README

Fengzhe Shi

December 20, 2022

1 Introduction

we extend recently proposed GPU based method for execution on large scale clusters using Message Passing Interface (MPI). Our approach requires minimal changes to the TensorFlow runtime - making the proposed implementation generic and readily usable to increasingly large users with accesses to GPU clusters. We evaluate our implementation on a supercomputer and perform kernel Ridge regression with several well known datasets. Our evaluation indicates the efficiency of our proposed implementation.

2 Repository Description

The repository is organized as follows:

- `configs`: a director storing the configurations of experiments.
- `data`: a director storing datasets. However, the dataset is not uploaded in the repository due to its size. They are available at UC Irvine machine learning repository.
- `shs`: a director storing the `SLURM` scripts. Some information are removed due to the sensitivity of the supercomputer.
- `utils`: a director storing scripts that contains fundamental functions:
  - `dataloader.py`: Load and pre-process dataset.
  - `funcs.py`:
  - `integral.py`:
  - `kernels.py`: Provide kernel functions other than the proposed algorithm for comparison.
  - `math_basics.py`: Define math concepts for `funcs.py` and `integral.py`.
  - `parser.py`: Parse the configurations from `configs`.
- `dckrr.py`: a Python script for performing distributed kernel Ridge regression (KRR).
- `bessel.py`: a Python script for performing distributed computation of Bessel function.

Table 1: Test set root mean square error of different kernel methods together with the running times.

Dataset	Linear kernel	Matern by Sklearn	Our method
CT Slices	65 secs 48.12	966 secs 4.03	43.60 secs 8.78
Forest Cover	24 mins 4.85	> 12 hrs -	13 mins 0.97

3 How to Execute

- To run the Bessel function experiment, execute `sbatch shs/bessel.sh`.
- To run the KRR experiment, execute `sbatch shs/krr.sh`.

4 Results

Fig. 1 and Fig. 2 indicates the algorithm to be scalable with large numbers of data. It outperforms the implementation from `Scikit-Learn` and extends its performance with the growth of the number of devices. Tbl. 1 illustrates the efficiency of the method on KRR tasks.

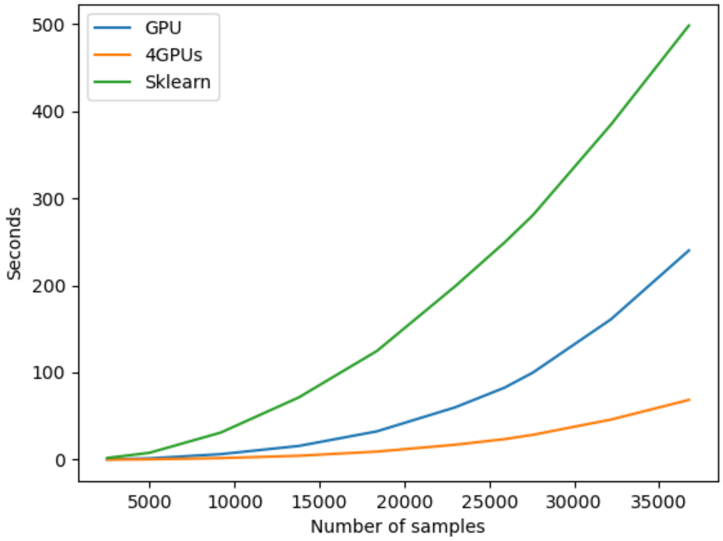

Figure 1: This frog was uploaded via the file-tree menu.

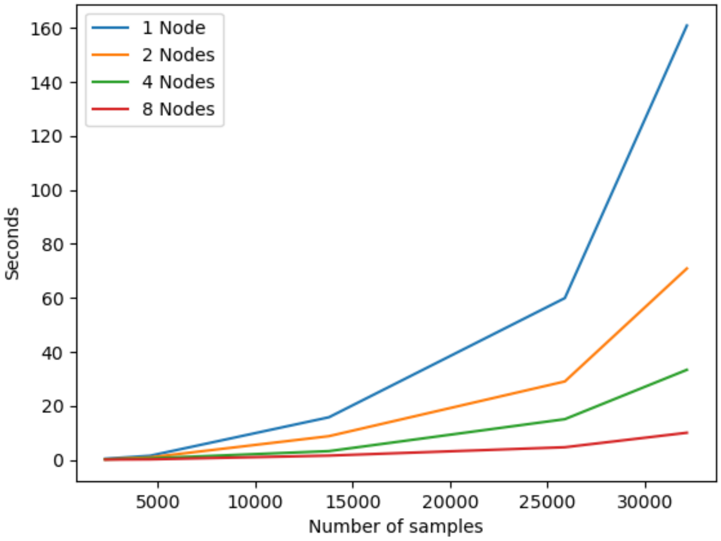

Figure 2: This frog was uploaded via the file-tree menu.