Convolutional Neural Networks Optimized by Logistic Regression Model

Bo Yang, Zuopeng Zhao, Xinzheng Xu

Convolutional Neural Networks Optimized By Logistic Regression Model

Bo Yang, Zuopeng Zhao, Xinzheng Xu

School of Computer Science and Technology,China University of Mining and Technology
Xuzhou 221116, China
`xuxinzh@163.com`

Abstract. In recent years, convolutional neural networks have been widely used, especially in the field of large scale image processing. This paper mainly introduces the application of two kinds of logistic regression classifier in the convolutional neural network. The first classifier is a logistic regression classifier, which is a classifier for two classification problems, but it can also be used for multi-classification problems. The second kind of classifier is a multi-classification logistic regression classifier, also known as softmax regression classifier. Two kinds of classifiers have achieved good results in MNIST handwritten digit recognition.

Keywords: Convolution neural network, Logistic regression, Softmax regression

1 Introduction

In recent years, since the convolutional neural network is proposed [1], it is widely used in pattern recognition [5], image processing, especially achieved good results in the large field of image processing [6].This paper [9] makes a detailed theoretical analysis of the convolution neural network, then the various classification algorithms and models have been proposed. In the paper [4] proposed the multilayer perceptron as a convolutional neural network classifier, the paper [4] also used k nearest neighbor algorithm as a convolutional neural network classifier, the paper [13] used support vector machine (SVM) as the convolutional neural network classifier, both of them have achieved good results in the handwritten numeral recognition experiment. The paper [12] mainly introduces the model of linear regression, logistic regression and so on. The paper [15] introduces the softmax regression model, and the detailed formula derivation of the algorithm combined with the back propagation algorithm.

The structure of this paper is divided into five parts. The second part mainly introduces the convolutional neural network structure. The third part mainly introduces two kinds of classification model. The first is a logistic regression model and how to use logistic regression to solve multi classification problem, on the other is the Soft regression model. The forth part of the thesis is the experiment and the result analysis. The last part of the paper is the summary.

2 Structure of convolutional neural network

Convolutional neural network can be functionally divided into two parts, one is the image feature extraction section, the other part of the classifier. In our experiment, convolutional neural network in the structure can be divided into seven layers including an input layer, convolutional layer C1, sub-sampling layer S2, convolutional layer C3, sub-sampling layer S4, sub-sampling layer S4 unfolded layer (not included in the number of layers), fully connected layer and output layer. The feature extraction part includes C1, S2, C3, S4 layer, the classifier part includes the sub-sampling layer S4 expansion layer (also can be used as the input layer of the classifier), fully connected layer and output layer.

Convolutional neural network structure has many variants, this article uses a classic convolution neural network. In this paper, we mainly introduce the application of two kinds of logistic regression classifiers in the convolutional neural network, so we simply introduce the convolution neural network. We can more detailed understanding of convolutional neural networks in this classic paper [9]. Figure 1 is the structure of the convolution neural network used in this paper. This paper carries out a handwritten numeral recognition experiment on the MNIST data set.
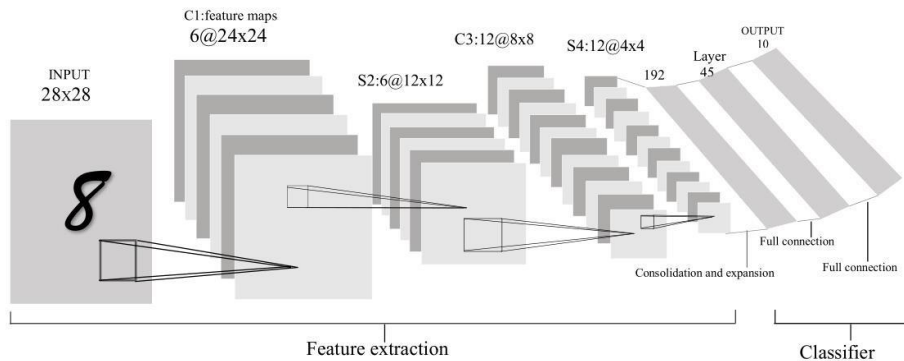

Fig. 1. Network structure of convolutional neural network

3 Classifier model based on logistic regression

Logistic regression is a kind of classifier for two classification problems. If it is used for multi classification problem, we need to train a logistic regression classifier for each category. The Softmax regression classifier is applied to solve the multi classification problem. The image feature extracted from the convolutional neural network is used as the input layer of the multi-layer neural network, and the classifier is used to classify the image, and the back propagation algorithm is used to update the weight parameters in the whole training process. Next, we will introduce these two kinds of classifiers in detail.

3.1 Logistic Regression Model

The two classification problem is divided into C_1 and C_2 two categories, then the output layer has only one neuron and its output is y. Assuming that the number of weight parameters of the output layer is n, then $w^L = [w_1, w_2, \dots, w_n]$. Since the output value y is processed by the sigmoid function, and the output range of the sigmoid function is 0 to 1, then we can put the output value y as the probability value which the input vector x^1 belongs to the first category C_1. Conversely, using one minus the output y gives the probability value that the input vector x^1 belongs to the second category C_2.

$$p(C_1|x^1) = y \tag{1}$$

$$p(C_2|x^1) = 1 - y \tag{2}$$

For the two classification problem with N training samples, the cost function E can be expressed as:

$$E = \frac{1}{N}\sum_{i=1}^{N}\left[-z^{(i)}\log\left(y^{(i)}\right) - (1 - z^{(i)})\log(1 - y^{(i)})\right] \tag{3}$$

If we use logistic regression to solve the multi classification problem, we need to train a logistic regression classifier for each category. For the K classification problem, then the output layer has K neural units, the use of logistic regression to solve the problem of K classification need to train K logical regression classifier. For a specific category, the rest of the category as a category, then it becomes a two classification problem. For the K classification problem with N training samples, the cost function E can be expressed as:

$$E = \frac{1}{N}\sum_{i=1}^{N}\sum_{k=1}^{K}\left[-Z_k^{(i)}\log\left(y_k^{(i)}\right) - (1 - Z_k^{(i)})\log(1 - y_k^{(i)})\right] \tag{4}$$

In order to prevent over fitting in the training process of neural network, we can add the regularization in the cost function. Regularization controls the model's complexity and makes the model more generalizable to unseen data. In order to simplify the problem, it is assumed that the neural network model is only the input layer and the output layer. We assume that the number of neural units in the input layer is M, while the output layer has K neural units. The weights between the input layer and the output layer are w, then the cost function is as follows:

$$E = \frac{1}{N}\sum_{i=1}^{N}\sum_{k=1}^{K}\left[-Z_k^{(i)}\log\left(y_k^{(i)}\right) - (1 - Z_k^{(i)})\log(1 - y_k^{(i)})\right] + \frac{\lambda}{2}\sum_{m=1}^{M}\sum_{k=1}^{K}w_{mk}^2 \tag{5}$$

In the above model, we only add the L2 regularization. The w_{mk} represents the connection weights of neural unit m and neural unit k.

3.2 Softmax Regression Model

We need to use the softmax function instead of the activation function in solving the K classification problem and obtain the class probabilities:

$$p(C_k|x^1) = y_k = \frac{e^{u_k}}{\sum_{i=1}^{K} e^{u_i}} \tag{6}$$

This is called as a multinomial logit model。For the target vector z to meet the relationship $\sum_{i=1}^{K} z_i = 1$. Assuming K = 5, if the correct classification of a training sample is C_3, then the target vector should be $z^{(i)} = [0,0,1,0,0]^T$. Then the probability formula of the input vector x^1 and the target output vector z is:

$$p(z|x^1) = \prod_{k=1}^{K} y_k^{z_k} \tag{7}$$

For the multi classification problem with N training samples, the cost function of the Softmax regression model is:

$$E = -\frac{1}{N} \sum_{i=1}^{N} \sum_{k=1}^{K} z_k^{(i)} \log y_k^{(i)} \tag{8}$$

Because the model has the redundant weight parameters, we need to add a weight decay term which penalizes large values of parameters to modify the cost function, our cost function is now:

$$E = -\frac{1}{N} \sum_{i=1}^{N} \sum_{k=1}^{K} z_k^{(i)} \log y_k^{(i)} + \frac{\lambda}{2} \sum_{m=1}^{M} \sum_{k=1}^{K} w_{mk}^2 \tag{9}$$

3.3   Back propagation

The "errors" which we propagate backwards through the network can be thought of as "sensitivities" of each unit with respect to perturbations of the bias [16]. That is to say,

$$\delta = \frac{\partial E}{\partial b} = \frac{\partial E}{\partial u} \frac{\partial u}{\partial b} \tag{10}$$

As a result of $\frac{\partial u}{\partial b} = 1$, the sensitivity of the output layer can be obtained.

$$\delta^L = \frac{\partial E}{\partial u^L} = y^{(i)} - z^{(i)} \tag{11}$$

By deriving the formula, we can obtain the sensitivity of the previous layer as follows:

$$\delta^l = (w^L)^T \delta^L \cdot f'(u^l) \tag{12}$$

Due to the use of the back propagation algorithm, we also need to know the gradient in the process of updating the weight parameters. The gradient formula is as follows:

$$\frac{\partial E}{\partial w^l} = \frac{\partial E}{\partial u^l} \frac{\partial u^l}{\partial w^l} = \delta^l (x^l)^T + \frac{\lambda}{N} w^l \tag{13}$$

$$\Delta w^l = -\eta \frac{\partial E}{\partial w^l} \tag{14}$$

4 Experiment and Result analysis

We have made a good effect on the handwritten digit recognition based on MNIST data set with two kinds of logic regression classifier combined with convolutional neural network. We use three kinds of classifiers to do the contrast experiment, one is based on the logistic regression classifier, one is the softmax regression classifier, and the last one is the multilayer perceptron. We find that whether the classifier uses the hidden layer has a great influence on the convergence speed and the test results. The experimental results of the classifier without using the hidden layer are as follows:

Method Iterations	CNN+MLP	CNN + LRM	CNN + Softmax
1	11.15%	14.96%	5.71%-7.49%
5	4.34%	8.01%	2.67%-3.26%
10	2.73%	5.89%	2.37% -2.75%
50	1.51%	3.24%	1.69%-2.68%
100	1.21%	2.27%	1.13%-1.61%

Table 1. Comparison of experimental results

The experimental results of the classifier using the hidden layer are as follows:

Method Iterations	CNN + MLP	CNN + LRM	CNN + Softmax
1	88.65%	5.46%	7.94%
5	5.17%	1.80%	1.86%
10	3.25%	1.37%	1.44%
20	1.76%	1.09%	1.12%
50	1.23%	0.86%	0.93%
80	1.00%	0.85%	0.94%
100	1.02%	0.74%	0.82%

Table 2. Comparison of experimental results

In this experiment, there are 60000 training samples, each of the 50 samples to update a weight parameter, then an iteration will update the weight 1200 times.

5 Conclusions

In this thesis, we mainly study the use of the classifier based on logistic regression in the convolutional neural network, and introduce two kinds of classifier models based on logistic regression. There are many factors that affect the classifier performance, such as whether to use the hidden layer, the number of hidden layer neurons, and the

learning rate of the regular term and so on. We can improve the convergence speed and test results by increasing the depth of the convolutional neural network. In practical applications, the classifier is selected according to the classification of the data set. In a word, it is a relatively simple and good method to apply logistic regression to convolutional neural network.

References

1. Y. LeCun, B. Boser, J. Denker, D. Henderson, R. Howard, W. Hubbard, and L. Jackel. "Backpropagation applied to handwritten zip code recognition", Neural Computation, 1(4), 1989.
2. B. LeCun, J. Denker, D. Henderson, R. Howard,W. Hubbard, and L. Jackel. Handwritten Digit Recognition with a Back-Propagation Network. NIPS, 1990.
3. Y. LeCun and Y. Bengio, "Convolutional networks for images, speech, and time-series," in The Handbook of Brain Theory and Neural Networks,M. A. Arbib, Ed. Cambridge, MA, USA: MIT Press, 1995.
4. Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. "Gradient-based learning applied to document recognition", Proceedings of the IEEE, vol. 86, pp. 2278–2324, November 1998.
5. M. Bishop. Pattern Recognition and Machine Learning. Springer,2006
6. A. Krizhevsky, I. Sutskever, and G. E. Hinton. ImageNet classification with deep convolutional neural networks. In Proc. of NIPs, pages 1106–1114, 2012.
7. Simonyan K, Zisserman A. Very Deep Convolutional Networks for Large-Scale Image Recognition [J]. Eprint Arxiv, 2014.
8. Y. LeCun, K. Kavukcuoglu, and C. Farabet. Convolutional networks and applications in vision. In Circuits and Systems (ISCAS), Proceedings of 2010 IEEE International Symposium on, pages 253–256. IEEE, 2010
9. J. Bouvrie, Notes on convolutional neural networks, 2006.
10. M O'Neil, Neural network for recognition of handwritten digits.
11. Fabien Lauer, Ching Y. Suen, Gérard Bloch. A trainable feature extractor for handwritten digit recognition. Pattern Recognition, Elsevier, 2007, 40 (6), pp.1816-1824.
12. Eric Vittinghoff, Davis V. Glidden, and Stephen C. Shiboski. Regression methods in biostatistics: linear, logistic, survival, and repeated measures models. Springer New York, 2005.
13. S.Dreiseitl,L.Ohno-Machado. "Logistic regression and artificial neural network classification models: a methodology review", Journal of Biomedical Informatics, 2002, 35(5–6):352-359
14. F.J. Huang and Y. LeCun. "Large-scale Learning with SVM and Convolutional for Generic Object Categorization", In: Proc. 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 1, pp. 284-291, 2006.
15. A. Graves, Supervised sequence labelling with recurrent neural networks, vol. 385, Springer, 2012.