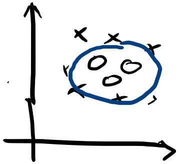
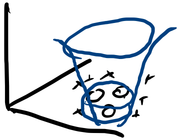
SVM Support vector machines

- Grundidé : hyperplan för att separera klasser





$O(n^3) \sim O(n^2)$

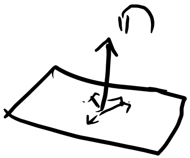
↑

design matrisens storlek

- skulle kunna göra feature expansion
- kerneltrick

· Hyperplan har vi redan sett men inte namngivit:

$$\beta_0 + \beta_1 x_1 + \dots + \beta_p x_p = 0$$



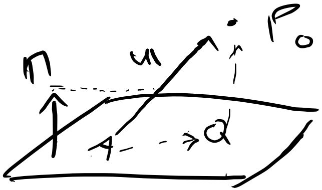
(Sparr 5.75) $(p = 3)$

$$ax + by + cz + d = 0$$

$\bar{n} = (a, b, c)$, om $p > 3$ $n = (\beta_1, \dots, \beta_p)$

$\beta_0 = 0$ så ingår origo, annars inte.

Avstånd punkt/plan



$P_0 = (x_0, y_0, z_0)$

$A = (p, q, r)$

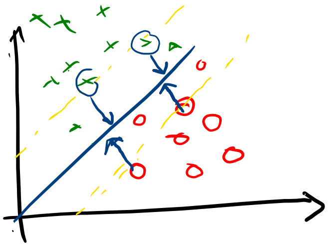
$$|\vec{QP}| = \frac{|(x_0-p)a + (y_0-q)b + (z_0-r)c|}{\sqrt{a^2+b^2+c^2}}$$

$\iff$

$$|\vec{QP}| = \left|\frac{u \cdot n}{n \cdot n} n\right| \quad \text{villkor: } |n| = 1 = \sqrt{\beta_1^2 + \dots + \beta_p^2}$$

$$n \cdot n = |n|^2 \qquad \sum_{i=1}^{p} \beta_i^2 = 1$$

$$\left|\frac{u \cdot n}{|n|^2}\right| |n| \to u \cdot n$$



Maximal margin

Om marginalen måste vara tom blir metoden känslig för brus.

→ Soft Margin

$$\max_{\beta_0, \dots, \beta_p} M$$

sådan att $\sum_{j=1}^{p} \beta_j^2 = 1$

(i → varje rad i indatan)

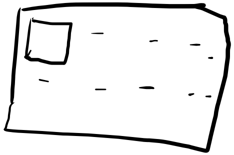
$$\beta_0 + \beta_1 x_{i1} + \dots + \beta_p x_{ip} \geq M(1 - \varepsilon_i)$$

$$\varepsilon_i \geq 0, \quad \sum_{i=1}^{n} \varepsilon_i \leq C \quad \text{(budget)}$$

· Kernel

- Matris med "vikter"

$$u_j \cdot v_i = \sum_{j=1}^{p} u_i v_i$$



$$\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \leftarrow$$

$$\langle u, v \rangle = u \cdot v \quad \text{(euclidiskt)}$$

Klassificeringen kan skrivas som funktion (svår härledning):

$$f(\underline{x}) = \beta_0 + \sum_{i=1}^{n} \alpha_i \langle \underline{x}, \underline{x}_i \rangle$$

$$\Rightarrow \beta_0 + \sum_{i \in S} \hat{\alpha}_i \langle x, x_i \rangle$$

S = support set

Stödvektorerna vi valt

Kernel funktioner

$$K(\mathbb{u}, \mathbb{v}) = \left(1 + \sum_{j=1}^{p} u_j v_j\right)^d \Leftrightarrow (1 + \mathbb{u} \cdot \mathbb{v})^d$$

beräknar inre produkter för d-dimensionella polynom.

$\binom{p+d}{d}$ bas, motsvarar polynom expansion

RBF (Radial basis funktion)

$$K(\mathbb{u}, \mathbb{v}) = e^{-\gamma \sum_{j=1}^{p} (u_j - v_j)^2}$$

$\gamma \gg 0$, liten σ^2

$\gamma \ll 1$, stor σ^2

(mjuka linjer)

$\gamma \Rightarrow$ hyperparameter

Multinomiella klasser

Ova : One-versus-all

ovo : one - versus-one

Kan bli dyrt för stora datamängden (dvs gigabyte +)

Gradient descent

grundidé: ta ett "litet" steg i en "bra" riktning för att nå minimum

Iterativa funktioner

Iteration

$$x_{n+1} = x_n$$

Basfall

$$x_0 = a$$

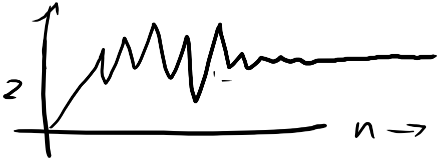
Logistisk ekvation (iterativ)

$$x_{n+1} = r(1 - x_n)$$

$$\boxed{x_0 = 0.5}$$

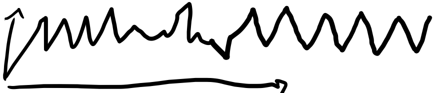
$r = 3.58$

≈!2



n →

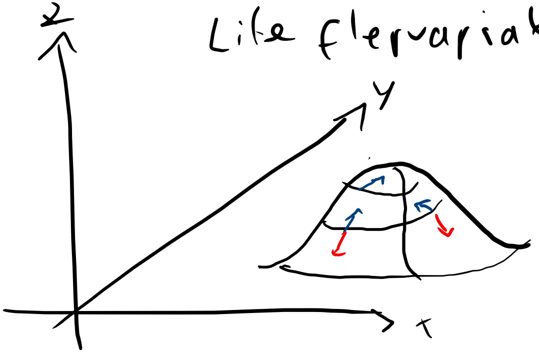
r ≈ 2

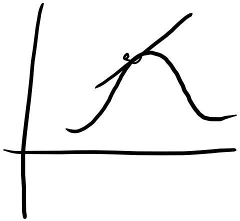


- Differens ekvation: itererar över tid
- Differential ekvation

"iteration" (integration) över kontinuerlig tid

Lite flervariabel:





$$\left(\frac{\partial RSS}{\partial b_0} , \frac{\partial RSS}{\partial b_1} \right) = -\nabla$$

- pekar ~~alltid~~ i den rikting som har störst förändringshastighet

jacobi matris
$$\begin{bmatrix} \frac{\partial x}{\partial a} & \frac{\partial y}{\partial b} \\ \vdots & \vdots \end{bmatrix}$$

OLS:

$$C(\beta) = MSE$$

Ridge

$$C(\beta) = MSE + \lambda \sum_{i=1}^{p} \beta_i^2$$

GD

$$C(\theta) = \frac{1}{n-k-1} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2$$

$$= \frac{1}{n-k-1} \sum_{i=1}^{n} (y_i - \theta \cdot x_i)^2$$

som iterativ (rekurrent) funktion:

som iterativ funktion:

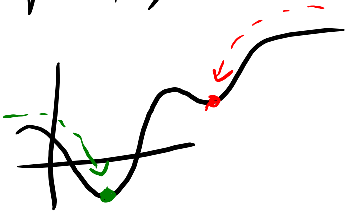
$$\theta_{i+1} = \theta_i - \eta \nabla C(\theta)$$

$$\theta_{j+1} = \theta_j - \eta \left(\frac{2}{n-k-1} X^T (X\theta_j - y) \right)$$

Batch $\longleftrightarrow$ stickprov (stat. sample)

Sample $\longleftrightarrow$ datapunkt (data point)

Batch Gradient Descent



tenderar att fastna i lokala min.

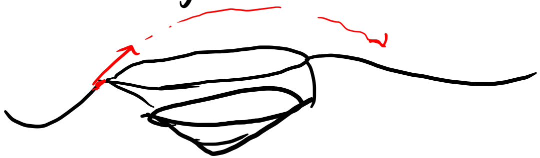
Stochastic Gradient Descent

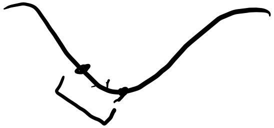
Välj en slumpmässig datapunkt.
Räkna ∇ på den punkten.
upprepa givet antal gånger.

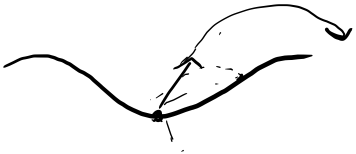
→ ger en "epok" (epoch)

η = hur stora steg tar vi

konvergerar ej!







Minibatch

Välj delmängd av punkter och räkna gradient annars som SGD.

Gradient Descent with Momentum

$$x_{k+1} = x_k - s z_k \qquad x_{k+1} = x_k - \eta \cdot \nabla f_k$$

$$z_k = \underline{\nabla f_k} + \underline{\beta z_{k-1}} \qquad x_{k+1} \; \dots \; x_k, \dots x_{k-1}$$