DATA MINING CON Rweka

Grupo de Usuarios de R de Madrid

Mauricio Beltrán Pascual

Madrid, 28 de febrero de 2017

PERCEPTRÓN MULTICAPA

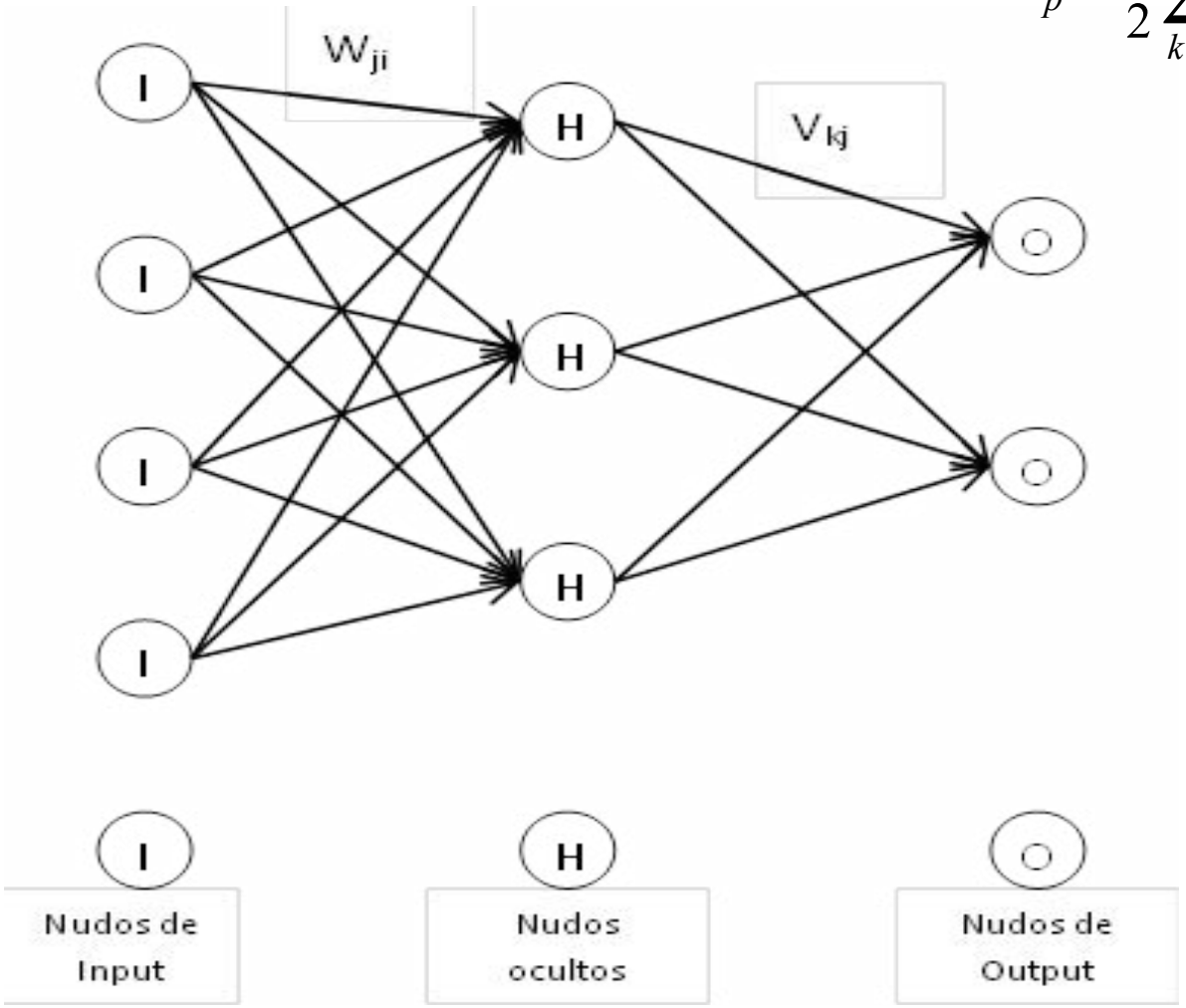
WOW("weka/classifiers/functions/MultilayerPerceptron")

WPM("install-package","c:/WEKA_ZIP/RBFNetwork1.0.6.zip")
WOW("weka/classifiers/functions/RBFNetwork")

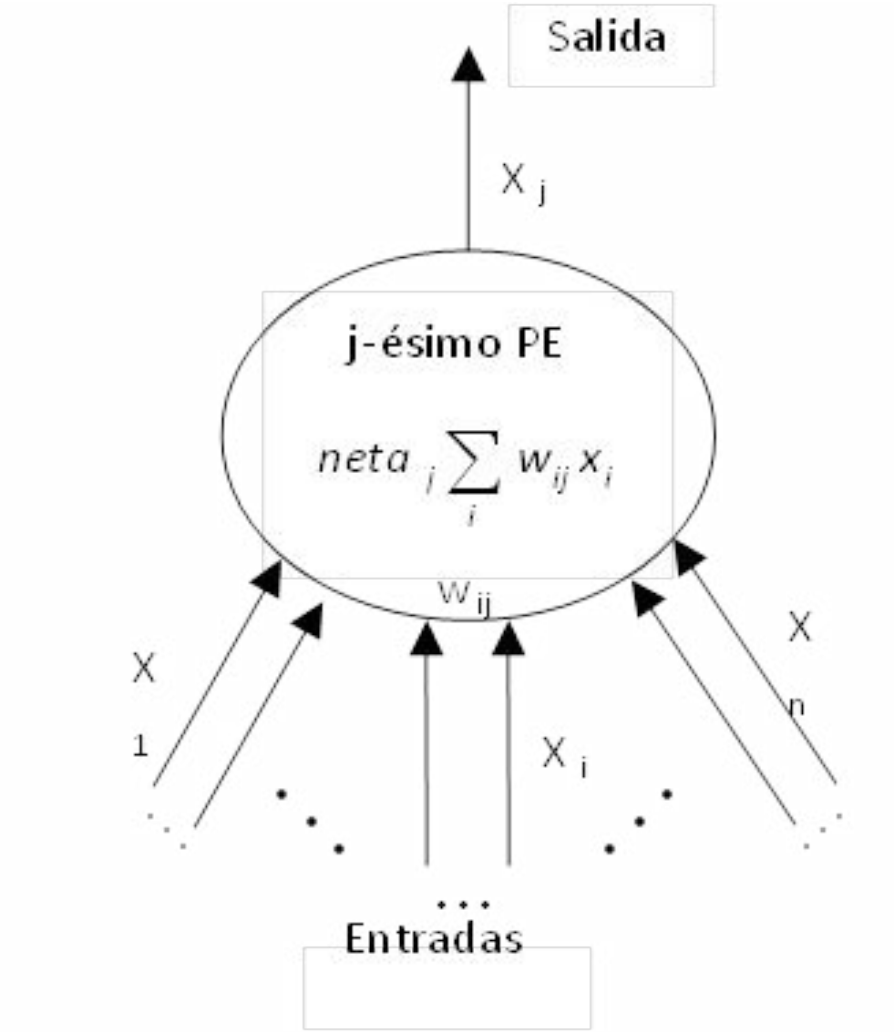
WPM("install-package","c:/WEKA_ZIP/multiLayerPerceptrons.zip")
WOW("weka/classifiers/functions/MLPClassifier")

Estructura de una red neuronal

$$E_p = \frac{1}{2}\sum_{k=1}^{M}\left(d_{pk} - y_{pk}\right)^2$$

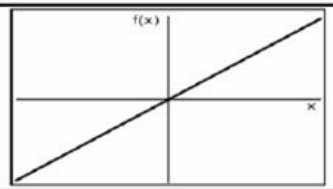
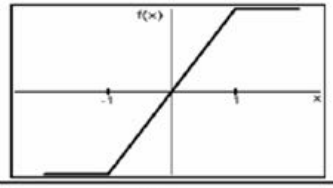
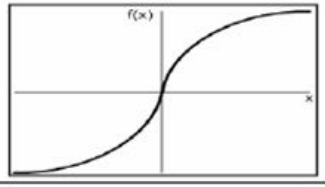
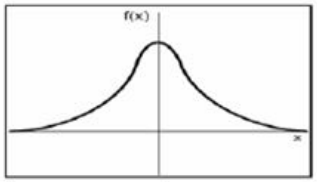
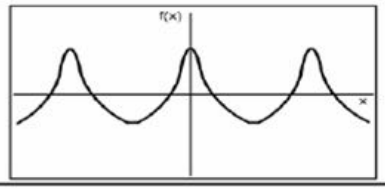


Unidad básica de procesamiento de una red neuronal



$$y_{pk} = f\left(net_{pk}\right)$$

Funciones activación más utilizadas en redes neuronales

NOMBRE	FUNCIÓN	RANGO	GRÁFICA
Identidad	$y = x$	$[-\infty, +\infty]$	
Escalón	$y = signo(x)$ $y = H(x)$	$[-1, +1]$ $[0, +1]$	
Lineal a tramos	$y = \begin{cases} -1, \text{ si } x < -1 \\ x, \text{ si } +1 \leq x \leq -1 \\ +1, \text{ si } x > +1 \end{cases}$	$[-1, +1]$	
Sigmoidea o Logística	$y = \frac{1}{1+e^{-x}}$ $y = tgh(x)$	$[0, +1]$ $[-1, +1]$	
Gaussiana	$y = a \cdot e^{-bx^2}$	$[0, +1]$	
Sinusoidal	$y = a \cdot sen(\omega x + \varphi)$	$[-1, +1]$	

Fuente: Martín del Brío y Sanz (2001)

What are the weights?

- They're learned from the training set
- Iteratively minimize the error using steepest descent
- Gradient is determined using the "backpropagation" algorithm
- Change in weight computed by multiplying the gradient by the "learning rate" and adding the previous change in weight multiplied by the "momentum":

$$W_{next} = W + \Delta W$$
$$\Delta W = -\text{learning\_rate} \times \text{gradient} + \text{momentum} \times \Delta W_{previous}$$

Can get excellent results

- Often involves (much) experimentation
  - *number and size of hidden layers*
  - *value of learning rate and momentum*

What are the weights?

- ❖ They're learned from the training set
- ❖ Iteratively minimize the error using steepest descent
- ❖ Gradient is determined using the "backpropagation" algorithm
- ❖ Change in weight computed by multiplying the gradient by the "learning rate"

and adding the previous change in weight multiplied by the "momentum":

$$W_{next} = W + \Delta W$$

$$\Delta W = -\text{learning\_rate} \times \text{gradient} + \text{momentum} \times \Delta W_{previous}$$

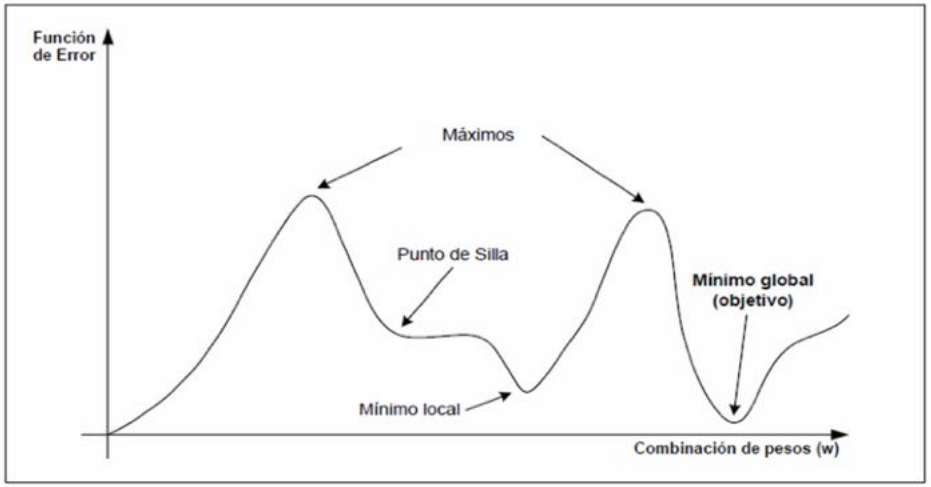
Can get excellent results

- ❖ Often involves (much) experimentation

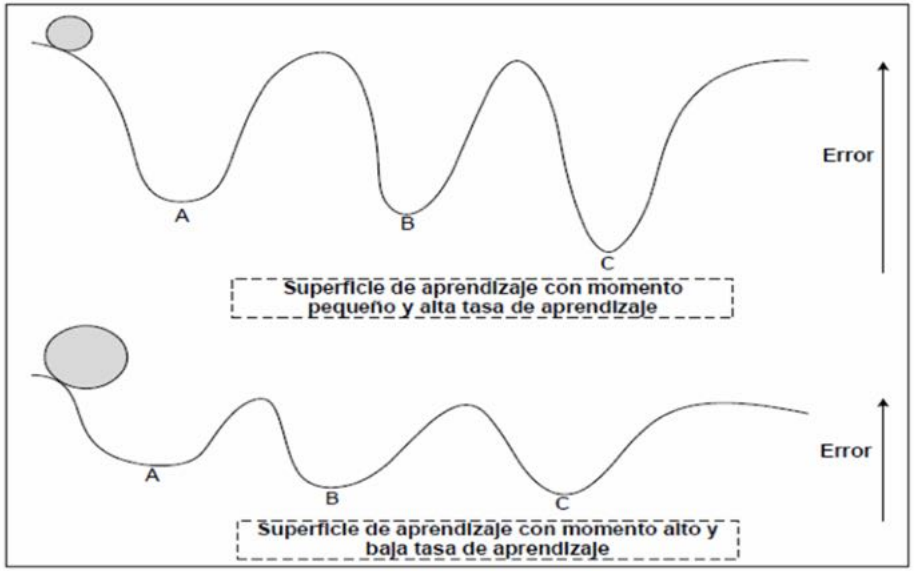
– number and size of hidden layers

– value of learning rate and momentum

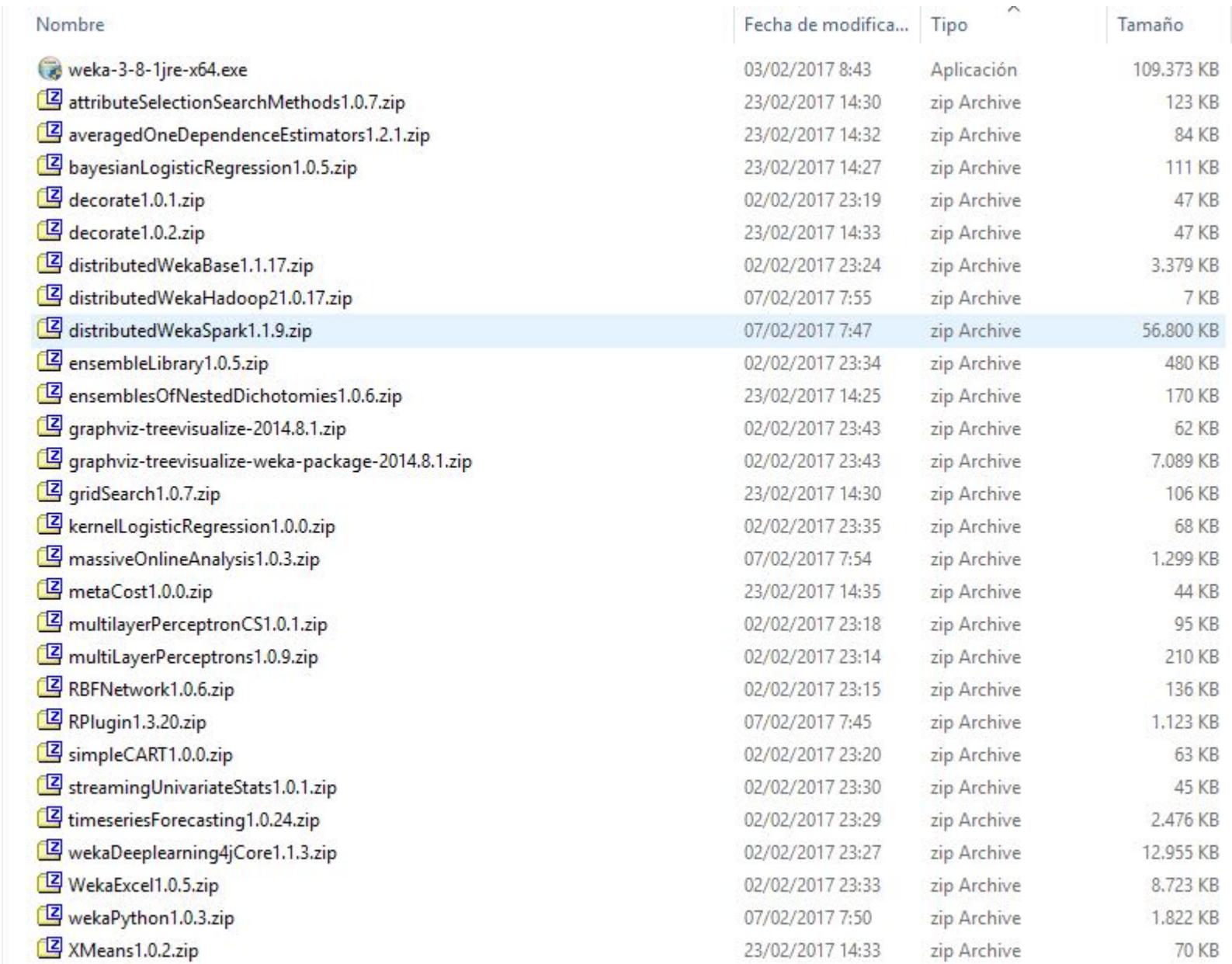
Complejidad en la búsqueda del mínimo global



Dinámica en la búsqueda del mínimo global

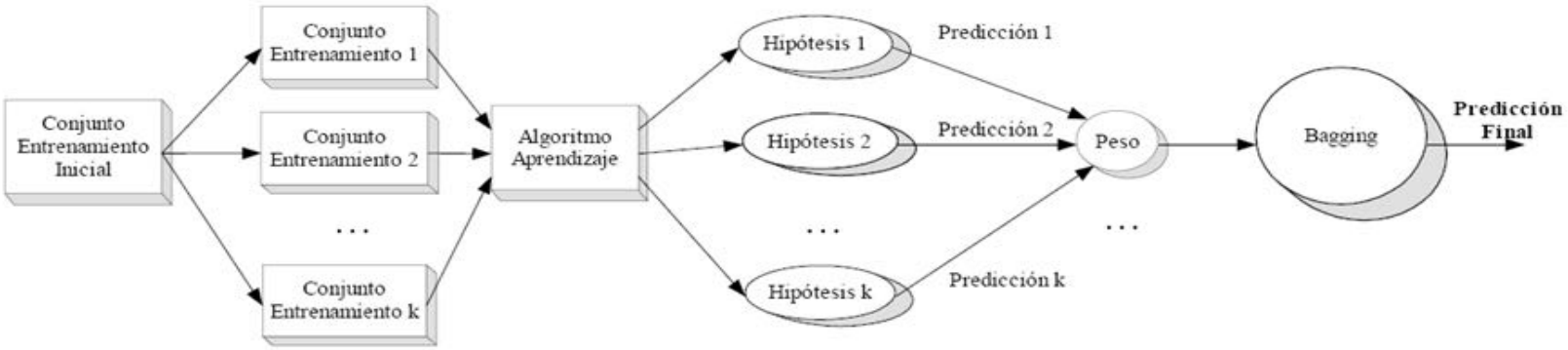


Fuente: Urquijo y Mendivil (2011)

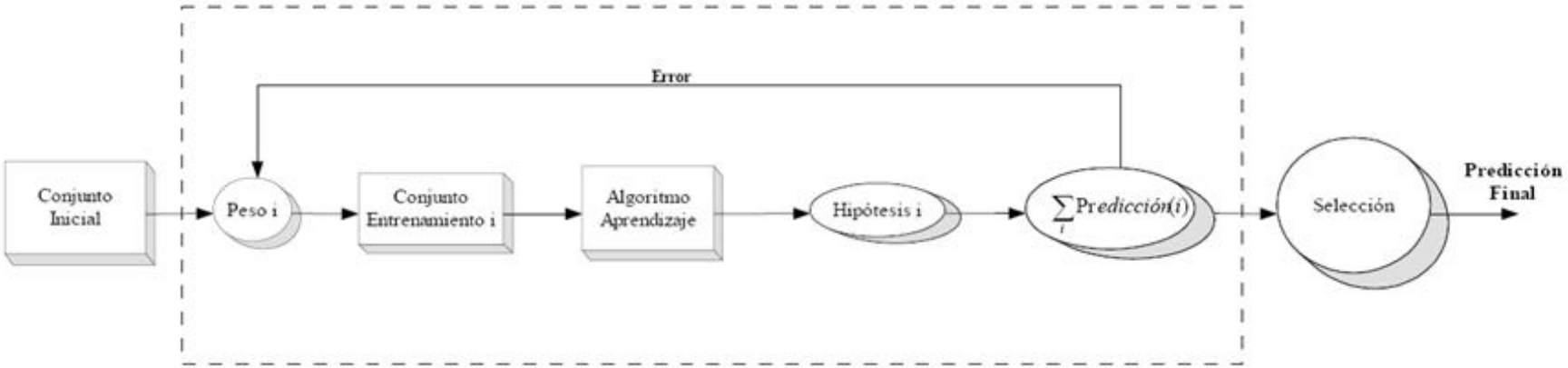


Nombre	Fecha de modifica...	Tipo	Tamaño
weka-3-8-1jre-x64.exe	03/02/2017 8:43	Aplicación	109.373 KB
attributeSelectionSearchMethods1.0.7.zip	23/02/2017 14:30	zip Archive	123 KB
averagedOneDependenceEstimators1.2.1.zip	23/02/2017 14:32	zip Archive	84 KB
bayesianLogisticRegression1.0.5.zip	23/02/2017 14:27	zip Archive	111 KB
decorate1.0.1.zip	02/02/2017 23:19	zip Archive	47 KB
decorate1.0.2.zip	23/02/2017 14:33	zip Archive	47 KB
distributedWekaBase1.1.17.zip	02/02/2017 23:24	zip Archive	3.379 KB
distributedWekaHadoop21.0.17.zip	07/02/2017 7:55	zip Archive	7 KB
distributedWekaSpark1.1.9.zip	07/02/2017 7:47	zip Archive	56.800 KB
ensembleLibrary1.0.5.zip	02/02/2017 23:34	zip Archive	480 KB
ensemblesOfNestedDichotomies1.0.6.zip	23/02/2017 14:25	zip Archive	170 KB
graphviz-treevisualize-2014.8.1.zip	02/02/2017 23:43	zip Archive	62 KB
graphviz-treevisualize-weka-package-2014.8.1.zip	02/02/2017 23:43	zip Archive	7.089 KB
gridSearch1.0.7.zip	23/02/2017 14:30	zip Archive	106 KB
kernelLogisticRegression1.0.0.zip	02/02/2017 23:35	zip Archive	68 KB
massiveOnlineAnalysis1.0.3.zip	07/02/2017 7:54	zip Archive	1.299 KB
metaCost1.0.0.zip	23/02/2017 14:35	zip Archive	44 KB
multilayerPerceptronCS1.0.1.zip	02/02/2017 23:18	zip Archive	95 KB
multiLayerPerceptrons1.0.9.zip	02/02/2017 23:14	zip Archive	210 KB
RBFNetwork1.0.6.zip	02/02/2017 23:15	zip Archive	136 KB
RPlugin1.3.20.zip	07/02/2017 7:45	zip Archive	1.123 KB
simpleCART1.0.0.zip	02/02/2017 23:20	zip Archive	63 KB
streamingUnivariateStats1.0.1.zip	02/02/2017 23:30	zip Archive	45 KB
timeseriesForecasting1.0.24.zip	02/02/2017 23:29	zip Archive	2.476 KB
wekaDeeplearning4jCore1.1.3.zip	02/02/2017 23:27	zip Archive	12.955 KB
WekaExcel1.0.5.zip	02/02/2017 23:33	zip Archive	8.723 KB
wekaPython1.0.3.zip	07/02/2017 7:50	zip Archive	1.822 KB
XMeans1.0.2.zip	23/02/2017 14:33	zip Archive	70 KB

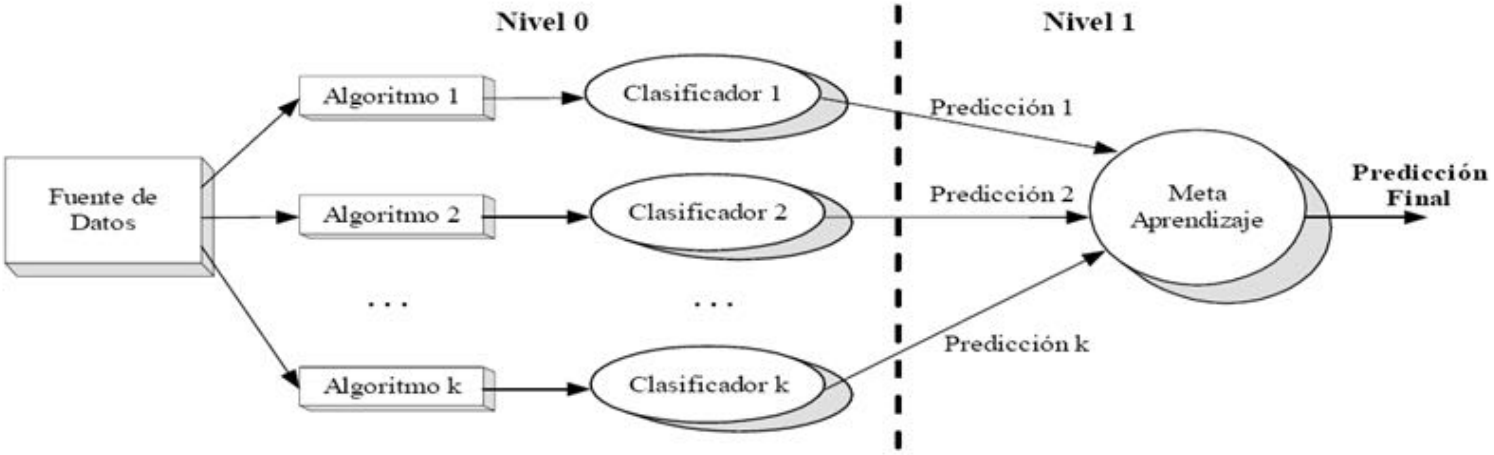
Estructura del multiclasificador Bagging

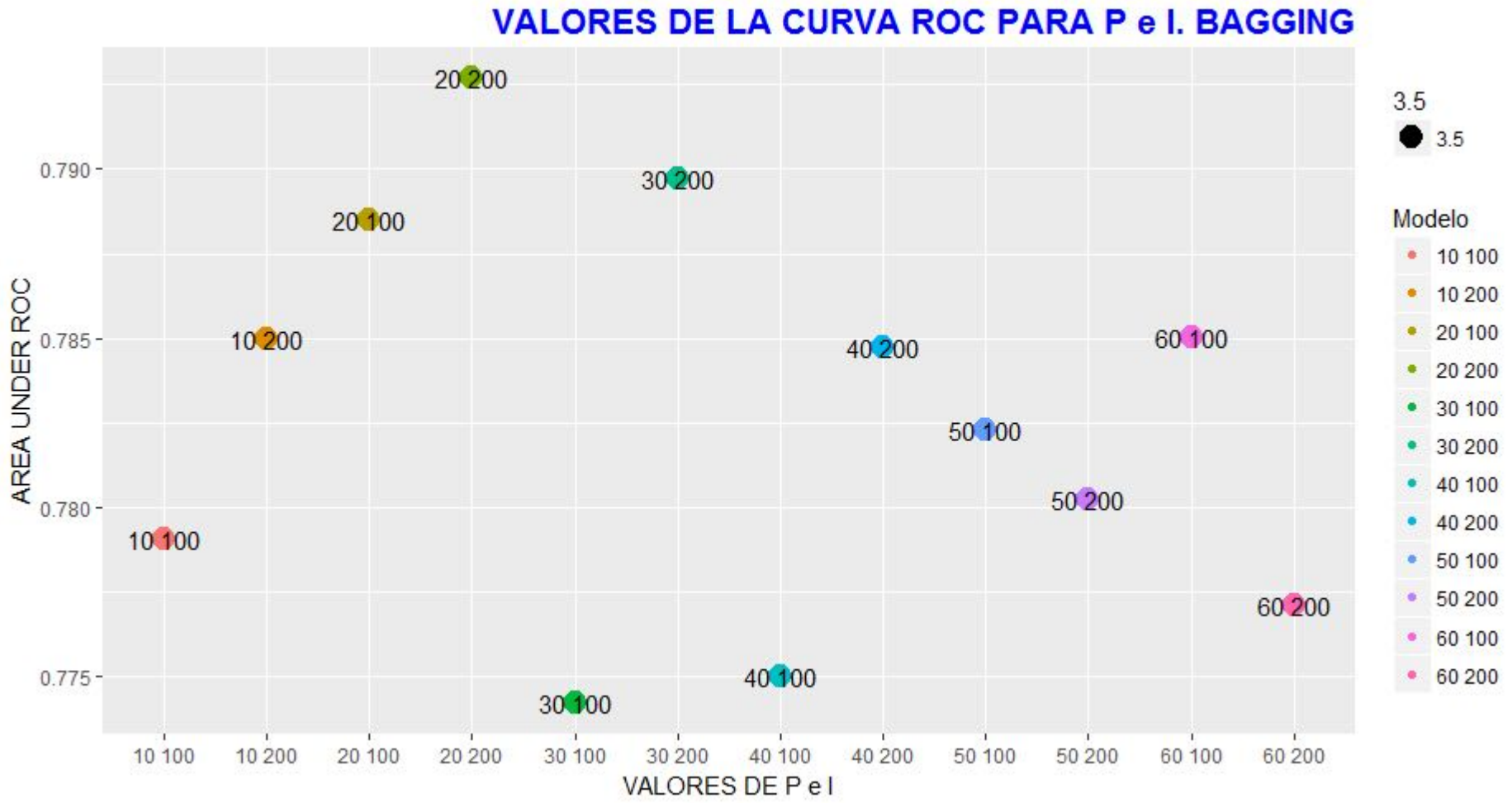


Estructura del multiclasificador Boosting



Estructura del multiclasificador Stacking




VALORES DE LA CURVA ROC PARA P e I. BAGGING
AREA UNDER ROC
0.790
0.785
0.780
0.775
10 100
10 200
20 100
20 200
30 100
30 200
40 100
40 200
50 100
50 200
60 100
60 200
VALORES DE P e I
3.5
3.5
Modelo
10 100
10 200
20 100
20 200
30 100
30 200
40 100
40 200
50 100
50 200
60 100
60 200

Home / Browse / Science & Engineering / Artificial Intelligence / Machine Learning / Weka / Files

Weka

Machine learning software to solve data mining problems

Brought to you by: eibe, fracpete, mbatchelor, weka

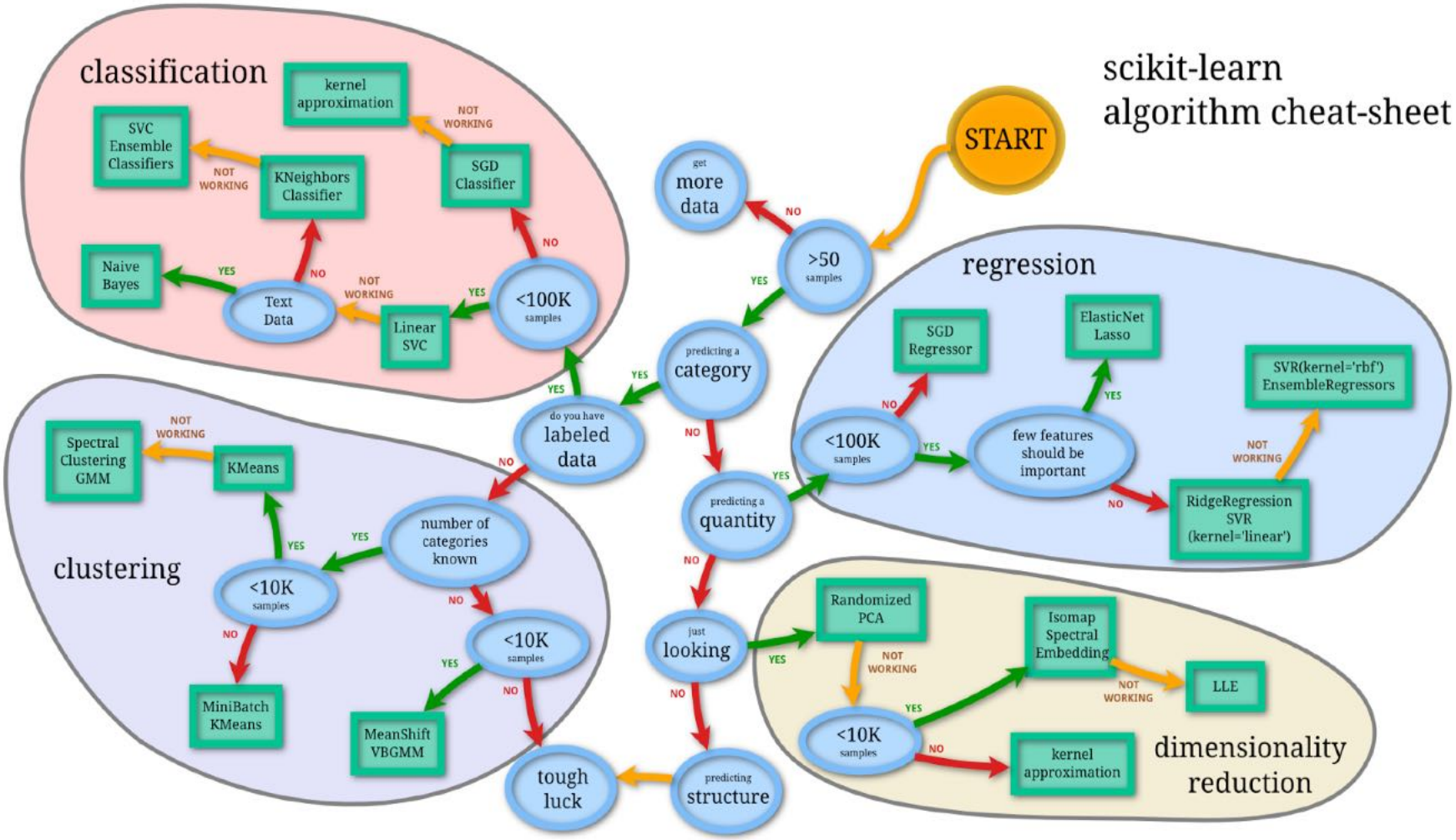
Summary | Files | Reviews | Support | Wiki | Code | News

Looking for the latest version? **Download weka-3-8-1jre-x64.exe (112.0 MB)**

Home / weka-packages

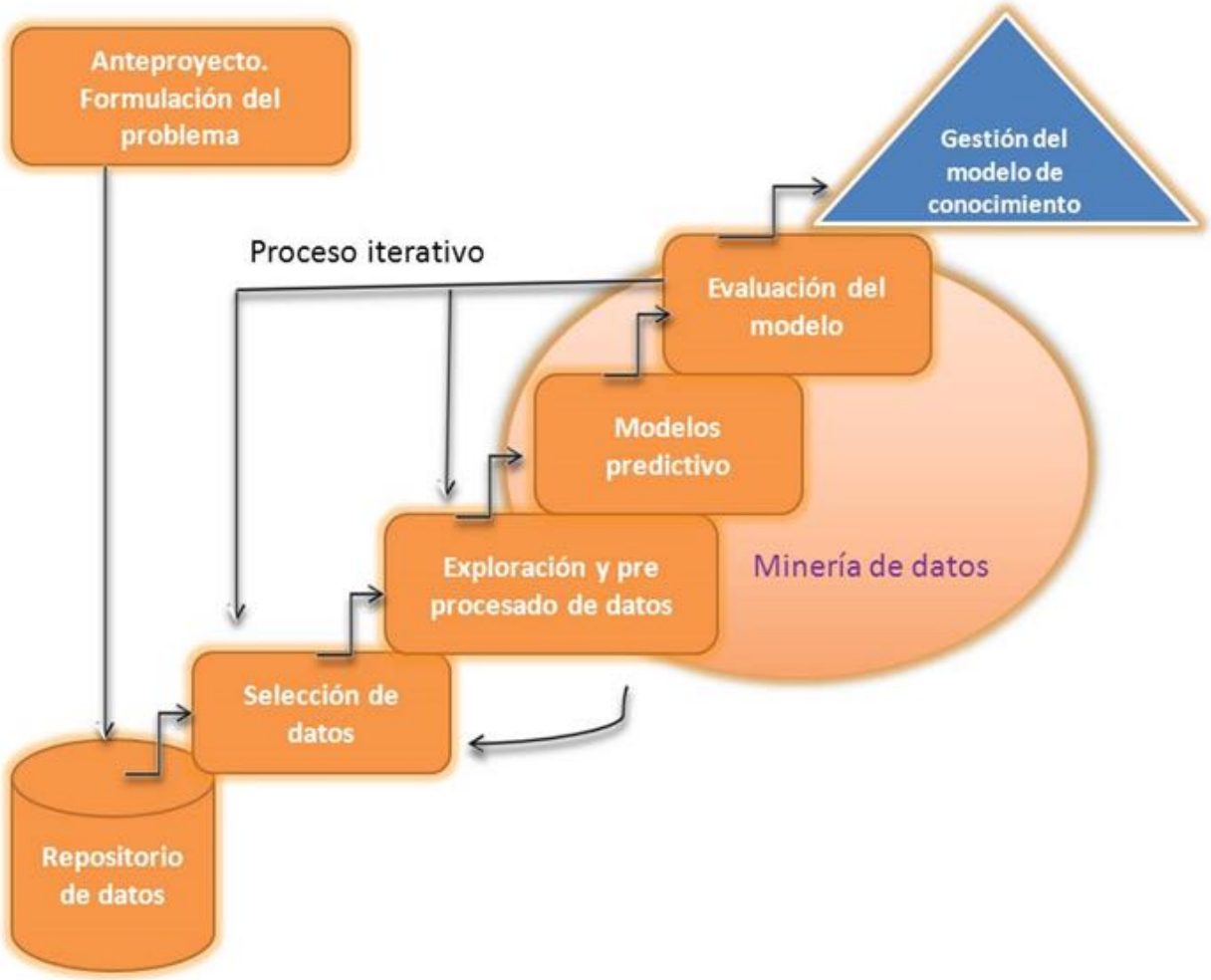
Name	Modified	Size	Downloads / Week
Parent folder			
wekaDeeplearning4jGPULibs1.0.1.zip	2017-02-23	196.6 MB	70
wekaDeeplearning4jGPU1.0.3.zip	2017-02-23	955 Bytes	14
multiInstanceLearning1.0.9.zip	2017-02-21	559.0 kB	69
ensemblesOfNestedDichotomies1.0....	2017-02-21	173.7 kB	65
distributedWekaSpark1.1.9.zip	2017-02-02	58.2 MB	22
distributedWekaSpark1.0.9.zip	2017-02-02	58.2 MB	52
distributedWekaBase1.0.17.zip	2017-02-02	3.5 MB	64
distributedWekaBase1.1.17.zip	2017-02-02	3.5 MB	21
RPlugin1.3.20.zip	2017-01-23	1.1 MB	21
RPlugin1.2.20.zip	2017-01-23	1.1 MB	71
timeseriesForecasting1.1.25.zip	2017-01-18	2.5 MB	40

http://scikit-learn.org/dev/tutorial/machine_learning_map/index.html
"Chuleta" de algoritmos de Machine Learning



Fuente: http://1.bp.blogspot.com/-ME24ePzpzIM/UQLWTwurfXI/AAAAAAAAANw/W3EETIroA80/s1600/drop_shadows_background.png)

Fases de la metodología aplicada en la tesis doctoral



1. **Formulación del problema. Integración de la información.**
2. **Selección de datos, limpieza y transformación de la base de datos. (Datos faltantes)**
3. **Exploración y preprocesado de los datos. (Variables muy asimétricas, muchos outliers)**
4. **Análisis y evaluación de los modelos predictivos.**
5. **Gestión del modelo de conocimiento.**

Cuestiones importantes:

- **Equilibrado de la muestra**
- **Selección de variables.**
- **Uso de la muestras (Validación cruzada o Training/Validación/Test)**
- **Métodos de evaluación de modelos de clasificación:**
  - **Basados en métricas.**
  - **Basados en curvas ROC.**
  - **Métodos que incorporan una matriz de costes.**
- **Contrastación de los modelos**

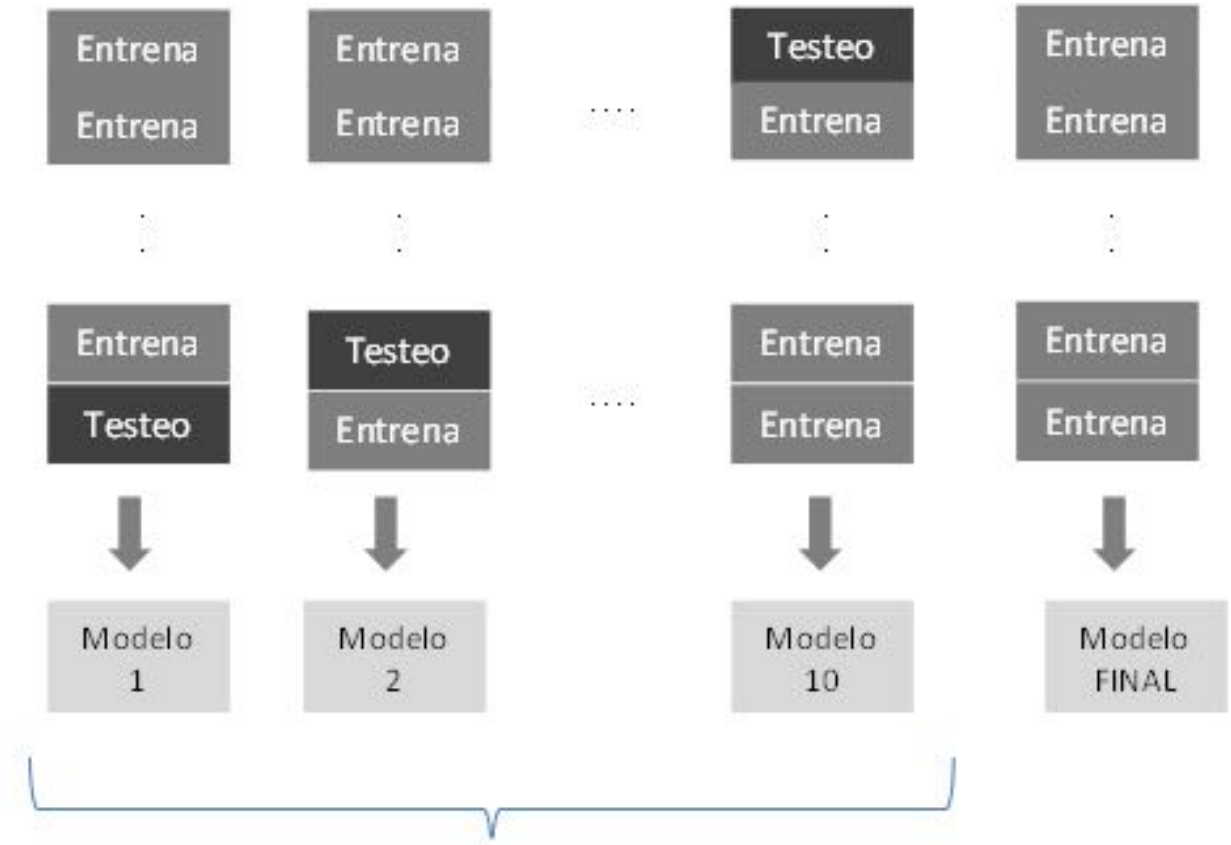
Muestra desbalanceada

Técnica	CLASE SÍ (%)	CLASE NO (%)	TOTAL (%)	AREA ROC
Regresión Logística	97,5	31.3	90,9	0,777
C 4.5	97,9	35,2	91,7	0,885
Maq. Vect. Soporte	99,9	0,1	89,9	0,500
Perceptrón Mult.	94,6	35,8	88,7	0,817
Redes Base Radial	100,0	0,0	90,0	0,825
Naïve Bayes	66,4	81,6	68,0	0,832
Red Bayesiana(TAN)	95,4	49,2	90,8	0,885
Red Bayesiana(K2)	88,3	69,8	86,4	0,884
AODE1	91,9	64,8	89,1	0,894
AODE2	94,5	51,4	90,2	0,896
Metaclasificadores				
Bagging	99,3	20,7	91,4	0,879
Adaboost	97,3	39,7	91,5	0,893
Random Forest	98,4	33,5	91,9	0,846
Random Committee	99,8	14,0	91,2	0,891
RandomSubSpace	98,9	38,2	92,3	0,888
STAKING C (5 modelos)	97,7	24,6	90,4	0,772
Decorate	97,1	37,4	91,1	0,860
Metacost 1/1	97,1	43,6	91,7	0,787
Metacost 3/1	94,9	49,2	90,3	0,831
Metacost 9/1	86,9	75,4	85,8	0,828

Muestra balanceada

Técnica	%CLASE SÍ	%CLASE NO	% CLASE TOTAL	ROC AREA
Regresión Logística	64,3	91,7	76,9	0,893
C 4.5	80,8	79,6	80,2	0,771
Maq. Vect. Soporte	73,7	75,4	74,6	0,746
Perceptrón Mult.	72,5	73,1	72,8	0,823
Redes Base Radial	75,4	74,3	74,9	0,809
Naïve Bayes	80,2	81,4	80,8	0,881
Red Bayesiana(TAN)	81,4	81,4	81,4	0,873
Red Bayesiana(K2)	80,2	81,4	80,8	0,881
AODE1	80,2	82,0	81,1	0,887
AODE2	79,6	80,8	80,2	0,885
Metaclasificadores				
Bagging	80,2	80,2	80,2	0,860
Adaboost	85,4	77,8	81,4	0,891
Random Forest	82,6	82,0	82,3	0,886
Random Committee	81,4	81,4	81,4	0,874
RandomSubSpace	79,6	83,2	81,4	0,882
STAKING C (5 modelos)	74,5	80,2	77,5	0,749
Decorate	81,4	81,4	81,4	0,816
Metacost	82,0	80,8	81,4	0,810

Esquema de validación cross validation



$$ErrorTesteoFinal = \frac{ErrorTesteo1 + ErrorTesteo2 + ... + ErrorTesteo10}{10}$$

Métodos de evaluación más habituales utilizados en la clasificación

- **MÉTODOS BASADOS EN MÉTRICAS.**
- **MÉTODOS BASADOS EN CURVAS ROC.**
- **MÉTODOS QUE INCORPORAN UN MATRIZ DE COSTES.**

		Clase clasificada como:		
		A+ (SI)	A- (NO)	Total
Estado real	A+ (SI)	Verdaderos positivos (VP) $FVP = \frac{VP}{TCP}$	Falsos negativos (FN) $FFN = \frac{FN}{TCP}$	1
	A- (NO)	Falsos positivos (FP) $FFP = \frac{FP}{TCA}$	Verdaderos negativos (TN) $FVN = \frac{VN}{TCA}$	1

La precisión, exactitud o accuracy (AC) de un clasificador es el cociente entre el número de ejemplos que están bien clasificados, que se corresponde en la matriz de confusión con la suma de los elementos de la diagonal, entre el total de instancias.

$$AC = \frac{VP + VN}{N}$$

$$F - Measure = \frac{2 \cdot \text{Precisión} \cdot \text{Recall}}{\text{Precisión} + \text{Recall}}$$

Otro índice muy utilizado es el estadístico Kappa. Es un coeficiente estadístico que determina la precisión del modelo a la hora de predecir la clase verdadera. Este estadístico está ampliamente difundido.

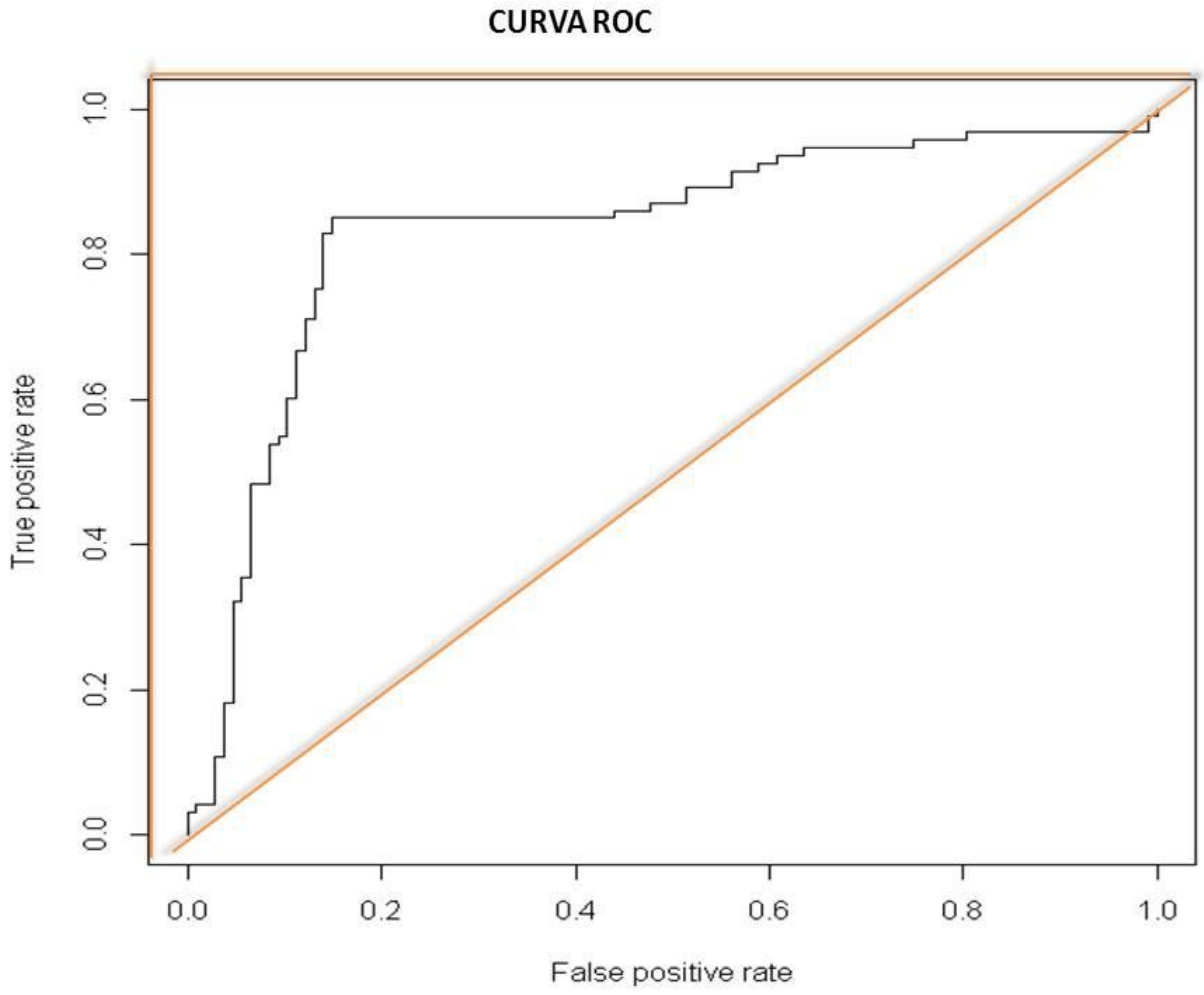
$$k = \frac{P(A) - P(E)}{1 - P(E)}$$

P(A) es el porcentaje de casos acertados y P(E) es el porcentaje de casos cambiados. Para medir P(E) existen varias formas. El programa WEKA, utilizado en esta tesis lo proporciona de forma habitual lo calcula a través de la siguiente fórmula:

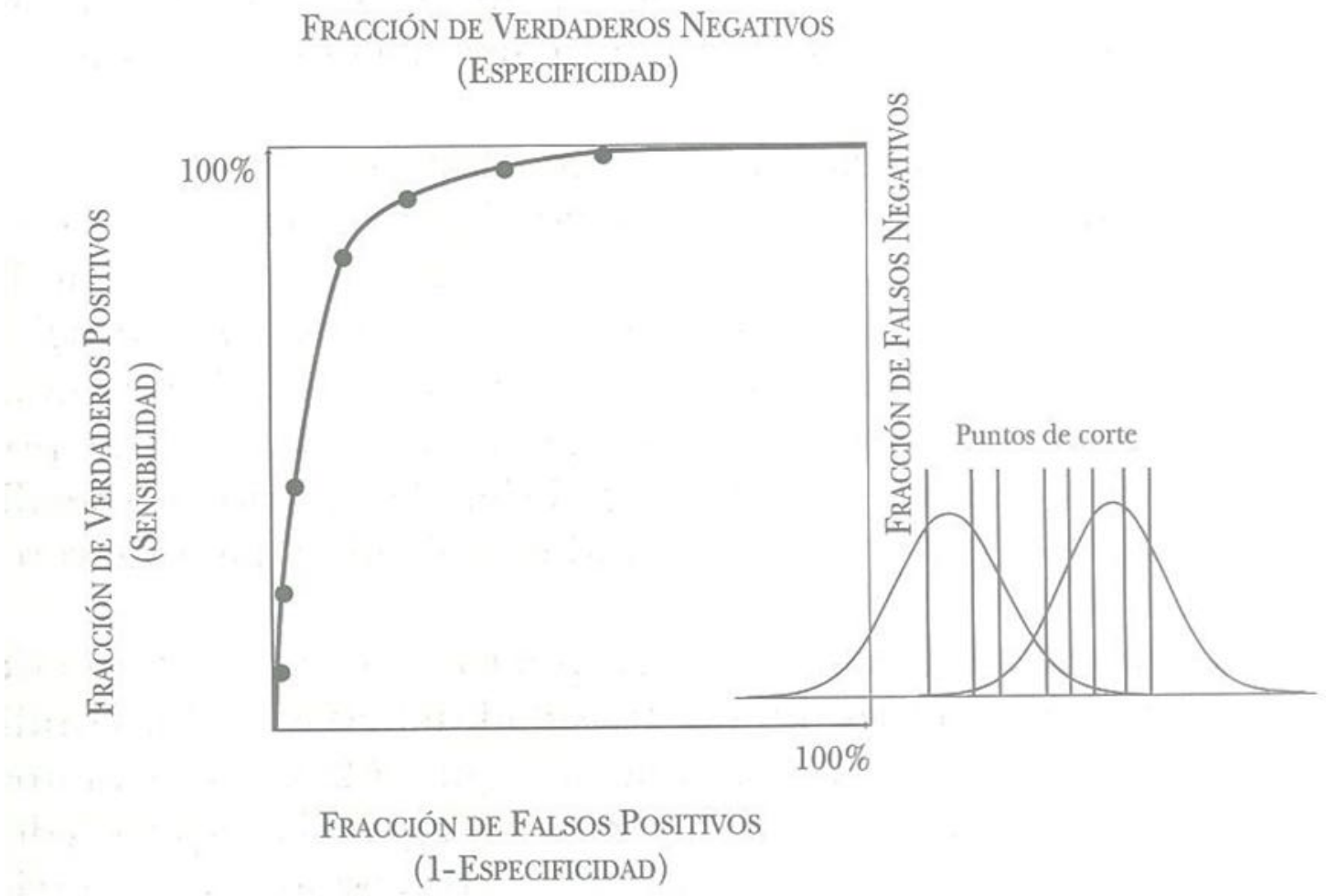
$$P(A) = \frac{SumaDiagonales}{NúmeroCasos} = \frac{VP + VN}{N}$$

$$P(E) = \frac{(VP + FN) \cdot (VP + FP}{2N} + \frac{(VN + FP) \cdot (VN + FN}{2N}$$

La curva ROC (Receiver Operating Characteristic) es una representación gráfica del rendimiento de un clasificador que muestra la distribución de las fracciones de verdaderos positivos y la fracción de falsos negativos.



Curva ROC y posibles criterios de decisión



Fuente: Franco y Vivo (2007)

Los factores de riesgo de los modelos de credit scoring, es decir, los factores que están detrás de los errores tipo I (admitir como sana una operación insolvente) y tipo II (rechazar como insolvente una operación sana) no son los mismos.

$$C_e = \pi_{no} C_I + \pi_{si} C_{II}$$

CHAID. **Chi-squared Automatic Interaction Detection (detector automático de interacciones mediante Ji cuadrado)**

CART. Classification And Regression Trees (Árboles de decisión y de regresión)

QUEST. Quick, Unbiased, Efficient Statistical Tree (árbol estadístico eficiente, insesgado y rápido)

C 4.5

Randomm Forest

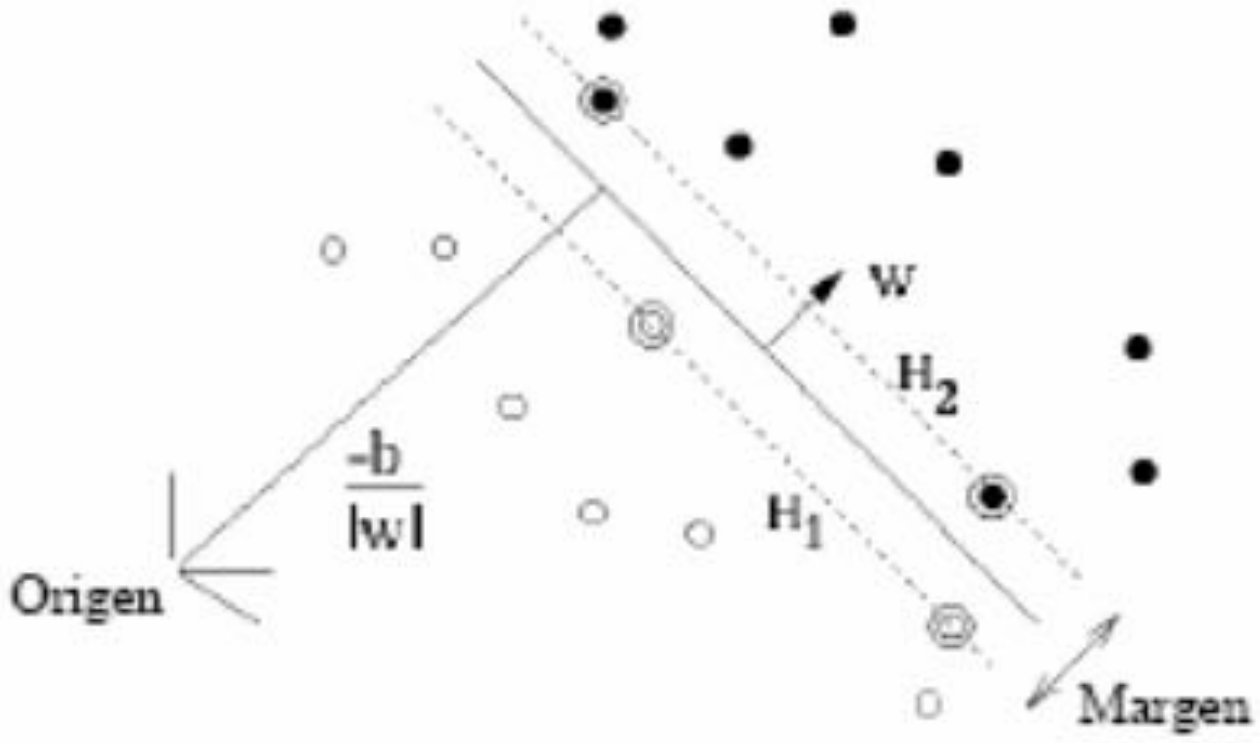
Decisión Stum

…….

Los fundamentos teóricos de las máquinas de vectores soporte (Support Vector Machines, SVM) fueron presentados en el año 1992 en la conferencia COLT (Computacional Learning Theory) por Boser et al. (1992)

Las máquinas de vectores soporte pertenecen a la familia de los clasificadores lineales dado que inducen hiperplanos o separadores lineales de muy alta dimensionalidad introducidos por funciones núcleo o kernel. Es decir, el enfoque de las SVM adopta un punto de vista no habitual, en vez de reducir la dimensión buscan una dimensión mayor en la cual los puntos puedan separarse linealmente.

Separación de datos con margen máximo



Fuente: Burges (1998)

Hay que tener en cuenta que la dimensión del espacio necesario para separar los datos puede ser grande aumentando el coste computacional. Sin embargo, existe una forma muy efectiva de calcular los productos escalares en el espacio de las características a través de ciertas transformaciones usando las denominadas funciones núcleo (funciones kernel).

Una función kernel es una función K: X . X→ R tal que K(X, Y) = <Φ(x), Φ(z)> donde Φ es una transformación de X en un espacio de Hilbert, H.
Sin embargo, hay una gran cantidad de posibles funciones núcleo que pueden ser utilizadas para crear tal espacio de características de alta dimensional.

Funciones kernel

$$K(x_i, y_j) = (x_i.y_j + 1)^p$$

$$K(x_i, y_j) = \exp\left(-\frac{\|x_i - y_j\|^2}{2\sigma^2}\right)$$

$$K(x_i, y_j) = \tanh(ax_i.y_j + b) \quad a, b \in R$$

$$K(x_i, y_j) = \frac{1}{\sqrt{\|x_i - y_j\|^2 + c^2}} \quad c \geq 0$$

$$K(x_i, y_j) = \left(\sum_i \exp(-\gamma(x_i - y_j)^d\right.$$

$$K(x_i, y_j) = \sum_{r=0}^{k}(x_i^r y_j^r) + \sum_{s=1}^{N}(x_i - t_s)^k + (y_j - t_s)^k$$

$$K(x_i, y_j) = \sum_n K_n(x_i, y_j)$$

$$K(x_i, y_j) = \prod_n K_n(x_i, y_j)$$

$$K(x_i, y_j) = \frac{1}{\left[1 + \left(\frac{2\sqrt{\| x_i - y_j \|^2}\ \sqrt{2^{(1/w)} - 1}}{\sigma}\right)^2\right]^w}$$

$$K(x_i, y_j) = \frac{sen(N + 1/2)(x_i - y_j)}{sen(1/2(x_i - y_j))}$$

$$Logit\left(P\left(E\right)\right)=\ln\left(\frac{P\left(E\right)}{1-P\left(E\right)}\right)$$

$$Logit\left(P\left(E\right)\right)=\ln\left(\frac{P\left(E\right)}{1-P\left(E\right)}\right)=\beta_0+\beta_1x_1+\beta_2x_2+.....+\beta_px_p$$

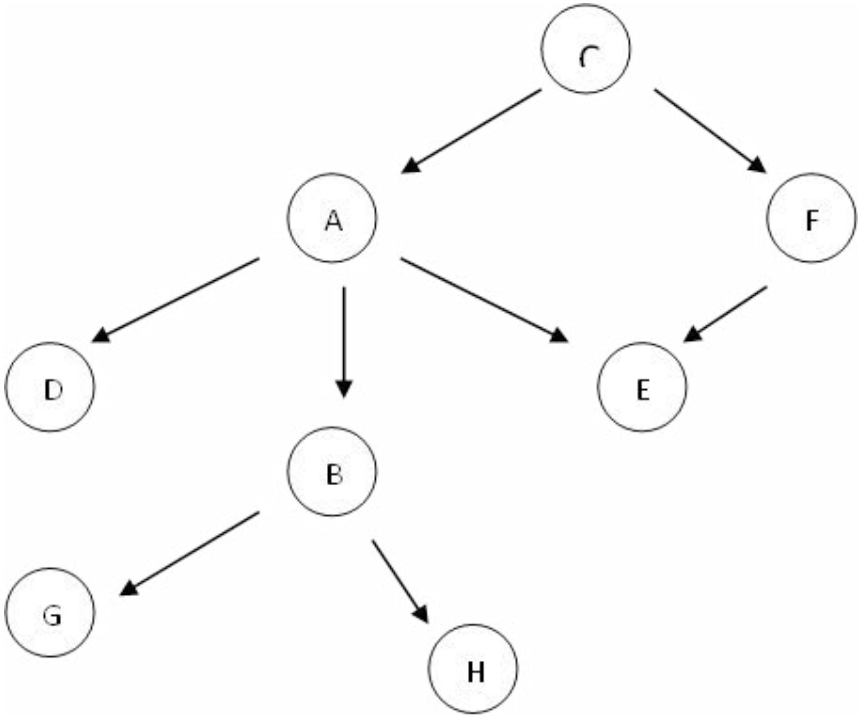
$$P(Y=1)=\frac{\exp(\beta_0+\beta_1x_1+\beta_2x_2+.....+\beta_px_p)}{1+\exp(\beta_0+\beta_1x_1+\beta_2x_2+.....+\beta_px_p)}$$

Redes Bayesiana

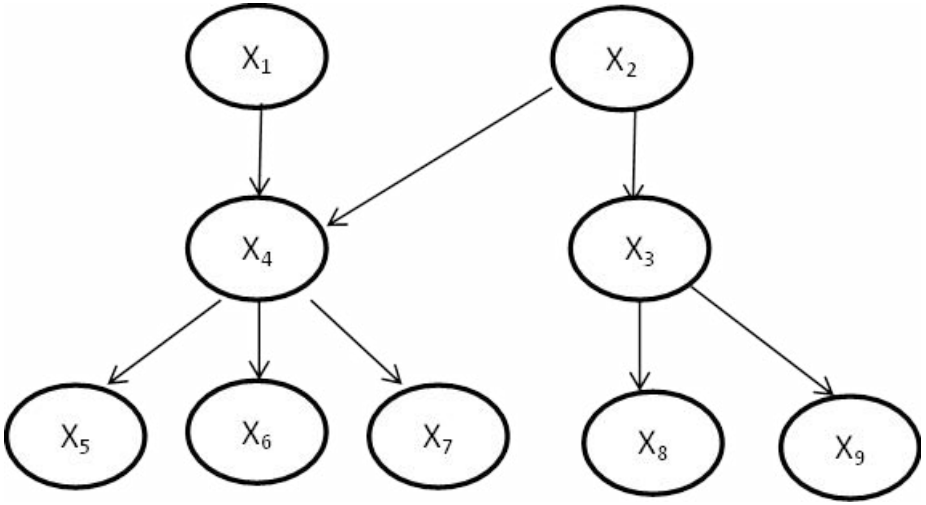
En los problemas de clasificación disponemos de una variable clase (C) y un conjunto de variables predictoras o atributos que denominaremos A_1, A_2…A_n. Con estas especificaciones el teorema de Bayes tiene la siguiente expresión:

$$P(C / A_1, A_2, \ldots.. A_n) = \frac{P(C)P(A_1, A_2, \ldots.. A_n / C)}{P(A_1, A_2, \ldots.. A_n)}$$

Una red bayesiana queda especificada formalmente por una dupla B=(G,Θ) donde G es un grafo dirigido acíclico (GDA) y Θ es el conjunto de distribuciones de probabilidad. Definimos un grafo como un par G= (V, E), donde V es un conjunto finito de vértices nodos o variables y E es un subconjunto del producto cartesiano V x V de pares ordenados de nodos que llamamos enlaces o aristas.



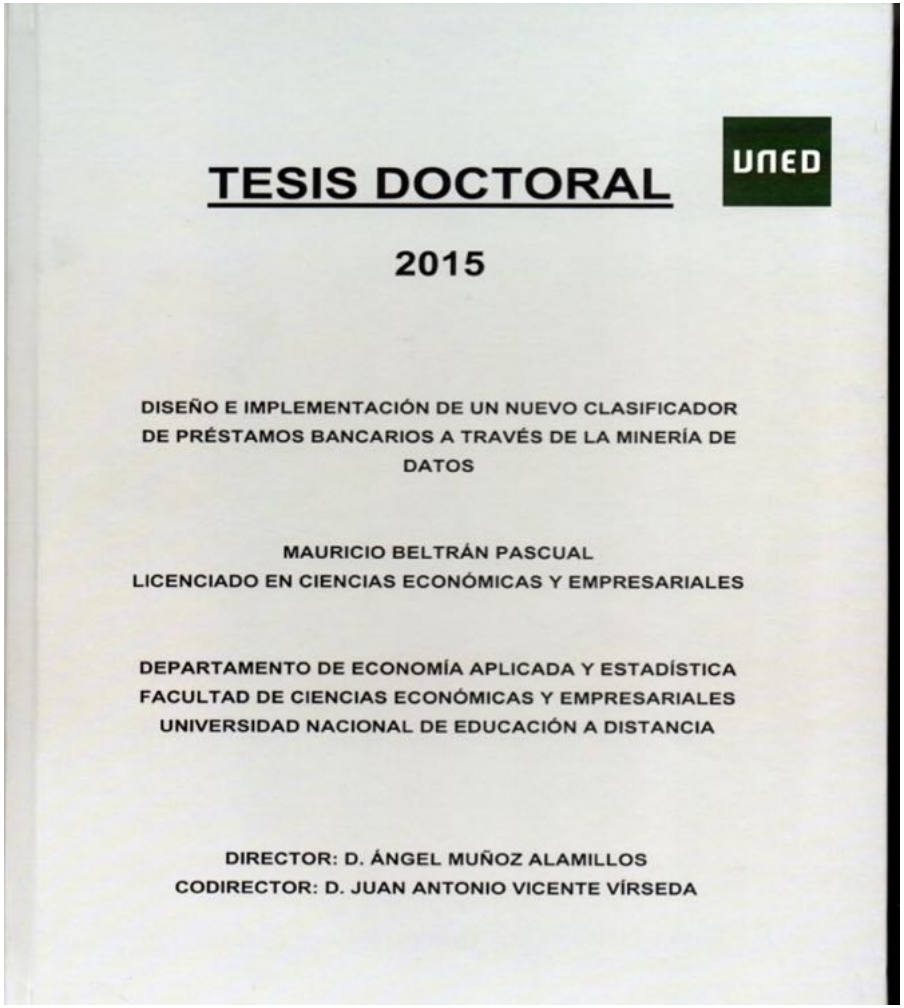
Topología de una red con nueve parámetros



$$p(x_i, x_2, \ldots\ldots x_n) = \prod_{i=1}^{n} p\left(x_i \,\middle|\, padres(x_i)\right)$$

Los grafos definen un modelo probabilístico con las mismas dependencias utilizando una factorización mediante el producto de varias funciones de probabilidad condicionada

RECURSOS



TESIS DOCTORAL

UNED

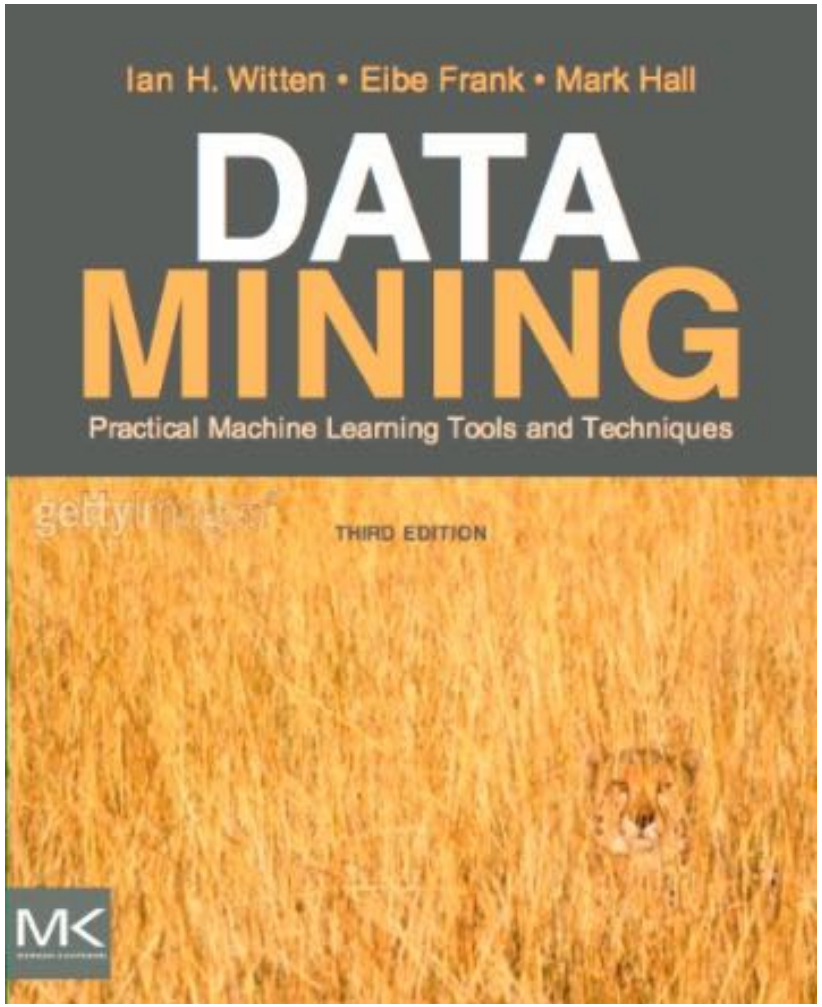
2015

DISEÑO E IMPLEMENTACIÓN DE UN NUEVO CLASIFICADOR DE PRÉSTAMOS BANCARIOS A TRAVÉS DE LA MINERÍA DE DATOS

MAURICIO BELTRÁN PASCUAL
LICENCIADO EN CIENCIAS ECONÓMICAS Y EMPRESARIALES

DEPARTAMENTO DE ECONOMÍA APLICADA Y ESTADÍSTICA
FACULTAD DE CIENCIAS ECONÓMICAS Y EMPRESARIALES
UNIVERSIDAD NACIONAL DE EDUCACIÓN A DISTANCIA

DIRECTOR: D. ÁNGEL MUÑOZ ALAMILLOS
CODIRECTOR: D. JUAN ANTONIO VICENTE VÍRSEDA



https://onedrive.live.com/redir?resid=A56AD5453314F568!506&authkey=!ADSa2MUaoFKn30I&ithint=folder%2crar

RECURSOS

- Weka in the ecosystem for Scientific Computing. (Universidad de Waikato)
- R talk to Weka about Data Mining (https://www.r-bloggers.com/r-talks-to-weka-about-data-mining/
- Open Source machine Learning: R Meets Weka (Kurt Hornik et al.)

- Rweka Odd and End Weka (Kurt Hornik)

GRACIAS

mauricio.beltran@jcyl.es
beltranpascual@gmail.com