Neural networks: Assignment 1

HABIMANA Martin

March 2024

Problem 1: Simulate a boolean function

Neural Network

A neural network NN with four inputs (x_1, x_2, x_3, x_4) and one output is defined as follows:

$$NN(x_1, x_2, x_3, x_4) = 1 + x_1x_2 + x_2x_3 + x_3x_4 \pmod 2$$

where x_1, x_2, x_3, x_4 are binary variables, i.e., $x_i \in \{0, 1\}$ for every $i = 1, 2, 3, 4$.

The output of the neural network for all possible combinations of inputs is as follows:

```
NN(0, 0, 0, 0) = 1
NN(0, 0, 0, 1) = 1
NN(0, 0, 1, 0) = 1
NN(0, 0, 1, 1) = 0
NN(0, 1, 0, 0) = 1
NN(0, 1, 0, 1) = 1
NN(0, 1, 1, 0) = 0
NN(0, 1, 1, 1) = 1
NN(1, 0, 0, 0) = 1
NN(1, 0, 0, 1) = 1
NN(1, 0, 1, 0) = 1
NN(1, 0, 1, 1) = 0
NN(1, 1, 0, 0) = 0
NN(1, 1, 0, 1) = 0
NN(1, 1, 1, 0) = 1
NN(1, 1, 1, 1) = 0
```

Problem 2: Equivalence of activations

(a) Recall the step function $H(x) = 1(x \geq 0)$ and the sigmoid $S(x) = \frac{1}{1+\exp(-x)}$. For every $\varepsilon > 0$, describe a neural network NN with one input, one output, and sigmoid activations such that $|NN(x) - H(x)| \leq \varepsilon$ for every x such that $|x| > \varepsilon$.

Answer:
The neural network is,

$$NN(x) = \frac{1}{1 + \exp(-kx)}$$

Let's prove that for every $\varepsilon > 0$, there exists a k such that $|NN(x) - H(x)| \leq \varepsilon$ for every x such that $|x| > \varepsilon$.

$$|NN(x) - H(x)| = \left| \frac{1}{1 + \exp(-kx)} - 1(x \geq 0) \right|$$

Let's choose $k = \frac{1}{\varepsilon}$. Then, for $|x| > \varepsilon$:

$$\left| \frac{1}{1 + \exp\left(-\frac{x}{\varepsilon}\right)} - 1 \right| \leq \varepsilon$$

Now, since $|x| > \varepsilon$, we have $\frac{x}{\varepsilon} > 1$, which means $\exp\left(\frac{x}{\varepsilon}\right) > \exp(1)$. Therefore:

$$\begin{aligned}
\frac{1}{1 + \exp(-kx)} &\leq \frac{1}{1 + \exp(-k\varepsilon)} \\
&= \frac{1}{1 + \exp\left(-\frac{\varepsilon}{\varepsilon}\right)} \\
&= \frac{1}{1 + \exp(-1)} \\
&= \frac{1}{1 + \frac{1}{e}} \\
&= \frac{e}{e+1} \\
&\leq \varepsilon
\end{aligned}$$

So, $NN(x) = \frac{1}{1+\exp(-kx)}$ satisfies the condition $|NN(x) - H(x)| \leq \varepsilon$ for every x such that $|x| > \varepsilon$.

(b) Let $f : [-1, 1] \to \mathbb{R}$ be a continuous function and $\varepsilon > 0$. Describe a neural network NN with one input, one output, using step and identity activations, such that for every $-1 \leq x \leq 1$,

$$|NN(x) - H(x)| \leq \varepsilon$$

Hint: Note that for $b_1 \leq b_2$, we have $H(x - b_1) - H(x - b_2) = 1$ if $b_1 \leq x < b_2$. **Answer:**

We have,

$$|NN(x) - f(x)| < \epsilon \quad \text{for} \quad x \in [-1, 1].$$
$$|f(x) - (H(x+1) - H(x-1))f(x)| < \epsilon$$
$$\boxed{NN(x) = [H(x+1) - H(x-1)]f(x)}$$

Problem 3: Gradients for one neuron

a. In order to find the partial derivative of the square loss with respect to w_i for the given ReLU neuron, we need to consider the chain rule.

Let's break down the expression:

$$f_R(x) = \text{ReLU}(w^T \cdot x + b)$$
$$C_{\text{sq}}(f_R(x), \bar{y}) = (f_R(x) - \bar{y})^2$$

Now, let's compute the partial derivative $\frac{\partial}{\partial w_i} C_{\text{sq}}(f_R(x), \bar{y})$ using the chain rule

$$\frac{\partial}{\partial w_i} C_{\text{sq}}(f_R(x), \bar{y}) = 2(f_R(x) - \bar{y}) \cdot \frac{\partial}{\partial w_i} f_R(x)$$

Next, let's compute $\frac{\partial}{\partial w_i} f_R(x)$.

The ReLU activation function has a piecewise definition,

$$\text{ReLU}(z) = \begin{cases} z, & \text{if } z > 0 \\ 0, & \text{otherwise} \end{cases}$$

So, the partial derivative of $\text{ReLU}(w^T \cdot x + b)$ with respect to w_i is follow

$$\frac{\partial}{\partial w_i} f_R(x) = \begin{cases} x_i, & \text{if } w^T \cdot x + b > 0 \\ 0, & \text{otherwise} \end{cases}$$

Hence,

$$\boxed{\frac{\partial}{\partial w_i} C_{\text{sq}}(f_R(x), \bar{y}) = 2(f_R(x) - \bar{y}) \cdot \begin{cases} x_i, & \text{if } w^T \cdot x + b > 0 \\ 0, & \text{otherwise} \end{cases}}$$

b. Consider a sigmoid neuron $f_S = S(w^T \cdot x + b)$ and the cross-entropy loss function $C_{\text{cross}}(y, f_S(x)) = -\bar{y} \ln(f_S) - (1 - \bar{y}) \ln(1 - f_S)$ for $0 \le y \le 1$.

We want to calculate the partial derivative of the cross-entropy loss with respect to the weight w_i, denoted as $\frac{\partial C_{\text{cross}}(f_S(x), \bar{y}}{\partial w_i}$.

Answer:
The cross-entropy loss function,

$$C_{\text{cross}}(f_S(x), \bar{y}) = -\bar{y} \ln(f_S(x)) - (1 - \bar{y}) \ln(1 - f_S(x))$$

Let's apply the chain rule,

$$\frac{\partial C_{\text{cross}}}{\partial w_i} = \frac{\partial C_{\text{cross}}}{\partial f_S(x)} \cdot \frac{\partial f_S(x)}{\partial (w_i \cdot x)} \cdot \frac{\partial (w_i \cdot x)}{\partial w_i}$$

- $\frac{\partial C_{\text{cross}}}{\partial f_S(x)} = -\frac{y}{f_S(x)} + \frac{1-y}{1-f_S(x)}$
- $\frac{\partial f_S(x)}{\partial (w_i \cdot x)} = f_S(x) \cdot (1 - f_S(x))$
- $\frac{\partial (w_i \cdot x)}{\partial w_i} = x_i$

Substitute these into the chain rule equation, we get

$$\boxed{\frac{\partial C_{\text{cross}}}{\partial w_i} = \left(-\frac{y}{f_S(x)} + \frac{1 - y}{1 - f_S(x)} \right) \cdot (f_S(x) \cdot (1 - f_S(x))) \cdot x_i}$$

c.

- Let $f_T(x) = \tanh(w^T \cdot x + b)$ be a neuron with activation $\tanh(x) = \frac{\exp(2x)-1}{\exp(2x)+1}$, and
- let $C_{hinge}(f_T(x), \bar{y}) = \max(0, 1 - \bar{y} \cdot f_T(x))$ be the hinge loss for $\bar{y} \in \{-1, 1\}$.

We want to calculate $\frac{\partial}{\partial w_i} C_{hinge}(f_T(x), \bar{y})$.

Answer:

Let's calculate the partial derivative $\frac{\partial}{\partial w_i} C_{hinge}(f_T(x), y)$, we need to apply the chain rule.

$$\frac{\partial}{\partial w_i} C_{hinge}(f_T(x), y) = \frac{\partial C_{hinge}}{\partial f_T(x)} \cdot \frac{\partial f_T(x)}{\partial w_i}$$

First, let's find $\frac{\partial f_T(x)}{\partial w_i}$:

$$\frac{\partial f_T(x)}{\partial w_i} = \frac{\partial}{\partial w_i} \tanh(w^T \cdot x + b)$$

The derivative of the hyperbolic tangent function $\tanh(x)$ is $1 - \tanh^2(x)$. Therefore,

$$\frac{\partial f_T(x)}{\partial w_i} = x_i(1 - \tanh^2(w^T \cdot x + b))$$

The derivative of the hinge loss $\frac{\partial C_{hinge}}{\partial f_T(x)}$ with respect to $f_T(x)$ is a piecewise function:

$$\frac{\partial C_{hinge}}{\partial f_T(x)} = \begin{cases} -\bar{y} & \text{if } 1 - \bar{y} f_T(x) > 0 \\ 0 & \text{otherwise} \end{cases}$$

Now, we put them all together:

$$\boxed{\frac{\partial}{\partial w_i} C_{hinge}(f_T(x), \bar{y}) = \begin{cases} -\bar{y} \cdot x_i(1 - \tanh^2(w^T \cdot x + b)) & \text{if } 1 - \bar{y} f_T > 0 \\ 0 & \text{otherwise} \end{cases}}$$

Problem 4: Accuracy at initialization

Assume there is a dataset with M points and labels $x^{(i)} \in \mathbb{R}^d$, $y^{(i)} \in \{0, 1, 2, 3\}$ for $i = 1, \ldots, M$. What accuracy can you expect for this dataset at initialization?

Answer:

In a classification task with C classes, the random accuracy can be calculated as follow:

$$\text{Accuracy} = \frac{1}{C}$$

Since we have $C = 4$ classes (0, 1, 2, 3), the accuracy would be $\frac{1}{4} = 0.25$ or 25%.

$$\boxed{\text{Accuracy} = 25\%}$$