(1)

a) y_i is distributed as $N(\mu, \sigma^2)$, where $\mu = \log(wx_i)$ and $\sigma^2 = 1$

$$p(y_i; \mu, \sigma^2) = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{(y_i - \log(wx_i))^2}{2}\right)$$

The full data likelihood is written by (assuming i.i.d):

$$L = \prod_{i=1}^{N} p(y_i; \mu, \sigma^2) = \prod_{i=1}^{N} \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{(y_i - \log(wx_i))^2}{2}\right)$$

$$= (2\pi)^{-N/2} \prod_{i=1}^{N} \exp\left(-\frac{(y_i - \log(wx_i))^2}{2}\right)$$

Then, log-likelihood is:

$$LL = \log\left[(2\pi)^{-N/2} \prod_{i=1}^{N} \exp\left(\frac{-(y_i - \log(wx_i))^2}{2}\right)\right]$$

$$= -N/2 \log 2\pi - \frac{1}{2} \sum_{i=1}^{N} (y_i - \log(wx_i))^2$$

To infer the unknown parameter w, we maximize the log-likelihood

which means we maximize the probability/likelihood that the data is generated by the distribution.

b) To maximize the likelihood, we maximize log-likelihood as it is computationally easier to work with:
For this, we take the derivative of LL w.r.t. w & set it to zero:

$$\frac{\partial LL}{\partial w} = 0 \Rightarrow \frac{\partial \sum_{i=1}^{N} (y_i - \log(wx_i))^2}{\partial w} = 0$$

$$\Rightarrow \sum_{i=1}^{N} \frac{x_i}{w x_i} (y_i - \log(w x_i)) = 0$$

$$\Rightarrow \frac{1}{w} \sum_{i=1}^{N} (y_i - \log(w x_i)) = 0$$

$$\Rightarrow \underline{\sum_{i=1}^{N} y_i = \sum_{i=1}^{N} \log(w x_i)} \quad \underline{= \sum_{i=1}^{N} \log(x_i) + N \log w}$$

c) for scalar w we have: $\|w\|_2^2 = w^2$

The new log-likelihood is:

$$LL^{new} = -\frac{N}{2}\log(2\pi) - \frac{1}{2}\sum_{i=1}^{N} (y_i - \log(w x_i))^2 - \alpha w^2$$

Thus the derivative is:

$$\frac{\partial LL^{new}}{\partial w} = \frac{-1}{2} \frac{\partial \sum_{i=1}^{N} (y_i - \log(w x_i))^2}{\partial w} - \frac{\alpha \partial w^2}{\partial w}$$

$$= -\frac{1}{2} \sum_{i=1}^{N} \frac{-2}{w} (y_i - \log(w x_i)) - 2 w \alpha$$

$$= \frac{+1}{w} \sum_{i=1}^{N} (y_i - \log(w x_i)) - 2 w \alpha$$

$$\frac{\partial LL^{new}}{\partial w} = 0 \Rightarrow \frac{+1}{w} \sum_{i=1}^{N} (y_i - \log(w x_i)) - 2 w \alpha = 0$$

$$\Rightarrow 2 \alpha w^2 = \sum_{i=1}^{N} (y_i - \log(w x_i)) \Rightarrow$$

$$\Rightarrow \sum_{i=1}^{N} y_i = \sum_{i=1}^{N} \log(w x_i) + 2 \alpha w^2$$

d) if $\alpha \to \infty \Rightarrow -\alpha \|w\|_2^2 \to -\infty$

Since we want to maximize the likelihood then $\|w\|_2^2$ should become small, i.e. $w \to 0$ as $\alpha \to \infty$ in order to maximize the likelihood.

Thus this regularizer shift w towards zero!

---

(2)

a) We first introduce some notations:

$\hat{y}$: output of neural network

z_0 : input to the activation of the output node

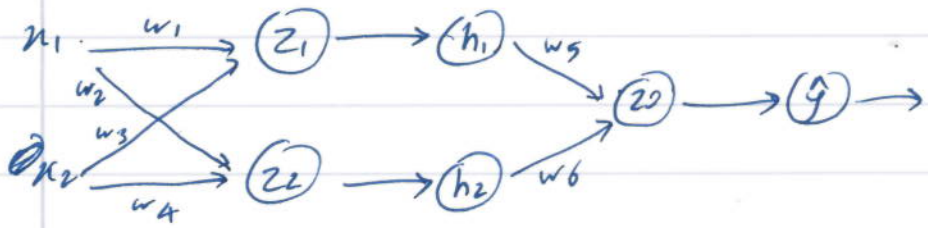
h_1 : the output of the first hidden node

h_2 : the output of the second hidden node

z_1 : input to the activation of first hidden node

z_2 : input to the activation of second hidden node

The network then can be seen as:



Thus:

$$\frac{\partial \Sigma}{\partial w_5} = \frac{\partial \Sigma}{\partial \hat{y}} \frac{\partial \hat{y}}{\partial w_5} = \frac{\partial \Sigma}{\partial \hat{y}} \frac{\partial \hat{y}}{\partial z_0} \frac{\partial z_0}{\partial w_5}$$

$$\frac{\partial \Sigma}{\partial w_6} = \frac{\partial \Sigma}{\partial \hat{y}} \frac{\partial \hat{y}}{\partial w_6} = \frac{\partial \Sigma}{\partial \hat{y}} \frac{\partial \hat{y}}{\partial z_0} \frac{\partial z_0}{\partial w_6}$$

$$\frac{\partial \Sigma}{\partial w_1} = \frac{\partial \Sigma}{\partial \hat{y}} \frac{\partial \hat{y}}{\partial w_1} = \frac{\partial \Sigma}{\partial \hat{y}} \frac{\partial \hat{y}}{\partial z_0} \frac{\partial z_0}{\partial w_1} = \frac{\partial \Sigma}{\partial \hat{y}} \frac{\partial \hat{y}}{\partial z_0} \frac{\partial z_0}{\partial h_1} \frac{\partial h_1}{\partial w_1}$$

$$= \frac{\partial \Sigma}{\partial \hat{y}} \frac{\partial \hat{y}}{\partial z_0} \frac{\partial z_0}{\partial h_1} \frac{\partial h_1}{\partial z_1} \frac{\partial z_1}{\partial w_1}$$

$$\frac{\partial \mathcal{E}}{\partial w_2} = \frac{\partial \mathcal{E}}{\partial \hat{y}} \frac{\partial \hat{y}}{\partial w_2} = \frac{\partial w}{\partial \hat{y}} \frac{\partial \hat{y}}{\partial z_0} \frac{\partial z_0}{\partial w_2} = \frac{\partial \mathcal{E}}{\partial \hat{y}} \frac{\partial \hat{y}}{\partial z_0} \frac{\partial z_0}{\partial h_2} \frac{\partial h_2}{\partial w_2}$$

$$= \frac{\partial \mathcal{E}}{\partial \hat{y}} \frac{\partial \hat{y}}{\partial z_0} \frac{\partial z_0}{\partial h_2} \frac{\partial h_2}{\partial z_2} \frac{\partial z_2}{\partial w_2}$$

similarly, you should obtain $\frac{\partial \mathcal{E}}{w_3}$, $\frac{\partial \mathcal{E}}{w_4}$

b)

1. initialize the parameters and set a learning rate γ.
2. repeat:

$$w_j^{(t+1)} = w_j^{(t)} - \gamma \frac{\partial \mathcal{E}^{(t)}}{\partial w_j^{(t)}}, \quad j = 1 \dots 6$$

until convergence.

↳ the error does not change or the parameters do not change anymore.

$\Longrightarrow$ yields a local optima.

c) the activation function can be written as:

$$f(z) = (2z)^q$$

if $q = 1 \Longrightarrow f(z) = 2z \Longrightarrow$ it is just a simple (linear) transformation.

thus:

$$\left.\begin{aligned} \hat{y} = f(z_0) &= 2z_0 = 2(w_5 h_1 + w_6 h_2) \\ h_1 &= 2z_1 = 2(w_1 x_1 + w_3 x_2) \\ h_2 &= 2z_2 = 2(w_2 x_1 + w_4 x_2) \end{aligned}\right\}$$

$$\Longrightarrow \hat{y} = 2\left(w_5 (2(w_1 x_1 + w_3 x_2)) + w_6 (2(w_2 x_1 + w_4 x_2))\right)$$

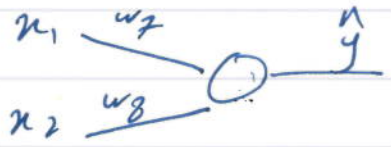
$$= 4 w_1 w_5 x_1 + 4 w_3 w_5 x_2 + 4 w_2 w_6 x_1 + 4 w_4 w_6 x_2$$

$$= (4 w_1 w_5 + 4 w_2 w_6) x_1 + (4 w_3 w_5 + 4 w_4 w_6) x_2$$

$\Longrightarrow \hat{y} = w_7 x_1 + w_8 x_2$

where $\begin{cases} w_7 = 4 w_1 w_5 + 4 w_2 w_6 \\ w_8 = 4 w_3 w_5 + 4 w_4 w_6 \end{cases}$

Thus, we can replace the network with a simpler network that just computes the weighted sum of the input attributes:



and we do not need any activation function $f(z)$

d) yes: as we saw in (c), for $q=1$ we have $\hat{y} = w_7 x_1 + w_8 x_2$

i.e.: $\hat{y} = w^T x$, where $w^T = [w_7, w_8]$

This is just the formulation of ~~a straight~~ a basic linear regression model.

---

(3)

a) The training error would be related to misclassifican error, i.e., number of training items that are wrongly classified.

In this data set, for optimal solution, no item would be misclassified, thus ~~training~~ training error is zero.

b) The support vectors corresponding to γ's:
see 5.3.2.4 in the text book page(189-190) for details:
.... at the optimum, all of the α_n that do not correspond to support vectors will be zero, ...

in optimum: $t_{new} = \text{sign}\left(\sum_{n=1}^{N} \alpha_n t_n x_n^T x_{new} + b\right)$, $b = t_n - \sum_{m=1}^{N} \alpha_m t_m x_m^T x_n$

Thus zero α_n's will not have an impact on t_{new}, b, and they can be discarded. Therefore we can train the model using only the support vectors.

c) $O(d)$

see 5.3.2.3 : Making predictions.

we have:

$$t_{new} = \text{sign}\left(\underbrace{\sum_{n=1}^{N} \alpha_n t_n x_n^T}_{w^*} x_{new} + b\right), \quad b = t_n - \sum_{m=1}^{N} \alpha_m t_m x_m^T x_n$$

w^* and b: $O(N*d) \rightarrow b$

computing w^*, b in general takes $O(dN)$, but they can be computed independent of test (prediction). In other words, they can be obtained immediately after training and do NOT depend on particular test data point $\Rightarrow$ we do not take them into account for test.

Note that if number of support vectors $\ll N$

$\Rightarrow$ complexity of w^*, b would be $O(d)$ and $O(d)$ (instead of $O(Nd)$)

$\Rightarrow$ for test : we need compute $w^* x_{new}$ $\rightarrow O(d)$

and $\text{sign}(w^* x_{new} + b)$

$\Rightarrow$ test complexity $= O(d)$

(4)

a) K-means cost function is written by:

$$R(\mu, Z; X) = \sum_{n=1}^{N} \sum_{k=1}^{K} z_{nk} \|x_n - \mu_k\|_2^2 \quad , z_{nk} \in \{0,1\}, \sum_k z_{nk} = 1$$

if $k=1 \Rightarrow z_{nk}$ ... is always 1:

$$R(\mu, Z; X) = \sum_{n=1}^{N} z_{nk} \|x_n - \mu_k\|_2^2 = [illegible]$$

$$= \sum_{n=1}^{N} z_{nk} (x_n - \mu_k)^T (x_n - \mu_k)$$

There is only only one mean: $\Rightarrow$

$$R(\mu, Z; X) = \sum_{n=1}^{N} (x_n - \mu)^T (x_n - \mu)$$

$$\frac{\partial R}{\partial \mu} = \frac{\partial \sum_{n=1}^{N} (x_n - \mu)^T (x_n - \mu)}{\partial \mu} = 2 \sum_{n=1}^{N} x_n - \mu$$

$$\frac{\partial R}{\partial \mu} = 0 \Rightarrow \sum_{n=1}^{N} (x_n - \mu) = 0 \Rightarrow \mu = \frac{1}{N} \sum_{n=1}^{N} x_n$$

b) We know $R(\mu, Z; X) \geq 0$

Then if we show that for a solution $R(\mu, Z; X) = 0$ then this solution will be the optimal solution, because it yields the minimum possible cost.

if $\mu_k = x_k \quad \forall 1 \le k \le N$

$$\Rightarrow R(\mu, Z; X) = \sum_{i=1}^{N} z_{nk} \| x_n - x_n \|_2^2 = 0$$

Thus the solution in which each mean is exactly one of the datapoints yield's $R=0$ which is an optimal solution.

c) for $k^{min} \le K \le k^{max}$

compute AIC_k

compute BIC_k

choose K^*_{AIC} and K^*_{BIC} for which AIC and BIC numbers are minimal, i.e.:

$$K^*_{AIC} = \arg\min_k AIC_k$$

$$K^*_{BIC} = \arg\min_k BIC_k$$

AIC_k is obtained by: $-ll$ + $c(U, Z)$

($-ll$: negative log-likelihood; $c(U,Z)$: complexity)

BIC_k is obtained by: $-ll + \frac{1}{2} c(U, Z) \ln N$

$c(U, Z) = K * d$ + (→ 0 (N/k))

($K*d$: related to means)

(related to π_k's (if we know the size of one cluster (π_1) the size of other cluster is the same.)

*all cluster have same size