Clasificador Bayes Ingenuo (Naïve Bayes)

I. Derivación

A. Clasificación Bayesiana: Maximum A-Posteriori (MAP).

Asuma que tenemos un vector de datos $\mathbf{x} = (x_1, x_2, ..., x_d)^T$, donde d es la cantidad de features (descriptores). De acuerdo al teorema de Bayes, podemos calcular la probabilidad de que el vector de datos $\mathbf{x}$ pertenezca a la clase $C_K \in \{1, ..., K\}$ de acuerdo con la expresión

$$\begin{aligned} P(C_K|x_1, x_2, ..., x_d) &= P(C_K|\mathbf{x}) \\ &= \frac{P(C_K)P(\mathbf{x}|C_K)}{P(\mathbf{x})} \end{aligned} \tag{1}$$

Entonces, la *regla de decisión MAP* para clasificación es aquella que maximiza la expresión (1). Más formalmente

$$MAP : \underset{C_K}{\text{argmax}}\ P(C_K|\mathbf{x}) = \underset{C_K}{\text{argmax}}\ P(C_K)P(\mathbf{x}|C_K), \tag{2}$$

debido a que el denominador $P(\mathbf{x})$ de la expresión (1) es una constante y no afecta la operación de maximización.

B. Bayes Ingenuo.

$$\begin{aligned} MAP :& \underset{C_K}{\text{argmax}}\ P(C_K)P(\mathbf{x}|C_K) \\ & \underset{C_K}{\text{argmax}}\ P(C_K, \mathbf{x}) \\ & \underset{C_K}{\text{argmax}}\ P(C_K, x_1, x_2, ..., x_d) \qquad \text{Por regla de la cadena:} \\ & \underset{C_K}{\text{argmax}}\ P(x_1|x_2, ..., x_d, C_K)P(x_2, ..., x_d, C_K) \\ & \underset{C_K}{\text{argmax}}\ P(x_1|x_2, ..., x_d, C_K)P(x_2|x_3, ..., x_d, C_K)\ ...\ P(x_{d-1}|x_d, C_K)P(x_d|C_K)P(C_K). \end{aligned}$$

En Bayes ingenuo adoptamos la asunción que la probabilidad de cada feature es condicionalmente independiente de cualquier otro, dada la etiqueta de la clase (i.e., $P(x_i|x_{i+1}, ..., x_d, C_K) = P(x_i|C_K)$. Esta es una asunción *ingenua* porque no conocemos realmente las probabilidades conjuntas y/o condicionales entre los features, pero *conveniente* para reducir la complejidad de la estimación de estas probabilidades. Interesantemente, Naïve Bayes ha logrado buen desempeño en la práctica. Al implementar esta asunción:

$$\begin{aligned} \underset{\text{Naïve}}{MAP} :& \underset{C_K}{\text{argmax}}\ P(C_K|\mathbf{x}) \\ & \underset{C_K}{\text{argmax}}\ P(x_1|C_K)P(x_2|C_K)\ ...\ P(x_d|C_K)P(C_K). \\ \underset{\text{Naïve}}{MAP} :& \underset{C_K}{\text{argmax}}\ P(C_K)\prod_{i=1}^{d} P(x_i|C_K). \end{aligned} \tag{3}$$

Nótese que en la ecuación (3), los $P(C_K)$ son los llamados *priors* de las clases y los $P(x_i|C_K)$ son las probabilidades de los valores de los features x_i dentro de cada clase C_K.

II. Sumario

A. Función de Desempeño (Métrica Exactitud (Accuracy)).

$$\frac{1}{N}\sum_{t=1}^{N}\mathrm{I}(g(\mathbf{x}^t|\theta)=r^t) \tag{4}$$

donde $x^t \in \mathcal{X}$ son las muestras y $r^t \in \mathcal{X}$, son las etiquetas de las clases correspondientes, $N = |\mathcal{X}|$, I$(\cdot)$ es una función indicadora que resulta en 1 si su argumento es verdadero y 0 de otro modo, y la función $g(\mathbf{x}^t|\theta)$ especificada por (6) es nuestro estimador, que depende de los parámetros particulares del modelo $\theta : \{P(x_1|C_1), P(x_2|C_1),\ \ldots\ , P(x_d|C_1),\ \ldots\ , P(x_d|C_K), P(C_1),\ \ldots\ , P(C_K)\}$.

B. Aprendizaje.

Al trabajar con features de valores discretos, estimamos las probabilidades necesarias computando la frecuencia relativa de los datos.

Cuando trabajamos con fatures de valores continuos, una asunción típica es que los valores continuos asociados con cada clase poseen una distribución Gaussiana. De modo que, primeramente segmentamos los datos por clase y entonces computamos la media μ_i^K y desviación estándar σ_i^K de los valores en x_i asociados con la clase C_K para calcular las probabilidades como sigue:

$$P(x_i|C_K) = \frac{1}{\sqrt{2\pi}\sigma_i^K}\exp\big(\frac{-(x_i-\mu_i^K)^2}{2(\sigma_i^K)^2}\big) \tag{5}$$

Otra técnica común para manejar features de valores continuos es usar la técnica de *binning* para discretizar los valores de los features y obtener un grupo de valores con distribución Bernoulli.

C. Ejecución.

$$g(\mathbf{x}^t|\theta) = \underset{C_K}{\text{argmax}}\ P(C_K)\prod_{i=1}^{d}P(x_i^t|C_K). \tag{6}$$