Aprendizaje Bayesiano

Oscar Javier Prieto Izquierdo
Raúl Casillas Díaz

Contenidos

- Introducción.
- Teorema de Bayes.
- MAP (Maximum a posteriori).
- Aprendizaje MAP.
- Clasificador bayesiano óptimo.
- Aprendizaje bayesiano naive.
- Ejemplo.
- Clasificación de textos.
- Conclusiones generales.

Razonamiento bayesiano

- Nos da un enfoque probabilístico de la inferencia.
- Está basado en asumir que las incógnitas de interés siguen distribuciones probabilísticas.
- Se puede conseguir una solución óptima por medio de estas distribuciones y datos observados.
- Nos da la posibilidad de realizar una ponderación de la posibilidad de ocurrencia de una hipótesis de manera cuantitativa.

Importancia del razonamiento bayesiano

- Los algoritmos de aprendizaje bayesiano pueden calcular probabilidades explícitas para cada hipótesis.
- También nos proporcionan un marco para estudiar otros algoritmos de aprendizaje.

Aprendizaje bayesiano

- El aprendizaje se puede ver como el proceso de encontrar la hipótesis más probable, dado un conjunto de ejemplos de entrenamiento D y un conocimiento a priori sobre la probabilidad de cada hipótesis.

Características (I)

- Cada ejemplo de entrenamiento afecta a la probabilidad de las hipótesis. Esto es más efectivo que descartar directamente las hipótesis incompatibles.
- Se puede incluir conocimiento a priori: probabilidad de cada hipótesis; y la distribución de probabilidades de los ejemplos.
- Es sencillo asociar un porcentaje de confianza a las predicciones, y combinar predicciones en base a su confianza.

Características (II)

- Una nueva instancia es clasificada como función de la predicción de múltiples hipótesis, ponderadas por sus probabilidades.
- Incluso en algunos casos en los que el uso de estos métodos se ha mostrado imposible, pueden darnos una aproximación de la solución óptima.

Dificultades

- Necesidad de un conocimiento a priori. Si no se tiene este conocimiento estas probabilidades han de ser estimadas.
- Coste computacional alto. En el caso general es lineal con el número de hipótesis candidatas.

Teorema de Bayes

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

Donde:

- *P(h)* es la probabilidad a priori de la hipótesis h.
- *P(D)* es la probabilidad de observar el conjunto de entrenamiento D.
- *P(D|h)* es la probabilidad de observar el conjunto de entrenamiento D en un universo donde se verifica la hipótesis h.
- *P(h|D)* es la probabilidad a posteriori de h, cuando se ha observado el conjunto de entrenamiento D.

Selección de hipótesis (I)

- Máximo a posteriori (MAP):
  - Se denomina así a la hipótesis más probable aplicando el teorema de Bayes.

$$h_{MAP} \equiv \arg\max_{h \in H} P(h \mid D)$$

$$= \arg\max_{h \in H} \frac{P(D \mid h) P(h)}{P(D)}$$

$$= \arg\max_{h \in H} P(D \mid h) P(h)$$

Selección de hipótesis (II)

- En algunos casos la multiplicación por *P(h)* no tiene sentido ya que las hipótesis son equiprobables:

$$h_{ML} \equiv \arg\max_{h \in H} P(D \mid h)$$

- A este resultado se le denomina **máxima verosimilitud** (maximum likelihood).

Ejemplo

- **¿Un paciente está enfermo?**

  Un test de laboratorio ha dado positivo. Cuando el paciente está enfermo el test lo detecta en un 98% de los casos. Si el paciente no tiene cáncer, el test da un 3% de falsos positivos. Sólo el 0,8% de las personas están enfermas.

P(enfermo)	= 0.008	*P(¬enfermo)*	= 0.992
P(+|enfermo)	= 0.98	*P(-|enfermo)*	= 0.02
P(+|¬enfermo)	= 0.03	*P(-|¬enfermo)*	= 0.97

- Aplicando MAP para ¬enfermo en caso de que de test positivo

  P(+|enfermo)P(enfermo) = 0.98 * 0.008 = 0.0078

  P(+|¬enfermo)P(¬enfermo) = 0.03 * 0.992 = 0.0298

  normalizando:

$$P(enfermo \mid +) = \frac{0.0078}{0.0078 + 0.0298} = 0.21$$

Aprendizaje de hipótesis por fuerza bruta (I)

- Dado un conjunto de ejemplos D y un espacio de hipótesis H.

  1. Para cada hipótesis h perteneciente a H se computa:

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

  2. Se devuelve la hipótesis con la máxima probabilidad a posteriori.

$$h_{MAP} = \arg\max_{h \in H} P(h \mid d)$$

Aprendizaje de conceptos (I)

- Supongamos el siguiente problema de aprendizaje:
  - Descripción de instancias, X, (atributos, valores)
  - Descripción de las hipótesis, H
  - Concepto objetivo c : X→{0,1}
  - Ejemplos de entrenamiento D={<x_1, c(x_1)>, <x_2, c(x_2)>, ..., <x_m, c(x_m)>}

Aprendizaje de conceptos (II)

- Consideraremos que el conjunto de ejemplos (x_i) es fijo y lo que varía son los resultados asociados ($D=\{d_1, ..., d_m\}$).
- Partamos de las siguientes hipótesis:

  1. Los datos de entrenamiento no tienen ruido $D=\{d_1, ..., d_m\}$

  2. El concepto objetivo está contenido en el espacio de hipótesis H.

  3. No tenemos conocimiento de las probabilidades a priori, por lo que consideramos cada h_i equiprobable.

Aprendizaje conceptos (III)

- Debemos tener conocimiento de las probabilidades *P(h)*, *P(D|h)*, *P(D)*

$$P(h) = \frac{1}{|H|}$$

$$P(D|h) = \begin{cases} 1 \quad si \quad d_i = h(x_i) \forall d_i \in D \\ 0 \quad en \quad otro \quad caso \end{cases}$$

Aprendizaje conceptos (IV)

- Los valores de *P(D)* son hallados mediante el teorema de la probabilidad total dado en las fórmulas básicas, asumimos que las hipótesis son excluyentes.

$$P(D) = \sum_{h_i \in H} P(D \mid h_i) P(h_i)$$

$$= \sum_{h_i \in VS_{H,D}} 1 \cdot \frac{1}{|H|} + \sum_{h_i \notin VS_{H,D}} 0 \cdot \frac{1}{|H|}$$

$$= \sum_{h_i \in VS_{H,D}} 1 \cdot \frac{1}{|H|}$$

$$= \frac{|VS_{H,D}|}{|H|}$$

Aprendizaje conceptos (V)

- Ahora, con estas probabilidades halladas, podemos aplicar el teorema de Bayes a cada hipótesis, siguiendo el planteamiento de aprendizaje por fuerza bruta:

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

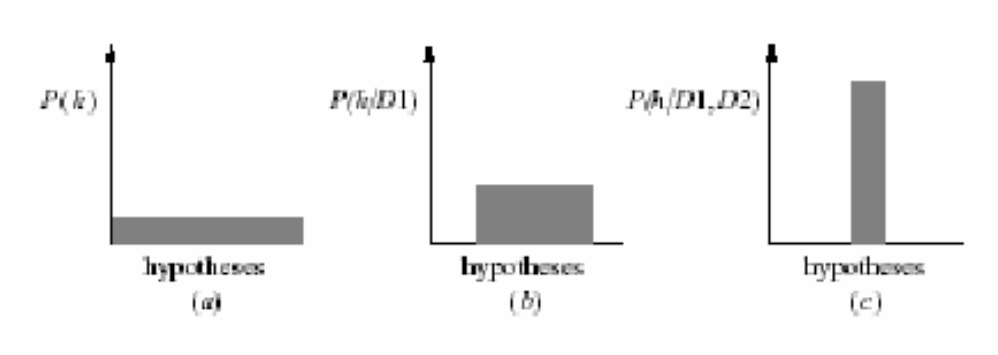
  - Si la hipótesis es inconsistente con los ejemplos de entrenamiento D

$$P(h \mid D) = \frac{0 \cdot \frac{1}{|H|}}{P(D)} = 0$$

  - Si la hipótesis es consistente con los ejemplos de entrenamiento D

$$P(h \mid D) = \frac{1 \cdot \frac{1}{|H|}}{\frac{|VS_{H,D}|}{|H|}} = \frac{1}{|VS_{H,D}|}$$

Aprendizaje conceptos (VI)



•A medida que acumulamos el conocimiento dado por el conjunto de ejemplos D, la probabilidad de las hipótesis inconsistentes se hace 0.

•Las hipótesis consistentes son equiprobables y son una solución MAP.

Hipótesis MAP y aprendizaje consistente

- Un algoritmo de aprendizaje es consistente si obtiene una hipótesis que no comete ningún error sobre los ejemplos de entrenamiento.
- Un algoritmo de aprendizaje consistente genera un hipótesis MAP si
  - Las hipótesis tienen la misma probabilidad a priori ($P(h_i)=P(h_j)$ $\forall i,j$
  - No hay ruido en los datos $\rightarrow$ ($P(D|h)=1$ si h es consistente y 0 en otro caso.

Clasificador bayesiano óptimo (I)

- ¿Cuál es la **clasificación más probable para un nuevo ejemplo**, dado el conjunto de los ejemplos de entrenamiento?

  ¿La clasificación de la hipótesis más probable, $h_{MAP}(x)$?
- Ejemplo
  - 3 hipótesis

    $P(h_1|D) = 0.4$ $P(-|h_1) = 0$ $P(+|h_1) = 1$

    $P(h_2|D) = 0.3$ $P(-|h_2) = 1$ $P(+|h_2) = 0$

    $P(h_3|D) = 0.3$ $P(-|h_3) = 1$ $P(+|h_3) = 0$
  - Vemos que ante una instancia x:

    $h_1(x) = +$ $h_2(x) = -$ $h_3(x) = -$

Clasificador bayesiano óptimo (II)

- La clasificación óptima viene para un nuevo ejemplo es el valor de entre el conjunto de valores posibles V que cumple

$$\arg\max_{v_j \in V} \sum_{h_i \in H} P(v_j \mid h_i) P(h_i \mid D)$$

- En el ejemplo anterior

$$\sum_{h_j \in H} P(+ \mid h_i) P(h_i \mid D) = 0{,}4$$

$$\sum_{h_j \in H} P(- \mid h_i) P(h_i \mid D) = 0{,}6$$

$$\arg\max_{v_j \in \{+,-\}} \sum_{h_i \in H} P(v_j \mid h_i) P(h_i \mid D) = -$$

Clasificador bayesiano óptimo (III)

- ¡¡La hipótesis que representa este clasificador puede no encontrarse dentro del espacio de hipótesis!!.
- Realmente se considera un espacio de hipótesis H, que tiene en cuenta combinaciones lineales de las hipótesis del espacio H.

Algoritmo de Gibbs

- El clasificador bayesiano óptimo proporciona los mejores resultados que se puede obtener dado un conjunto de ejemplos de entrenamiento.
  - Aplicado al aprendizaje de conceptos mediante espacio de versiones consistiría en sumar ‘votos’ para cada hipótesis ponderados por la probabilidad a posteriori de cada una.
  - Esto es muy costoso para muchas hipótesis

Algoritmo de Gibbs (II)

- Se escoge una hipótesis aleatoriamente, de acuerdo a la distribución de probabilidades a posteriori.
- Se devuelve la clasificación dada por esa hipótesis.

- Se ha demostrado que el error esperado es menor o igual que el doble del error del clasificador óptimo.
- En el espacio de versiones, si se supone una distribución uniforme de probabilidades, el algoritmo de Gibbs consistiría en tomar una hipótesis al azar.

Clasificador bayesiano naive (I)

- Uno de los mejores métodos de aprendizaje en la práctica.
- En algunos dominios comparable a redes de neuronas y árboles de decisión.

- Se puede aplicar cuando
  - Se dispone de conjuntos de entrenamiento de tamaño medio o grande
  - Los atributos que describen a los ejemplos son independientes entre sí con respecto al concepto que se pretende aprender

- Aplicado con éxito en: Diagnósticos, Clasificación de documentos.

Clasificador bayesiano naive (II)

- Cada ejemplo *x* se describe con la conjunción de los valores de sus atributos: $<a_1, a_2, \ldots a_n>$
- La función objetivo f(x) puede tomar cualquier valor de un conjunto finito V
- La clasificación viene dada por el valor de máxima probabilidad a posteriori: v_{MAP}

$$v_{MAP} = \arg\max_{vj \in V} P(v_j \mid a_1, a_2, \ldots a_n)$$

$$= \arg\max_{vj \in V} \frac{P(a_1, a_2, \ldots a_n \mid v_j) P(v_j)}{P(a_1, a_2, \ldots a_n)}$$

$$= \arg\max_{vj \in V} P(a_1, a_2, \ldots a_n \mid v_j) P(v_j)$$

Clasificador bayesiano naive (III)

- Los términos se han de estimar basándose en los ejemplos de entrenamiento.
  - $P(v_j)$ contando la frecuencia con la que ocurre cada valor v_j
  - Hay demasiados términos de la forma $P(a_1,a_2,\ldots a_n|v_j)$. Harían falta muchísimos ejemplos de entrenamiento para obtener una buena estimación.

Clasificador bayesiano naive (IV)

- La suposición del clasificador naive es que los atributos son independientes entre sí con respecto al concepto objetivo y, por lo tanto:

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

- La aproximación del clasificador bayesiano naive es:

$$v_{nb} = \arg\max_{v_j \in V} P(v_j) \prod_i P(a_i \mid v_j)$$

- Las probabilidades $P(a_i|v_j)$ resultan mucho más fácil de estimar que las $P(a_1,a_2,\ldots a_n)$

Algoritmo

Aprendizaje_Bayesiano_Naive(ejemplos)

  Para cada posible valor del resultado v_j

    Obtener estimación $P'(v_j)$ de la probabilidad $P(v_j)$

      Para cada valor a_i de cada atributo a

        Obtener una estimación $P'(a_i | v_j)$ de la probabilidad $P'(a_i | v_j)$

Clasificar_instancia(x)

  devolver $v_{nb} = \arg\max_{v_j \in V} P(v_j) \prod_i P(a_i | v_j)$

Ejemplo

Day	Outlook	Temperature	Humidity	Wind	PlayTennis
D1	Sunny	Hot	High	Weak	No
D2	Sunny	Hot	High	Strong	No
D3	Overcast	Hot	High	Weak	Yes
D4	Rain	Mild	High	Weak	Yes
D5	Rain	Cool	Normal	Weak	Yes
D6	Rain	Cool	Normal	Strong	No
D7	Overcast	Cool	Normal	Strong	Yes
D8	Sunny	Mild	High	Weak	No
D10	Rain	Mild	Normal	Weak	Yes
D11	Sunny	Mild	Normal	Strong	Yes
D12	Overcast	Mild	High	Strong	Yes
D13	Overcast	Hot	Normal	Weak	Yes
D14	Rain	Mild	High	Strong	No

Ejemplo

Estimación de probabilidades:

P’(PlayTennis=yes) = 9/14 = 0,64 P’(PlayTennis=no) = 5/14 = 0,36

P’(Outlook=sunny | PlayTennis=yes) = 1/9 = 0,11
P’(Outlook=sunny | PlayTennis=no) = 3/5 = 0,6
P’(Temperature=cool | PlayTennis=yes) = 3/9 = 0,33
P’(Temperature=cool | PlayTennis=no) = 1/5 = 0,2
P’(Humidity=high | PlayTennis=yes) = 3/9 = 0,33
P’(Humidity=high | PlayTennis=no) = 4/5 = 0,8
P’(wind=strong | PlayTennis=yes) = 3/9 = 0,33
P’(wind=strong | PlayTennis=no) = 3/5 = 0,6
...

Ejemplo

- Ejemplo a clasificar:

<Outlook = sunny, Temperature = cool, Humidity = high, Wind = strong>

$$v_{nb} = \arg\max_{v_j \in V} P(v_j) \prod_i P(a_i \mid v_j)$$

$$= \arg\max_{v_j \in V} P(v_j) P'(Outlook = sunny \mid v_j) P'(Temperature = cool \mid v_j)$$

$$P'(Humidity = high \mid v_j) P'(Wind = strong \mid v_j)$$

P’(yes)P’(sunny|yes)P’(cool|yes)P’(high|yes)P’(strong|yes) = 0,0053 (0,205)

P’(no) P’(sunny|no)P’(cool|no)P’(high|no)P’(strong|no) = 0,0206 (0,795)

Estimación de probabilidades

- Problema con las estimaciones $P(a_i|v_j) = 0$
  - Este témino dominará el clasificador al tener que multiplicar el resto de probabilidades por 0.
  - *Estimación-m:*

$$P'(a_i \mid v_j) = \frac{n_c + mp}{n + m}$$

    - n: número de ejemplos del entrenamiento con valor v_j
    - n_c: fracción de n con valor a_i para el atributo a
    - p: estimación a priori de $p(a_i|v_j)$
    - m: peso de la estimación a priori

Estimación de probabilidades (II)

- Si m=0 se tendría la estimación por defecto.
- Si no, la estimación observada (n_c/n) y el conocimiento previo (p) son combinados de acuerdo al peso m.

- Sin información adicional, la probabilidad a priori se puede obtener suponiendo probabilidad uniforme:

$$p = \frac{1}{k}$$

Siendo *k* el número de valores distintos para el atributo a.

Ejemplos de aplicaciones reales

- Diagnóstico
- Clasificación de textos

Reducción de la dimensionalidad (I)

- Es necesaria por la alta dimensión del espacio de términos existentes en un texto
- Selección de patrones más representativos dentro de un texto.
- Se pueden aplicar diferentes técnicas
- Pretende incrementar la eficiencia sin disminuir la precisión

Reducción de la dimensionalidad (II)

- Se pueden eliminar palabras características consideradas de poco valor
- Se puede reparametrizar el texto, sustituyendo algunas palabras por otras que las representen (lematización, sinonimia, …)

Reducción de la dimensionalidad (III)

- Diccionario de términos con palabras relevantes
- Diccionario de términos no relevantes
- Filtros:
  - Palabras
  - Frases
  - Conjuntos de palabras

Clasificación de textos (I)

- Aplicación que ilustra la importancia práctica de los métodos de aprendizaje bayesiano.
- Las instancias son documentos de texto.
  - Espacio de instancias X: Todos los posibles documentos de texto.
- Concepto a aprender:
  - "Artículos que me interesan"
  - "Páginas web sobre un determinado tema".
  - …
- Función objetivo: f: documento → $\{v_1, v_2, ...\}$

  Ej. clasificar documentos como interesantes o no para una persona.

Clasificación textos (II)

- Para aplicar el clasificador bayesiano naive:
  1. Cómo representar un documento de texto cualquiera en términos de valores de atributos.
  2. Cómo estimar las probabilidades requeridas por el clasificador bayesiano naive.

Clasificación textos (III)

1. Representación del documento

- Un vector con tantos atributos como palabras tiene el documento, donde el valor del atributo *i* es la palabra que hay en la posición *i*.

$$v_{nb} = \arg\max_{v_j \in \{+,-\}} P(v_j) \prod_{i=1}^{long(doc)} P(a_i = w_k \mid v_j)$$

W_k es la k-ésima palabra del vocabulario utilizado.

- La suposición del aprendizaje bayesiano $P(a_1, a_2, ...a_n \mid v_j) = \prod_i P(a_i \mid v_j)$ no se cumple. La probabilidad de una palabra en una posición depende de las palabras en el resto de posiciones.

Clasificación textos (IV)

- Probabilidades a estimar:
  - P(+) y P(-) : Fracciones de cada tipo obtenida durante el aprendizaje.
  - $P(a_i=w_k|+)$, $P(a_i=w_k|-)$ para todas las palabras del diccionario en cada una de las posibles posiciones.
    - Para simplificar: Suponemos que la probabilidad de encontrar una determinada palabra es independiente de la posición considerada (atributos independientes e igualmente distribuidos):

      $P(a_1=w_k|v_j)$, = $P(a_2=w_k|v_j)$ = … = $\mathbf{P(w_k|v_j)}$

Clasificación textos (V)

- Mejora las estimaciones de las probabilidades en casos con un reducido conjunto de datos de entrenamiento.
- Para evitar el problema de que alguna estimación de probabilidad se haga cero, se utiliza la *estimación-m:*

$$P(w_k \mid v_j) = \frac{n_i + 1}{n + \mid vocabulario \mid}$$

n es el número de palabras de todos los documentos con valor vj

*n*k es el número de veces que aparece la palabra wk entre las *n* palabras

|*vocabulario*| es el número de palabras distintas que aparecen en los documentos de entrenamiento

Clasificación textos (VI)

- Problema: clasificar artículos de grupos de noticias
  - 20 grupos de noticias
  - 1000 artículos de cada grupo
  - Dos tercios de los documentos para entrenamiento y un tercio para
  - Validación
- Solución: clasificador bayesiano naive eliminando del vocabulario
  - las 100 palabras más frecuentes (artículos, preposiciones, ...)
  - cualquier palabra que apareciese menos de tres veces
- Resultado: clasificación correcta en el 89% de los casos

Conclusiones generales (I)

- Firme fundamento matemático.
  - Marco para estudiar otros algoritmos de aprendizaje bajo el mismo enfoque.
- Esquema probabilístico que asocia una única descripción a cada clase:
  - Probabilidad a priori. Conocimiento previo.
  - Probabilidades condicionadas (distribución de probabilidad para cada atributo). Información extraída de la distribución de los casos de entrenamiento.

Conclusiones generales (II)

- La mayoría de versiones de los clasificadores bayesianos asumen independencia de atributos.
  - Los resultados prácticos parecen demostrar su validez extendiendo su comportamiento con independencia de esta suposición.

- Este aprendizaje permite decidir la influencia relativa del conocimiento previo en las observaciones.

Conclusiones generales (III)

- Las suposiciones subyacentes de caracterizar cada clase con una única descripción y de independencia de los atributos le han dado una reputación mala.
- Coste computacional alto y necesidad de conocimiento a priori con el cual no contamos en la mayoría de los casos.

- Ha demostrado en dominios naturales, donde no se cumplen las suposiciones, comportarse con resultados comparables a aquellos obtenidos por métodos más sofisticados.
- Tienen la habilidad de discriminar atributos relevantes de otros irrelevantes.

Bibliografía

- “*Machine Learning*”, Capítulo 6, Tom M. Mitchell, McGraw-Hill International Editions.