Assignment 2-Logistic regression in standalone and distributed setting using parameter

Anonymous Authors[1]

Algorithm 1 L2 regularized logistic regression

Input: features x_i, labels y_i
Initalize parameters W size [9655, 50], b size [50]
for $i = 1$ **to** $totalepochs$ **do**
  for $batch$ **in** $batches$ **do**
    loss $=$ **cross-entropy(softmax(**$Wx + b$**))** $+$ $0.01*$**L2(norm(**W**))**
    $W = W - learningrate * grad(loss)$
  end for
end for

1. Preprocessing

- All digits and punctuation were removed from text and text was converted to lowercase
- Words which had a frequency less than 100 were removed from corpus
- 'english' stopwords from NLTK library were removed

Using the above preprocessing techniques we obtained a vocab of size 9655. We use Bag of Words to generate features for each document. Our feature dimension is 9655, and we place 1 for a word if it is present in a document. We have used label dimension of size 50. If a document belongs to multiple classes, we give equal probability for all the classes, otherwise 1.

2. L2 regularized logistic regression standalone

The algorithm used is mentioned in algorithm 1. We use mini-batch SGD for gradient descent. The standalone version was trained in 3 settings as follows:

- Constant learning rate of 0.001 was used
- Exponentially decaying learning rate, with initial learning rate set at 0.01 decaying at a rate of 0.95 after each epoch
- Exponentially decaying learning rate, with initial learning rate set at 0.001 increasing at a rate of 1.05 after each epoch

Trainable Parameters: $9655 * 50 + 50$,
Batch Size: 4096

2.1. Observations

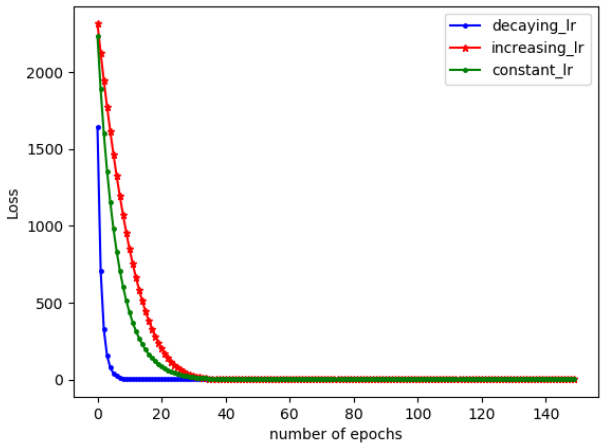

Figure 1. Comparison between various modes of learning

In Figure 1 we observe the variation of training loss with different learning rates. As expected, exponentially decaying algorithm converges first, followed by constant learning rate, followed by exponentially increasing learning rate. The final test accuracy at each setting is given in Table 1

3. L2 Regularized Logistic Regression in Distributed setting

We use Distributed Tensorflow for training and developing models in this section. We use Tensorflow's Monitored

[1]Anonymous Institution, Anonymous City, Anonymous Region, Anonymous Country. Correspondence to: Anonymous Author <anon.email@domain.com>.

Preliminary work. Under review by the International Conference on Machine Learning (ICML). Do not distribute.

Table 1. Classification accuracies for naive Bayes and flexible Bayes on various data sets.

DATA SET	NAIVE	FLEXIBLE	BETTER?
BREAST	95.9± 0.2	96.7± 0.2	√
CLEVELAND	83.3± 0.6	80.0± 0.6	×
GLASS2	61.9± 1.4	83.8± 0.7	√
CREDIT	74.8± 0.5	78.3± 0.6	
HORSE	73.3± 0.9	69.7± 1.0	×
META	67.1± 0.6	76.5± 0.5	√
PIMA	75.1± 0.6	73.9± 0.5	
VEHICLE	44.9± 0.6	61.5± 0.4	√

Training session for training on various worker nodes and parameter servers.

Hyperparameters:
Trainable Parameters: $9655 * 50 + 50$,
Batch Size: 4096,
Learning Rate: 0.001

We train the model in three distributed settings:

- **Asynchronous** with 2 parameter servers, and 2,3,4 worker nodes.
- **Synchronous** with 2 parameter servers, 2 worker nodes
- **Bounded Synchronous (SSP)** with 2 parameter servers, 3 worker nodes, staleness: 10, 15, 20.

3.1. Results

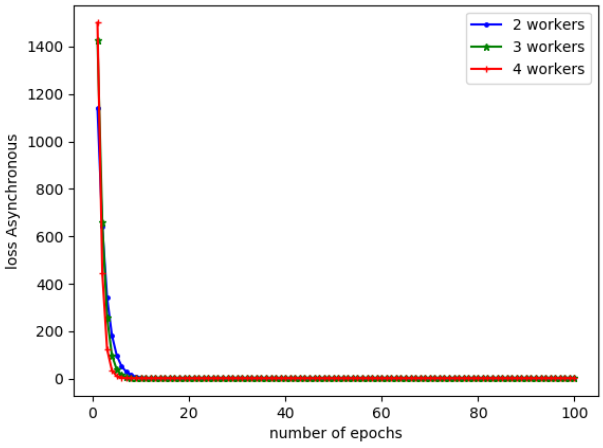

Figure 2. Asynchronous SGD 2, 3, 4 worker comparison

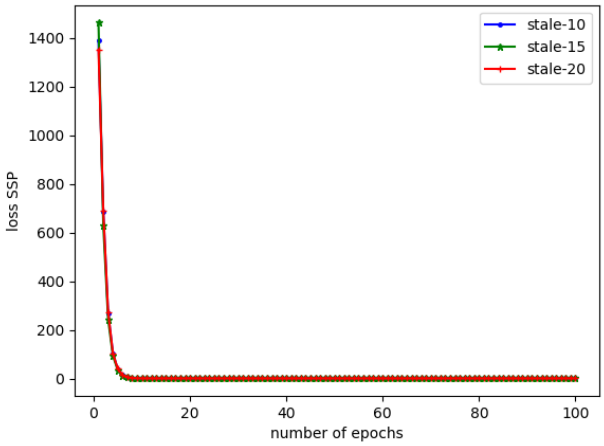

Figure 3. SSP SGD staleness 10, 15, 20

References

Langley, P. Crafting papers on machine learning. In Langley, Pat (ed.), *Proceedings of the 17th International Conference on Machine Learning (ICML 2000)*, pp. 1207–1216, Stanford, CA, 2000. Morgan Kaufmann.

A. Do *not* have an appendix here

Do not put content after the references. Put anything that you might normally include after the references in a separate supplementary file.

We recommend that you build supplementary material in a separate document. If you must create one PDF and cut it up, please be careful to use a tool that doesn't alter the margins, and that doesn't aggressively rewrite the PDF file. pdftk usually works fine.

Please do not use Apple's preview to cut off supplementary material. In previous years it has altered margins, and created headaches at the camera-ready stage.

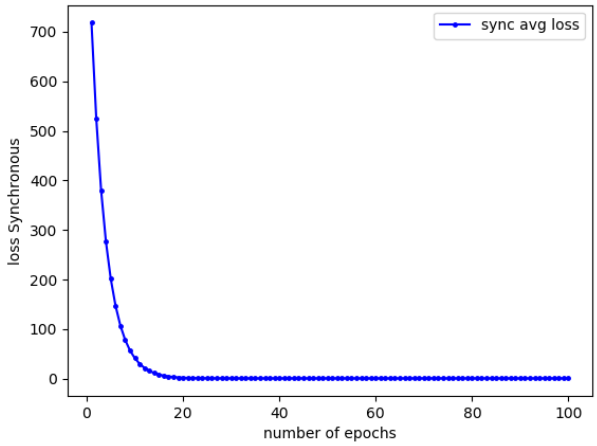

Figure 4. Synchronous SGD

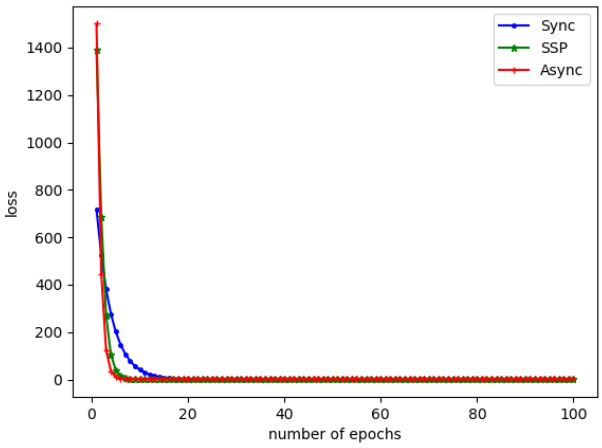

Figure 5. Synchronous Asynchronous SSP comparison