Scalable Machine Learning and Deep Learning - Review Questions 3

By

Akhil Yerrapragada (akhily@kth.se)

Gibson Chikafa (chikafa@kth.se)

Group Name – gibson_akhil

1 point. Is it OK to initialize all the weights of a neural network to the same value as long as that value is selected randomly using He initialization? Is it okay to initialize the bias terms to 0?

Ans: No, the weights must be initialized with random values to break the symmetry between neurons, else, the neurons will not make any progress with learning and training would serve no purpose.

Yes, it is ok to initialize the bias terms to 0 taking into consideration that there will not be any effect since random weight initialization would be enough to avoid redundancy.

0.5 point. In which cases would you want to use each of the following activation functions: ELU, leaky ReLU, ReLU, tanh, logistic, and softmax?

Ans: If accuracy is concerned: Logistic < tanh < ReLu < leaky ReLu (and its variants) < ELU. Meaning, ELU can be considered in this case.

If runtime performance is concerned, then leaky ReLu works better than ELU.

Softmax activation function is used in the output layer. For a multi class model, it assigns probabilities to each class. T

The output of tanh activation function is zero centered, meaning, it supports backpropagation. It is also used since it achieves better training performance.

logistic activation function is used in the output layer. It is used for predicting probability-based outputs.

0.5 point. What is batch normalization and why does it work?

Ans: Distribution of samples are usually changed in iterations over time due to various reasons. To address this problem, we use batch normalization. It first zero centers and normalizes, then scales and shifts the samples.

Zero centering and normalizing can be implemented using the equation:

$$\hat{\mathbf{x}}^{(i)} = \frac{\mathbf{x}^{(i)} - \mu_B}{\sqrt{\sigma_B^2 + \epsilon}}$$

Scaling and Shifting can be implemented using the equation:

$$\mathbf{z}^{(i)} = \gamma \hat{\mathbf{x}}^{(i)} + \beta$$

The result which is batch normalized ($z^{(i)}$) will be given as an input to the neuron.

---

1 point. Does dropout slow down training? Does it slow down inference (i.e., making predictions on new instances)?

Ans: Yes, dropout slows training by a factor of 2 approximately. With increase in training time, convergence decreases and ultimately leads to a better model.

No, it does not slow down inference as it is applied during training.

---

1 point. Consider a CNN composed of three convolutional layers, each with 3 3 filters, a stride of 2, and SAME padding. The lowest layer outputs 100 feature maps, the middle one outputs 200, and the top one outputs 400. The input images are RGB images of 200 300 pixels. What is the total number of parameters w in the CNN?

Ans:

Given,

3 convolutional layers,
3*3 Filters,
Stride of 2
SAME padding

The Lowest Layer outputs 100 feature maps:

For one feature map we get, Weights: 3*3*3*1 = 27 + 1 bias = 28

Considering 100 feature maps, we get 100 * 28 = 2800 <Equation 1>

The Middle Layer outputs 200 feature maps:

For 100 feature maps as an input from previous layer we get, Weights: 3*3*100 = 900 + 1 bias = 901

Considering 200 feature maps, we get 200 * 901 = 180200 <Equation 2>

The Top Layer outputs 400 feature maps:

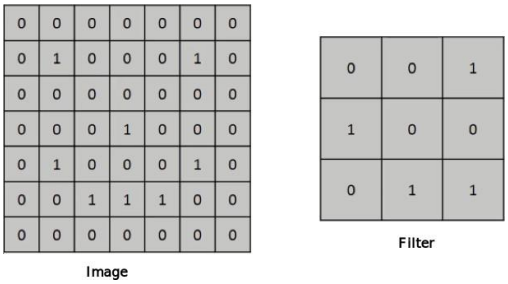
For 200 feature maps as an input from previous layer, we get, Weights: 3*3*200 = 1800 + 1 bias = 1801

Considering 400 feature maps, we get 400 * 1801 = 720400 <Equation 3>

Adding Equations 1, 2, 3 we get:

Total number of parameters (w) = 2800 + 180200 + 720400 = 903400

1 point. Consider a CNN with one convolutional layer, in which it has a 3 3 filter (as shown below) and a stride of 2. Please write the output of this layer for the given input image (the left image in the following figure)?



Ans:

0	**0**	**0**
1	**0**	**1**
0	**1**	**1**