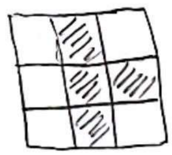
Re-exam 2: FFR135

[redacted]

1) Convolutional net

Choose the following filter:



Legend:

□ corresponds to $w = 0$

▨ corresponds to $w = 1$

- Pattern $x^{(1)}$ can be represented as:

$$\begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 \end{bmatrix}$$

The local fields of the feature map of pattern $x^{(1)}$ can be determined as follows (assume the threshold θ is zero):

Consider the top left 3×3 block of the inputs

$$\underbrace{\begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}}_{\text{3x3 block of inputs}} \underset{\text{multiply}}{\times} \underbrace{\begin{bmatrix} 0 & 1 & 0 \\ 0 & 1 & 1 \\ 0 & 1 & 0 \end{bmatrix}}_{\text{kernel}} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix} \xrightarrow{} \underset{\text{sum everything}}{2}$$

[redacted]

Sweep the kernel over the 2-D input matrix with stride [1,1], padding 0.
It follows that the local fields of the feature map of pattern $x^{(1)}$ are:

$$\begin{bmatrix} 2 & 2 \\ 1 & 2 \\ 1 & 1 \\ 1 & 2 \\ 2 & 2 \\ 1 & 2 \end{bmatrix}$$

- Now, pattern $x^{(2)}$ can be represented as follows:

$$\begin{bmatrix} 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 \end{bmatrix}$$

Similarly, the local fields of the feature map of pattern $x^{(2)}$ can be determined as follows:

$$\underbrace{\begin{bmatrix} 0 & 0 & 0 \\ 1 & 1 & 1 \\ 1 & 0 & 0 \end{bmatrix}}_{\text{3x3 block of inputs}} \underset{\text{multiply}}{\times} \underbrace{\begin{bmatrix} 0 & 1 & 0 \\ 0 & 1 & 1 \\ 0 & 1 & 0 \end{bmatrix}}_{\text{kernel}} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 1 \\ 0 & 0 & 0 \end{bmatrix} \longrightarrow \underset{\text{sum everything}}{2}$$

[redacted]

Sweep the kernel over the 2-D input matrix with stride [1, 1], and padding [0, 0, 0, 0]. It follows that the local fields of the feature map of pattern $x^{(2)}$ are:

$$\begin{bmatrix} 2 & 1 \\ 1 & 1 \\ 1 & 0 \\ 1 & 0 \\ 1 & 0 \\ 1 & 1 \end{bmatrix}$$

The ReLU - activation function does not exert any effect, since all local fields are possible. Thus, the feature maps are therefore equal to the above local fields.

- Apply max-pooling operation on the resulting feature map of pattern $x^{(1)}$.

$$\begin{bmatrix} 2 & 2 \\ 1 & 2 \\ 1 & 1 \\ 1 & 2 \\ 2 & 2 \\ 1 & 2 \end{bmatrix}$$

For 2×2 max pooling, the maximum element within the top 2×2 block is 2

[redacted]

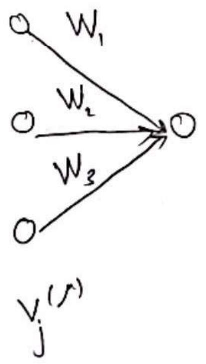
Sweep over the resulting feature map with stride 2 and padding 0
The output of the max-pooling layer is:

$$\begin{bmatrix} 2 \\ 2 \\ 2 \end{bmatrix}$$

- Similarly for pattern $x^{(2)}$, the output of the max-pooling layer is

$$\begin{bmatrix} 2 \\ 1 \\ 1 \end{bmatrix}$$

- To determine the weights W_k and threshold Θ, the layout of the output of the network can be represented as follows:



$$O^{(\mu)} = \Theta_H\left(\sum_{j=1}^{3} W_j V_j^{(\mu)} - \Theta\right)$$

[redacted]

where

$V^{(1)} = \begin{bmatrix} 2 \\ 2 \\ 2 \end{bmatrix}$ for pattern $x^{(1)}$

$V^{(2)} = \begin{bmatrix} 2 \\ 1 \\ 1 \end{bmatrix}$ for pattern $x^{(2)}$

To classify the 2 patterns we need the output of the network to be

$O^{(1)} = 0$ for $x^{(1)}$

$O^{(2)} = 1$ for $x^{(2)}$

This can be done by setting the output of the network to be

$$O^{(\mu)} = \theta_H\left(-V_1^{(\mu)} - V_2^{(\mu)} - V_3^{(\mu)} + 5\right)$$

where $W = \begin{bmatrix} -1 \\ -1 \\ -1 \end{bmatrix}$ and $\Theta = -5$

In this case we get

For $\mu = 1$: $O^{(1)} = \theta_H\left(-2-2-2+5\right) = \theta_H(-1) = 0$

For $\mu = 2$: $O^{(2)} = \theta_H(-2-1-1+5) = \theta_H(+1) = 1$ Verified

[redacted]

2) Kullback - Leibler divergence

a) $$D_{KL} = \sum_{\mu=1}^{p} P_{data}(x^{(\mu)}) \log \left[\frac{P_{data}(x^{(\mu)})}{P_B(s = x^{(\mu)})} \right]$$

$$= \sum_{\mu=1}^{p} P_{data}(x^{(\mu)}) \left(- \log \left[\frac{P_B(s = x^{(\mu)})}{P_{data}(x^{(\mu)})} \right] \right)$$

Since $\ln x \leqslant x - 1 \quad \forall\ x > 0$, it follows that

$$D_{KL} \geqslant \sum_{\mu=1}^{p} P_{data}(x^{(\mu)}) \left(- \left[\frac{P_B(s = x^{(\mu)})}{P_{data}(x^{(\mu)})} \right] + 1 \right)$$

$$\geqslant \sum_{\mu=1}^{p} P_{data}(x^{(\mu)}) \left[\frac{- P_B(s = x^{(\mu)}) + P_{data}(x^{(\mu)})}{P_{data}(x^{(\mu)})} \right]$$

$$\geqslant \sum_{\mu=1}^{p} \left[P_{data}(x^{(\mu)}) - P_B(s = x^{(\mu)}) \right]$$

$$\geqslant \sum_{\mu=1}^{p} P_{data}(x^{(\mu)}) - \sum_{\mu=1}^{p} P_B(s = x^{(\mu)})$$

The total probability is normalized to 1

$$\geqslant 1 - 1$$

$$\geqslant 0$$

[redacted]

Thus the Kullback - Leibler divergence is non-negative.

b) For $P_B(s=x^{(\mu)}) = P_{data}(x^{(\mu)})$:

$$D_{KL} = \sum_{\mu=1}^{p} P_{data}(x^{(\mu)}) \log \left[\frac{P_{data}(x^{(\mu)})}{P_B(s=x^{(\mu)})} \right]$$

$$= \sum_{\mu=1}^{p} P_{data}(x^{(\mu)}) \log (1)$$

$$= 0 \qquad \text{verified}$$

c) Show that minimizing D_{KL} is equivalent to maximizing log-likelihood function:

$$D_{KL} = \sum_{\mu=1}^{p} P_{data}(x^{(\mu)}) \log \left[\frac{P_{data}(x^{(\mu)})}{P_B(s=x^{(\mu)})} \right]$$

$$= \sum_{\mu=1}^{p} P_{data}(x^{(\mu)}) \log P_{data}(x^{(\mu)}) - \sum_{\mu=1}^{p} P_{data}(x^{(\mu)}) \log P_B(s=x^{(\mu)})$$

$$= \underbrace{\sum_{\mu=1}^{p} P_{data}(x^{(\mu)}) \log P_{data}(x^{(\mu)})}_{\text{Fixed by the given data distribution}} - \langle \log P_B(s=x^{(\mu)}) \rangle_{P_{data}}$$

[redacted]

Now,

$$\text{Log}\,\mathcal{L} = \log \prod_{\mu=1}^{p} P_B(s=x^{(\mu)}) = \sum_{\mu=1}^{p} \log P_B(s=x^{(\mu)})$$

using the law of large numbers $\frac{1}{N}\sum_{i=1}^{N} f(i) \overset{N\gg1}{\approx} \underbrace{\int dx\, P(x)\, f(i)}_{\langle f(i)\rangle_{P(x)}}$

Thus, Log $\mathcal{L}$ can be written as:

$$\begin{aligned}
\text{Log}\,\mathcal{L} &= \sum_{\mu=1}^{p} \log P_B(s=x^{(\mu)}) \\
&= p\,\frac{1}{p}\sum_{\mu=1}^{p} \log P_B(s=x^{(\mu)}) \\
&\overset{p\gg1}{\approx} p\,\langle \log P_B(s=x^{(\mu)})\rangle_{P_{data}}
\end{aligned}$$

It follows that:

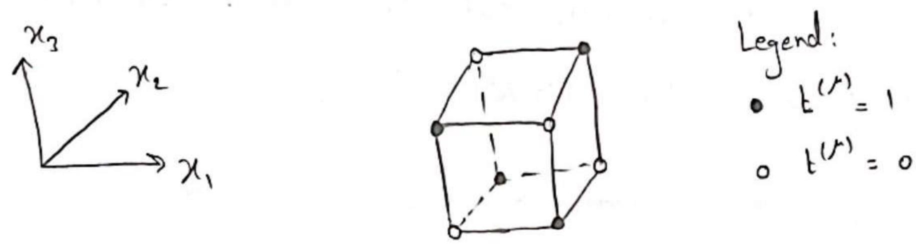
$$\text{Log}\,\mathcal{L} \overset{p\gg1}{\approx} p\sum_{\mu=1}^{p} P_{data}(x^{(\mu)}) \log P_{data}(x^{(\mu)}) - p\,D_{KL}$$

Thus, minimizing D_{KL} is equivalent to maximizing Log $\mathcal{L}$.

[redacted]

3) Boolean function I:

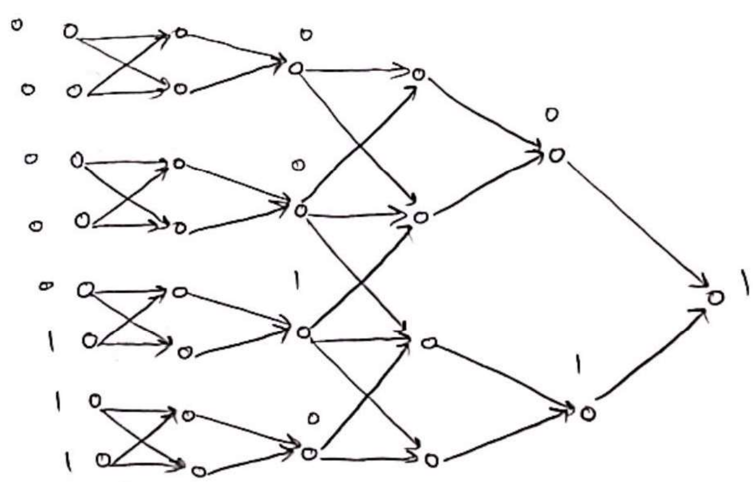
a) Graphical representation in the input space:



As can be seen from the above representation, there is no decision boundary (i.e. plane) that can seperate patterns that map to $t^{(\mu)}=1$ from those that map to $t^{(\mu)}=0$.

Thus, there is no plane that can seperate the two classes of outputs. Hence, the problem is not linearly seperable.

b)



[redacted]

The network is built from XOR units.

Each XOR unit has a hidden layer with 2 neurons.

On the figure drawn, only the states of the inputs and outputs of the XOR units are shown, not those of hidden neurons.

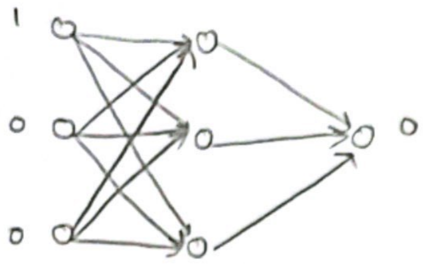
In total the network has $O(N)$ neurons.

For $N = 2^k$ input units with $k = 1, 2, 3 \ldots$ the whole network has $3(N-1)$ neurons.

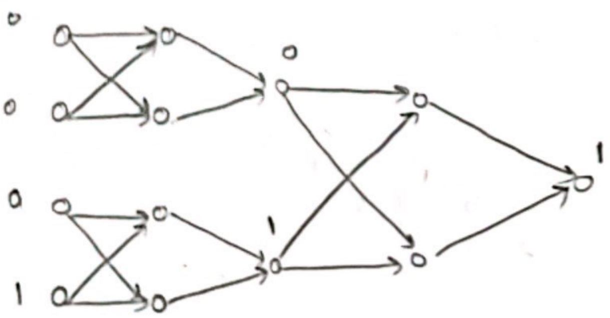
[redacted]

4) Boolean Function II:

For N = 3:



For N = 4:



The network is built from XOR units.

Each XOR unit has one hidden layer with 2 neurons.

Only the states of the inputs and outputs of XOR units are shown on the above figure.

For $N = 4$, the network has $3(N-1) = 3(4-1) = 9$ neurons.

[redacted]

5) Backpropagation:

a) $$\delta w^{(l)} = -\eta \frac{\partial H}{\partial w^{(l)}}$$

$$= -\eta \, 2 \cdot \tfrac{1}{2} \left(t - V^{(L)}\right) \left(- \frac{\partial V^{(L)}}{\partial w^{(l)}} \right)$$

$$= \eta \left(t - V^{(L)}\right) \left(\frac{\partial V^{(L)}}{\partial w^{(l)}} \right)$$

$$= \eta \left(t - V^{(L)}\right) \frac{\partial V^{(L)}}{\partial V^{(l)}} \frac{\partial V^{(l)}}{\partial w^{(l)}}$$

where $V^{(l)} = g\left(w^{(l)} V^{(l-1)} - \theta^{(l)}\right) = g\left(b^{(l)}\right)$

$$\frac{\partial V^{(l)}}{\partial w^{(l)}} = g'\left(b^{(l)}\right) V^{(l-1)}$$

It follows that:

$$\delta w^{(l)} = \eta \left(t - V^{(L)}\right) \frac{\partial V^{(L)}}{\partial V^{(l)}} g'\left(b^{(l)}\right) V^{(l-1)}$$

$$= \eta \, \delta^{(l)} V^{(l-1)}$$

such that $\delta^{(l)} = \left(t - V^{(L)}\right) \frac{\partial V^{(L)}}{\partial V^{(l)}} g'\left(b^{(l)}\right)$

in otherwords $\delta^{(l-1)} = \left(t - V^{(L)}\right) \frac{\partial V^{(L)}}{\partial V^{(l-1)}} g'\left(b^{(l-1)}\right)$ verified

[redacted]

b) Evaluate $\dfrac{\partial V^{(L)}}{\partial V^{(\ell-1)}}$:

We have: $V^{(L)} = g\left(w^{(L)} V^{(L-1)} - \theta^{(L)}\right) = g\left(b^{(L)}\right)$

- $$\frac{\partial V^{(L)}}{\partial V^{(L-1)}} = g'\left(b^{(L)}\right) w^{(L)}$$

- $$\frac{\partial V^{(L)}}{\partial V^{(L-2)}} = g'\left(b^{(L)}\right) w^{(L)} \frac{\partial V^{(L-1)}}{\partial V^{(L-2)}} = g'\left(b^{(L)}\right) w^{(L)} g'\left(b^{(L-1)}\right) w^{(L-1)}$$

- $$\frac{\partial V^{(L)}}{\partial V^{(L-3)}} = g'\left(b^{(L)}\right) w^{(L)} \frac{\partial V^{(L-1)}}{\partial V^{(L-3)}}$$

$$= g'\left(b^{(L)}\right) w^{(L)} \frac{\partial V^{(L-1)}}{\partial V^{(L-2)}} \frac{\partial V^{(L-2)}}{\partial V^{(L-3)}}$$

$$= g'\left(b^{(L)}\right) w^{(L)} \cdot g'\left(b^{(L-1)}\right) w^{(L-1)} \cdot g'\left(b^{(L-2)}\right) w^{(L-2)}$$

$\vdots$

It follows that:

$$\boxed{\frac{\partial V^{(L)}}{\partial V^{(\ell)}} = \prod_{k=L}^{\ell+1} \left[g'\left(b^{(k)}\right) w^{(k)}\right]}$$

[redacted]

6) <u>Reinforcement learning:</u>

a) $$P(y|x) = \prod_{i=1}^{M} \begin{cases} p(b_i) & \text{for } y_i = +1 \\ 1 - p(b_i) & \text{for } y = -1 \end{cases}$$

$$= (1)\, p(b_i) + (-1)\,(1 - p(b_i))$$

$$= \frac{1}{1+e^{-2\beta b_i}} - 1 + \frac{1}{1+e^{-2\beta b_i}}$$

$$= \frac{2}{1+e^{-2\beta b_i}} - 1$$

$$= \frac{2 - 1 - e^{-2\beta b_i}}{1+e^{-2\beta b_i}}$$

$$= \frac{1 - e^{-2\beta b_i}}{1+e^{-2\beta b_i}} \times \frac{e^{\beta b_i}}{e^{\beta b_i}}$$

$$= \frac{e^{\beta b_i} - e^{\beta b_i}}{e^{\beta b_i} + e^{\beta b_i}}$$

$$= \tanh(\beta b_i)$$

[redacted]

b) $\langle \delta w_{mn} \rangle = \eta \dfrac{\partial \langle r \rangle}{\partial w_n}$

Compare with $\dfrac{\partial \langle r \rangle}{\partial w_n} = \langle r(x,y)\, [y - \tanh(b)] \rangle\, x_n$ leads to

$$\delta w_{mn} = r\, [y - \tanh(b)]\, x_n$$

- unbiased estimator of a parameter is the estimator whose expected value is equal to the value of the parameter.