Almacenes y Minería de Datos
Proyecto Final
Entregable 1

Adolfo Marín Arriaga Juan Carlos López López Luis Rodrigo Rojo Morales

October 15, 2016

Algoritmo 1: Naive Bayes

Objetivo

El objetivo de Naive Bayes o Clasificador bayesiano ingenuo es la construccion de clasificadores en base a datos que ya se conocen por ejemplo clasificar si una persona es hombre o mujer basandose en los datos que se tienen de su altura, peso y tamaño del pie. Se usa basandose en el teorema de Bayes y asume que las variables que se usan para predecir son independientes

Descripción

Lo que tenemos es un objeto que es lo que se quiere clasificar, las variables que se van a usar para predecir que son los datos que ya conocemos y distintas clases a donde podria ir cada objeto después de la clasificación. Se usa la regla de bayes que dice que si se tienen 2 eventos A y B se saca la probabilidad de A dado B como:

$$P(A \mid B) = \frac{P(B \mid A)\, P(A)}{P(B)}$$

- $P(A \mid B)$ Es la probabilidad de que se cumpla A dado B, es la probabilidad a posteriori.
- $P(B \mid A)$ Es la probabilidad de B dado que se cumple A.
- $P(A)$ y $P(B)$ Son las probabilidades que se cumplan A y B respectivamente, estas son la probabilidad a priori

Para el clasificador bayesiano ingenuo hay distintas clases en las que puede caer un objeto después de ser clasificado y distintas variables predictivas, entonces si tenemos un conjunto de variables predictivas $X = \{x_1, ..., x_n\}$ y m clases para la clasificación $C_1, ..., C_m$, el clasificador dice que la probabilidad de que un objeto pertenezca a la clase C_k se calcula como:

$$p(C_k|X) = \frac{1}{Z} p(C_k) \prod_{i=1}^{n} p(x_i|C_k)$$

- X es el conjunto de variables predictivas $X = \{x_1, ..., x_n\}$.
- C_k es una clase de $C_1, ..., C_m$
- Z es igual a un factor que depende solo de $x_1, ..., x_n$, es una constante si sus valores son conocidos

Ejemplo:

Un ejemplo es si queremos clasificar frutas, las clases son platano, naranja y otras, se hizó un conjunto de entrenamiento con las variables de longitud, sabor y color y nos dio que:

Fruta	Alargada	No Alargada	Dulce	No Dulce	Amarilla	No Amarilla	Total
Platano	400	100	350	150	450	50	500
Naranja	0	300	150	150	300	0	300
Otras	100	100	150	50	50	150	200
Total	500	500	650	350	800	200	1000

Ahora queremos predecir en que clase cae una nueva fruta que llega y es amarilla, alargada y dulce. Calculamos las probabilidades:

- $P(platano) = 500/1000 = 0.5$
- $P(platano \mid alargada) = \frac{P(alargada|platano)\,P(platano)}{P(alargada)} = \frac{400/500 \cdot 500/1000}{500/1000} = 0.8$
- $P(platano \mid dulce) = \frac{P(dulce|platano)\,P(platano)}{P(dulce)} = \frac{350/500 \cdot 500/1000}{650/1000} = 0.5384$
- $P(platano \mid amarilla) = \frac{P(amarilla|platano)\,P(platano)}{P(amarilla)} = \frac{450/500 \cdot 500/1000}{800/1000} = 0.5625$
- $P(naranja) = 300/1000 = 0.3$
- $P(naranja \mid alargada) = \frac{P(alargada|naranja)\,P(naranja)}{P(alargada)} = \frac{0/300 \cdot 300/1000}{500/1000} = 0$
- $P(naranja \mid dulce) = \frac{P(dulce|naranja)\,P(naranja)}{P(dulce)} = \frac{150/300 \cdot 300/1000}{650/1000} = 0.2307$
- $P(naranja \mid amarilla) = \frac{P(amarilla|naranja)\,P(naranja)}{P(amarilla)} = \frac{300/300 \cdot 300/1000}{800/1000} = 0.375$
- $P(otras) = 200/1000 = 0.2$
- $P(otras \mid alargada) = \frac{P(alargada|otras)\,P(otras)}{P(alargada)} = \frac{100/200 \cdot 200/1000}{500/1000} = 0.2$
- $P(otras \mid dulce) = \frac{P(dulce|otras)\,P(otras)}{P(dulce)} = \frac{150/200 \cdot 200/1000}{650/1000} = 0.2307$
- $P(otras \mid amarilla) = \frac{P(amarilla|otras)\,P(otras)}{P(amarilla)} = \frac{50/200 \cdot 200/1000}{800/1000} = 0.0625$

Ahora para el clasificador con su fórmula:

- $p(platano|alargada, dulce, amarilla) = \frac{1}{Z}p(platano)p(platano|alargada)p(platano|dulce)p(platano|amarilla)$
  $= \frac{1}{Z}0.5 \cdot 0.8 \cdot 0.5384 \cdot 0.5625 = \frac{1}{Z}0.121147$
- $p(naranja|alargada, dulce, amarilla) = \frac{1}{Z}p(naranja)p(naranja|alargada)p(naranja|dulce)p(naranja|amarilla)$
  $= \frac{1}{Z}0.5 \cdot 0 \cdot 0.2307 \cdot 0.375 = \frac{1}{Z}0$
- $p(otras|alargada, dulce, amarilla) = \frac{1}{Z}p(otras)p(otras|alargada)p(otras|dulce)p(otras|amarilla)$
  $= \frac{1}{Z}0.2 \cdot 0.2 \cdot 0.2307 \cdot 0.0625 = \frac{1}{Z}0.00057675$

Conocemos la probabilidad de que lo que recibamos sea dulce, alargado y amarillo entonces tomamos Z como:

- $p(alargada)p(dulce)p(amarilla) = \frac{500}{1000} \cdot \frac{650}{1000} \cdot \frac{800}{1000} = 0.26$

Entonces reemplazando Z

- $p(platano|alargada, dulce, amarilla) = \frac{0.121147}{0.26} = 0.46595$
- $p(naranja|alargada, dulce, amarilla) = \frac{0}{0.26} = 0$
- $p(otras|alargada, dulce, amarilla) = \frac{0.00057675}{0.26} = 0.2218$

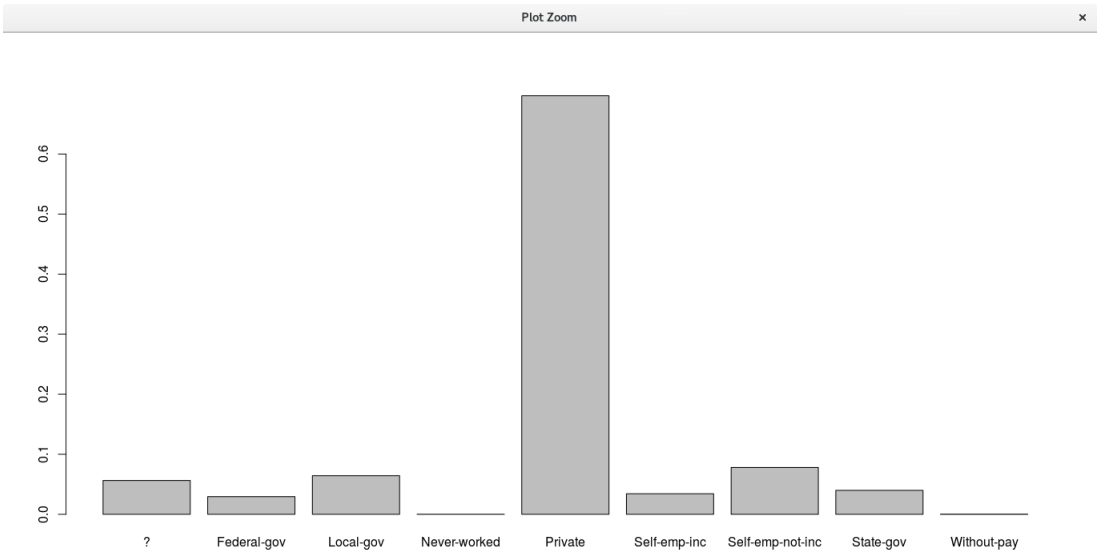
La probabilidad mas grande es la de que sea platano, entonces si llega una fruta que sea alargada, dulce y amarilla la clasificamos como que va a ser un platano.

Bibliografía

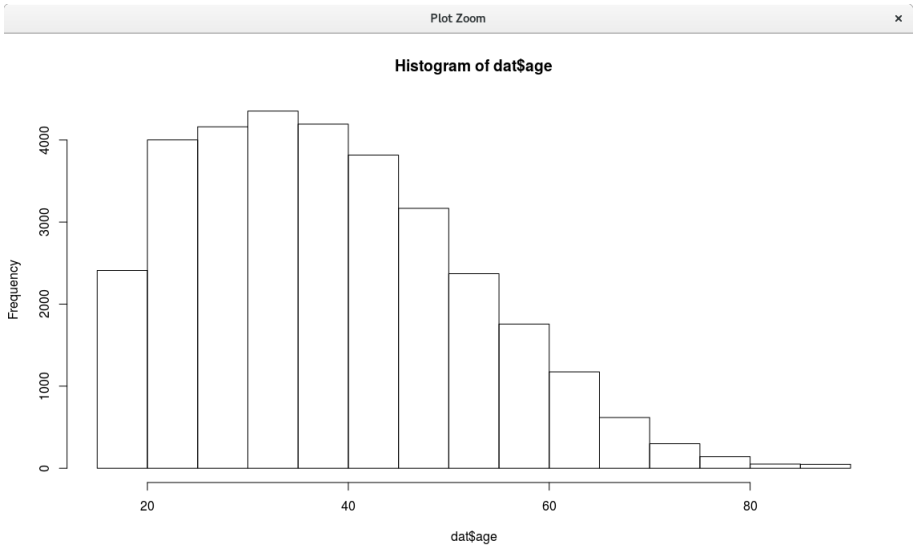
- https://en.wikipedia.org/wiki/Naive_Bayes_classifier
- https://en.wikipedia.org/wiki/Bayes%27_theorem

Plan de trabajo

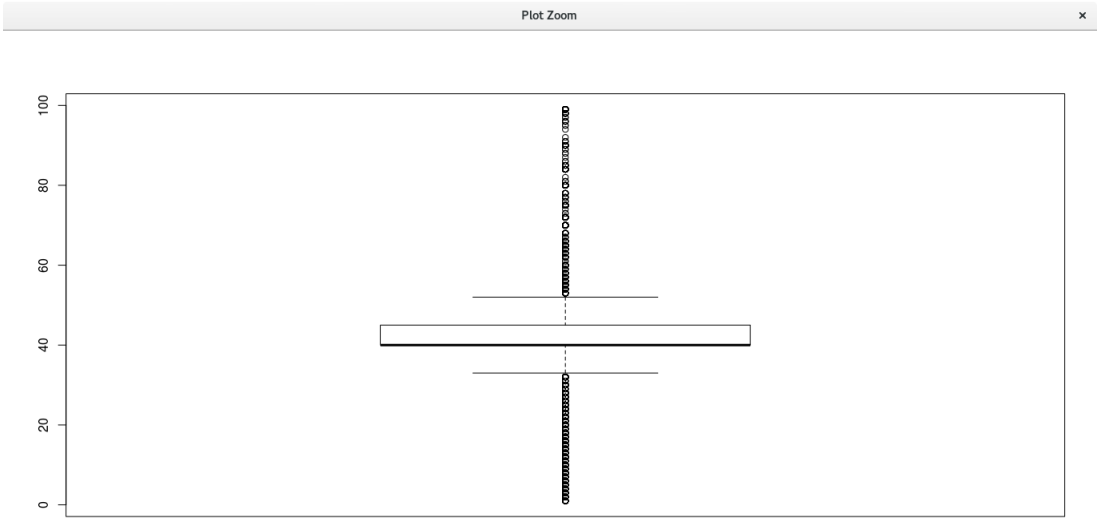
Para nuestro problema en el preprocesamiento de datos es necesario hacer limpieza en los campos workclass, occupation y native-country, porque en algunas filas tienen un ? en vez de alguno de los datos de las opciones que se dan.



También para los datos que son numéricos los datos varian en muchos valores por ejemplo edades tiene 71 valores distintos, entonces para estos valores numéricos para que no halla tantas probabilidades, lo mejor para hacer naive bayes seria hacer discretización de los valores numéricos como por ejemplo con binning para tomarlos por rangos. Así en vez de tomar los 71 valores posibles de edades, tomaría una menor cantidad de rangos.



Los datos que no son númericos si caen en opciones predefinidas por lo que salvo los campos que tienen valores con ? no hay inconsistencias en estos datos. También otros problemas que hay en los datos que son numéricos es que hay muchos outliers, por ejemplo en horas a la semana la mayoria trabaja alrededor de 40 pero hay datos que van desde 1 hasta 99.



En este caso los outliers se podrían en rangos que sean mayor que un valor y menor que un valor en la discretización que se había planteado previamente.

Algoritmo 2: C4.5

Objetivo

Generar un árbol de decisión, por tanto es útil en problemas de clasificación, y por esta razón, C4.5 está casi siempre referido como un clasificador estadístico.

Descripción

Para mencionar la forma de trabajar de este algoritmo en primer lugar hay que mencionar que lasteorías de Shannon

son la base del algoritmo C4.5. La entropía de Shannon es la más conocido y más aplicado. Ésta define la cantidad de información proporcionada por un evento.

Entropía de Shannon:
En general, si damos una distribución de probabilidad

$$P = (p_1, p_2, ..., p_n)$$

y un ejemplar S entonces la información acarreada por esta distribución, también llamada entropía de P está dada por:

$$Entropia(P) = -\sum_{i=1}^{n} p_i log(p_i)$$

Ganancia de Información G(p,T):
Tenemos funciones que nos permiten medir el grado de clases mezcladas para todas las muestras y por tanto cualquier posición de el árbol en la construcción. Queda por definir una función para seleccionar la prueba que debe etiquetar al nodo actual.
Lo siguiente define una ganancia para una prueba T y su posición p

$$Ganancia(p, T) = -Entropia(p) - \sum_{j=1}^{n} (p_j Entropia(p_j))$$

donde los valores (pj) es el conjunto de todos los posibles valores para el atributo T. Podemos usar esta medida para clasificar atributos y construir el árbol de decisión donde en cada nodo se encuentra su atributo con la mayor ganancia de información entre los atributos.
C4.5 utiliza la "ganancia de información", que permite medir una razón de ganancia. La razón de ganancia se define como:

$$RazonGanancia(p, T) = \frac{Ganancia(p, T)}{DivideInfo(p, T)}$$

donde DivideInfo es:

$$DivideInfo(p, test) = -\sum_{j=1}^{n} p' \frac{j}{p} log(p'(\frac{j}{p}))$$

donde:

$$p'(\frac{j}{p})$$

es la proporción de los elementos presentes en la posición de p, tomando el valor de la prueba j-ésima. La definición anterior es independiente de la distribución de ejemplos dentro de las diferentes clases.
Entonces los datos se clasifican en cada nodo del árbol con el fin de determinar el mejor atributo de división. Utiliza el método de la razón de ganancia para evaluar el atributo de división. Los árboles de decisión se construyen en C4.5 mediante el uso de un conjunto de datos de entrenamiento o conjuntos de datos. En cada nodo de la árbol, C4.5 elige un atributo de los datos que más eficazmente divida su conjunto de muestras en subconjuntos. Su criterio es la ganancia de información normalizada (Diferencia de entropía) que resulta de la elección de una atributo para dividir los datos. El atributo con la ganancia de información normalizada más alta es elegido para tomar la decisión.

Ejemplo
Conjunto de datos S:

Day	Outlook	Temperature	Humidity	Wind	Play
D1	Sun	Hot	**85**	Low	No
D2	Sun	Hot	**90**	High	No
D3	Overcast	Hot	**78**	Low	Yes
D4	Rain	Sweet	**96**	Low	Yes
D5	Rain	Cold	**80**	Low	Yes
D6	Rain	Cold	**70**	High	No
D7	Overcast	Cold	**65**	High	Yes
D8	Sun	Sweet	**95**	Low	No
D9	Sun	Cold	**70**	Low	Yes
D10	Rain	Sweet	**80**	Low	Yes
D11	Sun	Sweet	**70**	High	Yes
D12	Overcast	Sweet	**90**	High	Yes
D13	Overcast	Hot	**75**	Low	Yes
D14	Rain	Sweet	**80**	High	No

Entropía(S) = -9/14*log2(9/14)-5/14*log2(5/14) = 0.94

Cálculos para los primeros atributos:

Ganancia(S, Outlook) = Entropía(S) – 5/14*Entropía (S_Sun) - 4/14*Entropía (S_Rain) - 5/14*Entropía ($S_Overcast$) = 0.94 – 5/14*0.9710 – 4/14*0 – 5/14*0.9710

Gain(S, Outlook) = 0.246

Cálculos de Entropías:

Entropía(S_Sun) = - 2/5*log2(2/5) – 3/5*log2(3/5) = 0.9710

Entropía(S_Rain) = -4/4*log2(4/4) – 0*log2(0) = 0

Entropía($S_Overcast$) = - 3/5*log2(3/5) – 2/5*log2(2/5) = 0.9710

Ahora las ganancias de forma similar que se hizo para Ganancia(S, Outlook)

Ganancia(S, Wind) = 0.048

Ganancia(S, Temperature) = 0.0289

Como Humidity tiene valores continuos, la ganancia se calcula de la siguiente forma:

Ganancia(S, Humidity) = ?

Apoyándonos en el conjunto de datos S hay que ordenar los valores del atributo Humidity en orden ascendente, el conjunto de valores es el siguiente:

[65, 70, 70, 70, 75, 78, 80, 80, 80, 85, 90, 90, 95, 96]

Y removemos los valores repetidos:

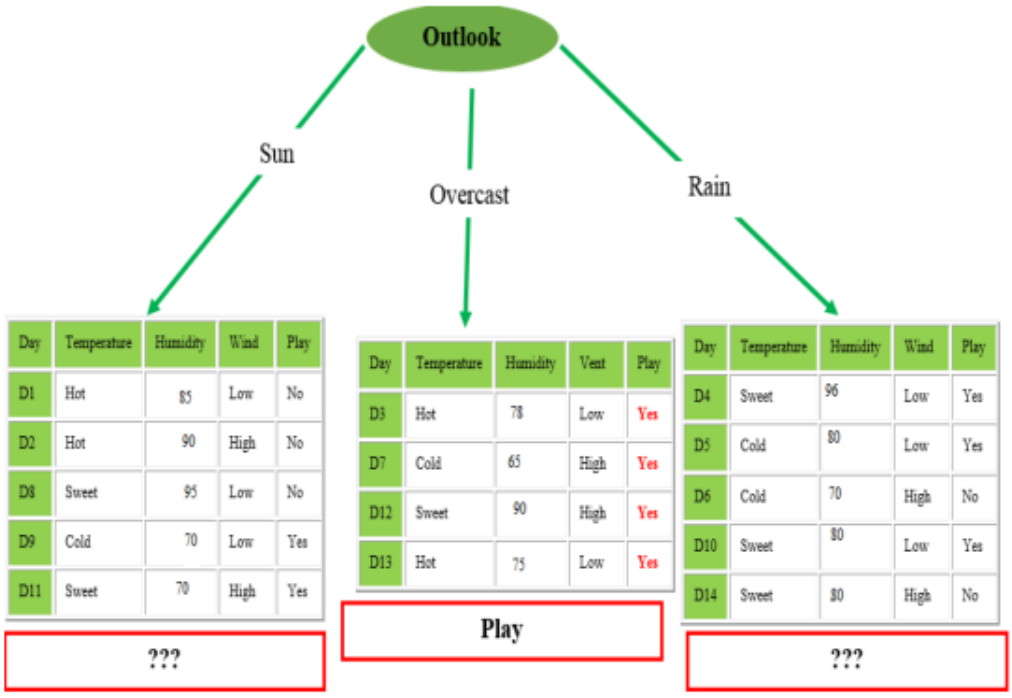
[65, 70, 75, 78, 80, 85, 90, 95, 96]

Utilizando el algoritmo C4.5 la ganancia para el atributo continuo Humidity:

	65		70		75		78		80		85		90		95		96	
interval	≤	>	≤	>	≤	>	≤	>	≤	>	≤	>	≤	>	≤	>	≤	>
Yes	1	8	3	6	4	5	5	4	7	2	7	2	8	1	8	1	9	0
No	0	5	1	4	1	4	1	4	2	3	3	2	4	1	5	0	5	0
Entropy	0	0.961	0.811	0.971	0.721	0.991	0.65	1	0.764	0.971	0.881	1	0.918	1	0.961	0	0.94	0
Info(S,T)	0.892		0.925		0.8950		0.85		0.838		0.915		0.929		0.892		0.94	
Gain	0.048		0.015		0.045		0.09		0.102		0.025		0.011		0.048		0	

Ganancia(S, Humidity) = 0.102

Entonces quien tiene la mayor ganancia de información es el nodo raíz del árbol de decisión C4.5



Bibliografía

- https://en.wikipedia.org/wiki/C4.5_algorithm
- Articulo: A comparative study of decision tree ID3 and C4.5. Autores: Badr HSSINA, Abdelkarim MERBOUHA,Hanane EZZIKOURI,Mohammed ERRITALI

Plan de trabajo

Apoyandonos de las gŕaficas obtenidas anteriormente en el algoritmo 1 y dado que el algoritmo C4.5 es una mejora de ID3, tenemos que en éste podemos:

- Usar datos continuos
- Manejar datos de entrenamiento con valores faltantes
- Usar atributos con distintos valores

Entonces dado que en nuestro conjunto de datos fuente tenemos valores desconocidos (?) podríamos no hacer limpieza de datos en los campos workclass, occupation y native-country y así ejecutar el algoritmo.

También habría necesidad de hacer discretización de los datos ya que algunos rangos de atributos difieren demasiado.