

Modelos para el aprendizaje automático



Introducción a las Redes Bayesianas

Bibliografía:

- *Learning Bayesian Networks*. Neapolitan, R. (1990). Prentice Hall
- *Bayesian Networks in DM*. Heckerman, D. *DM&KD*. 1,79-119. (1997)
- *Redes Bayesianas*. Sangüesa i Solé. UOC.

Clasificación multiclase con Naïve Bayes

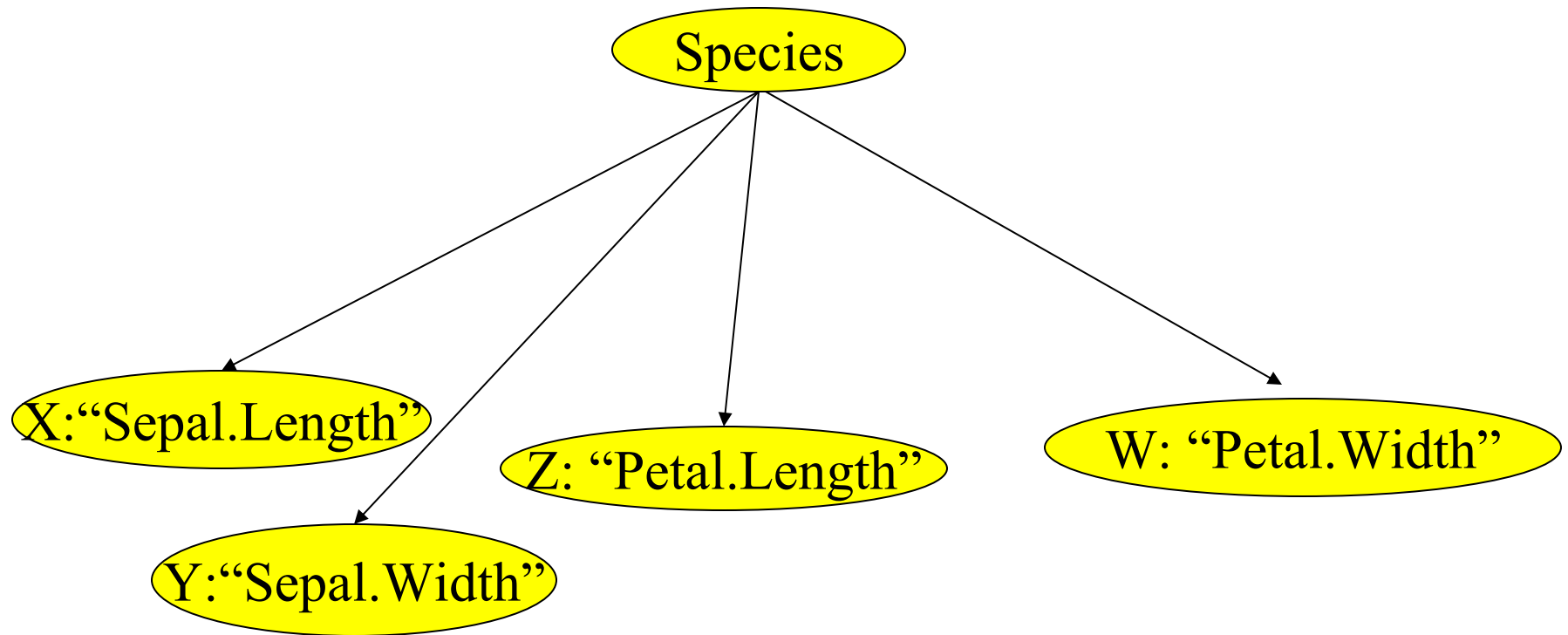
En cualquier red bayesiana, la variable respuesta puede tener más de dos categorías.

La metodología es la misma al caso visto en el ejemplo de “hipoteca”, aunque hay que prestar atención al cálculo de las métricas a partir de la curva de confusión y al cálculo de la curva ROC.

Ejemplo: Naive Bayes con datos iris

Los datos corresponden a 50 muestras para cada variedad de iris (Species: Iris setosa, Iris virginica y Iris versicolor) para las que se midió X:“Sepal.Length”, Y:“Sepal.Width”; Z: “Petal.Length” W: “Petal.Width”

Naive Bayes con datos iris



Vamos a R

las variables predictoras son todas continuas, se suponen Normales.

Lo siguiente da la media y la desviación de las 3 distribuciones dependientes (condicionales) para la variable Z:

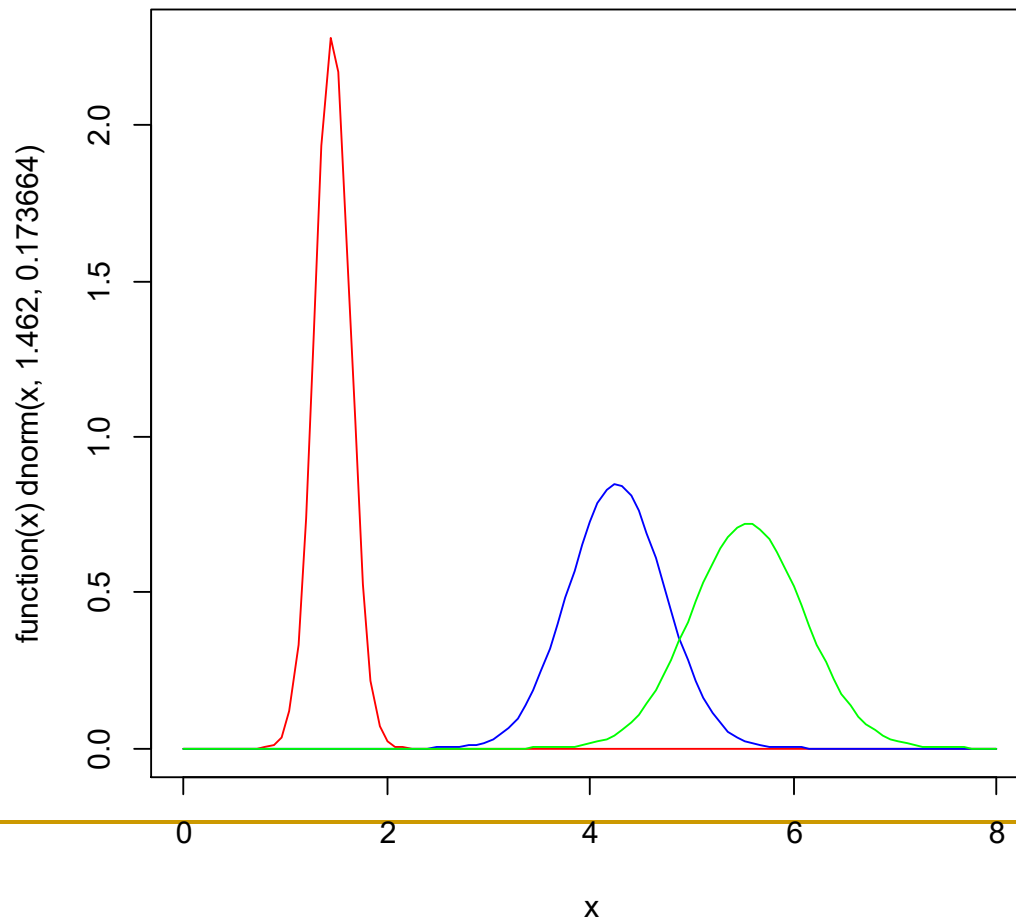
`clasificador$tables$Petal.Length`

<i>Petal.Length</i>		
<i>species</i>	[,1]	[,2]
setosa	1.462	0.1736640
versicolor	4.260	0.4699110
virginica	5.552	0.5518947

Ejemplo: iris

Graficamos las tres densidades normales ajustadas

Petal length distribution for the 3 different species



Ejemplo: iris

Como ejemplo, clasificamos una nueva observación ajustando el modelo sin la última observación

```
irisT= iris[-150,]  
clasificador1<-naiveBayes(irisT[,1:4], irisT[,5])  
predict(clasificador1, iris[150,],type = "raw")
```

Y se tiene las probabilidades:

	setosa	versicolor	virginica
[1,]	3.000941e-143	0.06483664	0.9351634

*es la mayor prob
⇒ se clasifica
(predice)*

Lo que la clasificó como “virginica”. El dato original es: *cons
Ving*

```
iris[150,]  
Sepal.Length Sepal.Width Petal.Length Petal.Width Species  
150          5.9         3         5.1         1.8 virginica
```

Redes bayesianas más generales

Construcción de una Red Bayesiana

El aprendizaje (construcción) de una red bayesiana se puede dividir en dos componentes principales: aprendizaje de estructura y aprendizaje de parámetros.

Aprendizaje estructurado: dado un conjunto de muestras de datos, estime un gráfico acíclico dirigido (DAG) que capture las dependencias entre variables.

Existen distintas métricas para seleccionarlo (entropía, AIC, MDL, Bayesiana, etc) y distintas estrategias de búsqueda para explorar el espacio de estructuras posibles.

Aprendizaje de parámetros: dado un conjunto de muestras de datos y un DAG que captura las dependencias entre variables, estima las distribuciones de probabilidad (condicional) de las variables individuales.

Aprendizaje estructural

El primer paso para ajustar una RB es especificar su estructura. Esto puede hacerse basado en **rutinas automáticas** ó en **juicio de expertos**.

Con el aprendizaje estructural, queremos determinar la estructura del gráfico que mejor captura las dependencias causales entre las variables en el conjunto de datos.

Ejemplo 3: asia

Data set de Lauritzen and Spiegelhalter (1988) que relaciona enfermedades (tuberculosis, cancer de pulmón o bronquitis) y visitas a Asia.

D (*dyspnoea*), a two-level factor with levels yes and no.

T (*tuberculosis*), a two-level factor with levels yes and no.

L (*lung cancer*), a two-level factor with levels yes and no.

B (*bronchitis*), a two-level factor with levels yes and no.

A (*visit to Asia*), a two-level factor with levels yes and no.

S (*smoking*), a two-level factor with levels yes and no.

X (*chest X-ray*), a two-level factor with levels yes and no.

E (*tuberculosis ó lung cancer/bronchitis*), a two-level factor with levels yes and no.

Lauritzen, S. L. and D. J. Spiegelhalter (1988), "Local computations with probabilities on graphical structures and their application to expert systems (with discussion)," *Journal of Royal Statistical Society, Series B*, 50(2), 157-224.

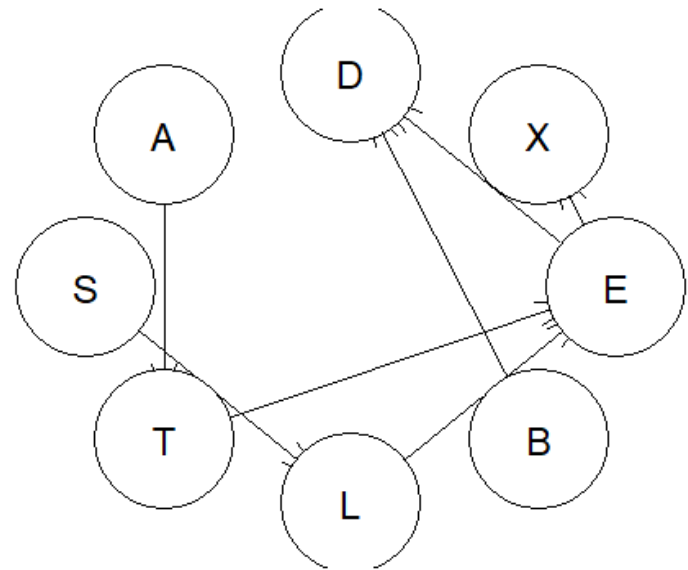
Ejemplo 3: asia

La librería bnlearn de R (<http://www.bnlearn.com/>) permite aprender la estructura de la red.

Además permite ajustar un modelo dando la estructura.

```
library(bnlearn)
data(asia)
#con la estructura de la red prefijada

modelstring(modelA1) =
```



```
[A][S][B][T|A][L|S][D|B:E][E|T:L][X|E]
```

Ejemplo 3: asia

Para ver resultados de la estimación de parámetros:

```
fit1 = bn.fit(modelA1, asia)
```

```
fit1
```

```
bn.fit.barchart(fit1$D)
```

.....

Parameters of node A (multinomial distribution)

Conditional probability table:

no yes

0.9916 0.0084

Parameters of node T (multinomial distribution)

Conditional probability table:

		A	
T		no	yes
no	0.9915288	0.9523809	
yes	0.0084711	0.0476190	

Esto da las probabilidades
condicionales de T/A=yes

.....

Parameters of node D (multinomial distribution)

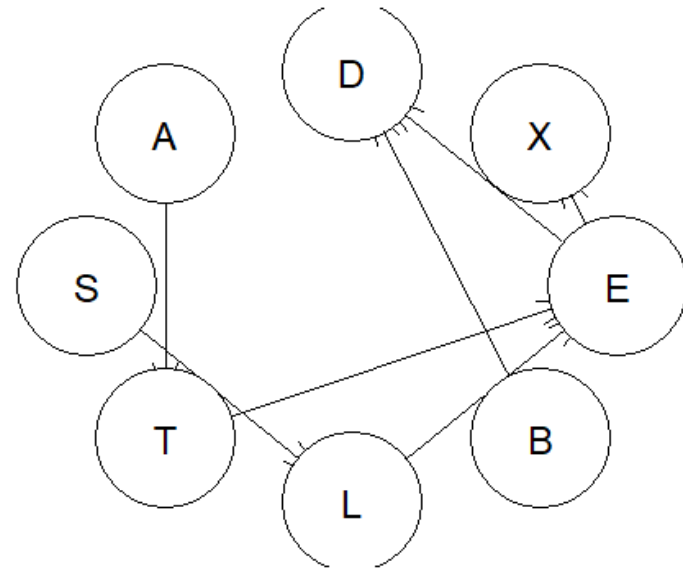
Conditional probability table:

E = no

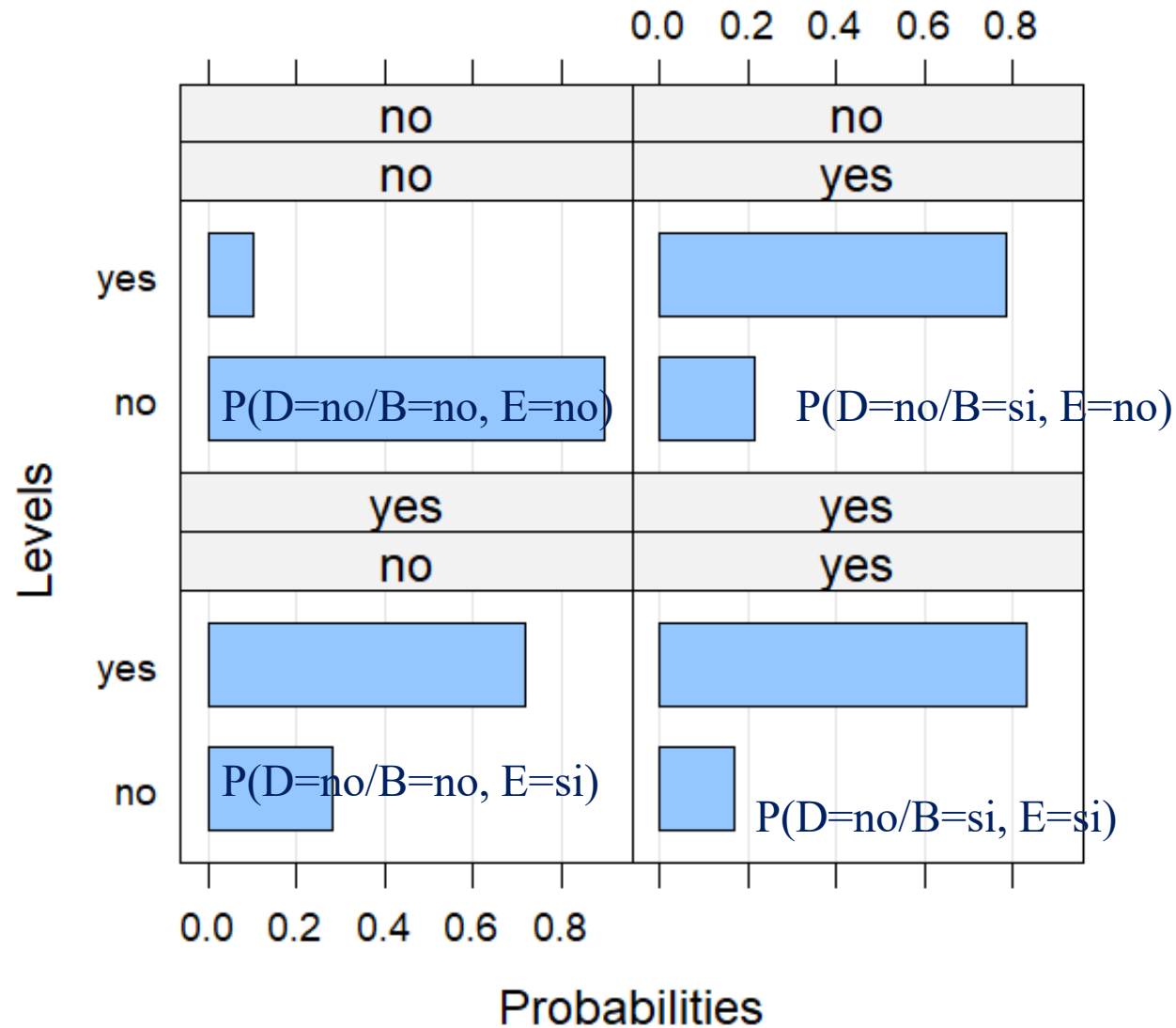
		B	
D		no	yes
		no	yes
	no	0.8962500	0.2163636
	yes	0.1037500	0.7836364

E = yes

		B	
D		no	yes
		no	yes
	no	0.2826087	0.1708861
	yes	0.7173913	0.8291139



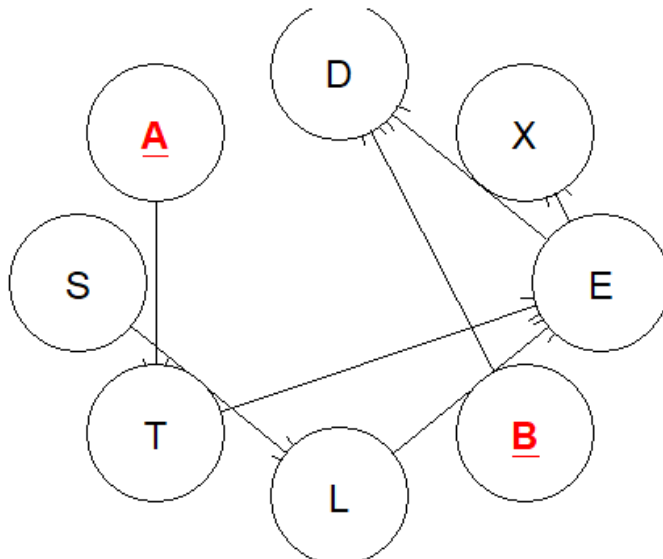
Conditional Probabilities for Node D



Otra opción: le pedimos que ajuste la estructura con Hill Climbing:

"[A][S][T][L|S][B|S][E|T:L][X|E][D|B:E]"

Estructura dada



Hill-Climbing

