

# Tarea 1

Aprendizaje supervisado, funciones de pérdida y dimensión VC

Reconocimiento de Patrones y AA 26-01

**Instrucciones:** Resuelve los siguientes ejercicios de manera individual. La entrega es en un documento PDF. Puedes escribir tus respuestas en el documento mediante algún software o completamente a mano. Si lo haces a mano, cada paso debe ser legible, y las fotos que tomes se deben incorporar como parte del documento PDF.

## 1. [2.5 puntos] Dimensión VC

Para una clase de hipótesis de triángulos  $\mathcal{H}_{\text{tri}}$ :

- Escribe una función indicadora para evaluar si un punto 2D pertenece o no a una hipótesis de un triángulo particular  $h_{\theta} \in \mathcal{H}_{\text{tri}}$ .

### Paso 1: Entender el problema

Necesitamos una función que determine si un punto  $(x, y) \in \mathbb{R}^2$  está dentro o fuera de un triángulo. Esta función debe regresar 1 si el punto está dentro y 0 si está fuera.

### Paso 2: Método de coordenadas baricéntricas

Las coordenadas baricéntricas expresan cualquier punto del plano como una combinación lineal de los vértices del triángulo. Para un triángulo con vértices  $V_1 = (x_1, y_1)$ ,  $V_2 = (x_2, y_2)$ ,  $V_3 = (x_3, y_3)$ , cualquier punto  $P = (x, y)$  puede escribirse como:

$$P = \lambda_1 V_1 + \lambda_2 V_2 + \lambda_3 V_3$$

donde  $\lambda_1 + \lambda_2 + \lambda_3 = 1$ .

### Paso 3: Cálculo de las coordenadas baricéntricas

Para encontrar  $\lambda_1$ ,  $\lambda_2$  y  $\lambda_3$ , resolvemos el sistema:

$$x = \lambda_1 x_1 + \lambda_2 x_2 + \lambda_3 x_3 \quad (1)$$

$$y = \lambda_1 y_1 + \lambda_2 y_2 + \lambda_3 y_3 \quad (2)$$

$$1 = \lambda_1 + \lambda_2 + \lambda_3 \quad (3)$$

La solución es:

$$\lambda_1 = \frac{(y_2 - y_3)(x - x_3) + (x_3 - x_2)(y - y_3)}{(y_2 - y_3)(x_1 - x_3) + (x_3 - x_2)(y_1 - y_3)}$$

$$\lambda_2 = \frac{(y_3 - y_1)(x - x_3) + (x_1 - x_3)(y - y_3)}{(y_2 - y_3)(x_1 - x_3) + (x_3 - x_2)(y_1 - y_3)}$$

$$\lambda_3 = 1 - \lambda_1 - \lambda_2$$

#### Paso 4: Criterio de pertenencia

Un punto está dentro del triángulo si y solo si todas sus coordenadas baricéntricas son no negativas:

$$P \in \text{Triángulo} \Leftrightarrow \lambda_1 \geq 0 \wedge \lambda_2 \geq 0 \wedge \lambda_3 \geq 0$$

#### Paso 5: Función indicadora

$$h_{\boldsymbol{\theta}}(x, y) = \begin{cases} 1 & \text{si } \lambda_1 \geq 0 \wedge \lambda_2 \geq 0 \wedge \lambda_3 \geq 0 \\ 0 & \text{en caso contrario} \end{cases}$$

- b) Debes escribir la función parametrizada por los parámetros  $\boldsymbol{\theta}$  que tú consideres.

#### Paso 1: Identificar los parámetros necesarios

Un triángulo en  $\mathbb{R}^2$  queda completamente determinado por sus tres vértices. Cada vértice tiene dos coordenadas  $(x_i, y_i)$ , por lo que necesitamos 6 parámetros en total.

#### Paso 2: Definir el vector de parámetros

$$\boldsymbol{\theta} = (x_1, y_1, x_2, y_2, x_3, y_3) \in \mathbb{R}^6$$

#### Paso 3: Escribir la función parametrizada

La función  $h_{\boldsymbol{\theta}} : \mathbb{R}^2 \rightarrow \{0, 1\}$  se define como:

$$h_{\boldsymbol{\theta}}(x, y) = \mathbb{1}_{\{\lambda_1(\boldsymbol{\theta}, x, y) \geq 0\}} \cdot \mathbb{1}_{\{\lambda_2(\boldsymbol{\theta}, x, y) \geq 0\}} \cdot \mathbb{1}_{\{\lambda_3(\boldsymbol{\theta}, x, y) \geq 0\}}$$

donde  $\mathbb{1}_{\{A\}}$  es la función indicadora del evento  $A$ .

#### Paso 4: Forma explícita

Definiendo el denominador común:

$$D(\boldsymbol{\theta}) = (y_2 - y_3)(x_1 - x_3) + (x_3 - x_2)(y_1 - y_3)$$

Si  $|D(\boldsymbol{\theta})| > \epsilon$  (triángulo no degenerado), entonces:

$$h_{\boldsymbol{\theta}}(x, y) = \begin{cases} \frac{(y_2 - y_3)(x - x_3) + (x_3 - x_2)(y - y_3)}{D(\boldsymbol{\theta})} \geq 0 & \text{y} \\ 1 & \text{si } \frac{(y_3 - y_1)(x - x_3) + (x_1 - x_3)(y - y_3)}{D(\boldsymbol{\theta})} \geq 0 \text{ y} \\ & 1 - \lambda_1 - \lambda_2 \geq 0 \\ 0 & \text{en caso contrario} \end{cases}$$

Si  $|D(\boldsymbol{\theta})| \leq \epsilon$ , entonces  $h_{\boldsymbol{\theta}}(x, y) = 0$  (triángulo degenerado).

## 2. [2.5 puntos] Clases de hipótesis

Para una clase de hipótesis de elipses  $\mathcal{H}_{\text{eli}}$  donde una elipse con centro en la coordenada  $(h, k)$  se describe con la siguiente ecuación:

$$\frac{(x - h)^2}{a^2} + \frac{(y - k)^2}{b^2} = 1 \quad (4)$$

donde  $a > 0$  y  $b > 0$  son los parámetros para los semiejes de la elipse.

- a) Usa esta ecuación para escribir una función indicadora para evaluar si un punto en 2D pertenece o no a una hipótesis de una elipse particular  $h_{\theta} \in \mathcal{H}_{\text{eli}}$ , donde  $\theta = \{h, k, a, b\}$ .

**Solución:**

### Paso 1: Entender la ecuación de la elipse

La ecuación estándar de una elipse con centro en  $(h, k)$  y semiejes  $a$  y  $b$  es:

$$\frac{(x - h)^2}{a^2} + \frac{(y - k)^2}{b^2} = 1$$

Esta ecuación define la *frontera* de la elipse. Los puntos que satisfacen esta ecuación exactamente están sobre el perímetro de la elipse.

### Paso 2: Generalizar para el interior

Para determinar si un punto está dentro, sobre, o fuera de la elipse, evaluamos:

$$d(x, y) = \frac{(x - h)^2}{a^2} + \frac{(y - k)^2}{b^2}$$

La interpretación geométrica es:

- Si  $d(x, y) < 1$ : el punto está *dentro* de la elipse
- Si  $d(x, y) = 1$ : el punto está *sobre* la elipse
- Si  $d(x, y) > 1$ : el punto está *frente* de la elipse

### Paso 3: Definir la función indicadora

Para clasificación binaria, consideramos que un punto pertenece a la clase positiva si está dentro o sobre la elipse:

$$h_{\theta}(x, y) = \begin{cases} 1 & \text{si } \frac{(x-h)^2}{a^2} + \frac{(y-k)^2}{b^2} \leq 1 \\ 0 & \text{en caso contrario} \end{cases}$$

**Interpretación matemática:** La función indicadora  $h_{\theta} : \mathbb{R}^2 \rightarrow \{0, 1\}$  divide el plano en dos regiones: el interior (incluyendo la frontera) de la elipse y el exterior.

- b) Escribe a manera de pseudocódigo un algoritmo para encontrar los parámetros  $\theta' = \{h', k', a', b'\}$  de la elipse más pequeña que sólo permite un error menor o igual a un umbral  $\alpha$  (relativamente pequeño; por ejemplo, 10 %) de clasificaciones incorrectas (para una de dos clases) del total de puntos  $\mathcal{X} = \{(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)\}$ .

**Solución:**

**Paso 1: Formulación del problema**

Buscamos la elipse más pequeña (en área) que clasifique correctamente al menos  $(1 - \alpha) \cdot n$  puntos, donde  $n = |\mathcal{X}|$ .

**Definición matemática:** Minimizar  $\pi a' b'$  (área de la elipse) sujeto a:

$$\sum_{i=1}^n \mathbb{1}_{\{h_{\theta'}(x_i, y_i) \neq y_i\}} \leq \lfloor \alpha \cdot n \rfloor$$

**Paso 2: Estrategia de escalamiento**

La idea es:

- a) Estimar la forma y orientación de la elipse usando la clase positiva
- b) Encontrar la escala mínima que respete el presupuesto de error

**Paso 3: Algoritmo paso a paso**

---

**Algorithm 1** ElipseMinima - Encontrar elipse mínima con umbral de error

---

**Require:**  $\mathcal{X} = \{(x_i, y_i)\}_{i=1}^n$ ,  $\mathcal{Y} = \{y_i\}_{i=1}^n \subseteq \{-1, +1\}$ ,  $\alpha \in (0, 1)$

**Ensure:**  $\theta' = \{h', k', a', b'\}$

```
1: ▷ Separación por clases
2:  $\mathcal{X}^+ \leftarrow \{(x_i, y_i) : y_i = +1\}$ 
3:  $\mathcal{X}^- \leftarrow \{(x_i, y_i) : y_i = -1\}$ 
4:  $n^+ \leftarrow |\mathcal{X}^+|$ ,  $n^- \leftarrow |\mathcal{X}^-|$ 
5:  $\text{max\_errores} \leftarrow \lfloor \alpha \cdot n \rfloor$ 
6:
7: ▷ Estimación del centro
8:  $h' \leftarrow \frac{1}{n^+} \sum_{(x,y) \in \mathcal{X}^+} x$ 
9:  $k' \leftarrow \frac{1}{n^+} \sum_{(x,y) \in \mathcal{X}^+} y$ 
10:
11: ▷ Estimación de la forma base
12:  $\epsilon \leftarrow 1 \times 10^{-9}$ 
13:  $a_0 \leftarrow \sqrt{\frac{1}{n^+} \sum_{(x,y) \in \mathcal{X}^+} (x - h')^2 + \epsilon}$ 
14:  $b_0 \leftarrow \sqrt{\frac{1}{n^+} \sum_{(x,y) \in \mathcal{X}^+} (y - k')^2 + \epsilon}$ 
15:
16: ▷ Cálculo de distancias normalizadas
17: for cada punto  $(x_i, y_i) \in \mathcal{X}$  do
18:    $d_i \leftarrow \sqrt{\frac{(x_i - h')^2}{a_0^2} + \frac{(y_i - k')^2}{b_0^2}}$ 
19: end for
20:
21: ▷ Búsqueda de la escala óptima
22:  $\mathcal{S} \leftarrow$  valores únicos ordenados de  $\{d_i : i = 1, \dots, n\}$ 
23: for cada  $s \in \mathcal{S}$  (en orden ascendente) do
24:   errores  $\leftarrow 0$ 
25:   for  $i = 1$  to  $n$  do
26:     if  $d_i \leq s$  then
27:        $\text{pred}_i \leftarrow +1$ 
28:     else
29:        $\text{pred}_i \leftarrow -1$ 
30:     end if
31:     if  $\text{pred}_i \neq y_i$  then
32:       errores  $\leftarrow$  errores + 1
33:     end if
34:   end for
35:   if errores  $\leq \text{max\_errores}$  then
36:      $s^* \leftarrow s$ 
37:     break
38:   end if
39: end for
40:
41: ▷ Parámetros finales
42:  $a' \leftarrow a_0 \cdot s^*$ 
43:  $b' \leftarrow b_0 \cdot s^*$ 
44: return  $\theta' = \{h', k', a', b'\}$ 
```

---

- c) Ahora encuentra también a manera de pseudocódigo los parámetros del elipse más grande que no introduce nuevos errores. Puedes partir de los parámetros  $\theta'$  del inciso anterior.

**Solución:**

**Paso 1: Problema de optimización**

Dada la elipse mínima  $\theta' = \{h', k', a', b'\}$ , buscamos la elipse máxima  $\theta'' = \{h', k', a'', b''\}$  tal que:

$$\text{errores}(\theta'') = \text{errores}(\theta')$$

Es decir, no introducimos errores adicionales.

**Paso 2: Restricción geométrica**

Para no introducir nuevos errores, la elipse expandida no debe incluir puntos negativos que estaban correctamente clasificados como negativos.

**Criterio:** Si  $(x_i, y_i) \in \mathcal{X}^-$  y está correctamente clasificado por  $\theta'$ , entonces:

$$\frac{(x_i - h')^2}{(a')^2} + \frac{(y_i - k')^2}{(b')^2} > 1$$

**Paso 3: Factor de expansión**

Si escalamos uniformemente:  $a'' = s \cdot a'$  y  $b'' = s \cdot b'$  con  $s \geq 1$ , entonces la nueva distancia normalizada para el punto  $(x_i, y_i)$  es:

$$d_i^{(s)} = \sqrt{\frac{(x_i - h')^2}{(s \cdot a')^2} + \frac{(y_i - k')^2}{(s \cdot b')^2}} = \frac{d_i^{(\text{original})}}{s}$$

**Paso 4: Algoritmo de expansión**

---

**Algorithm 2** ElipseMaxima - Encontrar elipse máxima sin nuevos errores

---

**Require:**  $\theta' = \{h', k', a', b'\}$ ,  $\mathcal{X}$ ,  $\mathcal{Y}$

**Ensure:**  $\theta'' = \{h', k', a'', b''\}$

1: ▷ Calcular distancias originales  
2: **for** cada punto  $(x_i, y_i) \in \mathcal{X}$  **do**  
3:    $d_i^{(\text{original})} \leftarrow \sqrt{\frac{(x_i-h')^2}{(a')^2} + \frac{(y_i-k')^2}{(b')^2}}$   
4: **end for**  
5:  
6: ▷ Identificar puntos negativos externos  
7:  $\mathcal{N} \leftarrow \{i : y_i = -1 \text{ y } d_i^{(\text{original})} > 1\}$   
8:  
9: ▷ Calcular factor de expansión máximo  
10: **if**  $\mathcal{N} \neq \emptyset$  **then**  
11:    $\delta \leftarrow 1 \times 10^{-6}$   
12:    $s_{\max} \leftarrow \min_{i \in \mathcal{N}} d_i^{(\text{original})} - \delta$   
13:    $s_{\max} \leftarrow \max(s_{\max}, 1, 0)$   
14: **else**  
15:    $s_{\max} \leftarrow 1,0$   
16: **end if**  
17:  
18: ▷ Parámetros de la elipse máxima  
19:  $a'' \leftarrow s_{\max} \cdot a'$   
20:  $b'' \leftarrow s_{\max} \cdot b'$   
21:  
22: ▷ Verificación final  
23:  $\text{errores\_original} \leftarrow \text{contar\_errores}(\mathcal{X}, \mathcal{Y}, \theta')$   
24:  $\text{errores\_expandida} \leftarrow \text{contar\_errores}(\mathcal{X}, \mathcal{Y}, \{h', k', a'', b''\})$   
25: **while**  $\text{errores\_expandida} > \text{errores\_original}$  **do**  
26:    $s_{\max} \leftarrow s_{\max} \times 0,999$   
27:    $a'' \leftarrow s_{\max} \cdot a'$   
28:    $b'' \leftarrow s_{\max} \cdot b'$   
29:    $\text{errores\_expandida} \leftarrow \text{contar\_errores}(\mathcal{X}, \mathcal{Y}, \{h', k', a'', b''\})$   
30: **end while**  
31:  
32: **return**  $\theta'' = \{h', k', a'', b''\}$ 

---

- d) Finalmente propón como hipótesis para la función indicadora un elipse que esté entre ambos (de los incisos b y c), por ejemplo a la mitad del camino paramétricamente hablando.

**Solución:**

**Paso 1: Motivación para la interpolación**

La elipse mínima  $\theta'$  minimiza el error de entrenamiento pero puede ser muy conservadora (alto sesgo). La elipse máxima  $\theta''$  maximiza el margen pero puede ser muy permisiva. Una interpolación puede balancear estos extremos.

**Paso 2: Interpolación lineal**

Para un parámetro de interpolación  $\lambda \in [0, 1]$ , definimos:

$$h_{\text{final}} = h' \quad (\text{el centro no cambia}) \quad (5)$$

$$k_{\text{final}} = k' \quad (\text{el centro no cambia}) \quad (6)$$

$$a_{\text{final}} = (1 - \lambda)a' + \lambda a'' \quad (7)$$

$$b_{\text{final}} = (1 - \lambda)b' + \lambda b'' \quad (8)$$

**Caso especial ( $\lambda = 0,5$ ):** Para el punto medio:

$$a_{\text{final}} = \frac{a' + a''}{2} \quad (9)$$

$$b_{\text{final}} = \frac{b' + b''}{2} \quad (10)$$

### Paso 3: Función indicadora interpolada

$$h_{\text{final}}(x, y) = \begin{cases} 1 & \text{si } \frac{(x-h')^2}{a_{\text{final}}^2} + \frac{(y-k')^2}{b_{\text{final}}^2} \leq 1 \\ 0 & \text{en caso contrario} \end{cases}$$

### Paso 4: Interpretación geométrica

La elipse interpolada tiene:

- **Área:**  $\pi a_{\text{final}} b_{\text{final}} = \pi \cdot \frac{a' + a''}{2} \cdot \frac{b' + b''}{2}$
- **Error esperado:** Entre el error mínimo y el error de la elipse máxima
- **Generalización:** Potencialmente mejor que los extremos

### Paso 5: Validación

Para validar la elección de  $\lambda$ , se puede:

- a) Usar validación cruzada para seleccionar  $\lambda^* \in [0, 1]$
  - b) Evaluar el error en un conjunto de prueba
  - c) Comparar con  $\lambda = 0$  (elipse mínima) y  $\lambda = 1$  (elipse máxima)
- e) Implementarás esto en la práctica asociada a este ejercicio.

#### Guía para la implementación:

La implementación debe incluir las siguientes funciones matemáticas:

#### Función 1: Evaluación de la función indicadora

$$f_{\text{indicadora}}(\boldsymbol{\theta}, x, y) = \mathbb{1}_{\left\{ \frac{(x-h)^2}{a^2} + \frac{(y-k)^2}{b^2} \leq 1 \right\}}$$

#### Función 2: Cálculo de error

$$E(\boldsymbol{\theta}) = \frac{1}{n} \sum_{i=1}^n \mathbb{1}_{\{h_{\boldsymbol{\theta}}(x_i, y_i) \neq y_i\}}$$

#### Función 3: Optimización de parámetros

Implementar los algoritmos ElipseMinima y ElipseMaxima descritos en los incisos anteriores.

### 3. [2.5 puntos] Función de pérdida esperada

Dados los valores de una función de pérdida  $L_{kj}$ , el riesgo esperado

$$E[L] = \sum_k \sum_j \int_{R_j} L_{kj} p(x, \mathcal{C}_k) dx \quad (11)$$

se minimiza si para cada  $x$  escogemos la clase para minimizar la siguiente expresión (ya que  $p(x)$  no contiene al parámetro de clase):

$$\arg \min_j \left( \sum_k L_{kj} p(\mathcal{C}_k|x) p(x) \right) = \arg \min_j \left( \sum_k L_{kj} p(\mathcal{C}_k|x) \right) \quad (12)$$

Verifica que si  $L_{kj} = 1 - I_{kj}$ , donde  $I_{kj}$  son los elementos de la matriz identidad, entonces la minimización anterior se reduce a escoger la clase  $i$  con la probabilidad más grande  $p(\mathcal{C}_i|x)$ .

**Solución:**

**Paso 1: Entender el contexto del problema**

Estamos trabajando con un problema de decisión bayesiana donde:

- $\mathcal{C}_k$  representa las clases verdaderas ( $k = 1, 2, \dots, K$ )
- $j$  representa las decisiones o predicciones ( $j = 1, 2, \dots, K$ )
- $L_{kj}$  es la pérdida incurrida al decidir la clase  $j$  cuando la clase verdadera es  $k$
- $p(\mathcal{C}_k|x)$  es la probabilidad a posteriori de la clase  $k$  dado el input  $x$

**Paso 2: Definir la función de pérdida 0-1**

La función de pérdida  $L_{kj} = 1 - I_{kj}$  donde  $I_{kj}$  es el elemento  $(k, j)$  de la matriz identidad se puede escribir explícitamente como:

$$L_{kj} = 1 - I_{kj} = \begin{cases} 0 & \text{si } k = j \text{ (clasificación correcta)} \\ 1 & \text{si } k \neq j \text{ (clasificación incorrecta)} \end{cases}$$

**Interpretación:** Esta es la función de pérdida 0-1 estándar que penaliza igualmente todos los errores de clasificación.

**Paso 3: Calcular el riesgo esperado para una decisión específica**

Para una decisión fija  $j$ , el riesgo esperado es:

$$R(j|x) = \sum_k L_{kj} p(\mathcal{C}_k|x)$$

Sustituyendo  $L_{kj} = 1 - I_{kj}$ :

$$R(j|x) = \sum_k (1 - I_{kj}) p(\mathcal{C}_k|x) \quad (13)$$

$$= \sum_k p(\mathcal{C}_k|x) - \sum_k I_{kj} p(\mathcal{C}_k|x) \quad (14)$$

**Paso 4: Simplificar usando propiedades de las probabilidades**

Analizamos cada término:

**Primer término:**  $\sum_k p(\mathcal{C}_k|x) = 1$  (axioma de probabilidad)

**Segundo término:**  $\sum_k I_{kj}p(\mathcal{C}_k|x)$

Por definición de la matriz identidad,  $I_{kj} = 1$  solo cuando  $k = j$ , y  $I_{kj} = 0$  en todos los demás casos. Por tanto:

$$\sum_k I_{kj}p(\mathcal{C}_k|x) = I_{jj}p(\mathcal{C}_j|x) + \sum_{k \neq j} I_{kj}p(\mathcal{C}_k|x) = 1 \cdot p(\mathcal{C}_j|x) + \sum_{k \neq j} 0 \cdot p(\mathcal{C}_k|x) = p(\mathcal{C}_j|x)$$

### Paso 5: Expresión final del riesgo

Combinando los resultados:

$$R(j|x) = 1 - p(\mathcal{C}_j|x)$$

**Interpretación:** El riesgo de decidir la clase  $j$  es igual a la probabilidad de que la clase verdadera NO sea  $j$ .

### Paso 6: Encontrar la decisión óptima

Para minimizar el riesgo esperado, debemos encontrar:

$$j^* = \arg \min_j R(j|x) = \arg \min_j [1 - p(\mathcal{C}_j|x)]$$

Como el término constante 1 no afecta la minimización:

$$j^* = \arg \min_j [-p(\mathcal{C}_j|x)] = \arg \max_j p(\mathcal{C}_j|x)$$

### Paso 7: Conclusión y interpretación

**Resultado principal:** Con la función de pérdida 0-1, la regla de decisión óptima de Bayes es:

$$i^* = \arg \max_j p(\mathcal{C}_j|x)$$

### Interpretación práctica:

- Elegir siempre la clase con mayor probabilidad a posteriori
- Esta regla minimiza la probabilidad de error de clasificación
- Es la base teórica para muchos clasificadores bayesianos

### Paso 8: Verificación matemática

Para verificar que efectivamente hemos demostrado lo requerido, observemos que:

1. **Partimos de:**  $\arg \min_j (\sum_k L_{kj}p(\mathcal{C}_k|x))$  con  $L_{kj} = 1 - I_{kj}$
2. **Demostramos que esto es igual a:**  $\arg \min_j [1 - p(\mathcal{C}_j|x)]$
3. **Lo cual es equivalente a:**  $\arg \max_j p(\mathcal{C}_j|x)$
4. **Por tanto:** La minimización se reduce efectivamente a escoger la clase  $i$  con la probabilidad más grande  $p(\mathcal{C}_i|x)$  ■

### Extensión teórica:

Este resultado es fundamental en la teoría de decisiones bayesianas porque establece que, bajo pérdida 0-1:

- La frontera de decisión óptima está dada por  $p(\mathcal{C}_i|x) = p(\mathcal{C}_j|x)$  para clases  $i \neq j$
- El error de Bayes (error mínimo alcanzable) es  $1 - \max_j p(\mathcal{C}_j|x)$
- Cualquier clasificador que no siga esta regla tendrá error subóptimo

## 4. [2.5 puntos] Función de pérdida relajada

En los ejercicios 1 y 2 obtuviste funciones indicadoras para evaluar la pertenencia o no de puntos a hipótesis concretas.

- a) Debes extender la función indicadora para crear una nueva función que no evalúe de manera estrictamente “dura” la pertenencia a la hipótesis sino que permita considerar puntos cercanos a la frontera de la hipótesis de manera penalizada.

**Solución:**

### Paso 1: Motivación para la función suave

Las funciones indicadoras binarias presentan dos limitaciones principales:

- **Discontinuidad:** Cambio abrupto de 0 a 1 en la frontera
- **No diferenciabilidad:** Dificulta la optimización con métodos basados en gráfico
- **Rigidez:** No permite grados de pertenencia o incertidumbre

### Paso 2: Concepto de función de pérdida relajada

Una función de pérdida relajada  $L_{\text{soft}} : \mathbb{R}^2 \rightarrow [0, 1]$  debe satisfacer:

- a) **Continuidad:**  $L_{\text{soft}}(x, y)$  es continua
- b) **Monotonidad:** Mayor distancia a la frontera implica mayor pérdida
- c) **Compatibilidad:**  $\lim_{\delta \rightarrow 0^+} L_{\text{soft}}(x, y) = L_{\text{hard}}(x, y)$

### Paso 3: Definición matemática para elipse

Para una hipótesis de elipse con parámetros  $\boldsymbol{\theta} = \{h, k, a, b\}$ , definimos la **distancia normalizada**:

$$d(x, y; \boldsymbol{\theta}) = \sqrt{\frac{(x - h)^2}{a^2} + \frac{(y - k)^2}{b^2}}$$

**Interpretación geométrica:**

- $d(x, y) = 1$ : punto en la frontera de la elipse
- $d(x, y) < 1$ : punto en el interior
- $d(x, y) > 1$ : punto en el exterior

### Paso 4: Función de pérdida relajada por partes

Definimos la función de pérdida suave con parámetro de suavidad  $\delta > 0$ :

$$L_{\text{soft}}(x, y; \boldsymbol{\theta}, \delta) = \begin{cases} 0 & \text{si } d(x, y) \leq 1 \\ \frac{d(x, y) - 1}{\delta} & \text{si } 1 < d(x, y) \leq 1 + \delta \\ 1 & \text{si } d(x, y) > 1 + \delta \end{cases}$$

**Propiedades matemáticas:**

- a) **Rango:**  $L_{\text{soft}} \in [0, 1]$

- b) **Continuidad:** La función es continua en todo  $\mathbb{R}^2$
- c) **Límite:**  $\lim_{\delta \rightarrow 0^+} L_{\text{soft}}(x, y) = \mathbb{1}_{\{d(x,y) > 1\}}$
- b) Tú debes definir una manera de extender la frontera hasta cierta distancia máxima más allá de la frontera original. Por ejemplo: englobando a la hipótesis dentro de un círculo con cierto radio, o definiendo una distancia  $\Delta d$  más allá de la frontera de la hipótesis.

**Solución:**

### Paso 1: Concepto de zona de transición

La extensión de la frontera crea una **zona de transición** (o margen suave) alrededor de la hipótesis original donde la clasificación no es binaria sino gradual.

### Paso 2: Definición formal de la extensión

Para una elipse con parámetros  $\boldsymbol{\theta} = \{h, k, a, b\}$  y parámetro de extensión  $\delta > 0$ , definimos tres regiones:

$$\mathcal{R}_{\text{interior}} = \{(x, y) : d(x, y; \boldsymbol{\theta}) \leq 1\} \quad (15)$$

$$\mathcal{R}_{\text{transición}} = \{(x, y) : 1 < d(x, y; \boldsymbol{\theta}) \leq 1 + \delta\} \quad (16)$$

$$\mathcal{R}_{\text{exterior}} = \{(x, y) : d(x, y; \boldsymbol{\theta}) > 1 + \delta\} \quad (17)$$

### Paso 3: Interpretación geométrica de la extensión

La zona de transición  $\mathcal{R}_{\text{transición}}$  forma una “cáscara elíptica” definida por:

**Frontera interior:**  $\frac{(x-h)^2}{a^2} + \frac{(y-k)^2}{b^2} = 1$

**Frontera exterior:**  $\frac{(x-h)^2}{(a\sqrt{1+\delta})^2} + \frac{(y-k)^2}{(b\sqrt{1+\delta})^2} = 1$

### Paso 4: Parametrización alternativa usando distancia euclíadiana

#### Método 1: Extensión uniforme

Extender la elipse uniformemente en todas las direcciones:

$$\boldsymbol{\theta}_{\text{ext}} = \{h, k, a + \Delta a, b + \Delta b\}$$

donde  $\Delta a, \Delta b > 0$  son extensiones en los ejes principales.

#### Método 2: Extensión por factor de escala

Escalar la elipse por un factor  $s > 1$ :

$$\boldsymbol{\theta}_{\text{ext}} = \{h, k, s \cdot a, s \cdot b\}$$

La zona de transición está entre las elipses con factores de escala 1 y  $s$ .

### Paso 5: Elección del parámetro de extensión

El parámetro  $\delta$  controla el ancho de la zona de transición:

- **δ pequeño** ( $\delta < 0,1$ ): Transición muy estrecha, comportamiento casi binario
- **δ moderado** ( $0,1 \leq \delta \leq 1$ ): Balance entre suavidad y precisión

- **$\delta$  grande ( $\delta > 1$ ):** Transición muy gradual, alta tolerancia

**Criterio de selección:**  $\delta$  puede elegirse mediante validación cruzada o como fracción del tamaño característico de la hipótesis (ej.  $\delta = 0,1 \cdot \max(a, b)$ ).

- En la región donde extiendas la frontera la función de pérdida debe penalizar en lugar de evaluar a cero.

**Solución:**

### Paso 1: Principio de penalización gradual

En lugar de una penalización binaria (0 o 1), implementamos una penalización que crece monotónicamente con la distancia a la frontera original.

### Paso 2: Función de penalización lineal

La función de pérdida relajada con penalización lineal es:

$$L_{\text{relajada}}(x, y; \boldsymbol{\theta}, \delta) = \begin{cases} 0 & \text{si } d(x, y) \leq 1 \\ \frac{d(x, y) - 1}{\delta} & \text{si } 1 < d(x, y) \leq 1 + \delta \\ 1 & \text{si } d(x, y) > 1 + \delta \end{cases}$$

**Justificación matemática:**

**Región interior ( $d \leq 1$ ):**

$$L_{\text{relajada}}(x, y) = 0$$

*Interpretación:* Clasificación completamente correcta, sin penalización.

**Zona de transición ( $1 < d \leq 1 + \delta$ ):**

$$L_{\text{relajada}}(x, y) = \frac{d - 1}{\delta} \in (0, 1)$$

*Interpretación:* Penalización proporcional a la distancia de la frontera.

**Región exterior ( $d > 1 + \delta$ ):**

$$L_{\text{relajada}}(x, y) = 1$$

*Interpretación:* Penalización máxima para puntos claramente fuera de la hipótesis extendida.

### Paso 3: Propiedades de la función de penalización

**Continuidad:** Verificamos que la función es continua en los puntos de transición:

*En  $d = 1$ :*

$$\lim_{d \rightarrow 1^-} L(d) = 0, \quad \lim_{d \rightarrow 1^+} L(d) = \frac{1 - 1}{\delta} = 0$$

*En  $d = 1 + \delta$ :*

$$\lim_{d \rightarrow (1+\delta)^-} L(d) = \frac{(1 + \delta) - 1}{\delta} = 1, \quad \lim_{d \rightarrow (1+\delta)^+} L(d) = 1$$

**Monotonidad:** En la zona de transición:

$$\frac{dL}{dd} = \frac{1}{\delta} > 0$$

#### Paso 4: Interpretación probabilística

La función relajada puede interpretarse como la probabilidad de clasificación incorrecta:

$$L_{\text{relajada}}(x, y) \approx P(\text{clasificación incorrecta} | \text{posición en zona de transición})$$

donde la probabilidad aumenta linealmente con la distancia a la frontera.

- d) Puedes elegir para este ejercicio una de las hipótesis de los ejercicios 1 y 2.

**Solución:**

#### Elección: Hipótesis de elipse del Ejercicio 2

#### Paso 1: Justificación de la elección

Elegimos la clase de hipótesis de elipses  $\mathcal{H}_{\text{eli}}$  por las siguientes ventajas:

- a) **Simplicidad computacional:** La distancia normalizada se calcula eficientemente
- b) **Diferenciabilidad:** La función es diferenciable en casi todo punto
- c) **Flexibilidad geométrica:** Las elipses pueden adaptarse a diferentes formas de datos
- d) **Parametrización compacta:** Solo 4 parámetros  $\{h, k, a, b\}$  más  $\delta$

#### Paso 2: Definición completa de la función relajada

Para una elipse con parámetros  $\boldsymbol{\theta} = \{h, k, a, b\}$  y parámetro de suavidad  $\delta > 0$ :

$$L_{\text{elipse}}(x, y; \boldsymbol{\theta}, \delta) = \begin{cases} 0 & \text{si } d(x, y) \leq 1 \\ \frac{d(x, y) - 1}{\delta} & \text{si } 1 < d(x, y) \leq 1 + \delta \\ 1 & \text{si } d(x, y) > 1 + \delta \end{cases}$$

donde:

$$d(x, y) = \sqrt{\frac{(x - h)^2}{a^2} + \frac{(y - k)^2}{b^2}}$$

#### Paso 3: Familia de funciones relajadas

Variando  $\delta$ , obtenemos una familia de funciones:

$$\mathcal{F}_{\text{relajada}} = \{L_{\text{elipse}}(\cdot; \boldsymbol{\theta}, \delta) : \boldsymbol{\theta} \in \mathbb{R}^4, \delta \in \mathbb{R}^+\}$$

**Casos límite:**

- $\delta \rightarrow 0^+$ : Recuperamos la función indicadora binaria

- $\delta \rightarrow \infty$ : La función se vuelve completamente suave pero pierde precisión

#### Paso 4: Análisis de la superficie de pérdida

La función  $L_{\text{elipse}}(x, y)$  define una superficie en  $\mathbb{R}^3$  con:

**Valores en regiones clave:**

- **Centro de la elipse:**  $L(h, k) = 0$
- **Frontera original:**  $L(x, y) = 0$  para  $d(x, y) = 1$
- **Frontera extendida:**  $L(x, y) = 1$  para  $d(x, y) = 1 + \delta$

**Gradiente en la zona de transición:**

$$\nabla L(x, y) = \frac{1}{\delta} \cdot \frac{\nabla d(x, y)}{d(x, y)} \cdot \left( \frac{x - h}{a^2}, \frac{y - k}{b^2} \right)^T$$

- e) Implementarás esto en la práctica asociada a este ejercicio.

**Solución:**

#### Guía para la implementación

La implementación debe incluir las siguientes componentes matemáticas:

#### Función 1: Distancia normalizada

$$d(\mathbf{x}; \boldsymbol{\theta}) = \sqrt{(\mathbf{x} - \boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1} (\mathbf{x} - \boldsymbol{\mu})}$$

donde  $\boldsymbol{\mu} = (h, k)^T$  y  $\boldsymbol{\Sigma} = \text{diag}(a^2, b^2)$ .

#### Función 2: Pérdida relajada

$$L(\mathbf{x}; \boldsymbol{\theta}, \delta) = \max \left( 0, \min \left( 1, \frac{d(\mathbf{x}) - 1}{\delta} \right) \right)$$

#### Función 3: Gradiente de la pérdida

Para optimización, el gradiente respecto a los parámetros es:

$$\frac{\partial L}{\partial \boldsymbol{\theta}} = \begin{cases} \mathbf{0} & \text{si } d \leq 1 \text{ o } d > 1 + \delta \\ \frac{1}{\delta} \frac{\partial d}{\partial \boldsymbol{\theta}} & \text{si } 1 < d \leq 1 + \delta \end{cases}$$

**Algoritmo de optimización:**

---

**Algorithm 3** Optimización de Función de Pérdida Relajada

---

**Require:**  $\{\mathbf{x}_i\}_{i=1}^n, \{y_i\}_{i=1}^n, \delta > 0, \alpha > 0$  (tasa de aprendizaje),  $\epsilon > 0$  (tolerancia)

**Ensure:**  $\boldsymbol{\theta}^*$  (parámetros óptimos)

```
1: ▷ Inicialización
2: Inicializar  $\boldsymbol{\theta}^{(0)}$  aleatoriamente
3:  $t \leftarrow 0$ 
4: converged  $\leftarrow$  false
5:
6: while  $\neg$ converged do
7:   ▷ Calcular pérdida total
8:      $L_{\text{total}} \leftarrow \sum_{i=1}^n L(\mathbf{x}_i; \boldsymbol{\theta}^{(t)}, \delta)$ 
9:
10:    ▷ Calcular gradientes
11:     $\nabla_{\boldsymbol{\theta}} L_{\text{total}} \leftarrow \sum_{i=1}^n \nabla_{\boldsymbol{\theta}} L(\mathbf{x}_i; \boldsymbol{\theta}^{(t)}, \delta)$ 
12:
13:    ▷ Actualizar parámetros
14:     $\boldsymbol{\theta}^{(t+1)} \leftarrow \boldsymbol{\theta}^{(t)} - \alpha \nabla_{\boldsymbol{\theta}} L_{\text{total}}$ 
15:
16:    ▷ Verificar convergencia
17:    if  $\|\boldsymbol{\theta}^{(t+1)} - \boldsymbol{\theta}^{(t)}\| < \epsilon$  then
18:      converged  $\leftarrow$  true
19:    end if
20:     $t \leftarrow t + 1$ 
21: end while
22:
23: return  $\boldsymbol{\theta}^* = \boldsymbol{\theta}^{(t)}$ 
```

---

**Validación de la implementación:**

- a) Verificar que  $\lim_{\delta \rightarrow 0} L(\mathbf{x}) = L_{\text{hard}}(\mathbf{x})$
- b) Comprobar continuidad en las fronteras
- c) Validar monotonicidad en la zona de transición