Aprendizaje supervisado y los Clasificadores Paramétricos Bayesiano

27 MARZO

Pattern Recognition
María Elena Cruz Meza

Aprendizaje supervisado y los “Clasificadores Paramétricos Bayesianos”

Introducción

Iniciaremos con métodos basados en métricas (normas), donde interesa que manejes los tres tipos de normas básicos, los cuáles son:

- City – Block
- Euclidiana
- Infinito

Esto se debe a que, al diseñar un clasificador para una distribución normal o un clasificador paramétrico, se involucra una norma o métrica. Por tanto, es indispensable conocer los detalles de cómo diseñar un clasificador basado en la medición de la distancia entre un punto a otro, así como comprobar que la discriminación bajo este esquema, al introducir un punto desconocido y discriminarlo respecto a un punto tomado arbitrariamente como representante de una clase, comparando con la decisión de haber tomado el centroide de la clase (el punto medio del conjunto de patrones de la clase en cuestión), nos permite identificar la ventaja de elegir el proceso de éste último proceso.

Una vez que has comprendido lo anterior (diseñar el tipo de clasificadores basados en una norma y su funcionamiento), ahora deberá diseñarse un clasificador Bayesiano haciendo uso de una norma, notando que infiere este proceso, así como la importancia de la elección adecuada de la norma y de los parámetros conocidos ya que este modelo tiene distintas variaciones que dependen de los parámetros conocidos. En consecuencia, debemos poder identificar que con este modelo, el tipo de aprendizaje supervisado provee dos formas de conocer la verosimilitud entre las clases: haciendo la aproximación paramétrica o bien, calculando la aproximación no paramétrica.

Clasificadores basados en Métricas

1. Problemas en el Diseño de un Sistema de Reconocimiento de Patrones

Si conociéramos totalmente las características estadísticas del modelo en el proceso de generación de los patrones, esto es si supiéramos que $P(\omega i), p(x/\omega i)\ \forall\ i$; entonces podríamos diseñar el sistema de reconocimiento de patrones (SRP) optimo mediante aplicación directa de la teoría de decisión de Bayes. Sin embargo, en la práctica surgen los siguientes problemas:

- El modelo no se puede conocer totalmente y/o
- La complejidad del SRP a diseñar está restringida por consideraciones de coste (hardware, tiempo)

Usualmente la base de conocimiento disponible para el diseño de un SRP es un conjunto de entrenamiento constituido por observaciones, ya sea etiquetadas o no. Además de otras cuestiones, debemos considerar el modelo del tipo de aprendizaje con el clasificador adecuado para resolver el problema. En esta práctica estudiaremos el aprendizaje supervisado.
En el primer caso asumimos que para cada patrón o vector de observaciones $x_i\ i = 1, 2, \cdots, N$ en el conjunto de entrenamiento (CE) o conjunto de muestras de aprendizaje (CMA), un experto asigna una etiqueta con la clase correcta y_i. Por lo que al diseño de un sistema basado en un conjunto de datos clasificados de antemano se le conoce como aprendizaje supervisado.
Cuando no se dispone de conocimiento experto sobre el conjunto de datos, o si el etiquetado de los patrones de entrenamiento es impracticable por razones prácticas; entonces el problema de diseño implica la necesidad de una primera etapa de análisis de los datos. Este proceso primario de análisis se conoce como una etapa de aprendizaje no supervisado.

Dado lo anterior, tenemos que para el caso general, dado un conjunto de entrenamientos el diseño del SRP implica el desarrollo de tres procesos:

I- Inferencia del modelo a partir de los datos (aprendizaje).
II- Desarrollo de reglas de decisión prácticas.
III- Simulación y evaluación del rendimiento del sistema.

A continuación, se provee la información relevante y necesaria para el desarrollo de esta práctica.

2. Aprendizaje supervisado

La estimación en un modelo de aprendizaje supervisado está basada en un conjunto de entrenamiento y sus prototipos, es decir, para el CMA de los cuales conozco la clase a la que pertenecen se puede optar como primer parámetro a la media o centroide de la clase, el cual será conocido como el representante de la clase. En este modelo podemos encontrarnos con dos posibilidades para conocer la verosimilitud entre las clases:

* **Aproximación paramétrica:** Conozco la estructura estadística de las clases, funciones de densidad de probabilidad conocida estimo parámetros que las determinan.

* **Aproximación no paramétrica:** Estimo el valor de la función densidad o directamente la probabilidad a posteriori de que un x pertenezca a la clase w_j a partir de la información proporcionada por el conjunto de prototipos.

Decimos que una variable aleatoria X tiene densidad f, si

$$P\{a < X < b\} = \int_a^b f(x)\,dx \;\forall a < b$$

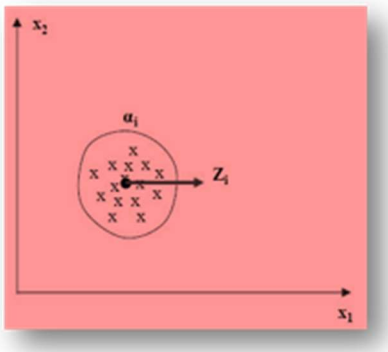
Estimar una densidad es reconstruir dicha función f a partir de un conjunto de variables X1, . . . Xn con la misma distribución que X. Entonces para los casos de:

1. Estimación paramétrica: Si conocemos la distribución a la cual pertenece f, (por ejemplo, gaussiana), entonces basta estimar los parámetros para tener una estimación de f.
2. Estimación no paramétrica: Se usa cuando no queremos asumir hipótesis sobre la distribución de nuestra muestra.

Para esta práctica requerimos conocimientos previos del uso de la métrica Euclidiana o de la métrica de Mahalanobis, por lo que primero, es necesario mostrar la habilidad y conocimiento para diseñar un clasificador basado en una norma o métrica, así como el manejo de los parámetros que en él se involucran.

2.1 Clasificadores basados en métricas:

Supongamos que existen M clases (C_1, C_2,..., C_M) con sus respectivos prototipos (Z_1,Z_2,...,Z_M) y considerando un espacio bidimensional, con las n muestras o patrones como una nube de puntos, un parámetro simple, intuitivo dada la naturaleza de este evento, es obtener la media como el centroide de esta nube, y será el vector prototipo al que llamaremos "representante de la clase" (figura 1).



Una clase $C_i=(X_i1,X_i2,...,X_iP)$ formada por P elementos, esta representada por un único vector prototipo que será su media ponderada:

$$Z_i = \frac{1}{P}\sum_{j=1}^{P} X_{ij}$$

Figura 1. Representación gráfica del prototipo de la clase α_1 y la obtención de prototipo o centroide (vector representante de la clase).

Recordemos que la forma general de una norma, basándonos en las métricas de Minkowski es:

Forma general

$d_r(x,y) = \left[\sum_{i=1}^{n}|x_i - y_i|^r\right]^{1/r}$, donde r es un número entero positivo. Dependiendo del valor de r, será la forma en cómo se comporte la métrica.

a) Cuando r=1, también conocida como City Block:

$$d_1(x,y) = \left[\sum_{i=1}^{n}|x_i - y_i|^1\right]^{1/1} = \sum_{i=1}^{n}((x_i - y_i)$$

b) Cuando r=2, se le denomina Euclidiana:

$$\boldsymbol{d_2(x,y) = \left[\sum_{i=1}^{n}|x_i - y_i|^2\right]^{1/2} = \sqrt{\sum_{i=1}^{n}|x_i - y_i|^2}}$$

c) Cuando $r = \infty$, se le conoce como norma Infinito:

$$\boldsymbol{d_\infty(x,y) = lim \to 0 \left[\sum_{i=1}^{n}|x_i - y_i|^r\right]^{1/r} = \max |x_i - y_i|^r}$$

Veamos, el ejemplo siguiente: *Sean los prototipos { Z_1, Z_2 y Z_3 }, y tomemos un vector desconocido X? a clasificar.*

El reconocedor por distancia, para cualquier métrica o norma elegida, dependiendo del valor de *r* (city block, euclídea o infinito), asociará el vector X desconocido a la clase cuyo prototipo esté más cerca o distancia sea menor. Si tomamos r=2 (la distancia euclídea o d_E, para este caso), como la norma con la que deseamos discriminar, el clasificador etiquetará al patrón desconocido X a la clase cuya distancia euclídea sea menor. En la figura 2 para clasificar un patrón desconocido se obtiene la d_E de este a cada uno de los prototipos, y dado que el clasificador etiqueta a un patrón desconocido como perteneciente a la clase cuya distancia es menor, en este caso, lo clasificará a la clase C_1 ya que la d_E a Z_1 es la menor de todas.

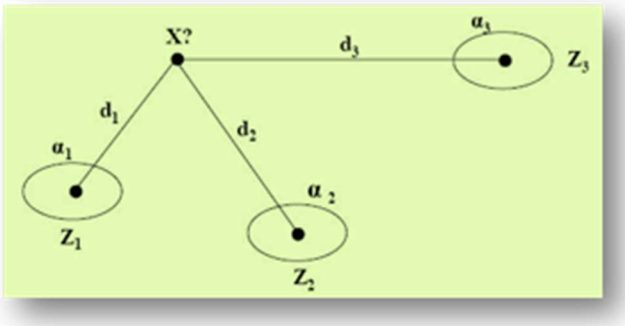


Figura 2. Representación gráfica de la d_E de un patrón desconocido a cada prototipo del sistema.

Algoritmo basado en la distancia *r*

2. Se elige una muestra de patrones clasificada de antemano con n clases {C_1, C_2,.., C_n} y la métrica d_r, donde, dependiendo del valor de r, manipularemos el tipo de norma: City Block, Euclídea o Infinito.

   a) Cityblock: en este caso, r=1

$$d_1(x,y) = \sum_{i=1}^{n}((x_i - y_i)$$

   b) Euclidiana: en este caso, r=2

$$\boldsymbol{d_2(x,y) = \sqrt{\sum_{i=1}^{n}|x_i - y_i|^2}}$$

   c) Infinito: con r= ∞

$$\boldsymbol{d_\infty(x,y) = \max\ |x_i - y_i|^r}$$

3. Con base en la muestra y para cada clase Ci, calcular el patrón representante

$$\boldsymbol{Z_i = \frac{1}{P}\sum_{j=1}^{P} X_{ij}}$$

   donde *P* es el número de elementos en la muestra que pertenece a C_i (vector medio).

4. En el momento de clasificar (recuperación), el patrón *x* será clasificado en la clase *i* si cumple lo siguiente:

$$\boldsymbol{di(X?, Zi) < dj(X?, Zj)}$$

Clasificador paramétrico Bayesiano

Las variaciones para este método se listan a continuación (repasar el capítulo 10 de Pajares y Martinsanz, del documento en pdf compartido):

- Caso normal multivariable: Clasificador Bayesiano con la media desconocida. Si consideramos a los patrones o n muestras como una nube de puntos, entonces la estima para la máxima verosimiltud es tomando la media de la distribución, exactamente la media aritmética de las muestras, es decir, la media simple. Recordemos que, en los clasificadores basado en una norma o métrica, este parámetro se considera como el centroide de esta nube, por lo que aquí también lo llamaremos "representante de la clase". Así, en esta variante, suponemos que las muestras siguen una distribución normal

con media m y matriz de covarianza C, y desde luego, ahora ya conocemos la media.

$$m_i = \frac{1}{n}\sum_{i=1}^{n} x_i$$

- Caso normal multivariable: Media y matriz de covarianza conocidas. Es el caso general y más común de una normal multivariable, donde ni la media m ni la matriz de covarianza C son conocidas. Siendo estos los componentes de vector de parámetros w={w1, w2}. Su análisis respecto al caso multivariable es similar, pero en este caso las estimas de máxima verosimilitud para los parámetros m y C son:

$$\boldsymbol{m_i = \frac{1}{n}\sum_{i=1}^{n} x_i \qquad C_i \frac{1}{n-1}\sum_{i=1}^{n}(x_k - m_i)\, C_2^{-1}\,(x_k - m_i)^t}$$

Notemos que la expresión de la estima de la máxima verosimilitud para el vector media es precisamente la media simple. Por lo tanto, la estima de la máxima verosimilitud para la matriz de covarianza es la media aritmética de las n matrices $(x_i - m_i)\,(x_i - m_i)^t$. Ya que la verdadera matriz de covarianza es el valor esperado de la matriz $(x_i - m_i)\,(x_i - m_i)^t$. Obteniéndose un resultado satisfactorio.

Dado lo anterior, podemos diseñar un clasificador bayesiano sabiendo que un patrón desconocido xk , puede clasificarse de acuerdo con la forma.

$$p(y=c_i/x) = \frac{p(x/y=c_j)P(y=c_j)}{p(x)}$$

Donde:

$$\boldsymbol{p(x) = \sum_{j=1}^{c} p\,(x\ /y\ =\ c_j)P(y\ =\ c_j)}$$

Además, por la regla de decisión establecida como:

$$\boldsymbol{x\ \in C_i\ sii\ p(y\ =c_j\ /\ x) >\ p(y\ =c_j\ /x), \forall\ i\ \neq j,\ i,j = 1,\ 2,\ \ldots,\ c}$$

Notemos que p(x) no aporta nada a la decisión, por lo que optaremos por una forma alternativa para clasificar al vector:

$$\boldsymbol{x\ \in C_i\ sii\ p(\,x\,/\ y = c_i)p(y\ =c_i) > p(\ \ x\ /\ y = c_j)p(y\ =c_j),}$$
$$\boldsymbol{\forall\ i\ \neq j,\ i,j = 1,\ 2,\ \ldots,\ c}$$

Normalmente las distribuciones de densidad de probabilidad se eligen Normales o Gaussianas, así, debemos calcular la distancias a cada clase, en este caso, **dado que las matrices de covarianzas son distintas (debemos verificar la verosimilitud entre las clases).**

- Para el caso del diseño de una fD para clases con distribución Gaussiana, usando la distancia de Mahalanobis:

$$\boldsymbol{d_M^2\,(x_k - m_i) = (x_k - m_i)\, C_2^{-1}\,(x_k - m_i)^t}$$

$$x \in C_i \; sii \; d_M^2 (x_k - m_i) < d_M^2 (x_k - m_j),$$
$$\forall \; i \neq j, \; i,j = 1, 2, \ldots, c$$

Con *{mi ,mj }* y *{Ci, Cj }* que corresponden a la Media y la Matriz de Covarianza estimada para cada clase, p*ara toda i distinta de j.*

- **Para el caso del diseño de una fD para clases con distribución Normal, usando la distancia Euclidiana:**

$$d_E = \|X - Z_i\| = \sqrt{(X - Z_i)^T - (X - Z_i)} = \sqrt{\sum_{j=1}^{N} (X_j - Z_{ij})^2}$$

Por lo que, el clasificador tomará la menor distancia encontrada: $di(x) = d_E(X?, \mu_i)$

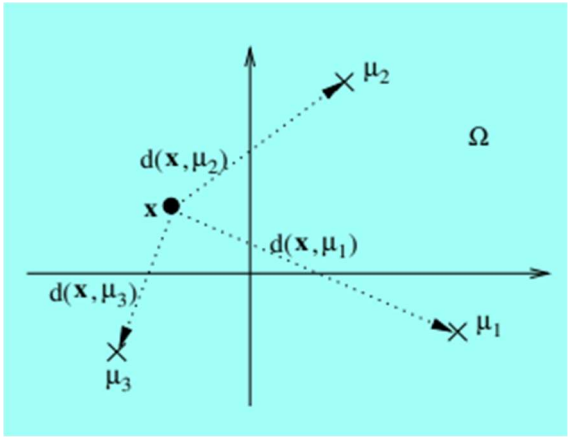


Figura 4. Regla de asignación por centroide más cercano.

Algoritmo Bayesiano para una Distribución Gaussiana

1. Se elige una muestra de patrones clasificada de antemano con n clases {c1 , c2 ,.., cn } y la métrica dM (distancia de Mahalanobis).
2. Con base en la muestra y para cada clase Ci estimar los parámetros: Media para cada clase {mi , mj }, así como las matrices de covarianza de cada clase {Ci , Cj }

Media: $m_i = \frac{1}{n} \sum_{i=1}^{n} x_i$

Matriz de covarianzas: $C_i = \frac{1}{n-1} \sum_{i=1}^{n} (x_k - m_i) \, C_2^{-1} \, (x_k - m_i)^t$

Siempre i es distinto de j, y mi es el vector medio o representante de la clase

3. *Generar funciones discriminantes dM(i,j)(x) para cada par de clases ci y cj de forma que:*

$$\boldsymbol{d_M^2\,(x_k\,-\,m_i) = (x_k\,-\,m_i)\,C_2^{-1}\,(x_k\,-\,m_i)^t}$$

4. *En el momento de clasificar (recuperación), el patrón x será clasificado en la clase ci si cumple lo siguiente:*

$$\boldsymbol{x\,\in C_i\,sii\;\; d_M^2\,(x_k\,-\,m_i) < d_M^2\,(x_k\,-\,m_j)\,,}$$
$$\boldsymbol{\forall\, i\,\neq j,\; i,j = 1,\; 2,\; \ldots,\; c}$$

Algoritmo Bayesiano para una Distribución Normal

1. *Se elige una muestra de patrones clasificada de antemano con n clases* $\{C_1, C_2, .., C_n\}$ *y la métrica dE, r=2.*
2. *Con base en la muestra y para cada clase* C_i*, calcular el patrón representante*

$$Z_i = \frac{1}{P}\sum_{j=1}^{P} x_{ij}$$

donde P es el número de elementos o patrones en la muestra que pertenece a Ci y Zi es el vector medio o patrón representante de la clase Ci.

3. *Generar funciones discriminantes* dij*(x) para cada par de clases* Ci,Cj*, de forma que:*

$$d_{ij}(x) = \left(z_i\,-\,z_j\right)^t x - \frac{1}{2}\left[\left(z_i\,-\,z_j\right)^t\left(z_i + z_j\right)\right], \text{con } x = (X1, X2)$$

5. *En el momento de clasificar (recuperación), el patrón x será clasificado en la clase i si cumple lo siguiente:*

$$\boldsymbol{\forall\, j,\, j\neq i\,,\, si\, \mathrm{dij}(x) \geq 0}$$

Fuentes

- Gonzálo Pajares Martinsansz & Jesús M. de la Cruz García. Visión por compuador. Ed. Alfaomega Ra-MA. Edición. Primera Edición. Abril 2008
- Silverman, B.W. (1986) Density Estimation for Statistics and Data Analysis. Chapman and Hall, London.
- Wand, M. P. and Jones, M. C. (1995).Kernel Smoothing. Chapman and Hall, London.
- Bowman, A.W. and Azzalini, A. (1997). Applied Smoothing Techniques for Data Analysis: the Kernel Approach with S-Plus Illustrations. Oxford University Press, Oxford