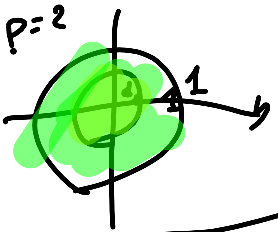
$$\mathbb{P}r\left[\text{all } N \text{ points have distance} \geq d\right] = \frac{1}{2}$$

$t_i :=$ distance between x_i and origin

$$\mathbb{P}r\left[t_i \geq d\right] = 1 - \mathbb{P}\left[t_i < d\right]$$

p=2



$$= 1 - \frac{\text{volume of ball radius } d}{\text{volume of ball radius } 1}$$

volume of a ball with radius r $= \dfrac{\pi^{p/2}}{(p/2)!} r^p$

$p=2$ $\quad \pi r^2$

$$= 1 - \frac{\frac{\pi^{p/2}}{(p/2)!} d^p}{\frac{\pi^{p/2}}{(p/2)!} 1^p} = 1 - d^p$$

$$\mathbb{P}\left[\underbrace{\forall i, t_i \geq d}_{= 1/2}\right] = \prod_{i=1}^{N} \mathbb{P}\left[t_i \geq d\right]$$

$$= (1 - d^p)^N$$

$$(1 - d^p)^N = 1/2$$

$$1 - d^p = \left(\frac{1}{2}\right)^{1/N}$$

$$d^p = 1 - \left(\frac{1}{2}\right)^{1/N}$$

$$d = \left(1 - \left(\frac{1}{2}\right)^{1/N}\right)^{1/p}$$

qed

find $\hat{f}(x)$ as a useful approximation of $f(x)$

last week: $L(y, f(x)) = (y - f(x))^2$ squared loss function

leads to $f(x) = E[Y \mid X = x]$

K-nearest neighbour

$$\underset{\text{average}}{E}[Y \mid \underbrace{X = x}_{\text{neighbour}}]$$

issues
- curse of dimensionality
  - e.g. nearest point gets far from the point of interest p increasing
- balance between bias - variance

Statistics vs Machine Learning

Statistical approach:

Starting from the model

$$Y = f(x) + \varepsilon \quad , E[\varepsilon] = 0 \quad , \varepsilon \perp\!\!\!\perp X$$

additive model $\longrightarrow$ approximation of the truth

statistics $\hat{f}(x)$ approx $f(x)$

- we do not suppose $Y = f(x)$ (deterministic)

BUT

we add an error term which captures:
- measurements errors;
- effects of non-measured variables;
- ...

often $\varepsilon \sim$ iid , $\varepsilon \overset{iid}{\sim} N(0, \sigma^2)$

most natural approach, least square

Machine learning approach

Assume $Y = f(x) + \varepsilon$

→ start from $\hat{f}(x)$, possibly simple $\hat{f}(x) = c$ (inizialization)

→ evaluate $\hat{f}(x)$ on our training set

e.g. $\sum_{i=1}^{N} (y_i - \hat{f}^{[k]}(x_i))^2$

→ modify $\hat{f}(x)$ to improve the prediction

$$\sum_{i=1}^{N} (y_i - \hat{f}^{[k+1]}(x_i))^2 < \sum_{i=1}^{N} (y_i - \hat{f}^{[k]}(x_i))^2$$

k is the step of the algorithm

→ use training set (x_i, y_i)

→ learning by examples

→ focus is on the learner, that does not aim to approximate a true $f(x)$

statistical/mathematical approach

(x_i, y_i) points of a $p+1$ dimensional space

$$f(x) : \underset{\mathbb{R}^p}{\overset{\cap}{X}} \to \underset{\mathbb{R}}{\overset{\cap}{y}}$$

goal: giving an approximation of $f(x)$ working $\forall$ points in X given T

Parametric approach

$$\mathcal{F} = \{ p(y, x; \theta), \ \theta \in \Theta \subseteq \mathbb{R}^p \}$$

$$Y = f(x; \theta) + \varepsilon \qquad f_\theta(x) = f(x;\theta)$$

e.g.:

- linear model : $f_\theta(x) = X^T \beta \qquad \theta = \beta$
- linear basis expansion : $f_\theta(x) = \sum_{k=1}^{K} h_k(x)\, \theta_k$

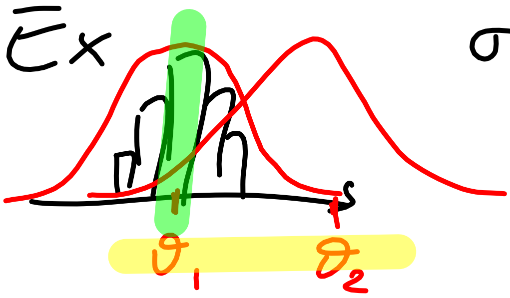
$$h_1(x) = x_1$$
$$h_2(x) = x_1^2$$

$$RSS(\theta) = \sum_{i=1}^{N} (y_i - f_\theta(x_i))^2$$

a function of θ

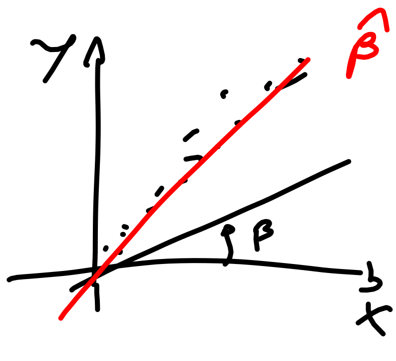
$$\hat{\theta} = \underset{\theta}{\arg\min}\, RSS(\theta)$$

Ex



$\sigma^2 = 1$

$\hat{\theta} = \theta_1$



simplest cases
- least squares

more complicated frameworks
- maximum likelihood estimation

Likelihood estimation

$Y_i \sim P$ $\quad p(y)$ is indexed by θ

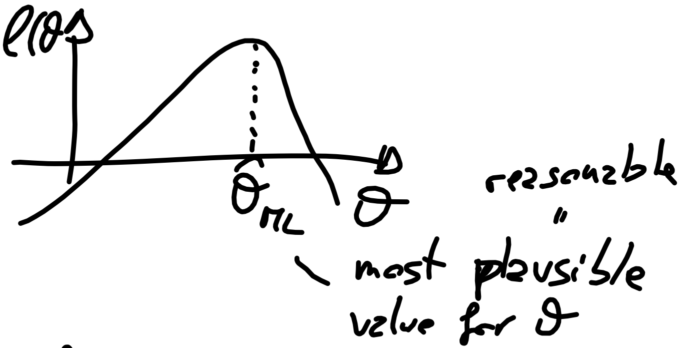
Y_i i.i.d. $\quad p(y;\theta)$

e.g. Gaussian distribution $\quad p(y;\underbrace{\mu,\sigma^2}_{\theta})$

$$L(\theta) = \prod_{i=1}^{N} p(y_i;\theta)$$

$$\ell(\theta) = \sum_{i=1}^{N} \log p(y_i;\theta)$$

$$\hat{\theta}_{ML} = \arg\max_{\theta} \ell(\theta)$$



Note: when $\varepsilon \sim N(0,\sigma^2)$

$$\hat{\theta}_{ML} \equiv \hat{\theta}_{LS}$$

Restricted estimators

instead of estimating θ as

$$\hat{\theta} = \arg\min_{\theta} RSS(\theta)$$

we look for

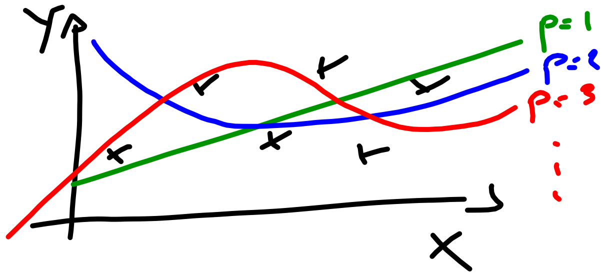
$$\hat{\theta} = \arg\min_{\theta} PRSS(\theta)$$

where: $PRSS(\theta) = RSS(\theta) + \lambda J(\theta)$

LASSO $\quad J(\theta) = \sum_{j=1}^{p} |\theta_j|$

RIDGE $\quad J(\theta) = \sum_{j=1}^{p} \theta_j^2$

Model selection and variance - bias trade - off



$$f(x,\theta) = \sum_{j=1}^{p} \theta_j x^j$$

given enough parameters θ_j, we will always be able to find a line which passes through all points (no bias)

penalization for the number of parameters

$$\hat{\theta} = \arg\min_{\theta} \{ RSS(\theta) + \lambda J(\theta) \}$$

Bias - variance trade off

$y \quad f(x_0)$

$Y = f(x) + \varepsilon$

$$\text{error} = \sigma^2 + \underbrace{\text{variance} + \text{bias}^2}_{MSE}$$

see figure 2.11

Linear methods for regression

$E[Y|X]$ is linear in inputs

$$E[Y|X] = X^T\beta$$

- → simple
- → often adequate
- → easy to interpret
- → often outperforms fancier methods

↓ simplicity!

- → sample size is small
- → sparse data
- → low signal-to-noise ratio

linear methods

→ regression (ch 3)

→ classification (ch 4)

Regression

- consider continuous Y
- linear regression $f(x) = \beta_0 + \beta_1 x_1 + \ldots + \beta_p x_p$

$$\hat{\beta} = \underset{\beta}{\operatorname{argmin}}\ RSS(\beta)$$

$$\sum_{i=1}^{N} (y_i - \beta_0 - \sum_{j=1}^{p} x_j \beta_j$$

matrix form $(y - X\beta)^T(y - X\beta)$

$$\frac{\partial RSS(\beta)}{\partial \beta} = -2X^T(y - X\beta)$$

$$\frac{\partial^2 RSS(\beta)}{\partial \beta^2} = 2X^TX$$

$$\frac{\partial RSS(\beta)}{\partial \beta} = 0 \qquad X^T(y - X\beta) = 0$$

$$X^Ty - X^TX\beta = 0$$

$$\hat{\beta} = (X^TX)^{-1}X^Ty$$

$$\frac{\partial^2 RSS(\beta)}{\partial \beta^2} = 2X^TX > 0 \implies \hat{\beta} \text{ minimum}$$

$$\hat{y} = X\hat{\beta} = \underbrace{X(X^TX)^{-1}X^T}_{H}y$$

hat matrix

projection matrix

Properties

$$Var(\hat{\beta}) = (X^TX)^{-1}\sigma^2$$

$$\hat{\sigma}^2 = \frac{1}{N-p-1}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2$$

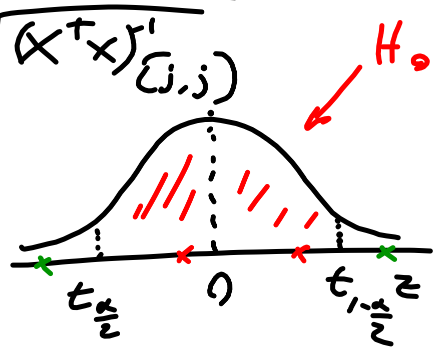
Focus on $\varepsilon \sim N(0, \sigma^2)$ ε_i iid

$$\hat{\beta} \sim N(\beta, (X^TX)^{-1}\sigma^2)$$

$$(N-p-1)\hat{\sigma}^2 \sim \sigma^2 \chi^2_{N-p-1}$$

$$H_0 : \beta_j = 0 \longrightarrow z_j = \frac{\hat{\beta}_j - 0}{sd(\hat{\beta}_j)} = \frac{\hat{\beta}_j}{\hat{\sigma}\sqrt{(X^T X)^{-1}_{(j,j)}}}$$

under H_0 $\quad z_j \overset{H_0}{\sim} t_{N-p-1}$

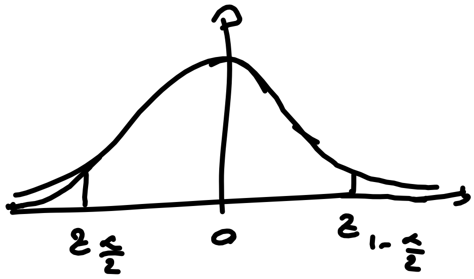


z_{obs}

H_1 ~~was~~ reject H_0

when σ^2 is known

under H_0 $\quad z_j \overset{H_0}{\sim} N(0;1)$



$$Y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 + \varepsilon$$

is x_1 useful to predict y?

$\beta_j = 0 \longrightarrow \varepsilon$ < reject H_0 Yes / do NOT reject H_0 not Yes

Are (x_2, x_3) useful to predict y?

$H_0 : \beta_2, \beta_3 = 0$

$$F = \frac{(RSS_0 - RSS_1)/(p_1 - p_0)}{RSS_1/(N - p_1 - 1)}$$