Digital Handwriting Numbers Identification with different methods

Xiao Weiqi, Zhou Chengyuan, Zhu Chang
Math6380: A Mathematical Introduction to Data Analysis Spring 2017

Introduction

Digital Handwriting Numbers are hard to be identified due to their shapes and colors vary among different people. Its complication impedes the computer to identify. However, with some statistical method we can find the similarity of different digital handwritten number then we can tell what the number is.

In this final assignment we use two methodology to help to identify the handwritten number and improve the result of our prediction. The first one is the Random Forest Classification to accurately classify the image. The second is SVM to optimize and predict the number.

Digital Handwriting Data

Normalized handwritten digits, automatically scanned from envelopes by the U.S. Postal Service. The original scanned digits are binary and of different size and orientations. The images here have been deslanted and size normalized, resulting in 16 *16 grayscale images(Le Cun et al., 1900). There are 7291 training observations and 2007 test observations, distributed as follows:

```
          0    1   2   3   4   5   6   7   8   9 Total
Train 1194 1005 731 658 652 556 664 645 542 644 7291
 Test  359  264 198 166 200 160 170 147 166 177 2007

or as proportions:
          0    1   2    3    4    5    6    7    8    9
Train 0.16 0.14 0.1 0.09 0.09 0.08 0.09 0.09 0.07 0.09
 Test 0.18 0.13 0.1 0.08 0.10 0.08 0.08 0.07 0.08 0.09
```

Methodology

➢ Random Forest Classification

Why?

To lower variance of the prediction results from decision trees,
we use bagging method to improve prediction accuracy at the expense of interpretability.

How?

Procedure of random forest:

- Data $(y_i, x_i), i = 1,2,\ldots,n$; and a learning method $\hat{f}$.
- Draw a boostrap sample from the data, and compute a $\widehat{f_1^*}$ based on this set of bootstrap sample.
- Draw another boostrap sample from the data, and compute a $\widehat{f_2^*}$ based on this set of bootstrap sample.
- ….
- Repeat M times, obtain $\widehat{f_1^*}, \ldots, \widehat{f_M^*}$.
- Produce the learning method with bagging as $\frac{1}{M}\sum_{j=1}^{M}\widehat{f_j^*}$.

➢ Support Vector Machine

Why?

SVM is based on minimizing structured risk so that it can solve over-learning problem. It is a convex optimization problem so local optimization is the global optimization.

How?

Procedure of SVMC:

- SVM Classification aims to solve the following problem

$$\min_{(a,a_0)\in R^{d+1}} \frac{1}{n}\sum_{i=1}^{n} l_{hinge}\big(y_i, f(x_i)\big) + \lambda\|a\|^2$$

- Use Lagrangian to solve it and then we reach the dual SVMC problem

$$(Dual)$$

$$\max_{\hat{\alpha}} \sum_{i} \hat{\alpha}_i - \frac{1}{2}\sum_{i,j} \hat{\alpha}_i y_i \hat{\alpha}_j y_j k(x_i, x_{j)}$$

$$s.t. \sum_{i} \hat{\alpha}_i\, y_i = 0$$

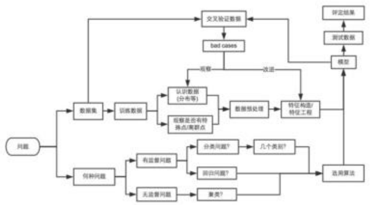
$$0 \le \hat{\alpha}_i \le \frac{1}{2n\lambda} =: \mathrm{C}\,, i = 1, \ldots, n$$

➢ Parameters Tuning

In order to improve the prediction accuracy we try to use grid searching method to test various of different values of parameters in the models. And then use mean of cross validation scores to choose the best parameters – the higher the scores the best the parameters.

Workflow For Digital Handwriting

Take SVM as a demonstration. The following picture is taken from http://blog.csdn.net/chunxiao2008/article/details/50448154

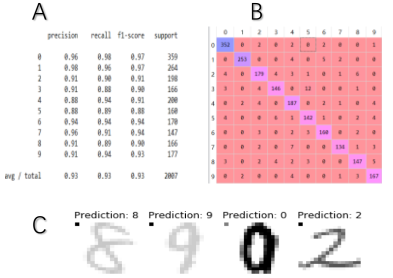


Results

➢ Random Forest Classification

n_estimators	Max_depth	Min_samples_split	Min_samples_leaf	Max_features
180	9	3	2	sqrt

Table 1. Best parameters of random forest classification modeling.


A
B
C
Prediction: 8
Prediction: 9
Prediction: 0
Prediction: 2

	precision	recall	f1-score	support
0	0.96	0.98	0.97	359
1	0.98	0.96	0.97	264
2	0.91	0.90	0.91	198
3	0.91	0.88	0.90	166
4	0.88	0.94	0.91	200
5	0.88	0.89	0.88	160
6	0.94	0.94	0.94	170
7	0.96	0.91	0.94	147
8	0.91	0.89	0.90	166
9	0.91	0.94	0.93	177
avg / total	0.93	0.93	0.93	2007

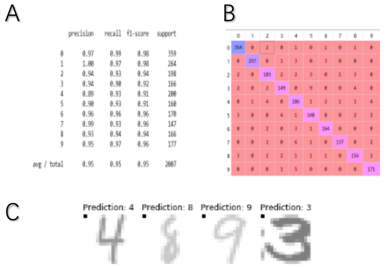
	0	1	2	3	4	5	6	7	8	9
0	352	0	2	0	2	0	2	0	0	1
1	0	253	0	0	4	0	5	2	0	0
2	4	0	179	4	3	1	0	1	6	0
3	3	0	4	146	0	12	0	0	1	0
4	0	2	4	0	187	0	2	1	0	4
5	4	0	0	6	1	142	1	0	2	4
6	0	0	3	0	2	3	160	0	2	0
7	0	0	2	0	7	0	0	134	1	3
8	3	0	2	4	2	3	0	0	147	5
9	0	2	0	0	4	0	0	1	3	167

Figure 1. A. Classification report.
B. Test data prediction metrics. C. Prediction from test data and verification (random picking).
We can see that test data prediction accuracy is 0.93 on average with a little bit low accuracy (0.88,0.89,0.89) for digits 3,5,8 respectively.

➢ Support Vector Machine

Kernel function	Penalty
rbf	10

Table 2. Best parameters of Gradient boosting regression modeling.


A
B
C
Prediction: 4
Prediction: 8
Prediction: 9
Prediction: 3

	precision	recall	f1-score	support
0	0.97	0.99	0.98	359
1	1.00	0.97	0.98	264
2	0.94	0.93	0.94	198
3	0.94	0.90	0.92	166
4	0.89	0.93	0.91	200
5	0.90	0.93	0.91	160
6	0.96	0.96	0.96	170
7	0.99	0.93	0.96	147
8	0.93	0.94	0.94	166
9	0.95	0.97	0.96	177
avg / total	0.95	0.95	0.95	2007

	0	1	2	3	4	5	6	7	8	9
0	354	0	2	0	1	0	1	0	1	0
1	0	257	0	1	3	0	3	0	0	0
2	2	0	185	2	2	3	0	1	3	0
3	2	0	2	149	0	9	0	0	4	0
4	0	1	4	0	186	1	2	1	1	4
5	3	0	0	4	1	148	0	0	2	2
6	0	0	2	0	3	1	164	0	0	0
7	0	0	1	0	6	1	0	137	0	2
8	3	0	1	2	1	1	1	0	156	1
9	0	0	0	1	5	0	0	0	0	171

Figure 2. A. Classification report. B. Test data prediction metrics. C. Prediction from test data and verification (random picking).
We see that test data prediction accuracy is 0.95 on average with all the accuracy above 0.9 for all digits

Conclusion And Drawback

- From these results, we made a conclusion that using SVM to identify hand-written digits is more effective than using random forest. However, using these data(256 inputs) might lead an over-fitting model. It may be better to use dimension reduction method to deal with the data first.

References

- Scikit-learn.org (tutorials of different functions of machine learning and model selection functions.