Linear regression: $y_{new} = \theta^T x_{new}$

Cost function:
$$J(\theta) = \sum_{i=1}^{N} (y^{(i)} - h_\theta(x^{(i)}))^2$$
$$= \sum_{i=1}^{N} (y^{(i)} - \theta^T x^{(i)})^2$$

also: $J(\theta) = \| \underset{(N\times d+1)}{X} \underset{(d+1,1)}{\theta} - Y \|_2^2$

$\hat{\theta} = \arg\min_\theta J(\theta)$ $\Downarrow \frac{\partial}{\partial\theta}$

$\Rightarrow \nabla J(\theta) = \frac{\partial J(\theta)}{\partial \theta} = 2X^T(X\theta - Y)$

$\Rightarrow X^T(X\theta - Y) = 0 \Rightarrow \boxed{\hat{\theta} = (X^TX)^{-1}X^TY}$ method I

method 2: (Gradient descent)

$$\theta^{(t)} = \theta^{(t-1)} - \rho \left.\frac{\partial J}{\partial \theta}\right|_{\theta^{(t-1)}}$$ ρ: learning rate

$$\Rightarrow \boxed{\theta^{(t)} = \theta^{(t-1)} - \rho X^T(X\theta^{(t-1)} - Y)}$$

method 3: stochastic Gradient descent

X is a sample of original [X] $N\times d+1$

$X \Rightarrow \begin{bmatrix} \cdots & 1 \\ & 1 \\ & 1 \end{bmatrix}_{N(d+1)}$

$\theta = \begin{bmatrix} \theta_1 \\ \theta_2 \\ \theta_d \\ b \end{bmatrix}_{(d+1)\times 1}$

Nonlinear features:

$x = \begin{pmatrix} x_1 \\ \vdots \\ x_d \end{pmatrix} \Rightarrow \varphi(x) = \begin{pmatrix} x_1 \\ x_1^2 \\ x_2 \\ x_2^2 \\ 1 \end{pmatrix}$

use $\varphi(x)$ instead of x

Generally: Cost function:

regularization: $J(\theta) = \sum_{i=1}^{N} l(y^{(i)} - \theta^T x^{(i)}) + \lambda f(\theta)$

where $\begin{cases} l(e) = e^2, f(\theta) = \|\theta\|_2^2 \longrightarrow \text{Ridge} \\ l(e) = e^2, f(\theta) = \|\theta\|_1 \longrightarrow \text{Lasso} \\ l(e) = |e|, \longrightarrow \text{Robust} \end{cases}$

Ridge Regression:

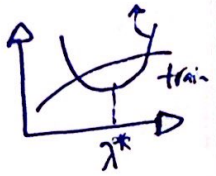
$$\boxed{\hat{\theta}_\lambda = (X^TX + \lambda I)^{-1} X^TY}$$

$\frac{\partial J(\theta)}{\partial(\theta)} = 0$

Set λ: K-fold cross validation

every time one $D^{(i)}$ is hold-out set, rest is tr set. . . .

find λ with min error:



Maximum Liklihood of observation ↓ Training

1) Coin problem:

$x \in \{0,1\}$ $P(x=1) = \theta$, $P(x=0) = 1-\theta$

$P(x) = \theta^x (1-\theta)^{1-x}$

$\max_\theta \log P(D|\theta) = \hat{\theta}_{ML} = \arg\max_\theta P(D|\theta)$

$l = \log P(D|\theta) = (\sum_{i=1}^{N} x^{(i)}) \log\theta + (N - \sum_{i=1}^{N} x_i) \log(1-\theta)$

$\Rightarrow \frac{\partial L}{\partial\theta} = 0 \Rightarrow \boxed{\hat{\theta}_{ML} = \frac{\sum_{i=1}^{N} x_i}{N}}$

2) linear regression:

$P(y^{(i)} | x^{(i)}, \theta) = \mathcal{N}(y^{(i)}, \mu = x^{(i)T}\theta, \sigma^2)$

$= \frac{1}{\sqrt{2\pi\sigma^2}} e^{-(y^{(i)} - x^{(i)T}\theta)^2 / 2\sigma^2}$

$\Rightarrow \boxed{\max_\theta \log P(D|\theta) = \hat{\theta}_{ML}}$

$l = \log P(D|\theta) \Rightarrow \hat{\theta} = \min_\theta \sum_{i=1}^{N} (y^{(i)} - \theta^T x^{(i)})$

MAP Estimation (Bayesian Est.)

$$\underbrace{P(\theta|D)}_{\text{Posterior}} \propto \underbrace{P(D|\theta)}_{\text{likelihood}} \underbrace{P(\theta)}_{\text{Prior}}$$

$$P(\theta) = \underset{\alpha,\beta}{\text{Beta}} = \frac{\theta^{\alpha-1}(1-\theta)^{\beta-1}}{B(\alpha,\beta)} \Rightarrow \begin{cases} \text{mean} = \frac{\alpha}{\alpha+\beta} \\ \text{mode} = \frac{\alpha-1}{\alpha+\beta-2} \end{cases}$$

$$\hat{\theta}_{MAP} = \arg\max_{\theta} P(\theta|D) \rightarrow \text{small } N$$

$$\hat{\theta}_{ML} = \arg\max_{\theta} P(D|\theta) \rightarrow \text{good for large } N$$

$$\begin{cases} N \to 0 \Rightarrow \hat{\theta}_{MAP} = \hat{\theta}_{Prior} \\ N \to \infty \Rightarrow \hat{\theta}_{MAP} = \hat{\theta}_{ML} \end{cases}$$

$\hat{\theta}_{Prior}$ → mode $P(\theta) = \frac{\alpha-1}{\alpha+\beta-2}$

for coin problem: $P(x) = \theta^x (1-\theta)^{1-x}$

$$\hat{\theta}_{MAP} = \frac{N}{N+\alpha+\beta-2}\left(\frac{\sum x_i}{N}\right) + \frac{\alpha+\beta-2}{N+\alpha+\beta-2}\left(\frac{\alpha-1}{\alpha+\beta-2}\right)$$

$\frac{\sum x_i}{N} = \hat{\theta}_{ML}$, $\frac{\alpha-1}{\alpha+\beta-2} = \hat{\theta}_{Prior}$

---

classification:
- Generative Modeling: $P(x|y), P(y)$
- Discriminative Modeling: $P(y|x)$

1) Generative:

$$\ell(\theta) = \sum_{i=1}^{N} \log P(x^{(i)}, y^{(i)}|\theta) =$$

$$= \sum_{i=1}^{N} \log P(y^{(i)}|\theta) + \sum_{i=1}^{N} \log P(x^{(i)}|y^{(i)},\theta)$$

$\Downarrow$ $\hat{y}$

$$\theta_j = \frac{\sum_{i=1}^{N} 1(\hat{y}=j)}{\sum_{j=1}^{k-1}\sum_{i=1}^{N} 1(\hat{y}=j)}$$

$\Downarrow$ Naive bayes with laplace Smoothing

$$\theta_{\bar{x}_k|j} = \frac{\left[\sum_{i=1}^{N} (\hat{y}=j \wedge \hat{x}_k = \bar{x}_k)\right] + t}{\left[\sum_{i=1}^{N} 1(\hat{y}=j)\right] + mt}$$

Naive Bayes: $P(x^{(i)}|y=j) = \prod_{k=1}^{k} P(\hat{x}_k|y=j)$

---

2) Discriminative Modeling: logistic regression ← Model

$$P(y=1|x) = \frac{1}{1+e^{-w^T\phi(x)}}$$

$$P(y=0|x) = \frac{1}{1+e^{w^T\phi(x)}}$$

$$\Rightarrow w = \arg\max_{w} \sum_{i=1}^{N} \log P(y^{(i)}|x^{(i)},w) \hookrightarrow \ell(w)$$

$$\Rightarrow J(w) = -\ell(w)$$

$$\Rightarrow \frac{\partial J}{\partial w} = \sum_{i=1}^{N} -\phi(x^{(i)})\left[y^{(i)} - \frac{1}{1+e^{-\phi(x)^T w}}\right]$$

$$\frac{\partial J}{\partial w} = \sum -\phi(x^{(i)})\left[y^{(i)} - P(y^{(i)}=1|x^{(i)},w)\right]$$

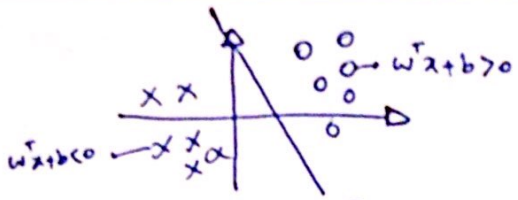
$\Rightarrow$ Gradient descent: $w^t = w^{t-1} - \rho\frac{\partial J}{\partial w}$

Neural Network update:

$$\frac{\partial J}{\partial w_{ij}^{(l)}} = \frac{\partial J}{\partial z_i^{(l+1)}} \times \frac{\partial z_i^{(l+1)}}{\partial (w_{ij}^{(l)})}$$

$$\frac{\partial J}{\partial b_i^{(l)}} = \frac{\partial J}{\partial z_i^{(l+1)}} \times \frac{\partial z_i^{(l+1)}}{\partial b_i^{(l)}}$$

SVM



$$g(w^T x+b)=\begin{cases}+1 & w^T x+b\geq 0\\ -1 & w^T x+b<0\end{cases}$$

$\hat{\gamma}^{(i)} = y^{(i)}(w^T x^{(i)}+b)$ → functional Margin

$\hat{\gamma} = \min_{i=1,\cdots,N} \hat{\gamma}^{(i)}$ → overal functional margin

$\gamma^{(i)} = \left(\frac{w^T x^{(i)}+b}{\|w\|_2}\right) y^{(i)}$ → geometric Margin

$\gamma = \min_{i=1,\cdots,N} \gamma^{(i)}$ → overall geometric Margin

fix $\hat{\gamma}=1$

Primal SVM:
$$\begin{cases}\min \|w\|_2^2\\ \text{s.t. } y^{(i)}(w^T x^{(i)}+b)\geq 1 \quad \forall i=1,2,\cdots,N\end{cases}$$

---

generally:
$$P^*:\begin{cases}\min f(w)\\ \text{s.t. } g_i(w)\leq 0 \quad i=1,2,\cdots,k\\ h_i(w)=0, \quad i=1,2,\cdots,l\end{cases}$$

$$\Rightarrow L(w,\alpha,\beta)\triangleq f(w)+\sum_{i=1}^{k}\alpha_i g_i(w)+\sum_{i=1}^{l}\beta_i h_i(w)$$

($\alpha_i \geq 0$)

$$\Rightarrow \max_{\beta,\alpha\geq 0} L(w,\alpha,\beta) = f(w) \quad \text{(if feasible: } g_i\leq 0,\ h_i=0\text{)}$$

$$\Rightarrow \min_w f(w) = \min_w \max_{\alpha\geq 0,\beta} L(w,\alpha,\beta)$$

$f_i(w)$, $g_i(w)$ Convex, $h_i(w)$ affine

$$\Rightarrow \max_{\alpha\geq 0,\beta}\min_w L = \min_w \max_{\alpha\geq 0,\beta} L(w,\alpha,\beta)$$

$$\Rightarrow \boxed{\min_w f(w) = \max_{\alpha\geq 0,\beta}\min_w L(w,\alpha,\beta)}$$

and
$$\frac{\partial L}{\partial w}=0 \qquad \frac{\partial L}{\partial \alpha}=0$$
$$\frac{\partial L}{\partial \beta}=0 \qquad \alpha_i g_i(w)=0$$
$$g_i(w)\leq 0 \qquad h_i(w)=0$$

---

back to SVM primal:

$$\Rightarrow \frac{\partial L}{\partial w}=0 \Rightarrow \boxed{w^* = \sum_{i=1}^{N}\alpha_i^* y^{(i)} x^{(i)}}$$

$$\Rightarrow \text{Dual SVM}: \max_{\alpha_1,\alpha_2,\cdots\alpha_N} -\frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_i\alpha_j\langle x^{(i)},x^{(j)}\rangle + \sum_{i=1}^{N}\alpha_i$$

$$\text{s.t } \begin{cases}\alpha_i\geq 0 \quad \forall i=1,\cdots,N\\ \sum_{i=1}^{N}\alpha_i y^i = 0\end{cases}$$

$$\boxed{b^* = -\frac{\max_{y^i=-1} w^{*T}x^j + \min_{y^i=1} w^{*T}x^j}{2}}$$

So for Test:

$$g(x) = \text{Sgn}(w^{*T}x+b^*) \quad \text{or} \quad K(x^{(i)},x)$$

$$\text{or } g(x) = \text{Sgn}\left(\sum_{i=1}^{N}\alpha_i^* y^i\langle x^i,x\rangle + b^*\right)$$

Kernels: $K(\text{Point1},\text{Point2}) = \text{number}$

$K(x,z) = \langle\varphi(x),\varphi(z)\rangle = \varphi^T(x)\varphi(z)$

* if $\varphi(x)=\begin{pmatrix}x_1^2\\ x_1x_2\\ \vdots\\ x_k^2\end{pmatrix}$ → $K(x,z)=(x^Tz)^2$

* if $\varphi(x)=\begin{pmatrix}x_1^n\\ x_1^{n-1}x_2\\ \vdots\\ x_k^n\end{pmatrix}$ → all monomials of degree n

$\Rightarrow K(x,z)=(x^Tz)^n$

* if $\varphi(x)=\begin{pmatrix}x_1^n\\ x_1^{n-1}x_2\\ \vdots\\ x_1\\ x_2\\ \vdots\\ x_k\\ 1\end{pmatrix}$ → all monomials of deg. n or less.

$K(x,z)=(x^Tz+c)^n$

* if $\varphi(x)=\infty$ dim $\Rightarrow K(x,z)=e^{-\|x-z\|_2^2/2\sigma^2}$

So (No need) to calculate $\varphi(x),\varphi(z)$

Kernel Matrix: for $D=\{(x^{(i)},y^{(i)})\}_{i=1}^N$

$$\begin{bmatrix}K_{11}&\cdots&K_{1N}\\ K_{N1}&\cdots&K_{NN}\end{bmatrix} \rightarrow K_{ij}=K(x^{(i)},x^{(j)})$$

K is kernel Matrix iff only if:
1) $K=K^T$
2) K is PSD: $z^TKz\geq 0 \ \forall z$

SVM - Soft Margin:

Primal: Soft Margin

$$\min_{(w,b)} \frac{1}{2}\|w\|_2^2 + C\sum_{i=1}^{N}\xi_i$$

$$\text{s.t. } y^{(i)}(w^T x^{(i)} + b) \geq 1 - \xi_i \quad \forall i = 1,\dots,N$$

$$\xi_i \geq 0, \quad \forall i = 1,\dots,N$$

Dual: Soft Margin

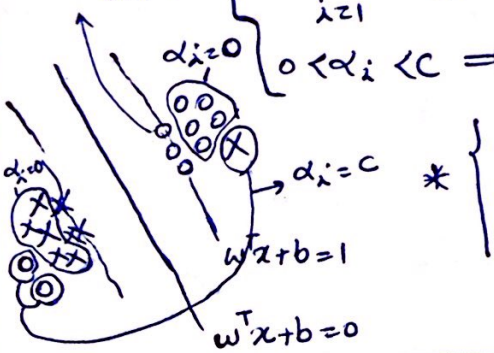
$$\max_{\alpha_1,\dots\alpha_N} \sum_{i=1}^{N}\alpha_i - \frac{1}{2}\sum y^i y^j k(x^i,x^j)\alpha_i\alpha_j$$

$$\text{s.t. } 0 \leq \alpha_i \leq C, \quad \forall i = 1,\dots,N$$

$$\sum_{i=1}^{N}\alpha_i y^i = 0$$

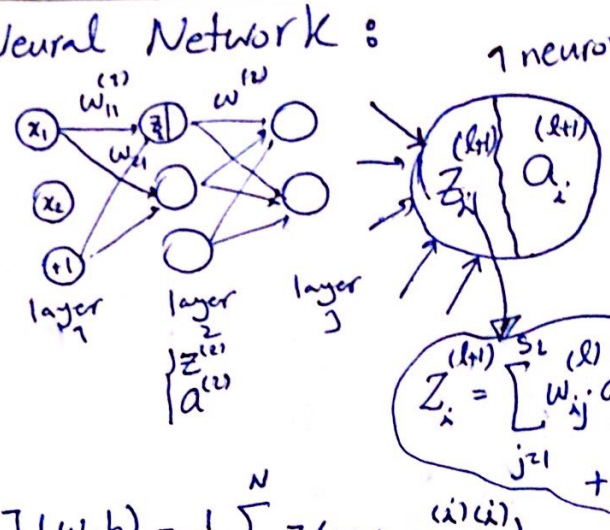
support vectors

$0 < \alpha_i < C \Rightarrow y^i(w^T x^i + b) = 1$



$*$ $\begin{cases} y^i(w^T x^i + b) > 1 \Rightarrow \alpha_i = 0 \\ y^i(w^T x^i + b) < 1 \Rightarrow \alpha_i = C \end{cases}$

Neural Network:



$$J_\lambda(W,b) = \frac{1}{N}\sum_{i=1}^{N} J(W,b,x^{(i)},y^{(i)}) + \frac{\lambda}{2}\sum_{l=1}^{n-1}\|W^{(l)}\|_F^2$$

where $J(W,b,x^{(i)},y^{(i)}) = \left\| h_{W,b}(x^{(i)}) - y^{(i)} \right\|_2^2$

$\Rightarrow$ Gradient Descent:

$$W^{(l)} \leftarrow W^{(l)} - \alpha \left.\frac{\partial J_\lambda}{\partial W}\right|_{W^{(l)}}$$

$$b^{(l)} \leftarrow b^{(l)} - \alpha \left.\frac{\partial J_\lambda}{\partial W}\right|_{b^{(l)}}$$

Forward Propagation: $\{W^{(l)}, b^{(l)}\}_{l=1}^{n-1}$ fixed

$$\begin{cases} z^{(l+1)} = W^{(l)} a^{(l)} + b^{(l)} \\ a^{(l+1)} = f(z^{(l+1)}) \end{cases} \quad \forall l = 1,2,\dots,n-1$$

Back Propagation: $\{z^{(l)}, a^{(l)}\}_{l=1}^{n}$ computed & fixed

$$\delta^{(n)} = (a^{(n)} - y) \odot f'(z^{(n)})$$

For $l = n-1, n-2, \dots, 2, 1$

① Set: $\delta^{(l)} = (W^{(l)T}\delta^{(l+1)}) \odot f'(z^{(l+1)})$

② $\left.\frac{\partial J}{\partial W}\right|_{W^{(l)}} = \delta^{(l+1)} a^{(l)T}$

③ $\left.\frac{\partial J}{\partial b}\right|_{b^{(l)}} = \delta^{(l+1)}$

Update W, b:

$$W^{(l)} \leftarrow W^{(l)} - \alpha\left[\frac{1}{N}\left.\frac{\partial J}{\partial W}\right|_{W^{(l)}} + \lambda W^{(l)}\right]$$

$$b^{(l)} \leftarrow b^{(l)} - \alpha\left[\frac{1}{N}\left.\frac{\partial J}{\partial b}\right|_{b^{(l)}}\right]$$

Ensemble Method:

$x \to g(x)$ Causes overfitting
↳ ensemble Methods are greate!

* use weak classifier (h)

Boosting:

(1) For $t = 1,\dots,T$:
(2) if $t = 1$ then:
(3) Initialize weights $D_{t,i} = \frac{1}{N}$
(4) else
(5) set weights to $D_{t,i} \propto D_{t-1,i}\, e^{-\alpha_{t-1} y^{(i)} h_{t-1}(x^{(i)})}$
(6) End if
(7) Train a hypothesis h_t using D_t
(8) Evaluate Training Error $\varepsilon_t = \sum_{i=1}^{N} D_{t,i}\, 1(h_t(x^{(i)}) \neq y^{(i)})$
(9) Set $\alpha_t = \frac{1}{2}\ln\frac{1-\varepsilon_t}{\varepsilon_t}$
(10) End for
(11) $H(x) = \text{sign}\left(\sum_{t=1}^{T}\alpha_t h_t(x)\right) \Rightarrow$ final classifier

Bagging: ① No weight, random sampling with replacement
② uniform weight for hypothesis: $\alpha_t = \frac{1}{T}$

in Boosting: (1) $\varepsilon_t \to 0 \Rightarrow \alpha_t \to \infty$, $\varepsilon_t = 0.5 \Rightarrow \alpha_t \to 0$

(2) if $\begin{cases} h_{t-1}(x^{(i)})\, y^{(i)} > 0 \xrightarrow{\text{correct}} D_{t-1,i} \times \text{small weight} \\ h_{t-1}(x^{(i)})\, y^{(i)} < 0 \xrightarrow{\text{incorrect}} D_{t-1,i} \times \text{large } w \end{cases}$

Principal Component Analysis (PCA)

(1) input : $D = \{x^{(i)}\}_{i=1}^{N} = [X]_{n \times N}$

(2) Output : $[U]_{n \times d}$, $[\mu]_{n \times 1}$, $[y]_{d \times N}$

(3) Compute $[\bar{x}] = \frac{1}{N} \sum_{i=1}^{N} x^{(i)}$

(4) Form : $X = [x^{(1)} - \bar{x}, x^{(2)} - \bar{x}, \dots x^{(N)} - \bar{x}]$

(5) Compute SVD of X

$$X = U_x \Sigma_x V_x^T \quad U_x \in \mathbb{R}^{n \times n}$$

(6) Optimal $U_{n \times d} = U_x(:, 1:d)$

(7) Principal Components : $[y] = U^T (x - \bar{x})$ ← original Data

PCA optz :
$$\min \frac{1}{2} \sum_{i=1}^{N} \| x^{(i)} - U y^{(i)} - \mu \|_2^2$$
$$\text{s.t.} \sum_{i=1}^{N} y^{(i)} = 0 \,, \; UU^T = I_d$$

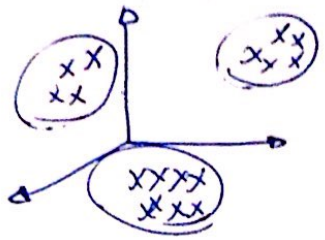
⇒ Then we write $L(\dots)$

Then $\frac{\partial L}{\partial \mu} = 0 \Rightarrow \mu = \frac{1}{N} \sum_{i=1}^{N} x^{(i)}$

$\frac{\partial L}{\partial y_i} = 0 \Rightarrow y^{(i)} = U^T (x^{(i)} - \bar{x})$

Plug in $y^{(i)}$ & μ , ... optz ⇒ ✓✓

Clustering



Cost function :

$$J = \sum_{i=1}^{N} \min_{j=1,2,\dots,k} \| x^{(i)} - c_j \|_2^2$$

$$\Rightarrow \text{optz:} \begin{cases} \min \sum_{i=1}^{N} \sum_{j=1}^{K} z_{ij} \| x^{(i)} - c_j \|_2^2 \\ \text{s.t. } z_{ij} \in \{0,1\}, \forall ij \,; \sum_{j=1}^{k} z_{ij} = 1 \quad \forall i \end{cases}$$

where: $z_{ij} = \begin{cases} 1 & \text{if } x^{(i)} \text{ is assigned to } c_j \\ 0 & \text{else} \end{cases}$

Solve Optz. with Coordinate Descent:

KMEANS

(1) Initialize $\{c_j^{(0)}\}_{j=1}^{K}$ and $\{z_{ij}\}_{i=1,\dots N}^{j=1,\dots K}$

(2) For $t = 1, 2, \dots$ do :

▸ For every i

$$z_{ij}^{(t)} = \begin{cases} 1 & \text{if } j = \arg\min_{l=1,\dots,k} \| x^{(i)} - c_l^{(t-1)} \|_2^2 \\ 0 \end{cases}$$

▸ For every j

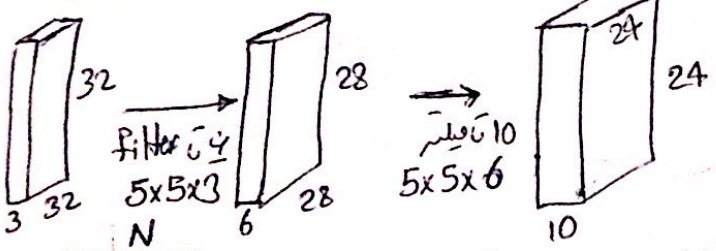
$$c_j^{(t)} = \frac{\sum_{i=1}^{N} 1(z_{ij}^{(t)} = 1)\, x^{(i)}}{\sum_{i=1}^{N} 1(z_{ij}^{(t)} = 1)}$$

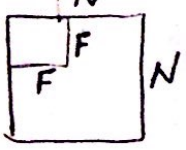
(3) Output: $\{c_j^{(t)}\}, \{z_{ij}^{(t)}\} \quad \forall i,j$

K-means Converges (Cost function always decreasing.)

$$J(\{z_{ij}^{(t)}\}, \{c_j^{(t)}\}) \geq J(\{z_{ij}^{(t+1)}\}, \{c_j^{(t+1)}\})$$

Convolutional Neural Network





▲ output size = (N-F)/stride+1

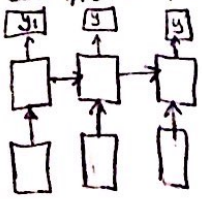
* for every 0 padding add 2 to N

→ input 32x32x3

10 filter 5x5x3 ⇒ #parameters = (5x5x3+1)x10 = 760

(+1 ↓ bias)

reccurrent N.N



$$h_t = f_W(h_{t-1}, x_t)$$

ex:

$$h_t = \tanh(W_{hh} h_{t-1} + W_{hx} x_t)$$

$$y_t = W_{hy} h_t$$

HW3

① $P(A|B) = \frac{P(A \cap B)}{P(B)}$ $\quad P(A \cap B)$

② $P(A|B)\,P(B) = P(B|A)\,P(A) = P(A \cap B)$

③ $P(A \cup B) = P(A) + P(B) - P(A \cap B)$

④ A, B indepe $\Rightarrow$ $P(A \cap B) = P(A)P(B)$

Distributions

① Gaussian: $f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2}$

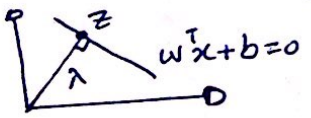
② Bernoulli: $0 < p < 1$ $\quad f(x;p) = p^x (1-p)^{1-x}$, $x \in \{0,1\}$

③ Uniform $\quad f(x;a,b) = \begin{cases} \frac{1}{b-a} & x \in [a,b] \\ 0 & \text{otherwise} \end{cases}$

④ Exponential Dist: $\quad f(x,\lambda) = \begin{cases} \lambda e^{-\lambda x} & x \geq 0, \lambda > 0 \\ 0 & x < 0 \end{cases}$

⑤ Poisson Dist: $\quad f(x,\lambda) = \frac{\lambda^x e^{-\lambda}}{x!}$ $\quad \lambda > 0$

Distance from origin to hyperplane:



$r = \frac{|b|}{\|w\|_2}$

$\vec{z} = r \frac{w}{\|w\|_2}$

$k(x,z) = c_1 k_1(x,z) + c_2 k_2(x,z)$ $\Leftarrow$ ج

$k_1(x,z) = \phi_1^T(x)\,\phi_1(z)$

$k_2(x,z) = \phi_2^T(x)\,\phi_2(z)$

$\Rightarrow k(x,z) = c_1 \phi_1^T(x)\,\phi_1(z) + c_2 \phi_2^T(x)\,\phi_2(z)$

$k(x,z) = \phi^T(x)\,\phi(z)$

where $\phi = \begin{bmatrix} \sqrt{c_1}\,\phi_1(x) \\ \sqrt{c_2}\,\phi_2(x) \end{bmatrix}$ ✓✓

$k(x,z) = k_1(x,z)\,k_2(x,z)$ $\Leftarrow$ ضرب

$k_1(x,z) \rightarrow \phi_1(x) = [f_1(x), f_2(x), \dots]$

$k_2(x,z) \rightarrow \phi_2(x) = [g_1(x), g_2(x), \dots]$

$k(x,z) = k_1(x,z)\,k_2(x,z) =$

$= \phi_1^T(x)\,\phi_1(z)\;\phi_2^T(x)\,\phi_2(z)$

$= \sum_{i=1}^{\infty} f_i(x) f_i(z) \sum_{j=1}^{\infty} g_j(x) g_j(z) =$

$= \sum_{i,j}^{\infty} (f_i(x)\,g_j(x))(f_i(z)\,g_j(z))$

$= \sum_{i,j} c_{i,j}(x)\,c_{i,j}(z)$

✓✓

Positive Semi definit (A_{nn})

if $z^T A z \geq 0$ for $\forall z$

$\Rightarrow A = B^T B$

$P(x_1, \dots, x_N) = \sum_{c_j} \prod_i P(x_i|c_j)\,P(c_j)$

Naive base
