FASTER LINEAR MIXED MODELS (LMM) FOR ONLINE GWAS OMICS ANALYSIS

Prasun Anand [1] and Pjotr Prins [2]

[1] Birla Institute of Technology and Science, Pilani, India, [2] University Medical Center Utrecht, The Netherlands

BACKGROUND

Faster-LMM-D is a lightweight linear mixed-model solver for use in high performance genome-wide association studies (GWAS). Faster-LMM-D can parse data in R/qtl2 format as input.

The new code is designed to handle complex population admixture and large numbers of cofactors and is designed to be sufficiently fast on modern hardware to handle on-the-fly large omics data-sets. This work follows up on FaST-LMM by Lippert and colleagues, and a Python implementation (PyLMM) by Nick Furlotte and colleagues.

PyLMM is also functionally part of GeneNetwork 2. By writing the new version in D, we have already achieved significant speedups and we are adding methods for efficiently using GPU and high-core count CPU hardware, including NVIDIA GPUs and Intel Xeon Phi processors.

METHODOLOGY

Faster-LMM-D is written in D and build on the LDC compiler and uses OpenBLAS, LAPACK, GNU - GSL, ArrayFire, OpenCL, cuBLAS and CUDA® libraries.

D Programming Language

D is a systems programming language with C-like syntax and static typing. It combines efficiency, control and modeling power with safety and programmer productivity.

Multicore Support

Faster-LMM-D is much faster than PyLMM for multiple core CPUs because of immutable data and modern functional coding practices.

GPU Support

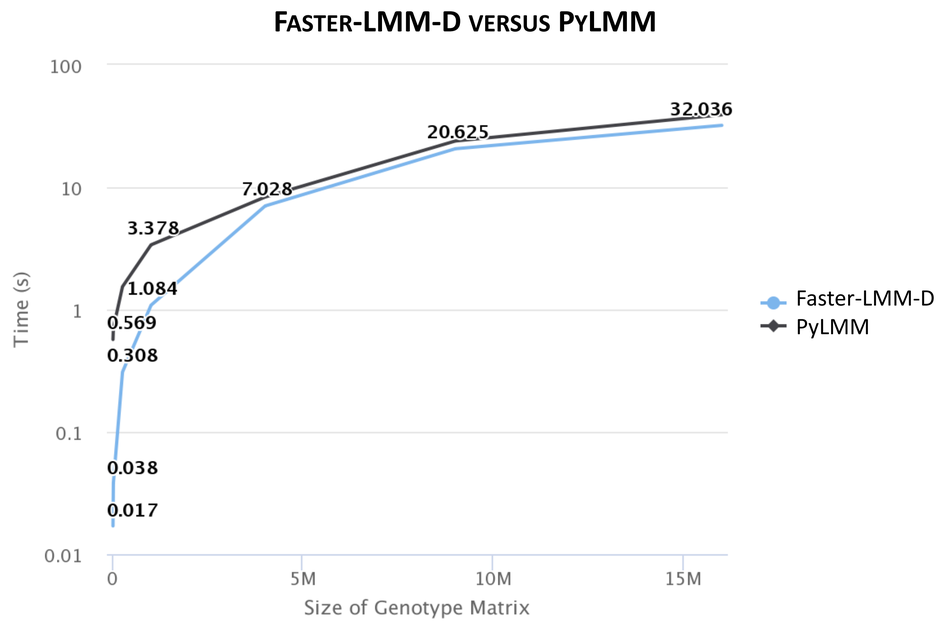
Faster-LMM-D for GPUs uses two computational back ends ArrayFire and CUDA®. Faster-LMM-D using ArrayFire is supported on all major GPU vendors, such as Nvidia, AMD and Intel. ArrayFire uses CUDA® and OpenCL as a dependency that helps it run on all GPU devices by detecting the hardware on runtime. Faster-LMM-D using CUDA® is only available for the Nvidia GPUs. The current implementation for CUDA® backend is faster than PyLMM on CUDA®, and we are still working on optimizing the CUDA® backend.

Performance Optimization

Google Performance Tools or gperftools were used to optimize Faster_LMM_D. The D bindings for ArrayFire, cuBLAS, CUDA® and gperftools were created to build this project.

RESULTS

Faster-LMM-D performs better than PyLMM on multicore CPUs. Faster-LMM-D on a single core CPU is slower than PyLMM because Faster-LMM-D uses a functional style of programming whereas the PyLMM uses an object oriented style of programming. The functional style helps Faster-LMM-D take better advantage of parallelism and hence makes it more suitable for multi-core CPUs.



Performance Guided Optimization

Faster-LMM-D is built on the top of LDC compiler which supports Performance Guided Optimization (PGO/POGO). PGO refers to the optimizations made by a compiler when it is provided with information about a typical execution of the program and uses this information for faster code execution. Building Faster-LMM-D with PGO aids in further improving its speed.

	Computation Time(seconds)	
Size of Genotype Matrix	PyLMM-GN2	Faster-LMM-D
10 X 10	0.569	0.017
100 X 100	0.706	0.038
500 X 500	1.537	0.308
1000 X 1000	3.378	1.084
2000 X 2000	8.314	7.028
3000 X 3000	23.872	20.625
4000 X 4000	39.859	32.036

Future Work

The existing GPU backend for Faster-LMM-D is 3 times faster than PyLMM. We are currently customizing matrix multiplication operations on the GPU to handle matrices larger than GPU RAM. Faster Linear Mixed Models (LMM) for online GWAS omics analysis. We will also introduce lazy parsing of R/qtl2 and tsv files and optimize eigenvector decomposition.

Download

Faster-LMM-D is under active development and would be released in late 2017 or early 2018. Faster-LMM-D is licensed under GPL-3 clause and the source code can be downloaded from: https://github.com/prasunanand/faster_lmm_d

GeneNetwork
University of Tennessee: www.genenetwork.org

REFERENCES

[1] Zachary Sloan, Danny Arends, Karl W. Broman, Arthur Centeno, Nicholas Furlotte, Harm Nijveen, Lei Yan, Xiang Zhou, Robert W. Williams and Pjotr Prins. 2016.“GeneNetwork: framework for web-based genetics”. The Journal of Open Source Software Vol. 1 Number 2. doi: 10.21105/joss.00025

[2] Lippert, C., J. Listgarten, Y. Liu, C. M. Kadie, R. I. Davidson, and D. Heckerman. 2011. “FaST linear mixed models for genome-wide association studies.” Nat Methods 8 (10): 833–35. doi:10.1038/nmeth.1681.