Tema 7. Regresión con regularización

Regresión lineal: problemas de modelado (I)

- Hasta ahora hemos asumido que la variable dependiente es una combinación lineal de las features.
  - O en el caso de la regresión logística pasando a través de una función de tipo sigmoide

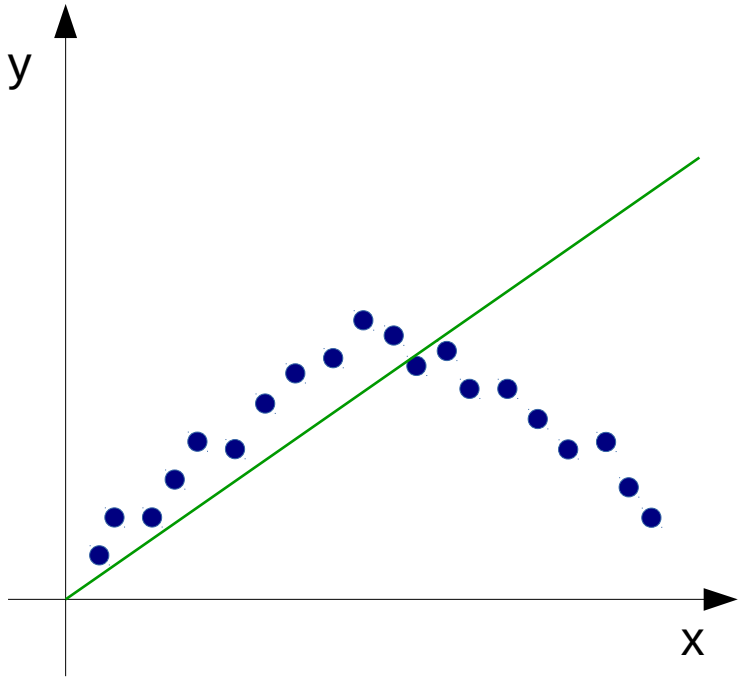
$$y = \alpha_0 + \alpha_1 x_1 + \alpha_2 x_2 + \ldots + \alpha_M x_M$$

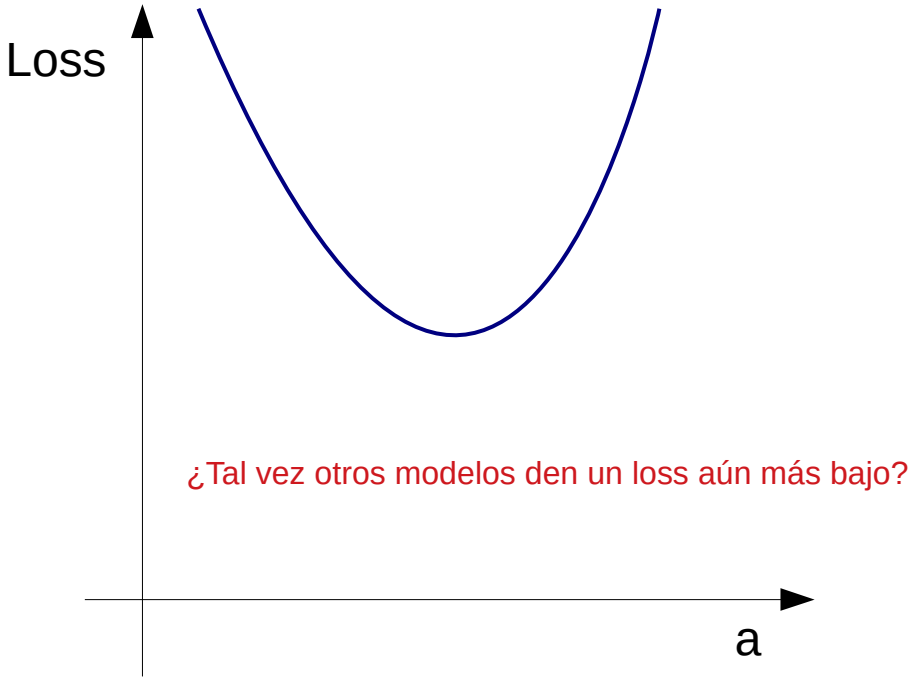
- Dado un modelo hemos visto cómo obtener valores óptimos de los parámetros que minimizan el coste.
- Sin embargo esto no garantiza que nuestro modelo describa los datos de manera correcta.
  - Sólo significa que de todos los posibles modelos con M parámetros, el calculado es el mejor.
- Resulta posible que nuestro modelo no tenga capacidad suficiente para realizar tal descripción.
- En ese caso necesitamos elegir modelos más complicados:
  - Una opción posible es aumentar (tal vez artificialmente, como vimos) el número de features.
  - En ocasiones la aproximación lineal es sencillamente insuficiente…
    - … siendo preciso el uso de modelos y tecnologías más avanzadas (Vector Machines, ANN, etc…)

Regresión lineal: problemas de modelado (II)

- Veamos un ejemplo sencillo en el caso del modelado de una variable dependiente continua “y”

Modelo: $y = a\,x$





- Resulta evidente que este modelo no ajusta bien los datos a pesar de que también tenga un mínimo.

El compromiso sesgo-varianza

- Vamos a suponer que “f” es la función interna que verdaderamente correlaciona “x” e “y”:

$$y^{(j)} = f(x^{(j)}) + \epsilon^{(j)}$$

- En donde hemos escrito explicitamente el residuo ε que es el término aleatorio que sale de pdf(y|x).

$$E[f(x)] = f \qquad E[y^{(j)} - f(x^{(j)})] = 0 \qquad E[\epsilon^{(j)}] = 0 \qquad Var[\epsilon^{(j)}] = \sigma^2 \qquad Var[y^{(j)}] = \sigma^2 \qquad E[y^{(j)}] = E[f(x^{(j)})]$$

- Ahora supongamos que contamos con una función”f̂“que simplemente aproxima “f” en cada punto j.

$$E[(y - \hat{f})^2] = E[(y^2 + \hat{f}^2 - 2y\hat{f})] = E[y^2] + E[\hat{f}^2] - E[2y\hat{f}] = Var(y) + Var(\hat{f}) + E[y]^2 + E[\hat{f}]^2 - E[2y\hat{f}]$$

- Como “y” y ”f̂“ son independientes podemos escribir la expresión de arriba como:

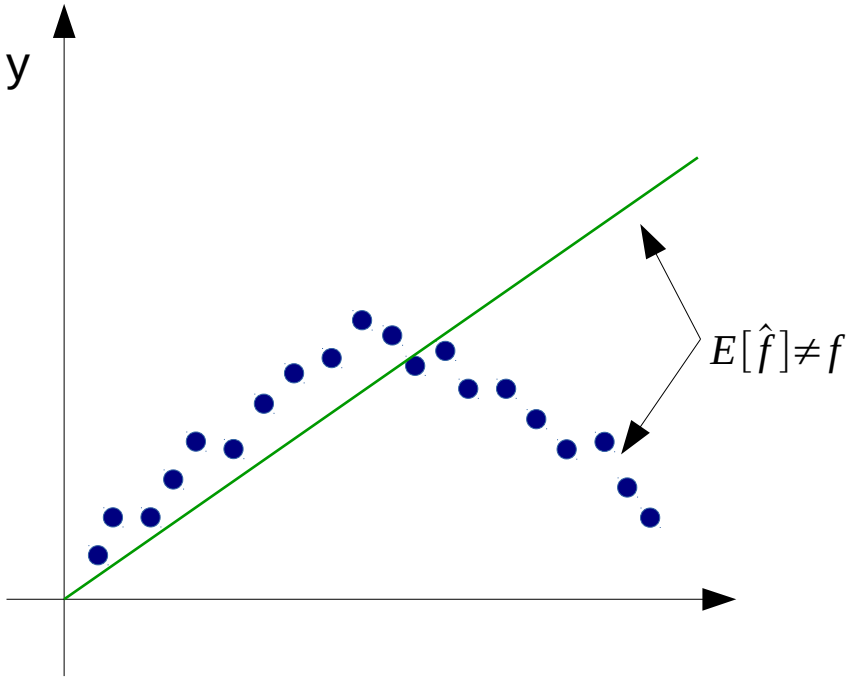
$$Var(y) + Var(\hat{f}) + E[y]^2 + E[\hat{f}]^2 - E[2y\hat{f}] = \sigma^2 + Var(\hat{f}) + f^2 + E[\hat{f}]^2 - 2fE[\hat{f}] = \sigma^2 + Var(\hat{f}) + (E[\hat{f}] - f)^2 = \sigma^2 + Var(\hat{f}) + Bias^2$$

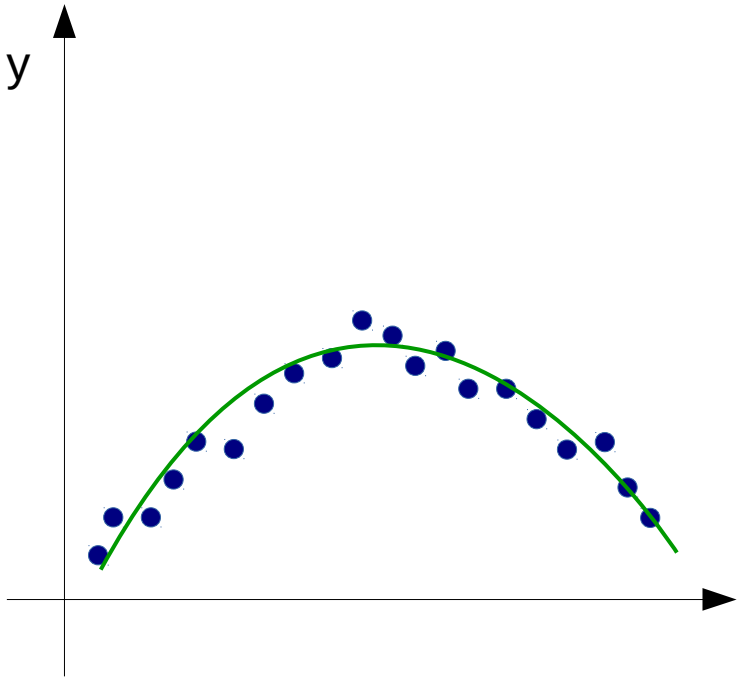
- Que nos describe la adecuación de la función en términos de su varianza y del posible sesgo.

Problema del sesgo (underfitting)

- El sesgo consiste en la falta de adecuación entre el modelo y el modelo supuestamente ideal.
- Consideremos el caso anterior en donde el modelo ideal es una parábola

Modelo: y = a x Modelo real: y = a x^2 + b

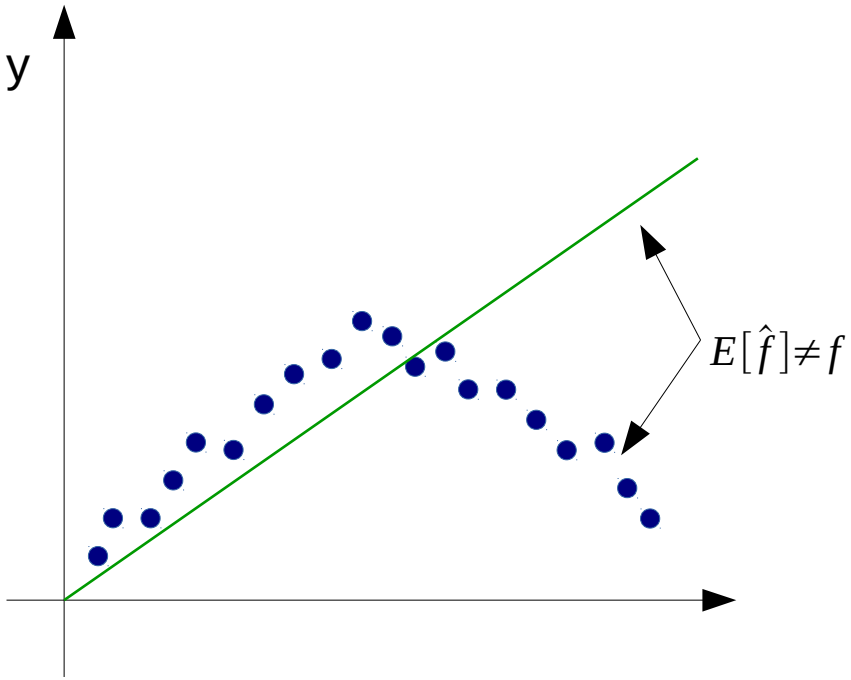




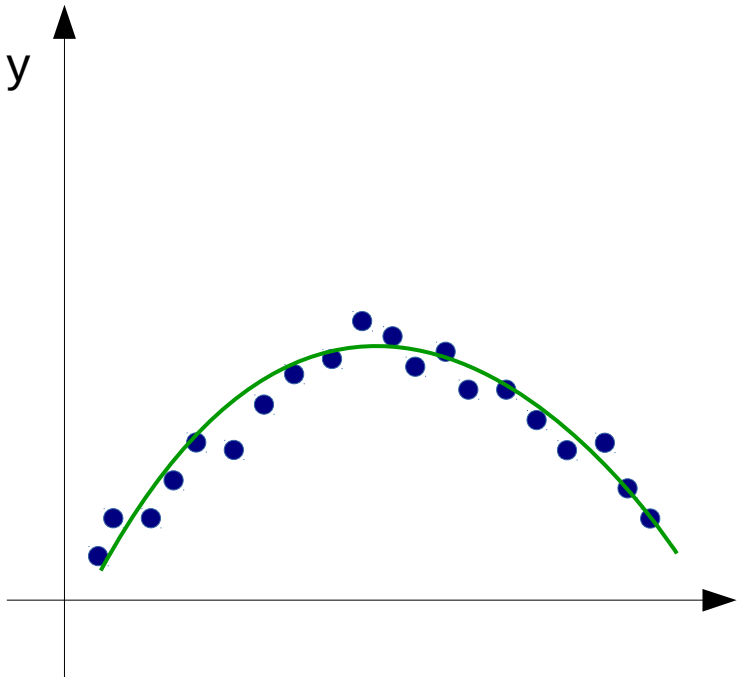
Problema del sesgo (underfitting)

- El sesgo consiste en la falta de adecuación entre el modelo y el modelo supuestamente ideal.
- El valor esperado del modelo (si repitiésemos la regresión infinitas veces…) difiere del modelo real
- Consideremos el caso anterior en donde el modelo ideal es una parábola

Modelo: y = a x

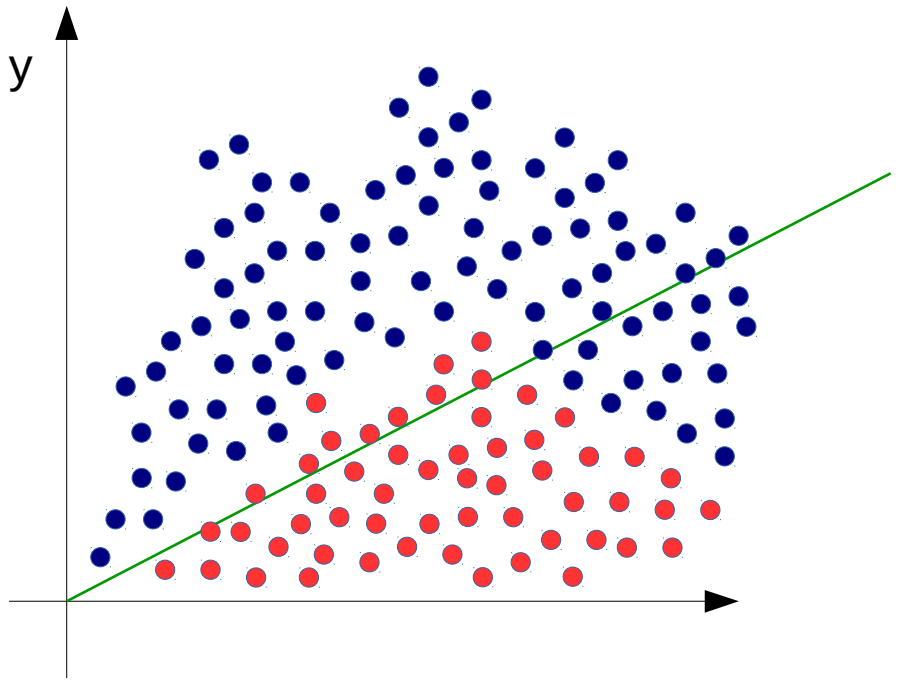


Modelo real: y = a x^2 + b



Problema del sesgo en clasificación (underfitting)

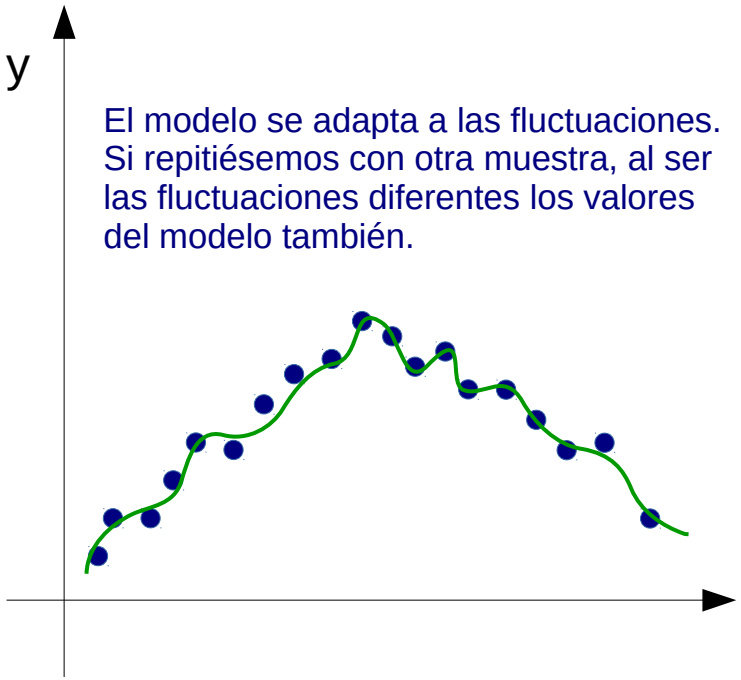
- El mismo efecto puede ocurrir cuando implementamos una clasificación.
- Evidentemente este efecto se traduce posteriormente en malos resultados de clasificación.



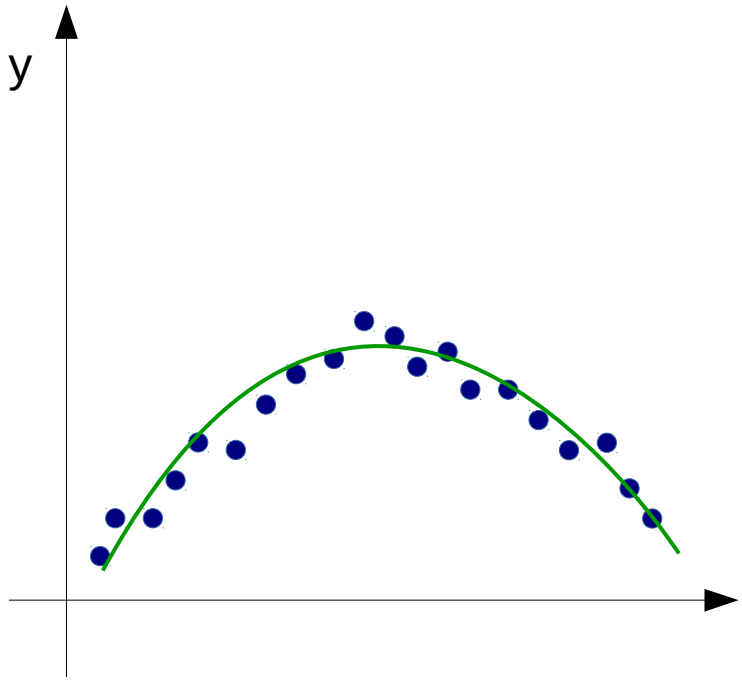
Problema de varianza (overfitting)

- El sesgo consiste en la excesiva capacidad del modelo para adaptarse al modelo ideal.
- La varianza del modelo (si repitiésemos la regresión infinitas veces…) es grande (da valores distintos).
- Consideremos el caso anterior en donde el modelo ideal es una parábola

Modelo: polinomio grado 15

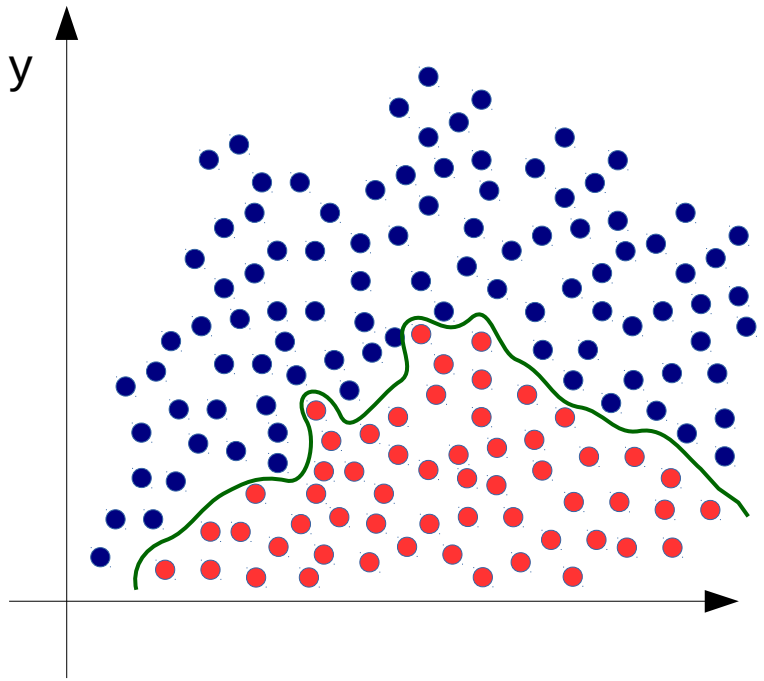


Modelo real: y = a x^2 + b



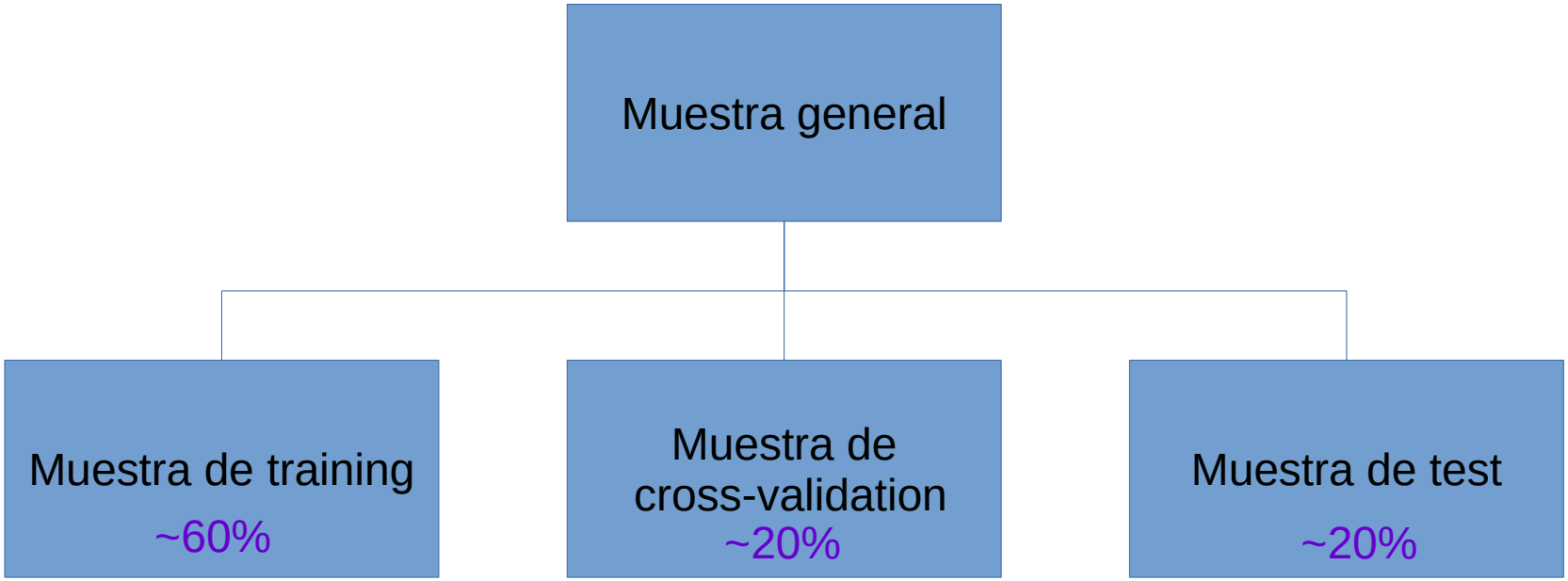
Problema de varianza en clasificación (underfitting)

- El mismo efecto puede ocurrir cuando implementamos una clasificación.
- Evidentemente este efecto se traduce posteriormente en malos resultados de clasificación.



Tecnicas de cross-validation para evaluar over/underfitting (I)

- Es posible utilizar técnicas de cross-validation para evaluar el sesgo/varianza de nuestro modelo.
- La aproximación más general consiste en dividir nuestra muestra en 3 componentes diferentes.



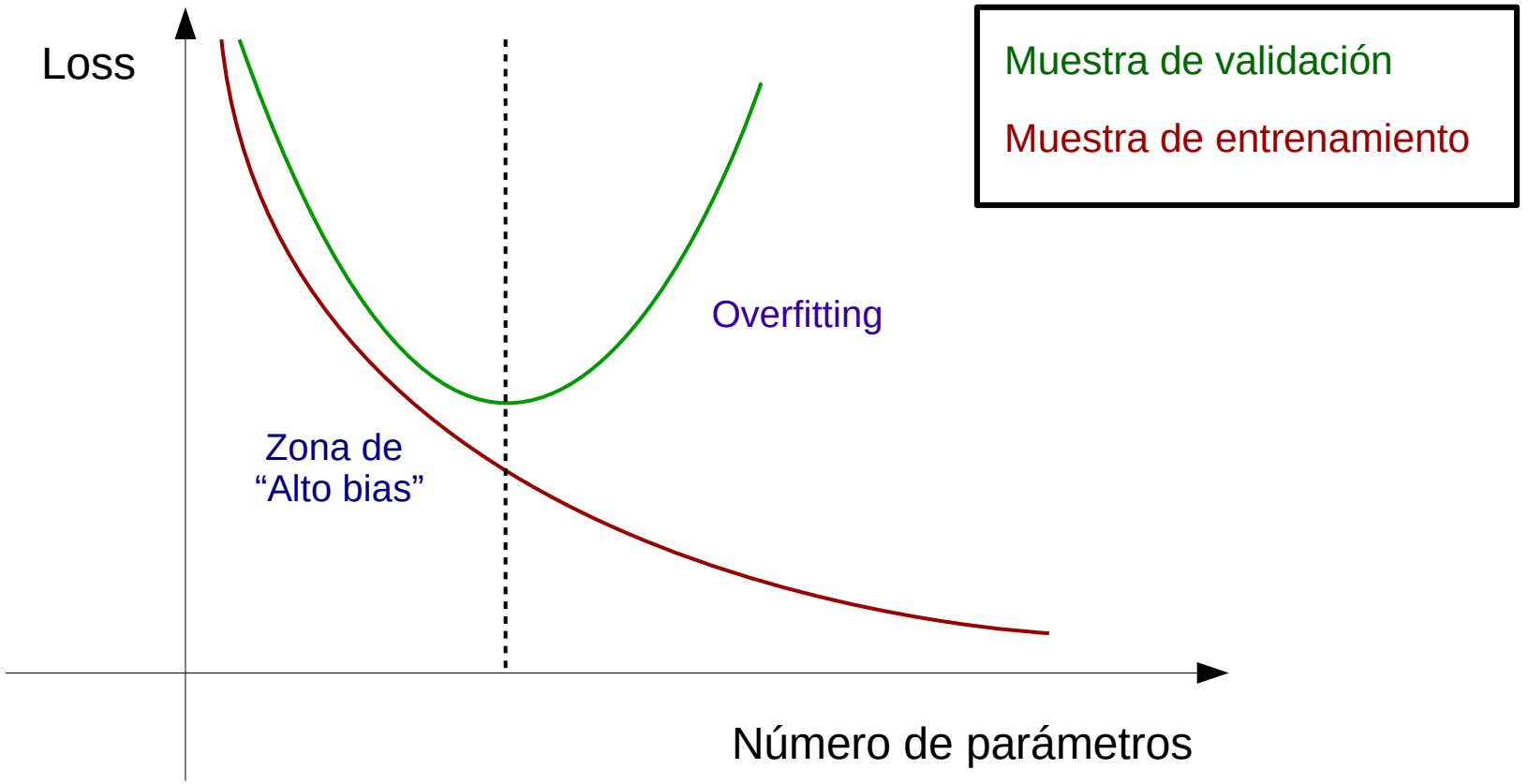
La usamos para entrenar nuestro modelo.

La usamos para evaluar el sesgo/bias y decidir cuál es el modelo más conveniente.

La usamos para evaluar el error de generlación que cometemos.

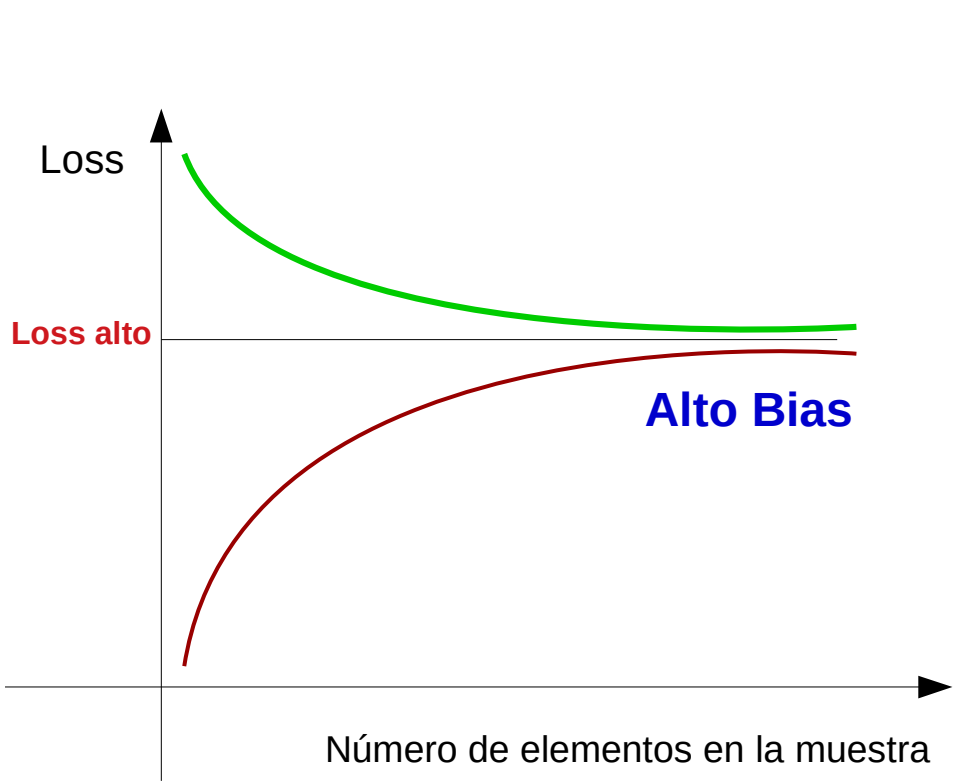
Tecnicas de cross-validation para evaluar over/underfitting (II)

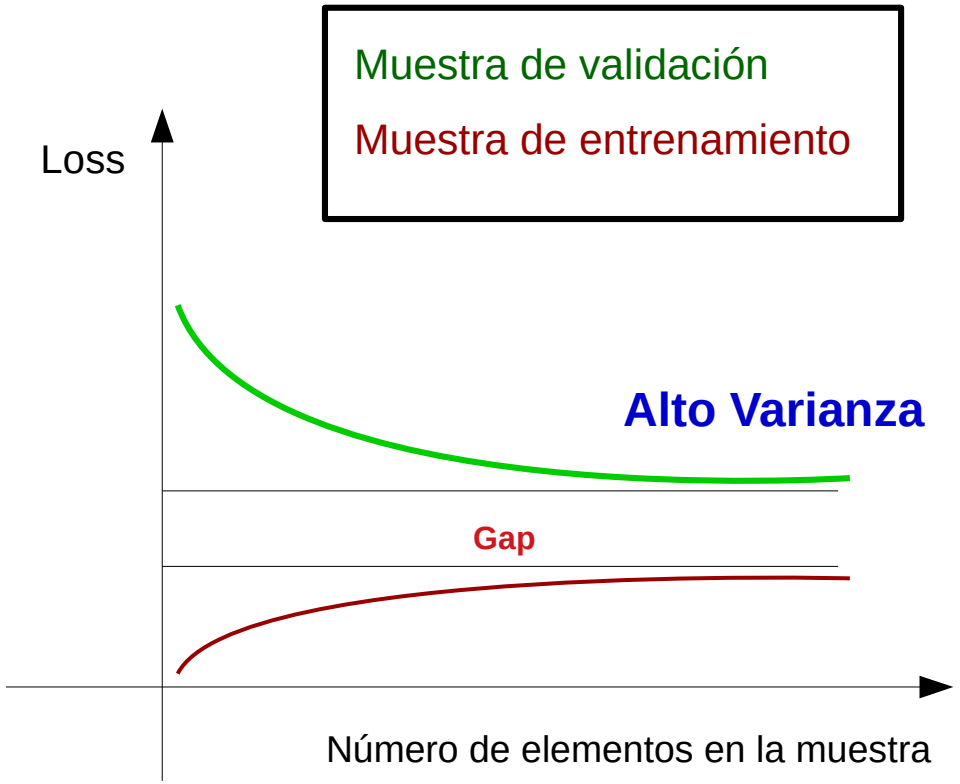
- El estudio del coste en una muestra de validación resulta fundamental para evaluar los problemas.
- Una de las curvas interesantes consiste en evaluar el coste en función del número de parámetros.



Tecnicas de cross-validation para evaluar over/underfitting (II)

- El estudio del coste en una muestra de validación resulta fundamental para evaluar los problemas.
- Otra curva interesante es la llamada learning curve (*) (coste en función del tamaño de la muestra)





(*) Para hacer esta curva es preciso que el loss esté normalizado por el número de elementos en la muestra

Solución a los problemas de sesgo y varianza

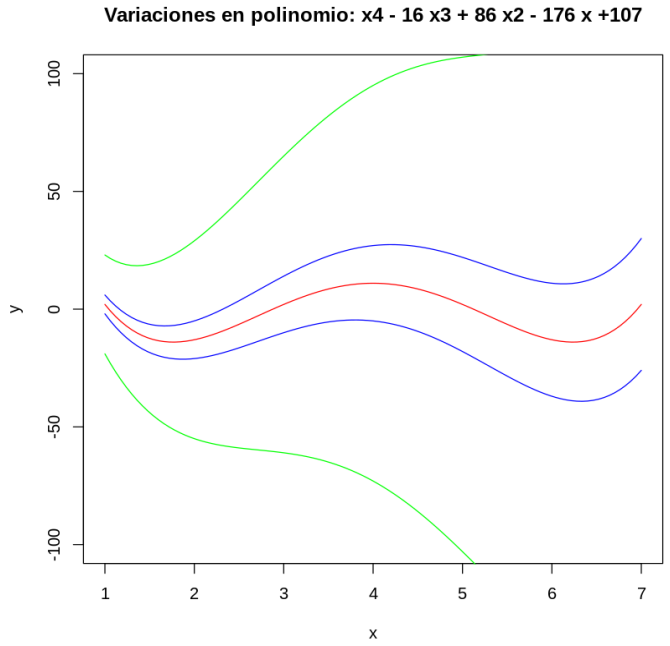
- Los dos problemas se deben a una deficiente elección de la función de modelado.
- Con frecuencia puede mejorarse **el sesgo** aumentando el número de parámetros de la función.
  - Recordemos como hicimos **la expansión de las features** a sus potencias para mejorar el modelo.
  - En muchos casos aumentar las features no basta: hay que buscar otras aproximaciones no lineales.
- El problema de **la varianza** es el caso contrario: hay que reducir el número de parámetros.
  - A esto se le llama **"Reducción de dimensión"** y consiste en prescindir de algunas features.
  - En algunas ocasiones aumentar el tamaño de la muestra también puede ayudar.
  - También se pueden utilizar **técnicas de regularización** (como veremos más adelante).

Regularización (Ridge regression)

- Una forma de combatir el overfitting consiste en impedir el crecimiento (alto) de los parámetros.
- En modelos polinomiales valores altos de los parámetros hacen diverger a la función rápidamente.

+/-3% en el cuarto parámetro

+/-10% en el cuarto parámetro



$$Loss(x, y; \alpha) = Loss(x, y; \alpha) + \lambda \alpha^T \alpha$$

Si los parámetros crecen, el loss también crece.

- Idealmente nos gustaría contar con un término que controlase el crecimiento de los parámetros.
- Una manera de hacerlo consiste en añadir a la función de coste un térmiono $\lambda \alpha \alpha$

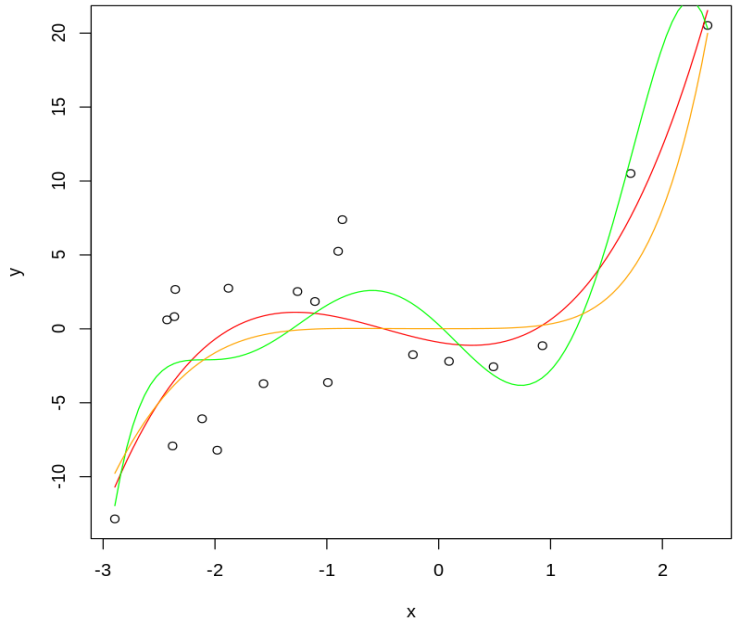
Regularización para regresión lineal

- En el caso de la regresión lineal podemos fácilmente adaptar la función de coste:

$$Loss=(y-X\,\alpha)^T(y-X\,\alpha)+\lambda\,\alpha^T\alpha$$

- De la misma forma que hicimos, podemos también calcular el gradiente e igualarlo a 0.

$$\nabla_\alpha Loss=-X^T(y-X\,\alpha)=-X^T y+X^T X\,\alpha+\lambda\,\alpha=0 \qquad \alpha=(X^T X+\lambda)^{-1}X^T y$$



Solucion con el modelo correcto
Overfitting no regularization
Overfitting with regularization

Regularización para regresión logística

- En el caso de la regresión logística podemos fácilmente adaptar la función de coste:

$$Loss = \frac{-1}{N} \sum_{i=0}^{N} y^{(i)} \log\left(\sigma(\alpha_0 + \alpha_1 x_1^{(j)} + \ldots + \alpha_M x_M^{(j)})\right) + (1 - y^{(i)}) \log\left(1 - \sigma(\alpha_0 + \alpha_1 x_1^{(j)} + \ldots + \alpha_M x_M^{(j)})\right) + \frac{1}{N} \lambda \alpha^T \alpha$$

- De la misma forma que hicimos, podemos también calcular el gradiente.

$$\nabla Loss = \frac{1}{N} \sum_{j=1}^{N} (\sigma(\alpha^T x^{(j)}) - y^{(j)}) x^{(j)} + \frac{1}{N} \lambda \alpha$$

- ¿Cómo podemos elegir lambda? Es algo a decidir mirando a la muestra de cross-validation

Ejercicio 7

- Escribe una función de R que reciba como input un vector x con la variable dependiente, un vector de parámetros alpha que contenga los coeficientes de un polinomio y una sigma “sigma”; y que devuelva el vector dependiente que siga la ley polinomial dada con pdf gaussiana y sigma dada.
- Construye un vector de 40 elementos distribuido uniformemente entre [-3, 3]. Usalo con la función anterior y un polinomio con parámetros: 1 x3 + 2 x2 - x - 2 and sigma = 4. Pinta la función.
- Calcula el valor mínimo de la función de loss para el caso en que hacemos un ajuste con una recta (dos parámetros), una parábola (tres parámetros), 4, 5, 6 y 7 parámetros, usando los primeros 20 puntos. Con los valores que hacen mínimo el Loss para la training sample, calcula el Loss para los siguientes 20 puntos. Pinta los resultados en función del número de parámetros para ambos casos.
- Escribe una función de R que encuentre el mínimo de la función de coste para el caso de “regularización”. El parámetro lambda será pasado como input. Utilízala para estimar las curvas que mejor ajustan para el caso de (4 parámetros, lambda = 0), (7 parámetros, lambda=0) y (7 parámetros, lambda=1000). Pinta todas las curvas juntas y escribe tus conclusiones.