parcial2IA2021-22SOLS.pdf

osnofla

Inteligencia Artificial

3º Grado en Ingeniería Informática

Escuela Politécnica Superior
Universidad Autónoma de Madrid

Artificial Intelligence 2021-2022 (EPS-UAM): Second midterm exam
2022/05/10

1. **Uncertainty (3.3 points).**

[Adapted from
Berry Groisman "The end of Sleeping Beauty's nightmare" Berry Groisman, British Journal for the Philosophy of Science 59 (3):409-416 (2008)]

Sleeping Beauty (SB) is a genius statistician, who makes accurate estimates of priors and expert use of Bayes' theorem to give the best predictions possible. On this auspicious day, SB undergoes the following experiment, whose setup is known to her: She is put to sleep on Sunday evening. Then, a practically-minded friend (possibly an engineer) tosses a fair coin (50% probability for Heads, 50% for Tails). If the result is Heads, she is woken on Friday only. If the result is Tails, SB is woken twice, on Monday and on Friday. In addition, she is given a special beverage that causes her to forget whether she was woken on Monday or not. Thus, when she is woken, she does not know whether it is Monday or Friday.

SOLUTION:
Heads = $'H'$; Tails = $'T'$.
Coin toss: $C \in \{'H', 'T'\}$
Sleeping beauty's belief of the result of the coin toss: $SC \in \{'H', 'T'\}$
Day SB is woken: $D \in \{'M', 'F'\}$
Sleeping beauty's belief on the day when asked: $SD \in \{'M', 'F'\}$

- $P\,(C = 'H'|A) = P(C = 'H') = 1/2.$
- $P\,(C = 'T'|A) = P(C = 'T') = 1/2.$
- $P(D = 'M'|C = 'H') = 0.$
- $P(D = 'F'|C = 'H') = 1.$
- $P(D = 'M'|C = 'T') = 1/2.$
- $P(D = 'F'|C = 'T') = 1/2.$

a. Consider a situation in which she is woken on Monday. Determine the probability that the coin toss is Heads and that it is Tails. Provide an answer to these questions using Bayes' theorem.

$$P(C = 'H'|D = 'M') = \frac{P(D = 'M'|C = 'H')P(C = 'H')}{P(D = 'M')} = \frac{0 \times \frac{1}{2}}{P(D = 'M')} = 0$$
$$P(C = 'T'|D = 'M') = \frac{P(D = 'M'|C = 'T')P(C = 'T')}{P(D = 'M')} = \frac{\frac{1}{2} \times \frac{1}{2}}{P(D = 'M')} = 1$$

b. Consider a situation in which she is woken on Friday. What is the probability that the coin landed Heads? And Tails? Provide an answer to these questions using Bayes' theorem.

$$P(C='H'|D='F')=\frac{P(D='F'|C='H')P(C='H')}{P(D='F')}=\frac{1\times\frac{1}{2}}{P(D='F')}=\frac{2}{3}$$

$$P(C='T'|D='F')=\frac{P(D='F'|C='T')P(C='T')}{P(D='F')}=\frac{\frac{1}{2}\times\frac{1}{2}}{P(D='F')}=\frac{1}{3}$$

c. At a time that she is woken, what is the probability that it is Monday? And Friday?

- $P(D='M')=P(D='M'|C='H')\,P(C='H') + P(D='M'|C='T')\,P(C='T')=0\times\frac{1}{2}+\frac{1}{2}\times\frac{1}{2}=\frac{1}{4}.$
- $P(D='F')=P(D='F'|C='H')\,P(C='H') + P(D='F'|C='T')\,P(C='T')=1\times\frac{1}{2}+\frac{1}{2}\times\frac{1}{2}=\frac{3}{4}.$

Assume that SB knows that she is asked to make predictions every time she is woken. She also knows that she will get rewarded by some amount (the same every time) whenever she makes a correct prediction. We know that she wants to maximize her total gain (i.e., the expected accumulated reward on Friday, once the experiment has been completed). Since she is a true blood Bayesian, her decisions are based on probability estimates that reflect her expectations. At a particular time that she is woken,

d. what is the probability estimate that the day is Monday **from her viewpoint**? And Friday? Based on these probabilities, what is her best prediction of the day (Monday or Friday) on which she has been woken?

$$P(SD='M')=\frac{1}{3}\ ;\ \ P(SD='F')=\frac{2}{3}$$

e. what is the probability estimate of the coin having landed Heads **from her viewpoint**? And Tails? Based on these probabilities, what is her best prediction of the result of the coin toss (Heads or Tails) at the time that she is woken?

- $P(SD='M'|SC='H')=0.$
- $P(SD='F'|SC='H')=1.$
- $P(SD='M'|SC='T')=1/2\,.$
- $P(SD='F'|SC='T')=1/2\,.$

- $P(SC='H'|SD='M')=0.$
- $P(SC='T'|SD='M')=1.$

- $P(SC = 'H'|SD = 'F') = 1/2$.
- $P(SC = 'T'|SD = 'F') = 1/2$.

Check consistency using Bayes' theorem

$$P(SD = 'M'|SC = 'H') = \frac{P(SC = 'H'|SD = 'M')P(SD = 'M')}{P(SC = 'H')} = \frac{0 \times \frac{1}{3}}{P(SC = 'H')} = 0$$

$$P(SD = 'F'|SC = 'H') = \frac{P(SC = 'H'|SD = 'F')P(SD = 'F')}{P(SC = 'H')} = \frac{\frac{1}{2} \times \frac{2}{3}}{P(SC = 'H')} = 1$$

$$P(SD = 'M'|SC = 'T') = \frac{P(SC = 'T'|SD = 'M')P(SD = 'M')}{P(SC = 'T')} = \frac{1 \times \frac{1}{3}}{P(SC = 'T')} = \frac{1}{2}$$

$$P(SD = 'F'|SC = 'T') = \frac{P(SC = 'H'|SD = 'F')P(SD = 'F')}{P(SC = 'H')} = \frac{\frac{1}{2} \times \frac{2}{3}}{P(SC = 'H')} = \frac{1}{2}$$

- $P(SC = 'H') = P(SC = 'H'|SD = 'M')\, P(SD = 'M')$
  $+ P(SC = 'H'|SD = 'F')\, P(SD = 'F') = 0 \times \frac{1}{3} + \frac{1}{2} \times \frac{2}{3} = \frac{1}{3}$
- $P(SC = 'T') = P(SC = 'T'|SD = 'M')\, P(SD = 'M')$
  $+ P(SC = 'T'|SD = 'F')\, P(SD = 'F') = 1 \times \frac{1}{3} + \frac{1}{2} \times \frac{2}{3} = \frac{2}{3}$

2. Árboles de Decisión (3.4 puntos). Estamos trabajando con una consulta pediátrica para facilitar la diagnosis de varicela en niños. Considera la siguiente muestra de pacientes que usaremos para entrenar un árbol de decisión con el algoritmo ID3 (la clase es la última columna):

id	Presencia de granos	Estudia en	Varicela
1	T	Colegio	Yes
2	T	Casita Montessori	Yes
3	T	Colegio	No
4	T	Casa	No
5	F	Casa	No
6	F	Colegio	Yes

a) Detalla la metodología y ecuaciones que utilizarías para poder construir el árbol.
b) ¿Qué pregunta pondría el algoritmo ID3 en la raíz? Detalla los cálculos que has de realizar para responder correctamente a esta pregunta.

Conseguimos completar la información por paciente con un atributo más:

id	Presencia de granos	Estudia en	Temperatura	Varicela
1	T	Colegio	38	Yes
2	T	Casita Montessori	38	Yes
3	T	Colegio	36	No
4	T	Casa	37	No
5	F	Casa	36	No
6	F	Colegio	37	Yes

c) ¿Qué cambios en tu estrategia tendrías que hacer y qué tipo de cálculos para introducir esta nueva variable? Se pide la descripción no los cálculos en sí.
d) Si desarrollamos un modelo k-NN con k-3, ¿cuál será la respuesta del modelo para un nuevo dato con granos "T", estudia en "casita montessori" y temperatura "37"? Codifica las variables de la forma que creas más conveniente y explica tus decisiones y razonamientos.

SOLUCIÓN:

a) Tenemos dos atributos "granos" y "estudia" y la clase a predecir "varicela". Por lo que tendremos un árbol con dos niveles, uno por cada atributo. Ambos atributos son categóricos por lo que podemos usar el enfoque ID3 sin problema. Uno de los niveles tendrá 2 ramas (el correspondiente a "granos") ya que tiene dos posibles valores el atributo y el otro niveles tendrá 3 ramas (el correspondiente a "estudia") ya que tiene tres posibles valores el atributo. Para construir el árbol necesitamos saber cual de los dos atributos va en la raíz del árbol (el primer nivel) y cual va en el segundo nivel. El atributo que va en la raíz es el que mejor separa el espacio de estados. Para ello usamos una medida de desorden. En concreto para Id3 usamos la ganancia de información el atributo que tenga este valor mayor será el que vaya en la raíz. Las fórmulas para calcular la IG son las siguientes:
IG= H(C) – H(C|A)
Entropy of a feature:

$$H(A) = -\sum_{i=1}^{M} P(A_i)\log_2 P(A_i) = -\sum_{i=1}^{M} \frac{Freq(A_i)}{N}\log_2\frac{Freq(A_i)}{N}$$

Conditional Entropy of C|A

$$H(C|A) = \sum_{i=1}^{M} P(A_i) \times H(C|A_i) = \sum_{i=1}^{M} \frac{Freq(A_i)}{N} \times H(C|A_i).$$

Entropy of C conditioned on A

$$H(C|A_i) = -\sum_{j=1}^{L} P(C_j|A_i)\log_2 P(C_j|A_i) = -\sum_{j=1}^{L} \frac{Freq(C_j)}{K}\log_2\frac{Freq(C_j)}{K}$$

b) Realizamos los cálculos:

Entropy(ClassChickenPox) = – p(Yes) * $\log_2$p(Yes) – p(No) * $\log_2$p(No) =

– (3/6) *$\log_2$(3/6) – (3/6) *$\log_2$(3/6) = -0.5*(-1) -0.5*(-1) = 1

Conditional Entropies of C|A

Partition using "spots"

- Step 1)

- P(SpotsYes) =4/6=0.66
- P(SpotsNo) = 2/6= 0.33;

- Step 2)
  - Entropy(Class|Spots=Yes) = – p(No) $\log_2$p(No) – p(Yes) $\log_2$p(Yes) = – (3/3)$\log_2$(3/6) – (3/6)$\log_2$(3/6) = 1
  - Entropy(Class| Spots =No) = – (1/2)$\log_2$(1/2) – (1/2)$\log_2$(1/2) = 1
- Step 3)
  - Entropy(Class|Spots) =0.66*1 + 0.33*1= 1
- Step 4) **IG(Class|A)= H(C)-H(C|A)**
  - IG(Class|Spots) = H(Class) - H(Class|spots)= 1-1= 0

Partition using "studies at"

- Step 1)
  - P(Home) =2/6=0.33
  - P(School) = 3/6= 0.5;
  - P(Montessori) = 1/6= 0.17;
  -
- Step 2)
  - Entropy(Class| Home) = – p(No) $\log_2$p(No) – p(Yes) $\log_2$p(Yes) = – (2/2)$\log_2$(2/2) – (0/2)$\log_2$(0/2) = 0
  - Entropy(Class| School) = – (1/3)$\log_2$(1/3) – (2/3)$\log_2$(2/3)= -0.33*(-1.6) -0.67*(-0.58) = 0,528+0.389 = 0,917
  - Entropy(Class| Montessori) = – (0/1)$\log_2$(0/1) – (1/1)$\log_2$(1/1) = 0
- Step 3)
  - Entropy(Class|Studies) =0.33*0 + 0.5*0.917 + 0.16*0 = 0,4585
- Step 4) **IG(Class|A)= H(C)-H(C|A)**
  - IG(Class|Studies) = H(Class) - H(Class|Studies)= 1-0,789= 0.5415

"studies at " is chosen at the root by having the maximum gain of information (minimum entropy of the class given the question)

c) Tenemos un Nuevo atributo, temperature, este atributo es de tipo ordinal. Id3 no fue diseñado para atributos ordinales, su siguiente versión c4.5 sí. Por eso tendríamos que cambiar a una estrategia c4.5. Dicha estrategia sí que considera atributos ordinales y los pasa a discretos. En dicha estrategia dividimos construimos ramas binarias. Al "spots" ser binario se quedaría igual. El atributo "studies" pasaríamos a codificarlo como onehot encoding teniendo 3 atributos binarios "yes/no", "home", "school" y "Montessori". Y pasaríamos a calcular el IG (o IG ratio) de cada uno de ellos. Por último para la variable continua "temperatura" calcularíamos el IG (o IG ratio) resultante de dividir el atributo de forma binaria (<= X y > X, donde X son los diferentes valores de temperatura existentes en el dataset ordenados de menor a mayor, menos el mayor). La partición que de un mejor IG (o IG ratio) será la que usemos para determinar qué atributo va en la raíz del árbol. Este proceso se repetirá con los siguientes niveles con el restante de instancias (filas) del dataset repartidos por las diferentes ramas según se baja por el árbol.

d) Codificamos los atributos, Para las dos variables categóricas usamos one hot encoding. Al "spots" ser binario se 0 = no, yes = 1. El atributo "studies" pasaríamos a codificarlo como one hot encoding teniendo 3 atributos binarios "yes/no", "home", "school" y "Montessori". El atributo "temperature" es ordinal y se queda como está. El dato a predecir codificado quedaría como un vector (1,0,0,1,37, ?) y debemos predecir su valor con las 3 instancias más parecidas. Para ello usaremos la distancia euclídea:

Euclidean:

$$d(x_i, x_j) = \sqrt{\sum_{a=1}^{n} (x_i[a] - x_j[a])^2}$$

Id	spots	Home	School	Montessori	Temperature	Chickenpox	Distance euclidean
1	1	0	1	0	38	1	3
2	1	0	0	1	38	1	1
3	1	0	1	0	36	0	3
4	1	1	0	0	37	0	2
5	0	1	0	0	36	0	4
6	0	0	1	0	37	1	3

1	0	0	1	37	?

Los dos más parecidos marca uno s´i y otro no. Asique la decisión estará en el tercero. Tenemos 3 empates, dos de ellos dicen sí y uno dice no. Dependiendo de la estrategia que se use para elegir el tercero desempatado se dirá si o no. Habiendo dos "sí" será más probable que salga el sí.

3. **Logistic Regression (3.3).** We have these two datasets:

Dataset 1:

x1	x2	class
0	0	1
1	1	0
0	1	0
1	0	0

Dataset 2:

x1	x2	class
0	0	1
1	1	1
0	1	0
1	0	0

a. (1.0) Is it possible to train a logistic regression to completely separate the classes in dataset 1? why? and in the case of dataset 2? why? **Help:** draw each dataset in the attribute space (x1, x2).
b. (1.0) If all the logistic regression weights are 0.2, what is the predicted probability for class 1 in each of the four patterns?
c. (0.65) Can C4.5 learn dataset 1 such that it correctly classifies all 4 cases? If your answer is yes, argue for it and draw that tree. If your answer is no, argue why.
d. (0.65) Repeat the previous question for dataset 2 including the reasoning and the tree drawing if your answer is yes. What is the information gain of the first attribute?

HELP:

Output: $h(\mathbf{x}_n) = \sigma\left(\mathbf{w}^T \cdot \mathbf{x}_n\right)$

$$\sigma(z) = \frac{1}{1+e^{-z}} \qquad \mathbf{z} = \mathbf{w}^T \cdot \mathbf{x}_n = \sum_{d=0}^{D} w_d x_n^{(d)}$$

a. (1.0) Dataset 1 is a NOR, which is linearly separable. Dataset 2 is a NXOR, which is not linearly separable. So the logistic regression can only solve completely Dataset 1.

b. (1.0)

Dataset 1:

x0	x1	x2	class	z	σ(z)
1	0	0	1	0.2	0.550
1	1	1	0	0.6	0.646
1	0	1	0	0.4	0.599
1	1	0	0	0.4	0.599

Dataset 2:

x0	x1	x2	class	z	σ(z)
1	0	0	1	0.2	0.550
1	1	1	1	0.8	0.690
1	0	1	0	0.4	0.599
1	1	0	0	0.4	0.599

c. (0.65) The only possible cuts at the tree are x1>0.5 (0+2-, 1+1-) and x2>0.5 (0+2-, 1+1-). So there are 2 possible trees that can separate all points in this dataset:

A possible tree is:

```
- x1>0.5
        yes: x2>0.5
                yes: class=1
                no: class=0
        no: class=0
```

The other possible tree is:

```
- x2>0.5
        yes: x1>0.5
                yes: class=1
                no: class=0
        no: class=0
```

d. (0.65) There are two possible trees:

```
- x1>0.5
        yes: x2>0.5
                yes: class=1
                no: class=0
        no: x2>0.5
                yes: class=0
                no: class=1
```

The other option is:

```
- x2>0.5
        yes: x1>0.5
                yes: class=1
                no: class=0
        no: x1>0.5
                yes: class=0
                no: class=1
```

In both cases the information gain of the first attribute is 0 since the frequencies of the classes before partitioning are 50%-50%, and after the partition remain the same in both branches (yes and no). In other words, the entropies before and after the partition are 1 bit.