Training and Predicting a 3T Data Set (splice_site) on a Machine with 120G RAM

Huei-Shin Chen and Meng-Yuan Yang

2015-07-24

1 Data Training with LIBLINEAR-CDBLOCK

With the file being 2.7TB in size, which is usually larger than the RAM of most machines, typical classifiers such as LIBLINEAR (Fan et al., 2008) cannot be used. The reason is that they store the whole data into the memory for training. To solve this issue, we consider LIBLINEAR-CDBLOCK available at `http://www.csie.ntu.edu.tw/~cjlin/libsvmtools/cdblock/`. LIBLINEAR-CDBLOCK (Yu et al., 2012) first splits the training data into pieces so that each can be stored into the memory. Its algorithm requires only one piece of data in the main memory at a time.

1.1 Data Uncompression

The following command uncompresses the training set.

```
$ unxz splice_site.xz
```

It took about 18.5 hours.
To see if the file is correctly downloaded and uncompressed, you can check the md5sum.

```
$ md5sum splice_site
```

It took about 7 hours.
If the md5sum is `df3bd1b65b9df5776907721dff4fdb4e`, the file is correctly uncompressed.

1.2 Data Split

We prepared one machine with around 120G memory for training, so we decided to split the data into 40 pieces.

```
$ ./blocksplit -m 40 splice_site
```

The running time is around 60 hours.

1.3 Training

The following command solves the dual problem of L1-loss support vector machines.

```
$ ./blocktrain -s 3 splice_site.40 splice.s3.model
```

The training process took about 35 hours (127890 seconds) for finishing 5 outer iterations.

The following command solves the dual problem of L2-regularized logistic regression.

```
$ ./blocktrain -s 7 splice_site.40 splice.s7.model
```

The training process took about 15 hours (55359 seconds) for finishing 1 outer iteration.

1.4 Prediction

Later we follow Sonnenburg and Franc (2010) to use AUPRC as the evaluation criterion. The calculation of AUPRC requires decision values, so we modify the code to output them. To calculate the AUPRC, decision values and true labels should be printed out during prediction. predict.c needs to be modified for this.

In predict.c change the following lines to print decision values:

```
 predict_label = predict(model_, x);

 fprintf(output,‘‘%g\n’’,predict_label);
```

into:

```
 double dec_values;

 predict_label = predict_values(model_, x, &dec_values);

 fprintf(output,‘‘%g %g\n’’, target_label, dec_values);
```

The following command predicts instances in the test file splice_site.t, and outputs true labels and predicted decision values.

```
$ ./predict splice_site.t splice.s3.model s3_out

$ ./predict splice_site.t splice.s7.model s7_out
```

It took around 45 minutes.

1.5 Results

We use Matlab Statistics to calculate AUPRC

```
>> load s3_out

>> [Xpr,Ypr,Tpr,AUCpr] = perfcurve(s3_out(:,1), s3_out(:,2), 1, 'xCrit',

  'reca', 'yCrit', 'prec'); AUCpr
```

The AUPRC values are
L1-loss support vector machines (dual): 0.5773
L2-regularized logistic regression (dual): 0.5772
These values are similar to those in Sonnenburg and Franc (2010).

References

R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, and C.-J. Lin. LIBLINEAR: a library for large linear classification. *Journal of Machine Learning Research*, 9:1871–1874, 2008. URL `http://www.csie.ntu.edu.tw/~cjlin/papers/liblinear.pdf`.

S. Sonnenburg and V. Franc. COFFIN : A computational framework for linear SVMs. In *Proceedings of the Twenty Seventh International Conference on Machine Learning (ICML)*, pages 999–1006, 2010.

H.-F. Yu, C.-J. Hsieh, K.-W. Chang, and C.-J. Lin. Large linear classification when data cannot fit in memory. *ACM Transactions on Knowledge Discovery from Data*, 5(4):23:1–23:23, February 2012. URL `http://www.csie.ntu.edu.tw/~cjlin/papers/kdd_disk_decomposition.pdf`.