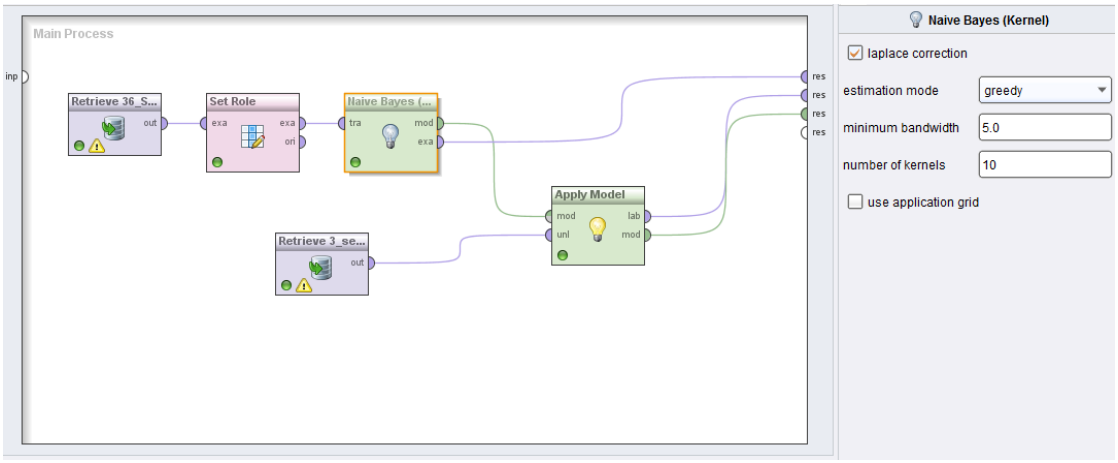


Figura 2.8 Proceso completo del operador CBI-K en RapidMiner

Como salida se obtiene la distribución Kernel para la clase *sexo*, 0.611 para *hombre* y 0.389 para *mujer*.

Kernel Distribution

```
Distribution model for label attribute Sexo

Class hombre (0.611)
7 distributions

Class mujer (0.389)
7 distributions
```

Las gráficas correspondientes para cada atributo y la tabla de predicción para los datos muestra se pueden observar en las figuras 2.9 a) y b).

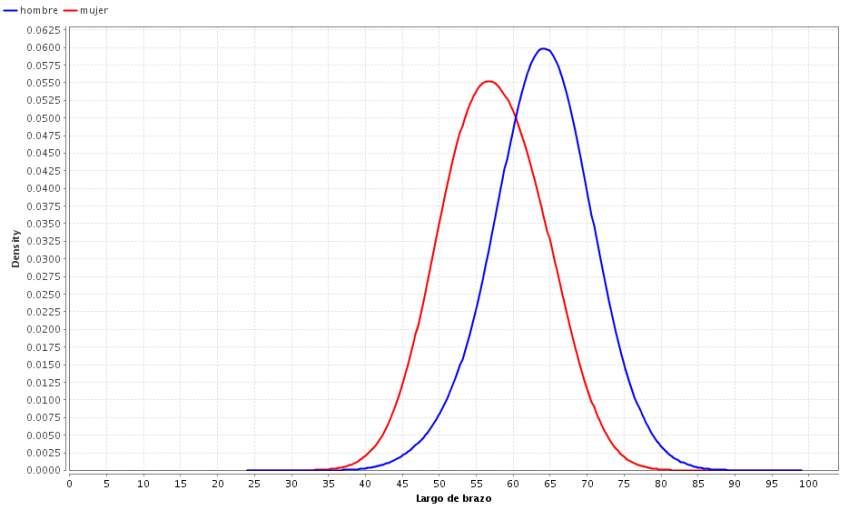
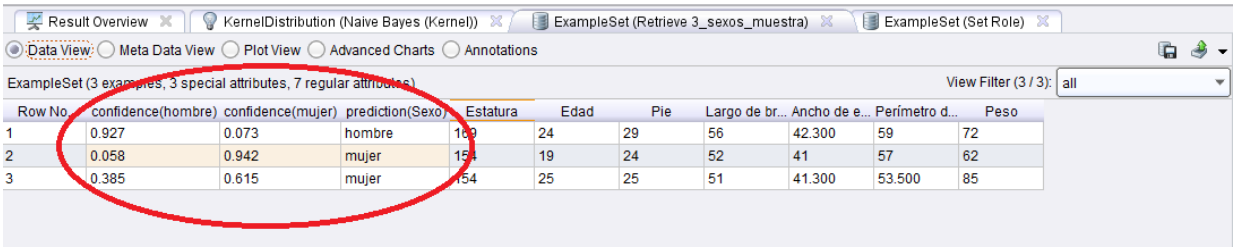


Figura 2.9 a) Gráfica de densidad para el atributo "largo de brazo"



Row No.	confidence(hombre)	confidence(mujer)	prediction(Sexo)	Estatura	Edad	Pie	Largo de br...	Ancho de e...	Perímetro d...	Peso
1	0.927	0.073	hombre	169	24	29	56	42.300	59	72
2	0.058	0.942	mujer	154	19	24	52	41	57	62
3	0.385	0.615	mujer	154	25	25	51	41.300	53.500	85

Figura 2.9 b) Tabla de predicción para los datos muestra, usando el CBI-K

Como se observa en la figura 2.9 b), la predicción es correcta, la muestra consta de un hombre y 2 mujeres.

ii) Para el modo de estimación **Completo**:

Seguir los pasos 1 al 9 mostrados anteriormente para el modo voraz, solo que ahora con el modo completo se requiere sólo un parámetro, que es el ancho de banda mínimo. Esta puede elegirse fija o heurística. Las figuras 2.10 a) y b) muestran el proceso completo y los parámetros necesarios para cada opción.

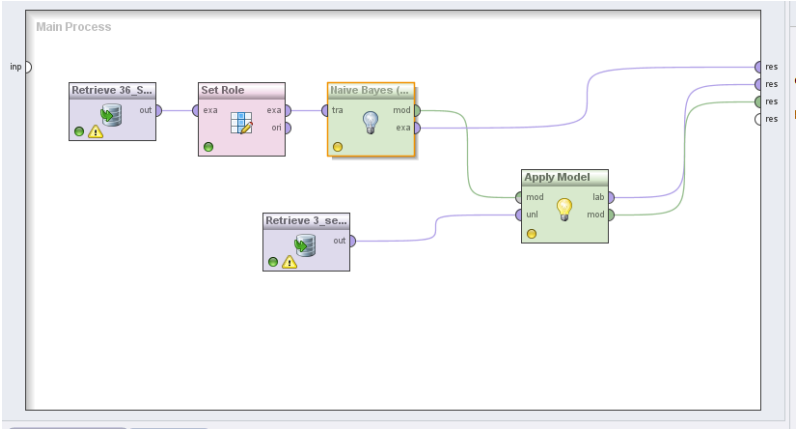
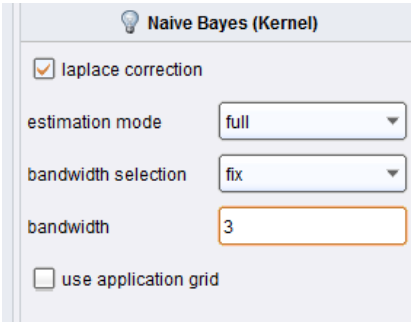


Figura 2.10 a) Proceso completo del operador CBI-K



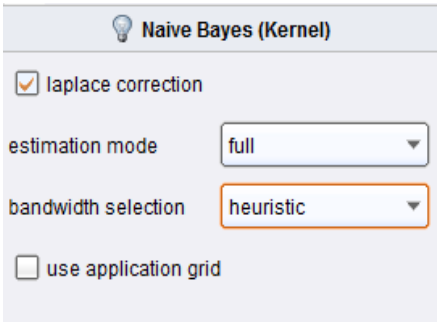


Figura 2.10 b) Opciones para el modo de estimación Completo: ancho de banda **fija** o **heurística**

Se ejecutó el operador con ambas opciones y se obtuvo un buen desempeño, considerando el valor del ancho de banda que se elija. Para un análisis completo, consulte el ejemplo 2 del capítulo PRUEBAS.

CAPÍTULO 3

PRUEBAS

En este capítulo se presentan las pruebas realizadas en *RapidMiner* utilizando el Clasificador Bayesiano Ingenuo y Kernel. Se utilizaron distintas bases de datos para realizar las pruebas y verificar la eficiencia del clasificador, de acuerdo a los distintos tipos de datos y al número de clases en cada caso.

En el ejemplo 1, se usó el CBI para predecir el *sexo* de una muestra de 3 personas, dada una base de datos de entrenamiento que consta de 36 elementos. Esta es la prueba más sencilla del uso del clasificador, que se presenta para comprender su funcionamiento.

En el ejemplo 2 se usa la base de datos de 36 personas y se usa el CBI-Kernel en sus dos formas de estimación, realizando pruebas con distintos valores de ancho de banda y número de kernels.

El ejemplo 3 consta de varias pruebas con distintos tamaños de bases de datos de entrenamiento inicial, para realizar la clasificación de un total de 768 pacientes que pueden ser clasificados como *positivo* o *negativo* para tener diabetes mellitus.

Los ejemplo 4 y 5 se llevan a cabo con bases de datos que realizan la clasificación con más de dos clases, para hacer notar la diferencia en el desempeño del clasificador al aumentar las categorías utilizadas.

En el ejemplo 4 se usa una base de datos de siluetas de vehículos y se realizan pruebas con distintos tamaños de muestra usando los clasificadores CBI y CBI-Kernel para obtener una clasificación de 4 siluetas. En este caso se observa que los datos no siguen una distribución normal, lo que seguramente afectará la eficiencia del clasificador

Finalmente, en el ejemplo 5, se usa el CBI para clasificar una base de datos de animales, con atributos nominales y 7 clases. Se eligió este caso para observar el desempeño del clasificador con datos no numéricos y un número de clase mayor a 2.

En todos los casos la eficiencia del clasificador se medirá con base en el *porcentaje de error*, que se calcula con la siguiente ecuación:

$$porcentaje\ de\ error = \frac{número\ de\ errores\ en\ la\ clasificación}{número\ total\ de\ datos}$$

Denotando como *mejor clasificación* el menor porcentaje de error.

Es importante notar en todos los ejemplos, que la elección del conjunto de entrenamiento inicial es muy importante para el clasificador, se debe mantener la proporción de datos que la base de datos original, ya que de no ser así se verá afectado su desempeño. En cada caso se hace esta aclaración.

Ejemplo 1. Clasificación de Sexo

Se desea clasificar a una persona en hombre o mujer, tomando en cuenta 7 características: estatura, edad, tamaño de pie, largo de brazo, ancho de espalda, perímetro de cráneo y peso. Se definen 2 clases, *hombre* y *mujer*. La tabla 3.1 con el total de los datos se muestra a continuación:

Sexo	Estatura	Edad	Pie	Largo de brazo	Ancho de espalda	Perímetro de craneo	Peso
hombre	161	47	26	67	50	57	82
mujer	156	45	23	62	49	55	77
mujer	158	24	23	65	44	56	60
mujer	129	8	21	52	32	52	29
hombre	165	22	27	67	52	57	85
mujer	159	41	24	62	46	56	54.5
mujer	153	49	21.5	55	41	53.5	60
mujer	151	23	22.5	49	35	52.5	48
mujer	155	22	22	53	34	52	46
hombre	172	42	26.5	66	53	58.7	106
hombre	170	23	29	56	43	59	65
hombre	161	23	25	51.5	38	54.2	56
mujer	145	51	22.5	53	32	57	50
hombre	162	69	27	59	46	59	90
mujer	161	54	27	58	40	57	70
mujer	162	51	25	61	43	57	75
hombre	170	30	28.5	63	43	59	85
hombre	167	53	27	61	46	59	77
hombre	161	88	26	60	41	54	65
mujer	154	45	23	57	37	56	65
hombre	163	56	24	63	38	55	60
hombre	178	28	28.5	66.5	53	62	97
mujer	160	33	25	56	34	55	62
hombre	170	18	27	70	40	59.5	81.9
hombre	178.5	18	28.5	68	38	58.5	82.6
hombre	173	20	27.5	64	41	61	69.1
hombre	172	19	27	65	44	59	87.3
hombre	179	19	28	64	51	60	70
hombre	174	19	27.5	69.5	48	55.5	66.7
hombre	187	19	29	74	48	63	99.4
hombre	169	19	27.5	62	44	60	77.4
hombre	166	19	28	65	44	58.5	70.8
hombre	165	18	26.5	61	48	59.5	75.1
mujer	164	18	25	55	39	56	60.4
mujer	169	20	25	64.5	39	58.5	57.9
hombre	168	18	25	62	47	57.5	67.9

Tabla 3.1. Datos reales para clasificación de sexo.

Se ejecutó el clasificador con los 36 datos y se obtuvo lo siguiente:

Simple Distribution

```
Distribution model for label attribute Sexo

Class hombre (0.611)
7 distributions

Class mujer (0.389)
7 distributions
```

La tabla 3.2 muestra las medias y desviaciones estándar:

Atributo	parámetro	hombre	mujer
Estatura	mean	169.613636	155.428571
Estatura	standard deviation	6.80324121	9.63738143
Edad	mean	31.2272727	34.5714286
Edad	standard deviation	19.5884607	15.1490761
Pie	mean	27.0909091	23.5357143
Pie	standard deviation	1.32410224	1.68093708
Largo de brazo	mean	63.8409091	57.3214286
Largo de brazo	standard deviation	4.88531903	4.92875059
Ancho de espalda	mean	45.2727273	38.9285714
Ancho de espalda	standard deviation	4.8025967	5.28370915
Perímetro de craneo	mean	58.45	55.25
Perímetro de craneo	standard deviation	2.30914332	2.03573838
Peso	mean	78.0090909	58.2
Peso	standard deviation	13.0280616	12.5067982

Figura 3.2 Tabla de medias y desviaciones estándar de los datos de clasificación de sexo

Se probó el clasificador con 3 datos muestra mostrados en la tabla 3.3

Sexo	Estatura	Edad	Pie	Largo de brazo	Ancho de espalda	Perímetro de craneo	Peso
muestra	169	24	29	56	42.3	59	72
muestra	154	19	24	52	41	57	62
muestra	154	25	25	51	41.3	53.5	85

Tabla 3.3 Datos muestra para clasificar

Y se obtuvo la predicción, mostrada en la tabla 3.4:

hombre	mujer	predicción
0.999139411	8.61E-04	hombre
5.72E-04	0.99942821	mujer
0.005111288	0.99488871	mujer

Figura 3.4 Tabla de predicción de sexo

Para todos los casos el resultado es correcto, se trata de un hombre y dos mujeres.

Ejemplo 2. Predicción sexo. CBI- Kernel.

Se desea clasificar a una persona en hombre o mujer, tomando en cuenta 7 características: estatura, edad, tamaño de pie, largo de brazo, ancho de espalda, perímetro de cráneo y peso. Se definen 2 clases, *hombre* y *mujer*. Los datos iniciales de entrenamiento se muestran en la tabla 3.1.
Se realizaron pruebas con el Clasificador Bayesiano Ingenuo – Kernel, con ambos métodos de estimación: *greedy* y *full*.
Los datos muestra para clasificar se pueden ver en la tabla 3.3. Estos datos representan información de un hombre y dos mujeres. A continuación se muestra a detalle el resultado de cada prueba.

GREEDY (voraz)

Para este modo de estimación, se requieren dos parámetros, ancho de banda mínimo y número de kernels. Para cada atributo se realizó una combinación de estos dos parámetros y se utilizó el ancho de banda óptimo, calculado de acuerdo a la teoría, con la ecuación (10).

Como ejemplo, se calculará el ancho de banda para el atributo *estatura*. Para calcular la variable h sólo se necesita conocer la desviación estándar (σ) y el tamaño de la muestra (n). En este caso, puede verse con los datos de la tabla 3.1, que $n = 36$ y $\sigma = 10.55$, entonces:

$$h = \left(\frac{4\hat{\sigma}^5}{3n}\right)^{\frac{1}{5}} \approx 1.06\hat{\sigma}n^{-\frac{1}{5}}$$

$$h = \left(\frac{4(10.55)^5}{3(36)}\right)^{\frac{1}{5}} = \mathbf{5.46}$$

Se muestra a continuación las tablas resumen y las gráficas para cada atributo, resaltando el valor del ancho de banda óptimo para cada uno.

Se puede observar que la elección del ancho de banda es de suma importancia en el resultado final. Si el ancho de banda es muy pequeño o muy grande, con respecto al óptimo, se generan errores en la clasificación, y se observa gráficamente una diferencia significativa en la densidad de los datos.

NOTA: En todos los casos se probó el operador con los atributos dados por omisión en *RapidMiner*, *ancho de banda=0.1* y *# de kernels=10*.

ESTATURA - Ancho de banda óptimo: **5.46**

Las figuras 3.5 a), b) y c) muestran la tabla resumen con los diferentes valores para *h* y número de kernels, así como las gráficas correspondientes.

Ancho_banda (h)	#Kernels	Predicción	Error
0.1	10	h, m, h	1
1	10	h, m, h	1
1	5	h, m, h	1
2	10	h, m, h	1
2	5	h, m, m	0
5	10	h, m, m	0
5.46	5	h, m ,m	0
5.46	10	h, m, m	0
10	5	h, m, h	1
10	10	h, m,h	1

Figura 3.5 a) Tabla resumen con distintos valores de *h* y kernels

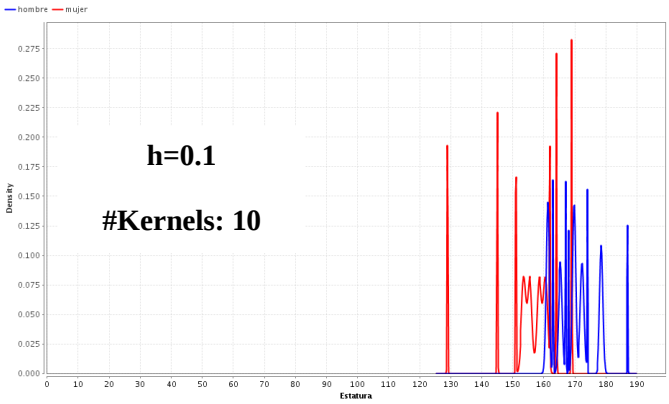


Figura 3.5 b) Gráfica de densidad para el atributo *estatura*, *h*=0.1

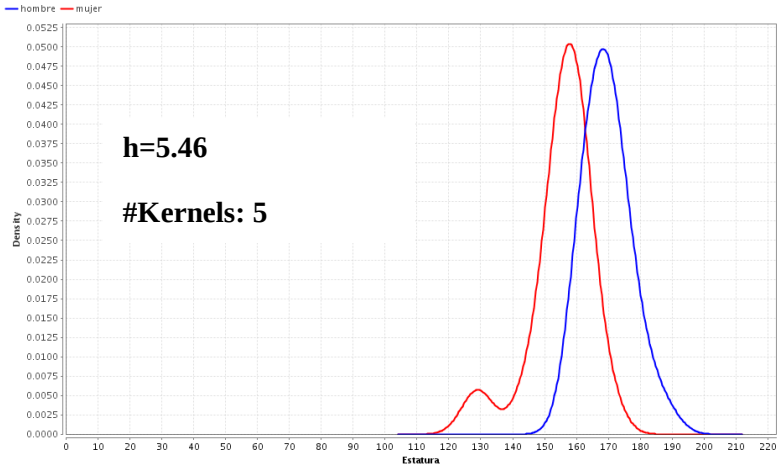


Figura 3.5 c) Gráfica de densidad para el atributo *estatura*, h=5.46

PIE - Ancho de banda óptimo: **1.18**

Las figuras 3.6 a), b) y c) muestran la tabla resumen con los diferentes valores para *h* y número de kernels, así como las gráficas correspondientes.

Ancho_banda (h)	#Kernels	Predicción	Error
0.1	10	h, m, h	1
1	10	h, m, h	1
1	5	h, m, m	0
1.18	5	h, m, m	0
1.18	2	h, m, m	0

Figura 3.6 a) Tabla resumen con distintos valores de *h* y kernels

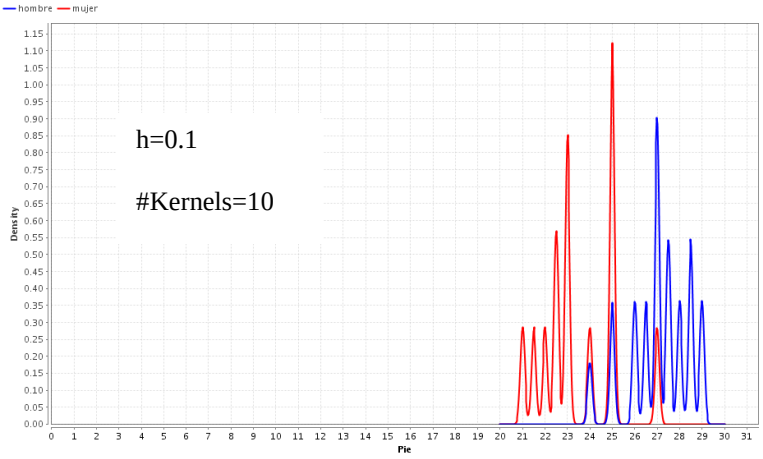


Figura 3.6 b) Gráfica de densidad para el atributo *pie*, *h*=0.1

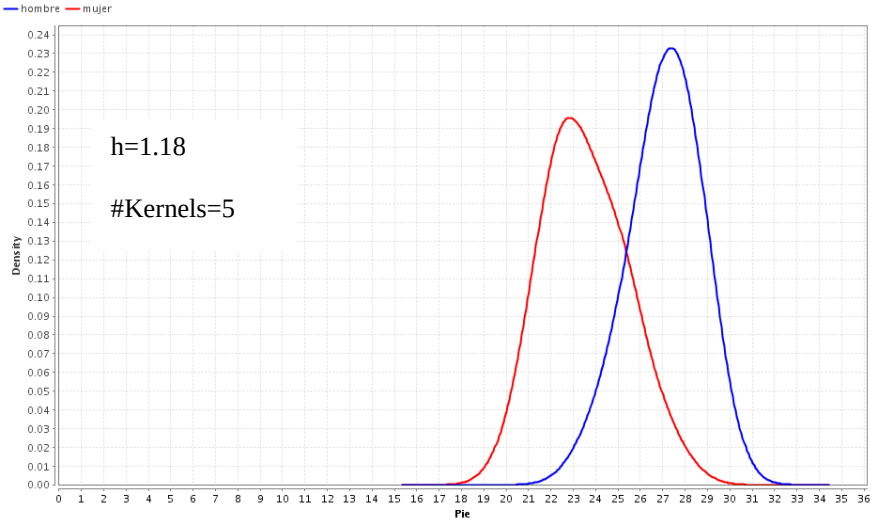


Figura 3.6 c) Gráfica de densidad para el atributo *pie*, h=1.18

LARGO DE BRAZO - Ancho de banda óptimo = **3**

Las figuras 3.7 a), b) y c) muestran la tabla resumen con los diferentes valores para *h* y número de kernels, así como las gráficas correspondientes.

Ancho_banda (h)	#Kernels	Predicción	Error
0.1	10	h, m, h	1
1	10	h, m, h	1
3	10	h, m, m	0
3	5	h, m, m	0
3	3	h, m, m	0

Figura 3.7 a) Tabla resumen con distintos valores de *h* y kernels

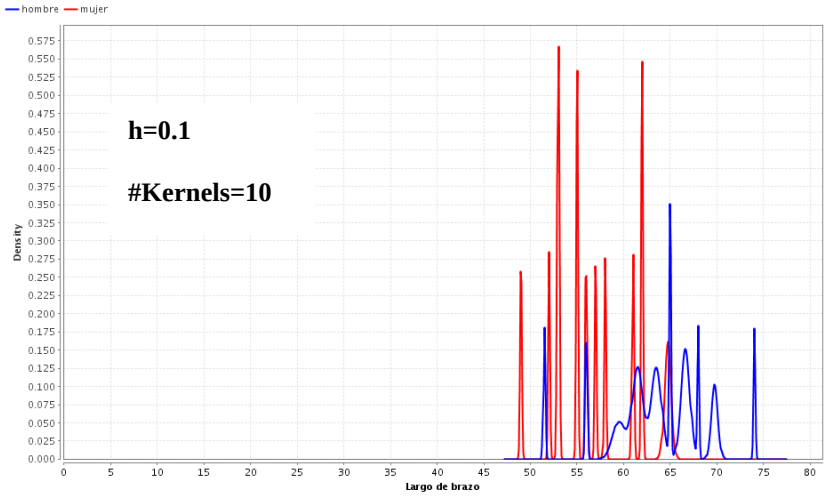


Figura 3.7 b) Gráfica de densidad para el atributo *largo de brazo*, *h*=0.1

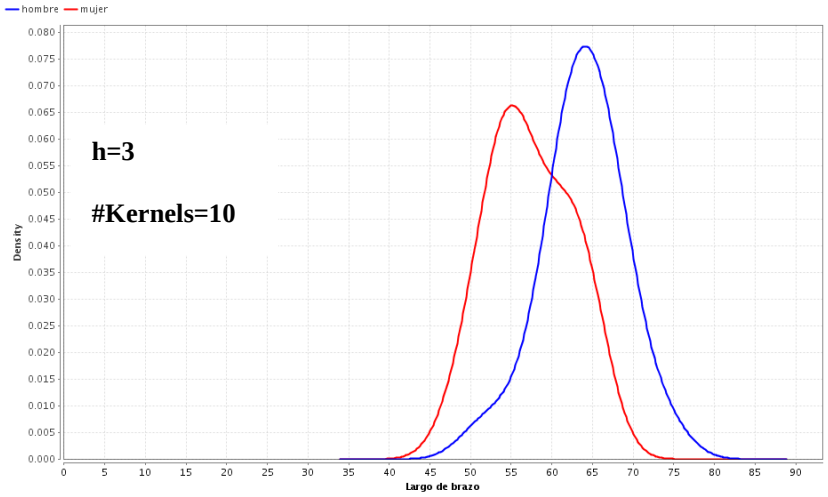


Figura 3.7 c) Gráfica de densidad para el atributo *largo de brazo*, *h*=3

PERÍMETRO DE CRÁNEO: Ancho de banda óptimo = **1.39**

Las figuras 3.8 a), b) y c) muestran la tabla resumen con los diferentes valores para *h* y número de kernels, así como las gráficas correspondientes.

Ancho_banda (h)	#Kernels	Predicción	Error
0.1	10	h, m, h	1
1	10	h, m, h	1
1.39	10	h, m, h	1
1.39	5	h, m, m	0

Figura 3.8 a) Tabla resumen con distintos valores de *h* y kernels

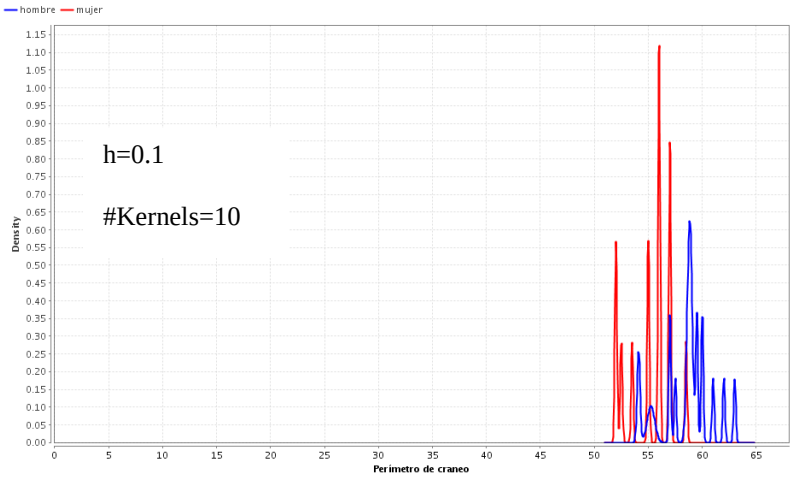


Figura 3.8 b) Gráfica de densidad para el atributo *perímetro de cráneo*, *h*=0.1

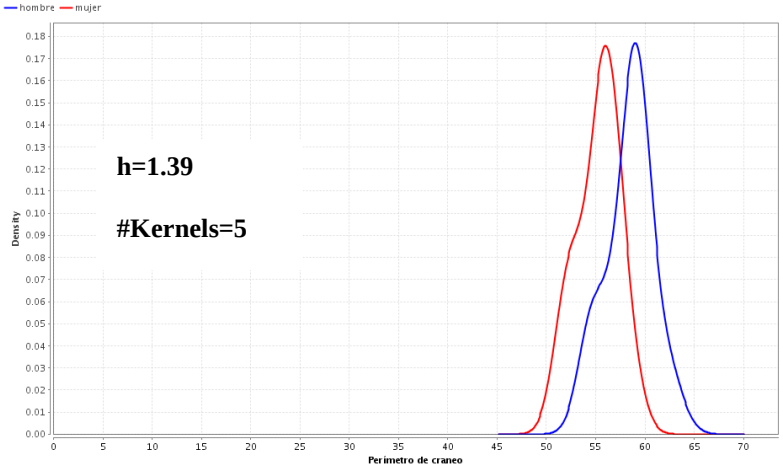


Figura 3.8 c) Gráfica de densidad para el atributo *perímetro de cráneo*, *h*=1.39

FULL (completo)

Para este caso, se necesita sólo un atributo, que es el ancho de banda. Se puede seleccionar un ancho de banda *heurística* o elegir la opción *fija*.
Se realizaron pruebas con ambas opciones, se muestran a continuación los resultados.

a) HEURÍSTICA:

Con la selección de ancho de banda heurística, se logró una clasificación sin errores. La predicción fue correcta, hombre, mujer, mujer. Se muestran a continuación las gráficas correspondientes a los atributos más representativos, en las figuras 3.9 a), b), c) y d).

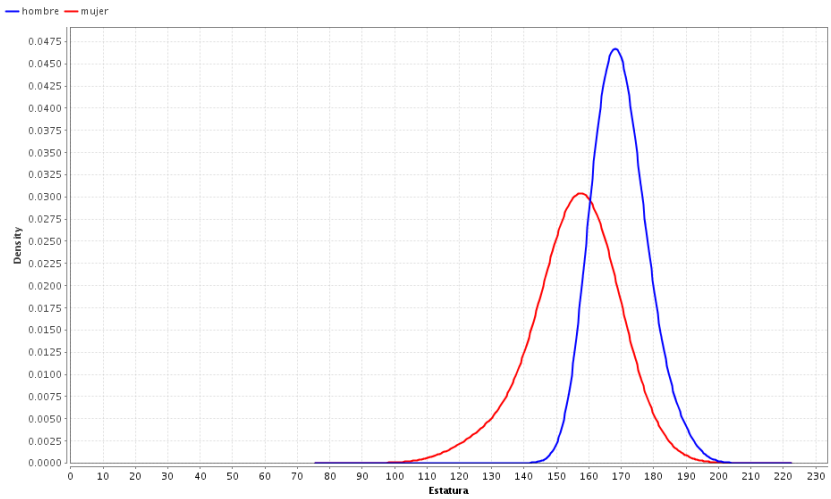


Figura 3.9 a) Gráfica de densidad para el atributo *estatura*

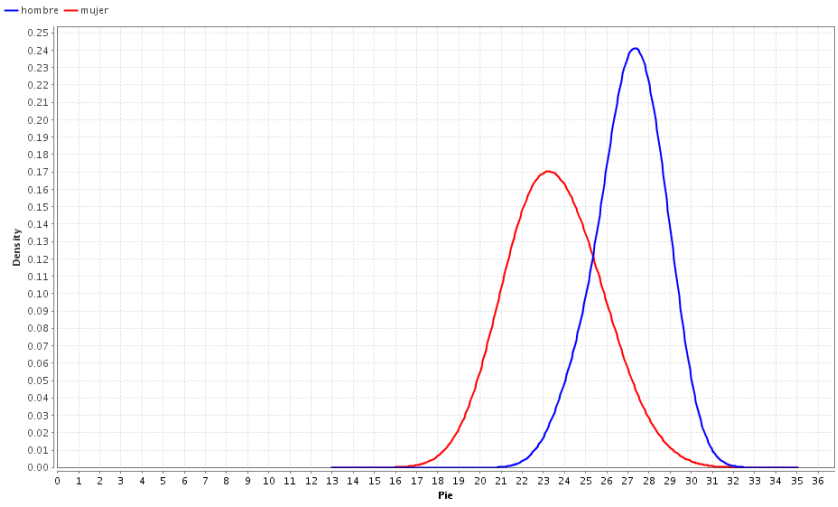


Figura 3.9 b) Gráfica de densidad para el atributo *pie*

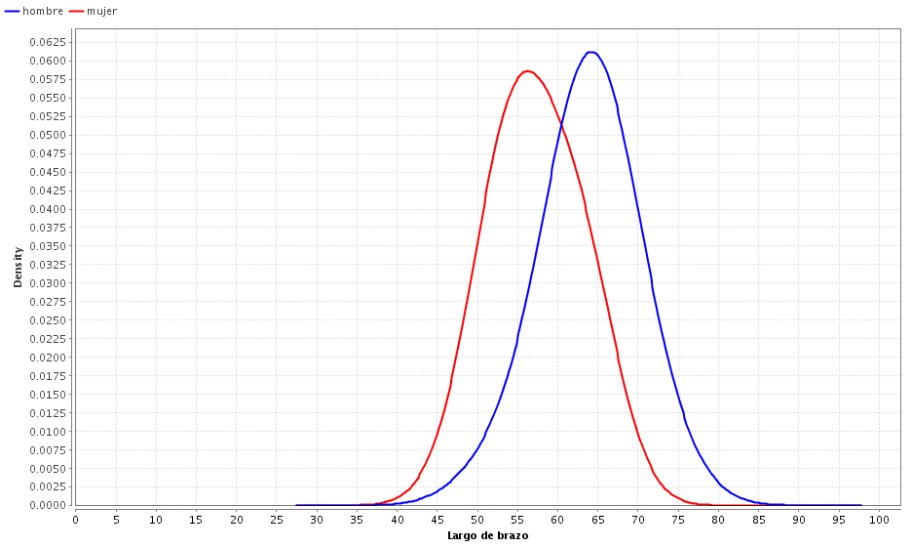


Figura 3.9 c) Gráfica de densidad para el atributo *largo de brazo*

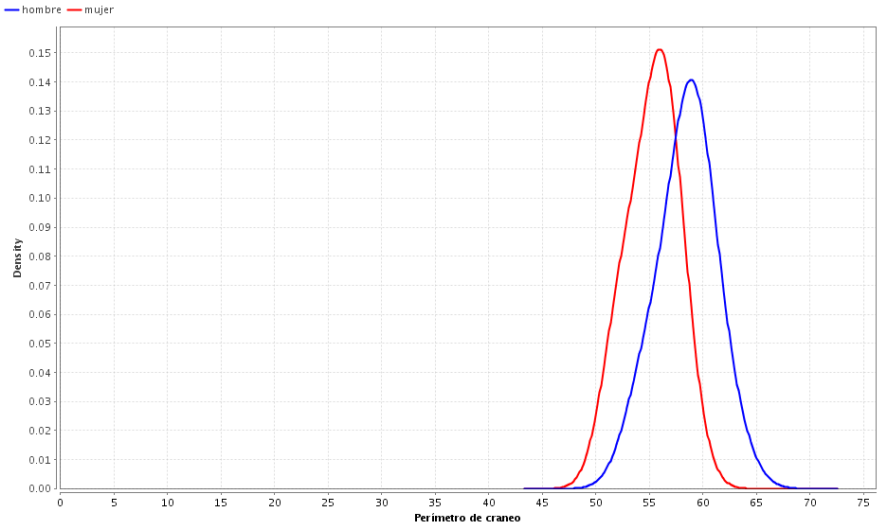


Figura 3.9 d) Gráfica de densidad para el atributo *perímetro de cráneo*

b) ANCHO DE BANDA FIJO

Para este caso, se probó el operador con los anchos de banda óptimos calculados de acuerdo a la teoría y se usó un ancho de banda muy pequeño, 0.1 y uno muy grande, 10. La tabla resumen se muestra a continuación en la figura 3.10:

Ancho_banda (h)	Predicción	Error
0.1	h, m, h	1
1.18	h, m, m	0
1.39	h, m, m	0
3	h, m, m	0
5.46	h, m, m	0
10	h, m, h	1

Figura 3.10 Tabla resumen con distintos valores para *h* y número de kernels.

Se observa que con los anchos de banda óptimos, se logra una buena clasificación, mientras que con el pequeño y grande, se obtuvieron errores. Conocer el valor óptimo del ancho de banda es indispensable para obtener una clasificación exitosa.

El mejor resultado obtenido para este operador, se da cuando se usa la opción de estimación *completa* y la elección del ancho de banda *heurística*. El usuario final no debe preocuparse por calcular un ancho de banda óptimo ni conocer el número de kernels apropiado para lograr una buena clasificación.

Ejemplo 3. PIMA Diabetes.

Este ejemplo se realizó con la base de datos PIMA-INDIANS-DIABETES [5]. Consta de 768 entradas que representan pacientes de sexo femenino, de al menos 21 años de edad, a quienes se les realizaron pruebas para determinar si mostraban síntomas de diabetes mellitus, de acuerdo con el criterio de la OMS. Se definen 2 clases: 1, "positivo para diabetes" y 0, "negativo para diabetes". Los 8 atributos que se tomaron como referencia son:

1. Número de embarazos
2. Concentración de glucosa en sangre
3. Presión diastólica (mm Hg)
4. Espesor del pliegue cutáneo del tríceps
5. Insulina en suero (2 horas) (mu U/ml)
6. Índice de masa corporal
7. Función Pedigree Diabetes
8. Edad

Se probó el clasificador Bayesiano Ingenuo, y se obtuvo lo siguiente:

Simple Distribution

```
Distribution model for label attribute clase

Class TRUE (0.349)
8 distributions
Class FALSE (0.651)
8 distributions
```

La tabla de distribuciones se muestra a continuación, en la figura 3.11:

ATRIBUTO	PARÁMETRO	TRUE	FALSE
No. Emb	mean	4.86567164	3.298
No. Emb	standard deviation	3.74123904	3.01718458
Plasma	mean	141.257463	109.98
Plasma	standard deviation	31.9396221	26.1411998
Pre_Dias	mean	70.8246269	68.184
Pre_Dias	standard deviation	21.4918117	18.0630754
espesor_tric	mean	22.1641791	19.664
espesor_tric	standard deviation	17.6797114	14.8899471
2h-serum	mean	100.335821	68.792
2h-serum	standard deviation	138.689125	98.8652893
IMC	mean	35.1425373	30.3042
IMC	standard deviation	7.26296724	7.68985501
Diab_ped_func	mean	0.5505	0.429734
Diab_ped_func	standard deviation	0.37235448	0.2990853
edad	mean	37.0671642	31.19
edad	standard deviation	10.9682537	11.6676548

Figura 3.11 Tabla de distribución para los atributos de PIMA-Diabetes

Posteriormente se ejecutó el clasificador con algunos datos de entrenamiento de diferentes tamaños, y se probó su eficiencia en la predicción.
Es importante mencionar que la elección del conjunto de entrenamiento es muy importante para lograr una clasificación exitosa. Se debe considerar mantener la misma proporción de clases que en la base de datos original, pues de no ser así, el desempeño del clasificador se ve afectado, aumentando el número de errores en la predicción.
Para este caso, la proporción de clases en la base de datos original es de 35% "positivo" y 65% "negativo".
La tabla 3.12 muestra el resumen de las pruebas realizadas.

PRUEBA	TAMAÑO MUESTRA	PROPORCIÓN		DISTRIBUCIÓN		NÚMERO ERRORES	PORCENTAJE ERROR
		Positivo	Negativo	Positivo	Negativo		
1	50	19	31	0.38	0.62	206	26.82%
2	100	44	56	0.44	0.56	196	25.52%
3	200	68	132	0.34	0.66	179	23.31%
4	300	115	185	0.383	0.617	190	24.74%
5	300	105	195	0.35	0.65	181	23.57%
6	400	146	254	0.365	0.635	202	26.30%
7	400	140	260	0.35	0.65	200	26.04%
8	**50**	**25**	**25**	**0.5**	**0.5**	**226**	**29.43%**
9	**100**	**65**	**35**	**0.65**	**0.35**	**235**	**30.60%**

Tabla 3.12 Resultados de las pruebas para PIMA

Se puede notar que, en general, el desempeño del clasificador es bueno. Las pruebas **3** y **5** dan los mejores resultados, cuando la muestra es de tamaño **200** y **300**, respectivamente; además la distribución de clases, se mantiene como la original. Aquí el número de errores es sólo de 179, teniendo una eficiencia de casi 77%.

Se observa en las pruebas **8** y **9**, que el porcentaje de error aumenta hasta el 30%, aquí no importa el tamaño de la muestra de entrenamiento, sino que no se mantuvo la proporción de clases que en la base original.

Ejemplo 4. Clasificación de Vehículos

Para este ejemplo se toma como referencia la base de datos académica conocida como Vehicle Silhouettes [6], que guarda información sobre 846 siluetas de vehículos descritas a través de 18 atributos.
El objetivo es clasificar una silueta dada como uno de cuatro tipos de vehículos, utilizando un conjunto de características extraídas de la silueta. El vehículo puede ser visto desde diferentes ángulos. Las 4 clases disponibles son *opel*, *saab*, *bus* y *van*.

Se ejecutó el CBI con esta base de datos y se obtuvo la distribución para el atributo CLASE:

Simple Distribution

```
Distribution model for label attribute CLASE

Class van (0.236)
18 distributions

Class saab (0.256)
18 distributions

Class bus (0.257)
18 distributions

Class opel (0.250)
18 distributions
```

La tabla de distribución para cada atributo se muestra a continuación en la figura 3.13.

Atributo	Parámetro	van	saab	bus	oppel
COMPACT	mean	90.535	97.281106	91.5917431	95.0613208
COMPACT	standard deviation	3.88254372	9.08138423	8.6180245	8.23073734
CIRCUL	mean	42.04	45.5345622	45.0688073	46.5801887
CIRCUL	standard deviation	4.09539513	6.81908772	5.03076612	7.23526016
DIST_ CIRCUL	mean	73.295	88.6728111	76.7201835	89.0896226
DIST_ CIRCUL	standard deviation	10.8538938	17.0051421	12.0925637	15.5866782
RADI_RATIO	mean	147	180.801843	166.004587	180.301887
RADI_RATIO	standard deviation	29.8680515	30.8048276	30.5741522	31.3549228
PR_AX_ASPECT_RAT	mean	61.23	61.1428571	63.4311927	60.8773585
PR_AX_ASPECT_RAT	standard deviation	11.3605639	4.32187129	8.80287654	4.9561161
MAX_LENG_ASPECT_RATIO	mean	9.69	8.79262673	7.01376147	8.85849057
MAX_LENG_ASPECT_RATIO	standard deviation	7.21416112	2.15130043	4.75722042	1.98065851
SCATTER RATIO	mean	141.43	179.668203	170.022936	182.165094
SCATTER RATIO	standard deviation	14.0431889	31.5238087	33.3568118	32.8218819
ELONGATEDNESS	mean	47.98	38.3179724	40.1146789	37.8773585
ELONGATEDNESS	standard deviation	4.69144415	7.48673515	6.50137633	7.72415775
PR.AX_RECTANG	mean	18.57	21.4470046	20.5733945	21.5896226
PR.AX_RECTANG	standard deviation	1.03462663	2.45292772	2.72835063	2.55303664
MAX_LENG_RECTAN	mean	145.09	148.691244	146.701835	151.273585
MAX_LENG_RECTAN	standard deviation	11.0545427	16.1346064	10.4923978	18.1580909
SCALED VAR_ALONG_MAJ_AX	mean	163.89	197.152074	192.889908	198.617925
SCALED VAR_ALONG_MAJ_AX	standard deviation	19.7037838	27.9200231	33.9965679	28.8409638
SCALED VAR_ALONG_MIN_AX	mean	297.77	493.797235	448.894495	508.537736
SCALED VAR_ALONG_MIN_AX	standard deviation	56.0862321	163.15624	193.138314	172.45925
SCALED RADIUOS_GYR	mean	157.145	179.456221	180.949541	179.773585
SCALED RADIUOS_GYR	standard deviation	22.8394874	33.7760759	31.2473737	34.7214462
SKEWNESS_MAJ_AX	mean	72.78	69.7557604	77.1238532	70.1415094
SKEWNESS_MAJ_AX	standard deviation	8.84492871	5.29152681	7.68494128	5.08028189
SKEWNESS_MIN_ AX	mean	6.39	7.65898618	4.8440367	6.60377358
SKEWNESS_MIN_ AX	standard deviation	4.66978957	5.81694721	3.22053982	5.19328631
KURTOSIS_MIN_AX	mean	9.74	15.2995392	10.2110092	15.0141509
KURTOSIS_MIN_AX	standard deviation	6.25524102	10.0667977	6.87287247	10.1666114
KURTOSIS_MAJ_AX	mean	188.925	189.714286	187.811927	189.278302
KURTOSIS_MAJ_AX	standard deviation	6.36726672	4.99179221	7.32123009	5.59306451
HOLLOWS RATIO	mean	196.115	198.041475	191.325688	197.113208
HOLLOWS RATIO	standard deviation	7.33969554	6.61039681	7.91832622	5.84608388

Figura 3.13 Tabla de distribución para los atributos de Vehicle Silhouette

Las gráficas de densidad para los atributos más representativos se muestran en las figuras 3.14 a), b) y c).

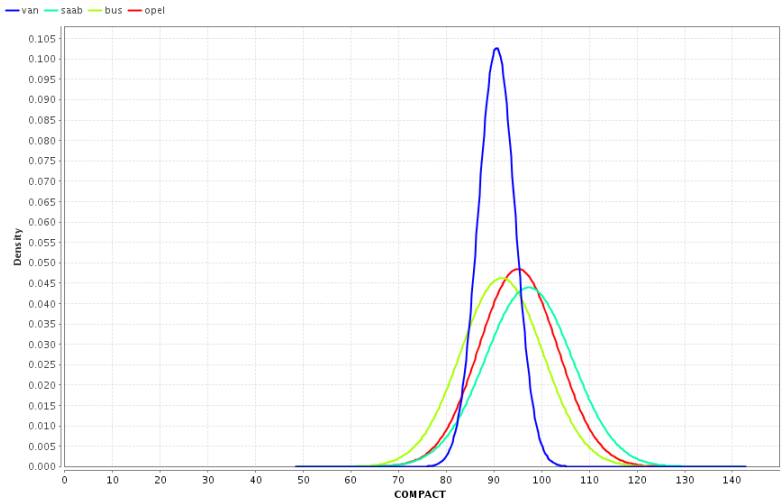


Figura 3.14 a) Gráfica de densidad para el atributo *compact*

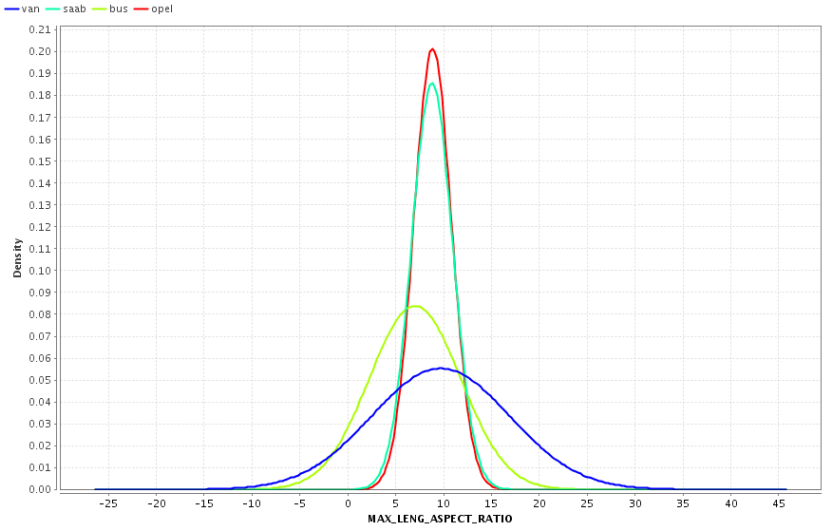


Figura 3.14 b) Gráfica de densidad para el atributo *max_leng_aspect_RATIO*

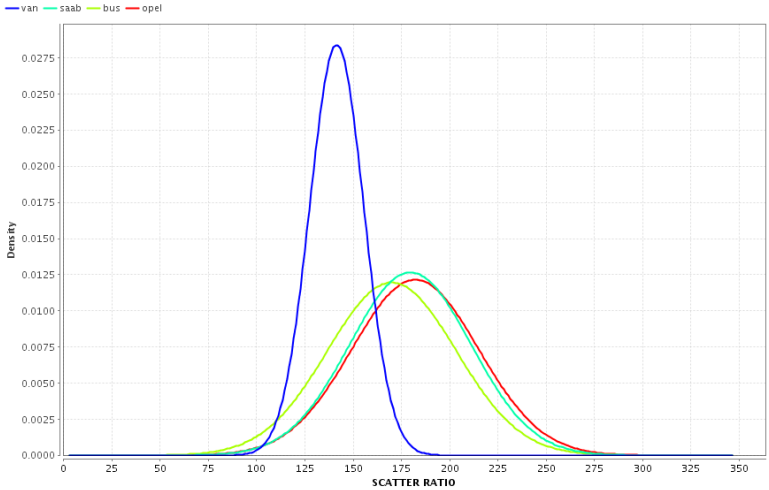


Figura 3.14 c) Gráfica de densidad para el atributo *SCATTER RATIO*

También se ejecutó el CBI y CBI-Kernel con algunos datos de entrenamiento de diferentes tamaños, y se probó su eficiencia en la predicción. En el caso de CBI-Kernel se utilizó la opción completa y el ancho de banda heurística.

Es importante recordar que la elección del conjunto de entrenamiento es muy importante para lograr una buena clasificación. Se debe considerar mantener la misma proporción de clases que en la base de datos original.
Para este caso, la proporción de clases en la base de datos original es de 23.6% para *van*, 25.6% para *saab*, 25.7% para *bus* y 25% para *opel*. Se tomaron las muestras con una proporción del 25% cada una.

La figura 3.15 muestra el resumen de las pruebas realizadas.

TAMAÑO MUESTRA	NÚMERO DE ERRORES		PORCENTAJE DE ERROR	
	CBI	CBI-Kernel	CBI	CBI-Kernel
100	469	460	55.44%	54.37%
200	465	397	54.96%	46.93%
300	484	390	57.21%	46.10%
400	462	367	54.61%	43.38%

Figura 3.15 Resultados obtenidos en la clasificación de siluetas de vehículos

Se puede observar que el resultado para este caso varía mucho de acuerdo al clasificador que se elija. CBI-Kernel funcionó mejor al reducir el número de errores mientras más grande sea la muestra inicial.

Ejemplo 5: Base de datos Zoo.

Este ejemplo es una base de datos que contiene animales, descritos a través de características que se pueden evaluar como falsas o verdaderas. Incluye variables numéricas y nominales. [4]
Para este caso, existen 7 clases, que corresponden a los 7 tipos de animales: mamíferos, aves, peces, anfibios, invertebrados, insectos y reptiles. Se ejecutó el clasificador ingenuo con los 100 datos y se obtuvo:

Simple Distribution

```
Distribution model for label attribute tipo

Class mamífero (0.410)
15 distributions

Class pez (0.130)
15 distributions

Class ave (0.200)
15 distributions

Class invertebrado (0.100)
15 distributions

Class insecto (0.080)
15 distributions

Class anfibio (0.030)
15 distributions

Class reptil (0.050)
15 distributions
```

Se tomó una muestra al azar del total de datos disponibles, considerando la proporción de cada clase en la base de datos original, para realizar el entrenamiento del clasificador y posteriormente se probó con el total de los datos.

La tabla 3.16 muestra los resultados obtenidos:

Muestra	**Distribución**							**Error**
	Mamífero	Ave	Pez	Invertebrado	Insecto	Reptil	Anfibio	
20	0.381	0.19	0.143	0.095	0.095	0.048	0.048	65
50	0.4	0.2	0.12	0.1	0.08	0.06	0.04	60

Tabla 3.16 Resultados del entrenamiento del CBI para los datos Zoo.

Se pudo notar que la elección de la muestra para el entrenamiento afecta en gran medida el resultado del clasificador. Se debe tener en cuenta que la muestra mantenga la misma proporción de elementos de cada clase que la base de datos original, para que el desempeño del clasificador sea lo más eficiente posible.

En este caso, al usar una muestra de 20 y 50 datos iniciales para el entrenamiento, produjo casi el mismo número de errores al clasificar los datos, 60 y 65, respectivamente.

Se observó que, para este ejemplo cuyos datos son nominales y el número de clases es 7, la eficiencia del CBI no es buena, ya que puede verse en [13], que al aplicar K-Medias, otra técnica de minería de datos, se obtiene un mejor resultado.

CONCLUSIONES

Al finalizar este trabajo se puede concluir que:

La clasificación está presente en muchos ámbitos de la vida diaria. Es importante contar con herramientas que faciliten este proceso cuando se cuenta con una gran cantidad de información que no es fácil de analizar.

El Clasificador Bayesiano Ingenuo y su variante de Kernel, son dos algoritmos que funcionan adecuadamente para obtener una clasificación exitosa. La suposición de independencia de sus atributos lo hace uno de los más sencillos clasificadores disponibles.

Estos clasificadores están presentes en algunas aplicaciones, una de ellas es el software *RapidMiner*, que ofrece estos operadores junto con una gran cantidad de herramientas para el análisis y minería de datos. Además de tener una interfaz muy sencilla, el resultado obtenido es fácil de interpretar, incluso para usuarios que no son expertos en el área.

El desempeño del Clasificador Bayesiano Ingenuo es mucho mejor cuando se usan sólo 2 clases y los datos son numéricos, como se pudo ver en los ejemplos 1, 2 y 3, aunque su desempeño es en general bastante aceptable.

El CBI no es adecuado para elaborar clasificaciones en donde los datos no sigan una distribución normal y/o los datos sean nominales, como se pudo ver en los ejemplos 4 y 5.

En el caso del Clasificador Bayesiano Ingenuo Kernel, funciona sin problema cuando se conoce el valor óptimo de la variable *h*, ancho de banda de los datos, y el número de kernels, que casi siempre está asociado con *h*.

Una *ventaja* del Clasificador Bayesiano Ingenuo Kernel es que puede usarse en el modo de estimación completo y usar un ancho de banda heurístico, esto ofrece al usuario final una experiencia más agradable, ya que no se necesita calcular un ancho de banda ni saber cuántos kernels son apropiados para sus datos; además reduce significativamente el número de errores cuando los datos a clasificar no tienen una distribución normal, como pudo verse en la prueba 4.4.

En el caso del Clasificador Bayesiano Ingenuo Kernel, funciona sin problema cuando se conoce el valor óptimo de la variable *h*, ancho de banda de los datos, y el número de kernels, que casi siempre está asociado con *h*.

Una ventaja del Clasificador Bayesiano Ingenuo Kernel es que puede usarse en el modo de estimación completo y usar un ancho de banda heurístico, esto ofrece al usuario final una experiencia más agradable, ya que no se necesita calcular un ancho de banda ni saber cuántos kernels son apropiados para sus datos.

Al presentar una parte de este trabajo en el *VII Congreso Nacional de Tecnología Aplicada a las Ciencias de la Salud 2016,* se obtuvo el primer lugar en la modalidad de cartel en la categoría de Posgrado.

Esta participación permitió conocer otras formas de aplicar el CBI. Una propuesta para trabajo futuro es el análisis de otras bases de datos reales en el área médica, para comparar la eficiencia del CBI y CBI-Kernel con los resultados obtenidos con otros clasificadores y otras aplicaciones que se están usando actualmente.

BIBLIOGRAFÍA

[**1**] Sucar, Luis Enrique. "Clasificadores Bayesianos: de Datos a Conceptos". Instituto Nacional de Astrofísica, Óptica y Electrónica, Tonantzintla, Puebla, México. http://www.ai.org.mx/ai/images/sitio/2014/05/ingresos/less/trabaj_final_dr._sucar.pdf. Consultado el 1 de febrero de 2016.

[**2**] Rich, Irina. "An Empirical Study of the naive Bayes Classifier". IBM Research Division. http://www.research.ibm.com/people/r/rish/papers/RC22230.pdf. Consultado el 25 de enero de 2016.

[**3**] Mitchell, Tom, Machine Learning, Ed. McGraw-Hill (1997).

[**4**] UCI Machine Learning Repository. https://archive.ics.uci.edu/ml/datasets/Zoo. Consultado el 16 de febrero de 2016

[**5**] UCI Machine Learning Repository. http://archive.ics.uci.edu/ml/datasets/Pima+Indians+Diabetes Consultado el 17 de febrero de 2016.

[**6**] UCI Machine Learning Repository. http://archive.ics.uci.edu/ml/machine-learning-databases/statlog/vehicle/. Consultado el 13 de febrero de 2016.

[**7**] *http://rapidminer.com/documentation/*

[**8**] Cuevas, Antonio. "El análisis estadístico de grandes masas de datos: algunas tendencias recientes". Departamento de Matemáticas. Universidad Autónoma de Madrid. http://www.mat.ucm.es/~rrdelrio/documentos/acuevas.pdf. Consultado el 29 de marzo de 2016

[**9**] Rosenblatt, M. (1956). "Remarks on Some Nonparametric Estimates of a Density Function". The Annals of Mathematical Statistics 27 (3): 832. *doi*:*10.1214/aoms/1177728190* Consultado el 29 de marzo de 2016.

[**10**] Wand, M.P; Jones, M.C. (1995). "Kernel Smoothing". London: Chapman & Hall/CRC. ISBN 0-412-55270-1. Consultado el 2 de abril de 2016.

[**11**] Epanechnikov, V. A. (1969). "Non-Parametric Estimation of a Multivariate Probability Density". *Theory Probab. Appl.* 14 (1): 153–158.
doi:10.1137/1114019.
Consultado el 4 de abril de 2016.

[**12**] Silverman, B. W. (1986). "Density Estimation for Statistics and Data Analysis". Chapman and Hall, London.

[**13**] Hernández Baez Imelda. Tesis de licenciatura. "K-Medias en RapidMiner". FCC-BUAP, 2015

[**14**] Schölkopf, Bernard; Smola, Alexander J. "Learning with Kernels, Support Vector Machines, Regularization, Optimization, and Beyond". The MIT Press. Cambridge, Massachusetts, London, England.

[15] Shawe Taylor, J; Cristianini, N. "Kernel Methods for Pattern Analysis". Primera edición. Cambridge University Press.