

1 Math

Bayes rule: $P(A|B) = \frac{P(A,B)}{P(B)} = \frac{P(B|A)P(A)}{P(B)}$

$$P(A,B|C) = \frac{P(A,B,C)}{P(C)}$$

Cond indep: $P(A,B|C) = P(A|C)P(B|C)$

Chain rule: $\frac{P(A_n, \dots, A_1)}{P(A_n|A_{n-1}, \dots, A_1) \cdot P(A_{n-1}, \dots, A_1)}$

Joint: $P\left(\bigcap_{k=1}^n A_k\right) =$

$$\prod_{k=1}^n P\left(A_k \middle| \bigcap_{j=1}^{k-1} A_j\right)$$

Posterior: unobserved θ , observed x :

$$p(\theta|x) = \frac{p(x|\theta)p(\theta)}{p(x)}$$

Prior: prior belief in dist. of θ : $p(\theta)$

Likelihood: prob of observed given parameters: $p(x|\theta)$

MAP: $h_{MAP} = \arg\max_{h \in H} P(h|D) = \arg\max_{h \in H} P(D|h)P(h)$

Lagrange: $L(\mathbf{x}, \lambda) = f(\mathbf{x}) + \lambda g(\mathbf{x})$ Set constraint $g(\mathbf{x})$ to zero and add multiplier.

$$\mathbf{xy}^T = \begin{bmatrix} x_1 \\ \vdots \\ x_m \end{bmatrix} \begin{bmatrix} x_1 \dots x_n \end{bmatrix} = \begin{bmatrix} x_1 y_1 & \dots & x_1 y_n \\ \vdots & \ddots & \vdots \\ x_m y_1 & \dots & x_m y_n \end{bmatrix}$$

$$\text{Matrix-vector product: } \mathbf{Ax} = \begin{bmatrix} a_1^T \mathbf{x} \\ \vdots \\ a_m^T \mathbf{x} \end{bmatrix}$$

Matrix mult': $(\mathbf{AB})_{ij} = \sum_{k=1}^m A_{ik}B_{kj}$

$$\mathbf{AB} = \begin{bmatrix} a_1^T b_1 & a_1^T b_2 & \dots & a_1^T b_p \\ \vdots & \vdots & \ddots & \vdots \\ a_m^T b_1 & a_m^T b_2 & \dots & a_m^T b_p \end{bmatrix}$$

$$(n \times p)(p \times m) = n \times p$$

2 todo

$$\begin{aligned} \Pr(X = x) &= \sum_y \Pr(X = x, Y = y) = \\ &= \sum_y \Pr(X = x | Y = y) \Pr(Y = y) \end{aligned}$$

• Eigendecomposition

$$\sin(x)' = \cos x$$

$$\cos(x)' = -\sin x$$

•

$$(\mathbf{AB})^T = \mathbf{B}^T \mathbf{A}$$

$$\log_b(x^y) = y \log_b(x)$$

$$\text{Uniform distribution: } P_{\theta_1, \theta_2}(x) = \frac{1}{\theta_2 - \theta_1}$$

$$\mathbf{AA}^{-1} = \mathbf{A}^{-1} \mathbf{A} = \mathbf{I} \text{ for square matrices.}$$

$$F'(x) = f'(g(x))g'(x)$$

$$(f \cdot g)' = f' \cdot g + f \cdot g'$$

• Log properties

• Convexity

$$\|\mathbf{x}\|_d = \sum_i x_i^d$$

• Polynomial multiplication

• Matrix transpose and inverse tproperties

• Gradient vs partial derivative

$$\sigma(x)' = \sigma(x)(1 - \sigma(x))$$

$$\begin{aligned} \mathbb{E}\left[\left(y - \hat{f}(x)\right)^2\right] &= \mathbb{E}\left[\hat{f}(x) - f(x)\right]^2 + \\ &\mathbb{E}[\hat{f}(x)^2] - \mathbb{E}[f(x)]^2 + \epsilon^2 \end{aligned}$$

3 Regression

3.1 L2 regularization

Weights do not reach zero. Faster.

$$J_{\mathbf{w}} = \frac{1}{2}(\Phi \mathbf{w} - \mathbf{y})^T (\Phi \mathbf{w} - \mathbf{y}) + \frac{\lambda}{2} \mathbf{w}^T \mathbf{w}$$

$$\mathbf{w} = (\Phi^T \Phi + \lambda \mathbf{I})^{-1} \Phi^T \mathbf{y}$$

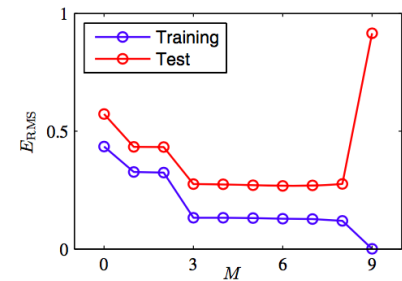
3.2 L1 regularization

Some weights set to zero. More expensive.

3.3 Gradient Descent

$$\mathbf{w} \leftarrow \mathbf{w} - \alpha \nabla \log L(\mathbf{w}) \quad \mathbf{w} \leftarrow \mathbf{w} - \alpha \nabla \log J(\mathbf{w})$$

3.4 Bayesian regularization



4 Kernels

$k(\mathbf{x}, \mathbf{z}) = \phi(\mathbf{x}) \cdot \phi(\mathbf{z})$ **Mercer theorem:** $K(\mathbf{x}, \mathbf{z})$ is kernel iff Gram matrix \mathbf{K} symmetric and positive semidefinite: $\mathbf{K}_{ij} = \mathbf{K}_{ji}$ and $\mathbf{z}^T \mathbf{K} \mathbf{z} \geq 0$ $\mathbf{K} = \Phi \Phi^T$ where $\mathbf{K}_{nm} = \phi(\mathbf{x}_n)^T \phi(\mathbf{x}_m) = k(\mathbf{x}_n, \mathbf{x}_m)$ Gram matrix size of input.

• Kernel properties

• Proving kernels

5 SVM

Minimize absolute error. More robust to outliers. Max margin is convex optimization.

$$h_{\mathbf{w}}(\mathbf{x}) = \text{sign}\left(\sum_{i=1}^m \alpha_i y_i (\mathbf{x}_i \mathbf{x}) + \mathbf{w}_0\right)$$

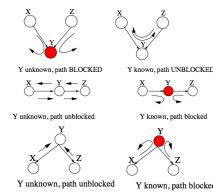
$\alpha_i > 0$ only for support vectors. Soft error SVM: $0 < \zeta \leq 1$ if inside margin. $\zeta > 1$ if misclassified. Total errors: $C \sum_i \zeta$. Large C higher variance.

6 EM/Active Learning/Missing Data

When missing data: many local maxima (normally likelihood has unique max), no closed form solutions. So we do gradient ascent or EM. $\log L(\theta) = \sum_{\text{complete data}} \log P(\mathbf{x}_i, y_i | \theta) + \sum_{\text{incomplete data}} \log \sum_y P(\mathbf{x}_i, y | \theta)$. **E step:** compute expected assignment (hard or soft) of points to distributions (estimate $p(y_i = k | \theta)$). **M step:** recompute parameters to maximize likelihood of current assignments $p(\theta | y_i)$. Good for low dimensionality data.

7 Bayes Nets

$$p(x_1, \dots, x_n) = \prod_{i=1}^n p(x_i | \pi_i)$$



Markov blanket: parents, children, spouses.

Moral graph: graph U : edge $(X, Y) \in U$ if X in Y 's markov blanket.

$$\begin{aligned} \text{Belief propagation: } m_{ji} &= \\ \sum_{x_j} \left(\psi^E(x_j) \psi(x_i, x_j) \prod_{k \in \text{enghbr}(x_j)} m_{kj}(x_j) \right) \\ p(y | \hat{x}_E) &\propto \psi^E(y) \prod_{k \in \text{enghbr}(Y)} m_{ky}(y) \end{aligned}$$

