Bayes Learning

- Aprendizaje basado en la regla de Bayes
- Bases probabilísticas a metodologías comunes en machine learning

$$\boxed{P(A|B) = \frac{P(A) \times P(B|A)}{P(B)}}$$

Sea D un conjunto de datos observados

Queremos encontrar el $\boxed{\text{mejor}}$ modelo/hipótesis h para los datos

El modelo $\boxed{\text{más probable}}$ para el fenómeno observado, dados:

- los datos observados
- algún conocimiento previo sobre las probabilidades de las diferentes hipótesis en el espacio H de hipótesis

Ejemplo: Si las etiquetas reflejan cercanía, es más probable un modelo que asigne etiquetas cercanas a datos cercanos!

Denotamos:

Prior probability $P(h)$: Probabilidad inicial de que un modelo ocurra, previa a la observación de los datos

Conocimiento previo sobre el fenómeno

> ¡Si no tenemos conocimiento previo, asumimos $P(h)$ igual para todas las hipótesis!

$P(D)$. Probabilidad de que los datos D sean observados (sin saber nada sobre la hipótesis

$P(D|h)$: Probabilidad de observar D en un mundo en donde h se cumple

$P(h|D)$: Probabilidad de que h se cumpla dados los datos observados D.

→ "Posterior Probability"

Queremos encontrar el modelo más probable $\boxed{h}$ dados los datos $\boxed{D}$

En otras palabras, queremos maximizar $P(h|D)$.

$$\boxed{\underset{h \in H}{\operatorname{argmax}} \; P(h|D)}$$

Cualquier hipótesis que maximice esta expresión se denomina "Máximo a Posteriori" (MAP)

Contamos con la regla de Bayes

$$P(h|D) = \frac{P(D|h)P(h)}{P(D)}$$

hipótesis máxima a posteriori (MAP)

$$h_{MAP} = \underset{h \in H}{\arg\max}\ P(h|D)$$

$$= \underset{h \in H}{\arg\max}\ \frac{P(D|h)\,P(h)}{P(D)}$$

$$= \underset{h \in H}{\arg\max}\ P(D|h)\,P(h)$$

$$= \underset{h \in H}{\arg\max}\ P(D|h)$$

Maximum Likelihood hypothesis (ML)

hipótesis máxima a posteriori (MAP)

$$h_{MAP} = \underset{h \in H}{\arg\max}\ P(h|D)$$

$$= \underset{h \in H}{\arg\max}\ \frac{P(D|h)P(h)}{P(D)}$$ (Bayes)

$$= \underset{h \in H}{\arg\max}\ P(D|h)P(h)$$ ($P(D)$ no depende de h)

$$= \underset{h \in H}{\arg\max}\ P(D|h)$$ (Asumiremos que $P(h)$ es igual para toda h, por lo que no depende de h)

Utilicemos lo anterior para el caso de variable objetivo $\boxed{\text{continua}}$ bajo algunos supuestos:

X: Espacio de instancias

H: Espacio de hipotesis

Cada $h \in H$ es de la forma $h: X \longrightarrow \mathbb{R}$

D: Conjunto con m datos $\{(x_1, d_1) \cdots (x_m, d_m)\}$

y aqui un $\boxed{\text{supuesto importante:}}$

Las etiquetas d_i estan afectadas por un ruido:

$$d_i = f(x_i) + e_i$$

en donde el error tiene distribución normal $e_i \sim N(\underbrace{0}_{\text{Media } 0}, \sigma^2)$

$$d_i \sim N(f(x), \sigma^2)$$

Nota: La variable objetivo es continua, así que las probabilidades están dadas por una función de densidad.

En particular, recordemos la función de densidad de la distribución normal:

$$\boxed{f_{N(\mu,\sigma^2)}(x) = \frac{1}{\sqrt{2\pi\sigma^2}}\, e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2}}$$

Ahora sí calculemos la hipótesis más probable

$$h_{ML} = \underset{h \in H}{\operatorname{argmax}}\, P(D|h)$$ "Maximum Likelihood"

→ Asumiendo un conjunto fijo de instancias de entrenamiento $\{x_1, \dots, x_m\}$ podemos considerar el conjunto D como el conjunto de las etiquetas

$D = \{d_1, \dots, d_m\}$ $\quad d_i = f(x_i) + e_i$.

→ Asumiendo también independencia entre los datos observados tenemos

$$h_{ML} = \underset{h \in H}{\operatorname{argmax}} \prod_{i=1}^{m} P(d_i | h)$$

$$h_{ML} = \underset{h \in H}{\arg\max} \prod_{i=1}^{m} p(d_i \mid h)$$

$$= \underset{h \in H}{\arg\max} \prod_{i=1}^{m} \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{1}{2\sigma^2}(d_i - \mu)^2}$$

$$= \underset{h \in H}{\arg\max} \prod_{i=1}^{m} \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{1}{2\sigma^2}(d_i - h(x_i))^2}$$

$$= \underset{h \in H}{\arg\max} \sum_{i=1}^{m} \ln \frac{1}{\sqrt{2\pi\sigma^2}} - \frac{1}{2\sigma^2}(d_i - h(x_i))^2$$

$$= \underset{h \in H}{\arg\max} \sum_{i=1}^{m} -\frac{1}{2\sigma^2}(d_i - h(x_i))^2 = \underset{h \in H}{\arg\min} \sum_{i=1}^{m} \frac{1}{2\sigma^2}(d_i - h(x_i))^2$$

$$\shortparallel$$

$$\underset{h \in H}{\arg\min} \sum_{i=1}^{m} (d_i - h(x_i))^2$$

$$h_{ML} = \underset{h \in H}{\arg\max} \prod_{i=1}^{m} p(d_i \mid h)$$

$$= \underset{h \in H}{\arg\max} \prod_{i=1}^{m} \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{1}{2\sigma^2}(d_i - \mu)^2}$$

densidad de la dist. normal

$$= \underset{h \in H}{\arg\max} \prod_{i=1}^{m} \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{1}{2\sigma^2}(d_i - h(x_i))^2}$$

prop. de logaritmo

$$= \underset{h \in H}{\arg\max} \sum_{i=1}^{m} \ln \frac{1}{\sqrt{2\pi\sigma^2}} - \frac{1}{2\sigma^2}(d_i - h(x_i))^2$$

el primer término no depende de h

$$= \underset{h \in H}{\arg\max} \sum_{i=1}^{m} -\frac{1}{2\sigma^2}(d_i - h(x_i))^2 = \underset{h \in H}{\arg\min} \sum_{i=1}^{m} \underbrace{\frac{1}{2\sigma^2}}_{\text{constante}}(d_i - h(x_i))^2$$

$$\| $$

$$\underset{h \in H}{\arg\min} \sum_{i=1}^{m} (d_i - h(x_i))^2$$

$$h_{ML} = \underset{h\in H}{\arg\max} \prod_{i=1}^{m} p(d_i \mid h)$$

$$= \underset{h\in H}{\arg\max} \prod_{i=1}^{m} \frac{1}{\sqrt{2\pi\sigma^2}}\, e^{-\frac{1}{2\sigma^2}(d_i-\mu)^2}$$

densidad de la dist. normal

$$= \underset{h\in H}{\arg\max} \prod_{i=1}^{m} \frac{1}{\sqrt{2\pi\sigma^2}}\, e^{-\frac{1}{2\sigma^2}(d_i-h(x_i))^2}$$

prop. de logaritmo

$$= \underset{h\in H}{\arg\max} \sum_{i=1}^{m} \ln \frac{1}{\sqrt{2\pi\sigma^2}} - \frac{1}{2\sigma^2}(d_i-h(x_i))^2$$

el primer término no depende de h

$$= \underset{h\in H}{\arg\max} \sum_{i=1}^{m} -\frac{1}{2\sigma^2}(d_i-h(x_i))^2 = \underset{h\in H}{\arg\min} \sum_{i=1}^{m} \underbrace{\frac{1}{2\sigma^2}}_{\text{constante}}(d_i-h(x_i))^2$$

$$\underset{h\in H}{\arg\min} \sum_{i=1}^{m} (d_i-h(x_i))^2$$

¡Diferencia de cuadrados!

Lo anterior quiere decir que [bajo los supuestos mencionados] cualquier algoritmo que minimice el error cuadrático arrojará una hipótesis de máxima verosimilitud (máximum likelihood)

Ej: regresión

red neuronal

Pero ojo: Los supuestos incluyen ruido (ε_i) en la variable objetivo más no en las variables de entrada!!

Caso de predicción de probabilidades

Ahora consideremos el caso de una función $f : X \longrightarrow \{0,1\}$ para la cuál queremos predecir una probabilidad.

Es decir queremos aprender la función

$$f' : X \longrightarrow [0,1] \quad \text{en donde} \quad f'(x) = P(f(x)=1)$$

Sea $D = \{(x_1, d_1), \dots, (x_m, d_m)\}$ en donde $d_i = f(x) \in \{0,1\}$ queremos entonces calcular

$$P(D|h) = \prod_{i=1}^{m} P(x_i, d_i | h)$$

asumiremos además que x_i es independiente de la hipótesis h. Así;

$$\boxed{P(D|h) = \prod_{i=1}^{m} P(x_i, d_i | h) = \prod_{i=1}^{m} P(d_i | h, x_i) P(x_i)}$$

$$\boxed{P(D|h) = \prod_{i=1}^{m} P(x_i, d_i | h) = \prod_{i=1}^{m} \boxed{P(d_i | h, x_i)}\, P(x_i)}$$

???

Recordando que queremos modelar $h(x) = \hat{f}(x) = P(f(x)=1)$

es decir $P(d_i = 1 \,/\, h, x_i) = h(x_i)$ así que

$$P(d_i | h, x_i) = \begin{cases} h(x_i) & \text{si } d_i = 1 \\ 1 - h(x_i) & \text{si } d_i = 0 \end{cases}$$

escrito de otra forma

$$P(d_i | h, x_i) = h(x)^{d_i} (1 - h(x_i))^{1-d_i}$$

Luego $$\boxed{P(D|h) = \prod_{i=1}^{m} h(x)^{d_i} (1 - h(x_i))^{1-d_i} \, P(x_i)}$$

$$H_{ML} = \underset{h \in H}{\arg\max} \prod_{i=1}^{m} h(x)^{d_i} (1 - h(x_i))^{1-d_i} \, \underbrace{p(x_i)}_{\text{independiente de } h}$$

$$= \underset{h \in H}{\arg\max} \prod_{i=1}^{m} h(x)^{d_i} (1 - h(x_i))^{1-d_i}$$

$$= \underset{h \in H}{\arg\max} \sum_{i=1}^{m} d_i \ln h(x_i) + (1 - d_i) \ln (1 - h(x_i))$$

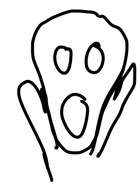
$$H_{ML} = \underset{h \in H}{\operatorname{argmax}} \prod_{i=1}^{m} h(x)^{d_i} \left(1 - h(x_i)\right)^{1-d_i} p(x_i)$$

Independiente de h

$$= \underset{h \in H}{\operatorname{argmax}} \prod_{i=1}^{m} h(x)^{d_i} \left(1 - h(x_i)\right)^{1-d_i}$$

$$= \underset{h \in H}{\operatorname{argmax}} \sum_{i=1}^{m} d_i \ln h(x_i) + (1-d_i) \ln\left(1 - h(x_i)\right)$$

¡¡Cross entropy!!



Así es como encontramos que minimizar la función de cross entropy en el caso de predicción de probabilidades puede arrojar un modelo de máxima verosimilitud

Clasificador óptimo de Bayes (Bayes optimal classifier)

Hasta ahora pretendíamos responder la pregunta

¿Cuál es la hipótesis más probable dados los datos de entrenamiento? $P(h|D)$

Ahora, preguntémonos

¿Cuál es la clasificación más probable de una nueva instancia dados los datos de entrenamiento?

¿Cuál es la clasificación más probable de una nueva instancia dados los datos de entrenamiento?

Ejemplo

	$h_i(x)$	$Pr(h/D)$
h_1	+	0.4
h_2	−	0.3
h_3	−	0.3

h_1 máximum a priori hypothesis

Dada una instancia nueva x,

¿Cuál es la etiqueta más probable para x?

Considerando todas las hipótesis, es más probable la etiqueta negativa (0.6 vs 0.4)

En general,

$$P(v_j|D) = \sum_{h_i \in H} P(v_j|h_i)\, P(h_i|D)$$

$$\boxed{\begin{array}{l} \underline{\text{Clasificación óptima de Bayes}} \\[1em] \underset{v_j \in V}{\arg\max} \displaystyle\sum_{h_i \in H} P(v_j|h_i)\, P(h_i|D) \end{array}}$$

Este clasificador resulta óptimo: Dado el mismo espacio de hipótesis H y el mismo conocimiento previo (a saber, los valores de $P(h_i|D)$), ningún clasificador podría hacerlo mejor!

El inconveniente del "Bayes optimal classiffier"....

El costo computacional!

Debe calcular probabilidades para cada $h \in H$ y para cada instancia nueva.

La alternativa?

Naive Bayes Classifier

Clasificador Naive Bayes

$$f : X \longrightarrow V \hookrightarrow \text{Conjunto finito}$$

Llamaremos $a_1, a_2, \ldots, a_n$ a los atributos de un registro X_i

Dada una nueva instancia con atributos $a_1, \ldots, a_n$, se pretende asignar la etiqueta mas probable "a priori" (MAP)

$$v_{MAP} = \underset{v_j \in V}{\operatorname{argmax}} \, P(v_j \mid a_1, a_2, \ldots, a_n)$$

Bayes

$$= \frac{P(a_1, a_2, \ldots, a_n \mid v_j) P(v_j)}{P(a_1, a_2, \ldots, a_n)}$$

denominador independiente

$$v_{MAP} = P(a_1, a_2, \ldots, a_n \mid v_j) P(v_j)$$

$$v_{MAP} = P(a_1, a_2, \ldots, a_n \mid v_j)\, P(v_j)$$

$$v_{MAP} = \underbrace{P(a_1, a_2, \ldots, a_n \mid v_j)}_{???}\, P(v_j)$$

fácil contando frecuencias

asumimos una condición más: que los a_i son condicionalmente independientes, dada la etiqueta

así, $P(a_1, a_2, \ldots, a_n \mid v_j) = \prod_i P(a_i \mid v_j)$

$$v_{NB} = \underset{v_j \in V}{\operatorname{argmax}}\; P(v_j) \prod_i P(a_i \mid v_j)$$

etiqueta según Naive Bayes

Ejercicio

Calcular la etiqueta según Naive Bayes para un día con características:

(sunny, cool, high, strong)

$$\boxed{V_{NB} = \underset{v_j \in \{yes, no\}}{\operatorname{argmax}} \; P(v_j) \prod_i P(a_i \mid v_j)}$$

Outlook	Temperature	Humidity	Wind	Play
Sunny	Hot	High	Weak	No
Sunny	Hot	High	Strong	No
Overcast	Hot	High	Weak	Yes
Rain	Mild	High	Weak	Yes
Rain	Cool	Normal	Weak	Yes
Rain	Cool	Normal	Strong	No
Overcast	Cool	Normal	Strong	Yes
Sunny	Mild	High	Weak	No
Sunny	Cool	Normal	Weak	Yes
Rain	Mild	Normal	Weak	Yes
Sunny	Mild	Normal	Strong	Yes
Overcast	Mild	High	Strong	Yes
Overcast	Hot	Normal	Weak	Yes
Rain	Mild	High	Strong	No