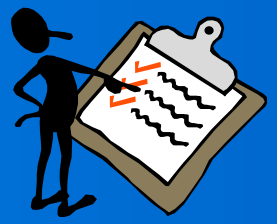
Inteligencia Artificial

Técnicas de clasificación

Prof. Dra. Silvia Schiaffino
ISISTAN - CONICET

Clasificación: Agenda

- Concepto
  - Clasificación
  - Predicción
  - Evaluación
- Árboles de Decisión
  - Construcción
  - Uso
  - Poda
- Clasificador Bayesiano
- Ejemplos



Supervisado vs. No Supervisado

- Aprendizaje Supervisado: Clasificación
  - Conocemos las clases y el número de clases
- Aprendizaje No Supervisado: Clustering
  - No conocemos las clases y podemos no conocer el número de clases

Concepto

- El objetivo de la clasificación de datos es organizar y categorizar los datos en clases diferentes
  - Se crea un modelo basándose en la distribución de los datos
  - El modelo es luego usado para clasificar nuevos datos
  - Dado el modelo, se puede predecir la clase de un nuevo dato

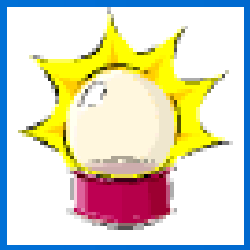
- Clasificación = predicción para valores discretos y nominales

Predicción

- El objetivo de la predicción es deducir el valor de un atributo basándose en los valores de otros atributos
  - Se crea un modelo basándose en la distribución de los datos
  - El modelo se usa para predecir valores futuros o desconocidos

Si se deduce un valor discreto → Clasificación

SI se deduce un valor continuo → Predicción (Regresión)



Definición de clasificación

Dada una base de datos $D = \{t1, t2, \ldots, tn\}$ de tuplas o registros (individuos) y un conjunto de clases $C = \{C1, C2, \ldots, Cm\}$, el problema de la clasificación es encontrar una función

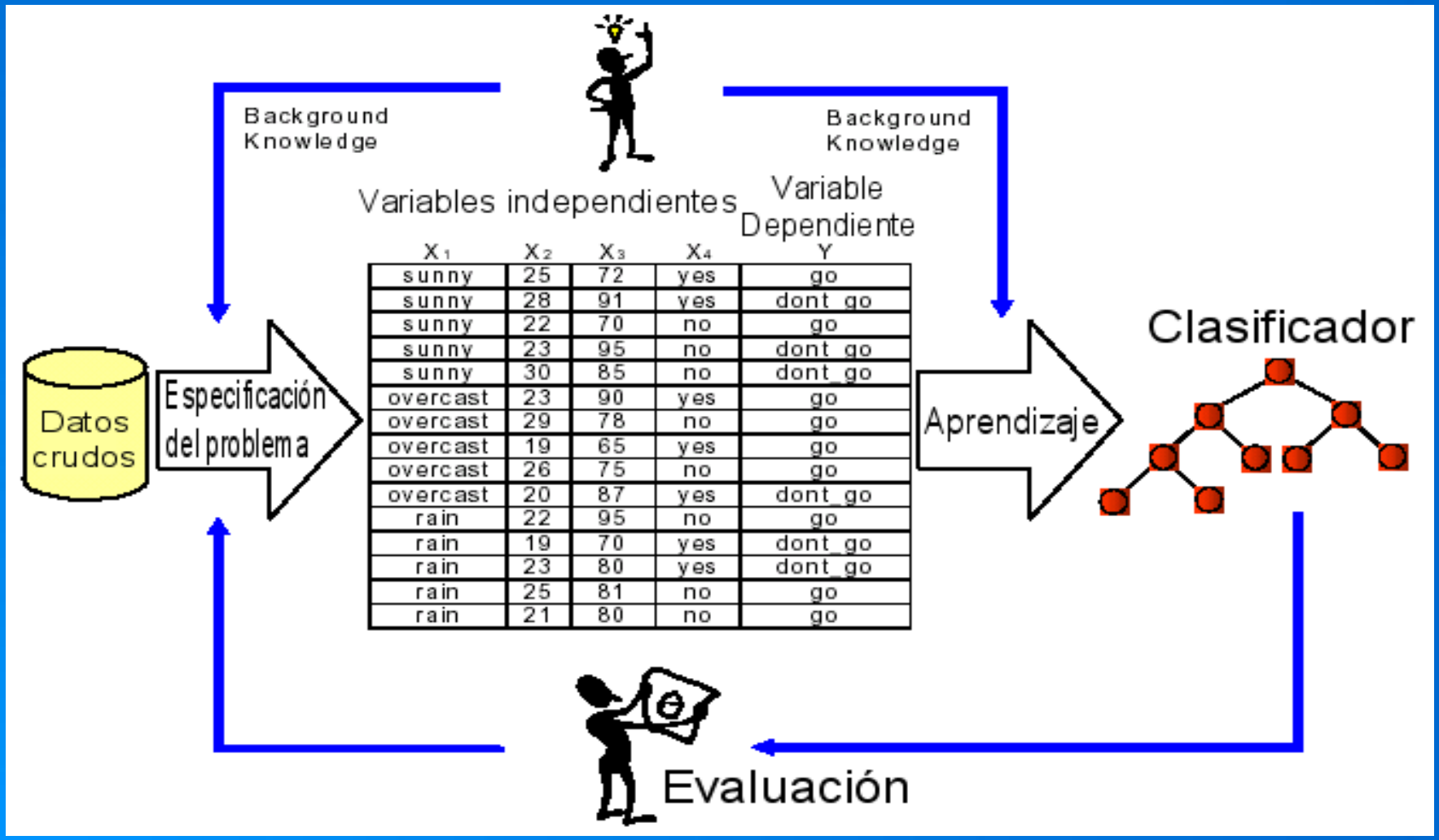
$f: D \rightarrow C$ tal que cada ti es asignada una clase Cj .

$f: D \rightarrow C$ podría ser una Red Neuronal, un Árbol de Decisión, un clasificador Bayesiano...

Aplicaciones

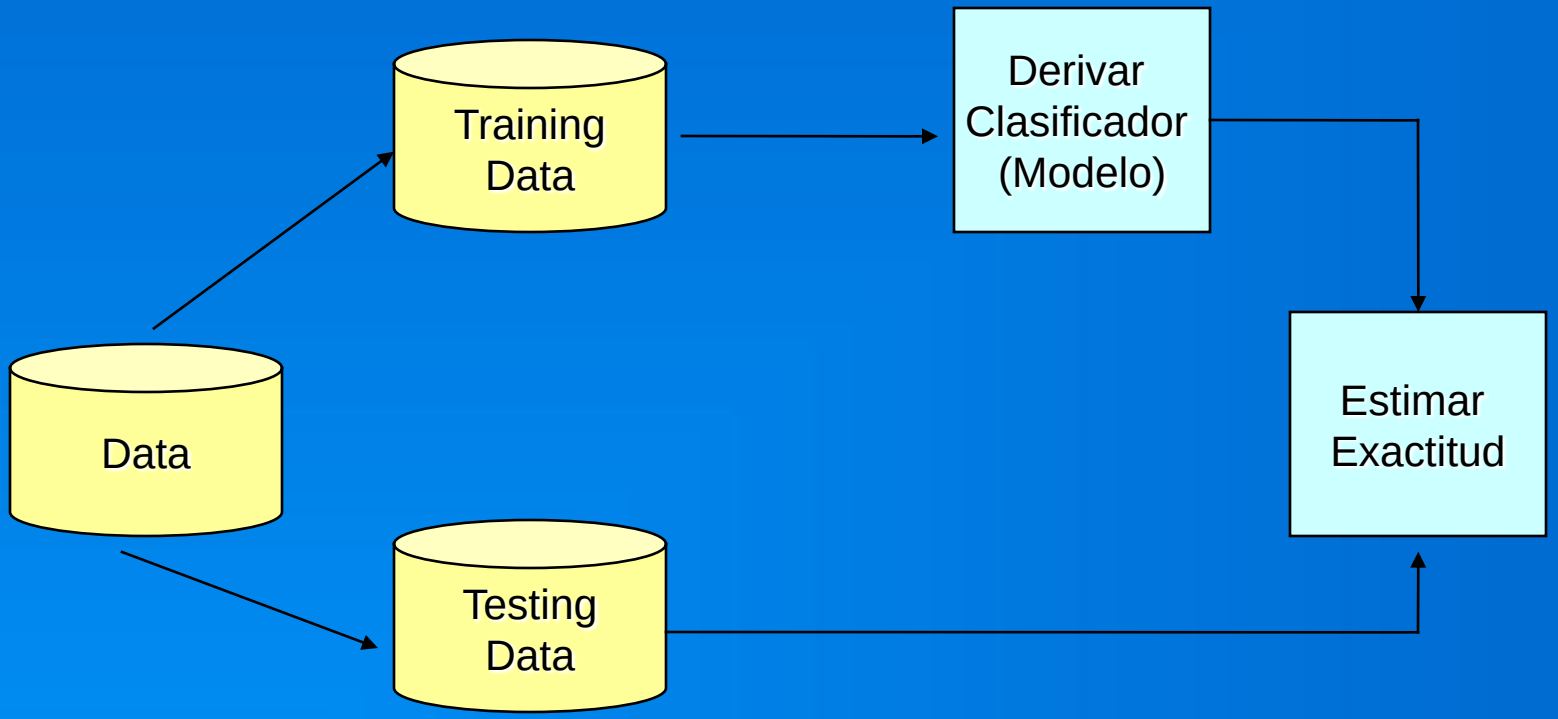
- Aprobación de créditos
- Marketing
- Diagnóstico médico
- Identificación de partes defectuosas en manufactura
- Zonificación (crímenes)
- Clasificación de e-mail
- Clasificación de documentos
- ...

Proceso de clasificación



X_1	X_2	X_3	X_4	Y
sunny	25	72	yes	go
sunny	28	91	yes	dont_go
sunny	22	70	no	go
sunny	23	95	no	dont_go
sunny	30	85	no	dont_go
overcast	23	90	yes	go
overcast	29	78	no	go
overcast	19	65	yes	go
overcast	26	75	no	go
overcast	20	87	yes	dont_go
rain	22	95	no	go
rain	19	70	yes	dont_go
rain	23	80	yes	dont_go
rain	25	81	no	go
rain	21	80	no	go

Etapas: Terminología



Etapas: Construcción del modelo - Aprendizaje

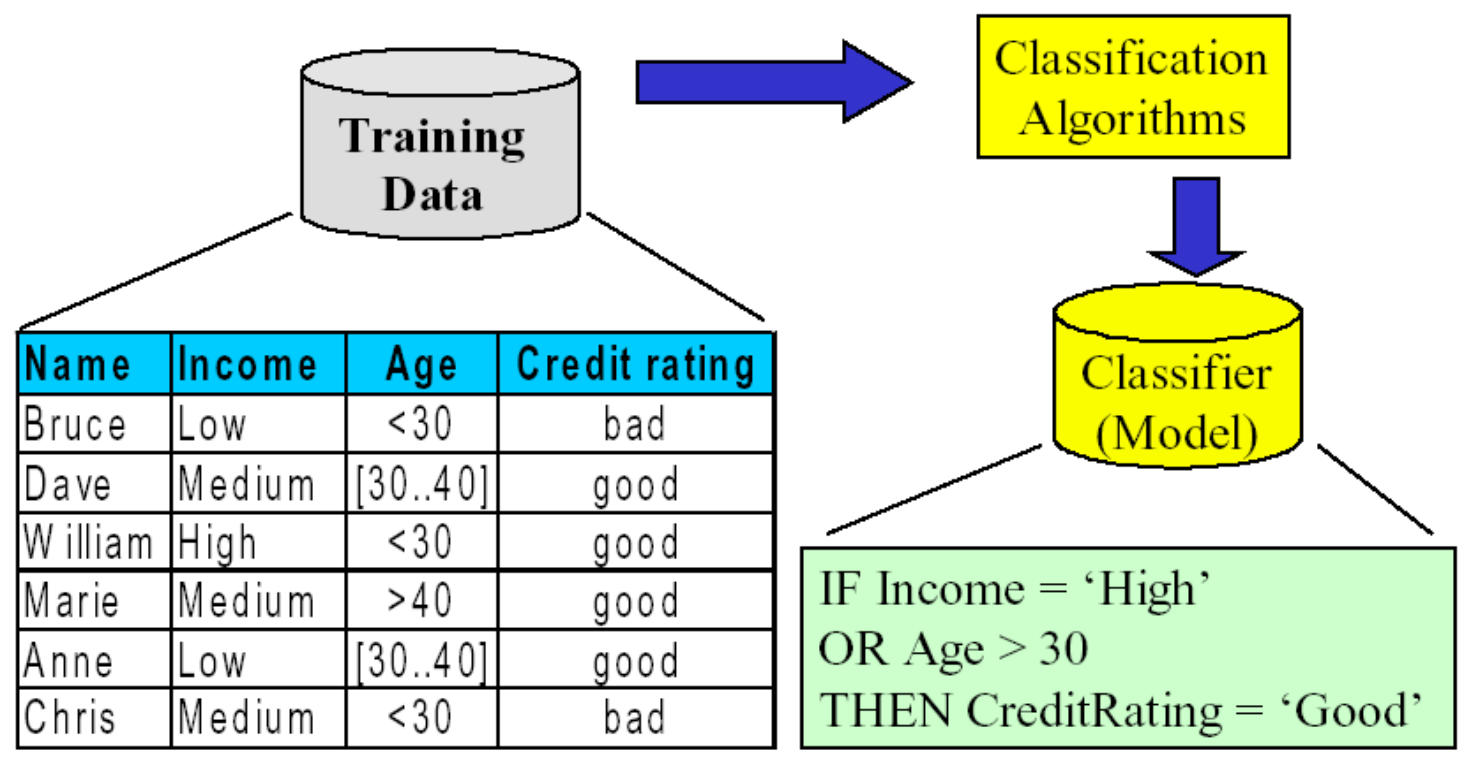
Cada tupla se supone que pertenece a una clase predefinida, dada por uno de los atributos, llamada etiqueta de clase

El conjunto de todas las tuplas usadas para la construcción del modelo se llama conjunto de entrenamiento

El modelo se representa mediante alguna de las siguientes formas:

- Reglas de clasificación (sentencias IF-THEN)
- Árbol de decisión
- Fórmulas matemáticas
- …

Aprendizaje

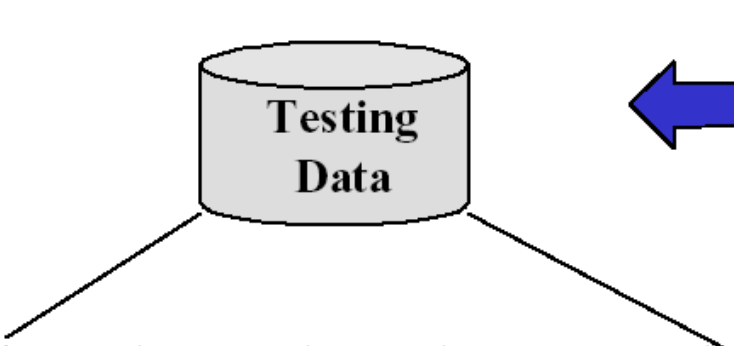


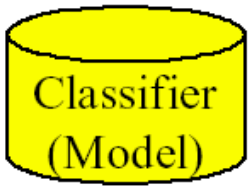
Name	Income	Age	Credit rating
Bruce	Low	<30	bad
Dave	Medium	[30..40]	good
William	High	<30	good
Marie	Medium	>40	good
Anne	Low	[30..40]	good
Chris	Medium	<30	bad

Etapas: Evaluación del modelo

- Se estima la exactitud del modelo basándose en un conjunto de test
  - Se compara la etiqueta conocida de una muestra de testeo con el resultado de aplicar el modelo de clasificación
  - *Accuracy rate* es el porcentaje de muestras del conjunto de test que son correctamente clasificadas por el modelo
  - El conjunto de test es independiente del conjunto de entrenamiento (método holdout)

Evaluación de Exactitud





Name	Income	Age	Credit rating
Tom	Medium	<30	bad
Jane	High	<30	bad
Wei	High	>40	good
Hua	Medium	[30..40]	good

How accurate is the model?

IF Income = 'High'
OR Age > 30
THEN CreditRating = 'Good'

Etapas: Evaluación del modelo

- Holdout
  - Los datos se particionan aleatoriamente en 2 conjuntos independientes: training set (2/3 de los datos) y test set (1/3 de los datos)
- K-fold cross validation
  - Datos iniciales particionados en k subconjuntos mutuamente excluyentes de aproximadamente igual tamaño. Se hace training y testing k veces, se calcula la exactitud promediando los resultados.
- Stratisfied cross-validation
  - Los subconjuntos son armados de tal manera que la distribución de clase de los ejemplos en cada uno es aproximadamente igual a la que tienen los datos iniciales

Evaluación del modelo

- Tasa/Coeficiente de Error

$$ce(h) = \frac{1}{n} \sum_{i=1}^{n} \|y_i \neq h(x_i)\|$$

- Precisión/Coeficiente de acierto

$$ca(h) = 1 - ce(h)$$

Evaluación del modelo: Matriz de Confusión

Etiqueta de clase	Predicciones C1	Predicciones C2	...	Predicciones Ck
Verdaderos C1	M(C1,C1)	M(C1,C2)	...	M(C1,Ck)
Verdaderos C2	M(C2,C1)	M(C2,C2)	...	M(C2,Ck)
...	...	...	...	...
Verdaderos Ck	M(Ck,C1)	M(Ck,C2)	...	M(Ck,Ck)

Clasificador ideal

- M(Ci, Ci) Casos correctamente clasificados
- M(Ci, Cj) i<>j Errores de clasificación

	C1	C2	...	Ck
C1	M(C1,C1)	0	...	0
C2	0	M(C2,C2)	...	0
...	...	...	...	0
Ck	0	0	...	M(Ck,Ck)

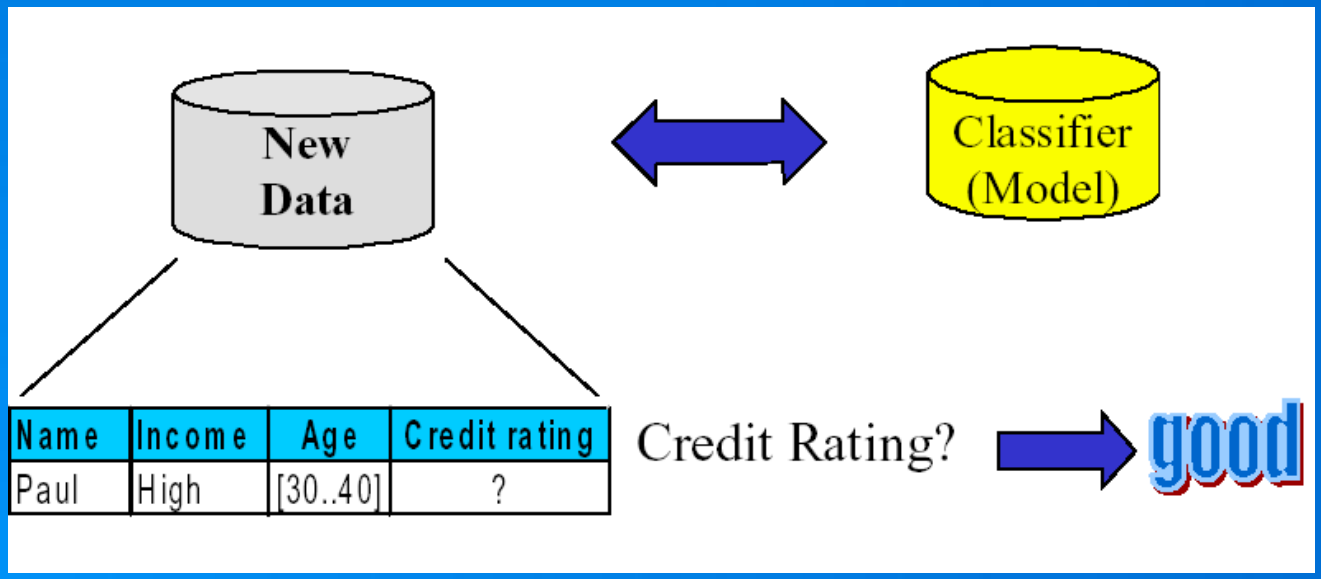
Evaluación del Modelo (Documentos)

- Precision
  - De la cantidad de veces que se predijo una clase, cuántas fueron correctas?

- Recall
  - Se encontraron todos los ejemplos que pertenecen a la clase?

Etapas: Uso del modelo - Clasificación

- El modelo se utiliza para clasificar nuevos objetos
  - Dar una etiqueta de clase a una nueva tupla
  - Predecir el valor de un atributo



Métodos de clasificación

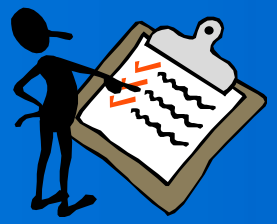
- Inducción de árboles de decisión
- Redes Neuronales
- Clasificador Bayesiano
- Clasificación basada en asociación
- Vecinos más cercanos
- Razonamiento Basado en Casos
- Algoritmos Genéticos
- Conjuntos difusos
- Support Vector Machines
- ...

Evaluación y comparación de métodos de clasificación

- Exactitud de predicción
  - Habilidad del modelo de predecir correctamente la etiqueta de clase de nuevos ejemplos
- Velocidad
  - Tiempo para construir el modelo
  - Tiempo para usar el modelo
- Robustez
  - Manejo de valores faltantes y ruido (predicciones correctas)
- Escalabilidad
  - Eficiencia en grandes bases de datos
- Facilidad de interpretación
  - Nivel de entendimiento provisto por el modelo
- Forma de las reglas
  - Tamaño del árbol de decisión
  - Qué tan compactas son las reglas de clasificación

Clasificación

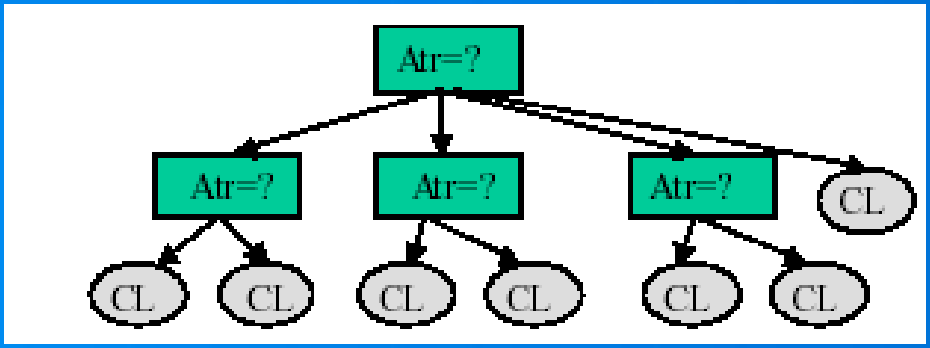
- Concepto
  - Clasificación
  - Predicción
  - Evaluación
- Árboles de Decisión
  - Construcción
  - Uso
  - Poda
- Clasificador Bayesiano
- Ejemplos



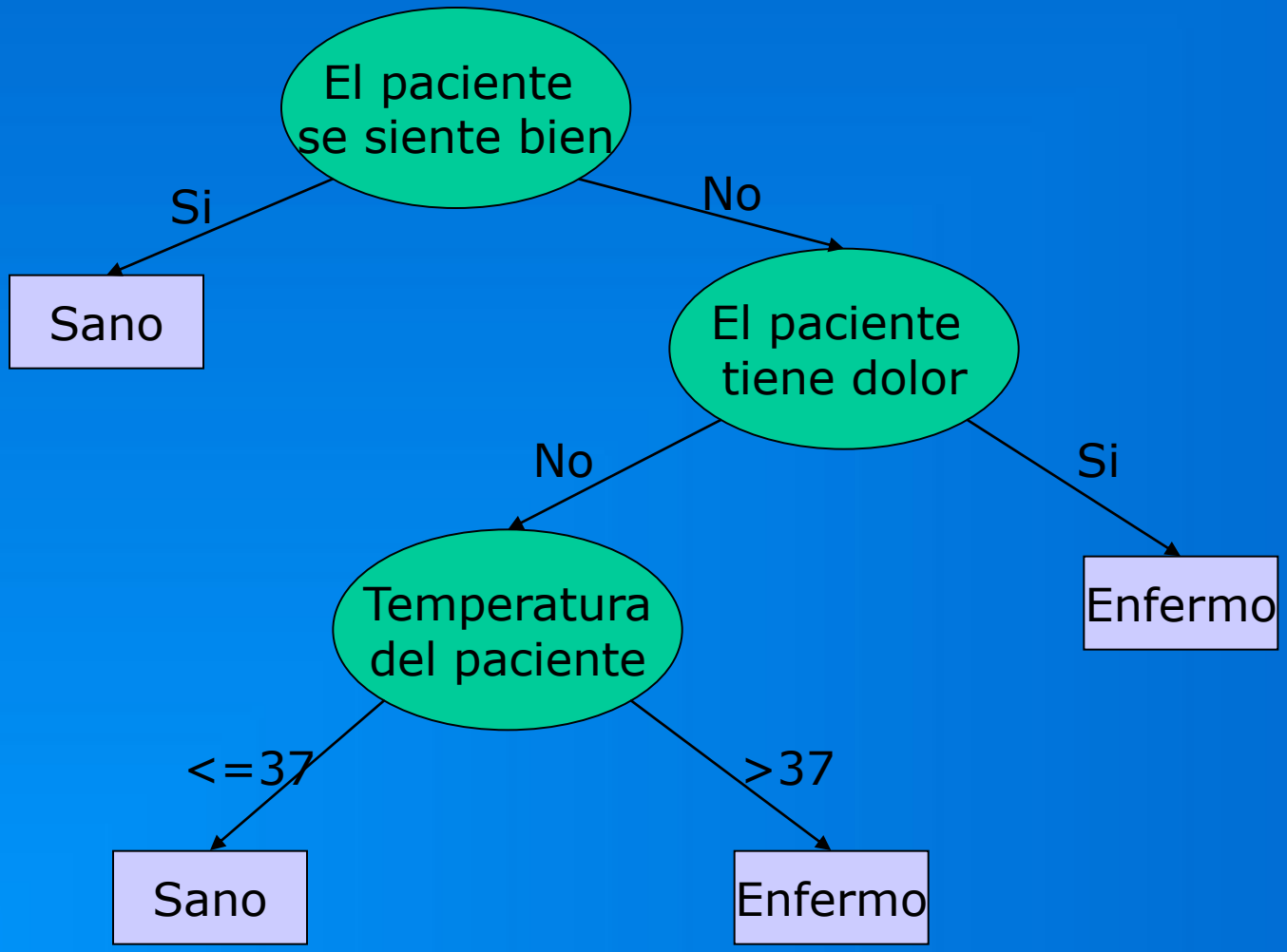
Árboles de Decisión

Un árbol de decisión consta de:

- nodos internos que denotan un test (comprobación) sobre un atributo
- Ramas que representan una salida del test
  - Todas las tuplas en una rama tienen el mismo valor para el atributo evaluado
- Nodos hojas que representan las etiquetas de clase



Ejemplo


El paciente se siente bien
Si
No
Sano
El paciente tiene dolor
No
Si
Temperatura del paciente
Enfermo
<=37
>37
Sano
Enfermo

Equivalente en reglas de clasificación

- **Si** El paciente se siente bien = Si **entonces**
  - Clase = Sano
- **Sino**
  - **Si** El paciente tiene dolor = No **entonces**
    - **Si** Temperatura del paciente <= 37 **entonces**
      - Clase = Sano
    - **Sino (**Temperatura del paciente > 37)
      - Clase = Enfermo
  - **Sino** (El paciente tiene dolor = Si)
    - **Clase = Enfermo**

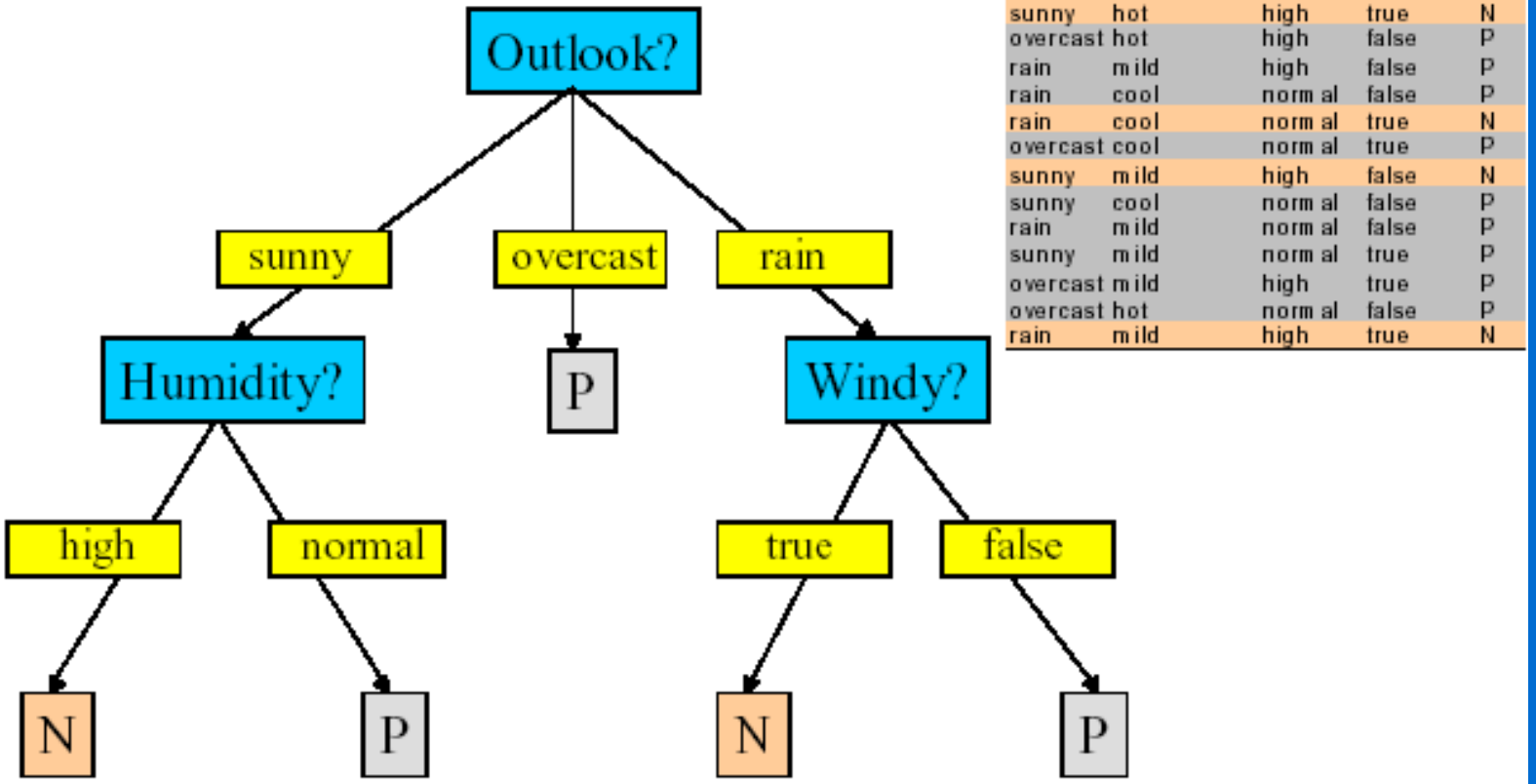
Equivalente en reglas de clasificación

- **Si** El paciente se siente bien = Si **entonces**
  - Clase = Sano
- **Si** El paciente se siente bien = No **and** El paciente tiene dolor = No **and** Temperatura del paciente <=37 **entonces**
  - **Clase = Sano**
- **Si** El paciente se siente bien = No **and** El paciente tiene dolor = No **and** Temperatura del paciente >37 **entonces**
  - **Clase = Enfermo**
- **Si** El paciente se siente bien = No **and** El paciente tiene dolor = Si **entonces**
  - **Clase = Enfermo**

Ejemplo: Datos de entrenamiento

Outlook	Tempreature	Humidity	Windy	Class
sunny	hot	high	false	N
sunny	hot	high	true	N
overcast	hot	high	false	P
rain	mild	high	false	P
rain	cool	normal	false	P
rain	cool	normal	true	N
overcast	cool	normal	true	P
sunny	mild	high	false	N
sunny	cool	normal	false	P
rain	mild	normal	false	P
sunny	mild	normal	true	P
overcast	mild	high	true	P
overcast	hot	normal	false	P
rain	mild	high	true	N

Ejemplo árbol de decisión



Outlook	Tempreature	Humidity	Windy	Class
sunny	hot	high	false	N
sunny	hot	high	true	N
overcast	hot	high	false	P
rain	mild	high	false	P
rain	cool	normal	false	P
rain	cool	normal	true	N
overcast	cool	normal	true	P
sunny	mild	high	false	N
sunny	cool	normal	false	P
rain	mild	normal	false	P
sunny	mild	normal	true	P
overcast	mild	high	true	P
overcast	hot	normal	false	P
rain	mild	high	true	N

Métodos de clasificación con árboles de decisión

- La generación de árbol básica de arriba hacia abajo consiste de dos fases:

1. Construcción del árbol
   - Al inicio todos los ejemplos de entrenamiento están en la raíz
   - La partición de los ejemplos se realiza recursivamente basándose en la selección de atributos
2. Podado del árbol
   - Tiene por objetivo eliminar ramas del árbol que reflejen ruido en los datos de entrenamiento y lleven a errores cuando se clasifiquen los datos de test → mejorar la exactitud de clasificación

¿Cómo construir un árbol?

- Algoritmo
  - Greedy
    - Hacer una elección optimal en cada paso: seleccionar el mejor atributo para cada nodo del árbol
  - Divide y conquista recursivo top-down
    - De la raíz a las hojas
    - Partir un nodo en varias ramas
    - Para cada rama, correr recursivamente el algoritmo

Construcción del árbol de decisión

- El árbol comienza con un solo nodo que representa todos los datos de entrenamiento
- Si los ejemplos tienen todos la misma clase, entonces el nodo se convierte en una hoja con esa etiqueta de clase
- Si no, se selecciona un atributo que mejor separe la muestra en clases individuales. Este atributo se convierte en el atributo de test.
- Se crea una rama por cada valor conocido del atributo de test, y se particionan los ejemplos de acuerdo a estas ramas
- Se aplica el proceso recursivamente, para formar un árbol para los ejemplos en cada partición
- La recursión termina cuando:
  - Los ejemplos en el nodo corresponden a la misma clase
  - No quedan más atributos sobre los cuales separar (hoja con clase mayoritaria)
  - No hay ejemplos con el valor del atributo (para la rama)

Cuestiones principales en la construcción de árboles

- Criterio de separación
  - Usado para seleccionar el atributo para separar en un nodo del árbol durante la fase de generación del árbol
  - Diferentes algoritmos pueden usar distintas funciones: ganancia de información, gini, etc.
- Esquema de ramificado
  - Determinará la rama a la cual pertenece una muestra
  - Separación binaria (gini) versus separación múltiple (ganancia de información)
- Decisiones de terminación
  - Cuando detenerse y dejar de separar un nodo (medida de impureza) - Poda
- Reglas de etiquetado
  - Un nodo es etiquetado como la clase a la cual pertenecen la mayoría de los ejemplos que pertenecen a él

Elección del mejor atributo para clasificar

- Aleatoria
- Menos valores
- Más valores
- Ganancia de información (máxima)
- Índice Gini
- Razón de ganancia
- ...

Algoritmo ID3 (Induction Decision Trees)

- Algoritmo matemático para construir árboles de decisión
- Desarrollado por R. Quinlan en 1979
- Hace uso de Teoría de la Información desarrollada por Shannon
- Construye árbol de arriba hacia abajo
- Usa ganancia de la información para seleccionar atributo para particionar
- ID3, C4.5, J48…

Ganancia de Información

- El atributo que tiene mayor ganancia de información se utiliza para particionar
- Minimizar la información que se necesita para clasificar ejemplos en las particiones resultantes
- Mide qué tan bien separa los ejemplos de entrenamiento de un conjunto dado de acuerdo a su clasificación objetivo (impurezas)
- Minimiza el número esperado de tests que se necesitan para clasificar un objeto y garantiza la construcción de un árbol simple.

Entropía

• Medida del grado de incertidumbre asociado a una distribución de probabilidad.

• En una distribución uniforme, todos los valores son igualmente probables Pi = 1/N y por tanto la entropía es máxima, lo cual indica máxima incertidumbre.

•Por el contrario, en una distribución pico en la que Pi = 1 y Pj=0, para todo j≠i la entropía es mínima lo cual indica mínima incertidumbre o sea máxima información.

Entropía: mínimo teórico de bits promedio necesarios para trasmitir un conjunto de mensajes sobre {x1,x2,…,xn} con distribución de probabilidades P(X=xi)=pi

Entropy(P) = -(p1*log(p1) + p2*log(p2) + .. + pn*log(pn))

Entropía

- Supongamos que hay 2 clases, P y N
  - Sea el conjunto de ejemplos S que contiene x elementos de la clase P e y elementos de la clase N
  - La cantidad de información que se necesita para decidir si un ejemplo arbitrario en S pertenece a P o N se define como

$$I(S_P, S_N) = -\frac{x}{x+y}\log_2\frac{x}{x+y} - \frac{y}{x+y}\log_2\frac{y}{x+y}$$

En gral.

$$I(s_1, s_2, \ldots, s_n) = -\sum_{i=1}^{n} p_i \log_2(p_i)$$

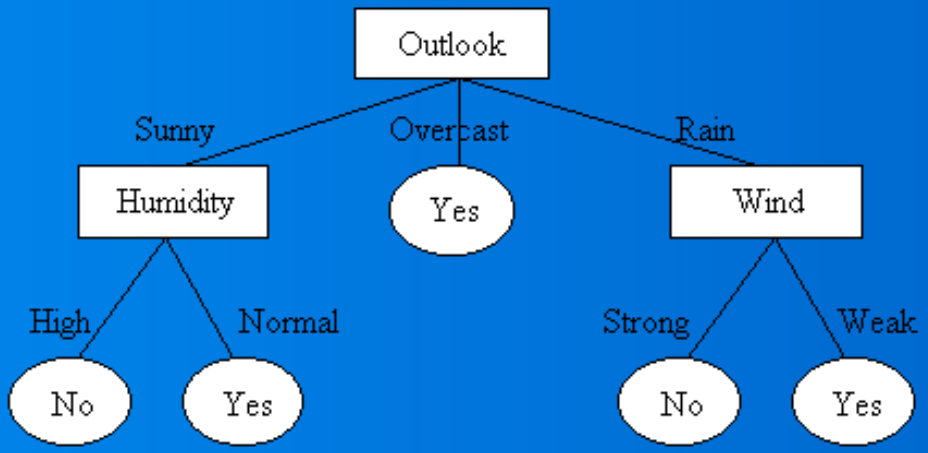
Ejemplo

$$E(S) = \sum_{i=1}^{c} - p_i \log_2 p_i$$

Play Golf	
Yes	No
9	5

Entropy(PlayGolf) = Entropy (5,9)
= Entropy (0.36, 0.64)
= - (0.36 $\log_2$ 0.36) - (0.64 $\log_2$ 0.64)
= 0.94

Atributos				Objetivo
Outlook	Temp	Humidity	Windy	Play Golf
Rainy	Hot	High	False	No
Rainy	Hot	High	True	No
Overcast	Hot	High	False	Yes
Sunny	Mild	High	False	Yes
Sunny	Cool	Normal	False	Yes
Sunny	Cool	Normal	True	No
Overcast	Cool	Normal	True	Yes
Rainy	Mild	High	False	No
Rainy	Cool	Normal	False	Yes
Sunny	Mild	Normal	False	Yes
Rainy	Mild	Normal	True	Yes
Overcast	Mild	High	True	Yes
Overcast	Hot	Normal	False	Yes
Sunny	Mild	High	True	No



Ganancia de información

• Criterio para elegir un split: seleccionar el split con la mayor ganancia de información (Gain)

• Equivalente a seleccionar el split con la menor entropía ponderada

• Medida de cuanto ayuda el conocer el valor de una variable aleatoria X para conocer el verdadero valor de otra Y.

• En nuestro caso, X es un atributo de un ejemplo dado mientras que Y es la clase a la que pertenece el ejemplo.

• Una alta ganancia implica que el atributo X permite reducir la incertidumbre de la clasificación del ejemplo de entrada.

Ganancia de Información: Ejemplo

- El atributo A es seleccionado tal que la ganancia de información

$$\mathbf{Gain(A)=I(S_P, S_N) – E(A)}$$

sea maximal, es decir, E(A) es minimal dado que $I(S_P, S_N)$ es la misma para todos los atributos del nodo (reducción esperada en la entropía causada por conocer el valor del atributo A)

- En el ejemplo dado, el atributo outlook es elegido para separar la raíz:

Gain(outlook) = 0.247

Gain (temperature) = 0.029

Gain (humidity) = 0.151

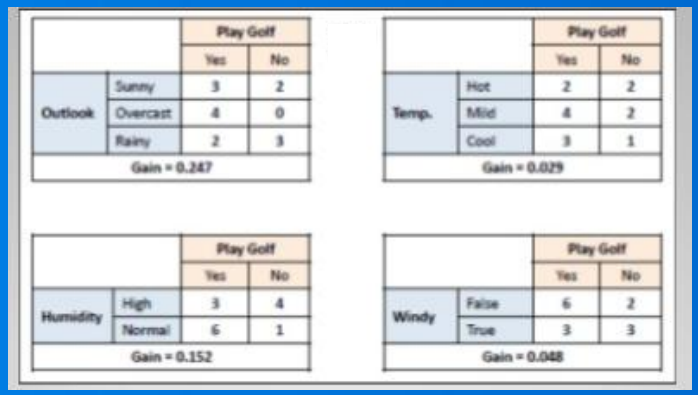
Gain(windy) = 0.048

Ejemplo

E(PlayGolf, Outlook)
=p(sunny)*E(3,2)+p(overcast)*E(4,0)+p(rainy)*E(3,2)=
(5/14)*0.971+(9/14)*0.971 = 0.693

G(PlayGolf, Outlook) = E(PlayGolf) - E(PlayGolf, Outllok) =
0.940 – 0.693 = 0.247

Lo mismo se hace para los demás atributos: temperature, humidity y windy.



		Play Golf	
		Yes	No
Outlook	Sunny	3	2
	Overcast	4	0
	Rainy	2	3
Gain = 0.247			

		Play Golf	
		Yes	No
Temp.	Hot	2	2
	Mild	4	2
	Cool	3	1
Gain = 0.029			

		Play Golf	
		Yes	No
Humidity	High	3	4
	Normal	6	1
Gain = 0.152			

		Play Golf	
		Yes	No
Windy	False	6	2
	True	3	3
Gain = 0.048			

Algoritmo ID3

- Se calcula la entropía del nodo N

$$H(E_i)=p^{+}\log_2(1/p^{+})+p^{-}\log_2(1/p^{-})$$

- Se calcula el valor medio de la entropía del nivel siguiente, generado por el atributo X al que se está aplicando el test

$$H(X, E_i)=p(X=v_1/E_i)H(E_{i1})+\ldots+$$
$$p(X=v_n/E_i)H(E_{in})$$

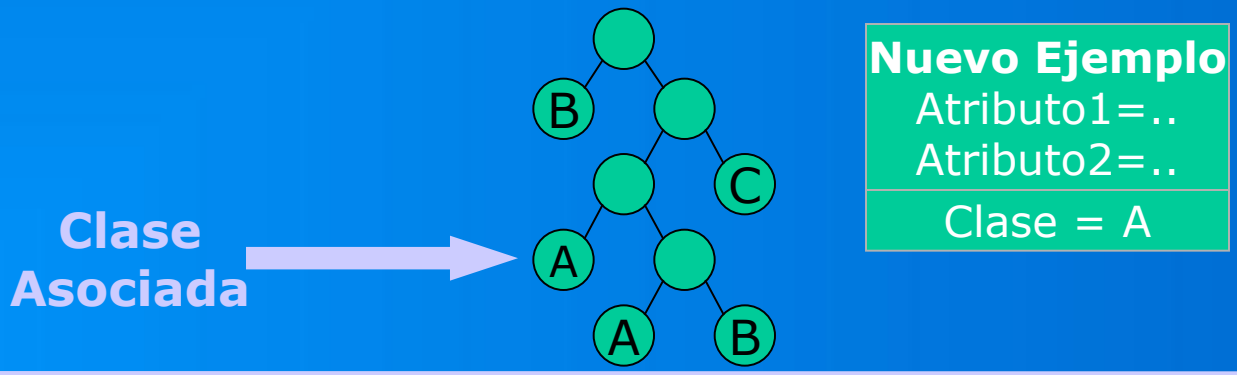
Utilización del árbol

- Directamente
  - Verificar el valor de un atributo de un ejemplo no conocido con el árbol
  - Se sigue el camino desde la raíz a la hoja que posea la etiqueta

- Indirectamente
  - El árbol de decisión se convierte en reglas de clasificación
  - Se crea una regla por cada camino de la raíz a las hojas
  - Las reglas IF-THEN son más fáciles de entender

Clasificación de nuevos ejemplos

- Partir desde la raíz
- Avanzar por los nodos de decisión hasta alcanzar una hoja
- La clase del nuevo ejemplo es la clase que representa la hoja.



Overfitting

Un árbol generado puede sobre-clasificar (super-ajustar) los ejemplos de entrenamiento debido a ruido o tamaño pequeño del conjunto de entrenamiento. Resulta en baja exactitud de clasificación de nuevos ejemplos.

- Existen 2 enfoques para evitar overfitting
  - Parar antes (Pre-poda): detener el crecimiento del árbol antes que se produzca, decidiendo no particionar uno o más nodos
  - Podado: se construye todo el árbol, se permite overfitting y luego se poda el árbol. Se elige el árbol con menor tasa de error.

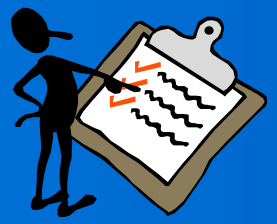
Podado del árbol

Podar el árbol: transformar un subárbol en una hoja

- Usar un conjunto de datos diferentes del conjunto de entrenamiento (conjunto de poda)
- En un nodo del árbol, si la exactitud sin particionar el nodo es más alta que la exactitud particionando el nodo, reemplazar el subárbol por una hoja, etiquetándolo con la clase mayoritaria

Clasificación

- Concepto
  - Clasificación
  - Predicción
  - Evaluación
- Árboles de Decisión
  - Construcción
  - Uso
  - Poda
- Clasificador Bayesiano
- Ejemplos



Clasificador Bayesiano

- Es un clasificador estadístico basado en el Teorema de Bayes
- Usa aprendizaje probabilístico para calcular explícitamente las probabilidades de las hipótesis
- Un clasificador bayesiano simple o naive asume independencia total entre atributos
- Funciona bien con grandes conjuntos de datos y tiene alta precisión de clasificación
- El modelo es incremental es el sentido que cada ejemplo de entrenamiento puede aumentar o disminuir la probabilidad de que una hipótesis sea correcta. Conocimiento previo puede combinarse con datos observados

Clasificador Bayesiano

El objetivo de un clasificador es identificar a qué clase pertenece un objeto, basándose en ciertos atributos.

Consideremos un objeto X, cuyos atributos son (a_1,a_2...a_n), y queremos clasificarlo dentro de un conjunto de clases S.

Sea A una clase, uno de los elementos de S. Entonces, trataremos de encontrar un A que maximice P(A| a_1,a_2...a_n). Podemos decir entonces:

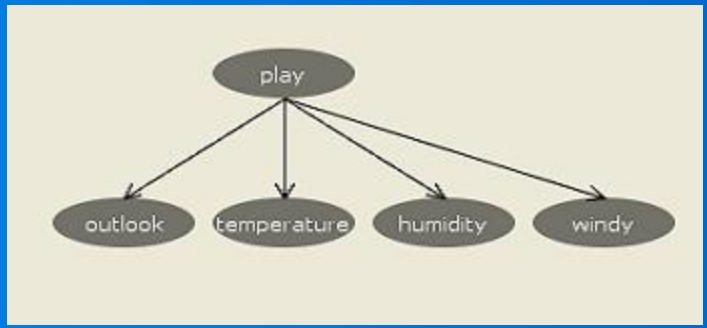
"Es muy probable que X pertenezca a la clase A porque para un objeto, dada la condición de tener los atributos (a_1,a_2...a_n), la probabilidad de que la clase sea A es máxima."

Teorema de Bayes

Dado un objeto X (descripción del clima) con etiqueta de clase desconocida, H es la hipótesis de que X pertenece a una clase C (apto para jugar golf)

La probabilidad a posteriori de la hipótesis H, P(H/X), probabilidad condicional de H dado X, sigue el teorema de Bayes

$$P(H/X) = \frac{P(X/H)\,P(H)}{P(X)}$$



P(H/X) probabilidad a posteriori

P(H), P(X) probabilidad a priori (datos)

P(X/H) probabilidad a posteriori (que el clima esté de cierta forma dado que juego golf)

Clasificador Naive Bayes

- Supongamos que tenemos n clases C_1, C_2,..., C_n. Dado un ejemplo desconocido A, el clasificador predecirá que $A=(a_1,...,a_m)$ pertenece a la clase con la mayor probabilidad a posteriori:

$A \in C_i$ si $P(C_i/A) > P(C_j/A)$ para $1 \leq j \leq n$, $j \neq i$

Maximizar $P(A/C_i)\ P(C_i) / P(A)$ → Maximizar $P(A/C_i)\ P(C_i)$

$P(C_i) = s_i/s$

$P(A/C_i) = P(a_1,...,a_m /C_i)$

Suposición de Naive Bayes

$$P(a_1, a_2, \ldots a_n | A) = \prod_{i=1}^{n} P(a_i | A)$$

Siempre que $(a_1, a_2 \ldots a_n)$ sean condicionalmente independientes unas de otras.

Dado que $P(a_i|A)$ se obtiene fácilmente de los datos de entrenamiento, el clasificador tiene que encontrar una clase A que maximice la expresión:

$$P(A) \prod_{i=1}^{n} P(a_i | A)$$

Ejemplo

Tamaño	Precio	Peso	Color	Compro?
Grande	Barato	Pesado	azul	Si
Grande	Caro	Medio	Azul	No
Chico	Caro	Medio	Verde	Si
Chico	Barato	Liviano	Azul	No
Chico	Caro	pesado	Verde	No

(chico, barato, pesado, azul) compro?

Ejemplo

P(C = Si | T = Chico, Pr = Barato, P = Pesado, C = Azul) y

P(C = No |T = Chico, Pr = Barato, P = Pesado, C = Azul)

Equivalente a:

(1): P(C = S) P(T = Chico | C = S) P(Pr = Barato|C = S) P(P = Pesado | C = S) P(C = Azul|C = S)

(2): P(C = N) P(T = Chico | C = N) P(Pr = Barato|C = N) P(P = Pesado | C = N) P(C = Azul|C = N)

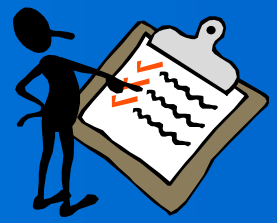
Resultados:

(1): (2/5)(1/2)(1/2)(1/2)(1/2) = 0.025

(2): (3/5)(2/3)(1/2)(1/3)(2/3) = 0.044

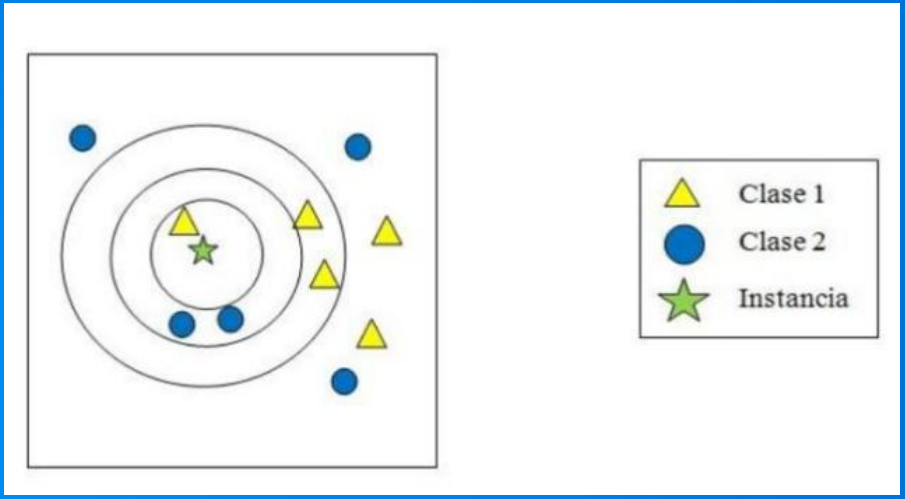
Clasificación

- Concepto
  - Clasificación
  - Predicción
  - Evaluación
- Árboles de Decisión
  - Construcción
  - Uso
  - Poda
- Clasificador Bayesiano
- Vecinos más cercanos



Vecinos más cercanos

Para K=1 (círculo más pequeño), la clase de la nueva instancia sería la Clase 1, ya que es la clase de su vecino más cercano, mientras que para K=3 la clase de la nueva instancia sería la Clase 2 pues habrían dos vecinos de la Clase 2 y solo 1 de la Clase 1



Algoritmo

COMIENZO

Entrada: $D = \{(\mathbf{x}_1, c_1), \ldots, (\mathbf{x}_N, c_N)\}$

$\mathbf{x} = (x_1, \ldots, x_n)$ nuevo caso a clasificar

PARA todo objeto ya clasificado (x_i, c_i)

calcular $d_i = d(\mathbf{x}_i, \mathbf{x})$

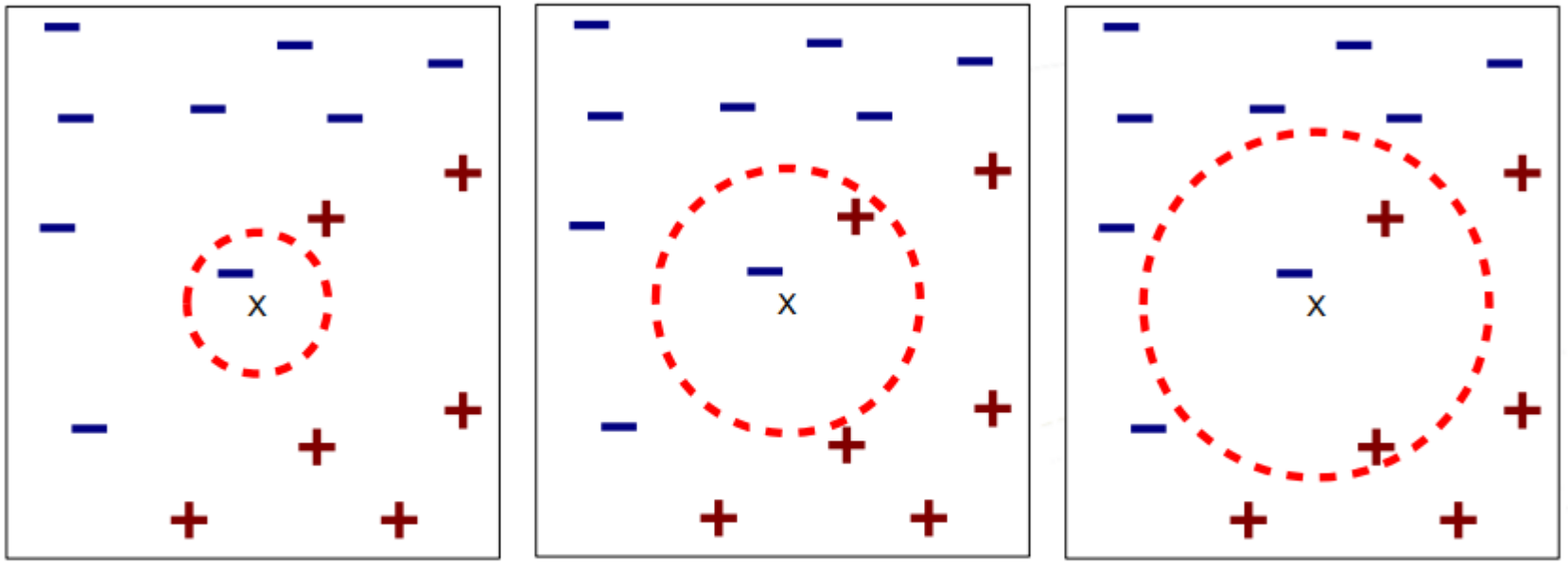
Ordenar $d_i (i = 1, \ldots, N)$ en orden ascendente

Quedarnos con los K casos $D_{\mathbf{x}}^K$ ya clasificados más cercanos a $\mathbf{x}$

Asignar a $\mathbf{x}$ la clase más frecuente en $D_{\mathbf{x}}^K$

FIN

Cómo elegir K?



(a) 1-nearest neighbor (b) 2-nearest neighbor (c) 3-nearest neighbor

Escogiendo el valor de K:

- Si K es muy pequeño el modelo será muy sensitivo a puntos que son atípicos o que son ruido (datos corruptos, outliers)
- Si K es muy grande, el modelo tiende a asignar siempre a la clase más grande.

Cómo elegir K?

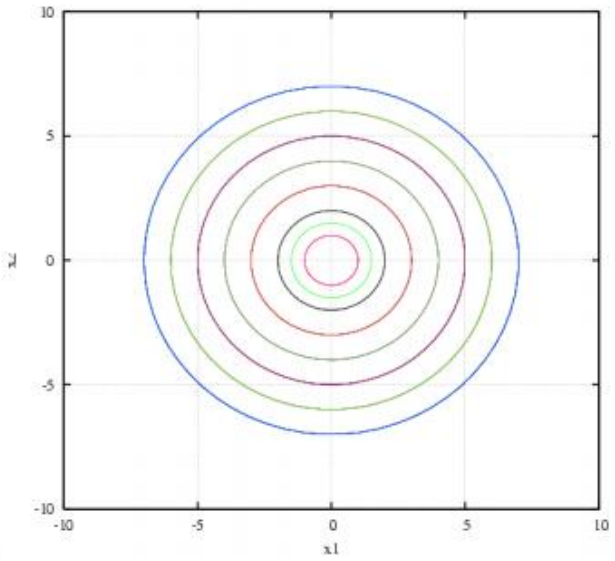
En algunos algoritmos, el modelo escogerá el valor de K que mejor clasificación logre en esta tabla, es decir, prueba con K=1, K=2, ….

Esto puede ser muy caro computacionalmente.

Cómo elegir la distancia?

$$d(A,B) \equiv \sqrt{\sum_{i=1}^{n} (A_i - B_i)^2} = \sqrt{(A-B)^T (A-B)}$$

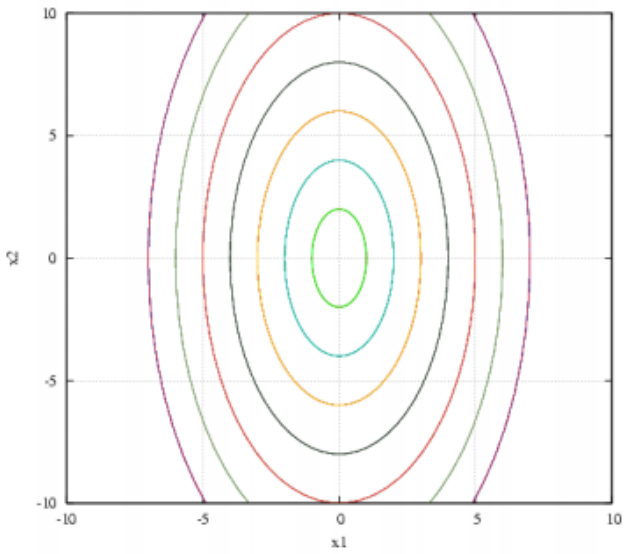
Distancia Euclídea



Cómo elegir la distancia?

$$d(A,B) \equiv \sqrt{(A-B)^T M^T M (A-B)}$$

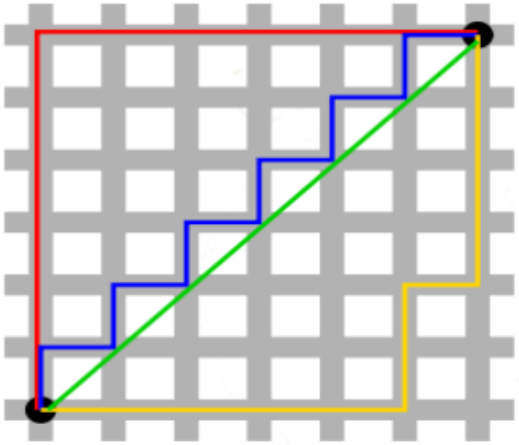
Distancia Euclídea Ponderada



Cómo elegir la distancia?

$$d(x, y) = \sum_{i=1}^{n} |x_i - y_i|$$

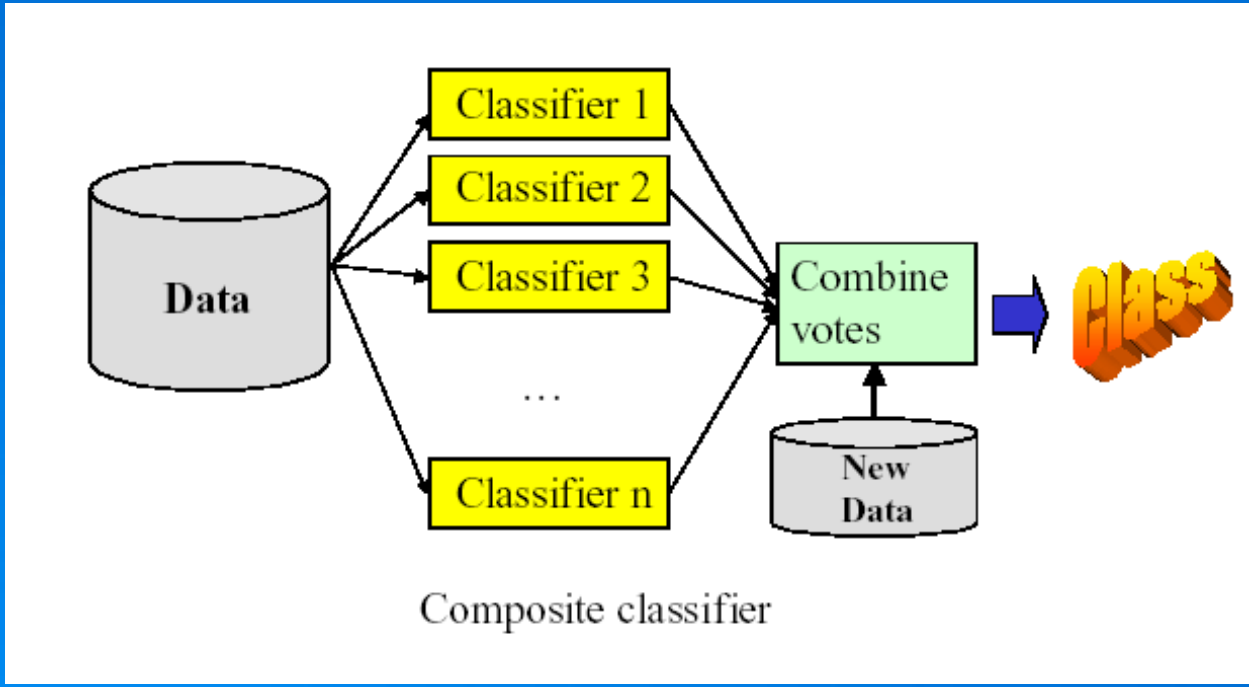
Distancia Manhattan (city-block)



Áreas de investigación en Arboles de Decisión

- Clases desbalanceadas
- Clasificadores multi-clase
- Aprendizaje semi-supervisado
- Combinación de clasificadores
- Selección de atributos
- …

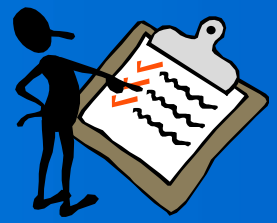
Mejorar la exactitud: Clasificadores compuestos



- Bagging: ej. consulto varios doctores y me quedo con la opinión mayoritaria (la que tenga más votos)
- Boosting: ej. pondero cada diagnóstico según la exactitud del médico (del clasificador)

Clasificación

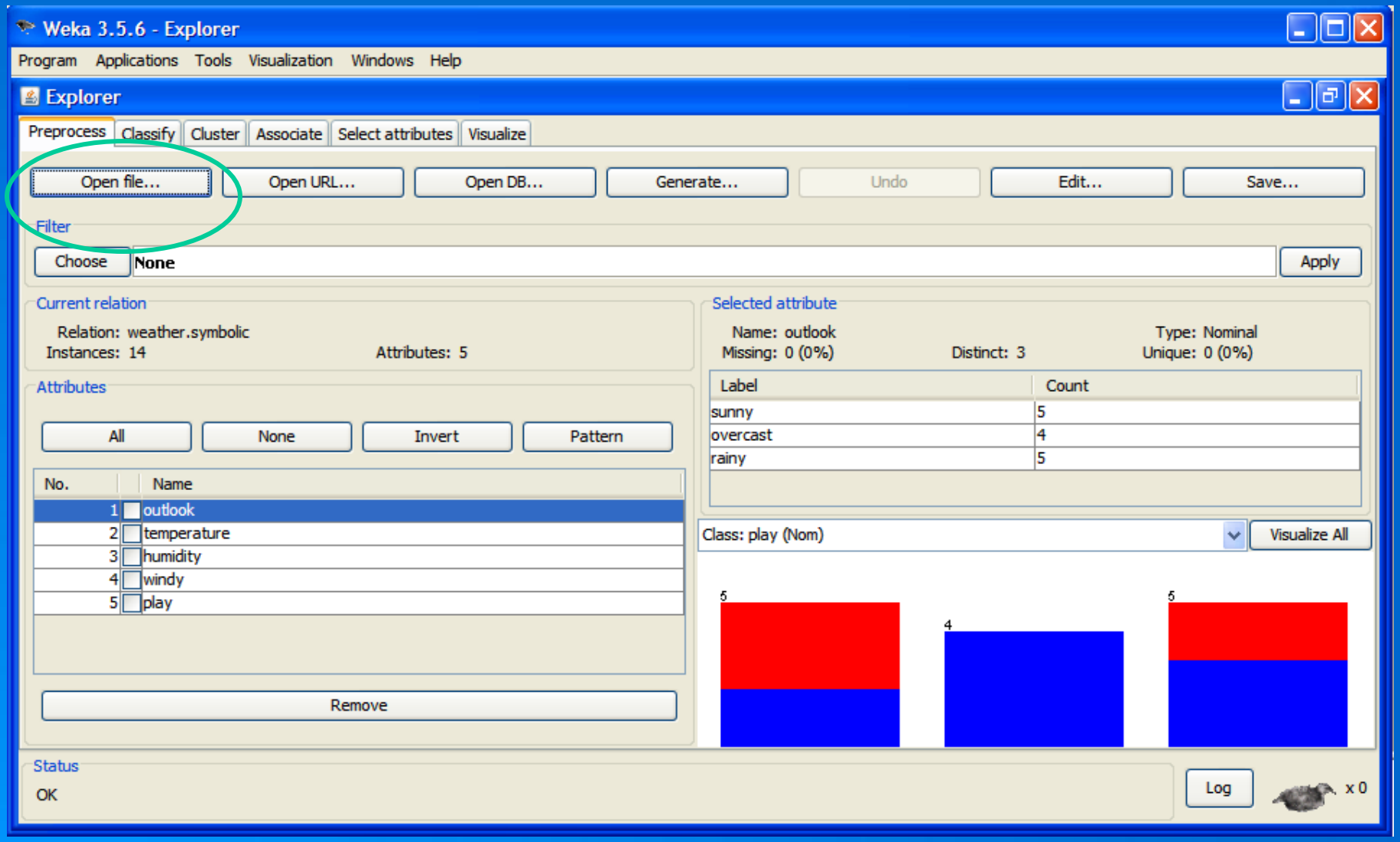
- Concepto
  - Clasificación
  - Predicción
  - Evaluación
- Árboles de Decisión
  - Construcción
  - Uso
  - Poda
- Clasificador Bayesiano
- Herramientas: Weka



Software: WEKA

- Machine Learning Software in Java
  http://www.cs.waikato.ac.nz/~ml/weka/
- Data Mining: Practical Machine Learning Tools and Techniques - Ian H. Witten, Eibe Frank (Third Edition) – Morgan Kaufmann - 2011

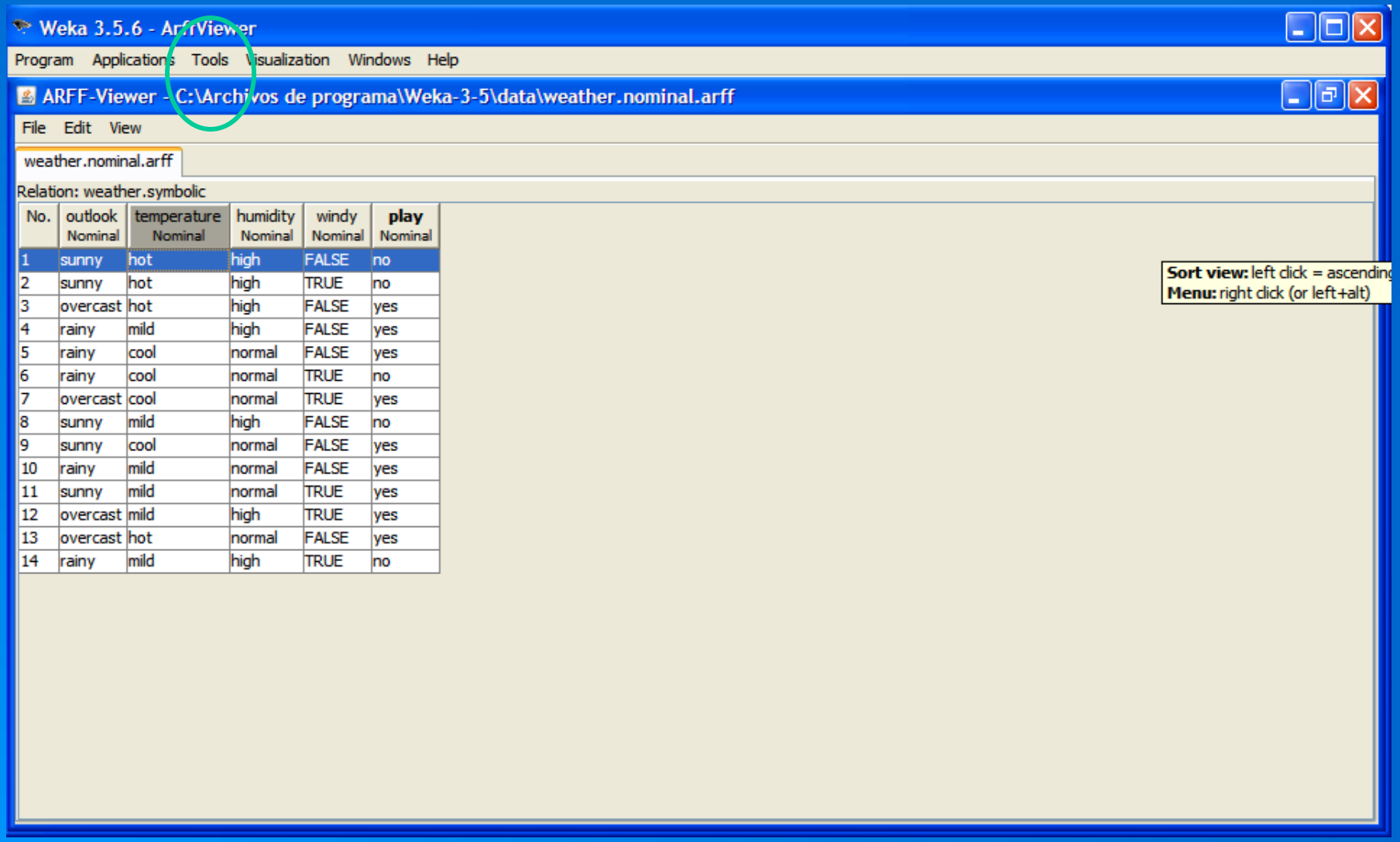
WEKA


Weka 3.5.6 - Explorer
Program Applications Tools Visualization Windows Help
Explorer
Preprocess Classify Cluster Associate Select attributes Visualize
Open file... Open URL... Open DB... Generate... Undo Edit... Save...
Filter
Choose None Apply
Current relation
Relation: weather.symbolic
Instances: 14 Attributes: 5
Selected attribute
Name: outlook Type: Nominal
Missing: 0 (0%) Distinct: 3 Unique: 0 (0%)
Label Count
sunny 5
overcast 4
rainy 5
Attributes
All None Invert Pattern
No. Name
1 outlook
2 temperature
3 humidity
4 windy
5 play
Remove
Class: play (Nom)
Visualize All
5
4
5
Status
OK
Log
x 0

WEKA: Input file

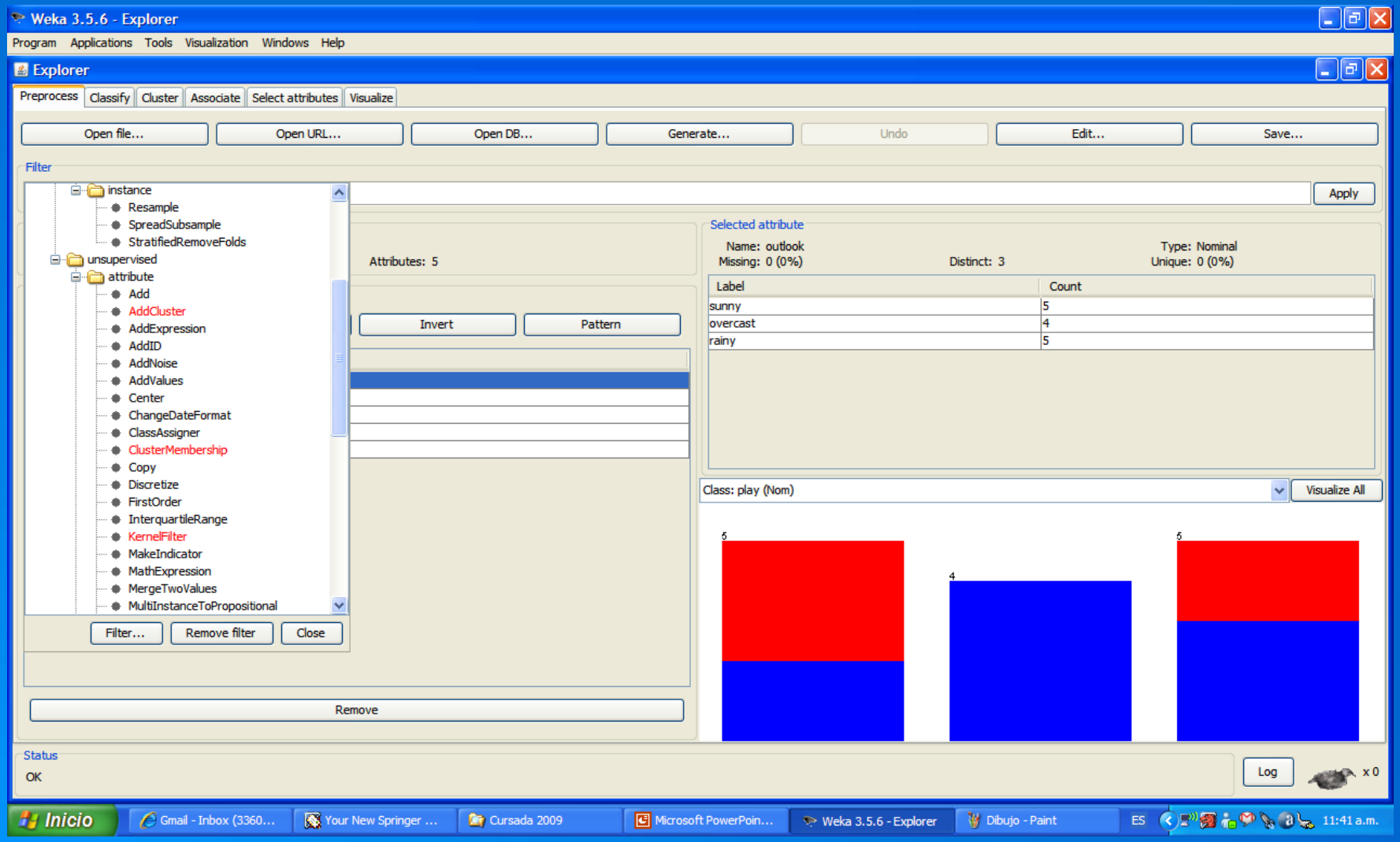
```
@relation weather.symbolic
@attribute outlook {sunny, overcast, rainy}
@attribute temperature {hot, mild, cool}
@attribute humidity {high, normal}
@attribute windy {TRUE, FALSE}
@attribute play {yes, no}

@data
sunny,hot,high,FALSE,no
sunny,hot,high,TRUE,no
overcast,hot,high,FALSE,yes
rainy,mild,high,FALSE,yes
rainy,cool,normal,FALSE,yes
rainy,cool,normal,TRUE,no
overcast,cool,normal,TRUE,yes
sunny,mild,high,FALSE,no
sunny,cool,normal,FALSE,yes
rainy,mild,normal,FALSE,yes
sunny,mild,normal,TRUE,yes
overcast,mild,high,TRUE,yes
overcast,hot,normal,FALSE,yes
rainy,mild,high,TRUE,no
```

Weka: Arff viewer

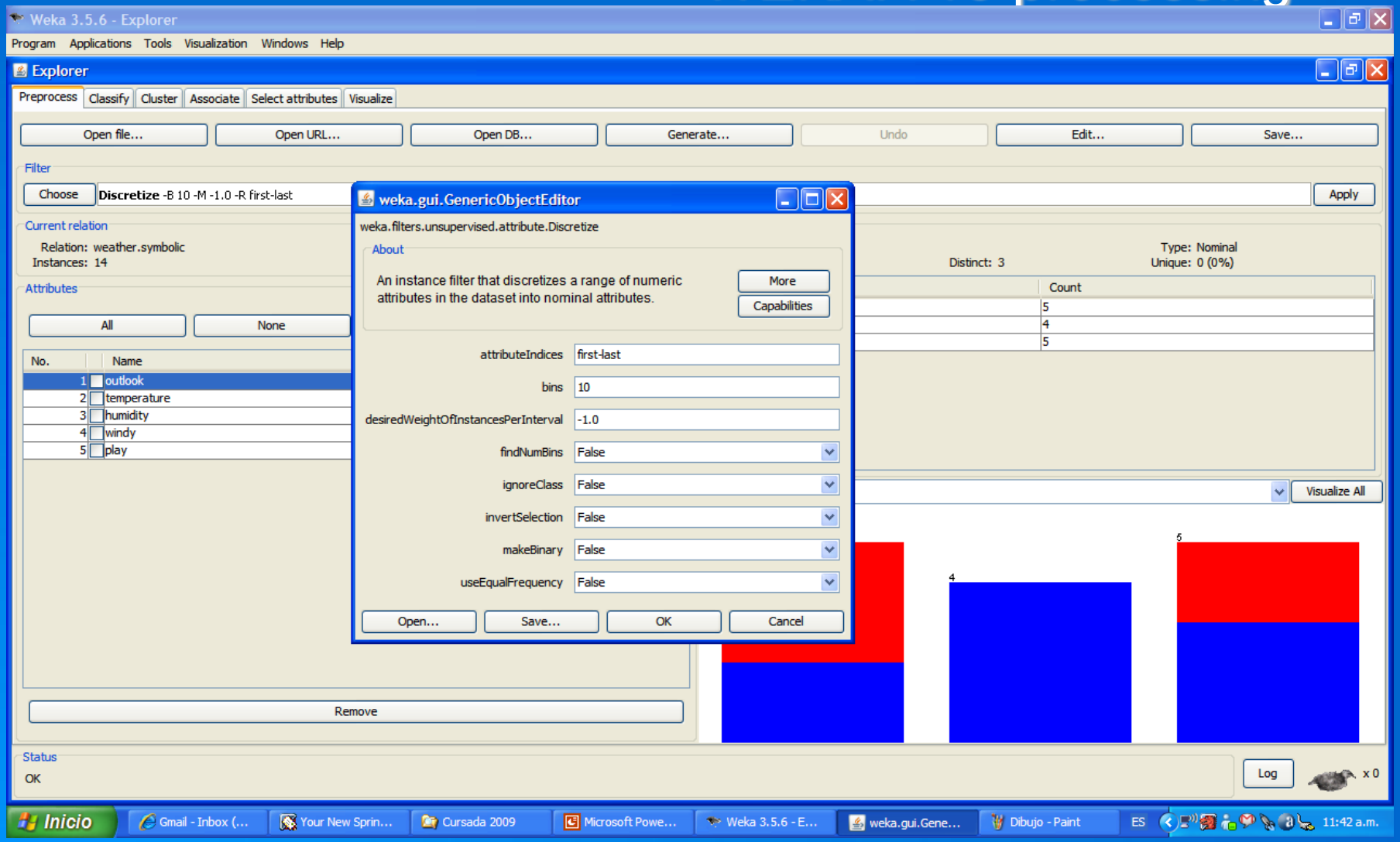


No.	outlook Nominal	temperature Nominal	humidity Nominal	windy Nominal	play Nominal
1	sunny	hot	high	FALSE	no
2	sunny	hot	high	TRUE	no
3	overcast	hot	high	FALSE	yes
4	rainy	mild	high	FALSE	yes
5	rainy	cool	normal	FALSE	yes
6	rainy	cool	normal	TRUE	no
7	overcast	cool	normal	TRUE	yes
8	sunny	mild	high	FALSE	no
9	sunny	cool	normal	FALSE	yes
10	rainy	mild	normal	FALSE	yes
11	sunny	mild	normal	TRUE	yes
12	overcast	mild	high	TRUE	yes
13	overcast	hot	normal	FALSE	yes
14	rainy	mild	high	TRUE	no

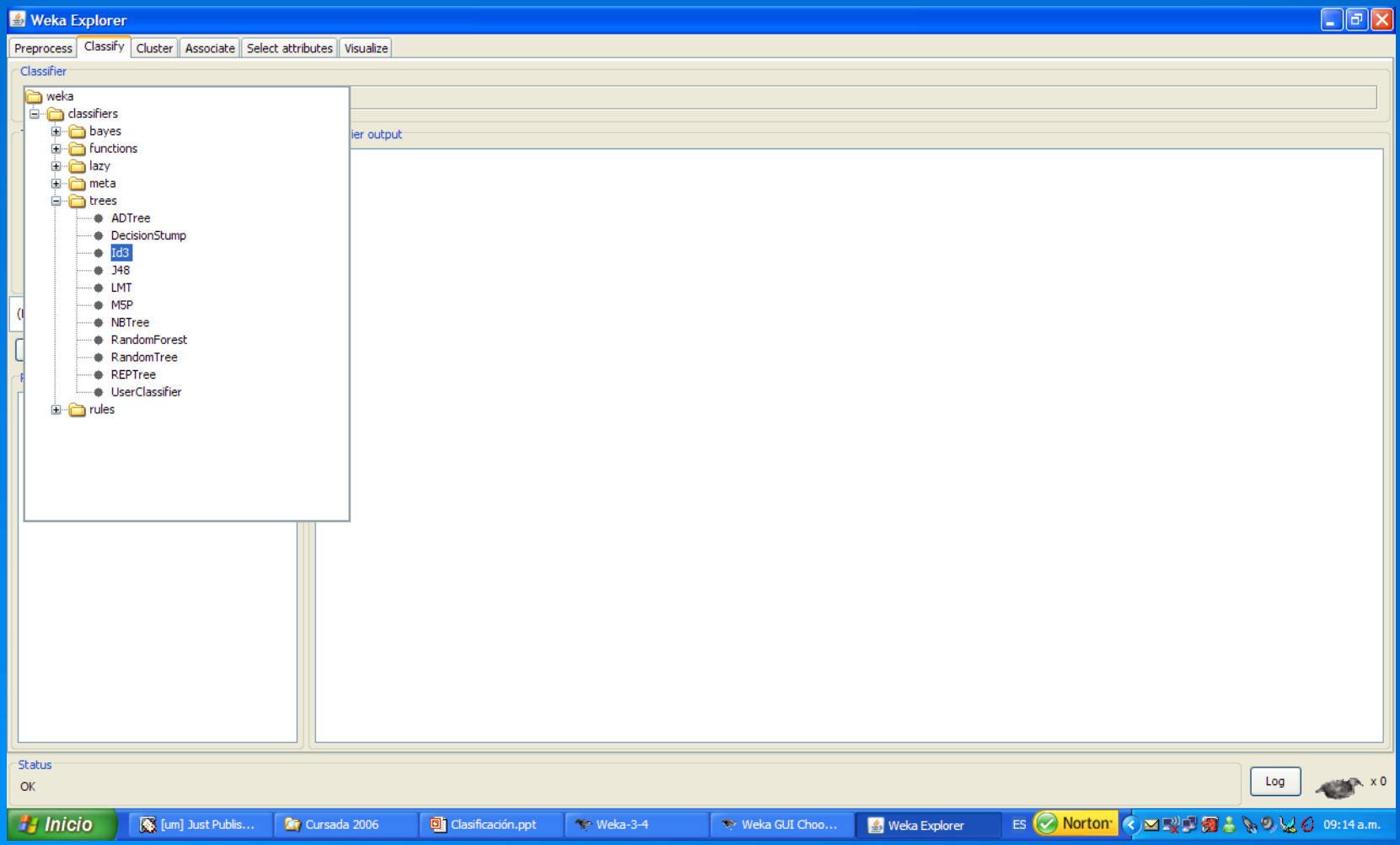
WEKA: Pre-processing



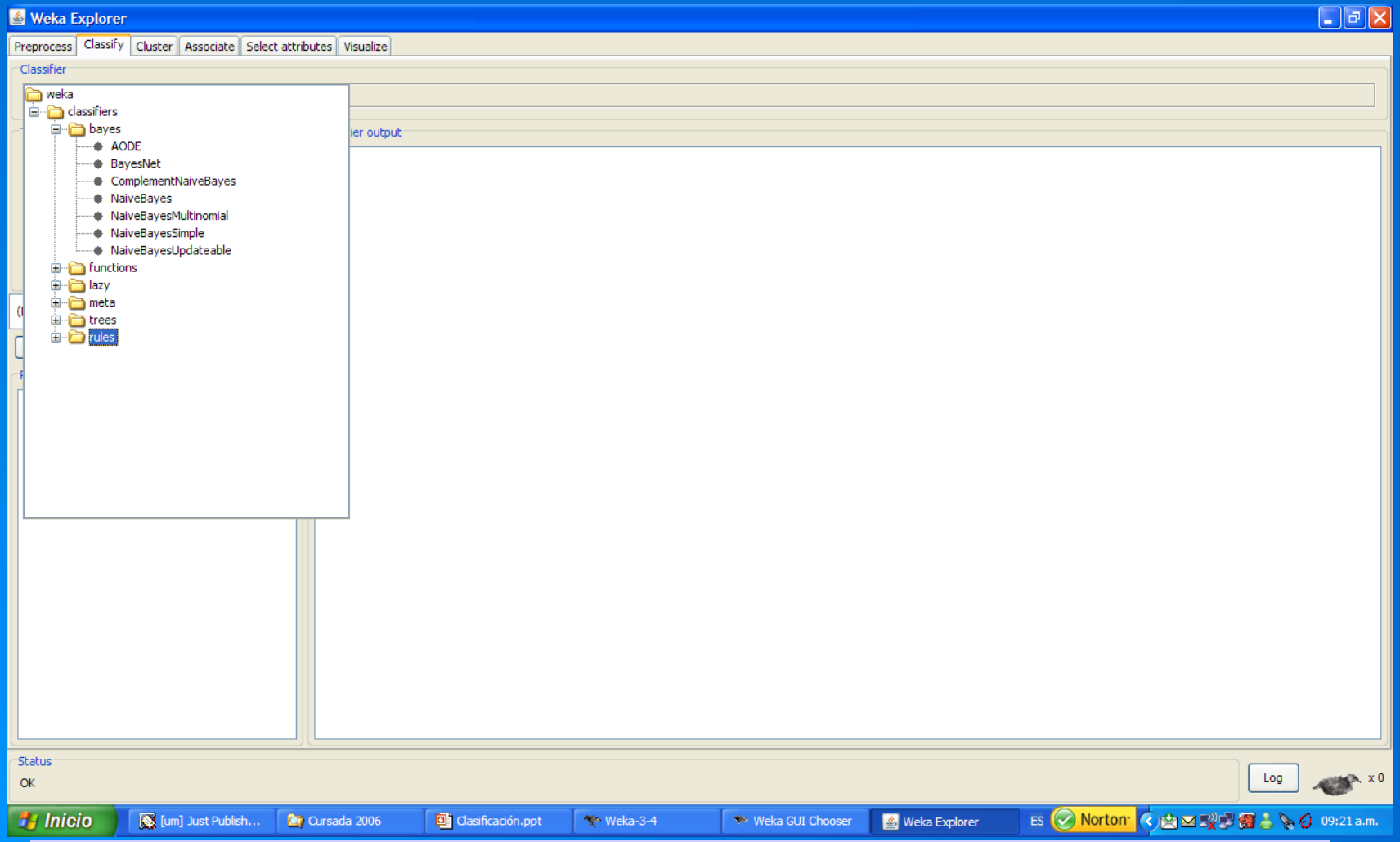
WEKA: Pre-processing



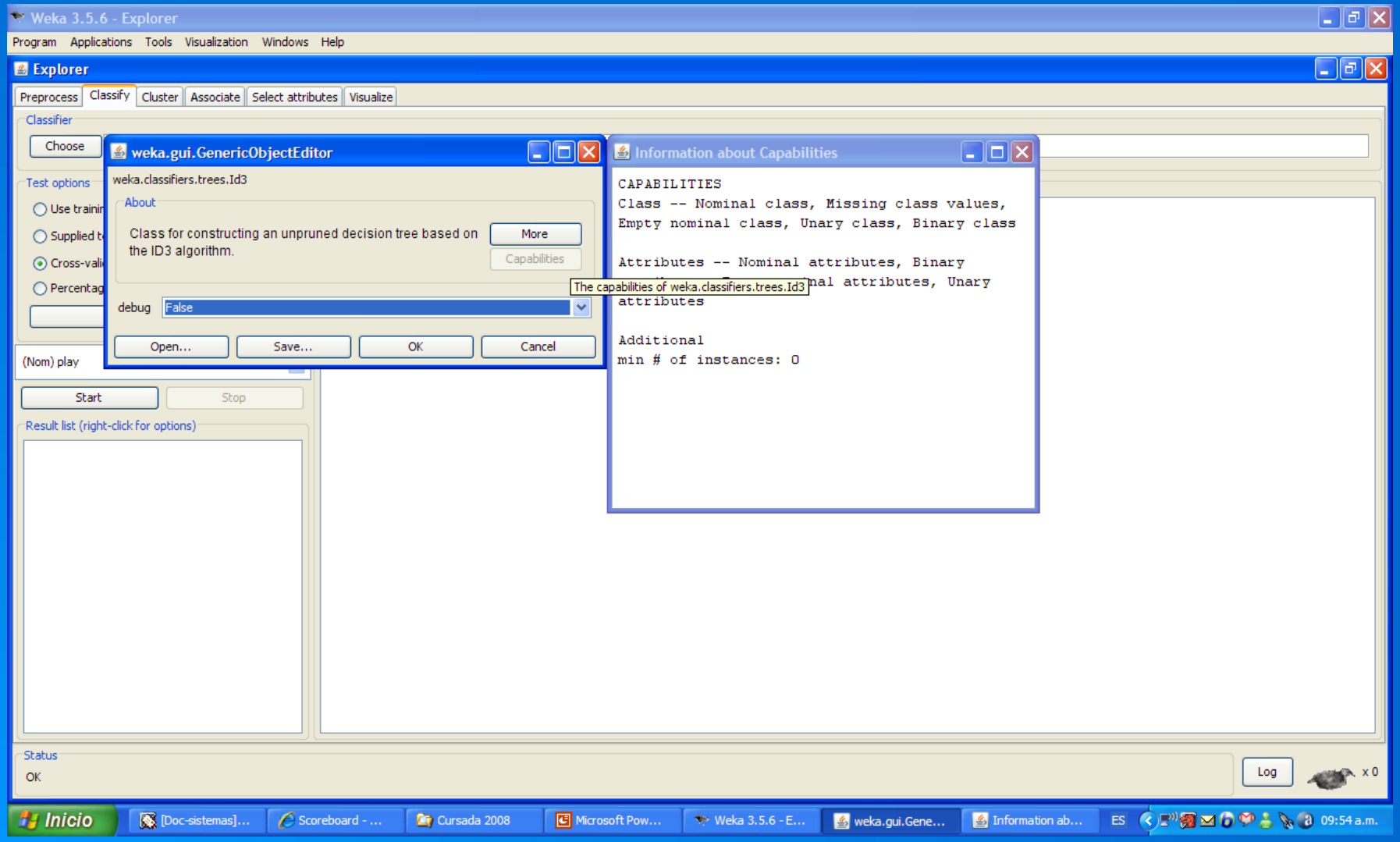
WEKA: Clasificadores



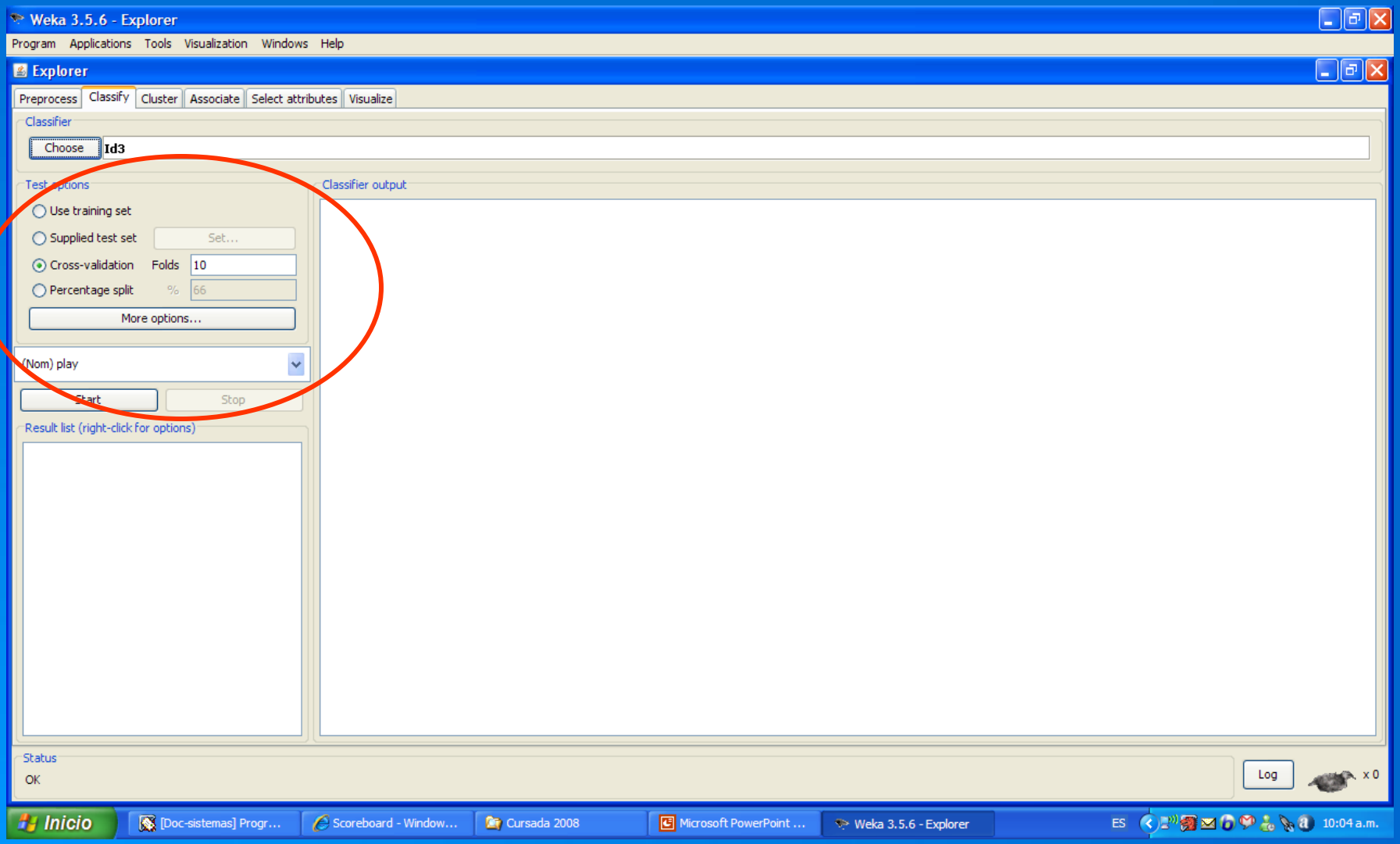
WEKA: Clasificadores



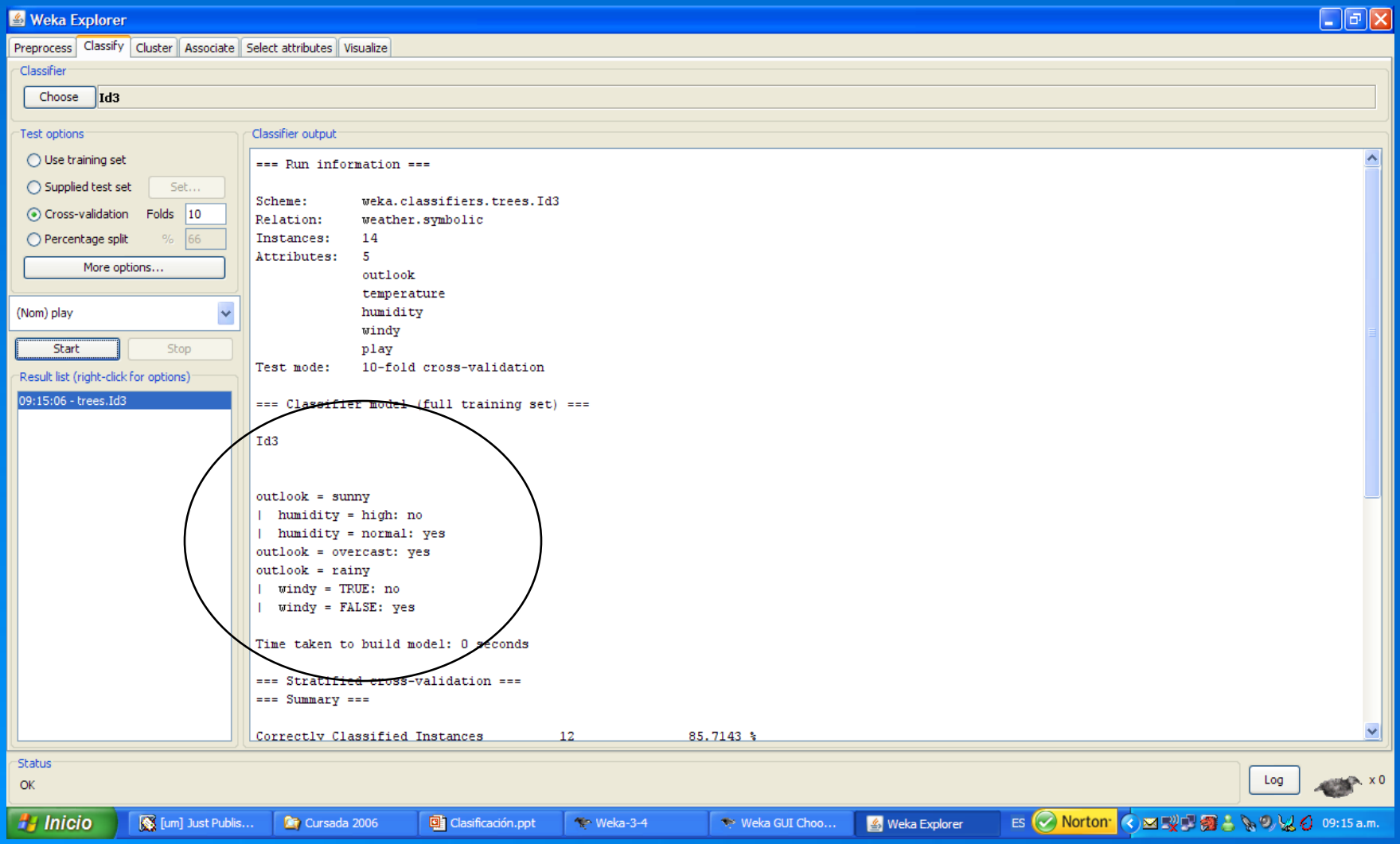
WEKA: Parámetros



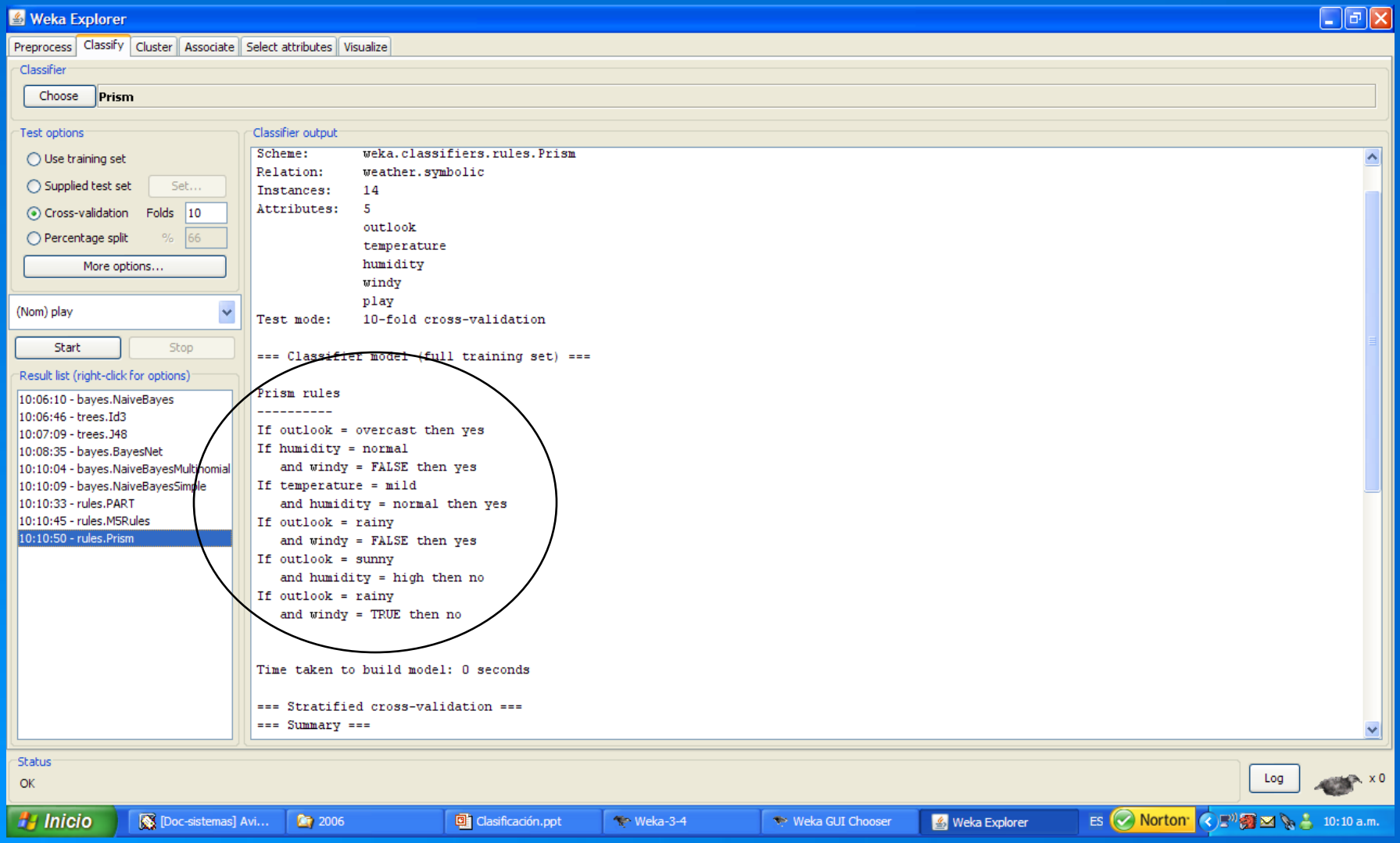
WEKA: Training y Test



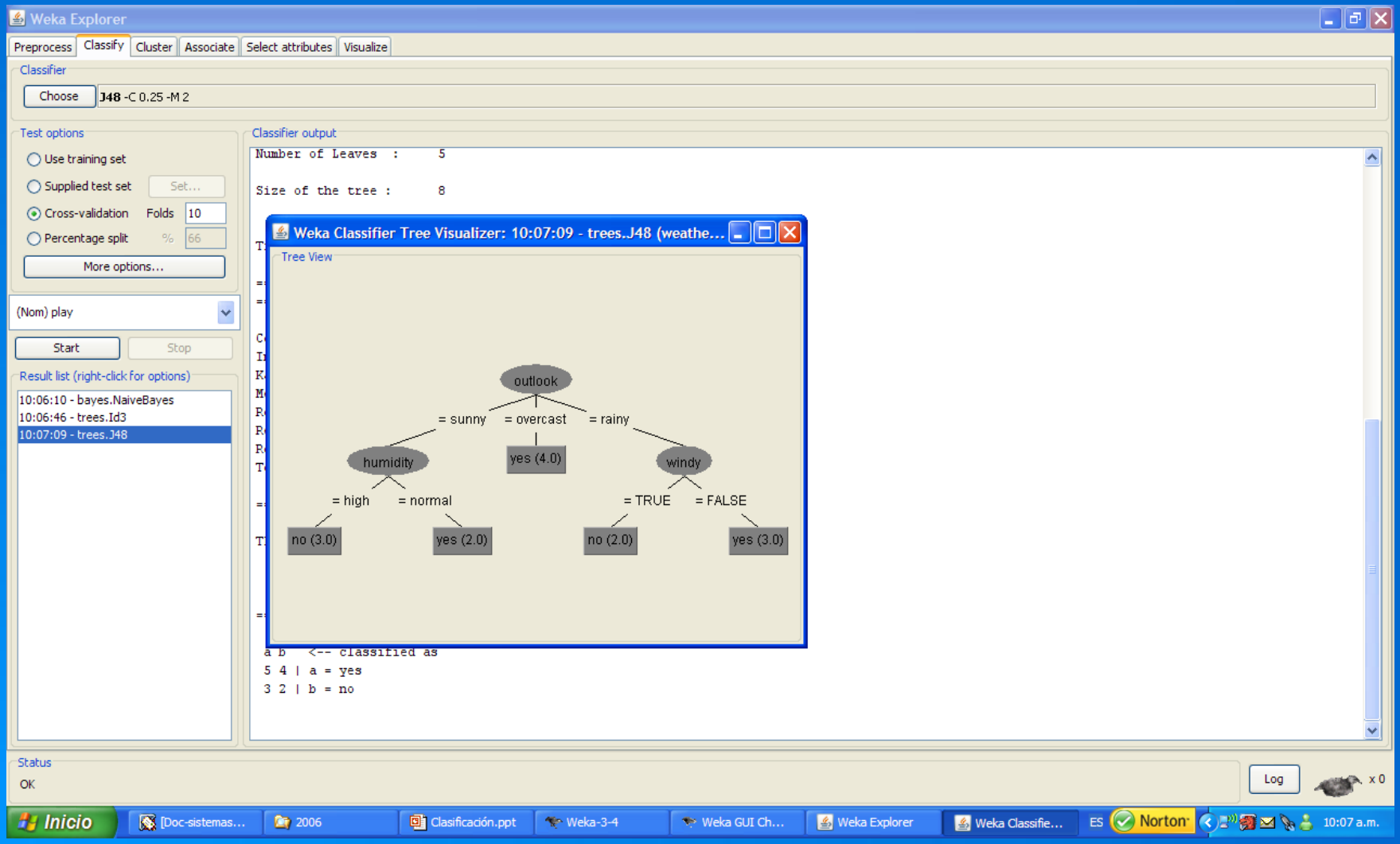
WEKA: Resultados



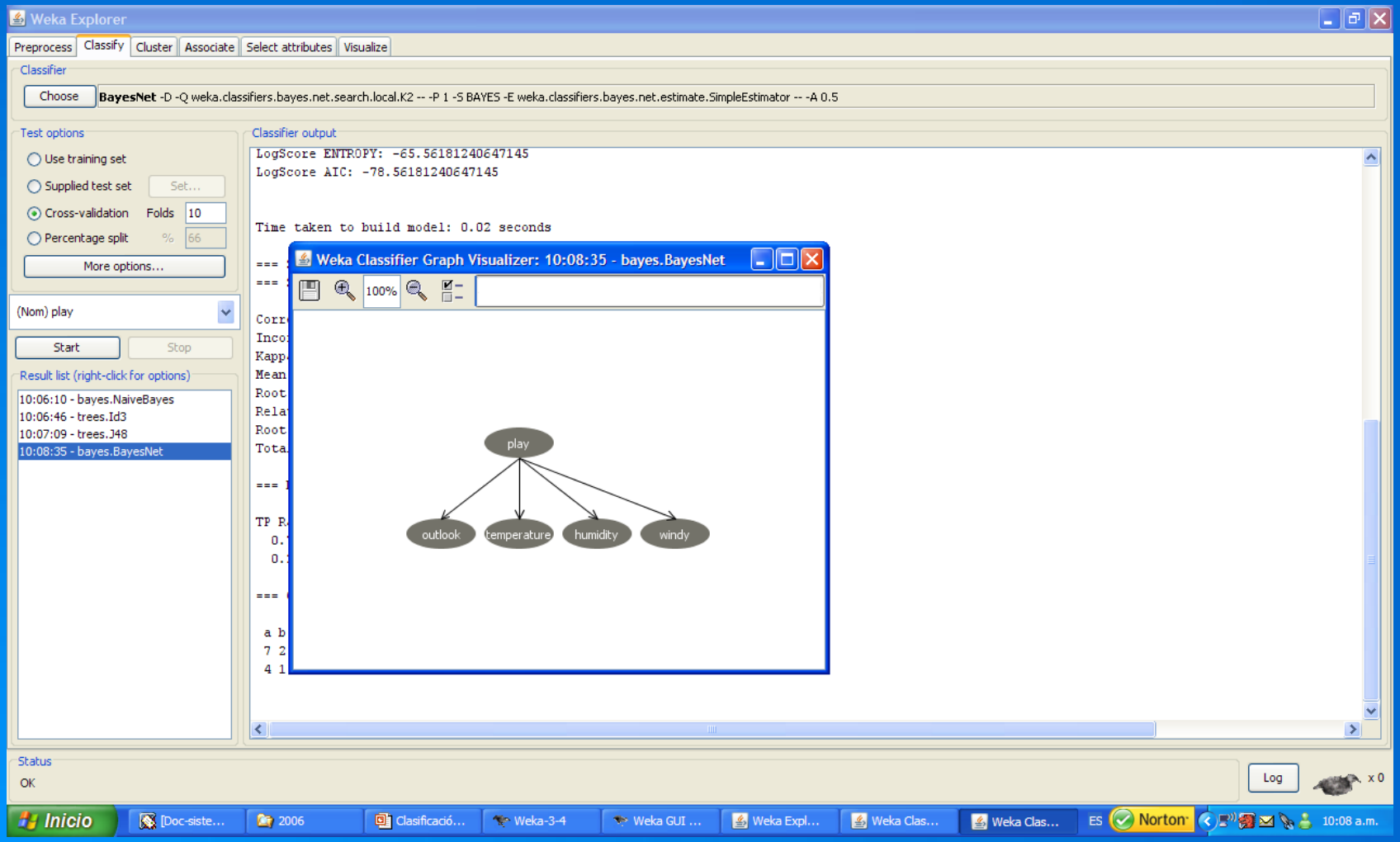
WEKA: Resultados



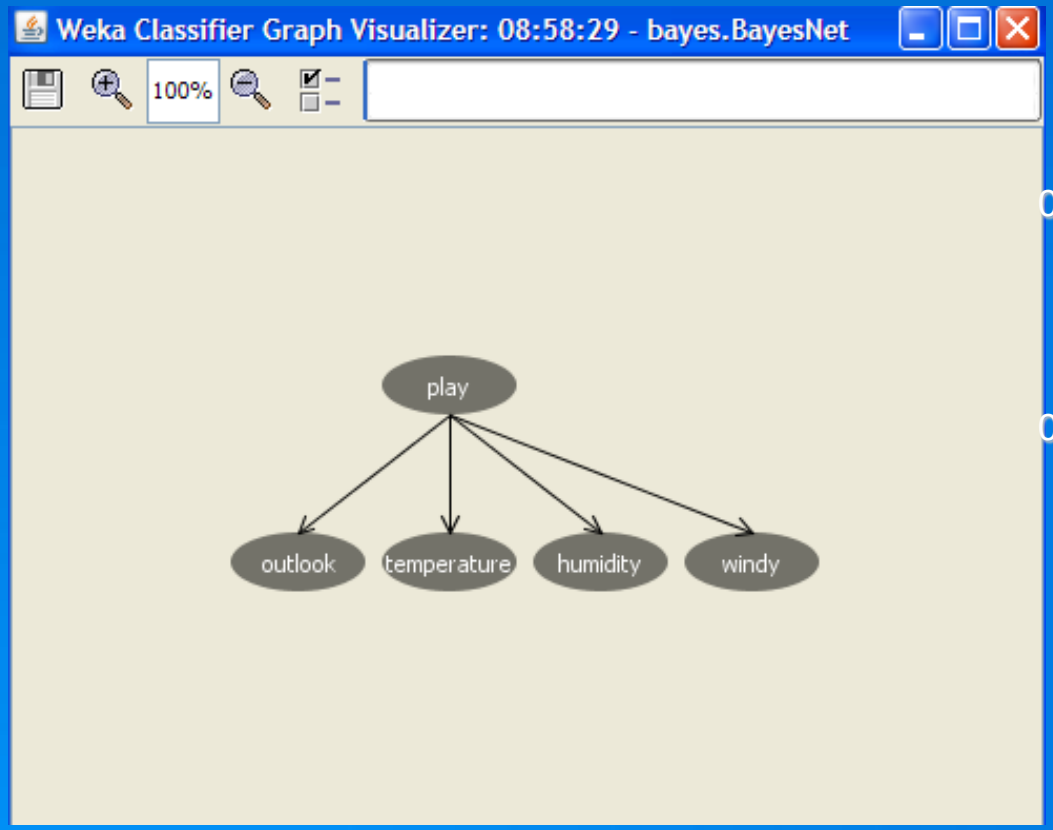
WEKA: Visualización



WEKA: Visualización



Clasificador Bayesiano



Class yes: P(C) = 0.625

Attribute outlook

sunny	overcast	rainy
0.25	0.41666667	0.33333333

Attribute temperature

hot	mild	cool
0.25	0.41666667	0.33333333

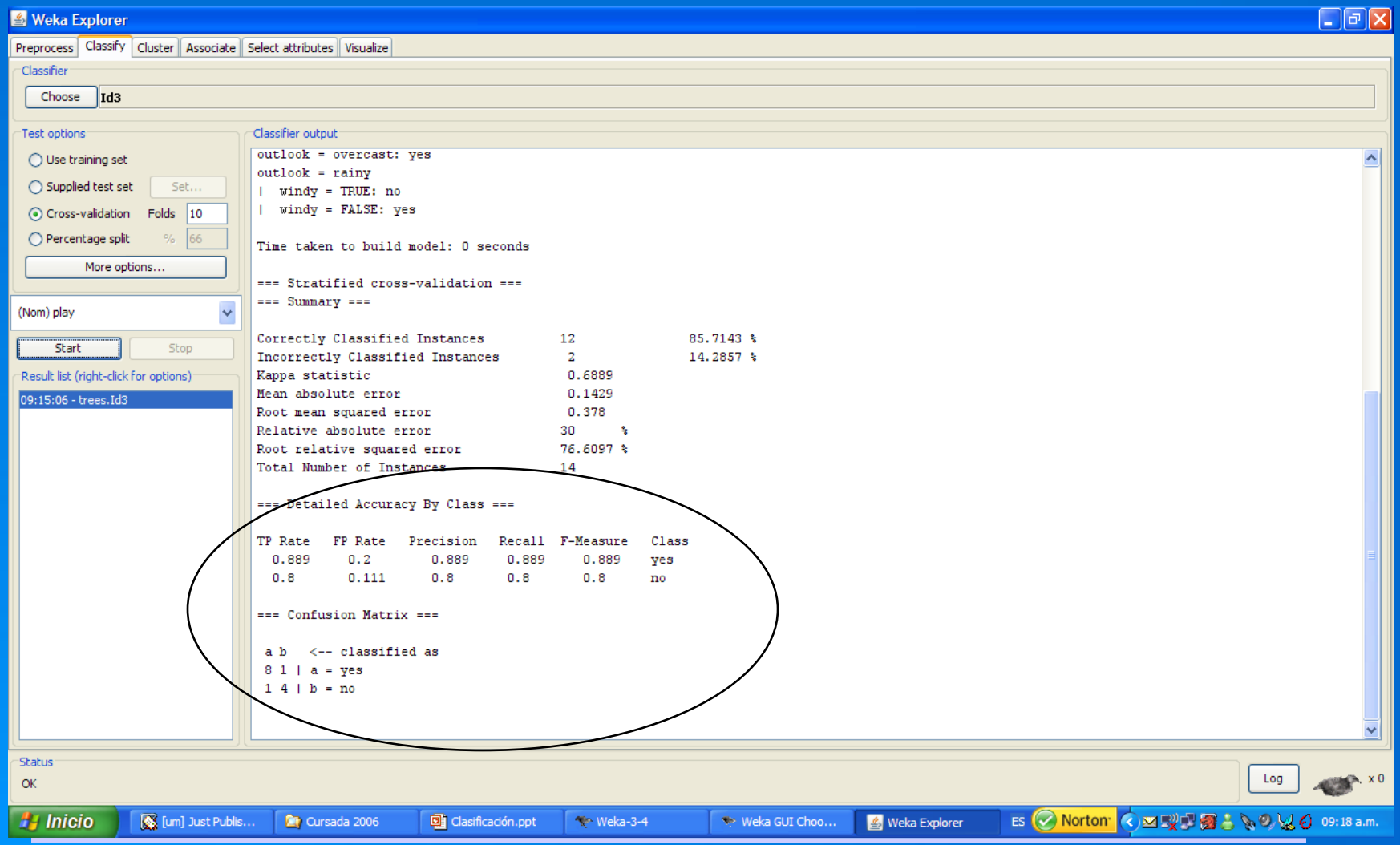
Attribute humidity

high	normal
0.36363636	0.63636364

Attribute windy

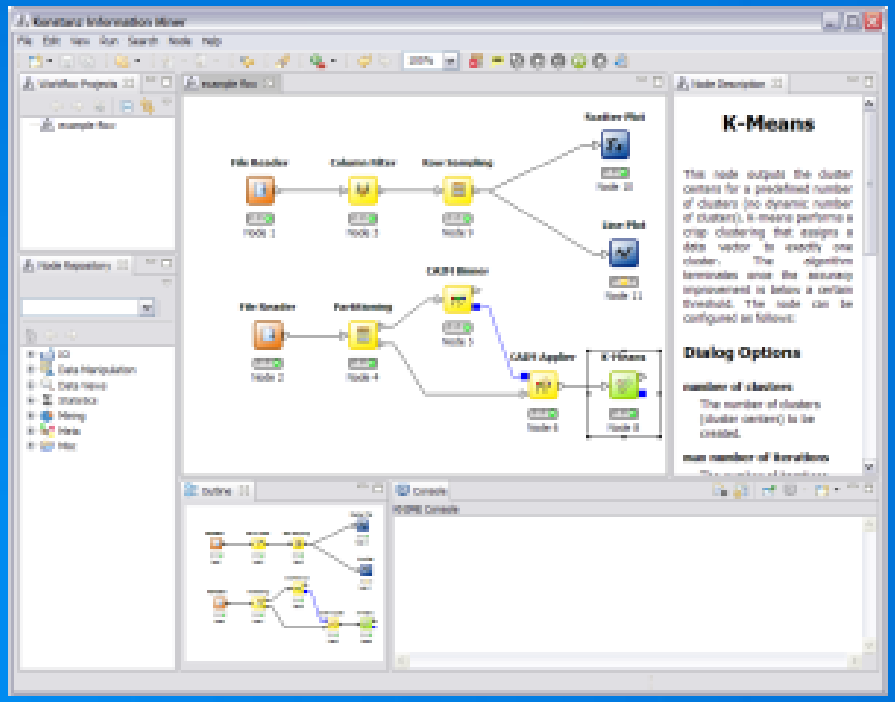
TRUE	FALSE
0.36363636	0.63636364

WEKA: Evaluación



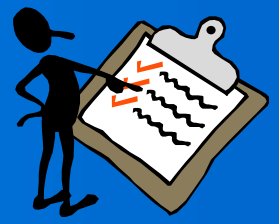
Otras herramientas

- Knime: http://www.knime.org/



Clasificación

- Concepto
  - Clasificación
  - Predicción
  - Evaluación
- Árboles de Decisión
  - Construcción
  - Uso
  - Poda
- Clasificador Bayesiano
- Ejemplo: Detección de líderes de equipo



Ejemplo: clasificación de mascotas

Ejemplos	Estatura	Pelaje	Ojos	Clase
e_1	Baja	Amarillo	Negros	A
e_2	Alta	Amarillo	Castaños	B
e_3	Alta	Cobrizo	Negros	A
e_4	Baja	Marrón	Negros	B
e_5	Alta	Marrón	Negros	B
e_6	Alta	Amarillo	Negros	A
e_7	Alta	Marrón	Castaños	B
e_8	Baja	Amarillo	Castaños	B

ID3 Ejemplo

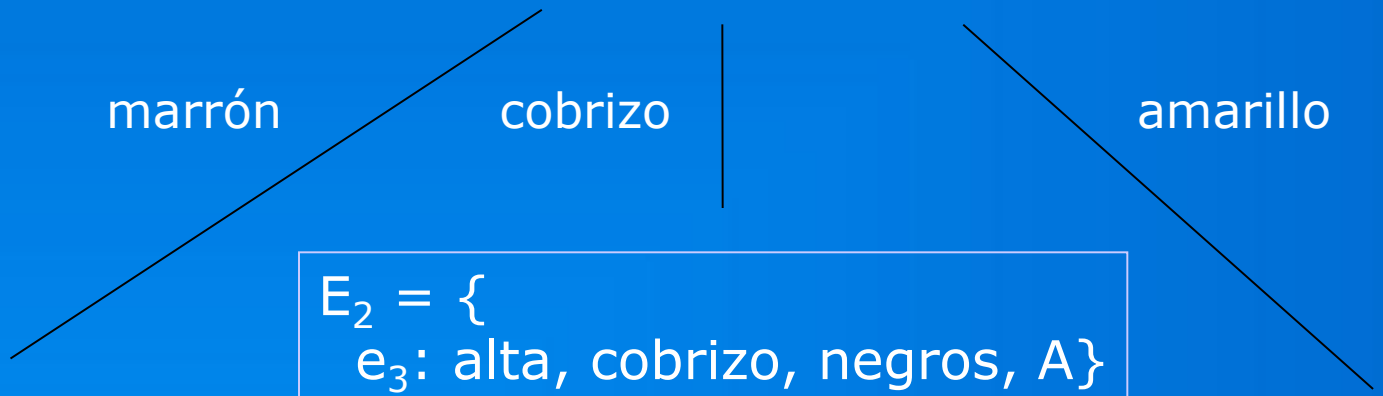
3 ejemplos en clase A; 5 ejemplos en clase B

- $H(E)=I(sA,sB) = 3/8*\log_2(8/3) + 5/8*\log_2(8/5) = 0.954$

- E_1 = {Ejemplos de E que tienen pelaje=marrón}
- E_2 = {Ejemplos de E que tienen pelaje=cobrizo}
- E_3 = {Ejemplos de E que tienen pelaje=amarillo}

ID3 Ejemplo

$E = \{e_1, e_2, e_3, e_4, e_5, e_6, e_7, e_8\}$



E_2 = {
e_3: alta, cobrizo, negros, A}

E_1 = {
e_4: baja, marrón, negros, B
e_5: alta, marrón, negros, B
e_7: alta, marrón, castaños, B}

E_3 = {
e_1: baja, amarillo, negros, A
e_2: alta, amarillo, castaños, B
e_6: alta, amarillo, negros, A
e_8: baja, amarillo, castaños, B}

ID3 Ejemplo

- **H(pelaje, E) = p(pelaje=marrón/E)*H(E_1)+**
  p(pelaje=cobrizo/E) *H(E_2)+
  p(pelaje=amarillo/E) *H(E_3)

$H(E_1)=p^A\log_2(1/p^A)+p^B\log_2(1/p^B) = 0\ \log_2(1/0)+1\ \log_2(1/1) = 0$
$H(E_2)=p^A\log_2(1/p^A)+p^B\log_2(1/p^B) = 1\ \log_2(1/1)+0\ \log_2(1/0) = 0$
$H(E_3)=p^A\log_2(1/p^A)+p^B\log_2(1/p^B) = 2/4\ \log_2(4/2)+2/4\ \log_2(4/2)$
$= 0.5 + 0.5$
$= 1$

H(pelaje/E) = 3/8 0 + 1/8 0 +4/8 1 = 0.5

Ganancia(pelaje) = 0.954 – 0.5 = 0.454

ID3 Ejemplo

- Ganancia(pelaje) = 0.454
- Ganancia(estatura)= 0.003
- Ganancia(ojos) = 0.347