Reconocimiento de Patrones

Ingeniería en Informática

Examen 29 de enero de 2018

Hora: 09:30 horas

PROBLEMA 1 (5 puntos)

Diseñar un clasificador 1D calculando densidades de probabilidad estimadas mediante el método de Parzen con ventanas gaussianas. Para ello:

A) Realizar una función que genere un conjunto **aleatorio** y **desordenado** de **1000 datos en total**, compuesto por dos clases normalmente distribuidas. El 55% de ellos (clase 1) con media 0 y varianza 4, y el resto (clase 2) con media 2 y varianza 4.
B) Indique **de forma razonada** si podemos asegurar que el valor de la frontera principal entre ambas clases es menor/igual/mayor que un cierto valor, y cuál es éste.
C) **Realizar una función** que calcule a partir de los datos generados por la función descrita en A) las densidades de cada clase (d1 y d2) por el método de Parzen. Para ello:
   a. Obtener d1, el valor de la densidad de la clase 1 en el intervalo [-10,10] para puntos equiespaciados, y separados entre sí un valor de 1e-2
   b. Obtener d2, el valor de la densidad de la clase 2 en el intervalo [-10,10] para puntos equiespaciados, y separados entre sí un valor de 1e-2
D) **Realizar una función** que clasifique un dato real cualquiera (x) en una de las dos clases (1 ó 2)

Para calcular la densidad usando ventanas de Parzen utilice gaussianas con una desv. típica igual a 0.5

PROBLEMA 2 (5 puntos)

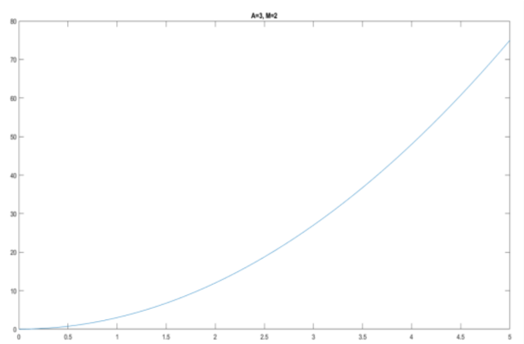
Un **ajuste potencial** se define por la fórmula

$$y = A \cdot x^M$$

A) Explicar **de forma razonada** cómo reducir el ajuste a un problema lineal.

B) Realizar la función `potfit.m` que calcule los parámetros de la función de ajuste a partir de un conjunto de valores

C) Realizar la función `potval.m` que utilice el modelo obtenido para obtener la y estimada para un valor dado de x



COMENTARIOS GENERALES

El lenguaje de codificación será `Octave/Matlab`

Se podrá hacer uso de la librería `pattern`

Se valorará la adecuada elección de los parámetros de entrada y salida de cada función.

PROBLEMA 1

A)

```
function [x,y]=genera_datos()
  x=[randn(1,550)*2 2+randn(1,450)*2];   % Genero la x
  y=[zeros(1,550) ones(1,450)]+1;        % Genero la y
   [x,y]=shuffle(x,y);                   % Las desordeno
end
```

B)

Dado que ambas distribuciones tienen la misma varianza, si las probabilidades a priori fuesen iguales, la frontera estaría en el punto medio entre las medias, es decir, en el punto 1.

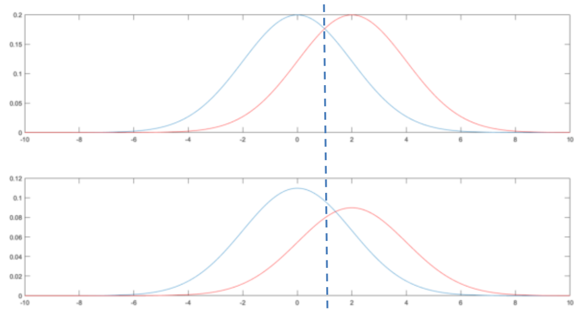
Como la probabilidad a priori de la clase 1 es mayor, esto hace que la probabilidad a posteriori también lo sea, ya que P(wi) es un factor en el numerador y P(X) es el mismo para ambas clases.

$$P(w_i \mid X) = \frac{P(X \mid w_i) \cdot P(w_i)}{P(X)}$$

Esto hace que la densidad a posteriori P(wi|X) sea mayor para la clase 1.

Por tanto, el punto pedido es 1 y la frontera estará a la derecha de este valor.

En la gráfica se compara el caso de probabilidades a priori iguales, con el caso tras aplicar las probabilidades a priori del problema(0.55 y 0.45), donde se ve claramente este efecto:



El valor del punto de corte se puede calcular de forma exacta, aunque en el examen la respuesta anterior se ha dado por buena.

El valor de la densidad 1 se calcula de la siguiente forma:

$$d1(x) = 0.55 \frac{1}{\sqrt{2\pi\sigma_1^{\,2}}} e^{-\frac{(x-\mu_1)^2}{2\sigma_1^{\,2}}}$$

La densidad 2 se calcula de forma similar. El punto exacto de la frontera se obtiene igualando d1(x)=d2(x). Sacando logaritmos se llega a que:

$$\log(0.55) - 0.5 * \log(2\pi\sigma_1^{\,2}) - \frac{(x-\mu_1)^2}{2\sigma_1^{\,2}} = \log(0.45) - 0.5 * \log(2\pi\sigma_2^{\,2}) - \frac{(x-\mu_2)^2}{2\sigma_2^{\,2}}$$

Como las desv. típicas son iguales:

$$\log(0.55) - \frac{x^2}{8} = \log(0.45) - \frac{(x-2)^2}{8}$$

$$8\log(0.55)-x^2=8\log(0.45)-(x^2-4x+4)$$
$$8\log(0.55)=8\log(0.45)+4x-4$$
$$x=\frac{8\log(0.55)-8\log(0.45)+4}{4}=2\log(\frac{0.55}{0.45})+1=1.4013$$

El punto pedido es 1.4013 de forma exacta, y por tanto, superior a 1, como se preveía.

C)

```
function [d1,d2]=calcula_densidades(x,y)

  I = -10:1e-2:10;

  d1=zeros(size(I));
  indices = find(y==1);
  for i=indices
    d1 = d1 + normpdf(I,x(i),0.5);  % Sumo gaussiana centrada en x(i)
  end
  d1 = d1/length(indices);

  d2=zeros(size(I));
  indices = find(y==2);
  for i=indices
    d2 = d2 + normpdf(I,x(i),0.5);  % Sumo gaussiana centrada en x(i)
  end
  d2 = d2/length(indices);
end
```

D)

```
function clase = clasifica(d1,d2,x)

  I = -10:1e-2:10;

  % Obligo a que x este en [-10..10]
  x = min(x,10);
  x = max(x,-10);

  % Busco la posicion de x en el vector I
  x   = round(x*100)/100;    % Dejamos x con 2 decimales
  pos = find(I==x);          % Buscamos la posicion de x en I

  % Clasifico en función del valor de d1 y d2 en la posicion pos
  if d1(pos)>d2(pos)
    clase=1;
  else
    clase=2;
  end
end
```

PROBLEMA 2

A) Para reducir el problema a un problema lineal basta sacar logaritmos a ambos lados, y renombrar las variables:

$$y = A \cdot x^M$$
$$\log(y) = \log(A) + M \cdot \log(x)$$
$$y' = a + b \cdot x'$$

B)

```
function [A,M]=potfit(x,y)

  x=x(:); y=y(:);

  T = [log(x) ones(size(x))];  % Construimos T con x' = log(x)

  coefs = pinv(T)*log(y);      % Obtenemos coefs con y' = log(y)

  % coefs contiene [b a], de los que obtenemos M y A
  M = coefs(1);                 % M = b
  A = exp(coefs(2));            % a=log(A), luego A = exp(a)
end
```

C)

```
function y = potval(A,M,x)

y = A*x.^M;

end
```