Private Computation with Genomic Data for Genome-Wide Association and Linkage Studies

Ali Shahbazi, Fattaneh Bayatbabolghani, and Marina Blanton

Department of Computer Science and Engineering
University of Notre Dame
Computer Science and Engineering Department
University at Buffalo, The State University of New York

3rd International Workshop on Genome Privacy and Security
November 12, 2016

Motivation

- GWAS play a crucial role in medicine and the pharmaceutical industry
- We treat the problem of securing computation associated with GWAS and GWLS
  - ✓ Hardy-Weinberg equilibrium (HWE)
  - ✓ linkage disequilibrium (LD)
  - ✓ Cochran-Armitage test for trend (CATT)
  - ✓ Fisher test
- There is a desire to protect highly sensitive DNA data of users participating in these tests
- We choose a flexible framework for privately computing with genomic data
  - ✓ secure joint computation by multiple entities
  - ✓ secure computation outsourcing to a number of computational servers

Statistical Tests

- **HWE**
  - ✓ is used to estimate the frequency of alleles in a population
  - ✓ is typically performed using chi-squared test

$$\chi^2 = \sum_{i \in \{AA, Aa, aa\}} \frac{(N_i - E_i)^2}{E_i}$$

⋆ E_i's represent expected values of the genotypes, defined as $E_{AA} = (N_A)^2/(4N)$, $E_{Aa} = (N_A N_a)/(2N)$, and $E_{aa} = (N_a)^2/(4N)$

Statistical Tests

- **LD**
  - ✓ occurs when genotypes at two different loci are not independent of each other
  - ✓ is computed by chi-squared for the hypothesis of no disequilibrium

$$\chi^2_{A,B} = \frac{2N(D_{AB})^2}{p_A(1 - p_A)p_B(1 - p_B)} = \frac{2N(D_{AB})^2}{p_A p_a p_B p_b}$$

⋆ D_{AB} is called the coefficient of LD and can be computed as $D_{AB} = p_{AB} - p_A p_B$

- **CATT**
  - ✓ is used to assess the presence of association between a variable with two different categories (cases and controls) and a variable with 3 different categories in application to GWAS

	Group 0	Group 1	Group 2	Total
Controls	N_{00}	N_{01}	N_{02}	R_0
Cases	N_{10}	N_{11}	N_{12}	R_1
Total	C_0	C_1	C_2	N

  - ✓ represents a modification of chi-squared test

$$\chi^2 = \frac{\left(\sum_{i=0}^{2} w_i (N_{0i} R_1 - N_{1i} R_0)\right)^2}{\frac{R_0 R_1}{N}\left(\sum_{i=0}^{2} w_i^2 C_i (N - C_i) - 2\sum_{i=0}^{1}\sum_{j=i+1}^{2} w_i w_j C_i C_j\right)}$$

⋆ $w = (w_0, w_1, w_2)$ corresponds to predetermined weights

- **Fisher test**
  - ✓ is used in the analysis of contingency tables similar to CATT to assess the presence of association between two categories of cases and controls and two groups of A and B alleles in application to GWAS and pharmaceutical drug tests

	A	B	Total
Controls	N_{0A}	N_{0B}	R_0
Cases	N_{1A}	N_{1B}	R_1
Total	C_A	C_B	N

  - ✓ is more accurate than chi-squared tests when sample sizes are small

$$p = \frac{R_0! \cdot R_1! \cdot C_A! \cdot C_B!}{N! \cdot N_{0A}! \cdot N_{0B}! \cdot N_{1A}! \cdot N_{1B}!}$$

⋆ controls correspond to category 0 and cases to category 1

Security Model

- We frame secure computation in a general setting where there are a number of input providers, a number of computational parties, and a number of output recipients
- These three sets of participants can be formed in an arbitrary way
- The focus of this work is on the **semi-honest model**. The techniques that we employ, however, can be extended to support the stronger **malicious model** as well using well-known results

Underlying Techniques

- We build solutions based on **secret sharing**
- (n, t) linear secret sharing:
  - ✓ A secret s is divided into n pieces.
  - ✓ No information will be learned regarding s from t or fewer shares.
  - ✓ With $t + 1$ or more shares, s can be reconstructed.
- We measure performance of secure computation in our framework in terms of interactive operations and rounds since local computation is very fast

Secure Hardy-Weinberg Equilibrium Computation

- We expand the HWE formula and χ^2 is being compared to the threshold τ
- Because the division operation is significantly more expensive that integer multiplication in our framework, we can re-write the formula to replace divisions with multiplications

$$\left(4N \cdot [N_{AA}] - [N_A]^2\right)^2 [N_a]^2 + 2(2N \cdot [N_{Aa}] - [N_A] \cdot [N_a])^2 [N_A] \cdot [N_a] + (4N \cdot [N_{aa}] - [N_a]^2)^2 [N_A]^2 \leq 4N \cdot \tau \cdot [N_A]^2 \cdot [N_a]^2$$

- This can be accomplished in $4\ell + 8$ interactive operations in 6 rounds, where ℓ is the bitlength of the values being compared in previous equation which is proportional to $\log(N)$

Secure Linkage Disequilibrium Computation

- We expand the LD formula and $\chi^2_{A,B}$ is being compared to the threshold τ
- We re-structure the computation to avoid the division operation

$$2N \cdot (N \cdot [N_{AB}] - [N_A] \cdot [N_B])^2 \leq \tau \cdot [N_A] \cdot [N_a] \cdot [N_B] \cdot [N_b]$$

- This can be accomplished in $4\ell + 2$ interactive operations in 5 rounds

Secure Cochran-Armitage Test for Trend Computation

- We expand the CATT formula and χ^2 is being compared to the threshold τ
- We re-structure the computation to avoid the division operation

$$N \cdot ([w_1] \cdot ([N_{01}] \cdot R_1 - [N_{11}] \cdot R_0) + [w_2] \cdot ([N_{02}] \cdot R_1 - [N_{12}] \cdot R_0))^2 \leq R_0 R_1 \tau \cdot ([w_1]^2 \cdot [C_1] \cdot (N - [C_1]) + [w_2]^2 \cdot [C_2] \cdot (N - [C_2]) - 2[w_1] \cdot [w_2] \cdot [C_1] \cdot [C_2])$$

- This can be accomplished in $4\ell + 6$ interactive operations in 5 rounds
- When the weights w_1 and w_2 are public and non-zero, evaluation of previous equation costs $4\ell + 2$ interactive operations in 4 rounds

Secure Fisher Test Computation

- We proceed with computing the logarithm of the p-value instead of directly implementing Fisher test equation
  - ✓ to avoid working with values of excessive bitlength
  - ✓ to replace the division operation with a very fast subtraction operation

$$\log(p) = \log(R_0!) + \log(R_1!) + \log(C_A!) + \log(C_B!) - \log(N!) \\ - \log(N_{0A}!) - \log(N_{0B}!) - \log(N_{1A}!) - \log(N_{1B}!)$$

Secure Fisher Test Computation

- We can simultaneously compute $\log([N_{0A}]!)$ and $\log([N_{0B}]!)$ using one set of R_0 comparisons. Therefore, oblivious computations of $\log(v_A!)$ and $\log(v_b!)$ for some private v_A and v_B

  $[v_A] = [v_B] = 0;$
  for $i = 2, \ldots, R_0 - 1$
  $\quad [c_i] = \mathrm{LTE}(i, [v]);$
  $\quad [v_A] = [v_A] + [c_i] \cdot \log(i);$
  $\quad [v_B] = [v_B] + (1 - [c_i]) \cdot \log(R_0 + 1 - i);$

- Our implementation of securely evaluating $\log(v!)$ for some private v proceeds similar to the computation of a table lookup with a private index
- Our solution has $O(N \log N)$ complexity and $O(\log N)$ round complexity

Performance Results

Test	N	Modulus size	Number M of alleles			
			10	100	1,000	10,000
HWE	200	98	0.042	0.321	3.21	32.5
	400	104	0.046	0.355	3.39	33.9
	800	110	0.047	0.361	3.64	36.3
	1600	116	0.051	0.374	3.87	38.9
LD	200	89	0.037	0.298	2.99	30.6
	400	94	0.040	0.313	3.08	31.9
	800	99	0.042	0.337	3.18	32.1
	1600	104	0.043	0.345	3.37	33.7

Performance Results

Test	N	Modulus size	Number M of alleles			
			10	100	1,000	10,000
CATT with private weights	200	86	0.036	0.297	2.92	29.5
	400	91	0.039	0.295	2.98	30.7
	800	96	0.040	0.319	3.02	31.3
	1600	101	0.045	0.348	3.23	32.6
CATT with public weights	200	86	0.035	0.291	2.86	29.1
	400	91	0.039	0.298	2.99	30.7
	800	96	0.039	0.308	3.07	31.5
	1600	101	0.041	0.340	3.27	32.7
Fisher	100	67	0.108	0.979	9.78	98.1
	200	68	0.217	2.09	20.9	N/A
	400	69	0.453	4.47	44.6	N/A

Q&A