Clase 8 - Clasificación (parte 1)

Aprendizaje de Máquinas - MA5204

Felipe Tobar

Department of Mathematical Engineering &
Center for Mathematical Modelling
Universidad de Chile

14 de marzo de 2021

Introducción - Problema de Clasificación

El problema de clasificación dice relación con la identificación del conjunto, categoría o *clase* a la cual pertenece un elemento en base a sus *características*.

En el contexto del aprendizaje supervisado, puede ser visto como un problema de regresión.

En efecto, basta suponer que y variable de salida (o variable dependiente) es *categórica* y usualmente denotada por $\{0, 1\}$ en el caso binario o para el caso multiclase $\{1 \ldots K\}$

Entre los métodos de clasificación existentes, trabajaremos en

1. k vecinos más cercanos
2. Regresión Logística
3. Support Vector Machines (SVM)

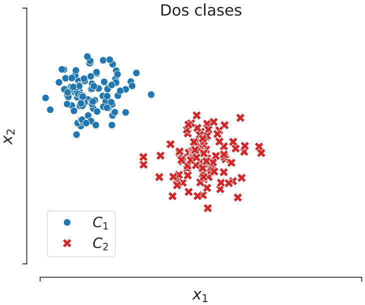

Fig.. Ejemplo del problema de clasificación binaria, donde la clase $\mathcal{C}_1$ está presentada en azul y la clase $\mathcal{C}_2$ en rojo.

Introducción - Problema de Clasificación

El problema de clasificación dice relación con la identificación del conjunto, categoría o *clase* a la cual pertenece un elemento en base a sus *características*.

En el contexto del aprendizaje supervisado, puede ser visto como un problema de regresión.

En efecto, basta suponer que y variable de salida (o variable dependiente) es *categórica* y usualmente denotada por $\{0, 1\}$ en el caso binario o para el caso multiclase $\{1 \ldots K\}$

Entre los métodos de clasificación existentes, trabajaremos en

1. k vecinos más cercanos
2. Regresión Logística
3. Support Vector Machines (SVM)

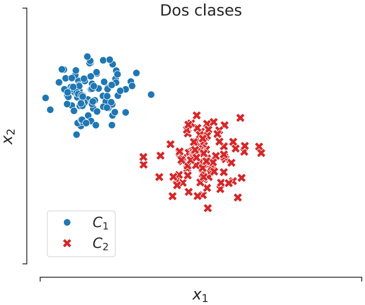

Fig.. Ejemplo del problema de clasificación binaria, donde la clase $\mathcal{C}_1$ está presentada en azul y la clase $\mathcal{C}_2$ en rojo.

Introducción - Problema de Clasificación

El problema de clasificación dice relación con la identificación del conjunto, categoría o *clase* a la cual pertenece un elemento en base a sus *características*.

En el contexto del aprendizaje supervisado, puede ser visto como un problema de regresión.

En efecto, basta suponer que y variable de salida (o variable dependiente) es *categórica* y usualmente denotada por $\{0, 1\}$ en el caso binario o para el caso multiclase $\{1 \dots K\}$

Entre los métodos de clasificación existentes, trabajaremos en

1. k vecinos más cercanos
2. Regresión Logística
3. Support Vector Machines (SVM)

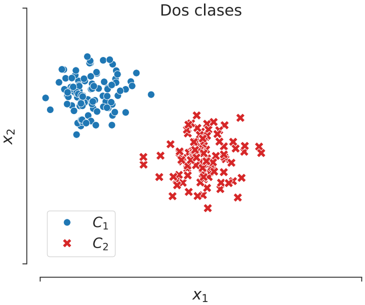

Fig.. Ejemplo del problema de clasificación binaria, donde la clase $\mathcal{C}_1$ está presentada en azul y la clase $\mathcal{C}_2$ en rojo.

Introducción - Problema de Clasificación

El problema de clasificación dice relación con la identificación del conjunto, categoría o *clase* a la cual pertenece un elemento en base a sus *características*.

En el contexto del aprendizaje supervisado, puede ser visto como un problema de regresión.

En efecto, basta suponer que y variable de salida (o variable dependiente) es *categórica* y usualmente denotada por $\{0, 1\}$ en el caso binario o para el caso multiclase $\{1 \dots K\}$

Entre los métodos de clasificación existentes, trabajaremos en

1. k vecinos más cercanos
2. Regresión Logística
3. Support Vector Machines (SVM)

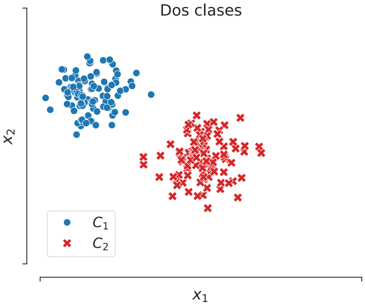

Fig.. Ejemplo del problema de clasificación binaria, donde la clase $\mathcal{C}_1$ está presentada en azul y la clase $\mathcal{C}_2$ en rojo.

Introducción - Problema de Clasificación

El problema de clasificación dice relación con la identificación del conjunto, categoría o *clase* a la cual pertenece un elemento en base a sus *características*.

En el contexto del aprendizaje supervisado, puede ser visto como un problema de regresión.

En efecto, basta suponer que y variable de salida (o variable dependiente) es *categórica* y usualmente denotada por $\{0, 1\}$ en el caso binario o para el caso multiclase $\{1 \dots K\}$

Entre los métodos de clasificación existentes, trabajaremos en

1. k vecinos más cercanos
2. Regresión Logística
3. Support Vector Machines (SVM)

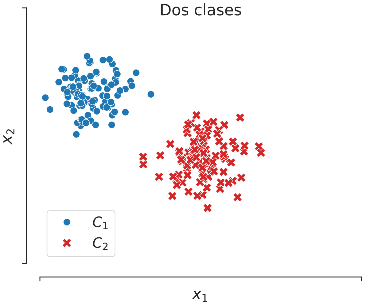

Fig.. Ejemplo del problema de clasificación binaria, donde la clase C_1 está presentada en azul y la clase C_2 en rojo.

Introducción - Problema de Clasificación

El problema de clasificación dice relación con la identificación del conjunto, categoría o *clase* a la cual pertenece un elemento en base a sus *características*.

En el contexto del aprendizaje supervisado, puede ser visto como un problema de regresión.

En efecto, basta suponer que y variable de salida (o variable dependiente) es *categórica* y usualmente denotada por $\{0, 1\}$ en el caso binario o para el caso multiclase $\{1 \ldots K\}$

Entre los métodos de clasificación existentes, trabajaremos en

1. k vecinos más cercanos
2. Regresión Logística
3. Support Vector Machines (SVM)

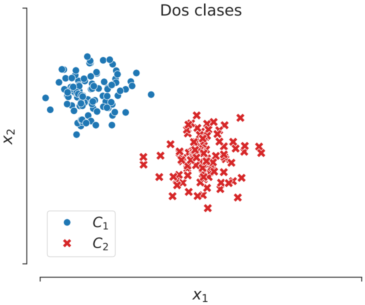

Fig.. Ejemplo del problema de clasificación binaria, donde la clase C_1 está presentada en azul y la clase C_2 en rojo.

Introducción - Problema de Clasificación

El problema de clasificación dice relación con la identificación del conjunto, categoría o *clase* a la cual pertenece un elemento en base a sus *características*.

En el contexto del aprendizaje supervisado, puede ser visto como un problema de regresión.

En efecto, basta suponer que y variable de salida (o variable dependiente) es *categórica* y usualmente denotada por $\{0, 1\}$ en el caso binario o para el caso multiclase $\{1 \dots K\}$

Entre los métodos de clasificación existentes, trabajaremos en

1. k vecinos más cercanos
2. Regresión Logística
3. Support Vector Machines (SVM)

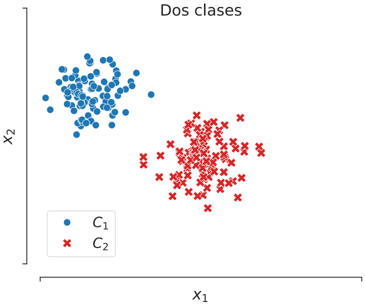

Fig.. Ejemplo del problema de clasificación binaria, donde la clase $\mathcal{C}_1$ está presentada en azul y la clase $\mathcal{C}_2$ en rojo.

Clasificación Lineal - Binaria

Consideremos el caso binario K=2 clases, proponemos un modelo lineal para relacionar la variable independiente con su clase, es decir, $y(x) = a^T x + b$ tal que $x \in \mathcal{C}_1$ si $y(x) \geq 0$, en caso contrario, $x \in \mathcal{C}_2$.

Para encontrar los parámetros a, b óptimos, sean x_1 y x_2 en la región de decisión $y(x) = 0$

$$\begin{aligned} 0 &= y(x_1) - y(x_2) \\ &= a^\top x_1 + b - a^\top x_2 - b \\ &= a^\top (x_1 - x_2). \end{aligned}$$

Además, observemos que para cualquier x en la región de decisión se tiene que

$$\|\mathrm{proy}_a(x)\| = \|x\|\, cos(\theta) = \|x\| \frac{a^\top x}{\|a\| \cdot \|x\|} = -\frac{b}{\|a\|}$$

Con lo que tenemos una intepretación geométrica de ambos parámetros.

Clasificación Lineal - Binaria

Consideremos el caso binario K=2 clases, proponemos un modelo lineal para relacionar la variable independiente con su clase, es decir, $y(x) = a^T x + b$ tal que $x \in \mathcal{C}_1$ si $y(x) \geq 0$, en caso contrario, $x \in \mathcal{C}_2$.
Para encontrar los parámetros a, b óptimos, sean x_1 y x_2 en la región de decisión $y(x) = 0$

$$\begin{aligned} 0 &= y(x_1) - y(x_2) \\ &= a^\top x_1 + b - a^\top x_2 - b \\ &= a^\top (x_1 - x_2). \end{aligned}$$

Además, observemos que para cualquier x en la región de decisión se tiene que

$$\|\text{proy}_a(x)\| = \|x\|\, cos(\theta) = \|x\| \frac{a^\top x}{\|a\| \cdot \|x\|} = -\frac{b}{\|a\|}$$

Con lo que tenemos una intepretación geométrica de ambos parámetros.

Clasificación Lineal - Binaria

Consideremos el caso binario K=2 clases, proponemos un modelo lineal para relacionar la variable independiente con su clase, es decir, $y(x) = a^T x + b$ tal que $x \in \mathcal{C}_1$ si $y(x) \geq 0$, en caso contrario, $x \in \mathcal{C}_2$.
Para encontrar los parámetros a, b óptimos, sean x_1 y x_2 en la región de decisión $y(x) = 0$

$$\begin{aligned} 0 &= y(x_1) - y(x_2) \\ &= a^\top x_1 + b - a^\top x_2 - b \\ &= a^\top (x_1 - x_2). \end{aligned}$$

Además, observemos que para cualquier x en la región de decisión se tiene que

$$\|\mathrm{proy}_a(x)\| = \|x\| \, cos(\theta) = \|x\| \frac{a^\top x}{\|a\| \cdot \|x\|} = -\frac{b}{\|a\|}$$

Con lo que tenemos una intepretación geométrica de ambos parámetros.

Clasificación Lineal - Binaria

Consideremos el caso binario K=2 clases, proponemos un modelo lineal para relacionar la variable independiente con su clase, es decir, $y(x) = a^T x + b$ tal que $x \in \mathcal{C}_1$ si $y(x) \geq 0$, en caso contrario, $x \in \mathcal{C}_2$.
Para encontrar los parámetros a, b óptimos, sean x_1 y x_2 en la región de decisión $y(x) = 0$

$$\begin{aligned} 0 &= y(x_1) - y(x_2) \\ &= a^\top x_1 + b - a^\top x_2 - b \\ &= a^\top (x_1 - x_2). \end{aligned}$$

Además, observemos que para cualquier x en la región de decisión se tiene que

$$\|\text{proy}_a(x)\| = \|x\| \, cos(\theta) = \|x\| \, \frac{a^\top x}{\|a\| \cdot \|x\|} = -\frac{b}{\|a\|}$$

Con lo que tenemos una intepretación geométrica de ambos parámetros.

Clasificación Lineal - Binaria

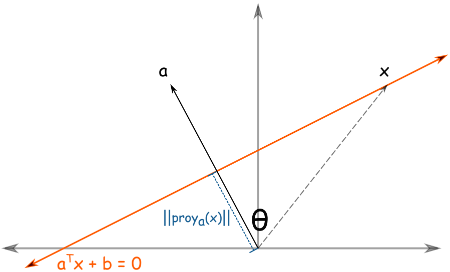

Fig.. Proyección de un punto sobre la región de decisión.

Clasificación Lineal - Binaria

También es posible interpretar $y(x)$ como una distancia con signo entre un $x \in \mathbb{R}^M$ cualquiera y la superficie de decisión.

Para ver esto, consideremos $x \in \mathbb{R}^M$ y descompongámoslo en dos componentes: la primera denotada por $x_\perp$, la cual es la proyección ortogonal de x en el hiperplano de decisión, y la segunda que es perpendicular al hiperplano (y consecuentemente paralela al vector a) denotada por $r\frac{a}{||a||}$, donde r denota la distancia (positiva o negativa) entre x y el hiperplano de decisión. Expresamos entonces

$$x = x_\perp + r\frac{a}{||a||},$$

y observamos que

$$y(x) = a^\top x + b = a^\top \left(x_\perp + r\frac{a}{||a||} \right) + b = \underbrace{a^\top x_\perp + b}_{=0} + r\frac{a^\top a}{||a||} = r||a||.$$

Luego $r = \frac{y(x)}{||a||}$ y como r es una medida con signo, $y(x)$ también lo es.

Clasificación Lineal - Binaria

También es posible interpretar $y(x)$ como una distancia con signo entre un $x \in \mathbb{R}^M$ cualquiera y la superficie de decisión.
Para ver esto, consideremos $x \in \mathbb{R}^M$ y descompongámoslo en dos componentes: la primera denotada por $x_\perp$, la cual es la proyección ortogonal de x en el hiperplano de decisión, y la segunda que es perpendicular al hiperplano (y consecuentemente paralela al vector a) denotada por $r\frac{a}{\|a\|}$, donde r denota la distancia (positiva o negativa) entre x y el hiperplano de decisión. Expresamos entonces

$$x = x_\perp + r\frac{a}{\|a\|},$$

y observamos que

$$y(x) = a^\top x + b = a^\top \left(x_\perp + r\frac{a}{\|a\|} \right) + b = \underbrace{a^\top x_\perp + b}_{=0} + r\frac{a^\top a}{\|a\|} = r||a||.$$

Luego $r = \frac{y(x)}{\|a\|}$ y como r es una medida con signo, $y(x)$ también lo es.

Clasificación Lineal - Binaria

También es posible interpretar $y(x)$ como una distancia con signo entre un $x \in \mathbb{R}^M$ cualquiera y la superficie de decisión.
Para ver esto, consideremos $x \in \mathbb{R}^M$ y descompongámoslo en dos componentes: la primera denotada por $x_\perp$, la cual es la proyección ortogonal de x en el hiperplano de decisión, y la segunda que es perpendicular al hiperplano (y consecuentemente paralela al vector a) denotada por $r\frac{a}{\|a\|}$, donde r denota la distancia (positiva o negativa) entre x y el hiperplano de decisión. Expresamos entonces

$$x = x_\perp + r\frac{a}{\|a\|},$$

y observamos que

$$y(x) = a^\top x + b = a^\top \left(x_\perp + r\frac{a}{\|a\|} \right) + b = \underbrace{a^\top x_\perp + b}_{=0} + r\frac{a^\top a}{\|a\|} = r||a||.$$

Luego $r = \frac{y(x)}{\|a\|}$ y como r es una medida con signo, $y(x)$ también lo es.

Clasificación Lineal - Binaria

También es posible interpretar $y(x)$ como una distancia con signo entre un $x \in \mathbb{R}^M$ cualquiera y la superficie de decisión.
Para ver esto, consideremos $x \in \mathbb{R}^M$ y descompongámoslo en dos componentes: la primera denotada por $x_\perp$, la cual es la proyección ortogonal de x en el hiperplano de decisión, y la segunda que es perpendicular al hiperplano (y consecuentemente paralela al vector a) denotada por $r\frac{a}{\|a\|}$, donde r denota la distancia (positiva o negativa) entre x y el hiperplano de decisión. Expresamos entonces

$$x = x_\perp + r\frac{a}{\|a\|},$$

y observamos que

$$y(x) = a^\top x + b = a^\top \left(x_\perp + r\frac{a}{\|a\|} \right) + b = \underbrace{a^\top x_\perp + b}_{=0} + r\frac{a^\top a}{\|a\|} = r||a||.$$

Luego $r = \frac{y(x)}{\|a\|}$ y como r es una medida con signo, $y(x)$ también lo es.

Clasificación Lineal - Binaria

También es posible interpretar $y(x)$ como una distancia con signo entre un $x \in \mathbb{R}^M$ cualquiera y la superficie de decisión.
Para ver esto, consideremos $x \in \mathbb{R}^M$ y descompongámoslo en dos componentes: la primera denotada por $x_\perp$, la cual es la proyección ortogonal de x en el hiperplano de decisión, y la segunda que es perpendicular al hiperplano (y consecuentemente paralela al vector a) denotada por $r\frac{a}{\|a\|}$, donde r denota la distancia (positiva o negativa) entre x y el hiperplano de decisión. Expresamos entonces

$$x = x_\perp + r\frac{a}{\|a\|},$$

y observamos que

$$y(x) = a^\top x + b = a^\top \left(x_\perp + r\frac{a}{\|a\|} \right) + b = \underbrace{a^\top x_\perp + b}_{=0} + r\frac{a^\top a}{\|a\|} = r||a||.$$

Luego $r = \frac{y(x)}{\|a\|}$ y como r es una medida con signo, $y(x)$ también lo es.

Clasificación Lineal - Múltiples clases

El caso de múltiples clases ($K > 2$) puede ser enfrentado mediante una extensión del caso binario, algunas de ellas

1. **One versus rest:** Construcción de $K - 1$ clasificadores binarios que discrimina una clase $\mathcal{C}_k$ del resto
2. **One versus one:** Construccion de $K(K-1)/2$ clasificadores binarios que discriminan entre cada par posible de clases

¿Qué problema presentan estos métodos? Busquemos otra forma

Una alternativa más robusta para resolver el problema de clasificación multiclase es construir un clasificador para K clases que contiene K funciones lineales de la forma

$$y_k(x) = a_k^\top x + b_k, \quad k = 1, \ldots, K.$$

Donde x es asignado a la clase $\mathcal{C}_k$ si y solo si $y_k(x) > y_j(x), \forall j \neq k$, es decir:

$$\mathcal{C}(x) = \arg\max_k y_j(x).$$

¿Qué ventajas posee este método?

Clasificación Lineal - Múltiples clases

El caso de múltiples clases ($K > 2$) puede ser enfrentado mediante una extensión del caso binario, algunas de ellas

1. **One versus rest:** Construcción de $K - 1$ clasificadores binarios que discrimina una clase $\mathcal{C}_k$ del resto
2. **One versus one:** Construccion de $K(K-1)/2$ clasificadores binarios que discriminan entre cada par posible de clases

¿Qué problema presentan estos métodos? Busquemos otra forma

Una alternativa más robusta para resolver el problema de clasificación multiclase es construir un clasificador para K clases que contiene K funciones lineales de la forma

$$y_k(x) = a_k^\top x + b_k, \quad k = 1, \dots, K.$$

Donde x es asignado a la clase $\mathcal{C}_k$ si y solo si $y_k(x) > y_j(x), \forall j \neq k$, es decir:

$$\mathcal{C}(x) = \arg\max_k y_j(x).$$

¿Qué ventajas posee este método?

Clasificación Lineal - Múltiples clases

El caso de múltiples clases ($K > 2$) puede ser enfrentado mediante una extensión del caso binario, algunas de ellas

1. **One versus rest:** Construcción de $K - 1$ clasificadores binarios que discrimina una clase $\mathcal{C}_k$ del resto
2. **One versus one:** Construccion de $K(K-1)/2$ clasificadores binarios que discriminan entre cada par posible de clases

¿Qué problema presentan estos métodos? Busquemos otra forma

Una alternativa más robusta para resolver el problema de clasificación multiclase es construir un clasificador para K clases que contiene K funciones lineales de la forma

$$y_k(x) = a_k^\top x + b_k, \quad k = 1, \dots, K.$$

Donde x es asignado a la clase $\mathcal{C}_k$ si y solo si $y_k(x) > y_j(x), \forall j \neq k$, es decir:

$$\mathcal{C}(x) = \arg\max_k y_j(x).$$

¿Qué ventajas posee este método?

Clasificación Lineal - Múltiples clases

El caso de múltiples clases ($K > 2$) puede ser enfrentado mediante una extensión del caso binario, algunas de ellas

1. **One versus rest:** Construcción de $K - 1$ clasificadores binarios que discrimina una clase $\mathcal{C}_k$ del resto
2. **One versus one:** Construccion de $K(K-1)/2$ clasificadores binarios que discriminan entre cada par posible de clases

¿Qué problema presentan estos métodos? Busquemos otra forma

Una alternativa más robusta para resolver el problema de clasificación multiclase es construir un clasificador para K clases que contiene K funciones lineales de la forma

$$y_k(x) = a_k^\top x + b_k, \quad k = 1, \ldots, K.$$

Donde x es asignado a la clase $\mathcal{C}_k$ si y solo si $y_k(x) > y_j(x), \forall j \neq k$, es decir:

$$\mathcal{C}(x) = \arg\max_k y_j(x).$$

¿Qué ventajas posee este método?

Clasificación Lineal - Múltiples clases

El caso de múltiples clases ($K > 2$) puede ser enfrentado mediante una extensión del caso binario, algunas de ellas

1. **One versus rest:** Construcción de $K - 1$ clasificadores binarios que discrimina una clase $\mathcal{C}_k$ del resto
2. **One versus one:** Construccion de $K(K-1)/2$ clasificadores binarios que discriminan entre cada par posible de clases

¿Qué problema presentan estos métodos? Busquemos otra forma

Una alternativa más robusta para resolver el problema de clasificación multiclase es construir un clasificador para K clases que contiene K funciones lineales de la forma

$$y_k(x) = a_k^\top x + b_k, \quad k = 1, \dots, K.$$

Donde x es asignado a la clase $\mathcal{C}_k$ si y solo si $y_k(x) > y_j(x), \forall j \neq k$, es decir:

$$\mathcal{C}(x) = \arg\max_k y_j(x).$$

¿Qué ventajas posee este método?

Clasificación Lineal - Múltiples clases

El caso de múltiples clases ($K > 2$) puede ser enfrentado mediante una extensión del caso binario, algunas de ellas

1. **One versus rest:** Construcción de $K - 1$ clasificadores binarios que discrimina una clase $\mathcal{C}_k$ del resto
2. **One versus one:** Construccion de $K(K-1)/2$ clasificadores binarios que discriminan entre cada par posible de clases

¿Qué problema presentan estos métodos? Busquemos otra forma

Una alternativa más robusta para resolver el problema de clasificación multiclase es construir un clasificador para K clases que contiene K funciones lineales de la forma

$$y_k(x) = a_k^\top x + b_k, \quad k = 1, \dots, K.$$

Donde x es asignado a la clase $\mathcal{C}_k$ si y solo si $y_k(x) > y_j(x), \forall j \neq k$, es decir:

$$\mathcal{C}(x) = \arg\max_k y_j(x).$$

¿Qué ventajas posee este método?

Clasificación Lineal - Múltiples clases

El caso de múltiples clases ($K > 2$) puede ser enfrentado mediante una extensión del caso binario, algunas de ellas

1. **One versus rest:** Construcción de $K - 1$ clasificadores binarios que discrimina una clase $\mathcal{C}_k$ del resto
2. **One versus one:** Construccion de $K(K-1)/2$ clasificadores binarios que discriminan entre cada par posible de clases

¿Qué problema presentan estos métodos? Busquemos otra forma

Una alternativa más robusta para resolver el problema de clasificación multiclase es construir un clasificador para K clases que contiene K funciones lineales de la forma

$$y_k(x) = a_k^\top x + b_k, \quad k = 1, \ldots, K.$$

Donde x es asignado a la clase $\mathcal{C}_k$ si y solo si $y_k(x) > y_j(x), \forall j \neq k$, es decir:

$$\mathcal{C}(x) = \arg\max_k \, y_j(x).$$

¿Qué ventajas posee este método?

Ajuste mediante mínimos cuadrados

Ya hemos planteado el modelo y analizado el rol y significado de cada uno de sus parámetros; ahora queda por estudiar cómo determinar dichos parámetros a y b, dado un conjunto de datos $\mathcal{D}$.

Consideremos el punto $x \in \mathbb{R}^M$ con clase $c \in \{\mathcal{C}_k\}_{k=1}^K$. Usaremos la *codificación* $t \in \{0,1\}^K$ para representar la pertenencia de x a su respectiva clase. Es decir,

$$c = \mathcal{C}_j \Leftrightarrow [t]_j = 1 \wedge [t]_i = 0, \quad \forall i \neq j.$$

Este tipo de codificación es conocida como *one-hot encoding.* **¿Por qué la usamos?** Asumiendo entonces un modelo lineal para cada clase $\mathcal{C}_k$, se tiene que

$$y_k(x) = a_k^\top x + b_k = \tilde{\theta}_k^\top \tilde{x}, \quad \text{donde } \tilde{x} = \begin{pmatrix} x \\ 1 \end{pmatrix} \in \mathbb{R}^{M+1}, \quad \tilde{\theta}_k = \begin{pmatrix} a_k \\ b_k \end{pmatrix} \in \mathbb{R}^{M+1}$$

Lo anterior se puede unir en un único sistema matricial:

$$\tilde{\Theta} = (\theta_1, \cdots, \theta_K) \in \mathbb{R}^{(M+1)\times K} \implies y(x) = \begin{pmatrix} y_1(x) \\ \vdots \\ y_K(x) \end{pmatrix} = \tilde{\Theta}^\top \tilde{x}.$$

Ajuste mediante mínimos cuadrados

Ya hemos planteado el modelo y analizado el rol y significado de cada uno de sus parámetros; ahora queda por estudiar cómo determinar dichos parámetros a y b, dado un conjunto de datos $\mathcal{D}$.

Consideremos el punto $x \in \mathbb{R}^M$ con clase $c \in \{\mathcal{C}_k\}_{k=1}^K$. Usaremos la *codificación* $t \in \{0,1\}^K$ para representar la pertenencia de x a su respectiva clase. Es decir,

$$c = \mathcal{C}_j \Leftrightarrow [t]_j = 1 \wedge [t]_i = 0, \quad \forall i \neq j.$$

Este tipo de codificación es conocida como *one-hot encoding*. **¿Por qué la usamos?**

Asumiendo entonces un modelo lineal para cada clase $\mathcal{C}_k$, se tiene que

$$y_k(x) = a_k^\top x + b_k = \tilde{\theta}_k^\top \tilde{x}, \quad \text{donde } \tilde{x} = \begin{pmatrix} x \\ 1 \end{pmatrix} \in \mathbb{R}^{M+1}, \quad \tilde{\theta}_k = \begin{pmatrix} a_k \\ b_k \end{pmatrix} \in \mathbb{R}^{M+1}$$

Lo anterior se puede unir en un único sistema matricial:

$$\tilde{\Theta} = (\theta_1, \cdots, \theta_K) \in \mathbb{R}^{(M+1)\times K} \implies y(x) = \begin{pmatrix} y_1(x) \\ \vdots \\ y_K(x) \end{pmatrix} = \tilde{\Theta}^\top \tilde{x}.$$

Ajuste mediante mínimos cuadrados

Ya hemos planteado el modelo y analizado el rol y significado de cada uno de sus parámetros; ahora queda por estudiar cómo determinar dichos parámetros a y b, dado un conjunto de datos $\mathcal{D}$.

Consideremos el punto $x \in \mathbb{R}^M$ con clase $c \in \{\mathcal{C}_k\}_{k=1}^K$. Usaremos la *codificación* $t \in \{0, 1\}^K$ para representar la pertenencia de x a su respectiva clase. Es decir,

$$c = \mathcal{C}_j \Leftrightarrow [t]_j = 1 \wedge [t]_i = 0, \quad \forall i \neq j.$$

Este tipo de codificación es conocida como *one-hot encoding.* **¿Por qué la usamos?**

Asumiendo entonces un modelo lineal para cada clase $\mathcal{C}_k$, se tiene que

$$y_k(x) = a_k^\top x + b_k = \tilde{\theta}_k^\top \tilde{x}, \quad \text{donde } \tilde{x} = \begin{pmatrix} x \\ 1 \end{pmatrix} \in \mathbb{R}^{M+1}, \quad \tilde{\theta}_k = \begin{pmatrix} a_k \\ b_k \end{pmatrix} \in \mathbb{R}^{M+1}$$

Lo anterior se puede unir en un único sistema matricial:

$$\tilde{\Theta} = (\theta_1, \cdots, \theta_K) \in \mathbb{R}^{(M+1)\times K} \implies y(x) = \begin{pmatrix} y_1(x) \\ \vdots \\ y_K(x) \end{pmatrix} = \tilde{\Theta}^\top \tilde{x}.$$

Ajuste mediante mínimos cuadrados

Ya hemos planteado el modelo y analizado el rol y significado de cada uno de sus parámetros; ahora queda por estudiar cómo determinar dichos parámetros a y b, dado un conjunto de datos $\mathcal{D}$.
Consideremos el punto $x \in \mathbb{R}^M$ con clase $c \in \{\mathcal{C}_k\}_{k=1}^K$. Usaremos la *codificación* $t \in \{0,1\}^K$ para representar la pertenencia de x a su respectiva clase. Es decir,

$$c = \mathcal{C}_j \Leftrightarrow [t]_j = 1 \wedge [t]_i = 0, \quad \forall i \neq j.$$

Este tipo de codificación es conocida como *one-hot encoding*. **¿Por qué la usamos?**
Asumiendo entonces un modelo lineal para cada clase $\mathcal{C}_k$, se tiene que

$$y_k(x) = a_k^\top x + b_k = \tilde{\theta}_k^\top \tilde{x}, \quad \text{donde } \tilde{x} = \begin{pmatrix} x \\ 1 \end{pmatrix} \in \mathbb{R}^{M+1}, \quad \tilde{\theta}_k = \begin{pmatrix} a_k \\ b_k \end{pmatrix} \in \mathbb{R}^{M+1}$$

Lo anterior se puede unir en un único sistema matricial:

$$\tilde{\Theta} = (\theta_1, \cdots, \theta_K) \in \mathbb{R}^{(M+1)\times K} \implies y(x) = \begin{pmatrix} y_1(x) \\ \vdots \\ y_K(x) \end{pmatrix} = \tilde{\Theta}^\top \tilde{x}.$$

Ajuste mediante mínimos cuadrados

Ya hemos planteado el modelo y analizado el rol y significado de cada uno de sus parámetros; ahora queda por estudiar cómo determinar dichos parámetros a y b, dado un conjunto de datos $\mathcal{D}$.
Consideremos el punto $x \in \mathbb{R}^M$ con clase $c \in \{\mathcal{C}_k\}_{k=1}^K$. Usaremos la *codificación* $t \in \{0, 1\}^K$ para representar la pertenencia de x a su respectiva clase. Es decir,

$$c = \mathcal{C}_j \Leftrightarrow [t]_j = 1 \wedge [t]_i = 0, \quad \forall i \neq j.$$

Este tipo de codificación es conocida como *one-hot encoding.* **¿Por qué la usamos?**
Asumiendo entonces un modelo lineal para cada clase $\mathcal{C}_k$, se tiene que

$$y_k(x) = a_k^\top x + b_k = \tilde{\theta}_k^\top \tilde{x}, \quad \text{donde } \tilde{x} = \begin{pmatrix} x \\ 1 \end{pmatrix} \in \mathbb{R}^{M+1}, \quad \tilde{\theta}_k = \begin{pmatrix} a_k \\ b_k \end{pmatrix} \in \mathbb{R}^{M+1}$$

Lo anterior se puede unir en un único sistema matricial:

$$\tilde{\Theta} = (\theta_1, \cdots, \theta_K) \in \mathbb{R}^{(M+1)\times K} \implies y(x) = \begin{pmatrix} y_1(x) \\ \vdots \\ y_K(x) \end{pmatrix} = \tilde{\Theta}^\top \tilde{x}.$$

Ajuste mediante mínimos cuadrados

Con la notación establecida, ahora podemos enfocarnos en el entrenamiento del modelo. Para esto consideremos un conjunto de entrenamiento $\{(x_n, t_n)\}_{n=1}^{N}$. El enfoque de entrenamiento será el correspondiente a mínimos cuadrados asociado al error de asignación:

$$J = \sum_{i=1}^{N} \left\| t_i - \tilde{\Theta}^\top \tilde{x}_i \right\|_2^2$$

Por otra parte, definiendo las siguientes matrices:

$$T = \begin{pmatrix} t_1^\top \\ \vdots \\ t_N^\top \end{pmatrix} \in \mathbb{R}^{N \times K}, \qquad \tilde{X} = \begin{pmatrix} t_1^\top \\ \vdots \\ t_N^\top \end{pmatrix} \in \mathbb{R}^{N \times (M+1)}$$

se tiene el siguiente resultado:

Ajuste mediante mínimos cuadrados

Con la notación establecida, ahora podemos enfocarnos en el entrenamiento del modelo. Para esto consideremos un conjunto de entrenamiento $\{(x_n, t_n)\}_{n=1}^{N}$. El enfoque de entrenamiento será el correspondiente a mínimos cuadrados asociado al error de asignación:

$$J = \sum_{i=1}^{N} \left\| t_i - \tilde{\Theta}^\top \tilde{x}_i \right\|_2^2$$

Por otra parte, definiendo las siguientes matrices:

$$T = \begin{pmatrix} t_1^\top \\ \vdots \\ t_N^\top \end{pmatrix} \in \mathbb{R}^{N \times K}, \qquad \tilde{X} = \begin{pmatrix} t_1^\top \\ \vdots \\ t_N^\top \end{pmatrix} \in \mathbb{R}^{N \times (M+1)}$$

se tiene el siguiente resultado:

Ajuste mediante mínimos cuadrados

Con la notación establecida, ahora podemos enfocarnos en el entrenamiento del modelo. Para esto consideremos un conjunto de entrenamiento $\{(x_n, t_n)\}_{n=1}^{N}$. El enfoque de entrenamiento será el correspondiente a mínimos cuadrados asociado al error de asignación:

$$J = \sum_{i=1}^{N} \left\| t_i - \tilde{\Theta}^\top \tilde{x}_i \right\|_2^2$$

Por otra parte, definiendo las siguientes matrices:

$$T = \begin{pmatrix} t_1^\top \\ \vdots \\ t_N^\top \end{pmatrix} \in \mathbb{R}^{N \times K}, \qquad \tilde{X} = \begin{pmatrix} t_1^\top \\ \vdots \\ t_N^\top \end{pmatrix} \in \mathbb{R}^{N \times (M+1)}$$

se tiene el siguiente resultado:

Ajuste mediante mínimos cuadrados

Lemma

Bajo la notación anterior, $J = \mathrm{Tr}\left((\tilde{\mathrm{X}}\tilde{\Theta} - \mathrm{T})^\top(\tilde{\mathrm{X}}\tilde{\Theta} - \mathrm{T})\right)$ *y su mínimo es alcanzado en:*

$$\tilde{\Theta} = (\tilde{X}^\top \tilde{X})^{-1}\tilde{X}^\top T$$

donde Tr *corresponde al operador traza:* $A \in \mathbb{R}^{n\times n} \mapsto \mathrm{Tr}(\mathrm{A}) := \sum_{\mathrm{i}=1}^{\mathrm{n}} \mathrm{a}_{\mathrm{ii}}$.

Ajuste mediante mínimos cuadrados

Demostración.

$$J = \sum_{i=1}^{N} \left\| t_i - \tilde{\Theta}^\top \tilde{x}_i \right\|_2^2 = \sum_{i=1}^{N} \left\| \left(T - \tilde{X}\tilde{\Theta}\right)_{i\cdot} \right\|_2^2 = \sum_{i=1}^{N} \sum_{j=1}^{K} \left(T - \tilde{X}\tilde{\Theta}\right)_{ij} \left(T - \tilde{X}\tilde{\Theta}\right)_{ij}$$

$$= \sum_{i=1}^{N} \sum_{j=1}^{K} \left(T - \tilde{X}\tilde{\Theta}\right)_{ji}^\top \left(T - \tilde{X}\tilde{\Theta}\right)_{ij} = \sum_{j=1}^{K} \left[\left(T - \tilde{X}\tilde{\Theta}\right)^\top \left(T - \tilde{X}\tilde{\Theta}\right) \right]_{jj}$$

$$= \mathrm{Tr}\left((\tilde{\mathrm{X}}\tilde{\Theta} - \mathrm{T})^\top (\tilde{\mathrm{X}}\tilde{\Theta} - \mathrm{T}) \right).$$

Por otra parte:

$$\frac{\partial J}{\partial \tilde{\Theta}} = 2(\tilde{X}\tilde{\Theta} - T)^\top \tilde{X} = 0 \iff \tilde{\Theta}^\top \tilde{X}^\top \tilde{X} - T^\top \tilde{X} = 0$$

$$\iff \tilde{\Theta}^\top = T^\top \tilde{X}(\tilde{X}^\top \tilde{X})^{-1} \iff \tilde{\Theta} = (\tilde{X}^\top \tilde{X})^{-1} \tilde{X}^\top T$$

Y dado que J es estrictamente convexo, su mínimo se alcanza en su único punto crítico.

$\square$

Ajuste mediante mínimos cuadrados

Problemáticas conceptuales de este enfoque:

1. Sensibilidad a presencia de puntos aislados (outliers)
2. Intervención "manual" de las etiquetas

Fig.. Ejemplo ilustrativo sobre cómo los puntos lejanos de una clase pueden afectar incorrectamente los resultados.

Ajuste mediante mínimos cuadrados

Problemáticas conceptuales de este enfoque:

1. Sensibilidad a presencia de puntos aislados (outliers)
2. Intervención "manual" de las etiquetas

Fig.. Ejemplo ilustrativo sobre cómo los puntos lejanos de una clase pueden afectar incorrectamente los resultados.

Ajuste mediante mínimos cuadrados

Problemáticas conceptuales de este enfoque:

1. Sensibilidad a presencia de puntos aislados (outliers)
2. Intervención ”manual” de las etiquetas

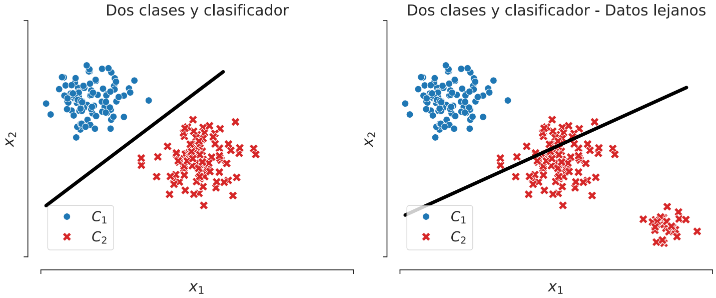

Fig.. Ejemplo ilustrativo sobre cómo los puntos lejanos de una clase pueden afectar incorrectamente los resultados.

Clase 8 - Clasificación (parte 1)

Aprendizaje de Máquinas - MA5204

Felipe Tobar

Department of Mathematical Engineering &
Center for Mathematical Modelling
Universidad de Chile

14 de marzo de 2021