Ciencias de Datos con R: Fundamentos Estadísticos

Ana M. Bianco, Jemina García y Mariela Sued.

Predicción - Parte II

Algunas referencias

Wasserman, L. (2013). All of statistics: a concise course in statistical inference. Springer Science & Business Media.

James, G., Witten, D., & Hastie, T. (2014). An Introduction to Statistical Learning: With Applications in R. http://www-bcf.usc.edu/ gareth/ISL/

Izbicki, R., & dos Santos, T. M. Machine Learning sob a ótica estatística.

Efron, B., & Hastie, T. (2016). Computer Age Statistical Inference (Vol. 5). Cambridge University Press.

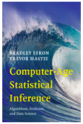
Wasserman, L. (2006). All of nonparametric statistics. Springer Science & Business Media.

Trevor, H., Robert, T., & JH, F. (2009). The elements of statistical learning: data mining, inference, and prediction. https://web.stanford.edu/ hastie/Papers/ESLII.pdf

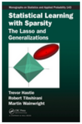
Hastie

MY PUBLICATIONS

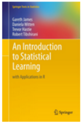
BOOKS



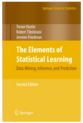
Computer Age Statistical Inference:Algorithms, Evidence and Data Science
by Bradley Efron and Trevor Hastie (August 2016)
Book Homepage
pdf (8.5 Mb, corrected online)



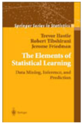
Statistical Learning with Sparsity: the Lasso and Generalizations
by Trevor Hastie, Robert Tibshirani and Martin Wainwright (May 2015)
Book Homepage
pdf (10.5Mb, corrected online)



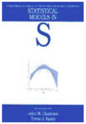
An Introduction to Statistical Learning *with Applications in R*
by Gareth James, Daniela Witten, Trevor Hastie and Robert Tibshirani (June 2013)
Book Homepage
pdf (9.4Mb, 6th corrected printing)



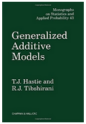
The Elements of Statistical Learning: Data Mining, Inference, and Prediction (Second Edition)
by Trevor Hastie, Robert Tibshirani and Jerome Friedman (2009)
Book Homepage
pdf (13Mb, correct. 12th print)



The Elements of Statistical Learning: Data Mining, Inference, and Prediction
by Trevor Hastie, Robert Tibshirani and Jerome Friedman (2001)



Statistical Models in S
edited by John Chambers and Trevor Hastie (1991)



Generalized Additive Models
by Trevor Hastie and Robert Tibshirani (1990)

Galton

ANTHROPOLOGICAL MISCELLANEA.

REGRESSION *towards* MEDIOCRITY *in* HEREDITARY STATURE.
By FRANCIS GALTON, F.R.S., &c.

[WITH PLATES IX AND X.]

Repaso de Probabilidad - (sin datos)

Predicción sin variables explicativas (error cuadrático)

- Y variable respuesta.
- Esperanza de Y: $\mu = \mathbb{E}(Y)$
- Esperanza desde la predicción.

$$\mu = \arg\min_{a} \mathbb{E}\{(Y-a)^2\}.$$

Predicción - Error cuadrático

- Y: variable respuesta, $\mathbf{X}$: variables explicativas, $g(\mathbf{X})$ posible predictor.
- Error error cuadrático medio al predecir con g:

$$\mathbb{E}\left[\{Y - g(\mathbf{X})\}^2\right].$$

- Mejor predictor: $r(\mathbf{X})$ satisfaciendo

$$\mathbb{E}\left[\{Y - r(\mathbf{X})\}^2\right] \leq \mathbb{E}\left[\{Y - g(\mathbf{X})\}^2\right] \ , \quad \forall g : \mathbb{R}^p \to \mathbb{R}$$

$r(\mathbf{X})$ minimiza el error cuadrático medio de predicción

$$r(\mathbf{x}) = \mathbb{E}(Y \mid \mathbf{X} = \mathbf{x}).$$

...the conditional expectation, also known as the regression function. (SL sin R)

Función de regresion $r(\mathbf{X})$ - A la carta

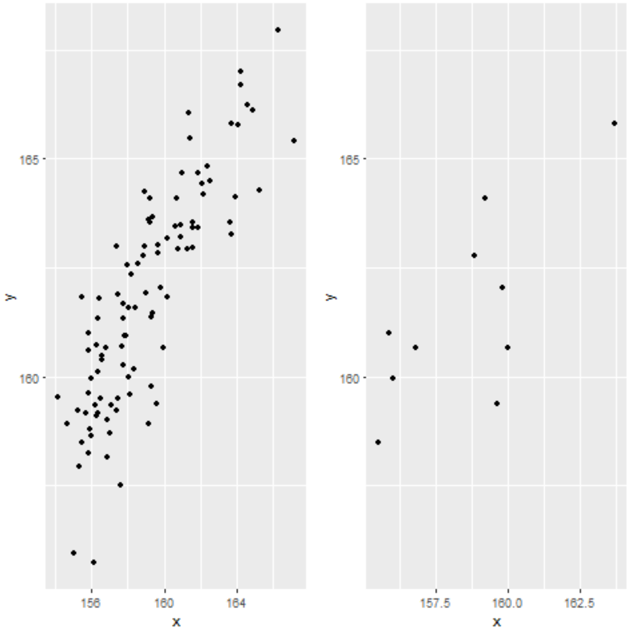
$$(\mathbf{X}, Y) \sim \mathcal{P} \,, \quad r(\mathbf{X}) = \mathbb{E}(Y \mid \mathbf{X})$$

$Y := r(\mathbf{X}) + \varepsilon$, $\mathbf{X}$ independiente de ε , $\mathbb{E}(\varepsilon) = 0$.

$$\mathbb{E}\left[\{Y - r(\mathbf{X})\}^2\right] \leq \mathbb{E}\left[\{Y - g(\mathbf{X})\}^2\right] \,, \quad \forall g : \mathbb{R}^p \to \mathbb{R}$$

Predecimos con $r(\mathbf{X})$.

Posibles Escenarios



Predicción: Estimación de $r(\mathbf{X})$ - Muchos Datos

$$(\mathbf{X}_1, Y_1), \ldots, (\mathbf{X}_n, Y_n) \quad \text{iid} \ , (\mathbf{X}_i, Y_i) \sim P$$

$$\widehat{r}(\ \cdot\ ) = \widehat{r}_n(\ \cdot\ ) \quad \text{construído con } \{(\mathbf{X}_1, Y_1), \ldots, (\mathbf{X}_n, Y_n)\}$$

Predecimos con $\widehat{r}_n(\mathbf{X})$.

Dos propuestas:

- Nadaraya-Watson. ventana: h. $\widehat{r}_h(\mathbf{x})$
- Vecinos próximos (knn). vecinos: k. $\widehat{r}_k(\mathbf{x})$

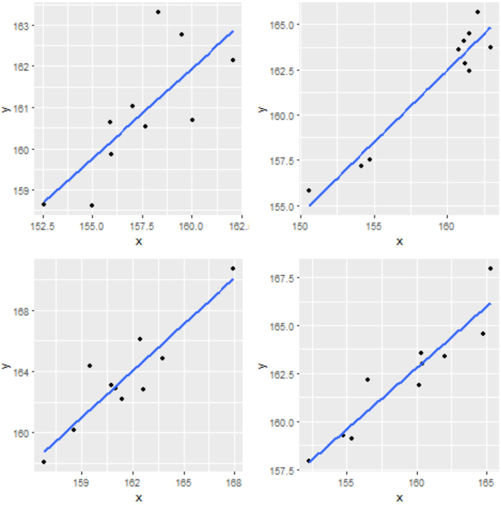
¿Por qué hacer otra cosa?

In light of this, why look further, since it seems we have a universal approximator? We often do not have very large samples. If the linear or some more structured model is appropriate, then we can usually get a more **stable estimate** *than k-nearest neighbors, although such knowledge has to be learned from the data as well. There are other problems though, ... The convergence above still holds, but the rate of convergence decreases as the dimension increases.*(SL sin R)

a vizinhança de x com alta probabilidade é vazia.(Rafael)

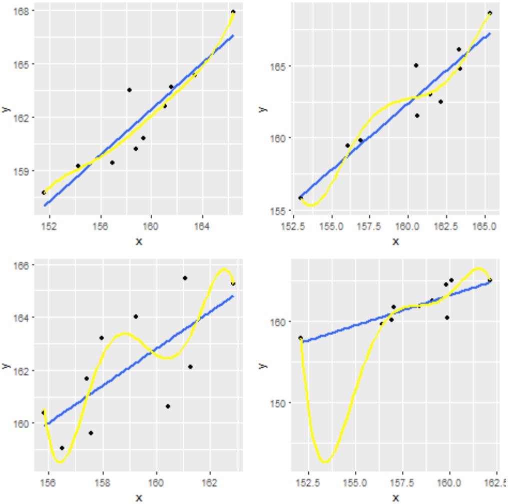
Cambiando de muestra - recta de mínimos cuadrados

Modelo más estructurado, más estable



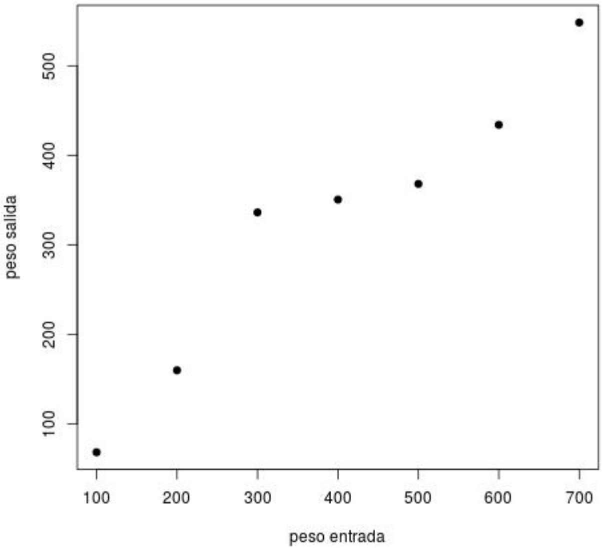
Cambiando de muestra

Modelo más flexible, menos estable

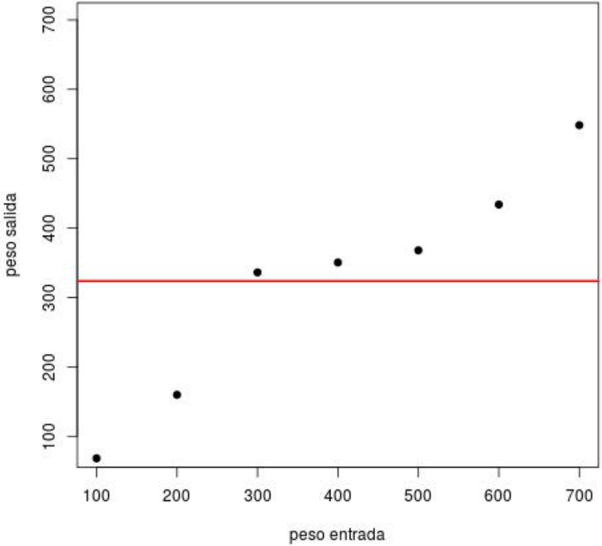


EL PRINCIPIO DE PARSIMONIA

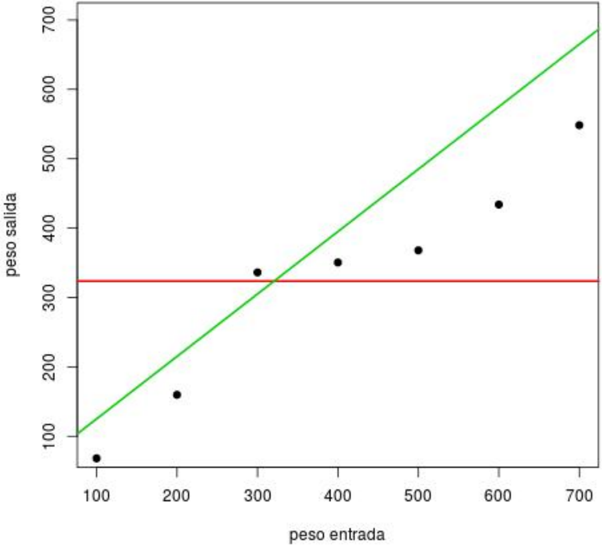
Regresión lineal simple - ejemplo:



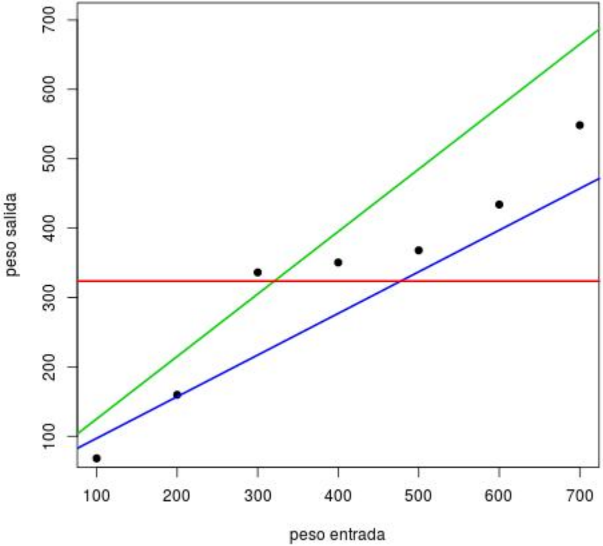
Algunos gráficos:



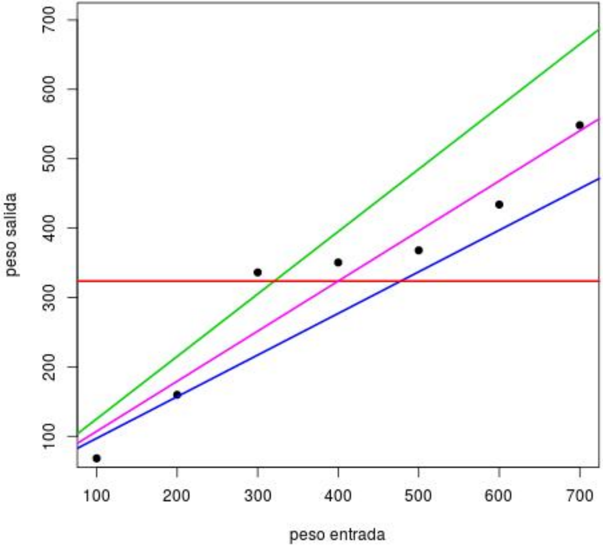
Algunos gráficos:



Algunos gráficos:



Algunos gráficos:



Regresión Lineal - simple - 1 variable explicativa

- Asumimos que $r(\mathbf{x}) = \beta_0^* + \beta_1^* \mathbf{x}$
- Estimación: mínimos cuadrados -

$$(\widehat{\beta}_0, \widehat{\beta}_1) = \arg\min_{\beta_0, \beta_1} \sum_{i=1}^{n} \{Y_i - (\beta_0 + \beta_1 \mathbf{X}_i)\}^2$$

- Predicción: $\widehat{r}(\mathbf{x}) = \widehat{\beta}_0 + \widehat{\beta}_1 \mathbf{x}$

Regresión Lineal - simple - 1 variable explicativa

- Asumimos que $r(\mathbf{x}) = \beta_0^* + \beta_1^* \mathbf{x}$
- Estimación: mínimos cuadrados -

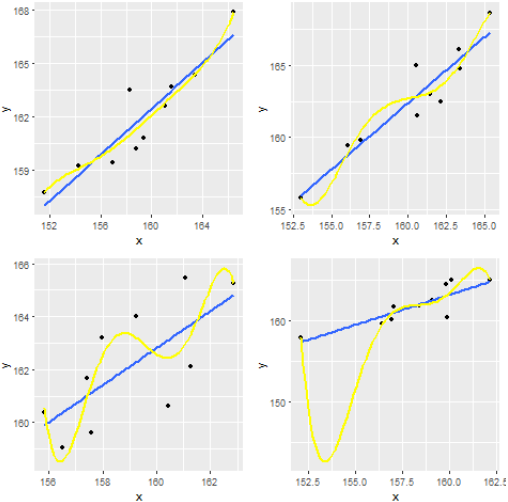
$$(\widehat{\beta}_0, \widehat{\beta}_1) = \arg\min_{\beta_0, \beta_1} \sum_{i=1}^{n} \{Y_i - (\beta_0 + \beta_1 \mathbf{X}_i)\}^2$$

- Predicción: $\widehat{r}(\mathbf{x}) = \widehat{\beta}_0 + \widehat{\beta}_1 \mathbf{x}$
- ¿Qué pasa si el modelo es incorrecto? Definimos

$$(\beta_0^*(F), \beta_1^*(F)) := \arg\min_{\beta_0, \beta_1} \mathbb{E}_F(\{Y - (\beta_0 + \beta_1 \mathbf{X})\}^2) \, , (\mathbf{X}, Y) \sim F$$

- Mejor predictor lineal: $\beta_0^*(F) + \beta_1^*(F)\mathbf{x} \neq r(\mathbf{x})$
- $\widehat{r}(\mathbf{x}) \to \beta_0^*(F) + \boldsymbol{\beta}^*(F)\mathbf{x}$ Mejor predictor lineal, SIN SER $r(\mathbf{x})$.

De la recta a los polinomios



Regresión polinomial - 1 variable explicativa - (ej grado 3)

- Asumimos que $r(\mathbf{x}) = \beta_0^* + \beta_1^* \mathbf{x} + \beta_2^* \mathbf{x}^2 + \beta_3^* \mathbf{x}^3$
- Estimación: mínimos cuadrados -

$$(\widehat{\beta}_0, \widehat{\beta}_1, \widehat{\beta}_2, \widehat{\beta}_3) = \arg \min_{\beta_0, \beta_1 \beta_2, \beta_3} \sum_{i=1}^{n} \{Y_i - (\beta_0 + \beta_1 \mathbf{X}_i + \beta_2 \mathbf{X}_i^2 + \beta_3 \mathbf{X}_i^3)\}^2$$

- Predicción: $\widehat{r}(\mathbf{x}) = \widehat{\beta}_0 + \widehat{\beta}_1 \mathbf{x} + \widehat{\beta}_2 \mathbf{x}^2 + \widehat{\beta}_3 \mathbf{x}^3$

Regresión polinomial - 1 variable explicativa - (ej grado 3)

- Asumimos que $r(\mathbf{x}) = \beta_0^* + \beta_1^*\mathbf{x} + \beta_2^*\mathbf{x}^2 + \beta_3^*\mathbf{x}^3$
- Estimación: mínimos cuadrados -

$$(\widehat{\beta}_0, \widehat{\beta}_1, \widehat{\beta}_2, \widehat{\beta}_3) = \arg \min_{\beta_0, \beta_1 \beta_2, \beta_3} \sum_{i=1}^{n} \{Y_i - (\beta_0 + \beta_1 \mathbf{X}_i + \beta_2 \mathbf{X}_i^2 + \beta_3 \mathbf{X}_i^3)\}^2$$

- Predicción: $\widehat{r}(\mathbf{x}) = \widehat{\beta}_0 + \widehat{\beta}_1\mathbf{x} + \widehat{\beta}_2\mathbf{x}^2 + \widehat{\beta}_3\mathbf{x}^3$
- ¿Qué pasa si el modelo es incorrecto?
- *Convergemos al mejor polinomio cúbico para predecir a Y.*

Regresión Lineal - Múltiple - $\mathbf{X} = (\mathbf{X}_1, \ldots, \mathbf{X}_p)$

- Asumimos que $r(\mathbf{x}) = \beta_0^* + \beta_1^* \mathbf{x}_1 + \ldots + \beta_p^* \mathbf{x}_p$
- Estimación: mínimos cuadrados - $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_p)$

$$(\widehat{\beta}_0, \widehat{\boldsymbol{\beta}}) = \arg\min_{\beta_0, \boldsymbol{\beta}} \sum_{i=1}^{n} \{Y_i - (\beta_0 + \boldsymbol{\beta}^t \mathbf{X}_i)\}^2$$

- Predicción: $\widehat{r}(\mathbf{x}) = \widehat{\beta}_0 + \widehat{\boldsymbol{\beta}}^t \mathbf{x}$

Regresión Lineal - Múltiple - $\mathbf{X} = (\mathbf{X}_1, \ldots, \mathbf{X}_p)$

- Asumimos que $r(\mathbf{x}) = \beta_0^* + \beta_1^* \mathbf{x}_1 + \ldots + \beta_p^* \mathbf{x}_p$
- Estimación: mínimos cuadrados - $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_p)$

$$(\widehat{\beta}_0, \widehat{\boldsymbol{\beta}}) = \arg\min_{\beta_0, \boldsymbol{\beta}} \sum_{i=1}^{n} \{Y_i - (\beta_0 + \boldsymbol{\beta}^t \mathbf{X}_i)\}^2$$

- Predicción: $\widehat{r}(\mathbf{x}) = \widehat{\beta}_0 + \widehat{\boldsymbol{\beta}}^t \mathbf{x}$
- ¿Qué pasa si el modelo es incorrecto? Definimos

$$(\beta_0^*(F), \boldsymbol{\beta}^*(F)) := \arg\min_{\beta_0, \boldsymbol{\beta}} \mathbb{E}_F(\{Y - (\beta_0 + \boldsymbol{\beta}^t \mathbf{X})\}^2) \,, (\mathbf{X}, Y) \sim F$$

- Mejor predictor lineal: $\beta_0^*(F) + \boldsymbol{\beta}^*(F)^t \mathbf{x} \neq r(\mathbf{x})$
- $\widehat{r}(\mathbf{x}) \to \beta_0^*(F) + \boldsymbol{\beta}^*(F)^t \mathbf{x}$ Mejor predictor lineal, SIN SER $r(\mathbf{x})$.

Modelando la regresión

- Sea $r_{\boldsymbol{\theta}} : \mathbb{R}^p \to \mathbb{R}$
- Familia $\mathcal{F} = \{r_{\boldsymbol{\theta}}, \boldsymbol{\theta} \in \Theta\}$.
- Familia paramétrica: $\Theta \subseteq \mathbb{R}^k$.

$$\boldsymbol{\theta}^* = \arg\min_{\boldsymbol{\theta} \in \Theta} \mathbb{E}\left[\{Y - r_{\boldsymbol{\theta}}(\mathbf{X})\}^2\right]$$

- $r_{\boldsymbol{\theta}^*}(\mathbf{X})$ es el mejor predictor en la clase.

Modelando la regresión

- Sea $r_{\boldsymbol{\theta}} : \mathbb{R}^p \to \mathbb{R}$
- Familia $\mathcal{F} = \{r_{\boldsymbol{\theta}}, \boldsymbol{\theta} \in \Theta\}$.
- Familia paramétrica: $\Theta \subseteq \mathbb{R}^k$.

$$\boldsymbol{\theta}^* = \arg\min_{\boldsymbol{\theta} \in \Theta} \mathbb{E}\left[\{Y - r_{\boldsymbol{\theta}}(\mathbf{X})\}^2\right]$$

- $r_{\boldsymbol{\theta}^*}(\mathbf{X})$ es el mejor predictor en la clase.

$$\widehat{\theta}_n := \arg\min_{\theta \in \Theta} \sum_{i=1}^{n} \{Y_i - r_{\boldsymbol{\theta}}(\mathbf{X}_i)\}^2 \ , \ \ \widehat{\boldsymbol{\theta}}_n \to \boldsymbol{\theta}^* \ , \ \ r_{\widehat{\boldsymbol{\theta}}_n}(\mathbf{X}) \to r_{\boldsymbol{\theta}^*}(\mathbf{X})$$

Modelando la regresión

- Sea $r_{\boldsymbol{\theta}} : \mathbb{R}^p \to \mathbb{R}$
- Familia $\mathcal{F} = \{r_{\boldsymbol{\theta}}, \boldsymbol{\theta} \in \Theta\}$.
- Familia paramétrica: $\Theta \subseteq \mathbb{R}^k$.

$$\boldsymbol{\theta}^* = \arg\min_{\boldsymbol{\theta}\in\Theta} \mathbb{E}\left[\{Y - r_{\boldsymbol{\theta}}(\mathbf{X})\}^2\right]$$

- $r_{\boldsymbol{\theta}^*}(\mathbf{X})$ es el mejor predictor en la clase.

$$\widehat{\theta}_n := \arg\min_{\theta\in\Theta} \sum_{i=1}^{n} \{Y_i - r_{\boldsymbol{\theta}}(\mathbf{X}_i)\}^2 \ , \ \ \widehat{\boldsymbol{\theta}}_n \to \boldsymbol{\theta}^* \ , \ \ r_{\widehat{\boldsymbol{\theta}}_n}(\mathbf{X}) \to r_{\boldsymbol{\theta}^*}(\mathbf{X})$$

SI $r(\mathbf{X}) \in \mathcal{F}$, entonces $r(\mathbf{X}) = r_{\boldsymbol{\theta}^*}(\mathbf{X})$ y

$$r_{\widehat{\boldsymbol{\theta}}_n}(\mathbf{X}) \to r(\mathbf{X})$$

Predecimos con $r_{\widehat{\boldsymbol{\theta}}_n}(\mathbf{X})$

Unas palabras más sobre modelo lineal

- es un mundo (materia cuatrimestral)
- No solo importa predecir bien

Modelo Lineal - Formato tradicional

- Y_i: respuestas
- $\mathbf{X}_i$: covariables o variables explicativas
  $\mathbf{X}_i = (X_{1i}, \ldots, X_{pi})'$

$$Y_i \quad = \quad \beta_1^* X_{1i} + \cdots + \beta_p^* X_{pi} + \epsilon_i \ ,$$

Modelo Lineal - Formato tradicional

- Y_i: respuestas
- $\mathbf{X}_i$: covariables o variables explicativas
  $\mathbf{X}_i = (X_{1i}, \ldots, X_{pi})'$

$$
\begin{aligned}
Y_i &= \beta_1^* X_{1i} + \cdots + \beta_p^* X_{pi} + \epsilon_i \ , \\
Y_i &= \beta_1^* \, log(Z_i) + \cdots + \beta_p^* W_i^2 + \epsilon_i \ ,
\end{aligned}
$$

Modelo Lineal - Formato tradicional

- Y_i: respuestas
- $\mathbf{X}_i$: covariables o variables explicativas
  $\mathbf{X}_i = (X_{1i}, \ldots, X_{pi})'$

$$
\begin{aligned}
Y_i &= \beta_1^* X_{1i} + \cdots + \beta_p^* X_{pi} + \epsilon_i \ , \\
Y_i &= \beta_1^* \, log(Z_i) + \cdots + \beta_p^* W_i^2 + \epsilon_i \ , \\
Y_i &= \mathbf{X}_i' \boldsymbol{\beta}^* + \epsilon_i \Rightarrow \text{ Lineal en} \boldsymbol{\beta}^*
\end{aligned}
$$

Modelo Lineal - Formato tradicional

- Y_i: respuestas
- $\mathbf{X}_i$: covariables o variables explicativas
  $\mathbf{X}_i = (X_{1i}, \ldots, X_{pi})'$

$$\begin{aligned} Y_i &= \beta_1^* X_{1i} + \cdots + \beta_p^* X_{pi} + \epsilon_i \ , \quad \text{¿y si hay intercept?} \\ Y_i &= \beta_1^* \, log(Z_i) + \cdots + \beta_p^* W_i^2 + \epsilon_i \ , \end{aligned}$$

$$Y_i = \mathbf{X}_i' \boldsymbol{\beta}^* + \epsilon_i \Rightarrow \boldsymbol{\beta}^* \text{ Parámetro a estimar}$$

Estimador de Mínimos Cuadrados

$\widehat{\boldsymbol{\beta}}$: Estimador de **Mínimos Cuadrados**

$$\widehat{\boldsymbol{\beta}} = \underset{\boldsymbol{\beta}}{\operatorname{argmin}} \sum_{i=1}^{n} (Y_i - \mathbf{X}_i' \boldsymbol{\beta})^2$$

Estimador de Mínimos Cuadrados

$\widehat{\boldsymbol{\beta}}$: Estimador de **Mínimos Cuadrados**

$$\widehat{\boldsymbol{\beta}} = \underset{\boldsymbol{\beta}}{\operatorname{argmin}} \sum_{i=1}^{n} (Y_i - \mathbf{X}_i' \boldsymbol{\beta})^2$$

Derivando e igualando 0, tenemos que $\widehat{\boldsymbol{\beta}}$ es solución del sistema

$$\begin{aligned} \sum_{i=1}^{n} (Y_i - \mathbf{X}_i' \boldsymbol{\beta}) \mathbf{X}_i &= 0 \\ \mathbf{X}'\mathbf{X}\boldsymbol{\beta} &= \mathbf{X}'\mathbf{Y} \quad \text{Ec. Normales} \end{aligned}$$

definiendo la matriz de diseño $\mathbf{X}$ y el vector de respuestas $\mathbf{Y}$ convenientemente.

Notacion Matricial: Modelo Lineal Simple

$$Y_i = \beta_0^* + \beta_1^* X_i + \epsilon_i \quad 1 \leq i \leq n$$

$$\mathbf{Y} = \begin{pmatrix} Y_1 \\ Y_2 \\ . \\ . \\ Y_n \end{pmatrix} \quad \mathbf{X} = \begin{pmatrix} 1 & X_1 \\ 1 & X_2 \\ . & . \\ . & . \\ 1 & X_n \end{pmatrix} \quad \boldsymbol{\beta}^* = \begin{pmatrix} \beta_0^* \\ \beta_1^* \end{pmatrix} \quad \varepsilon = \begin{pmatrix} \epsilon_1 \\ \epsilon_2 \\ . \\ . \\ \epsilon_n \end{pmatrix}$$

$$\mathbf{Y} = \mathbf{X}\boldsymbol{\beta}^* + \varepsilon$$

Ecuaciones Normales

De las ecuaciones normales tenemos que

$$\mathbf{X}'\mathbf{X}\widehat{\boldsymbol{\beta}} = \mathbf{X}'\mathbf{Y}$$

Cuando $\mathbf{X}'\mathbf{X}$ es no singular, la solución es única y resulta

$$\widehat{\boldsymbol{\beta}} = \left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{X}'\mathbf{Y}$$

Estimador de Mínimos Cuadrados - en R: `lm(respues~ predictoras)`

$\widehat{\boldsymbol{\beta}}$: Estimador de **Mínimos Cuadrados**
Propiedades: Bajo condiciones muy generales $\widehat{\boldsymbol{\beta}}$ es

- insesgado
- consistente
- asintóticamente normal

Bajo normalidad de los errores

Supongamos que las covariables son determinísticas.

Si $\varepsilon_i \sim N(0, \sigma^2)$, entonces $Y_i = \mathbf{X}_i' \boldsymbol{\beta}^* + \epsilon_i \sim N(\mathbf{X}_i' \boldsymbol{\beta}^*, \sigma^2)$.

Bajo normalidad de los errores

Supongamos que las covariables son determinísticas.

Si $\varepsilon_i \sim N(0, \sigma^2)$, entonces $Y_i = \mathbf{X}_i' \boldsymbol{\beta}^* + \epsilon_i \sim N(\mathbf{X}_i' \boldsymbol{\beta}^*, \sigma^2)$.

¿Como obtendríamos un estimador de $\boldsymbol{\beta}^*$ por máxima verosimilitud?

Bajo normalidad de los errores

Como obtendríamos un estimador de $\boldsymbol{\beta}^*$ por máxima verosimilitud?

$$L(\boldsymbol{\beta}) = L(\boldsymbol{\beta}, (y_1, X_1), ..., (y_n, X_n)) = \frac{1}{(\sigma\sqrt{2\pi})^n} e^{\left(\frac{-1}{2\sigma^2}\sum_{i=1}^{n}(y_i - \mathbf{X}_i'\boldsymbol{\beta})^2\right)}$$

Bajo normalidad de los errores

Como obtendríamos un estimador de $\boldsymbol{\beta}^*$ por máxima verosimilitud?

$$L(\boldsymbol{\beta}) = L(\boldsymbol{\beta}, (y_1, X_1), ..., (y_n, X_n)) = \frac{1}{(\sigma\sqrt{2\pi})^n} e^{\left(\frac{-1}{2\sigma^2}\sum_{i=1}^{n}(y_i - \mathbf{X}_i'\boldsymbol{\beta})^2\right)}$$

buscamos el máximo de $L(\boldsymbol{\beta})$

Bajo normalidad de los errores

Como obtendríamos un estimador de $\boldsymbol{\beta}^*$ por máxima verosimilitud?

$$L(\boldsymbol{\beta}) = L(\boldsymbol{\beta}, (y_1, X_1), ..., (y_n, X_n)) = \frac{1}{(\sigma\sqrt{2\pi})^n} e^{\left(\frac{-1}{2\sigma^2}\sum_{i=1}^{n}(y_i - \mathbf{X}_i'\boldsymbol{\beta})^2\right)}$$

buscamos el máximo de $L(\boldsymbol{\beta})$ o maximizamos la log verosimilitud

$$\log L(\boldsymbol{\beta}) \propto -\sum_{i=1}^{n}(y_i - \mathbf{X}_i'\boldsymbol{\beta})^2$$

Bajo normalidad de los errores

Como obtendríamos un estimador de $\boldsymbol{\beta}^*$ por máxima verosimilitud?

$$L(\boldsymbol{\beta}) = L(\boldsymbol{\beta}, (y_1, X_1), ..., (y_n, X_n)) = \frac{1}{(\sigma\sqrt{2\pi})^n} e^{\left(\frac{-1}{2\sigma^2}\sum_{i=1}^n (y_i - \mathbf{X}_i'\boldsymbol{\beta})^2\right)}$$

buscamos el máximo de $L(\boldsymbol{\beta})$ o maximizamos la log verosimilitud

$$\log L(\boldsymbol{\beta}) \propto -\sum_{i=1}^n (y_i - \mathbf{X}_i'\boldsymbol{\beta})^2$$

o equivalentemente el mínimo de

$$\sum_{i=1}^n (y_i - \mathbf{X}_i'\boldsymbol{\beta})^2$$

Bajo normalidad de los errores

Como obtendríamos un estimador de $\boldsymbol{\beta}^*$ por máxima verosimilitud?

$$L(\boldsymbol{\beta}) = L(\boldsymbol{\beta}, (y_1, X_1), ..., (y_n, X_n)) = \frac{1}{(\sigma\sqrt{2\pi})^n} e^{\left(\frac{-1}{2\sigma^2}\sum_{i=1}^{n}(y_i - \mathbf{X}_i'\boldsymbol{\beta})^2\right)}$$

buscamos el máximo de $L(\boldsymbol{\beta})$ o maximizamos la log verosimilitud

$$\log L(\boldsymbol{\beta}) \propto -\sum_{i=1}^{n}(y_i - \mathbf{X}_i'\boldsymbol{\beta})^2$$

o equivalentemente el mínimo de

$$\sum_{i=1}^{n}(y_i - \mathbf{X}_i'\boldsymbol{\beta})^2$$

Si las respuestas ε_i tienen distribución normal el estimador de mínimos cuadrados,$\widehat{\boldsymbol{\beta}}$, coincide con el estimador de máxima verosimilitud y $\widehat{\boldsymbol{\beta}}$ hereda la distribución normal.

Comandos de R

Sigamos con el ejemplo de los datos de LIDAR y realicemos un ajuste polinómico de orden 4, es decir ajustamos el modelo:

$$logratio_i = \beta_1\, rango_i + \beta_2\, rango_i^2 + \beta_3\, rango_i^3 + \beta_4\, rango_i^4 + \beta_5 + \epsilon_i$$

```
rango2<-rango^2
rango3<-rango^3
rango4<-rango^4

lm(logratio ~ rango+rango2+rango3+rango4)

Call:
lm(formula = logratio ~ rango + rango2 + rango3 + rango4)

Coefficients:
(Intercept)        rango       rango2       rango3       rango4
  5.019e+01   -4.031e-01    1.193e-03   -1.538e-06    7.260e-10
```

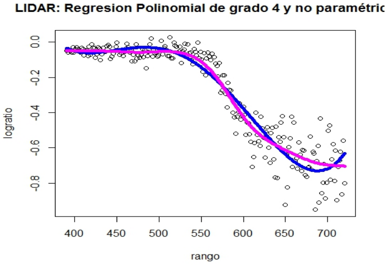
Comandos de R

Graficamos el ajuste obtenido con la ventana óptima con ksmooth y el ajuste hecho con el polinomio de grado 4 usando el estimador de mínimos cuadrados para el modelo:

$$logratio_i = \beta_1\, rango_i + \beta_2\, rango_i^2 + \beta_3\, rango_i^3 + \beta_4\, rango_i^4 + \beta_5 + \epsilon_i$$

```
plot(rango, logratio)
title("LIDAR: Regresion Polinomial de grado 4 y no paramétrica")

lines(range, predict(lm(logratio ~ range+range2+range3+range4)), col="blue", lwd=5)
lines(ksmooth(rango, logratio, "normal", bandwidth=h_cv), lwd=5, col="magenta")
```



Regresión Lineal - Virtudes.

- Popular.
- Intepretabilidad- Inferencia bajo errores normales
- Mínimos cuadrados es el estimador de Máxima verosimilitud si $\varepsilon \sim \mathcal{N}(0, \sigma^2)$.
- Intepretabilidad- Inferencia asintótica.
- Parsimonioso - Poca varianza.

Regresión Lineal - $p >> n$?

El estimador de mínimos cuadrados no está univocamente definido.
No sabemos como predecir

- Selección de Variables - Stepwise - AIC - BIC.
- Penalización.
- Reducción de dimensión.

$$\sum \left(y_i - \left(\hat{\beta}_0 + \hat{\beta}_1 x_i + \hat{\beta}_2 x_i^2 \right) \right)^2$$

$$\sum \left(y_i - \left(\hat{\alpha}_0 + \hat{\alpha}_1 x_i + \hat{\alpha}_2 x_i^2 \right) \right)^2$$

$$\text{(Training) Error} \left(\mathcal{M}_{gde}, \mathcal{D}_n \right)$$

$$\leq \text{(Training) Error} \left(\mathcal{M}_{chico}, \mathcal{D}_n \right)$$

$$\text{(Training) Error} \left(\mathcal{M}, \mathcal{D}_m \right) + \lambda_m \underbrace{\text{Tamaño}(\mathcal{M})}_{\text{parametros}}$$

Likelihood$(\mathcal{M}_{chico}, \mathcal{D}_n)$ Likelihood$(\mathcal{M}_{gde}, \mathcal{D}_n)$

$$\text{Likelihood}(\mathcal{M}, \mathcal{D}_n) - \lambda_n \, \text{Tamaño}(\mathcal{M})$$

#parametros

Likelihood cross validation.

Leave one out

$$\text{Bondad de } \mathcal{M} = \sum_i \lg f_{\mathcal{M}, \mathcal{D}_n^{(-i)}}(X_i)$$

k fold

$$\text{Bondad de } \mathcal{M} = \frac{1}{K} \sum_{k=1}^{K} B(k)$$

$$B(k) = \frac{1}{|\tau_k^c|} \sum_{j \in \tau_k^c} \lg f_{\mathcal{M}\, \tau_k}(X_j)$$

Regresión Lineal - Penalización - glmnet

$$(\widehat{\beta}^R_{0,\lambda}, \widehat{\boldsymbol{\beta}}^R_\lambda) = \arg\min_{\beta_0, \boldsymbol{\beta}} \sum_{i=1}^{n} \{Y_i - (\beta_0 + \boldsymbol{\beta}^t \mathbf{X}_i)\}^2 + \lambda \sum_{j=1}^{p} \beta_j^2$$

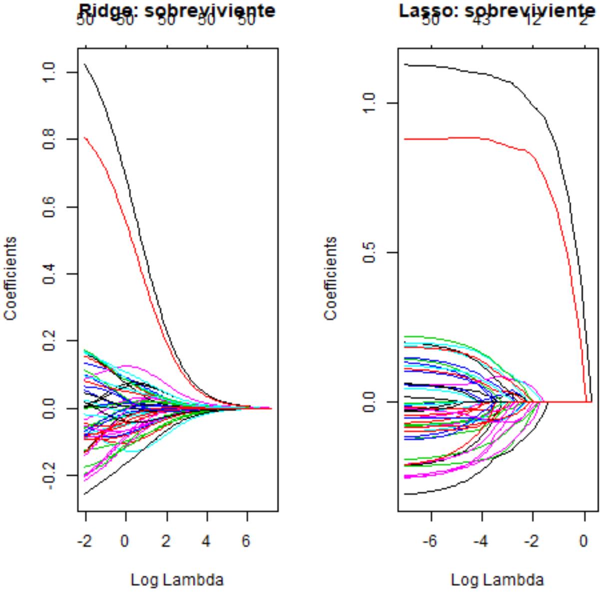
$$(\widehat{\beta}^L_{0,\lambda}, \widehat{\boldsymbol{\beta}}^L_\lambda) = \arg\min_{\beta_0, \boldsymbol{\beta}} \sum_{i=1}^{n} \{Y_i - (\beta_0 + \boldsymbol{\beta}^t \mathbf{X}_i)\}^2 + \lambda \sum_{j=1}^{p} |\beta_j|$$

Con predictoras estandarizadas!

Note that by default, the glmnet() function standardizes the variables so that they are on the same scale.

Hay lindas interpretaciones Bayesianas

Regresión Lineal - Penalización - glmnet



Modelos Aditivos

Medelo lineal $\quad r(\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_p) = \beta_0 + \beta_1 \mathbf{x}_1 + \ldots + \beta_p \mathbf{x}_p$

Estimamos β_0, β_1, ...β_p

Medelo aditivo $\quad r(\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_p) = \beta_0 + f_1(\mathbf{x}_1) + f_2(\mathbf{x}_2) + \ldots + f_p(\mathbf{x}_p)$

Estimamos β_0, $f_i : \mathbb{R} \to \mathbb{R}$, $i = 1, \ldots, p$

$p >> n$?

- Reducción de dimensión.
  - $\mathbf{x} \in \mathbb{R}^p \to \mathsf{Red}(\mathbf{x}) \in \mathbb{R}^d$, $d < p$.
  - $\mathbb{E}(Y \mid \mathsf{Red}(\mathbf{X}))$.
  - Componentes principales.
  - Reducción Suficiente : $Y \mid \mathbf{X} \sim Y \mid \mathsf{Red}(\mathbf{X})$. Forzani et al.

Al infinito y más alla!

Tunning Parameter - Selección de Método / Modelo

Tunning parameter

Siempre nos faltan dos mangos para el peso

- Nadaraya-Watson. ventana: h. $\widehat{r}_h(\mathbf{x})$
- Vecinos próximos (knn). vecinos: k. $\widehat{r}_k(\mathbf{x})$
- Regresión Polinomial: grado del polinomio: K. $\widehat{r}_K(\mathbf{x})$
- Ridge - Lasso. penalidad: λ: $\widehat{r}_\lambda(\mathbf{x}) = \beta_0 + \boldsymbol{\beta}_\lambda^t \mathbf{x}$
- Caso general: $\widehat{r}_t(\mathbf{x})$, t, *tunning parameter*

Predictor polinomial con UNA variable

Mejor predictor polinomial de grado k:

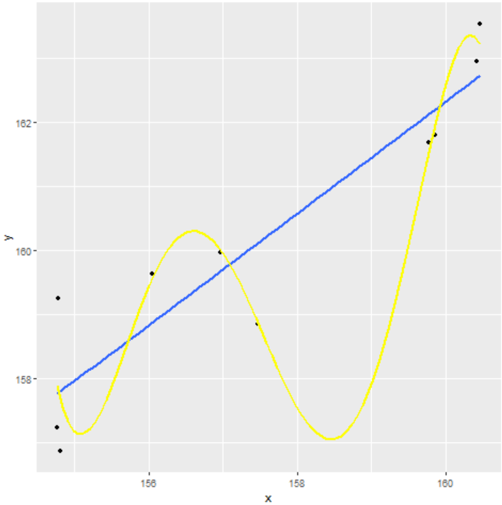
$$\widehat{r}_K(\mathbf{x}) = \widehat{\alpha}_0 + \widehat{\alpha}_1\mathbf{x} + \widehat{\alpha}_2\mathbf{x}^2 + \ldots + \widehat{\alpha}_k\mathbf{x}^k$$

$$(\widehat{\alpha}_0, \widehat{\boldsymbol{\alpha}}) = \arg\min \sum_{i=1}^{n} \{Y_i - (\alpha_0 + \alpha_1\mathbf{X}_i + \alpha_2\mathbf{X}_i^2 + \ldots + \alpha_k\mathbf{X}_i^k)\}^2$$

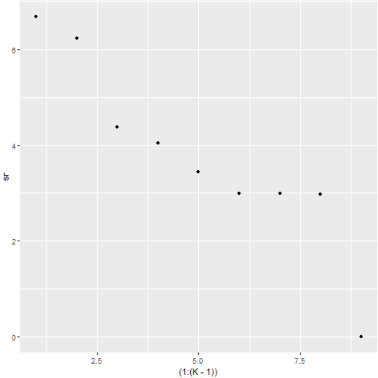
Error de **ENTRENAMIENTO** de $\widehat{r}_K$:

$$R(K) = \sum_{i=1}^{n} \{Y_i - \widehat{r}_K(\mathbf{X}_i)\}^2$$

Ajuste polinomial con diferente grado



Ajuste polinomial con diferente grado



Error de **ENTRENAMIENTO** de $\widehat{r}_K$:

$$R(K) = \sum_{i=1}^{n} \{Y_i - \widehat{r}_K(\mathbf{X}_i)\}^2$$

Test Error(es) (SL sin R)

...test error is the average error that results from using a statistical learning method to predict the response on a new observation - that is, a measurement that was not used in training the method. (SL)

- Test Error

$$\text{Error}_{\mathcal{D}_n} = \mathbb{E}\left[\{Y_{\text{new}} - \widehat{r}_{\mathcal{D}_n}(\mathbf{X}_{\text{new}})\}^2 \mid \mathcal{D}_n\right]$$

Test Error(es) (SL sin R)

...test error is the average error that results from using a statistical learning method to predict the response on a new observation - that is, a measurement that was not used in training the method. (SL)

- Test Error

$$\text{Error}_{\mathcal{D}_n} = \mathbb{E}\left[\{Y_{\text{new}} - \widehat{r}_{\mathcal{D}_n}(\mathbf{X}_{\text{new}})\}^2 \mid \mathcal{D}_n\right]$$

- Expected Test Error

$$\mathbb{E}\left[\{Y - \widehat{r}_{\mathcal{D}_n}(\mathbf{X})\}^2\right] \equiv \mathbb{E}_{(\mathbf{X},Y),\mathcal{D}_n}\left[\{Y - \widehat{r}_{\mathcal{D}_n}(\mathbf{X})\}^2\right]$$

- Conditioned Test Error

$$\text{Error}_{\mathbf{x}} = \mathbb{E}\left[\{Y - \widehat{r}_{\mathcal{D}_n}(\mathbf{X})\}^2 \mid \mathbf{X} = \mathbf{x}\right] \equiv \mathbb{E}_{Y,\mathcal{D}_n}\left[\{Y - \widehat{r}_{\mathcal{D}_n}(\mathbf{x})\}^2\right]$$

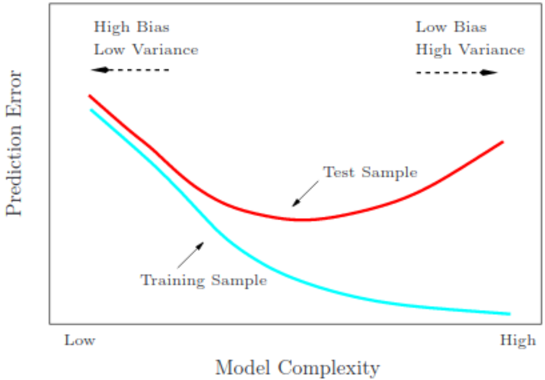
Conditioned error test

- $\mathbb{V}(Y \mid \mathbf{X}=\mathbf{x})$: Error irreducible $\mathbb{V}(\varepsilon)$
- $\text{Bias}(\widehat{r}(\mathbf{x})) = \mathbb{E}\{\widehat{r}(\mathbf{x})\} - r(\mathbf{x})$.
- $\text{Var}(\widehat{r}(\mathbf{x})) = \mathbb{E}\left\{[r(\mathbf{x}) - \mathbb{E}\{\widehat{r}(\mathbf{x})\}]^2\right\}$.

$$\mathbb{E}\left[\{Y - \widehat{r}(\mathbf{X})\}^2 \mid \mathbf{X}=\mathbf{x}\right] = \sigma^2 + \text{Bias}(\widehat{r}(\mathbf{x}))^2 + \text{Var}(\widehat{r}(\mathbf{x})).$$

Typical trend: underfitting means high bias and low variance, overfitting means low bias but high variance.
As a general rule, as we use more flexible methods, the variance will increase and the bias will decrease.

ElemStatLearn



IGURE 2.11. *Test and training error as a function of model complexity.*

- Test error: $\mathbb{E}\left(\{Y_{\text{new}} - \widehat{r}_{t,\mathcal{D}_n}(\mathbf{X}_{\text{new}})\}^2 \mid \mathcal{D}_n\right)$
- Training error : $\frac{1}{n}\sum_{i=1}^{n}\{Y_i - \widehat{r}_{t,\mathcal{D}_n}(\mathbf{X}_i)\}^2$

Todo muy lindo, pero ...

¿Qué hago con mis datos?

Splitting the data

do as well as possible within a given class of rules... This is achieved by splitting the data into a training sequence and a testing sequence. (DGL)

Data-rich situation

- The training set is used to fit the models
- The validation set is used for model selection - con esto elijo tunning parameters.
- The test set is used for assessment of the generalization error of the final chosen model - Aca compiten el mejor candidato de cada posible método.

Menos Datos

- Training - Cross Validation.
- The test set is used for assessment of the generalization error of the final chosen model - Aca compiten el mejor candidato de cada posible método.

Yapa

Reducción de Dimension

- $\mathbf{X} \in \mathbb{R}^p$.
- $\alpha_m \in \mathbb{R}^p$, $||\alpha_m|| = 1$, $m = 1, \ldots, M$
- $Z_m = \alpha_m^t \mathbf{X} \in \mathbb{R}$.
- $\mathbf{Z} = (\mathbf{Z}_1, \ldots, \mathbf{Z}_M) \in \mathbb{R}^M$.
- Modelo lineal para la la regresión de Y en $\mathbf{Z}$:
- ¿Cómo elegir α_m?
    - Componentes principales.
    - Partial Least Squares.

Reducción de dimensión: Projection Pursuit

- $\mathbf{X} \in \mathbb{R}^p$.
- $\alpha_m \in \mathbb{R}^p$, $||\alpha_m|| = 1$, $m = 1, \ldots, M$
- $Z_m = \alpha_m^t \mathbf{X} \in \mathbb{R}$.
- $\mathbf{Z} = (\mathbf{Z}_1, \ldots, \mathbf{Z}_M) \in \mathbb{R}^M$.
- Modelo aditivo para la regresión de Y en $\mathbf{Z}$:

$$r(\mathbf{X}) = \beta_0 + f_1(\alpha_1^t \mathbf{X}) + f_1(\alpha_2^t \mathbf{X}) + \ldots + f_m(\alpha_m^t \mathbf{X})$$

Reducción de dimensión: Projection Pursuit

- $\mathbf{X} \in \mathbb{R}^p$.
- $\alpha_m \in \mathbb{R}^p$, $||\alpha_m|| = 1$, $m = 1, \ldots, M$
- $Z_m = \alpha_m^t \mathbf{X} \in \mathbb{R}$.
- $\mathbf{Z} = (\mathbf{Z}_1, \ldots, \mathbf{Z}_M) \in \mathbb{R}^M$.
- Modelo aditivo para la regresión de Y en $\mathbf{Z}$:

$$r(\mathbf{X}) = \beta_0 + f_1(\alpha_1^t \mathbf{X}) + f_1(\alpha_2^t \mathbf{X}) + \ldots + f_m(\alpha_m^t \mathbf{X})$$

...the product $X_1 X_2$ can be written as $\left\{(X_1 + X_2)^2 - (X_1 - X_2)^2\right\}/4$, and higher-order products can be represented similarly(SL sin R).

Neural Networks - Redes Neuronales

There has been a great deal of hype surrounding neural networks, making them seem magical and mysterious. As we make clear in this section, they are just nonlinear statistical models, much like the projection pursuit regression model discussed above. (SL sin R)

Redes Neuronales - 1 capa

http://neuralnetworksanddeeplearning.com/chap1.html

$$\begin{aligned} Z_m &= \sigma(\alpha_{0,m} + \boldsymbol{\alpha}_m^t \mathbf{X}) \,, \quad m = 1, \ldots, M \\ T &= \beta_0 + \boldsymbol{\beta}^t \mathbf{Z} \,, \\ T &= T(\boldsymbol{\theta}, \mathbf{X}) \,, \quad \boldsymbol{\theta} = (\alpha_0, \boldsymbol{\alpha}, \beta_0, \boldsymbol{\beta}) \end{aligned}$$

$$\widehat{\boldsymbol{\theta}} = \arg\min_{\boldsymbol{\theta}} \sum_{i=1}^{n} \{Y_i - T(\boldsymbol{\theta}, \mathbf{X}_i)\}^2$$

Predicción: $T(\widehat{\boldsymbol{\theta}}, \mathbf{X})$

Muchas capas

- $\mathbf{X} = (X_1, \ldots, X_p)$ capa de entrada: layer $\ell = 1$

$$a_1^\ell = X_1, \ldots, a_p^\ell = X_p$$

- Construcción de capa ℓ con N_ℓ nodos, para $\ell \geq 2$:

$$z_j^\ell = \sum_{k=1}^{N_{\ell-1}} \omega_{k,j}^\ell a_k^{\ell-1} + b_j^\ell$$

$$a_j^\ell = \sigma\left(\sum_{k=1}^{N_{\ell-1}} \omega_{k,j}^\ell a_k^{\ell-1} + b_j^\ell\right) \ , \ \ j = 1 \ldots, N_\ell$$

- Notación matricial:

$$z^\ell = \omega^\ell a^{\ell-1} + b^\ell \ , \ \ a^\ell = \sigma\left(\omega^\ell a^{\ell-1} + b_\ell\right)$$

Muchas capas

- Notación matricial: $a^1 = \mathbf{X}$

$$z^\ell = \omega^\ell a^{\ell-1} + b_\ell \ , \ \ a^\ell = \sigma\left(\omega^\ell a^{\ell-1} + b^\ell\right) \ , \ \ \ell = 1, \ldots, K.$$

- ω: weights, b: bias
- Parámetros: $\boldsymbol{\theta} = \{(\omega^\ell, b^\ell) : \ell = 1, \ldots, K\}$
- $z^K = z(\boldsymbol{\theta}, \mathbf{X})$

$$\widehat{\boldsymbol{\theta}} = \arg\min_{\boldsymbol{\theta}} \sum_{i=1}^{n} \{Y_i - z(\boldsymbol{\theta}, \mathbf{X}_i)\}^2$$

Predicción: $z(\widehat{\boldsymbol{\theta}}, \mathbf{X})$

Redes Nauronales

$$R(\boldsymbol{\theta}) = \sum_{i=1}^{n} \{Y_i - z(\boldsymbol{\theta}, \mathbf{X}_i)\}^2$$

- is nonconvex, possessing many local minima.
- gradient descent, called back-propagation in this setting.
- Typically we don't want the global minimizer of $R(\boldsymbol{\theta})$, as this is likely to be an overfit solution. Instead some regularization is needed: this is achieved directly through a penalty term, or indirectly by early stopping.
- There is quite an art in training neural networks. The model is generally overparametrized, and the optimization problem is nonconvex and unstable unless certain guidelines are followed. In this section we summarize some of the important issues.
- there can be quite an art to the design of the hidden layers.

Modelos de Regresión

$$y = \mathbf{x}^t\beta + \varepsilon$$

$$y = r(\mathbf{x}) + \varepsilon$$

$$y = \mathbf{x}^{\mathbf{t}}\beta + \eta(t) + \varepsilon$$

$$y = r(\mathbf{x}) + \sigma(\mathbf{x})\varepsilon$$

Modelos de Regresión

$$y = \mathbf{x}^t\beta + \varepsilon$$

$$y = r(\mathbf{x}) + \varepsilon$$

$$y = \mathbf{x}^{\mathbf{t}}\beta + \eta(t) + \varepsilon$$

$$y = r(\mathbf{x}) + \sigma(\mathbf{x})\varepsilon$$

$$\varepsilon \sim \mathcal{N}(0, \sigma^2)$$

Modelos de Regresión

$$\varepsilon \sim \mathcal{N}(0, \sigma^2)$$

$$y \mid \mathbf{x} \sim \mathcal{N}(\mu(\mathbf{x}), \sigma^2(\mathbf{x}))$$

$$y = \mathbf{x}^t \beta + \varepsilon \;, \quad \mu(\mathbf{x}) = \mathbf{x}^t \beta \;, \sigma^2(\mathbf{x}) = \sigma^2$$

$$y = r(\mathbf{x}) + \varepsilon \;, \quad \mu(\mathbf{x}) = r(\mathbf{x}) \;, \sigma^2(\mathbf{x}) = \sigma^2$$

$$y = \mathbf{x}^{\mathbf{t}} \beta + \eta(t) + \varepsilon \;, \quad \mu(\mathbf{x}, t) = \mathbf{x}^{\mathbf{t}} \beta + \eta(t) \;, \sigma^2(\mathbf{x}, t) = \sigma^2$$

$$y = r(\mathbf{x}) + \sigma(\mathbf{x})\varepsilon \;, \quad \mu(\mathbf{x}) = r(\mathbf{x}) \;, \sigma^2(\mathbf{x}) = \sigma^2(\mathbf{x})$$

Otras Distribuciones

- Gaussiana $\mathcal{N}(\mu, \sigma^2)$
- Bernoulli $\mathcal{B}(1, p)$
- Poisson $\mathcal{P}(\lambda)$
- Exponencial (λ)
- Gamma $\Gamma(\alpha, \lambda)$
- Gumbel - Weibull (extreme value distributions)

$$Y \sim F_\theta \ , \quad \theta \in \Theta \subseteq \mathbb{R}^k$$

Familias exponenciales (parámetro natural).

$$f(y) = h(y) \exp^{\theta^t T(y) - A(\theta)}$$

- Normal ($\theta = (\mu/\sigma^2, -1/2\sigma^2)$)
- Bernoulli ($\theta = \log(p/(1-p))$)
- Poisson ($\theta = \log(\lambda))$)
- Normal ($\theta = (\mu/\sigma^2, -1/2\sigma^2)$)

Modelos de Regresión Lineal Generalizados

$$y \mid \mathbf{x} \sim F_{\theta(\mathbf{x})}$$

$$\theta(\mathbf{x}) = \mathbf{x}^t \beta$$

$$\theta(\mathbf{x}, t) = \mathbf{x}^t \beta + \eta(t)$$