Tutorial For A3

Thursday, September 25, 2014 3:00 PM

1) ridge regression

$$w_{MAP} = \underset{w}{\operatorname{argmin}} \frac{1}{2} \|Xw - Y\|^2 + \frac{\lambda}{2} \|w\|^2$$

minimize objective function

dont want w too large
λ is large, more important to minimize this

think back to MLE:

$$w_{MLE} = \underset{w}{\operatorname{argmin}} \|Xw - Y\|^2$$

$$y = \underbrace{x^T w}_{\text{deterministic}} + \underbrace{n}_{\text{noise term}}, \quad n \sim N(0, \sigma^2)$$

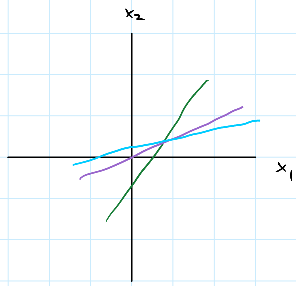
deterministic, gaussian, deterministic, probability distribution

ex: $n \sim N(0, \sigma^2)$
$p(y \mid w, x) = N(w^T x, \sigma^2)$

data given hypothesis

MLE: $p(D \mid h)$

MAP: $p(h \mid D)$



$w_1 x_1 + w_2 x_2$

given data, want to find weights

$p(w \mid y, x)$

(w that would maximize)

$\propto p(Y \mid w, x)\, p(w)$

corresponds to $p(y \mid w, x)$

corresponds to $p(w)$

$p(w \mid y, x) \propto \ldots$

$$\log(p(w \mid y, x)) = \log(p(Y \mid x, w)) + \log(p(w)) + c$$

$$-\log(p(w \mid y, x)) = -\log(p(Y \mid x, w)) - \log(p(w)) + c$$

$$-\log(p(w)) \propto \frac{\lambda}{2} \|w\|^2$$

$$p(w) \propto \exp\left(\frac{-\lambda}{2} \|w\|^2\right)$$

to get rid of $\propto$, have to normalize to sum up $\sum p = 1$

$$w \sim N(0, \tfrac{1}{\lambda} I)$$

$$w_i \sim N(0, \tfrac{1}{\lambda})$$

$\|w\|^2 = w^T I w$ (w is vector)

$$p(Y \mid x, w) \propto -\frac{1}{2} \|Xw - Y\|^2$$

$$= k \exp\left(\frac{-1}{2}(Xw-y)^T(Xw-y)\right)$$

$y \sim N(Xw, I)$ vector

$y_i \sim N(x_i^T w, 1)$ elements of the vector

$y_i \sim N(w^T x_i, 1)$

covariance

$w_i \sim N(0, \lambda_i)$ $\Rightarrow$ $\underset{w}{\operatorname{argmin}} \frac{1}{2}\|Xw - Y\|^2 + \frac{1}{2} w^T \Lambda w$

$$p(\bar{w}) = k \exp\left(\sum_{i=1}^{d} w_i^2 \left(-\frac{1}{2}\lambda_i\right)\right)$$

$$w^T \begin{pmatrix} \lambda_1 & & & \\ & \lambda_2 & & \\ & & \ddots & \\ & & & \lambda_d \end{pmatrix} w$$

$$\begin{pmatrix} \lambda_1 w_1 \\ \lambda_2 w_2 \\ \vdots \\ \lambda_d w_d \end{pmatrix}$$

---

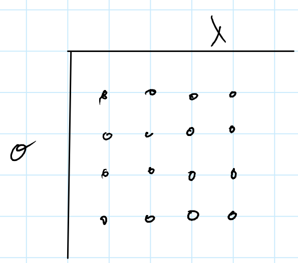
3)

divide into parts

I. X_{train} → $X_{train'}$ $\Rightarrow w$ (different w's depending on hyper parameters)

→ $X_{cross\ validation}$ $\Rightarrow$ parameters of the kernel

II. X_{test}



"hyper parameters"

⇒ iterate over $\lambda_i \, \sigma_j$ to find the best combination

---

4)

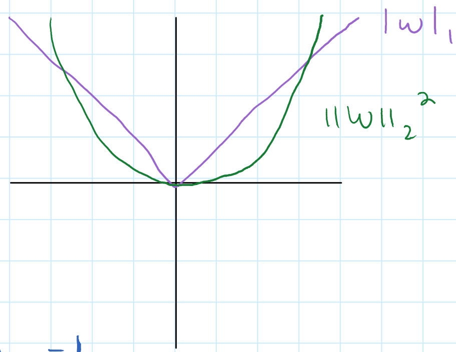
$$\underset{w}{\operatorname{argmin}} \ \frac{1}{2} \|Xw - Y\|^2 + \lambda \|w\|_1 = w_{MAP}$$

change to L1 norm

$p(Y|Xw) \quad p(w)$

→ laplacian (heavy tail)

$p(w_i) \propto \exp(-\lambda |w_i|)$



"this is the laplace distribution"

laplace: $f(z \mid \mu, b) = \frac{1}{2b} \exp\left(\frac{-|z-\mu|}{b}\right)$

$w_i \sim \text{laplace}(0, 1/\lambda)$

$$\int_{-\infty}^{\infty} p(w_i)\,dw_i = 1$$

upper bound L1 norm

$$\underset{w,v}{\operatorname{argmin}} \ \frac{1}{2} \|Xw - Y\|^2 + \lambda \sum V_i$$

$|w_i| < V_i$ each feature,

$w_i < V_i$ 2 constraints.

$-w_i < V_i$

---

5)

The algorithm is in the book.

Algorithm 13.1 : Coordinate Descent for lasso

Algorithm 13.1: Coordinate descent for lasso (aka shooting algorithm)

Initialize $\mathbf{w} = (\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}^T\mathbf{y}$;
repeat
  for $j = 1, \ldots, D$ **do**
    $a_j = 2\sum_{i=1}^n x_{ij}^2$;
    $c_j = 2\sum_{i=1}^n x_{ij}(y_i - \mathbf{w}^T\mathbf{x}_i + w_j x_{ij})$;
    $w_j = \text{soft}(\frac{c_j}{a_j}, \frac{\lambda}{a_j})$;
until *converged*;