

# Allmän linjär regression



Finns ett hyperplan i  $p$  dimensioner

$$Y = \beta_0 + \beta_1 X_1 + \dots + \beta_p X_p$$

som optimerar en kostnadsfunktion  $C(\beta)$

Vi kan betrakta distributionen för  $X$ :

(om  $X \sim N$  (gaussian) "vanlig" linjär regression

om  $X \sim \text{Bin}$  (binomial) logistisk regression

om  $X \sim \text{Poisson}$  (räknar förekomster)

$\Gamma$  (gamma) (undantag)

neg-binomial (tidsperioder, accumulation)

$IG$  (inverse gaussian) (stumpöpelse)

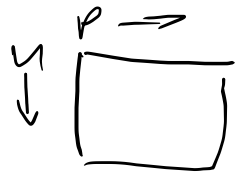
logit odds  
 $\eta / (1 - \eta)$

kostnadsfunktionerna ger oss regressionsmetoder:

$$C(\beta) = \text{MSE} \quad \text{OLS}$$

$$C(\beta) = \text{MSE} + \sum_{i=1}^p \beta_i^2 \quad \text{Ridge } l_2$$

$$C(\beta) = \text{MSE} + \sum_{i=1}^p |\beta_i| \quad \text{lasso } l_1$$



- kernel trick
- similarity

Att göra en regression  $\sim O(n^3)$

Vanligast metoden är SVD single value decomposition

$O(np^2) \rightarrow n$  är stickprovsstorlek  
 $p$  är antalet features

notera att våra parametrar, så  $g(x_1, x_2)$

mycket väl kan vara icke-linjära i  $Y$ .

$Y = \beta X$  är linjär, men  $X$  kan ha mkt fler features än parametrar. Tex polynomial expansion:

$f(x_1, x_2) \rightarrow$  två parametrar

$$\underline{f(u+v) \neq f(u) + f(v)}$$

$$Y = \underbrace{x_1 + x_2 + x_1^2 + x_2^2 + x_1 x_2}_{A+m \text{ features}}$$

$$u = (x_1, x_2)$$

$$g(x_1, x_2) = x_1 x_2$$

$$v = (a, b)$$

$$g(u) = x_1 x_2$$

$$g(v) = a b$$

$$g(u+v) = (x_1 + a)(x_2 + b)$$

$$x_1 x_2 + a b$$

$$\neq$$

$$(x_1 + a)(x_2 + b)$$

Liten feature-mängd generaliserar bättre

(Ockham's Razor). Den reagerar bättre på oböjd data.  
Undviker overfit.

# Dimensionsreduktion

- minska antalet features

- Forward Selection

Börja med enkel linjär reg., på varje  $x_i$

välj den som ger bäst  $R^2$ /MSE.

Lägg till fler tills  $R^2$ /MSE inte förbättras.

- Backwards Elimination

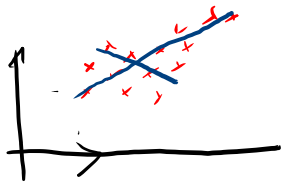
Börja med alla  $p$  parametrar.

För varje  $p-1$  parametrar, testa  $R^2$ /MSE

tag bort param. till  $R^2$ /MSE blir sämre

# PCA Principal Component Analysis

- Data-analys (dimensionsreducering)
- automatisk feature engineering
- överskådlig inläring



Hitta en bas till  $X$

skriv om  $X$  till  $Z$ :

$$z_1 = \phi_{11}x_1 + \phi_{21}x_2 + \dots + \phi_{p1}x_p$$

$\vdots$

$$z_2 = \phi_{12}x_1 + \dots + \phi_{p2}x_p$$

$$\sum_{i=1}^p \phi_{ji}^2 = 1$$

$$\max \frac{1}{n} \sum_{i=1}^n \left( \sum_{j=1}^p \phi_{ij} x_i \right)^2$$

notera: Varie principal komponent  
 efter den första förklarar mindre  
 och mindre av variansen i data.

Optimerings problemet ger oss maximal variation över  $Z$ .

Varie  $z_i$  är en principal komponent

$$Y = \beta X \quad \text{normalt}$$

$$\beta X = \bar{\Phi} X$$

delta fall

Oövervakad inläring!

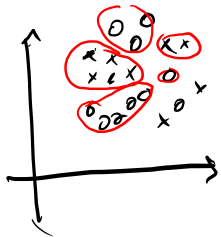
hitta grupperingar

generering

vi har ingen responsvektor!  
 inga lablar!

inga rätta svar!

(semi-supervised learning)



## Klustering

k-means algoritmen klustrar värden i  $k$  grupper.

k-means minimerar intraklustervariationen (dvs avståndet mellan punkterna i klustern).

$$\min_{c_1, \dots, c_k} \left[ \sum_{k=1}^k \frac{1}{|C_k|} \sum_{i,j \in C_k} \sum_{l=1}^p (x_{i,l} - x_{j,l})^2 \right] \quad \swarrow \text{similarity}$$

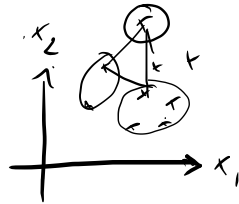
$|C_k|$  = antalet punkter i kluster  $k$ .



# Övervakad inlärning

- k-means
- PCA

\*



Skalning?

Standard skalning (centering)

inget y, alltså ingen test/train split

— självinlärning

Vanligt förfarande

1) k-means  $\rightarrow$  dim. reducer.

2) k-NN

kan se samband på  
"långt avstånd"