MLPR: Tutorial Sheet 5 Answers

School of Informatics, University of Edinburgh

Instructor: Charles Sutton

1. Consider the following classification problem. There are two real-valued features $x_1, x_2 \in \mathbb{R}$ and a binary class label. The class label is determined by
$$y = \begin{cases} 1 & \text{if } x_2 \geq |x_1| \\ 0 & \text{otherwise} \end{cases} \tag{1}$$
   (a) Can this be perfectly represented by a feedforward neural network without a hidden layer? Why or why not?

   (b) Let's consider a simpler problem for a moment. Consider the classification problem.
$$y = \begin{cases} 1 & \text{if } x_2 \geq x_1 \\ 0 & \text{otherwise} \end{cases} \tag{2}$$
   Design a single neuron that solves this problem. Pick the weights by hand. For an activation function, use the hard threshold function
$$h(a) = \begin{cases} 1 & \text{if } a \geq 0 \\ 0 & \text{otherwise} \end{cases}$$
   (c) Now go back to the classification problem at the beginning of this question. Design a two layer feedforward network (i.e., one hidden layer) that represents this function. Use the hard threshold activation function as in the previous question. *Hints:* Use two units in the hidden layer. The unit from the last question will be one of the units, and you will need to design one more. Your output unit will essentially perform a binary AND operation on the hidden units.

   (d) Consider the network that you created in the previous question. Suppose that we replace the hard threshold activation from that network function with a sigmoid activation function. Would your network still be correct? If not, is there a simply way to modify your network to make it approximately correct?

Solution:

(a) No. A neural network without a hidden layer simply computes a linear function of the inputs. The decision boundary required for this problem is not linear.

(b) Call the neuron z_1, and compute its output by
$$z_1 = h(v_{11}x_1 + v_{12}x_2 + v_{10})$$
Set
$$v_{11} = -1, v_{12} = 1, v_{10} = 0.$$
Now $z_1 > 0 \iff x_2 - x_1 \geq 0$.

(c) Define another hidden unit z_2 as
$$z_2 = h(v_{21}x_1 + v_{22}x_2 + v_{20})$$
set
$$v_{11} = 1, v_{12} = 1, v_{10} = 0.$$
Now $z_2 > 0 \iff x_1 + x_2 \geq 0$. So $x_2 \geq -x_1$.

The intersection of the area for which $z_1 \geq 0$ and the area for which $z_2 \geq 0$ is the area where $y = 1$. We can set up a logical AND with the following output unit:

$$y = h(w_1 z_1 + w_2 z_2 + w_0),$$

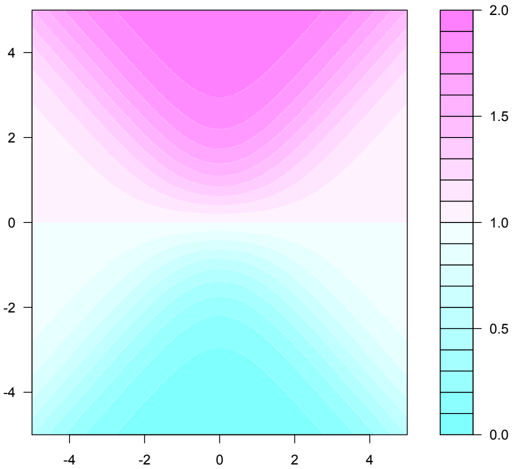
where we choose

$$w_1 = 1, w_2 = 1, w_0 = -1$$

You can show that now $y = 1 \iff x_2 \geq |x_1|$. Try this on a few example points to verify that it works.

(d) Something interesting happens here: Now along the line $x_1 = x_2$, the output unit $z_1 = 0.5$ instead of 0. Similarly along the line $x_2 = -x_1$, the output $z_2 = 0.5$ instead of 0.

Also notice if $x_2 = 0$, then necessarily $y = 1$. Here is a contour of the output unit as a function of x_1 and x_2:



We can use the sigmoid function to approximate a threshold function, though. Define a family of functions

$$\sigma_c(x) = \frac{1}{1 + \exp\{-cx\}}.$$

As we take $c \to \infty$, this will approximate the hard threshold function h.

What this means for us is that if we replace h with a sigmoids, we can just multiply all the weights by a large positive constant and we will recover the same network.

□

---

2. Let $p(\mathbf{x})$ be a mixture of Gaussians

$$p(\mathbf{x}) = \sum_{k=1}^{K} \pi_k \mathcal{N}(\mathbf{x}; \mu_k, \Sigma_k)$$

.

(a) What is $\mathrm{E}[X]$?

(b) What is $\mathrm{Cov}[X]$? *Hint:* Use $\mathrm{Cov}[X] = \mathrm{E}\left[XX^T\right] - \mathrm{E}[X]\,\mathrm{E}[X]^T$

Solution:

(a)

$$\begin{aligned}
\mathrm{E}[X] &= \int p(\mathbf{x})\mathbf{x}d\mathbf{x} \\
&= \int \sum_{k=1}^{K} \pi_k \mathcal{N}(\mathbf{x}; \mu_k, \Sigma_k)\mathbf{x}d\mathbf{x} \\
&= \sum_{k=1}^{K} \pi_k \mu_k .
\end{aligned}$$

That is, the mean of the mixture is the weighted average of the individual means.

(b) Define $E_k(f(X))$ to be the expectation of f according to component k, i.e.,

$$E_k(f(\mathbf{X})) = \int \mathcal{N}(\mathbf{x}; \mu_k, \Sigma_k) f(\mathbf{x}) d\mathbf{x}.$$

A similar argument to the previous question shows that

$$\mathrm{E}[X_i X_j] = \sum_k \pi_k E_k(X_i X_j).$$

So for the covariance matrix we have

$$\begin{aligned}
\mathrm{Cov}[X] &= \mathrm{E}\left[XX^T\right] - \mathrm{E}[X]\,\mathrm{E}[X]^T \\
&= \sum_{k=1}^{K} \pi_k E_k(XX^T) - \sum_{k=1}^{K}\sum_{j=1}^{K} \pi_j \pi_k \mu_j \mu_k^T
\end{aligned}$$

□

3. Show that the Metropolis Hastings algorithm for sampling from $P(x)$ satisfies detailed balance w.r.t. $P(x)$.

Solution: Consider the transition probability from x to y and y to x. Without loss of generality, presume $P(x) > P(y)$ (the symmetry of detailed balance means if it follows in one direction it follows in both). Then the transition probability combines the acceptance probability and the proposal distribution

$$P(y|x) = \frac{P(y)Q(x|y)}{P(x)Q(y|x)} Q(y|x) + \int dy' \left(1 - \frac{P(y')Q(x|y')}{P(x)Q(y'|x)}\right) \delta(y-x).$$

The second term comes from the case where the proposal is rejected: we propose y' and reject it, ending up back at x. We need to account for the fact that we could have proposed y' anywhere, hence the integral. Therefore

$$P(y|x)P(x) = \frac{P(y)Q(x|y)}{Q(y|x)} Q(y|x) + K(x)\delta(y-x)$$

where we have absorbed the integral term into the $K(x)$.

We can ignore the case where $x = y$ as detailed balance trivially holds then ($P(x|x)P(x) = P(x|x)P(x)$). So for $x \neq y$:

$$P(y|x)P(x) = P(y)Q(x|y)$$

In the other direction, the proposal is always accepted so we have

$$P(x|y)P(y) = Q(x|y)P(y)$$

This is equal to $P(y|x)P(x)$, so detailed balance holds.

This makes it clear why the use of proposals and acceptance is useful in building MCMC methods: as the $x = y$ case trivially satisfies detailed balance, any other imbalance in the probability elsewhere can just be sucked up using a rejection and put into the $x = y$ case without worrying about it.

□

4. Consider the univariate Gaussian model from the previous tutorial. The data is $\mathcal{D} = \{x_1 \dots x_n\}$, where each x_i is independent with

$$p(x_i|\mu, \tau) = \sqrt{\frac{\tau}{2\pi}} \exp\left\{-\frac{\tau}{2}(x_i - \mu)^2\right\}.$$

Now we will assume that both μ and τ are unknown. We will place prior distributions:

$$\begin{aligned} p(\mu|\mu_0, \tau_0) &= \sqrt{\frac{\tau_0}{2\pi}} \exp\left\{-\frac{\tau_0}{2}(\mu - \mu_0)^2\right\} \\ p(\tau|\alpha, \beta) &= \frac{\beta^\alpha}{\Gamma(\alpha)} \tau^{\alpha-1} \exp\{-\beta\tau\} \end{aligned}$$

Describe the Gibbs sampling algorithm to sample from the posterior $p(\mu, \tau|\mathcal{D})$. Give the specific equations for each step.

Solution: It will be easiest if we write down the joint distribution first (we will drop all terms that are constant with respect to x, τ, and μ):

$$\begin{aligned} p(\mathcal{D}, \mu, \tau) &= p(\tau|\alpha, \beta) p(\mu|\mu_0, \tau_0) \prod_i p(x_i|\mu, \tau) \\ &\propto \tau^{n/2+\alpha-1} \exp\left\{-\frac{\tau}{2}\sum_i (x_i - \mu)^2 - \frac{\tau_0}{2}(\mu - \mu_0)^2 - \beta\tau\right\} \end{aligned}$$

The Gibbs sampler maintains a current guess (μ_t, τ_t) at every iteration t. It resamples μ_{t+1} from $p(\mu|\tau_t, \mathcal{D})$ and τ_{t+1} from $p(\tau|\mu_t, \mathcal{D})$. We can compute these conditional distribution very quickly by using our tricks of ignoring constants.

First, for τ, we have:

$$\begin{aligned} p(\tau|\mu, \mathcal{D}) &\propto p(\mathcal{D}, \mu, \tau) \\ &\propto \tau^{n/2+\alpha-1} \exp\left\{-\frac{\tau}{2}\sum_i (x_i - \mu)^2 - \beta\tau\right\} \\ &= \tau^{n/2+\alpha-1} \exp\left\{-\tau\left(\beta + \frac{1}{2}\sum_i (x_i - \mu)^2\right)\right\} \\ &\propto \text{Gamma}\left(\alpha + n/2, \quad \beta + \frac{1}{2}\sum_i (x_i - \mu)^2\right) \end{aligned}$$

So we resample τ from a Gamma distribution whose shape depends only on the number of data points, and whose scale parameter depends on the average deviation of the data from the current value of μ.

The conditional distribution for μ requires a bit more algebra (we need to complete the square). There are two main tricks:

(a) We can ignore any constants with respect to μ. This works within the exponential too: $\exp\{\mu + C\} = \exp\{\mu\}\exp\{C\} \propto \exp\{\mu\}$ if C is constant with respect to μ.

(b) Suppose that we are able to work out that

$$p(\mu|\tau, \mathcal{D}) \propto \exp\left\{-\frac{t}{2}\left(\mu^2 - 2a\mu\right)\right\}$$

Then we know that $p(\mu|\tau, \mathcal{D})$ is normal with mean a and precision t (this can be shown by completing the square). So we're going to start with $p(\mathcal{D}, \mu, \tau)$ and try to massage it into this form.

Using these tricks gives us:

$$
\begin{aligned}
p(\mu|\tau, \mathcal{D}) &\propto p(\mathcal{D}, \mu, \tau) \\
&\propto \tau^{n/2+\alpha-1} \exp\left\{ -\frac{\tau}{2} \sum_i (x_i - \mu)^2 - \frac{\tau_0}{2}(\mu - \mu_0)^2 - \beta\tau \right\} \\
&\propto \exp\left\{ -\frac{\tau}{2} \sum_i (x_i - \mu)^2 - \frac{\tau_0}{2}(\mu - \mu_0)^2 \right\} \\
&= \exp\left\{ -\frac{\tau}{2} \sum_i x_i^2 + \tau\mu \sum_i x_i - \frac{\tau}{2} \sum_i \mu^2 - \frac{\tau_0}{2}\mu^2 + \tau_0\mu\mu_0 - \frac{\tau_0}{2}\mu_0^2 \right\} \\
&\propto \exp\left\{ \mu^2 \left(-\frac{n\tau}{2} - \frac{\tau_0}{2} \right) + \mu \left(\tau \sum_i x_i + \tau_0\mu_0 \right) \right\} \\
&\propto \exp\left\{ -\frac{n\tau + \tau_0}{2} \left(\mu^2 - 2\mu \frac{\tau \sum_i x_i + \tau_0\mu_0}{n\tau + \tau_0} \right) \right\} \\
&\propto \mathcal{N}\left(\mu; a, t\right),
\end{aligned}
$$

where

$$
\begin{aligned}
a &= \frac{n\tau}{n\tau + \tau_0}\bar{x} + \frac{\tau_0}{n\tau + \tau_0}\mu_0 \\
b &= (n\tau + \tau_0)^{-1},
\end{aligned}
$$

by $\bar{x} = n^{-1} \sum_i x_i$ we mean the sample mean of $\mathcal{D}$. The conditional mean a has a nice interpretation: it is the weighted average of the sample mean $\bar{x}$ and the prior mean μ_0, where the weights are given by the strength of the prior τ_0 and our current estimate τ of the strength of the likelihood. Notice that τ will change as we run the Gibbs sampler, and also that as n increases, we tend to give more weight to the sample mean rather than the prior mean.

□
