# Intro to Machine Learning Homework Assignment 4

Goktug Saatcioglu

NetId: gs2417

**1.** For the 2-dimensional XOR problem, we have the following data vectors and labels:

$$\begin{aligned}
\mathbf{x}^1 &= [-1,-1]^\top \text{has label } y^1 = 1 \\
\mathbf{x}^2 &= [1,1]^\top \quad\ \ \text{has label } y^2 = 1 \\
\mathbf{x}^3 &= [-1,1]^\top \ \ \text{has label } y^3 = -1 \\
\mathbf{x}^4 &= [1,-1]^\top \ \ \text{has label } y^4 = -1.
\end{aligned}$$

We use the following basis vectors:

$$\begin{aligned}
\mathbf{r}^1 &= [-1,-1]^\top \\
\mathbf{r}^2 &= [1,1]^\top \\
\mathbf{r}^3 &= [-1,1]^\top \\
\mathbf{r}^4 &= [1,-1]^\top,
\end{aligned}$$

and we tranform each two-dimensional input vector $\mathbf{x}^i$ into a four-dimensional vector $\boldsymbol{\phi}(\mathbf{x})$ such that

$$\boldsymbol{\phi}^i(\mathbf{x}) = \exp(-\left\|\mathbf{x} - \mathbf{r}^j\right\|^2),$$

where $i = 1,2,3,4$, $j = 1,2,3,4$ and $\|\cdot\|$ is the 2-norm of a vector (or $\|\mathbf{x} - \mathbf{r}\|$ is the Euclidean distance between $x$ and $r$). We compute each $\phi^i$ and the results are shown below:

$$\begin{aligned}
\boldsymbol{\phi}^1 &= [1, e^{-8}, e^{-4}, e^{-4}]^\top \\
\boldsymbol{\phi}^2 &= [e^{-8}, 1, e^{-4}, e^{-4}]^\top \\
\boldsymbol{\phi}^3 &= [e^{-4}, e^{-4}, 1, e^{-8}]^\top \\
\boldsymbol{\phi}^4 &= [e^{-4}, e^{-4}, e^{-8}, 1]^\top.
\end{aligned}$$

Next, we let $\mathbf{w} = [1,1,-1,-1]^\top$, $b = 0$ and solve $\mathbf{w}^\top\boldsymbol{\phi}^i + b$ for $i = 1,2,3,4$. If the answer is greater than 0 we label the class as 1 and $-1$ otherwise. The results are given below:

$$\begin{aligned}
\mathbf{w}^\top\boldsymbol{\phi}^1 + b &\approx 0.9637 &&\implies \mathbf{x}^1 \text{ has label } 1 = y^1 \\
\mathbf{w}^\top\boldsymbol{\phi}^2 + b &\approx 0.9637 &&\implies \mathbf{x}^2 \text{ has label } 1 = y^2 \\
\mathbf{w}^\top\boldsymbol{\phi}^3 + b &\approx -0.9637 &&\implies \mathbf{x}^3 \text{ has label } -1 = y^3 \\
\mathbf{w}^\top\boldsymbol{\phi}^4 + b &\approx -0.9637 &&\implies \mathbf{x}^4 \text{ has label } -1 = y^4.
\end{aligned}$$

Thus, we conclude that the radial basis function network with $\mathbf{w} = [1,1,-1,-1,0]^\top$ solves the XOR-problem.

**2.** In a multiclass classification setting the weight vector can be built as $\mathbf{W} = [y^1, y^2, \ldots, y^k]$ where $y^i$ is a one-hot vector corresponding to the class which the $i$-th basis vector belongs to. This weight matrix would then have size $n \times k$ where $n$ is the number of classes we have in the multiclass classification setting and $k$ is the amount of basis vector we have chosen (which in this case is all the input vectors). This works since the mutliplication of the weight matrix $\mathbf{W}$ with any of the radial bases vectors $\phi_i(\mathbf{x})$ will return us a column vector of size $n \times 1$ because $\mathbf{W}$ is of size $n \times k$ and $\phi_i$ is of size $k \times 1$. This column vector can be interpreted as a sort-of one hot vector where the entry with the highest value indicates the class that the input $\mathbf{x}$ belongs to. Matrix multiplication happens row by columns which means that the $i$-th entry in the resulting vector from $\mathbf{W}\phi_i(\mathbf{x})$ will measure the "likliness/similarity" the input vector $\mathbf{x}$ has to the class $i$. Technically, each $i$ in the resulting vector would measure the total inversely proportinal distance between input vector $\mathbf{x}$ and the bases that are given by class $i$. Thus, the highest entry for a resulting input vector $\mathbf{x}$ will be the entry its class label belongs to which means we correctly create a nearest-neighbor classifier from a radial basis function network in the multiclass classification setting. Finally, for the resulting vector we can apply a softmax transformation on the entries and select the element with the highest value to classify the input vector $\mathbf{x}$.

**3.** With K basis vectors, the distance function, as given in the lecture notes, is

$$D(y^*, M, \phi(\mathbf{x})) = -(y^* \log(M(\phi(\mathbf{x}))) + (1 - y^*)\log(1 - M(\phi(\mathbf{x})))),$$

where $y^* = M^*(\phi(\mathbf{x}))$, $M = M(\mathbf{x}) = \sigma(\mathbf{w}^\top \mathbf{x} + b)$ and

$$\phi(\mathbf{x}) = \begin{bmatrix} \exp(-(\mathbf{x} - \mathbf{r}^1)^2) \\ \vdots \\ \exp(-(\mathbf{x} - \mathbf{r}^k)^2) \end{bmatrix}.$$

To compute $\nabla_{\mathbf{r}^k} D(y^*, M, \phi(\mathbf{x}))$ we compute the partials $\frac{\partial D}{\partial a}$, $\frac{\partial a}{\partial \phi_k(\mathbf{x})}$ and $\nabla_{\mathbf{r}^k}\phi_k(\mathbf{x})$ where $a = \mathbf{w}^\top\phi(\mathbf{x}) + b$ such that

$$\frac{\partial D}{\partial a}\frac{\partial a}{\partial \phi_k(\mathbf{x})}\nabla_{\mathbf{r}^k}\phi_k(\mathbf{x}) = \nabla_{\mathbf{r}^k} D(y^*, M, \phi(\mathbf{x})).$$

We start with $\frac{\partial D}{\partial a}$:

$$\begin{aligned}
\frac{\partial D}{\partial a} &= \frac{\partial}{\partial a}(-(y^* \log(M(\phi(\mathbf{x}))) + (1 - y^*)\log(1 - M(\phi(\mathbf{x}))))) \\
&= \frac{\partial}{\partial a}(-(y^* \log(\sigma(\mathbf{w}^\top\phi(\mathbf{x}) + b)) + (1 - y^*)\log(1 - \sigma(\mathbf{w}^\top\phi(\mathbf{x}) + b)))) && \text{using the definition of } M \\
&= \frac{\partial}{\partial a}(-(y^* \log(\sigma(a)) + (1 - y^*)\log(1 - \sigma(a)))) && \text{since } a = \mathbf{w}^\top\phi(\mathbf{x}) + b \\
&= -(y^* \frac{\partial}{\partial a}(\log(\sigma(a))) + (1 - y^*)\frac{\partial}{\partial a}(\log(1 - \sigma(a)))) && \text{by linearity} \\
&= -(y^* \frac{\frac{\partial}{\partial a}\sigma(a)}{\sigma(a)} + (1 - y^*)\frac{\frac{\partial}{\partial a}(1 - \sigma(a))}{1 - \sigma(a)}) && \text{by chain rule} \\
&= -(y^* \frac{\sigma(a)(1 - \sigma(a))\frac{\partial}{\partial a}a}{\sigma(a)} + (1 - y^*)\frac{-\sigma(a)(1 - \sigma(a))\frac{\partial}{\partial a}}{1 - \sigma(a)}) && \text{by chain rule and since} \\
& && \frac{\partial}{\partial x}\sigma(x) = \sigma(x)(1 - \sigma(x)) \\
&= -(y^*(1 - \sigma(a)) + (1 - y^*) - \sigma(a)) && \text{simplify fractions} \\
&= -(y^* - y^*\sigma(a) - \sigma(a) + y^*\sigma(a)) && \text{expand terms} \\
&= -(y^* - \sigma(a)) && \text{simplify terms} \\
&= -(y^* - \sigma(\mathbf{w}^\top\phi(\mathbf{x}) + b)) && \text{since } a = \mathbf{w}^\top\phi(\mathbf{x}) + b
\end{aligned}$$

$$\therefore \frac{\partial D}{\partial a} = -(y^* - \sigma(\mathbf{w}^\top\phi(\mathbf{x}) + b))$$

Next we solve $\frac{\partial a}{\partial \phi_k(\mathbf{x})}$:

$$\begin{aligned}
\frac{\partial a}{\partial \phi_k(\mathbf{x})} &= \frac{\partial}{\partial \phi_k(\mathbf{x})} a \\
&= \frac{\partial}{\partial \phi_k(\mathbf{x})}(\mathbf{w}^\top \phi(\mathbf{x}) + b) && \text{since } a = \mathbf{w}^\top \phi(\mathbf{x}) + b \\
&= w_k && \text{since } \frac{\partial}{\partial \phi_k(\mathbf{x})}(\mathbf{w}^\top \phi(\mathbf{x})) = w_k \\
\therefore \frac{\partial a}{\partial \phi_k(\mathbf{x})} &= w_k
\end{aligned}$$

Finally we solve $\nabla_{\mathbf{r}^k} \phi_k(\mathbf{x})$:

$$\begin{aligned}
\nabla_{\mathbf{r}^k} \phi_k(\mathbf{x}) &= \frac{\partial}{\partial \mathbf{r}^k}(\begin{bmatrix} \exp(-(\mathbf{x} - \mathbf{r}^1)^2) \\ \vdots \\ \exp(-(\mathbf{x} - \mathbf{r}^k)^2) \end{bmatrix}) \\
&= \begin{bmatrix} \frac{\partial}{\partial \mathbf{r}^k}(\exp(-(\mathbf{x} - \mathbf{r}^1)^2)) \\ \vdots \\ \frac{\partial}{\partial \mathbf{r}^k}(\exp(-(\mathbf{x} - \mathbf{r}^k)^2)) \end{bmatrix} && \text{by linearity} \\
&= \begin{bmatrix} 0 \\ \vdots \\ (\exp(-(\mathbf{x} - \mathbf{r}^k)^2))(-2(\mathbf{x} - \mathbf{r}^k)) \end{bmatrix} && \text{by chain rule} \\
&= \begin{bmatrix} 0 \\ \vdots \\ 2\phi_k(\mathbf{x})(\mathbf{x} - \mathbf{r}^k) \end{bmatrix} && \text{since } \exp(-(\mathbf{x} - \mathbf{r}^k)^2) = \phi_k(\mathbf{x}) \\
&= 2\phi_k(\mathbf{x})(\mathbf{x} - \mathbf{r}^k) && \text{since all entries bu } k \text{ is zero} \\
\therefore \nabla_{\mathbf{r}^k} \phi_k(\mathbf{x}) &= 2\phi_k(\mathbf{x})(\mathbf{x} - \mathbf{r}^k)
\end{aligned}$$

Combining all three together:

$$\begin{aligned}
\frac{\partial D}{\partial a} \frac{\partial a}{\partial \phi_k(\mathbf{x})} \nabla_{\mathbf{r}^k} \phi_k(\mathbf{x}) &= (-(y^* - \sigma(\mathbf{w}^\top \phi(\mathbf{x}) + b)))(w_k)(2\phi_k(\mathbf{x})(\mathbf{x} - \mathbf{r}^k)) \\
&= -2(y^* - \sigma(\mathbf{w}^\top \phi(\mathbf{x}) + b)) w_k \phi_k(\mathbf{x})(\mathbf{x} - \mathbf{r}^k) \\
\therefore \nabla_{\mathbf{r}^k} D(y^*, M, \phi(\mathbf{x})) &= -2(y^* - \sigma(\mathbf{w}^\top \phi(\mathbf{x}) + b)) w_k \phi_k(\mathbf{x})(\mathbf{x} - \mathbf{r}^k)
\end{aligned}$$

Thus, we have derived the gradient asked by the question and our answer is the same as the answer given in the lecture notes which verifies our solution.