Methods Of Classification Implemented In MNIST Recognition

Zhengyi Ma
Renmin University of China
zymaa@ruc.edu.cn

ABSTRACT

MNIST handwritten digits recognition is a famous task of classification ,which has a training set of 60,000 examples, and a test set of 10,000 examples. We implement six models—Logistic Regression, Support Vector Machine, k-Nearest Neighbor, Convolutional Neural Network, Recurrent Neural Network—on the task of MNIST recognition, and make an analysis of overall performance and time-space efficiency. The source code of this paper can be obtained from https://github.com/zhengyima/mnist-classification

KEYWORDS

classification, logistic regression, SVM, nerual networks, pattern recognition

ACM Reference Format:
Zhengyi Ma. 2018. Methods Of Classification Implemented In MNIST Recognition. In *Woodstock '18: ACM Symposium on Neural Gaze Detection, June 03–05, 2018, Woodstock, NY*. ACM, New York, NY, USA, 3 pages. https://doi.org/10.1145/1122445.1122456

1 INTRODUCTION

MNIST is a famous database of handwritten digits, and usually used for a entrance instances in many books about deep learning.[1] It has a training set of 60,000 examples, and a test set of 10,000 examples. Every example of MNIST datasets is a photo of handwritten number ranging from 0 to 9. The size of an example is 28x28, and every example digit will be located in the center of an image. an example digit and its pixel matrix is shown in Figure 1 below.

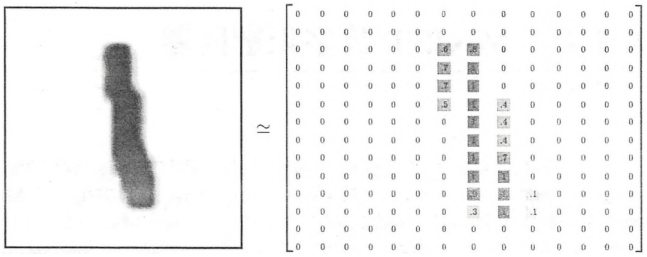

Figure 1: An example of MNIST and its pixel matrix

The MNIST handwritten digits recognition is a typical task of classification. The implemented model should take a feature vector of 28x28 as input and output an integer label ranging from 0 to 9. There are many methods of classification that has been implemented on MNIST and achieved good results. In this paper, we will implement 6 methods of classfication on MNIST, observe the results of these methods and analyze the advantages and disadvantages of them. These methods are Logistic Regression, Multi-Layer Perception (MLP), k-Nearest Neighbor (KNN), Support Vector Machine (SVM), Convolutional neural network (CNN), Recurrent nerual network (RNN).

2 METHODOLOGY

Task Definition

Let $X = \{x_1, x_2, ...\}$ be a set of feature vector of training examples and $Y = \{y_1, y_2, ...\}$ be the label of training examples. Each x_i is a vector of D dimension. In our task, $D = 28 * 28 = 784$. Our task is to provide a y_i given a new example of x_i based on its feature.

Logistic Regression

When deep learning is not popular, logistic regression is the most common method for classification. Logistic regression is a linear method which tries to learn a simple decision boundary for the samples. for every class c, it has a weight vector W_c of which the number of dimensions is the same as the feature vector of a sample. we multiply x with the weight vector W_c to obtain a score for the class c. Then we use softmax to get a normalized score for each class c. Finally we choose the class with the highest score as the predicted label. The equation (1) shows the above process of classification.

Woodstock '18, June 03–05, 2018, Woodstock, NY

ACM ISBN 978-1-4503-9999-9/18/06…$15.00
https://doi.org/10.1145/1122445.1122456

$$p(y|x) = \frac{exp(W_y \cdot x)}{\sum_{c=1}^{C} exp(W_c \cdot x)} \quad (1)$$

During training, we minimize the cross entropy loss function over the full dataset, which is the negative log probability of the right class.

$$J(\theta) = \frac{1}{N} \sum_{i=1}^{N} -log(\frac{e^{f_{y_i}}}{\sum_{c=1}^{C} e^{f_c}}) \quad (2)$$

Support Vector Machine

Like logistic regression, SVM also tries to find a decision boundary to classify the samples. There are many decision boundaries to split the samples, while SVM tries to find the one that has the maximum margin. The margin means the sum of the distances of two support vectors to the boundaries. The support vector is a sample which is close to the decision boundary.

k-Nearest Neighbor

The k-nearest neighbors algorithm (kNN) is a non-parametric method used for classification and regression. Unlike the above methods, it does not need to learn a large number of parameters to build a model. It remembers all the training samples into its memory. During prediction, It calculates the distance of the predicted sample and all the training samples. Then the model will choose k-Nearest neighbor of the predicted sample as the candidates. These neighbors will choose a label as the predicted label by voting.

Multi-Layer Perception

MLP is a multi-layer full-connected nonlinear neural network. At each layer l , there are I_l neurons with a nonlinear function, which is also called activation function. every neuron at layer l ($l > 1$) is a weighted sum of all neurons at layer $l$fk‼1. At training period, we also minimize the cross entropy loss function over the full dataset , like the equation (2).

Convolutional Neural Network

Convolutional neural network is a variant of nerual network, and it is organized by layers of neurons. Unlike MLP, there are only a few neurons conneting to each other between layers. Besides, CMM is also an unlinear model, which usually uses RELU or Tanh as an activate function to learn a more complex decision boundary. A CNN is usually formed as the following 5 modules: input layer, convolutional layer, pooling layer, full-connected layer and softmax layer.

The convolutional layer is the most important layer in CNN. It tries to extract more abstract feature from the former layer using a filter of small size over the whole former input. Taking the image recognition as an example, the convolutional layer will make the feature deeper than the former layer to get more abstract feature. A pooling layer usually appears after the convolutional layer. It does not usually change the depth of the feature, while reduce the amount of the feature to reduce the training parameter of the whole model.

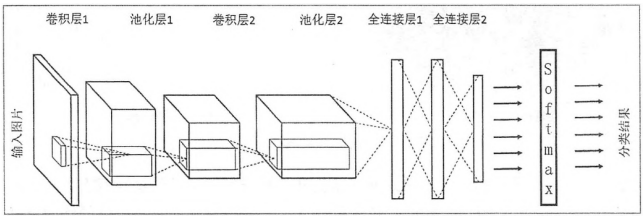

Figure 2: An architecture of CNN for image classification

Because of the modules of convolutional layer and pooling layer, CNN can extract more abstract features of the training data to get a more generalized model. Besides, CNN has a strong ability of Parallelism thanks to the convolutional process. As a result, a CNN model can be trained faster than other nerual network usually.

Recurrent Neural Network

RNN has a strong ability to mine the temporal information of data and model deep semantic information, which helps it to achieve some state-of-the-art performance in speech recognition, languange model, machine translation and Timing analysis. At time step t, RNN will output an output hidden vector h_t according to the current input x_t. The output h_t will be the input at time step $t+1$, which makes the modeling of sequential data a recurrent process.

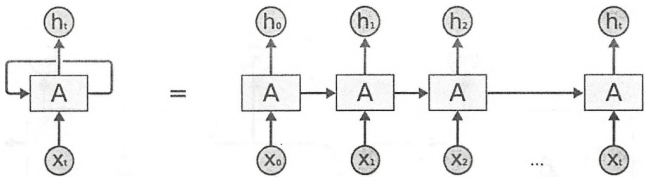

Figure 3: An architecture of RNN

In MNIST task, we will treat each image pixel featrue vector as a sequential data of length D. It will be fed to a RNN to get the final hidden state. We will use the final hidden state vector for classification.

3 EXPERIMENTS

Dataset

We use the official MNIST dataset as our dataset. It has a training set of 60,000 examples, and a test set of 10,000 examples. Every example of MNIST datasets is a photo of handwritten number ranging from 0 to 9. The size of an example is 28x28,

Table 1: Overall Performance

Model	epoch 1	epoch 3	epoch 5	epoch 7	epoch 10	epoch 20
Logistic Regression	.888	.901	.904	.904	.904	.911
SVM	.258	.373	.497	.491	.516	.707
KNN	.962	.962	.962	.962	.962	.962
MLP	.891	.903	.902	.897	.902	.903
CNN	**.989**	**.992**	**.993**	**.996**	**.981**	**.993**
RNN	.952	.973	.981	.973	.976	.959

and every example digit will be located in the center of an image.

Our Models

As described in 2, We will implement 6 models into MNIST recognition task to test their performance. The details of the implement ion of the above models is as below:

Logistic Regression: As introduced in 2.2, we use a weight matrix W_c to calculate the score of every class and use softmax to normalize the score.

Support Vector Machine: We use the Gaussian radial basis function kernal as the kernal function in our implemention.

k-Nearest Neighbor: We choose the odd number ranging from 1 to 30 to observe the effect of different k for our model.

Multi-Layer Perception: We will use a MLP architecture of 2 layer in our model. We choose RELU as our activate function.

Convolutional Neural Network: We will use a CNN architecture of 2 convulutional-pooling layer in our model. We choose RELU as our activate function.

Recurrent Neural Network: We will use a standard RNN architecture of in our model. We choose RELU as our activate function.

Evaluation Metrics

We use the precision as our evaluation metric for this task, which is the most common evaluation metric in classification.

Overall Performance

We first give the overall results and compare the six models with each other. We also compare the results in different epoch numbers to compare the time efficiency of these models. The overall results are shown in Table 1.

(1) All neural network models outperform other non-neural models significantly. The improvement of neural models confirm the effectiveness of deep learning and its ability to learn more abstract features than other models. Among three neural network models, the CNN model is better than RNN and MLP, which performs the best result on MNIST recognition task.
(2) The result of KNN is not concerned with the iterations of training, and it can achieve comparable results with neural models. We believe the large amount of data can benefit KNN a lot and make it comparable with neural models.
(3) due to the similarity of architecture between MLP and Logistic Regression, the two models achieve similar results with each other.

Efficiency Analysis

Firstly, we will make an time efficiency analysis on the six models mentioned above. Among the six models, CNN can achieve a better result and get a convergence faster than any other model due to its naturally strong ability of parallelism. the training speed of RNN is slower than CNN due to its interval structure. In non-neural models, KNN model is not in need of training of parameters, and can also get a fast training speed. SVM is the slowest model among the six models.

Secondly, we will analyze the space analysis among the six models. KNN is the most space-consuming model among the six models because it need to remember every training samples and their features. And the space consumption makes KNN achieves the fastest time efficiency. In neural world, the space consumption of models depends heavily on the number of layers of neural models. The more layer a models have, The more parameter and more training time the model need.

REFERENCES

[1] Yann LeCun and Corinna Cortes. 2010. MNIST handwritten digit database. http://yann.lecun.com/exdb/mnist/. (2010). http://yann.lecun.com/exdb/mnist/