Linear Classification

Perceptron Alg. ! Work for linearly separable D ! Not linearly separable : map to higher dim feature space, in which it is linearly separable. Normalize datapoints is good. Terminates after no more than $1/\gamma^2$ updates, where $\gamma = \min_{i=1,\dots,n} t_i \mathbf{w}_*^T \tilde{\phi}_i$
Normalize features $\phi_i \to \tilde{\phi}_i$!
Update rule (when misclassified $t_i \mathbf{w}^T \tilde{\phi}_\mathbf{i} \leq 0$): $\mathbf{w} \leftarrow \mathbf{w} + t_i \tilde{\phi}_\mathbf{i}$ until $t_i \mathbf{w}^T \tilde{\phi}_\mathbf{i} > 0$ for all i

Stochastic Gradient Descent $\mathbf{w}_{k+1} = \mathbf{w}_k - \eta \nabla_\mathbf{w} E_i(\mathbf{w}_k)$

Least square estimation

$$E(\mathbf{w}) = \frac{1}{2}\sum_{i=1}^{n}(y(\mathbf{x}_i) - t_i)^2 = \frac{1}{2}\|\mathbf{\Phi w} - \mathbf{t}\|^2$$

Gradient for Squared Error (yields normal equation for least squares if set to 0)

$$\nabla_\mathbf{w} E = \sum_{i=1}^{n} \frac{\partial E}{\partial y_i} \nabla_\mathbf{w} y_i = \mathbf{\Phi}^T(\underbrace{\mathbf{\Phi w} - \mathbf{t}}_{residual})$$

Multi-Layer Perceptron

Overfitting : Split data, early stopping. Parameters : w and b. Layers : Do not count entry but count output. Error function : not good if not bounded below, not good ig grow too fast for large residuals. Momentum : prevent zig zag, allow higher learning rates.

Forward pass

$$a_q^l = (\mathbf{w}_q^{(l)})^T \mathbf{z}^{(l-1)} + b_q^{(l)}$$

$$z_q^{(l)} = g(a_q^{(l)}),\ q = 1,..,h_l$$

Backward pass

$$r^{(L)} = \frac{\partial E_i}{\partial a^{(L)}} = \begin{cases} a^L - t_i & for\ E_{sq} \\ \sigma(a^L) - \tilde{t_i} & for\ E_{log} \end{cases}$$

$$r_q^{(l)} = g'(a_q^{(l)}) \sum_{j=1}^{h_{l+1}} w_{jq}^{(l+1)} r_j^{(l+1)}$$

Gradient computation

$$\nabla_{w^{(l)}} E_i = r^{(l+1)}\mathbf{z}^{(l)},\ \nabla_{w_q^{(1)}} E_i = r_q^{(1)}\mathbf{x}$$

For b: keep residual only ! (i.e $\mathbf{x} = 1$)

Momentum learning

$$\Delta\mathbf{w}_k = \mathbf{w}_{k+1} - \mathbf{w}_k$$

$$\Delta\mathbf{w}_k = -\eta(1-\mu)\nabla_{\mathbf{w}_k} E + \mu \Delta\mathbf{w}_{k-1}$$

η = learning rate e.g. $1/k$, μ = momentum term

Linear Regression. LSE

$E(a,b) = \frac{1}{2}\sum_{i=1}^n (y(x_i) - t_i)^2$ $\frac{\partial E}{\partial a} = \sum_i (ax_i + b - t_i)x_i = n(a\langle x^2\rangle + b\langle x\rangle - \langle tx\rangle)$ $\frac{\partial E}{\partial b} = \sum_i (ax_i + b - t_i) = n(a\langle x\rangle + b - \langle t\rangle)$ $a_* = \frac{\langle tx\rangle - \langle t\rangle\langle x\rangle}{\langle x^2\rangle - \langle x\rangle^2}$ $b_* = a_*\langle x\rangle + \langle t\rangle$
Overfit : add tikhonov regularizer

Univariate Linear Regression $\langle x\rangle = n^{-1}\sum_i x_i$. Similar for t, tx, x^2

$$y_i = wx_i + b$$

$$\to w = \frac{Cov(x,t)}{Var(x)}, b = \langle t\rangle - w\langle x\rangle$$

Normal Equations $(\mathbf{\Phi}^T\mathbf{\Phi})\mathbf{w} = \mathbf{\Phi}^T\mathbf{t}$

$$\hat{\mathbf{w}} = \underset{w}{\mathrm{argmin}}\, E(\mathbf{w}) = (\mathbf{\Phi}^T\mathbf{\Phi})^{-1}\mathbf{\Phi}^T\mathbf{t}$$

Probability. Decision Theory

Probability Independence : $p(a,b) = p(a)p(b)$ $E[X] = E[E[X|Y]]$
Sum rule: $P(X) = \sum_Y P(X,Y)$
Product rule: $P(X,Y) = P(X|Y)P(Y)$
Bayes $P(B|F) = \frac{P(F|B)P(B)}{P(F)}$
$Var(t) = E[Var(t|x)] + Var(E[t|x])$

Bayes-Optimal Classifier Do the best choice.

$$f^*(\mathbf{x}) = \underset{t\in\tau}{\mathrm{argmax}}\ \underbrace{P(t|\mathbf{x})}_{p(\mathbf{x}|t)P(t)}$$

Bayes error: $R = P\{f(x) \neq t\}$

$$R^* = R(f^*) = 1 - \mathrm{E}\left[\max_{k\in\tau} P(t=k|\mathbf{x})\right]$$

Optimal discriminant:

$$y^*(x) = \log\frac{p(\mathbf{x}|t=1)}{p(\mathbf{x}|t=0)} + \log\frac{P(t=1)}{P(t=0)} > 0 \to f(x) = 0$$

Bayes error : when the classifier is wrong.

Bayes under Loss function Risk: $R(f) = \mathrm{E}\left[L(f(\mathbf{x}), t)\right]$ Opt. Classifier:

$$f^*(\mathbf{x}) = \underset{j\in\tau}{\mathrm{argmin}} \sum_{k\in\tau} L(j,k)P(t=k|\mathbf{x})$$

$$R^* = \mathrm{E}\left[\min_{j\in\tau}\sum_{k\in\tau} L(j,k)P(t=k|\mathbf{x})\right]$$

Probab. Models. Max. Likelihood

Likelihood function : $\prod_{i=1}^n p(x_i|\gamma)$, derive using $\bar{x} = n^{-1}\sum_i x_i$, $p_{mixture}(x) = \sum_{k=1}^{blu} P(\omega_k)P(x|\gamma_k)$

MLE $\hat{p}_1 = \mathrm{argmax}_{p_1\in[0,1]} P(D|p_1)$
Maximize $\log P(D|p_1)$: $\frac{d\log P(D|p_1)}{dp_1} = 0$ or minimize $-\log P(D|p_1)$

Gaussian

$$N(x|\mu,\sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

Multivariate $N(\mathbf{x}|\mu, \mathbf{\Sigma}) =$

$$|2\pi\mathbf{\Sigma}|^{-1/2} e^{-\frac{1}{2}(\mathbf{x}-\mu)^T\mathbf{\Sigma}^{-1}(\mathbf{x}-\mu)}$$

Covariance $Cov(\mathbf{x},\mathbf{y}) = \mathrm{E}[\mathbf{x}\mathbf{y}^T] - \mathrm{E}[\mathbf{x}]E[\mathbf{y}]$, $Cov[\mathbf{Ax}] = \mathbf{A}Cov[\mathbf{x}]\mathbf{A}^T$, Sample

$$\mathbf{S} = \frac{1}{n}\sum_{i=1}^n (\mathbf{x_i} - \bar{\mathbf{x}})(\mathbf{x_i} - \bar{\mathbf{x}})^T = \frac{1}{n}\mathbf{X}^T\mathbf{X}$$

if $\bar{x} = \frac{1}{n}\sum_{i=1}^n x_i = 0$

Correlation

$$\frac{Cov(x_j, x_k)}{\sqrt{Var(x_j)Var(x_k)}} \in [-1, 1]$$

ML plugin discriminant

$$\hat{y}(\mathbf{x}) = \hat{\mathbf{w}}^T\mathbf{x} - \frac{1}{2}(||\hat{\mu}_{+1}||^2 - ||\hat{\mu}_{-1}||^2) + \log\frac{\hat{\pi}_1}{1-\hat{\pi}_1}\ where\ \hat{\pi}_1 = n_1/n\ and$$

$$\hat{\mathbf{w}} = \hat{\mu}_{+1} - \hat{\mu}_{-1}$$

$$y_k^*(\mathbf{x}) = -\frac{1}{2}||\mathbf{x} - \mu_k||^2 + \log P(t=k) + C$$

Naive Bayes Classifier $\hat{P}(\mathbf{x}|t=k)\hat{P}(t=k) = (\prod_{m=1}^M (\hat{p}_m^{(k)})^{\phi_m(\mathbf{x})}(1-\hat{p}_m^{(k)})^{1-\phi_m(\mathbf{x})})\frac{n_k}{n}$
$\hat{p}_m^{(k)} = \frac{\sum_{i=1}^n I_{\{t_i=k\}}\phi_m(\mathbf{x}_i)}{n_k}$

$$P(\mathbf{x}|N, t=k) = \prod_{m=1}^M \left(p_m^{(k)}\right)^{\phi_m(\mathbf{x})}$$

with $\phi_m(\mathbf{x}) = \sum_{j=1}^N I_{\{x_j=m\}}$

Generalization. Regularization

Training error:
$\hat{R}_n(f) = \frac{1}{n}\sum_{i=1}^n I_{\{(f(\mathbf{x_i})\neq t_i\}}$
Generalization error:
$R(f) = E^*[I_{\{(f(\mathbf{x})\neq t\}}]$
Tikhonov Regularization term $\frac{\nu}{2}||\mathbf{w}||^2$ $\to (\mathbf{\Phi}^T\mathbf{\Phi} + \nu\mathbf{I})\mathbf{w} = \mathbf{\Phi}^T\mathbf{t}$

MAP $\hat{p_1} = \mathrm{argmax}_{p_1}\, p(p_1|D) = \mathrm{argmax}_{p_1}\, P(D|p_1)p(p_1)$

Cond. Likelihood. Logistic Reg.

Maximize Conditional Likelihood $p(\mathbf{t}|\mathbf{w}) = \prod_{i=1}^n p(t_i|y_i)$

Conditional Likelihood Bridge

Likelihood $p(\mathbf{t}|\theta) \overset{-log}{\longrightarrow} E(t,\theta)$ Loss function

Logistic regression $P(t|y) = \sigma(ty)$
$E_{log}(\mathbf{w}) = -\log P(\mathbf{t}|\mathbf{w})$
$= \sum_{i=1}^n \log\left(1 + e^{-t_i y_i}\right)$
$= \sum_{i=1}^n -\log\sigma(t_i y_i)$

Generative Modeling $p(\mathbf{x},t|\theta) = p(\mathbf{x}|t,\theta)P(t|\theta)$ joint max. likeli. $max_\theta \prod_{i=1}^n p(\mathbf{x_i},t_i|\theta)$

Discriminative Modeling $P(t|\mathbf{x},\theta) \rightarrow max_\theta \prod_{i=1}^n P(t_i|\mathbf{x_i},\theta)$ conditional maximum likelihood

Multiway logistic Regression $P(t=k|\mathbf{x}) = \frac{e^{y_k^*(\mathbf{x})}}{\sum_{\tilde{k}} e^{y_{\tilde{k}}^*(\mathbf{x})}} = \sigma_k(\mathbf{y}^*(\mathbf{x}))$

Soft-max mapping $\sigma_k(\nu) = e^{\nu_k - lsexp(\nu)}$ with $lsexp(\nu) = \log \sum_{\tilde{k}} e^{\nu_{\tilde{k}}}$ $\nabla_v lsexp(\mathbf{v}) = \sigma(\mathbf{v})$

Support Vector Machines

Kernel function: $K(\mathbf{x},\mathbf{x}') = \phi(\mathbf{x})^T\phi(\mathbf{x}')$

Gaussian: $e^{-\frac{\tau}{2}||\mathbf{x}-\mathbf{x}'||^2}$, $\tau > 0$

Polynomial: $K(\mathbf{x},\mathbf{x}') = (\mathbf{x}^T\mathbf{x}')^r$

Max margin perceptron

$$\max_{w,b}\left\{\gamma_D(\mathbf{w},b) = \min_{i=1..n} \frac{t_i(\mathbf{w}^T\phi(\mathbf{x_i})+b)}{||\mathbf{w}||}\right\}$$

! D linearly separable !

Hard margin (convex optimization problem): $\min_{w,b} \frac{1}{2}||\mathbf{w}||^2$ subj. to $t_i(\mathbf{w}^T\phi(\mathbf{x_i})+b) \geq 1, i = 1..n$

Soft margin SVM w^2 regul term

$$\min_{w,b,\xi} \frac{1}{2}||\mathbf{w}||^2 + C\sum_{i=1}^n \xi_i$$

subj. to $t_i(\mathbf{w}^T\phi(\mathbf{x_i})+b) \geq 1-\xi_i,\ \xi_i \geq 0$

$$\equiv \min_{w,b} \frac{1}{2C}||\mathbf{w}||^2 + \underbrace{\sum_{i=1}^n [1-t_iy_i]_+}_{E_{svm}(w,b)}$$

Repr. Thm $\mathbf{w}_* = \sum_{i=1}^n \alpha_{*,i}\phi(x_i)$

$$\rightarrow y(\mathbf{x}) = \sum_{i=1}^n \alpha_i K(\mathbf{x},\mathbf{x_i}) + b$$

Solution Primal/Dual: $p_* = \min_{\mathbf{w},b}\max_{0\leq\alpha_i\leq C} L(\mathbf{w},b,\alpha)$ d_* max-min (reversed).Weak duality $d_* \leq p_*$ (strong duality iff =) $L(\mathbf{w},b,\alpha) = \frac{1}{2}||\mathbf{w}||^2 + \sum_{i=1}^n \alpha_i(1-t_iy_i)$

Maximize Criterion for dual:

$$\phi_D(\alpha) = \sum_i \alpha_i - \frac{1}{2}\sum_{i,j}\alpha_i t_i K_{ij} t_j \alpha_j$$

subj. to $\alpha_i \in [0,C], \sum_i \alpha_i t_i = 0$

Discriminant $y^*(\mathbf{x}) = \mathbf{w_*}^T\phi(\mathbf{x}) = \sum_{i=1}^n \alpha_{*,i}t_iK(\mathbf{x},\mathbf{x_i}) + b_*$

$b = \frac{1}{|S|}\sum_{i\in S}(t_i - \tilde{y}_i)$, S = essential support vectors

Support vectors

$$\begin{cases} \alpha_i = 0 & 1-t_iy_i \leq 0 \ \textit{not} \\ \alpha_i \in (0,C) & 1-t_iy_i = 0 \ \textit{essential} \\ \alpha_i = C & 1-t_iy_i \geq 0 \ \textit{bound} \end{cases}$$

Model Selection and Evaluation

Bias-Variance Decomposition

$$E[(\hat{y}(\mathbf{x}|D) - E[t|x])^2|\mathbf{x}] = \underbrace{(E[\hat{y}(\mathbf{x}|D)|\mathbf{x}] - E[t|\mathbf{x}])^2}_{Bias^2} + \underbrace{Var(\hat{y}(x|D)|\mathbf{x})}_{Variance}$$

Ens. meth. $\hat{y}_{ens}(\mathbf{x}) = \frac{1}{L}\sum_{l=1}^L \hat{y}_l(\mathbf{x})$

CV $\hat{R}_{CV}^{(M)}(D) = \frac{1}{n}\sum_{i=1}^n L(\hat{y}_\nu^{-m(i)}, t_i)$

Dimensionality Reduction

PCA First principal component direction $\mathbf{v}$ is the unit norm eigenvalue corresponding to the largest eigenvalue of S. Symmetry : empirical mean 0, can take half of the points. $\mathbf{S} = \frac{1}{n}\sum_{i=1}^n(\mathbf{x}_i - \bar{\mathbf{x}})(\mathbf{x}_i - \bar{\mathbf{x}})^T$ $\bar{\mathbf{x}} = \frac{1}{n}\sum_{i=1}^n \mathbf{x}_i$, $\mathbf{z} = \mathbf{U^T x}$ *with* $U^TU = I_{M\times M}$

$$\mathbf{u}_* = \underset{\mathbf{u}:||\mathbf{u}||=1}{\mathrm{argmax}}\ \mathbf{u}^T\mathbf{S}\mathbf{u}$$

PC directions = eigendirections of $Cov(\mathbf{x})$ **Goals**:maximize $Cov(z)$ / Minimize $E[||\hat{x}-x||^2]$ / decorrelate components of $\mathbf{z}$

! PCA doesn't depends on labels t !

Fischer

$$J(\mathbf{u}) = \frac{(m_1-m_0)^2}{s_0^2+s_1^2} = \frac{\mathbf{u}^TS_B\mathbf{u}}{\mathbf{u}^TS_W\mathbf{u}}$$

with $m_k = \mu^T\mu_k$. Maximize ratio!

$$S_B = (\hat{\mu}_1 - \hat{\mu}_0)(\hat{\mu}_1 - \hat{\mu}_0)^T = \mathbf{dd}^T$$

$$S_W = n^{-1}\sum_{i=1}^n(\mathbf{x_i} - \hat{\mu}_{\mathbf{t_i}})(\mathbf{x_i} - \hat{\mu}_{\mathbf{t_i}})^T$$

$$\hat{\mu}_{FLD} = \frac{S_W^{-1}\mathbf{d}}{||S_W^{-1}\mathbf{d}||},\ \mathbf{d} = \hat{\mu}_1 - \hat{\mu}_0$$

Total covariance $S = S_W + \alpha(1-\alpha)\mathbf{dd}^T$ with $\alpha = \frac{n_1}{n}$

LDA Generalization for multiple classes: $S = S_W + S_B$, $S_B = n^{-1}\sum_k n_k\mathbf{d}_k\mathbf{d}_k^T$, $\mathbf{d}_k = \hat{\mu}_k - \hat{\mu}$

$$\max_{U\in\Re^{d\times M}} tr(U^TS_BU)\ s.t.\ U^TS_WU = I$$

Unsupervised Learning

Kmeans Iterate until assignment no longer change. Assignment step: $||\mathbf{x_i} - \mu_{t_i}|| = \min_{k=1..K}||\mathbf{x_i} - \mu_k||$ Update prototypes: $\mu_k = n_k^{-1}\sum_{i=1}^n I_{\{t_i=k\}}\mathbf{x_i}$

$$\phi(\mathbf{t},\mu) = \sum_{i=1}^n\sum_{k=1}^K I_{\{t_i=k\}}||\mathbf{x_i} - \mu_\mathbf{k}||^2$$

Gaussian Mixture Model Soft assignement : assign to clusters with probabilities. If one mixture component assigned to a single central datapoint, its variance will shrink to small values : EM diverge with lare likelihood values.

$$p(\mathbf{x}) = \sum_{k=1}^K p(\mathbf{x}|t=k)P(t=k) = \sum_{k=1}^K N(\mathbf{x}|\mu_\mathbf{k},\Sigma_k)P(t=k)$$

Compute: $n_k = \sum_{i=1}^n P(t_i = k|\mathbf{x_i})$

Update: $\pi_k = \frac{n_k}{n}$

$\mu_k = \frac{1}{n_k}\sum_{i=1}^n P(t_i = k|\mathbf{x_i})\mathbf{x_i}$

Expectation Maximization E-step: $Q_i(\mathbf{h_i}) \leftarrow P(\mathbf{h_i}|\mathbf{x_i},\theta)$ M-step: maximize surrogate criterion

$$E(\theta;\{Q_i\}) = \sum_{i=1}^n E_i(\theta;Q_i) = \sum_{i=1}^n E_{Q_i}[\log P(\mathbf{x_i},\mathbf{h_i}|\theta)]$$

Perfect for missing data ! Hint: $\frac{\partial \log p(x_i)}{\partial\gamma_k} = \frac{1}{p(x_i)}\frac{\partial p(x_i|\omega_k)P(\omega_k)}{\partial\gamma_k}$ $= \frac{p(x_i|\omega_k)P(\omega_k)}{p(x_i)}\frac{\partial \log p(x_i|\omega_k)}{\partial\gamma_k}$ $= P(\omega_k|x_i)\frac{\partial \log p(x_i|\omega_k)}{\partial\gamma_k}$ Then insert derivative already computed (right part)

Beautiful Maths

Cauchy-Schwarz $|\mathbf{a}^T\mathbf{b}| \leq \|\mathbf{a}\|\|\mathbf{b}\|$

Logistic function $\sigma(v) = \frac{1}{1+e^{-v}}$, $\sigma'(v) = \sigma(v)\sigma(-v) = \sigma(v)(1-\sigma(v))$

tanh $g(a) = \tanh(a) = \frac{e^a - e^{-a}}{e^a + e^{-a}}$

$$g(a)' = 1 - g(a)^2$$

Trace $tr(\mathbf{A}) = \sum_{j=1}^d a_{jj} = \sum_{j=1}^d \lambda_j$

$$\mathbf{x}^T\mathbf{A}\mathbf{x} = tr(\mathbf{x}^T\mathbf{A}\mathbf{x}) = tr(\mathbf{A}\mathbf{x}\mathbf{x}^T)$$

Eigs $\mathbf{Av} = \lambda\mathbf{v}$, $\mathbf{A} = \mathbf{U\Lambda U^{-1}}, |A - \lambda I| = 0$

$|A| = \prod_{j=1}^d \lambda_j$

Positive semi-definite matrix $\mathbf{A} \in \Re^{d\times d}$ symmetric: $\mathbf{v}^T\mathbf{A}\mathbf{v} \geq 0\ \forall\mathbf{v} \in \Re^d, \mathbf{v} \neq 0$

Beta(α,β)

$$p(p_1|\alpha,\beta) = \frac{1}{B(\alpha,\beta)}(p_1)^{\alpha-1}(1-p_1)^{\beta-1}$$

With $B(\alpha,\beta) = \frac{\Gamma(\alpha)\Gamma(\beta)}{\Gamma(\alpha+\beta)}$. Mode $\frac{\alpha-1}{\alpha+\beta-2}$

Hinge function $[x]_+ = max(x,0)$

Cross-entropy, divergence

$$D(\mathbf{q}||\mathbf{p}) = \sum_{l=1}^L q_l\log(\frac{q_l}{p_l}) \geq 0$$