TDVI: Inteligencia Artificial

Regresión logística / Support Vector Machines

UTDT - LTD

Estructura de la clase

- Repaso de regresión logística
- Introducción a SVMs

Estructura de la clase

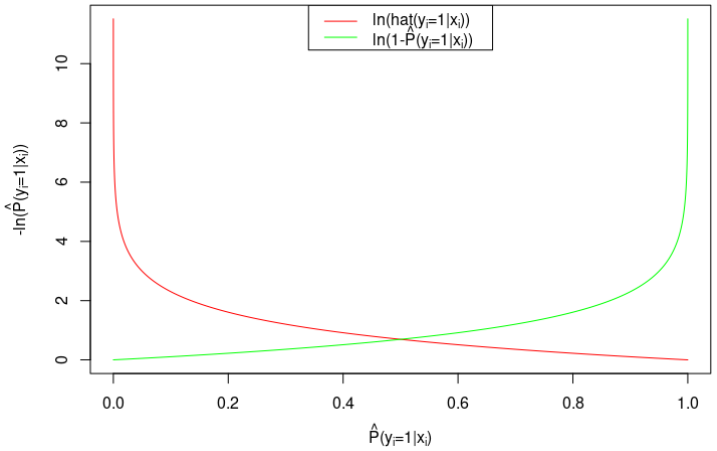
- Repaso de regresión logística
- Introducción a SVMs

Repaso de regresión logística

Recordemos qué mide *cross-entropy* (*log-loss*):

Para una observación *i*, toma valores altos si no se predice una alta probabilidad a la clase correcta

Para el caso binario: $$CE = -\frac{1}{n}\sum_{i=1}^{n}(y_i \cdot ln(\hat{P}(y_i = 1|x_i)) + (1 - y_i) \cdot ln(1 - \hat{P}(y_i = 1|x_i)))$$



Aclaración: minimizar cross-entropy es equivalente a maximizar likelihood como se presenta en ISLP

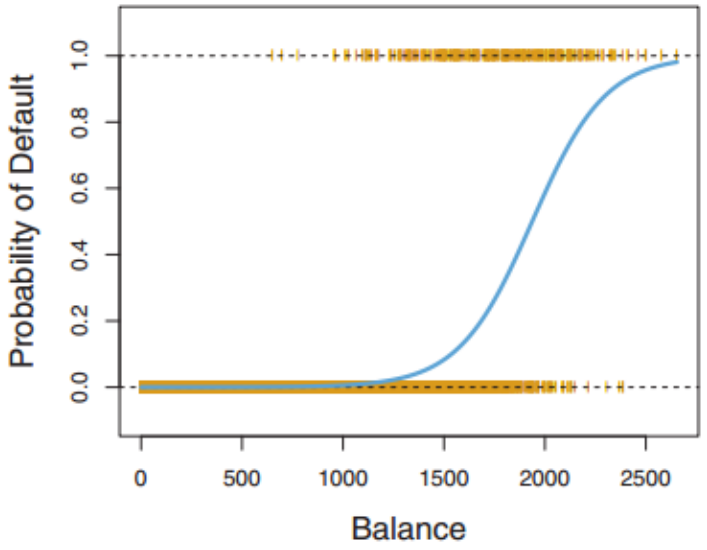
$$\prod_{i:y_i=1} p(x_i) \prod_{i':y_{i'}=0} (1 - p(x_{i'}))$$

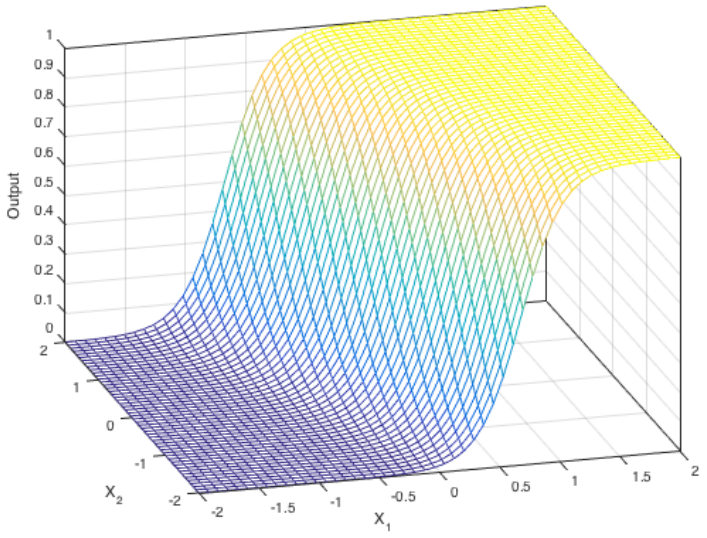
Repaso de regresión logística

En regresión logística (que es un clasificador), asumimos la siguiente forma funcional:

$$p(y_i = 1/x_{i1}, x_{i2}, \dots, x_{im}) = \frac{1}{1 + e^{-(\beta_0 + \beta_1 x_{i1} + \beta_1 x_{i2} + \cdots + \beta_m x_{im})}}$$

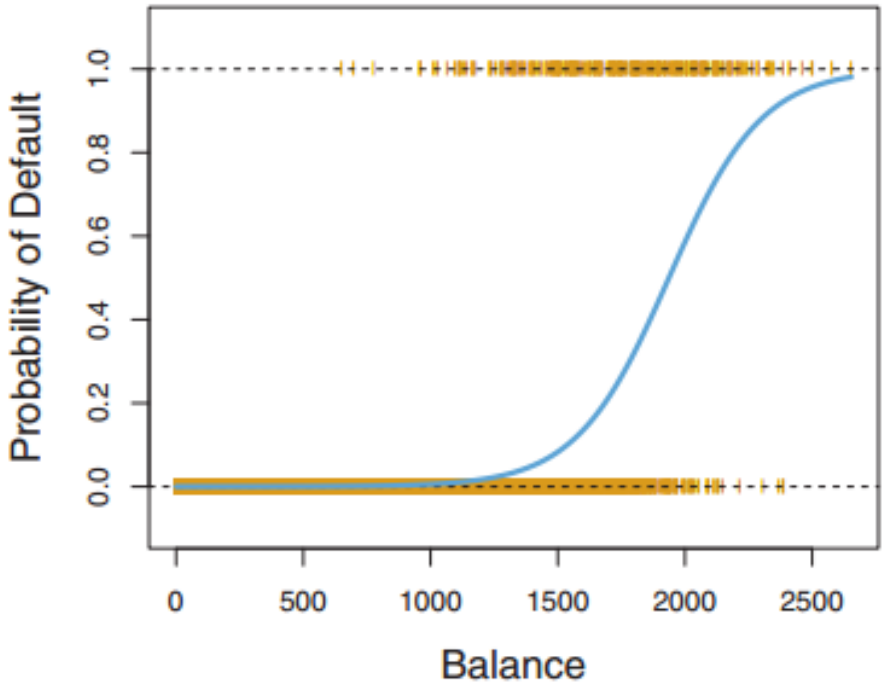
Es decir, la probabilidad predicha se relaciona con cada predictor mediante la función sigmoidea





Repaso de regresión logística

¿Cómo afecta el valor de los coeficientes al comportamiento de la predicción?



Pensemos en el caso de un único predictor

$$p(y_i = 1/x_{1i}) = \frac{1}{1 + e^{-(\beta_0 + \beta_1 x_i)}}$$

Repaso de regresión logística

¿Es regresión logística un clasificador lineal?

¿Cómo hacer para que logre captar no linealidades?

Pensemos en el caso de dos predictores:

$$\hat{P}(y_i = 1/x_{i1}, x_{i2}) = \frac{1}{1 + e^{-(\beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2})}}$$

Repaso de regresión logística

Tomaremos como los coeficientes estimados a aquellos que minimizan cross-entropy medido en el training set

$$CE = -\frac{1}{n}\sum_{i=1}^{n}(y_i \cdot ln(\hat{P}(y_i = 1|x_i)) + (1 - y_i) \cdot ln(1 - \hat{P}(y_i = 1|x_i)))$$

$$\hat{P}(y_i = 1/x_{i1}, x_{i2}, \dots, x_{im}) = \frac{1}{1 + e^{-(\widehat{\beta_0} + \widehat{\beta_1}x_{i1} + \widehat{\beta_2}x_{i2} + \cdots + \widehat{\beta_m}x_{im})}}$$

Es decir:

$$min_{\widehat{\beta_0}, \widehat{\beta_1}, \widehat{\beta_2} \dots, \widehat{\beta_m}; X, y} = CE(\widehat{\beta_0}, \widehat{\beta_1}, \widehat{\beta_2} \dots, \widehat{\beta_m}; X, y)$$

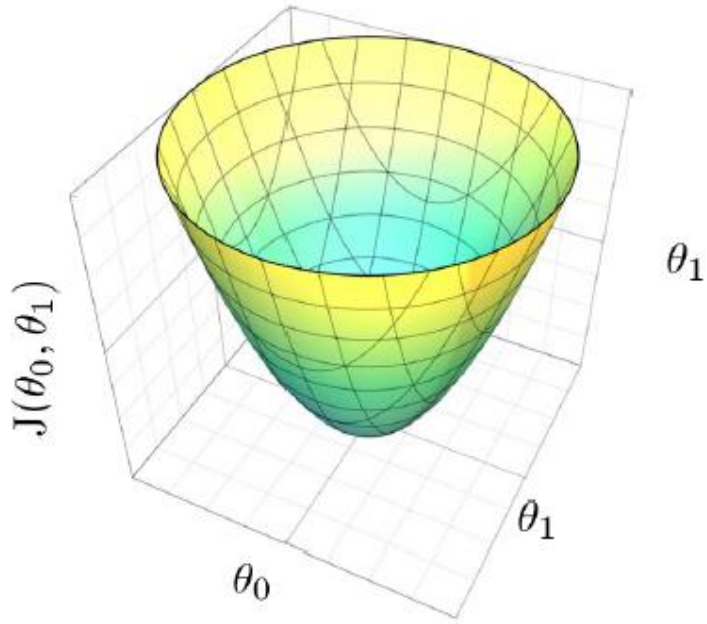
Los valores que minimizan esta expresión no pueden encontrarse mediante una fórmula cerrada

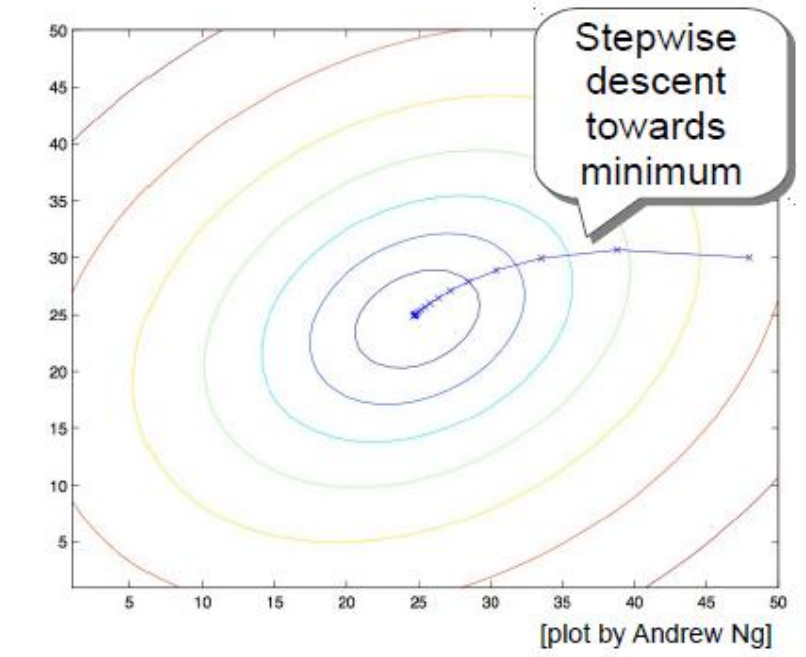
¿Cómo podremos encontrarlos?

Repaso de regresión logística

Lo haremos a través de algún método numérico de optimización. Por ejemplo, descenso gradiente

$$\theta_{i+1} = \theta_i - \gamma * \nabla\,(\theta_i)$$





Repaso de regresión logística

Para entrenar un modelo de regresión logística con descenso gradiente necesitamos las derivadas parciales de CE respecto a cada coeficiente

Son tediosas de derivar (link), pero la expresión final que se obtiene es sorprendentemente simple

Si el costo viene dato por: $J(\theta) = -\frac{1}{m}\sum_{i=1}^{m} y^i \log(h_\theta(x^i)) + (1-y^i)\log(1-h_\theta(x^i))$

La probabilidad predicha por: $h_\theta(x) = g(\theta^T x) \quad g(z) = \frac{1}{1+e^{-z}}$

La derivada parcial de CE respecto a cada parámetro es igual a:

$$\frac{\partial}{\partial \theta_j} J(\theta) = \frac{1}{m}\sum_{i=1}^{m} (h_\theta(x^i) - y^i) x^i$$

Antes de ejecutar el algoritmo es buena práctica escalar las variables (link)

Repaso de regresión logística

Cross-entropy puede modificarse de la siguiente manera:

$$CE_{reg} = CE(\Theta; X, y) + \frac{1}{2}\lambda \sum_{j=1}^{p} \theta_j^2$$

Ahora se penaliza la complejidad del modelo. Importante: el coeficiente asociado a la constante (o bias), no se penaliza.

La derivada parcial de la función de costo respecto a cada coeficiente penalizado vendrá dada por:

$$\frac{\partial}{\partial \theta_j} CE_{reg} = \frac{1}{m} \sum_{i=1}^{m} (h_\theta(x^i) - y^i) x^i + \lambda \theta_j$$

¿Qué efecto tiene aumentar λ?

¿A qué tenderá la predicción cuando λ tiende a infinito?

Repaso de regresión logística

En este diagrama se resume todo lo visto:

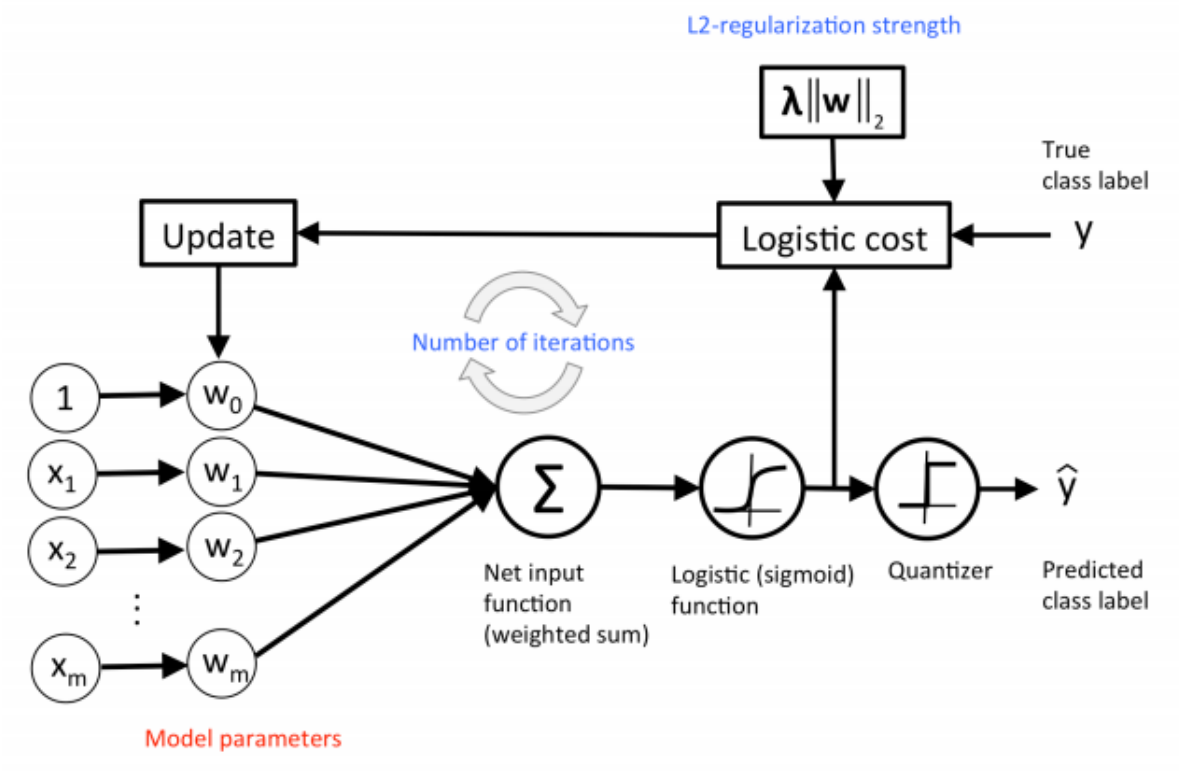


Figure 11: Conceptual overview of logistic regression.

Fuente: https://arxiv.org/pdf/1811.12808.pdf

Estructura de la clase

- Repaso de regresión logística
- Introducción a SVMs

Introducción a SVMs

Supongamos que trabajamos con *p* atributos (un espacio de atributos de dimensión *p*), ¿qué es un hiperplano en dicho espacio?

Dados unos valores de *β*, serán todos los valores de x que cumplen con la condición:

$$\beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_p X_p = 0$$

Si por el contrario, se tiene que:

$$\beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_p X_p > 0$$

Decimos que una observación se encuentra de un lado del hiperplano. Si fuera menor a 0, decimos que se encuentra del otro

Introducción a SVMs

Veamos un ejemplo con $p = 2$

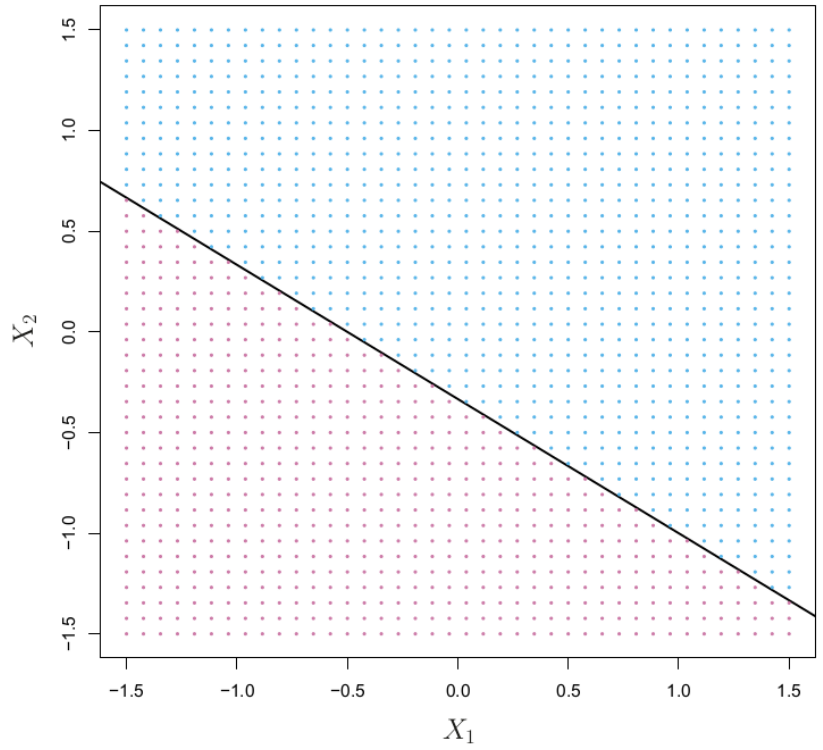


FIGURE 9.1. *The hyperplane* $1 + 2X_1 + 3X_2 = 0$ *is shown. The blue region is the set of points for which* $1 + 2X_1 + 3X_2 > 0$*, and the purple region is the set of points for which* $1 + 2X_1 + 3X_2 < 0$*.*

Introducción a SVMs

Supongamos que tenemos valores de X como siempre, pero ahora y puede valor -1 ó 1 (estamos en un problema de clasificación binaria)

Objetivo: queremos encontrar un hiperplano que separe el espacio de atributos de manera que de un lado del mismo queden las observaciones con *y* = 1 y del otro queden las observaciones con *y* = -1

Sin perder generalidad vamos a querer lo siguiente:

$$\beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \cdots + \beta_p x_{ip} > 0 \text{ if } y_i = 1,$$

$$\beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \cdots + \beta_p x_{ip} < 0 \text{ if } y_i = -1$$

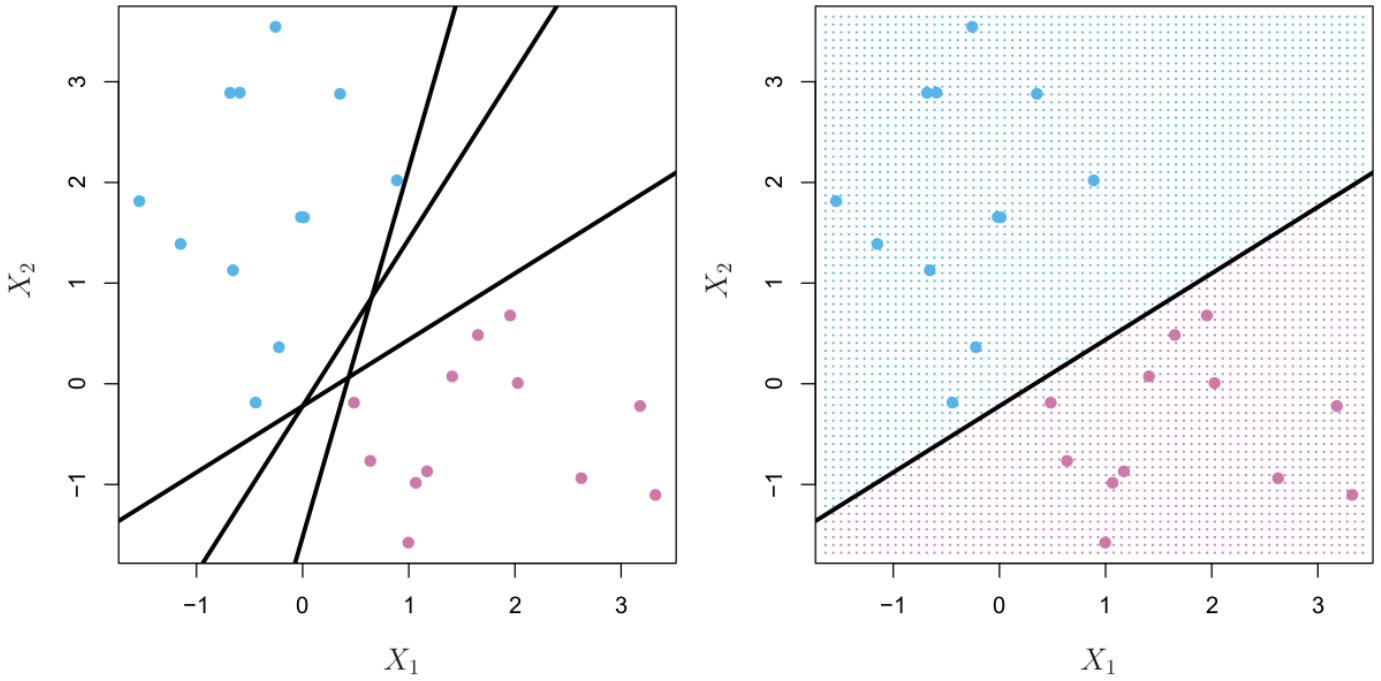
O lo que es lo mismo:

$$y_i(\beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \cdots + \beta_p x_{ip}) > 0$$

Introducción a SVMs

Si los datos son linealmente separables, seguramente existan infinitos hiperplanos que cumplan con:

$$y_i(\beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \cdots + \beta_p x_{ip}) > 0$$

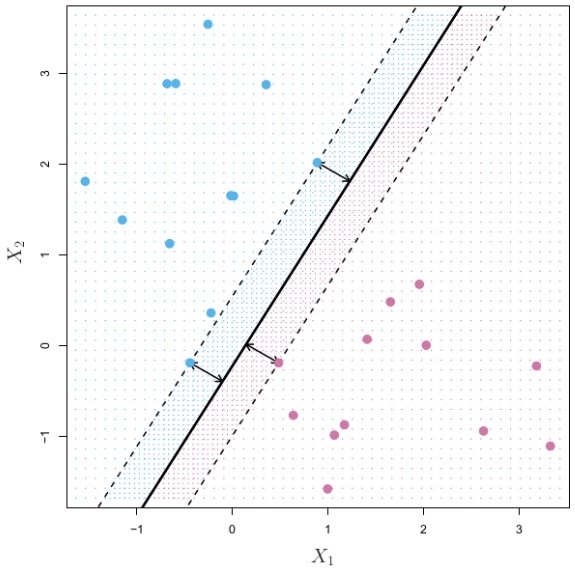


Vamos a decir que el mejor de ellos será el que tenga el mayor margen

Introducción a SVMs

Maximal Margin Classifier

El hiperplano de mayor margen será aquel que, clasificando bien las observaciones, tenga mayor distancia mínima con las mismas



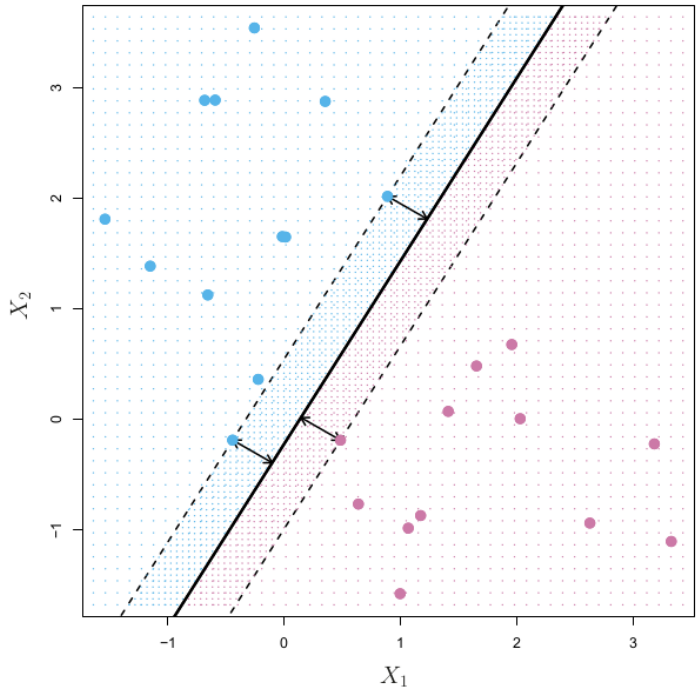
Si β_0, β_1,…, β_p son los coeficientes del maximal margin classifier, una observación será predicha por el signo de:

$$f(x^*) = \beta_0 + \beta_1 x_1^* + \beta_2 x_2^* + \cdots + \beta_p x_p^*$$

Introducción a SVMs

En la figura, son 3 las observaciones que definen la posición del hiperplano separador. Las otras podrían no estar y se obtendría el mismo hiperplano!

A las observaciones sobre el margen se las conoce como los vectores de soporte (*support vectors*)



Introducción a SVMs

¿Cómo se encuentra el hiperplano con mayor margen?

Es el que surge del siguiente problema de optimización

$$\underset{\beta_0,\beta_1,\ldots,\beta_p,M}{\text{maximize}} \; M \tag{9.9}$$

$$\text{subject to} \sum_{j=1}^{p} \beta_j^2 = 1, \tag{9.10}$$

$$y_i(\beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \cdots + \beta_p x_{ip}) \geq M \;\; \forall\, i = 1, \ldots, n. \tag{9.11}$$

La restricción 9.10 es importante. Si se cumple, entonces la distancia de cada observación *i* al hiperplano vendrá dada por:

$$y_i(\beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \cdots + \beta_p x_{ip})$$

Las observaciones para las que 9.11 es igualdad son los vectores de soporte

Introducción a SVMs

¿Qué sucede en casos no linealmente separables?

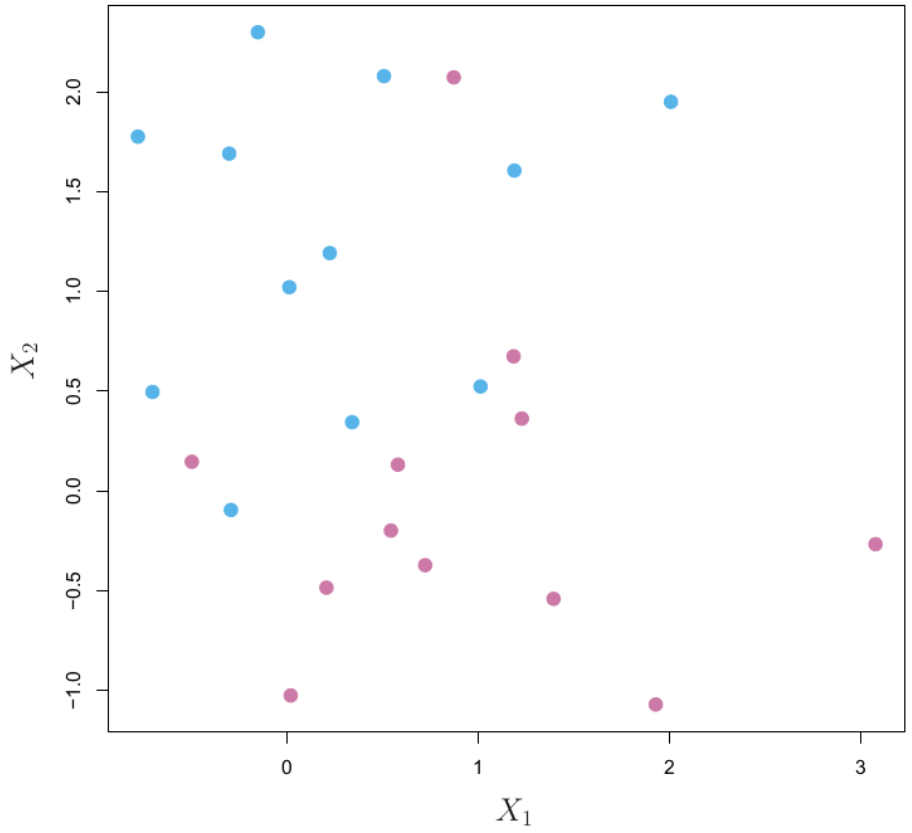


FIGURE 9.4. *There are two classes of observations, shown in blue and in purple. In this case, the two classes are not separable by a hyperplane, and so the maximal margin classifier cannot be used.*

Introducción a SVMs

Support vector classifier (*soft margin classifier*)

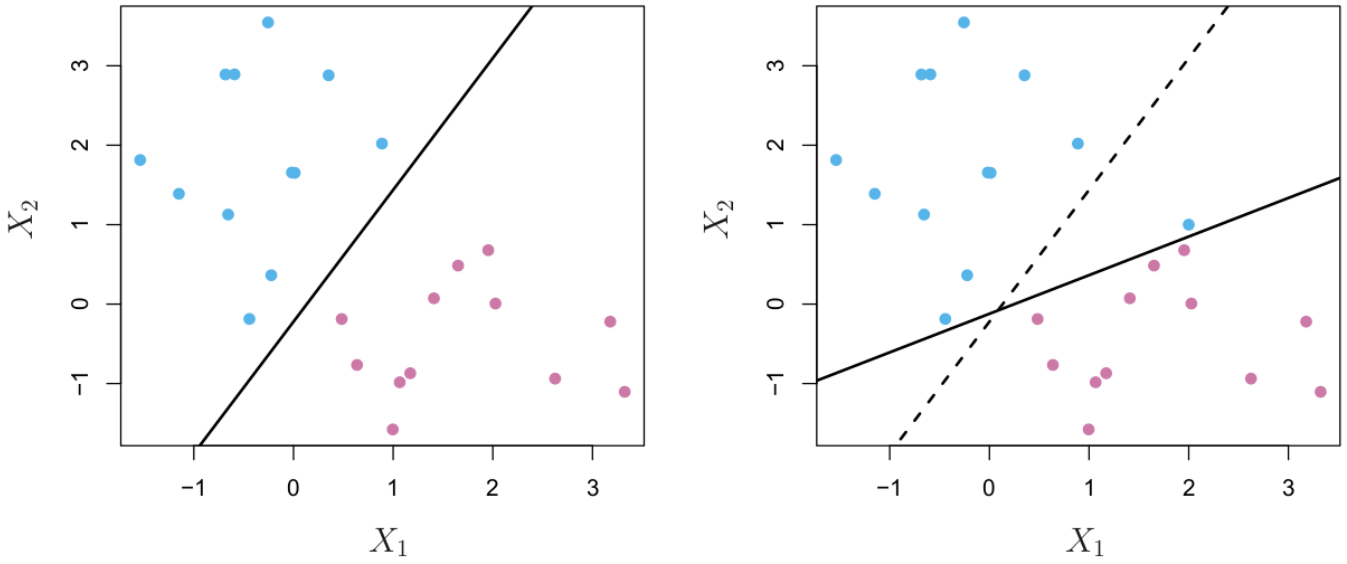


FIGURE 9.5. Left: *Two classes of observations are shown in blue and in purple, along with the maximal margin hyperplane.* Right: *An additional blue observation has been added, leading to a dramatic shift in the maximal margin hyperplane shown as a solid line. The dashed line indicates the maximal margin hyperplane that was obtained in the absence of this additional point.*

Incluso siendo un problema linealmente separable, podríamos preferir que se "equivoque" en algunas observaciones de entrenamiento

Introducción a SVMs

Ahora permitiremos que algunas observaciones estén del lado incorrecto del margen o incluso del lado incorrecto del hiperplano

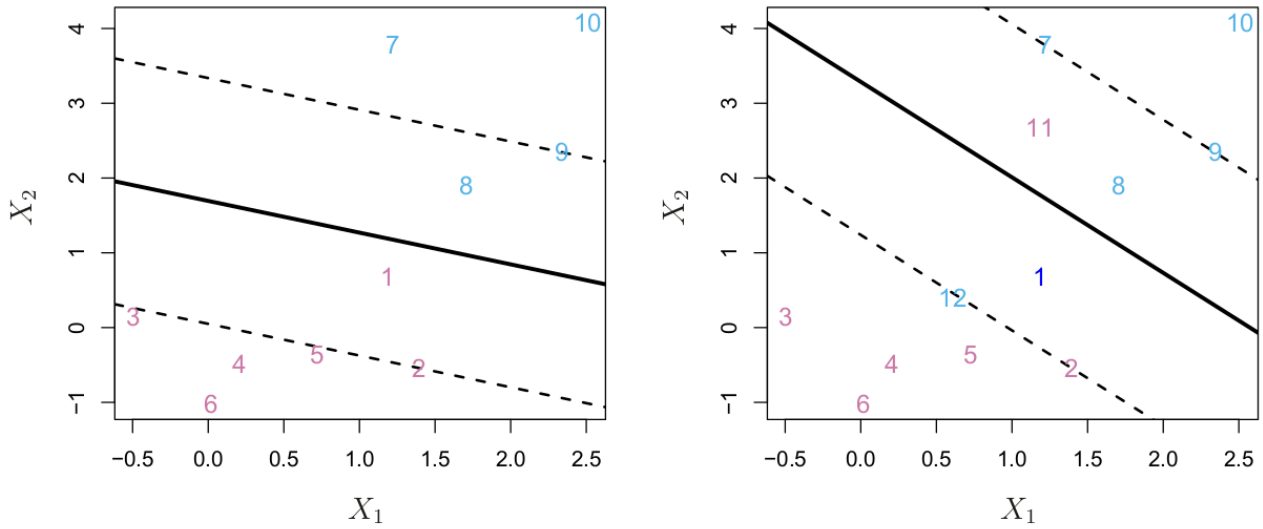


FIGURE 9.6. Left: *A support vector classifier was fit to a small data set. The hyperplane is shown as a solid line and the margins are shown as dashed lines.* Purple observations: *Observations* 3, 4, 5, *and* 6 *are on the correct side of the margin, observation* 2 *is on the margin, and observation 1 is on the wrong side of the margin.* Blue observations: *Observations* 7 *and* 10 *are on the correct side of the margin, observation* 9 *is on the margin, and observation* 8 *is on the wrong side of the margin. No observations are on the wrong side of the hyperplane.* Right: *Same as left panel with two additional points,* 11 *and* 12. *These two observations are on the wrong side of the hyperplane and the wrong side of the margin.*

Introducción a SVMs

Este comportamiento se obtiene al intentar a optimizar el siguiente problema de optimización:

$$\underset{\beta_0,\beta_1,\ldots,\beta_p,\epsilon_1,\ldots,\epsilon_n,\,M}{\text{maximize}} \quad M \tag{9.12}$$

$$\text{subject to} \quad \sum_{j=1}^{p} \beta_j^2 = 1, \tag{9.13}$$

$$y_i(\beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \cdots + \beta_p x_{ip}) \geq M(1 - \epsilon_i), \tag{9.14}$$

$$\epsilon_i \geq 0, \quad \sum_{i=1}^{n} \epsilon_i \leq C, \tag{9.15}$$

A las ε se las conoce como *slack variables*

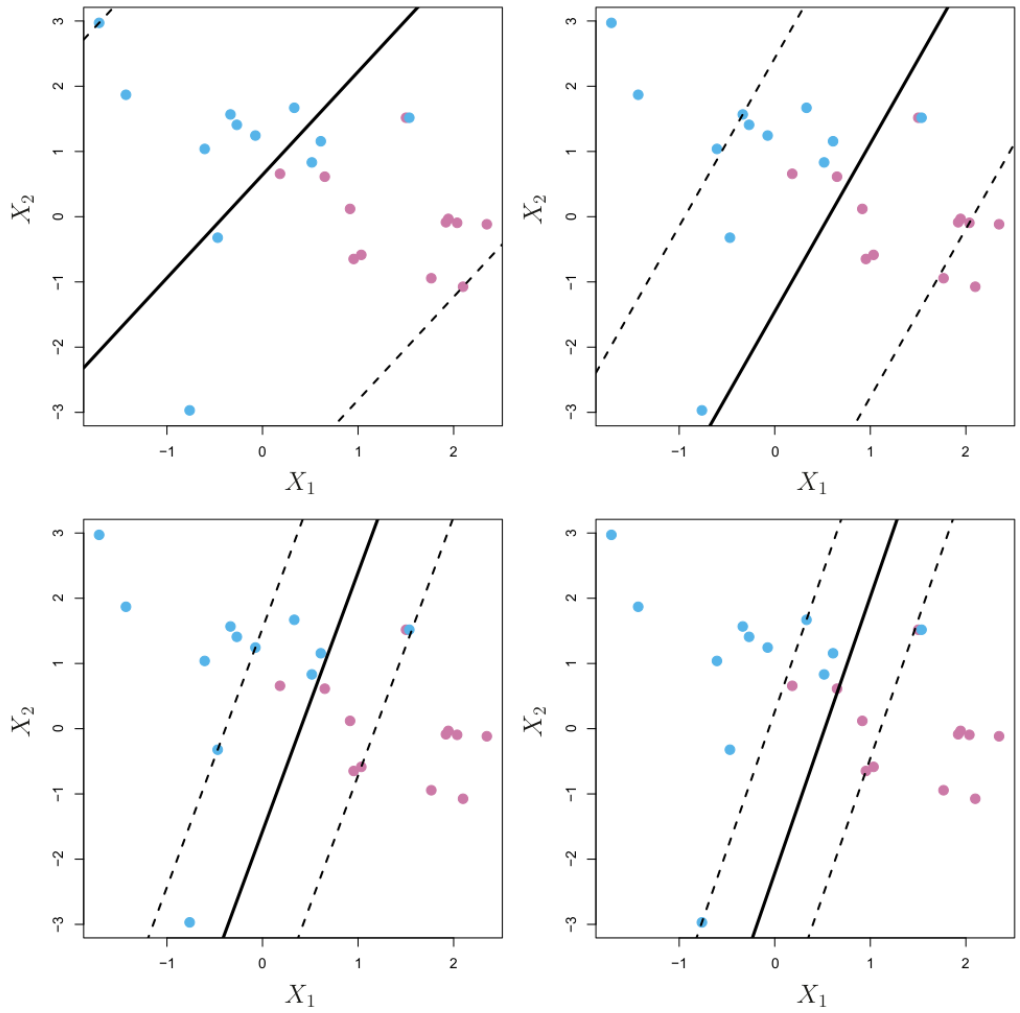
- $\varepsilon_i = 0 \rightarrow$ la observación i está del lado correcto del margen
- $\varepsilon_i > 0 \rightarrow$ la observación i está del lado incorrecto del margen
- $\varepsilon_i > 1 \rightarrow$ la observación i está del lado incorrecto del hiperplano

} Vectores de soporte + las $\varepsilon_i = 0$ en el margen

Más pequeño sea *C* (el "*budget*") → menos permitiremos que haya observaciones del lado incorrecto del margen

Introducción a SVMs

A menor valor de *C*, menos se toleran errores en entrenamiento



Introducción a SVMs

¿Qué sucede con los problemas que requieren fronteras no lineales?

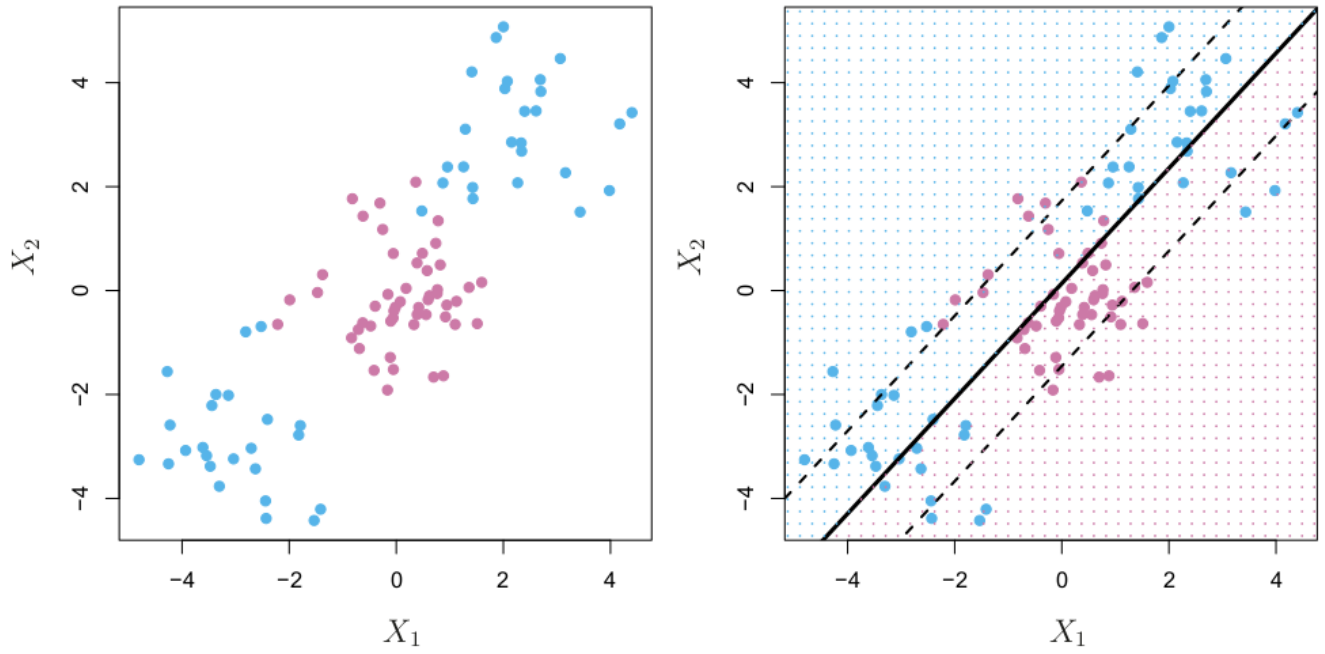


FIGURE 9.8. Left: *The observations fall into two classes, with a non-linear boundary between them.* Right: *The support vector classifier seeks a linear boundary, and consequently performs very poorly.*

Se podrían hacer expansiones polinómicas y se podría utilizar SVC (como venimos haciendo en regression logística)

Introducción a SVMs

La solución de 9.12 a 9.15, únicamente considera el producto interno de todos los pares de observaciones (el producto interno captura similitud)

$$\langle x_i, x_{i'} \rangle = \sum_{j=1}^{p} x_{ij} x_{i'j}$$

A su vez, el clasificador (la función utilizada para clasificar) puede expresarse como se muestra abajo (siendo *S* los vectores de sorporte):

$$f(x) = \beta_0 + \sum_{i \in \mathcal{S}} \alpha_i \langle x, x_i \rangle$$

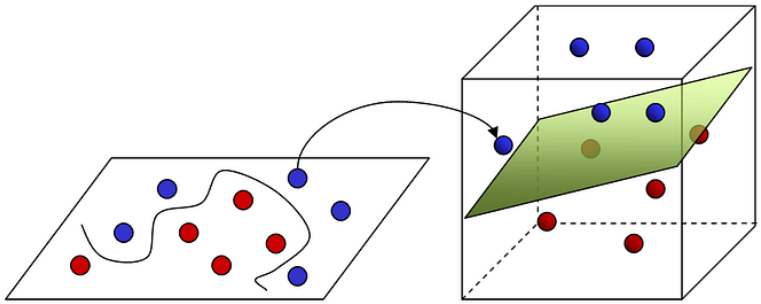
Kernel trick: si reemplazamos los productos internos tradicionales por generalizaciones del producto interno (*K*), se tiene que: 1) la solución de 9.12 a 9.15 sólo dependerá de $K(x_i, x_j)$ y 2) el clasificador vendrá dado por:

$$f(x) = \beta_0 + \sum_{i \in \mathcal{S}} \alpha_i K(x, x_i)$$

Introducción a SVMs

¿Qué es un kernel?

Una función que, dadas dos observaciones (en el espacio de atributos original), devuelve el valor del producto interno de dichas observaciones en un espacio expandido



Input Space

Recuerden:

- La solución (en el espacio expandido) sólo depende de los productos internos en dicho espacio (lo que devuelve el kernel)
- Para clasificar cada observación, alcanza con saber el valor del producto interno de dicha observación con cada vector de soporte (en el espacio expandido, lo que devuelve el kernel)

Introducción a SVMs

Polynomial kernel

$$K(x_i, x_{i'}) = (1 + \sum_{j=1}^{p} x_{ij} x_{i'j})^d$$

Si en toda expresión de SVC se reemplazan los productos internos por esta expression, se obtienen fronteras no lineales

Si se trabaja con *p* atributos originales y un grado *d*, es como si se hubiera estado trabajando con un espacio expandido de dimensión $\binom{p+d}{d}$

Introducción a SVMs

Radial kernel

$$K(x_i, x_{i'}) = \exp(-\gamma \sum_{j=1}^{p} (x_{ij} - x_{i'j})^2)$$

Si las observaciones son lejanas (en términos de distancia euclídea) → el valor del kernel será bajo

El espacio implícito de este kernel tiene dimensión infinita

Interesante: existen kernels que miden similitud de objetos muy variados (e.g., grafos, texto, imágenes, etc.), de este modo, eligiendo el kernel apropiado, se puede usar SVMs tomando como input estos tipo de datos

Introducción a SVMs

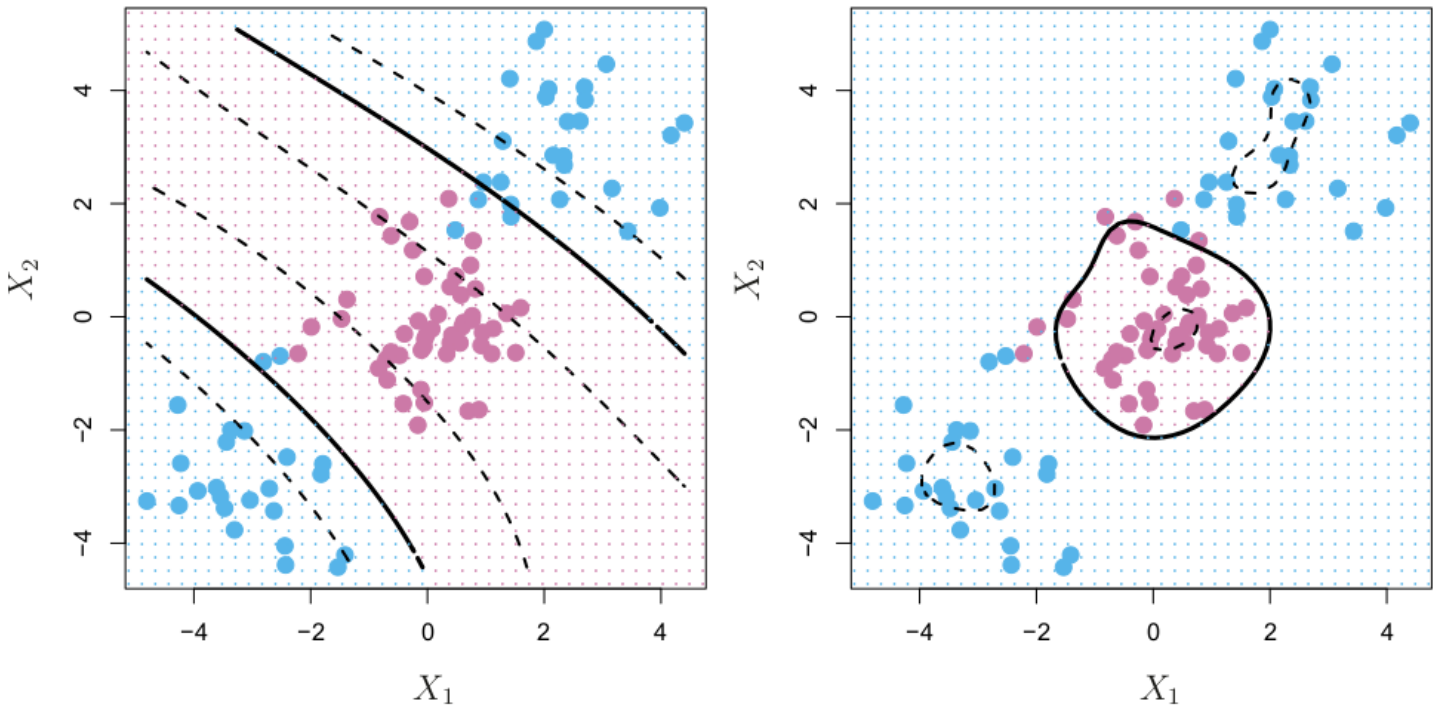


FIGURE 9.9. Left: *An SVM with a polynomial kernel of degree 3 is applied to the non-linear data from Figure 9.8, resulting in a far more appropriate decision rule.* Right: *An SVM with a radial kernel is applied. In this example, either kernel is capable of capturing the decision boundary.*

Introducción a SVMs

Probemoslo en Python

Bibliografía

- ISLP, Capítulo 4. Secciones 4.1, 4.2 y 4.3

- ISLP, Capítulo 9

- Avanzada:

  Alpaydin, E, “*Introduction to Machine Learning*”, Capítulo 13