TeraPCA: A fast and scalable method to study genetic variation
in tera-scale genotypes

Aritra Bose1, Vassilis Kalantzi2, Eugenia Kontopoulou1, Mai Elkady1, Peristera Paschou3, Petros Drineas1

1 Department of Computer Science, Purdue University, West Lafayette, IN
2 Department of Computer Science and Engineering, University of Minnesota, Minneapolis, MN
3 Department of Biological Sciences, Purdue University, West Lafayette, IN

*equal contribution

Motivation

- Advances in sequencing and genotyping technology leads to reduction in costs and availability of more data such as recently a clustering of 777,000 individuals were carried out across North America [1]. Publicly available data sets such as UK Biobank, UK10K, GnomAD, etc. are making more data available.
- Principal Components Analysis (PCA) of genotypes is an established approach for population stratification and detecting substructure within populations. As modern datasets breach the TB level of size, “out-of-core” approaches are necessary.
- Current state-of-the-art packages dealing with this problem are FastPCA [2] and flashPCA2 [3], flashPCA2 is a more recent approach which takes the advantage of the implicitly Restated Arnoldi method implemented in the Spectra library [4]. Our goal is to design an approach that is equally fast (or faster) and shares no dependencies to external libraries.
- TeraPCA is a multi-threaded C++ library based on Intel’s MKL library (or any other BLAS and LAPACK external libraries) and combines the robustness of subspace iteration with the power of randomization.

Method

TeraPCA is an out-of-core implementation of the randomized subspace iteration method. It is a two pass procedure where a) the mean vector is computed separately, and b) subspace iteration is applied to the data matrix. For large datasets, each block of rows is fetched and demeaned independently of each other.

Memory and Running Time

We ran our algorithm on a replicated matrix (from HGDP) of size 33,376 individuals and 2.65 ml. SNPs. We select block size of 100 rows and rhs of 30, 60, 90 and 300. For convergence criterion we used the singular value partial sum change between sequential iterations to be ≤ 10⁻³, i.e. at least 3 digits of accuracy.

Qualitative Results

To evaluate the quality of our method, we replicated the HGDP data set to create data sets containing 2K individuals and 1.4 ml. SNPs. TeraPCA computes the top 10 Principal Components (PCs) in 310 secs in comparison to 385 secs taken by flashPCA2 and FastPCA’s 785 secs (approximately).

Quantitative Results

We compared the performance of TeraPCA with flashPCA2, as it performs the best out of the available packages. We used both real and simulated data sets to show that TeraPCA performs better than FlashPCA2 in most of the cases (marked in red) for varying number of threads.

Acknowledgements

This work was supported by NSF BigData 1661760, NSF IIS 1319280 and NSF IIS 1302231.