Comp 540 Machine Learning

Yue He, Peng Yang

Feruary 2019

On my honor, I have neither given nor received any unauthorized aid on this assignment.

1 Deep neural networks

1. The reason behind the boost in performance from a deeper network, is that a more complex, non-linear function can be learned. If sufficient training data is given, deep network enables discriminate between different classes more easily. The advantage of multiple layers is that they can learn features at various levels. And it seems that for a fixed number of parameters (or a fixed order of magnitude), going deeper allows the models to capture richer structures.

2. Leaky ReLU is ReLU whose slope is small for negative values, instead of only zero. It has two benefits: It fixes the "dying ReLU" (no matter what input, the output is zero) problem, as it doesn't have zero-slope parts. It speeds up training. Pooling layers' function is to progressively reduce the spatial size of the representation to reduce the amount of parameters and computation in the network. Pooling layer operates on each feature map independently. The most common approach used in pooling is max pooling.

3. In a paragraph or so, and using sketches as appropriate, contrast: AlexNet, VGG-Net, GoogleNet and ResNet. What is the one dening characteristic of each network?

(1) AlexNet: contained eight layers; the first five were convolutional layers, some of them followed by max-pooling layers, and the last three were fully connected layers. It used the non-saturating ReLU activation function, which showed improved training performance over tanh and sigmoid.

(2) VGGNet: consists of 16 convolutional layers and is very appealing because of its very uniform architecture. Similar to AlexNet, only 3x3 convolutions, but lots of filters.

(3) GoogleNet: An inception module is the basic building block of the network. In short, the inception module does multiple convolutions, with different filter sizes, and as well as pooling in one layer. As a result, instead of having us decide when to use which type of layer for the best result, the network automatically figures this out after training.

(4) ResNet: Allow the original input information to be passed directly to the next layer. So that the neural network of this layer do not need to learn the whole output, but the residual of the output of the previous network.

Extra Credit: Take the following as example, the general convolution formula is:

$$C(x,y,z) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} G(x-x', y-y', z)S(x',y')\,dx'\,dy'.$$

Figure 1: original convolution function

When the broad-beam response C(x,y,z), C(x,y,z) has cylindrical symmetry, its convolution integrals can be rewritten as:

$$C(r,z) = \int_0^{\infty} S(r')r'\left[\int_0^{2\pi} G\left(\sqrt{r^2+r'^2-2rr'cos\phi'}, z\right)\,d\phi'\right]dr'$$
$$C(r,z) = \int_0^{\infty} G(r'',z)r''\left[\int_0^{2\pi} S\left(\sqrt{r^2+r''^2-2rr''cos\phi''}\right)\,d\phi''\right]dr''$$

Figure 2: cylindrical symmetry: convolution function

2 Decision trees, entropy and information gain

(a) Since we know that

$$H(S) = -S\ log_2(S) - (1-S)log_2(1-S)$$

Then, we know

$$\frac{\partial H(S)}{\partial S} = -log_2(S) - S\frac{1}{S\ ln2} + log_2(1-S) + (1-S)\frac{1}{(1-S)\ ln2} = 0$$
$$\Rightarrow S = \frac{1}{2}$$

Besides,

$$\frac{\partial^2 H(S)}{\partial S^2} = -\frac{1}{Sln2} - \frac{1}{(1-S)ln2} < 0$$

where $S \in (0,1)$ and $H(S)$ is a concave which indicates it has max value. When $S = \frac{1}{2}$, $H(S) = 1$, which is the max value of the entropy function. Thus, $H(S) \leq 1$. And, when $H(S) = 1$, $S = \frac{p}{p+n} = \frac{1}{2}$. We get $p = n$.

(b) Model A.

1. Misclassification rate

$Cost(D) = \frac{1}{2} \quad Cost(D_{left}) = \frac{100}{400} = \frac{1}{4} \quad Cost(D_{Right}) = \frac{100}{400} = \frac{1}{4}$

$$Reduction = Cost(D) - \frac{|D_{left}|}{|D|}Cost(D_{left}) - \frac{|D_{right}|}{|D|}Cost(D_{right})$$
$$= \frac{1}{2} - \frac{1}{2}\frac{1}{4} - \frac{1}{2}\frac{1}{4} = \frac{1}{4}.$$

2. Entropy

$$Cost(D) = -\frac{1}{2}log_2(\frac{1}{2}) - \frac{1}{2}log_2(\frac{1}{2}) = 1$$
$$Cost(D_{left}) = -\frac{3}{4}log_2-\frac{3}{4} - \frac{1}{4}log_2-\frac{1}{4} = 0.8113$$
$$Cost(D_{right}) = -\frac{3}{4}log_2-\frac{3}{4} - \frac{1}{4}log_2-\frac{1}{4} = 0.8113$$

$$\begin{aligned} Reduction &= Cost(D) - \frac{|D_{left}|}{|D|}Cost(D_{left}) - \frac{|D_{right}|}{|D|}Cost(D_{right}) \\ &= 1 - \frac{1}{2}0.8113 - \frac{1}{2}0.8113 = 0.1887. \end{aligned}$$

3. Gini Index

$Cost(D) = 2\frac{1}{2}(1-\frac{1}{2}) = \frac{1}{2} \quad Cost(D_{left}) = 2\frac{3}{4}(1-\frac{3}{4}) = \frac{3}{8} \quad Cost(D_{Right}) = 2\frac{1}{4}(1-\frac{1}{4}) = \frac{3}{8}$

$$\begin{aligned} Reduction &= Cost(D) - \frac{|D_{left}|}{|D|}Cost(D_{left}) - \frac{|D_{right}|}{|D|}Cost(D_{right}) \\ &= \frac{1}{2} - \frac{1}{2}\frac{3}{8} - \frac{1}{2}\frac{3}{8} = \frac{1}{8}. \end{aligned}$$

Model B.

1. Misclassification rate

$Cost(D) = \frac{1}{2} \quad Cost(D_{left}) = \frac{200}{600} = \frac{1}{3} \quad Cost(D_{Right}) = \frac{0}{200} = 0$

$$\begin{aligned} Reduction &= Cost(D) - \frac{|D_{left}|}{|D|}Cost(D_{left}) - \frac{|D_{right}|}{|D|}Cost(D_{right}) \\ &= \frac{1}{2} - \frac{3}{4}\frac{1}{3} - \frac{1}{4}0 = \frac{1}{4}. \end{aligned}$$

2. Entropy

$$Cost(D) = -\frac{1}{2}log_2(\frac{1}{2}) - \frac{1}{2}log_2(\frac{1}{2}) = 1$$
$$Cost(D_{left}) = -\frac{1}{3}log_2-\frac{1}{3} - \frac{2}{3}log_2-\frac{2}{3} = 0.9183$$
$$Cost(D_{right}) = -\frac{1}{1}log_2-\frac{1}{1} - \frac{0}{1}log_2-\frac{0}{1} = 0$$

$$\begin{aligned} Reduction &= Cost(D) - \frac{|D_{left}|}{|D|}Cost(D_{left}) - \frac{|D_{right}|}{|D|}Cost(D_{right}) \\ &= 1 - \frac{3}{4}0.9183 - \frac{1}{4}0 = 0.3113. \end{aligned}$$

3. Gini Index

$Cost(D) = 2\frac{1}{2}(1-\frac{1}{2}) = \frac{1}{2}$ $Cost(D_{left}) = 2\frac{1}{3}(1-\frac{1}{3}) = \frac{4}{9}$ $Cost(D_{Right}) = 2\times 1\times(1-1) = 0$

$$\begin{aligned} Reduction &= Cost(D) - \frac{|D_{left}|}{|D|}Cost(D_{left}) - \frac{|D_{right}|}{|D|}Cost(D_{right}) \\ &= \frac{1}{2} - \frac{3}{4}\frac{4}{9} - \frac{1}{4}0 = \frac{1}{6}. \end{aligned}$$

Comparing Model A with Model B, the misclassification rate is the same; however, Model B has more reduction in entropy and Gini Index. Therefore, Model B is the preferred split.

(c) No, the misclassification rate won't increase when splitting on a feature.
Let $D = (C_1, C_2)$, $C_1 = (C_{11}, C_{21})$, $C_2 = (C_{12}, C_{22})$, and $D_{left} = (C_{11}, C_{12})$, $D_{right} = (C_{21}, C_{22})$.
Assuming that $|C_1| < |C_2|$, then, there are 2 cases.
Case 1: when $C_{11} < C_{12}$, $C_{21} < C_{22}$, and $C_{11} + C_{21} < C_2$.

$$\begin{aligned} Cost(D) &= \frac{|C_{11}| + |C_{21}|}{|D|} = \frac{|C_1|}{|D|} \\ Cost(D_{left}) &= \frac{|C_{11}|}{|C_{11}| + |C_{12}|} = \frac{|C_{11}|}{|D_{left}|} \\ Cost(D_{right}) &= \frac{|C_{21}|}{|C_{21}| + |C_{22}|} = \frac{|C_{21}|}{|D_{right}|} \end{aligned}$$

Then,

$$\begin{aligned} Reduction &= \frac{|C_1|}{|D|} - \frac{|D_{left}|}{|D|}\frac{|C_{11}|}{|D_{left}|} - \frac{|D_{right}|}{|D|}\frac{|C_{21}|}{|D_{right}|} \\ &= \frac{|C_1| - |C_{11}| - |C_{21}|}{D} = 0 \end{aligned}$$

Case 2: when $C_{11} < C_{12}$, $C_{21} > C_{22}$, and $C_{11} + C_{21} < C_2$.

$$\begin{aligned} Cost(D) &= \frac{|C_{11}| + |C_{21}|}{|D|} = \frac{|C_1|}{|D|} \\ Cost(D_{left}) &= \frac{|C_{11}|}{|C_{11}| + |C_{12}|} = \frac{|C_{11}|}{|D_{left}|} \\ Cost(D_{right}) &= \frac{|C_{22}|}{|C_{21}| + |C_{22}|} = \frac{|C_{22}|}{|D_{right}|} \end{aligned}$$

Then,

$$\begin{aligned} Reduction &= \frac{|C_1|}{|D|} - \frac{|D_{left}|}{|D|}\frac{|C_{11}|}{|D_{left}|} - \frac{|D_{right}|}{|D|}\frac{|C_{22}|}{|D_{right}|} \\ &= \frac{|C_1| - |C_{11}| - |C_{22}|}{|D|} = \frac{|C_{21}| - |C_{22}|}{|D|} > 0 \end{aligned}$$

Thus, $Reduction \geq 0$,the misclassification rate won't increase when splitting on a feature.

3 Bagging

(a)

$$
\begin{aligned}
E_{bag} = E_x(\epsilon_{bag}(x)^2) &= E_x\{\big([\frac{1}{L}\sum_{l=1}^{L} f(x) + \epsilon_l(x)] - f(x)\big)^2)\} \\
&= \frac{1}{L^2} E_x[\sum_{l=1}^{L} \epsilon_l(x)]^2 = \frac{1}{L^2} E_x[\sum_{l=1}^{L} \epsilon_l(x)^2] + \frac{1}{L^2} E_x[\sum_{1\le i\neq j\le L}^{L} \epsilon_i(x)\epsilon_j(x)] \\
&= \frac{1}{L^2}\sum_{l=1}^{L} E_x[\epsilon_l(x)^2] = \frac{1}{L} E_{av}
\end{aligned}
$$

where, $E_x[\epsilon_i(x)\epsilon_j(x)] = 0$ for $i \neq j$, based on the i.i.d. assumption.

(b) Since we know that

$$
\begin{aligned}
E_{bag} = E_x(\epsilon_{bag}(x)^2) &= E_x\{\big([\frac{1}{L}\sum_{l=1}^{L} f(x) + \epsilon_l(x)] - f(x)\big)^2)\} \\
&= \frac{1}{L^2} E_x[\sum_{l=1}^{L} \epsilon_l(x)]^2 = E_x[\sum_{l=1}^{L} \frac{1}{L}\epsilon_l(x)]^2 \le E_x[\sum_{l=1}^{L} \frac{1}{L}\epsilon_l(x)^2] \\
\Rightarrow E_{bag} &\le E_{av}
\end{aligned}
$$

4 Fully connected neural networks and convolution neural network

extra credit: we test on number of filters, filter size, hidden dim and learning rate. From the result, we find that with higher number of filters, filter size and lower hidden dim the accuracy reach the best.