

# Scalable algorithms for genetic association studies, genotype imputation, and ancestry inference

Benjamin Chu

Graduate program in Biomathematics  
Department of Computational Medicine  
University of California, Los Angeles

University of Southern California (online), January 28, 2021

# Outline

- IHT for association studies
  - Introduction
  - What we did
  - Potential future projects
- Phasing, imputation, and admixture estimation
  - Introduction
  - What we did
  - Potential future projects

## Part 1: Iterative hard thresholding in Genome wide association studies

- **Paper:** [doi.org/10.1093/gigascience/giaa044](https://doi.org/10.1093/gigascience/giaa044)
- **Software:** [github.com/OpenMendel/MendelIHT.jl](https://github.com/OpenMendel/MendelIHT.jl)
- **Documentation:**  
[openmendel.github.io/MendelIHT.jl/latest/](https://openmendel.github.io/MendelIHT.jl/latest/)

# Genome wide association studies

- We want to study genetic basis of a (**usually continuous**) phenotype
  - Examples: Height, developed cancer, number of seeds per plant
- Recruit  $n$  samples, genotype each sample at  $p$  SNPs, producing

$$\mathbf{X}^{n \times p} = \begin{pmatrix} x_{11} & \cdots & x_{1p} \\ \vdots & & \vdots \\ x_{n1} & \cdots & x_{np} \end{pmatrix}, \quad x_{ij} \in \{0, 1, 2\}$$

- Also measure each individual's phenotype

$$\mathbf{y}^{n \times 1} = \begin{pmatrix} y_1 \\ \vdots \\ y_n \end{pmatrix}$$

- Goal: which SNPs are associated with the phenotype?

# Methods for Associations Testing

## 1. Naive model

- $\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \epsilon, \quad \epsilon \sim N(0, \sigma_e^2 \mathbf{I})$
- Method: solve  $\operatorname{argmin}_{\boldsymbol{\beta}} \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2^2$  with solution  $\hat{\boldsymbol{\beta}} = (\mathbf{X}^t \mathbf{X})^{-1} \mathbf{X} \mathbf{y}$
- Comments: There are  $\infty$  solutions since  $p > n$

## 2. Marginal association model

- $\mathbf{y} = \mathbf{x}_j \beta_j + \epsilon, \quad \epsilon \sim N(0, \sigma_e^2 \mathbf{I})$
- Method: hypothesis testing for each SNP:  $H_0 : \beta_j = 0, H_a : \beta_j \neq 0$ 
  - Score test, LRT, Wald test ...etc
  - Reject  $H_0$  if  $p < 5 \times 10^{-8}$
- Comment: Assumes every SNP is independent of each other

## 3. Linear mixed model

- $\mathbf{y} = \mathbf{x}_j \beta_j + \epsilon, \quad \epsilon \sim N(0, \frac{\sigma_g^2}{p} \mathbf{X} \mathbf{X}^t + \sigma_e^2 \mathbf{I})$
- Method: hypothesis testing for each SNP:  $H_0 : \beta_j = 0, H_a : \beta_j \neq 0$
- Comment: Hard to estimate  $\sigma_g, \sigma_e$ . Expensive to form  $\mathbf{X} \mathbf{X}^t$ . Have to calculate determinants and solve huge linear equations. **Current analysis methods scale poorly for non-Gaussian phenotypes.**

## 4. Penalized regression

## Option 4: Penalized regression methods

Minimize the original model  $\ell(\beta) = \|\mathbf{y} - \mathbf{X}\beta\|_2^2$  plus constraints  $p(\beta, \lambda)$ .

- **Lasso:** minimize  $f(\beta) = \ell(\beta) + \lambda \|\beta\|_1$
- **MCP:** minimize  $f(\beta) = \ell(\beta) + \sum_{i=2}^p q(|\beta_i|)$  where

$$q(\beta_i) = \begin{cases} \lambda\beta_i - \beta_i^2/(2\lambda) & 0 \leq \beta_i \leq \gamma\lambda \\ \gamma\lambda^2/2 & \beta_i > \gamma\lambda \end{cases}$$

- **Iterative hard thresholding (IHT):**

$$\text{minimize } \ell(\beta) \quad s.t. \quad \|\beta\|_0 \leq k.$$

Shrinkage of lasso tends to encourage too many false positives. The  $\ell_0$  norm of IHT enforces sparsity without shrinkage.

## IHT background

**Problem:** minimizes  $f(\beta) = \|\mathbf{y} - \mathbf{X}\beta\|_2^2 \quad s.t. \quad \|\beta\|_0 \leq k \in \mathbb{Z}$ .

**Solution:** IHT iterates via

$$\beta_{n+1} = \overbrace{P_{S_k}}^{(3)} \left( \beta_n - \underbrace{s_n}_{(2)} \overbrace{\nabla f(\beta_n)}^{(1)} \right).$$

1. Compute gradient  $\nabla f(\beta) = -\mathbf{X}^t(\mathbf{y} - \mathbf{X}\beta)$
2. Compute step size  $s = \frac{\|\nabla f(\beta)\|^2}{\nabla f(\beta)^t J(\beta) \nabla f(\beta)} > 0$
3. Project to sparsity by setting all but  $k$  largest entries to 0.

# Generalized linear models for non-Gaussian phenotypes

For **non-Gaussian** phenotypes (e.g.  $y_i \in \{0, 1\}$ ) we can model  $y_i$  as generalized linear models (GLM)

$$\mu_i = E(y_i) = g(\mathbf{x}_i^t \boldsymbol{\beta})$$

where  $g$  is a nonlinear inverse link function. The loglikelihood, score (gradient), and expected information (expected negative Hessian) is

$$L(\boldsymbol{\beta}) = \sum_{i=1}^n \left[ \frac{y_i \theta_i - b(\theta_i)}{a(\phi_i)} + c(y_i, \phi_i) \right]$$

$$\nabla L(\boldsymbol{\beta}) = \mathbf{X}^t \mathbf{W} (\mathbf{y} - \boldsymbol{\mu}), \quad \mathbf{W}_{ii} = \frac{1}{var(y_i)} \left( \frac{\partial \mu_i}{\partial \eta_i} \right)$$

$$J(\boldsymbol{\beta}) = \mathbf{X}^t \tilde{\mathbf{W}} \mathbf{X}, \quad \tilde{\mathbf{W}}_{ii} = \frac{1}{var(y_i)} \left( \frac{\partial \mu_i}{\partial \eta_i} \right)^2.$$

These provide all the necessary ingredients for running IHT on GLMs.

## We also did a few more things...

Theoretical:

- Derivation for nearly optimal step size  $s_n$  for each IHT iteration
- Doubly sparse projection for sparsity within and between groups.
- Incorporation of prior weights.
- Block estimation for nuisance parameters

Computational:

- Bit-level vectorized linear algebra for compressed genotype matrices
- Parallel computing for cross validation
- Efficient standardization of genotypes and handling of non-genetic covariates

## Simulation study: IHT vs lasso vs marginal testing

**True Positives:** higher is better. **False Positives:** lower is better

	Normal	Logistic	Poisson	Neg Bin
IHT True Positives	8.84	6.28	7.2	9.0
IHT False Positives	0.02	0.1	1.28	0.98
Lasso True Positives	9.52	8.16	9.28	NA
Lasso False Positives	31.26	45.76	102.24	NA
Marginal True Positives	7.18	5.76	9.04 (5.94)	5.98
Marginal False Positives	0.06	0.02	1527.9 (0.0)	0.0

**Table:** IHT achieves the best balance of false positives and true positives compared to lasso and marginal (single-snp) regression. TP = true positives, FP = false positives. There are  $k = 10$  causal SNPs. Best model size for IHT and lasso were chosen by cross validation. () = zero-inflated Poisson regression.

# Logistic GWAS on UK Biobank hypertension phenotypes

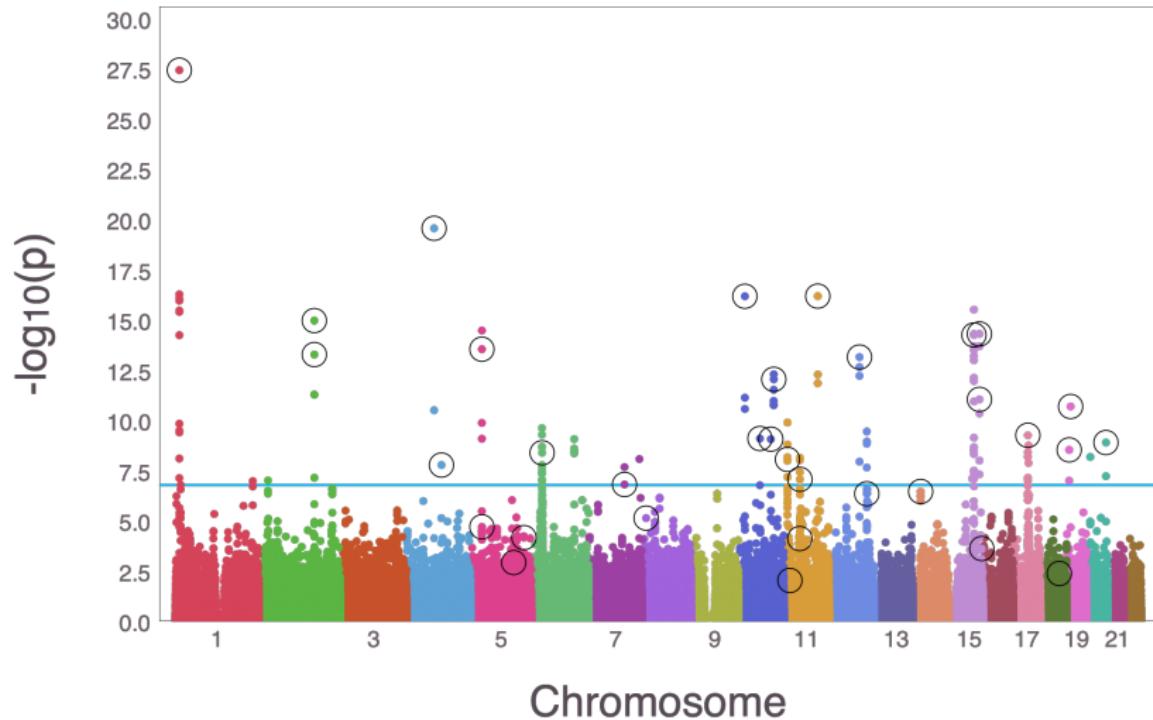


Figure: Manhattan plot comparing a logistic (univariate) GWAS vs logistic IHT on UK Biobank data ( $n \approx 200,000$  and  $p \approx 500,000$ ). Colored dots are  $\log_{10}$  p-values from a logistic GWAS, and the circled dots are SNPs recovered by IHT.

## Criticisms of IHT

1. No rigorous comparison of IHT against the best LMM methods.
  - This will be addressed in our next paper
  - Historically penalized regression struggled to beat simple marginal tests (Alexander & Lange *Genetic Epidemiology* 2011)
2. No follow-up studies for SNPs found in UKBB analysis.
3. Theoretical guarantees fail if predictors become significantly correlated (as with all penalized methods)

## IHT related ongoing and potential future work

1. Multivariate Gaussian IHT - each sample measures  $m$  (potentially correlated) phenotypes  $\mathbf{y}_i^t = [y_{i1}, \dots, y_{im}]$  (**ongoing**)
2. Heritability estimates using IHT:  $\text{var}(\hat{\mathbf{y}})/\text{var}(\mathbf{y})$ . Since SNPs are estimated as mean effects, IHT will likely outperform mixed-model estimates whose SNPs are in the variance (**potential project**)
3. Survival IHT - each sample  $i$  has time-to-event measurement  $y_i$  and indicator  $l_i$  for event (**potential project**)
4. Structured IHT - Each sample's phenotype  $y_i$  take ordered, discrete values. (**potential project**)
5. IHT vs transformed linear mixed model - Compare current IHT to mixed model approach where phenotypes are transformed to normal (**potential project**)

## Part 2: Imputation, phasing, and admixture inference

- **Paper:** [biorxiv.org/content/10.1101/2020.10.24.353755v2](https://www.biorxiv.org/content/10.1101/2020.10.24.353755v2)
- **Software:** [github.com/OpenMendel/MendelImpute.jl](https://github.com/OpenMendel/MendelImpute.jl)
- **Documentation:**  
[openmendel.github.io/MendelImpute.jl/dev/](https://openmendel.github.io/MendelImpute.jl/dev/)

# What is genotype imputation?

- Michigan Imputation Server imputes > 10 million genomes annually
- Purpose: increase number of markers; meta analysis
- Inputs
  - GWAS data with  $\sim 10^6$  unphased genotypes (entries 0, 1, 2)
  - Reference panel with  $10^7 \sim 10^8$  phased genotypes (entries 0, 1)
- Output: Phased genotypes at all markers

The diagram illustrates the process of genotype imputation. It shows a reference panel of haplotypes at the top, followed by a horizontal line representing a chromosome. Below the line is a grid of study genotypes. Brackets on the right side group the reference panel and the study samples.

0	0	1	1	1	0	0	1	1	0	0	0	1	1	1	
0	0	0	0	0	1	1	1	0	1	1	1	0	0	1	
1	1	1	1	1	0	0	0	1	0	0	0	0	0	0	
1	0	1	1	0	0	0	1	1	1	1	1	0	0	1	
1	?	?	?	?	2	?	0	?	?	?	?	0	1	?	1
1	?	?	?	?	1	?	0	?	?	?	?	?	0	?	0
0	?	?	?	?	1	?	1	?	?	?	?	1	0	?	1
1	?	?	?	?	2	?	0	?	?	?	?	0	1	?	1
?	?	?	?	?	2	?	0	?	?	?	?	0	0	?	0
1	?	?	?	?	1	?	1	?	?	?	?	1	0	?	?
0	?	?	?	?	2	?	0	?	?	?	?	0	1	?	1
1	?	?	?	?	1	?	1	?	?	?	?	1	1	?	2

Source: [http://mathgen.stats.ox.ac.uk/impute/impute\\_v2.html](http://mathgen.stats.ox.ac.uk/impute/impute_v2.html)

# Background

- Reference panels are getting **big!**
  - 2012: 1k samples at 28M SNPs (1000 genomes phase 1)
  - 2016: 32k samples at 39M SNPs (Haplotype reference consortium)
  - 2019: 97k samples at 308M sites from sequences (TOPMed)
- Genotype imputation usually employ hidden Markov models (HMM)
  - e.g. Minimac 4, Beagle 5.1, Impute 5
  - All based on Li and Stephens, *Genetics*, 165(4):2213–2233, 2003
  - Very similar accuracy
  - Requires prephasing data = chaining softwares
- HMM methods are slow!
  - They are > 10,000 times faster since version 1 of the initial software, but some problems are still (increasingly) unmanageable.

## Case study: HMM methods are slow

Using Minimac 4 (used on imputation server),

- Imputing 1000 samples<sup>1</sup>, with HRC panel (2016) requires 12,892 seconds  $\approx$  **3.5h** on chromosome 10
- With 23 pairs chromosome and 500k samples in UK Biobank, this imputation will take  $3.5 \times 23 \times 500$  hours  $\approx$  **4.7** years.

Our software MendelImpute.jl

- takes  $\sim$  20 days
- requires no prephasing and much less computer memory
- generates ancestry data automatically
- might lead to better data compression

---

<sup>1</sup>on local computer: 64 GB RAM; Intel i9 9920X CPU with 12 cores

## Imputation within a small genomic region (window)

Consider genotype vector  $\mathbf{x}$  and (unique) reference haplotypes  $\mathbf{h}_1, \dots, \mathbf{h}_d$  in the small genomic window. Imputation is done by minimizing

$$\|\mathbf{x} - \mathbf{h}_i - \mathbf{h}_j\|_2^2 = \|\mathbf{x}\|_2^2 + \|\mathbf{h}_i\|_2^2 + \|\mathbf{h}_j\|_2^2 + 2\mathbf{h}_i^t \mathbf{h}_j - 2\mathbf{x}^t \mathbf{h}_i - 2\mathbf{x}^t \mathbf{h}_j$$

over all  $\mathbf{h}_i, \mathbf{h}_j$  pairs. Solution:

1. Collect  $\|\mathbf{h}_i\|_2^2 + \|\mathbf{h}_j\|_2^2 + 2\mathbf{h}_i^t \mathbf{h}_j$  terms into matrix  $\mathbf{M}$
2. Collect  $-2\mathbf{x}^t \mathbf{h}_i - 2\mathbf{x}^t \mathbf{h}_j$  terms into matrix  $\mathbf{N}$
3. Search for the minimum in the upper triangular matrix  $\mathbf{M} + \mathbf{N}$

where

- $\mathbf{M}, \mathbf{N}$  are efficiently assembled from  $\mathbf{H}^t \mathbf{H}$  and  $\mathbf{X}^t \mathbf{H}$
- Parallelization is achieved by treating different windows independently

## Phasing: extend haplotypes across windows

- Unique haplotypes  $h_i, h_j$  expands to sets of equivalent haplotypes.
- These sets are intersected across windows.
- Eventually intersection becomes empty, representing ancient or contemporary recombination events
- We parallelize phasing over samples, since samples are independent.

Unphased haplotypes

$h_1, h_2, h_3$	$h_1, h_2, h_6$	$h_1, h_3$	$h_4, h_5, h_8$
$h_4, h_5, h_6$	$h_5, h_7, h_8$	$h_2, h_5$	$h_2, h_6$

Phased haplotypes

$h_1$	$h_1$	$h_1$	$h_2, h_6$
$h_5$	$h_5$	$h_5$	$h_5$

Window 1      Window 2      Window 3      Window 4

Switch



## Comparison with HMM methods

<b>1000G chr10</b>	Error Rate	Time (sec)	Memory (GB)
MendelImpute	1.09E-02	39	3.7
Beagle 5.1	5.51E-03	196	9.6
Minimac 4	5.24E-03	728	10.2

<b>1000G chr20</b>	Error Rate	Time (sec)	Memory (GB)
MendelImpute	3.28E-02	13	2.6
Beagle 5.1	1.68E-02	33	4.9
Minimac 4	1.65E-02	159	5.0

<b>HRC chr10</b>	Error Rate	Time (sec)	Memory (GB)
MendelImpute	6.87E-03	154	7.3
Beagle 5.1	1.90E-03	1961	32.4
Minimac 4	1.71E-03	14604	22.5

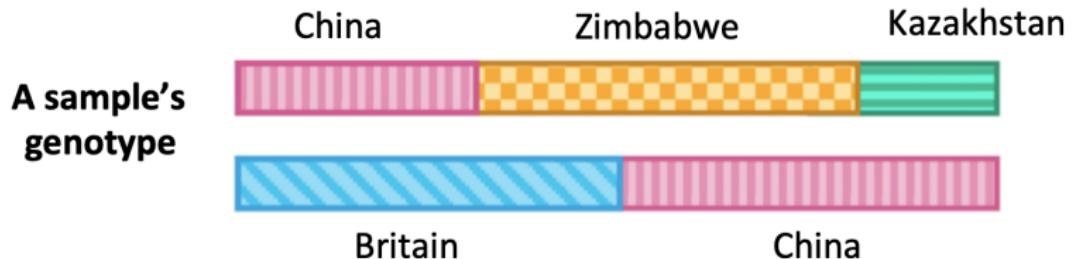
  

<b>HRC chr20</b>	Error Rate	Time (sec)	Memory (GB)
MendelImpute	1.36E-03	133	6.2
Beagle 5.1	5.28E-04	2457	27.4
Minimac 4	6.34E-04	17507	33.2

Conclusion: 10-100x faster, 3-5x more memory efficient, 2-4x less accurate.

## Extension to ancestry inference

- After imputation, each (unphased) genotype is decomposed into a mosaic of haplotype segments from the reference panel.
- If reference samples are labeled with ethnic or country origin, we can visualize local ancestry pattern. We call this **chromosome painting**.



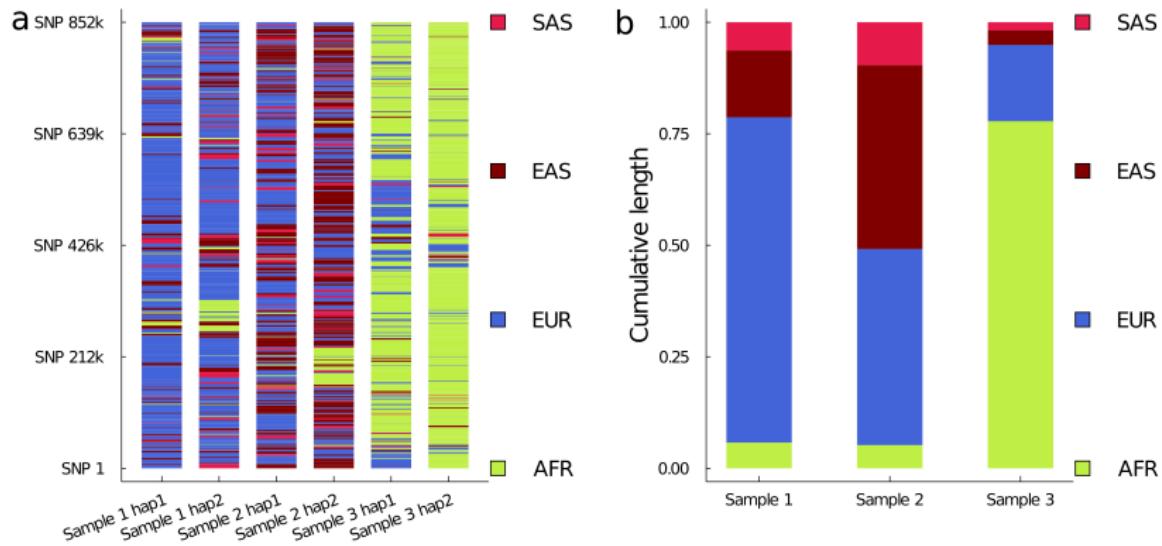
## Test data: 1000 genomes project (v3)

- 2504 samples from 26 populations
- Many admixed, many not (4 grandparents all from local area)
- Does not include Native American populations



Source: <https://gadget.biosci.gatech.edu/learn.html>

# Chromosome 18 painting on PUR, PEL, and ASW



For comparison, their ADMIXTURE estimates are

- PUR (Peurto Rican): 0.164589, 0.136001, 0.647827, 0.051582
- PEL (Peruvian): 0.803278, 0.175677, 1.0e-5, 0.021035
- ASW (African American): 1.0e-5, 0.140135, 1.0e-5, 0.859845

# Potential future work

1. MendelImpute2: Improve error rate
  - How to account for within-window breakpoints?
  - Implement overlapping windows strategy
  - Phasing step should account for ancestry origin
  - Parallel data import
2. Ancestry inference
  - Quantify linkage patterns
  - Promote new use for reference panels: deriving ancestry information
  - Construct different confidence level for different haplotype segments
3. Compare output haplotypes across programs

# Thank you!!

Visit us at: <https://github.com/OpenMendel/>



Figure: Weekly meetings from the OpenMendel group

## References

1. Chu, B. B., Keys, K. L., German, C. A., Zhou, H., Zhou, J. J., Sobel, E. M., ... & Lange, K. (2020). Iterative hard thresholding in genome-wide association studies: Generalized linear models, prior weights, and double sparsity. *GigaScience*, 9(6), giaa044.
2. Chu, B. B., Sobel, E., Wasiolek, R., Sinsheimer, J. S., Zhou, H., & Lange, K. (2020). A Fast Data-Driven Method for Genotyp Imputation, Phasing, and Local Ancestry Inference: MendellImpute. jl. *bioRxiv*.
3. Noah Zaitlen (2018) A Short Tutorial on Linear Mixed Model Association Testing in Genetics.  
<https://www.youtube.com/watch?v=pTAXVTA0YQQ>