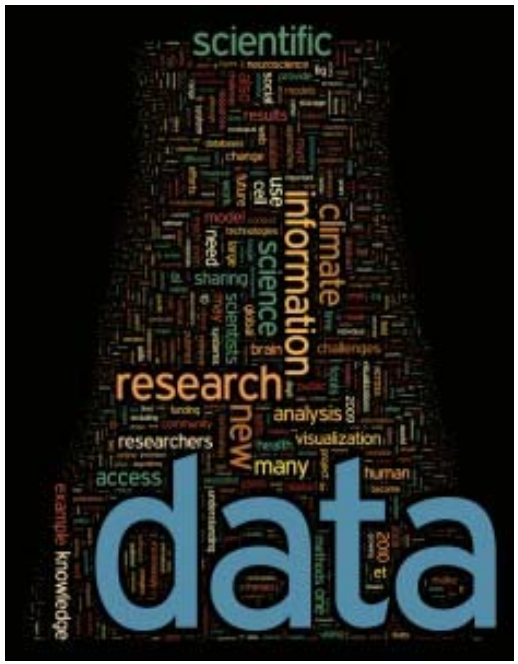
ScAi
SCALABLE ANALYTICS INSTITUTE

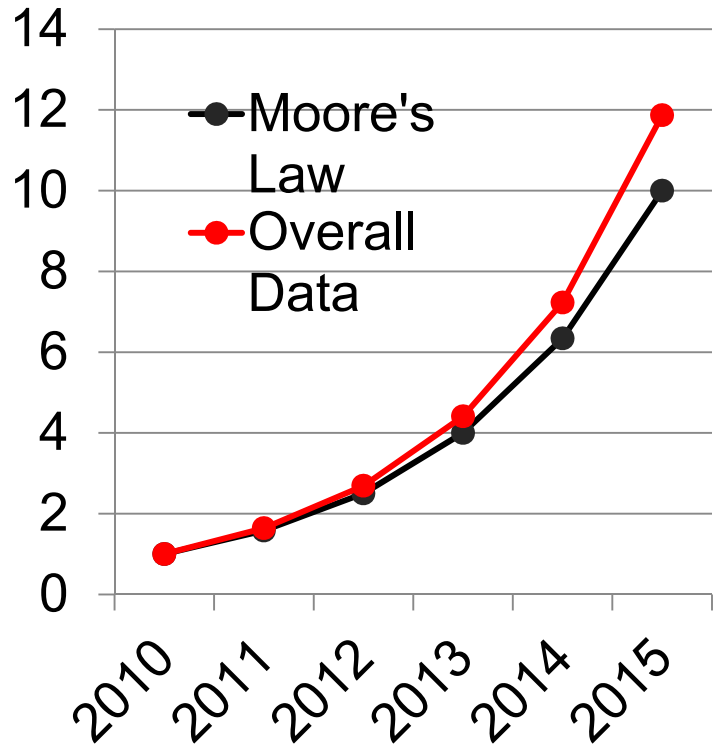
BIG DATA
BIG CHALLENGES



Wei Wang
UCLA

Data is Everywhere

- Easier and cheaper than ever to collect
- Data grows faster than Moore's law



International Data Corporation (IDC) forecasts Big growth for Big Data: market will grow at 40% annual rate


From the dawn of civilization until 2003, humankind generated five exabytes of data. Now we produce five exabytes every two days…and the pace is accelerating.
Eric Schmidt,
Executive Chairman, Google

ScAi
SCALABLE ANALYTICS INSTITUTE

The New Gold Rush

- Everyone wants to extract value from data
  - Big companies & startups alike
- Huge potential
  - Already demonstrated by Google, Facebook, …
- But, untapped by most places
  - “We have lots of data but no one is looking at it!”

ScAi
SCALABLE ANALYTICS INSTITUTE

Extracting Value from Data Hard

- Data is massive, unstructured, and dirty
- Question are complex
  - e.g., Predict the future.
- Processing, analysis tools still in their “infancy”
- Need tools that are
  - Faster
  - More sophisticated
  - Easier to use

Scalable Analytics Institute

scai.cs.ucla.edu

ScAi
SCALABLE ANALYTICS INSTITUTE

ScAi Projects

- Big data systems
- Graph based analytics
- Language design for big data and data streams
- Mining high dimensional data
- User and quality modeling in big data

Predictive Medicine

A video clip from Gattaca (1997), (from youtube)

Detecting Genetic Interactions: Motivation

- Example: Mouse **Colon Cancer**

http://mycanceradvisor.com/

Two important genes

{ ***Ptprj*** (chrom 2)
{ ***Lrig1*** (chrom 6)

Detectible only when studying **interactions**

- True for many common diseases

Detecting Genetic Interactions: Challenges

Statistical – Statistics to capture the interactions

Computational – **Hundreds of billions** of potential interactions

Detecting Genetic Interactions: Previous Approaches

- Exhaustive [Moore et al. '06, Purcell et al. '07]
  - Not scalable
- Heuristic [Carlborg et al.'00, Nakamichi et al.'01]
  - Not optimal
- Two-step [Evans et al. '06, Yang et al. '09]
  - Filter, then search (Not optimal)
- Algorithm development is **in early stage**

Detecting Genetic Interactions: Our Contributions

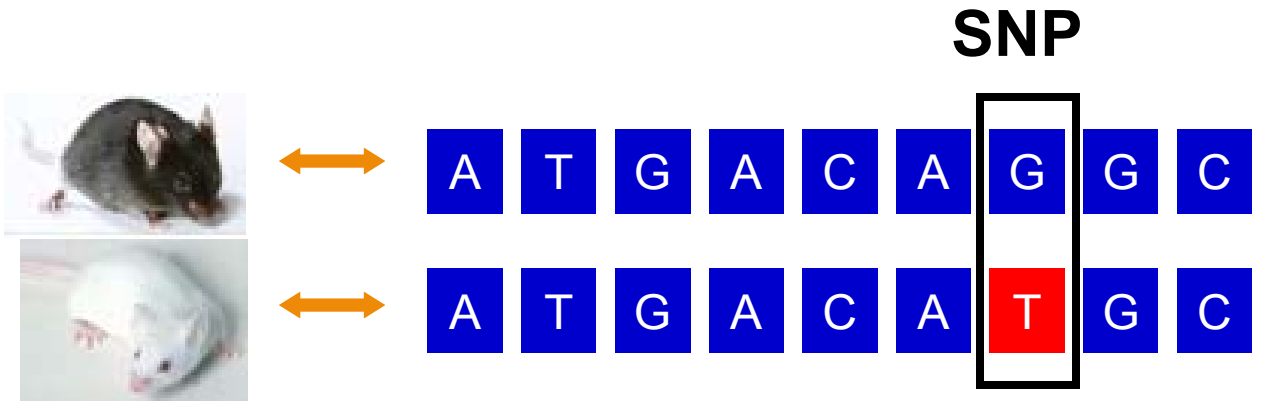
- **Efficiency** and **Optimality**
  - Dramatically reduced the computational burden
  - Guaranteed optimal solution
- **Applicability**
  - A wide range of study types and statistics
- The **first** to address these issues systematically

Outline of the Talk

- Background
  - SNPs and their interactions
  - Computational problems
- Algorithms for Detecting Genetic Interactions

Single Nucleotide Polymorphism: SNP

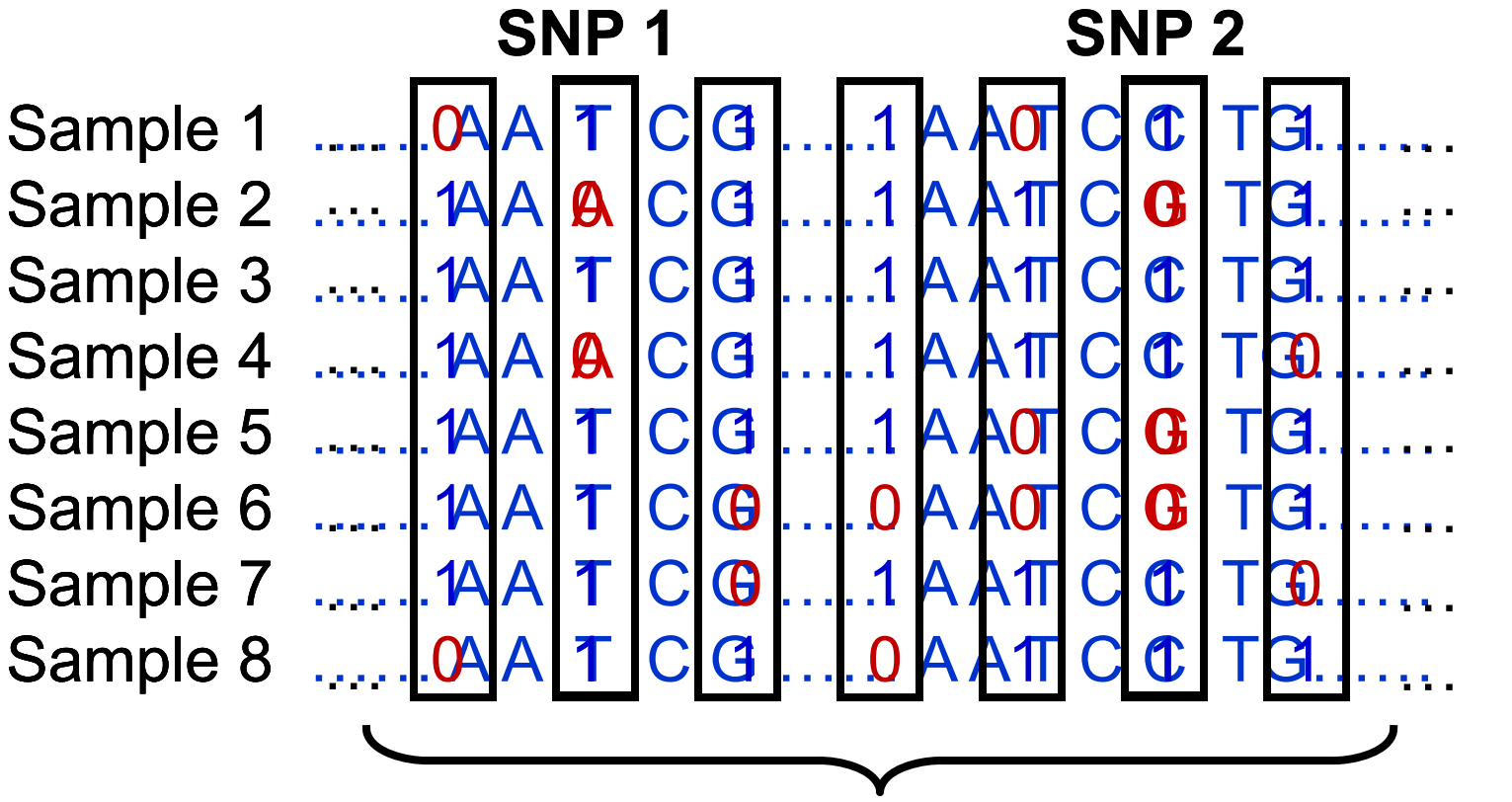
SNP – mutation of a single nucleotide in the DNA sequence



The most common form of genetic variation

Valuable for diagnostics and drug development

SNPs as Binary Variables

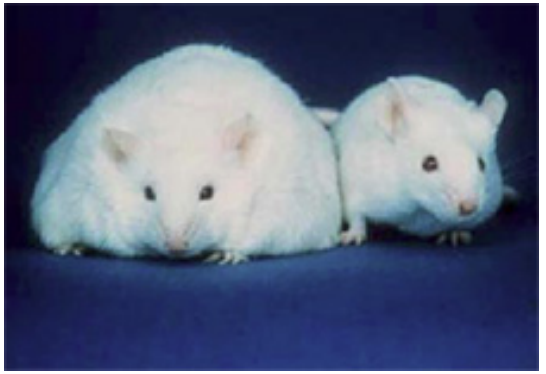


Phenotype Variation

Phenotype – an observable characteristic or trait

http://www.jax.org/

Coat color



http://derc.ucsd.edu

Body weight

SNP-Phenotype Association Study

- Which SNPs cause the phenotype variation?

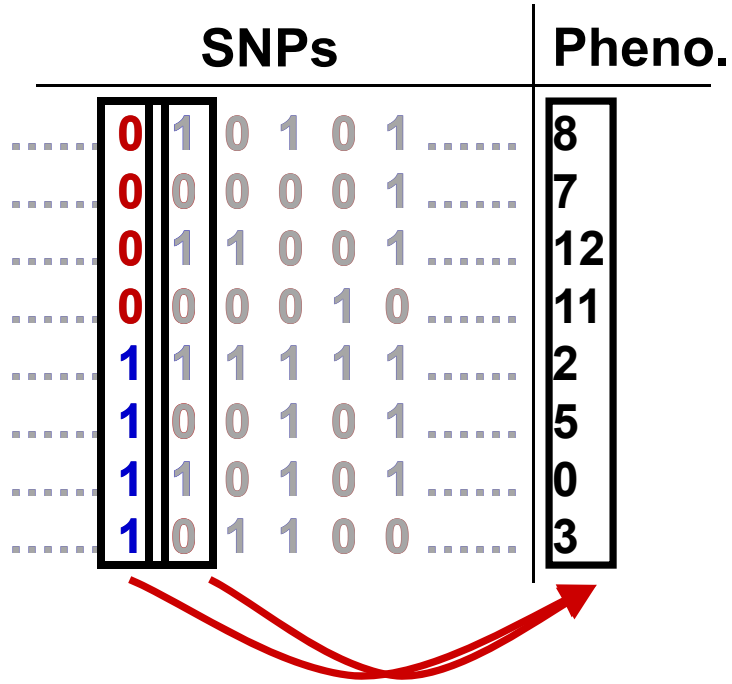
SNPs	Pheno.
..... 0 1 0 1 0 1	8
..... 0 0 0 0 0 1	7
..... 0 1 1 0 0 1	12
..... 0 0 0 0 1 0	11
..... 1 1 1 1 1 1	2
..... 1 0 0 1 0 1	5
..... 1 1 0 1 0 1	0
..... 1 0 1 1 0 0	3

Longstanding goal of genetic studies

Traditional Single-SNP Approach

- For every SNP
- Do a statistical test

Large test value
⇨ Strong association

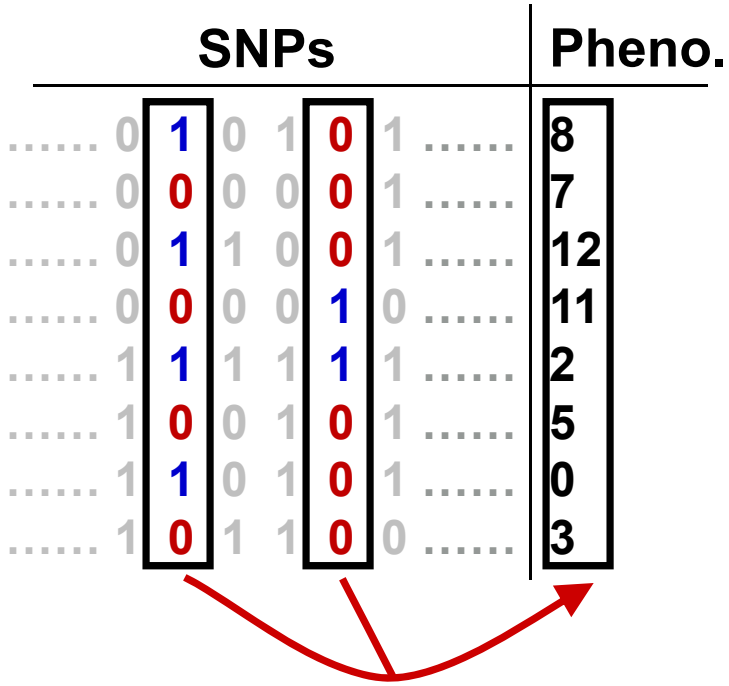


$$T(\text{SNP1}, \text{pheno}) = 28.2$$

$$T(\text{SNP2}, \text{pheno}) = 0.6$$

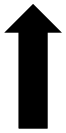
Detecting SNP-SNP Interactions

- Complex phenotypes
  - Diabetes, heart disease, etc …
  - **Joint effect** of genetic factors
- SNP-SNP interactions
  - Test for every **SNP-pair**
- A hot research area in Bioinformatics community [Hoh et al.'03, Hirschhorn et al.'05, Musani et al. '07]



The Computational Problem

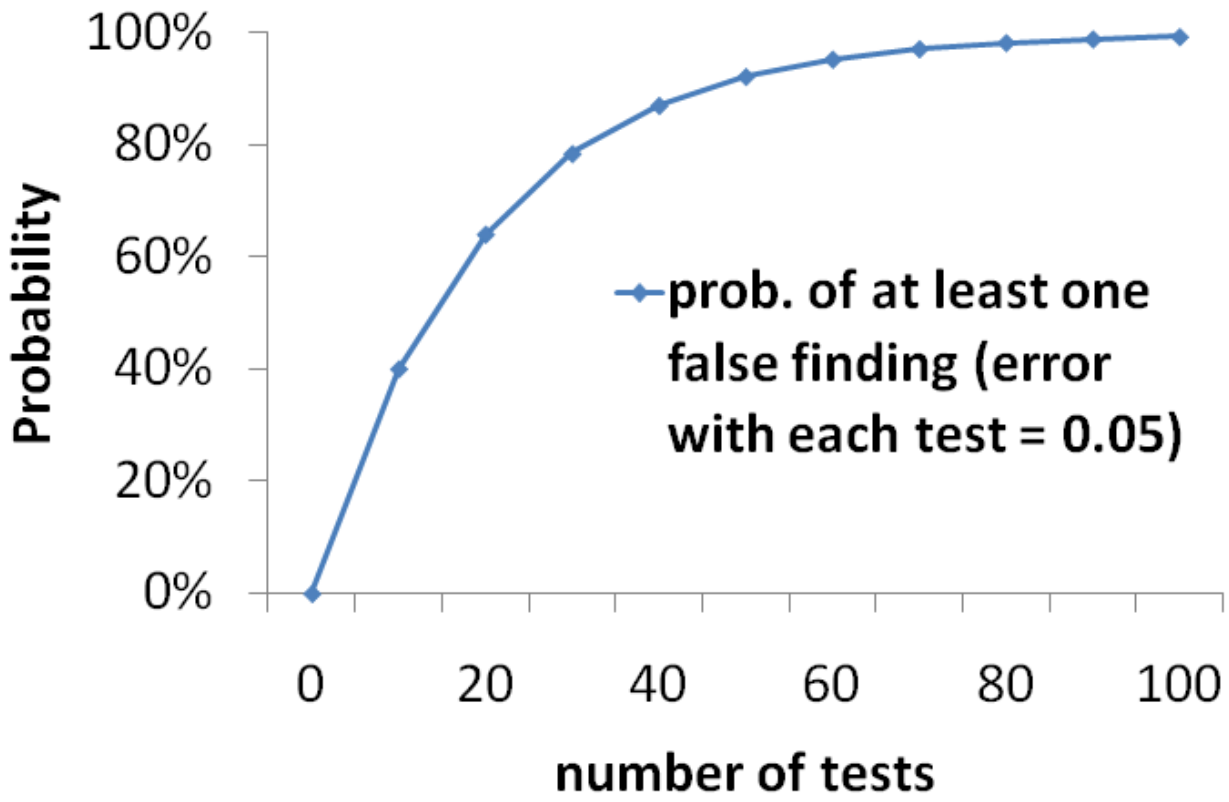
- Problem: Find all SNP-pairs that are **significantly associated** with phenotype



How to define it ?

Multiple Testing Problem

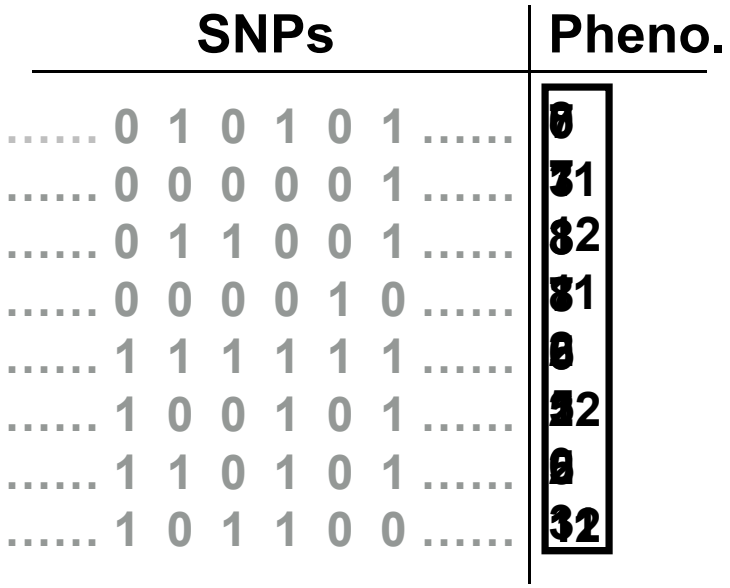
Multiple tests increase the probability of false findings

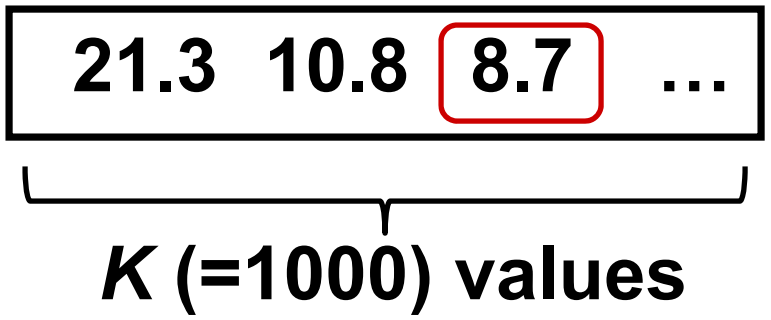


Permutation Test for Error Controlling

Goal: Find a **threshold θ**

- Permute phenotype K(=1000) times
- For each permutation, find max test value
- **Threshold θ** = $\alpha \times K$ -th largest value ($0.01 \leq \alpha \leq 0.05$)
- SNP-pairs $\geq \theta$ are **significant**





The Computational Problem (Revisited)

- Problem 1: Find threshold by **permutation test**
- Problem 2: Find all significant SNP-pairs (≥ $\boldsymbol{\theta}$)
- Brute force: enumerate all SNP-Pairs
- Permutation test is **computationally intensive**

Challenges

- Statistical – effective tests
  - ANOVA, chi-square, likelihood ratio, etc…
- Computational – huge search space
  - 100K SNPs and 1K permutations
    - Number of tests: **500 Billion**
    - Can be easily **MUCH LARGER**
- Must be handled together

Our Solutions

- Efficiency and Optimality
  - **Bound** on test statistic
  - **Indexing** search space for bound estimation
- Applicability
  - Common statistics are **convex**
  - Computing contingency tables

Outline of the Talk

- Background
  - SNP-SNP interactions
  - Computational problem & Challenges
- Detecting SNP-SNP Interactions
  - Algorithms for ANOVA and chi-square tests
  - A general approach COE
  - A more general approach TEAM

FastANOVA – Key Ideas

- **Bound** on test statistic
  - Filter out insignificant SNP-pairs
- **Indexing** structure
  - Compute the bound for **a group of** SNP-pairs
- Removal of redundant computation

The Upper Bound

$$T(\text{SNP pair}, \text{pheno}) \leq \underbrace{\text{constant} + R_1 + R_2}$$

Need to be **≥ *θ*** to be significant

The Upper Bound

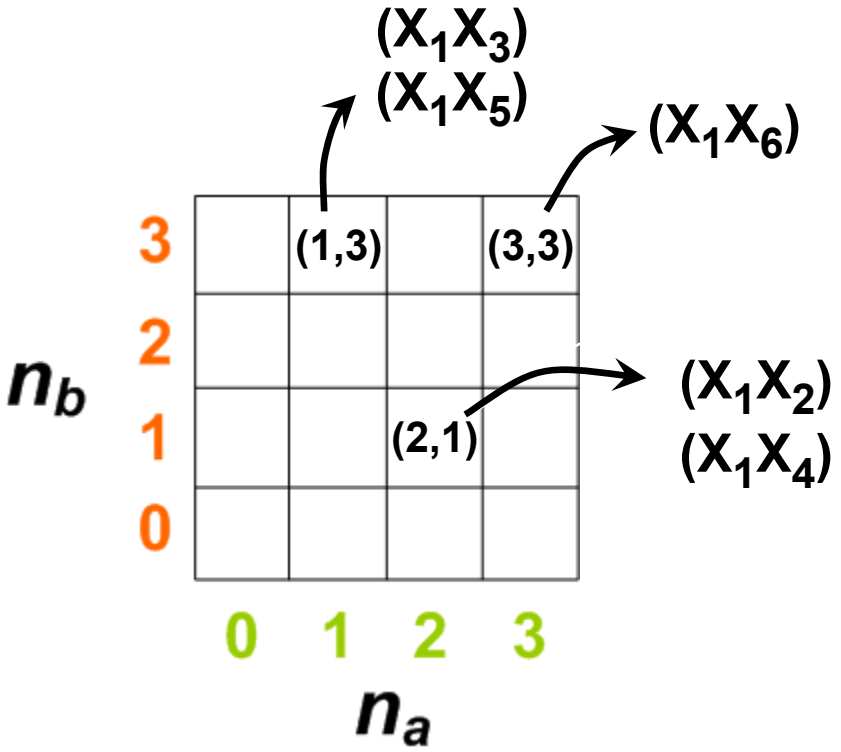
$$\begin{cases} R_1 = f(n_a) \\ R_2 = f(n_b) \end{cases}$$

$$\begin{cases} n_a : \text{min \# \{0,1\} in } \boldsymbol{X_j} \text{ (when } \boldsymbol{X_i} = 0) \\ n_b : \text{min \# \{0,1\} in } \boldsymbol{X_j} \text{ (when } \boldsymbol{X_i} = 1) \end{cases}$$

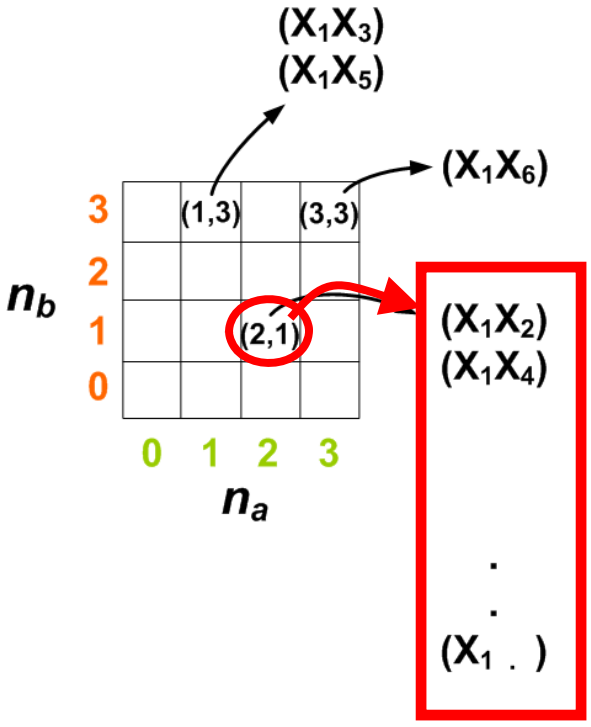
X_i	X_j	
0	0	
0	0	
0	1	
0	1	$n_a = 2$
0	1	
0	1	
1	0	
1	0	
1	1	
1	0	$n_b = 1$
1	0	
1	0	

Indexing SNP-Pairs

X_1	X_2	X_3	X_4	X_5	X_6
0	0	0	1	0	1
0	0	0	0	0	0
0	1	1	0	0	1
0	1	0	0	1	0
0	1	0	1	0	1
0	1	0	0	0	0
1	0	1	1	1	1
1	0	0	0	1	0
1	1	1	1	1	1
1	0	0	1	0	0
1	0	0	1	0	1
1	0	1	1	0	0



Properties of the Indexing Structure



- Many pairs **share** an entry
- Pairs in an entry have the **same upper bound**
- **Built only once,** reused in all permutations

same upper bound

FastANOVA – Overall Process

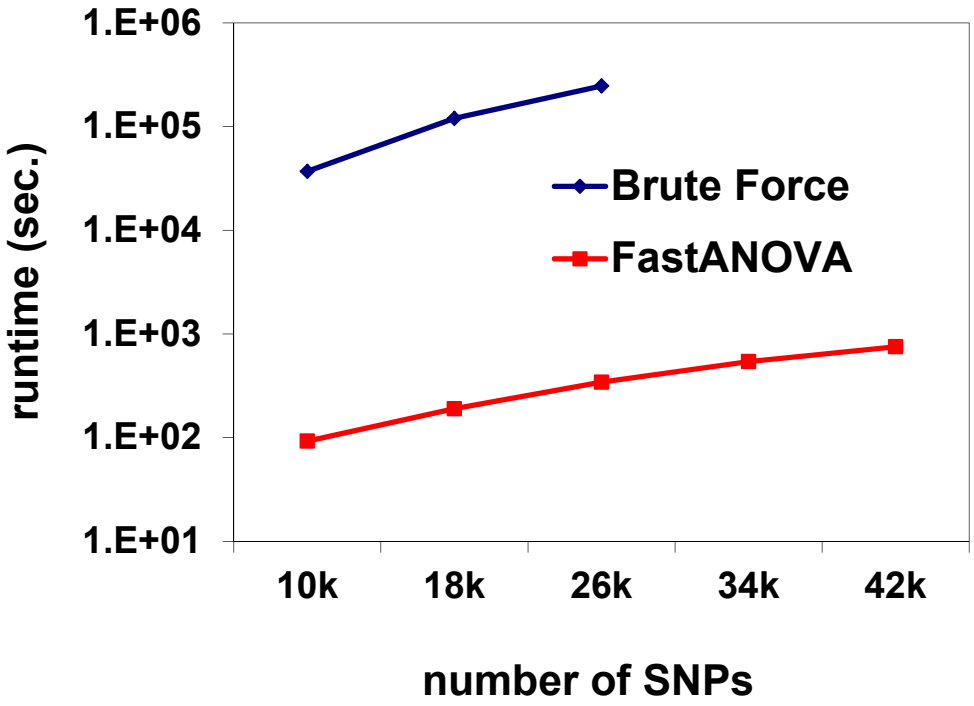
- For each SNP
  - Index its associated pairs
- For each permutation
  - Find the candidate pairs ($ub \geq \theta$)
  - Evaluate test values of the candidates

FastANOVA – Complexity

- Time
  - Brute force: $O(KN^2M)$
  - FastANOVA: $O(N^2M + KNM^2 + CM)$
- Space
  - $O((N+K)M)$

***N* = # SNPs**
***M* = # individuals**
} ***M* << *N***
***K* = # permutations**
***C* = # candidates**

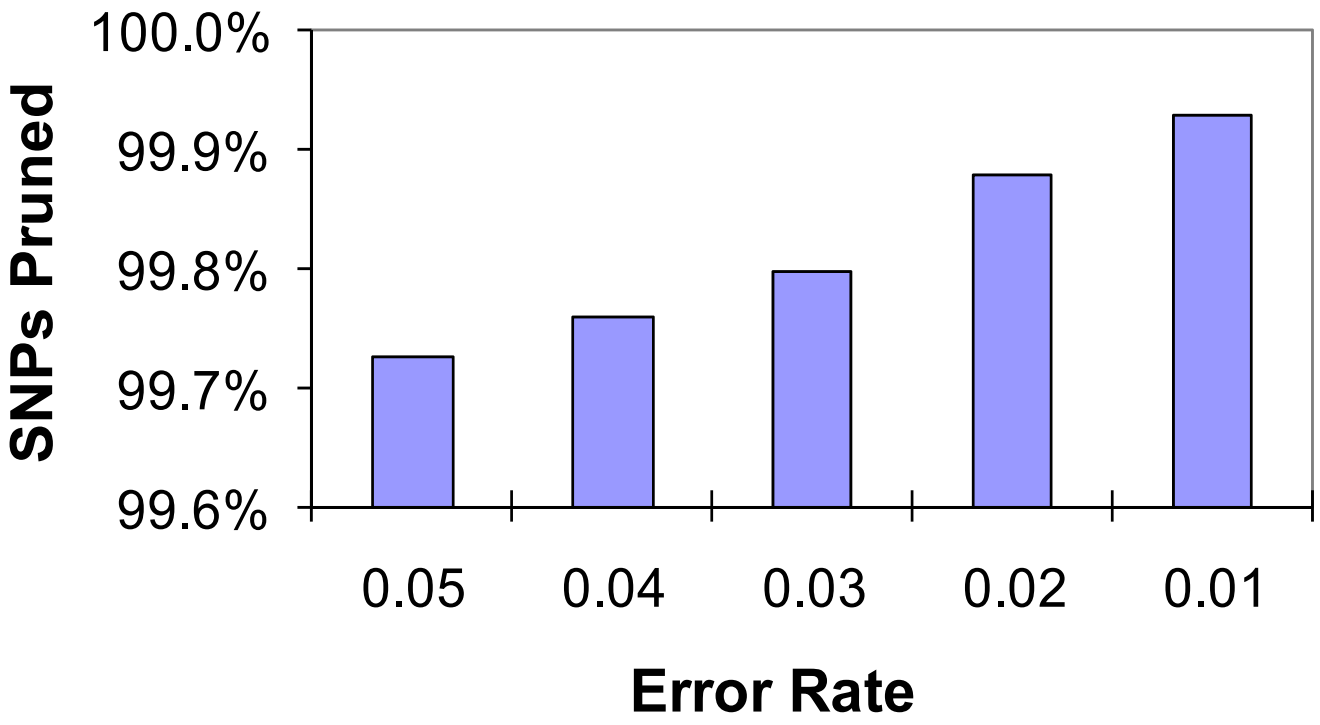
Brute Force v.s. FastANOVA



#SNPs = 44k, #individuals = 26, phenotype: metabolism (water intake)

Data available at http://www.jax.org

Pruning Power of the Bound



The FastChi Algorithm

ANOVA (for quantitative pheno.)

$$T(\text{SNP pair}, \text{pheno}) \leq \text{constant} + \underbrace{R_1 + R_2}$$

(SNP-SNP relation)

Chi-square (for binary pheno.)

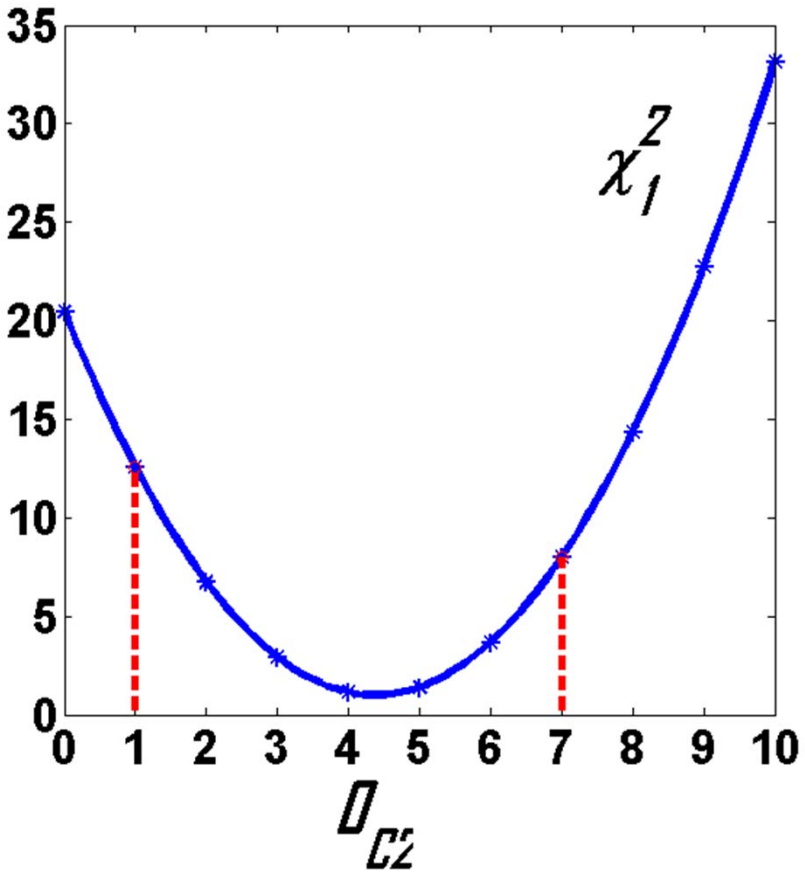
$$T'(\text{SNP pair}, \text{pheno}) \leq \text{constant}' + \underbrace{R_1' + R_2'}$$

(SNP-SNP relation)

COE – A General Approach

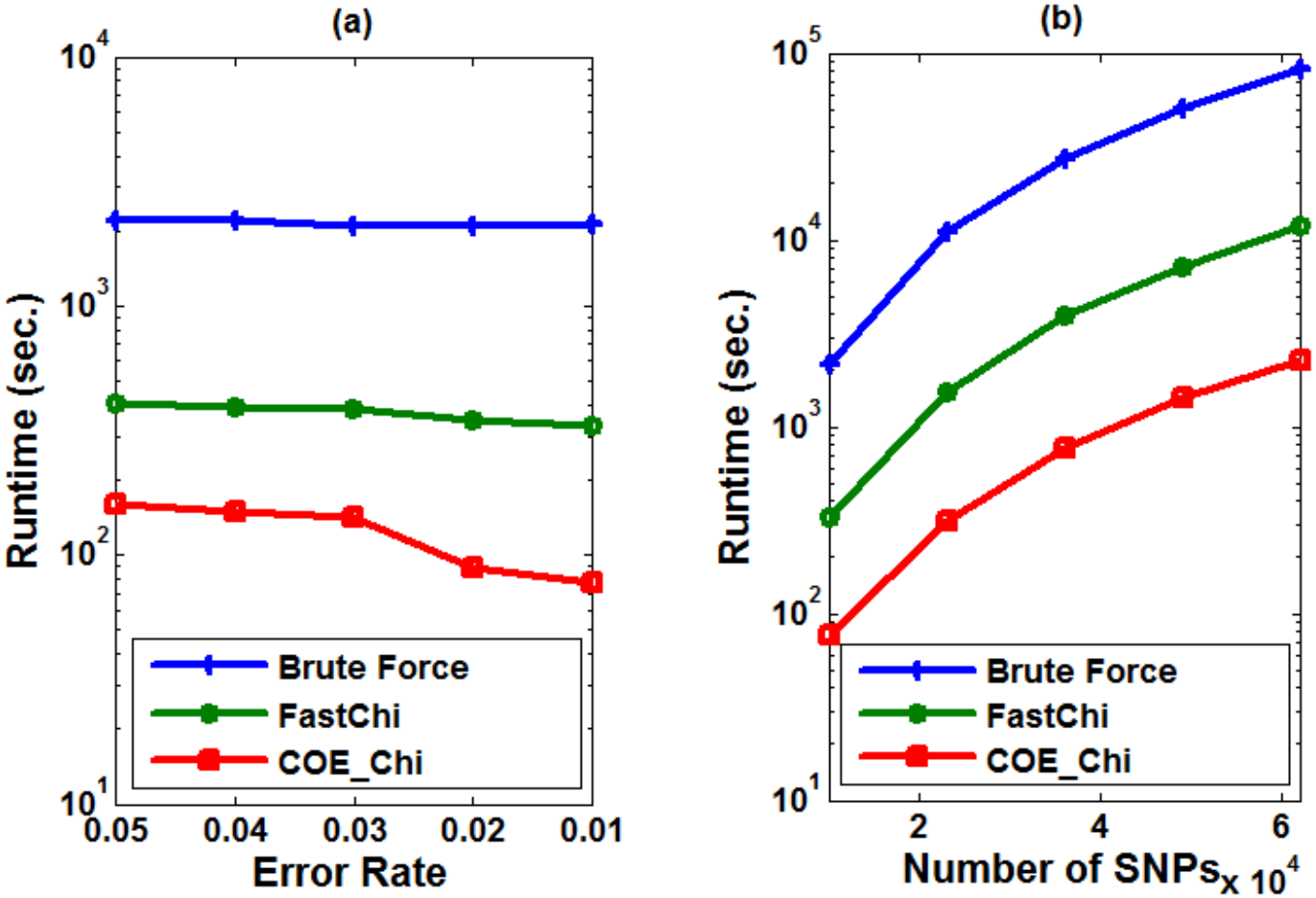
- Available tests
  - Chi-square, likelihood ratio, trend, entropy-based, etc …
  - Active research, more being proposed…
- A unified approach to all above tests?
- **Convexity** is the solution !

Convexity Example: Chi-square Test

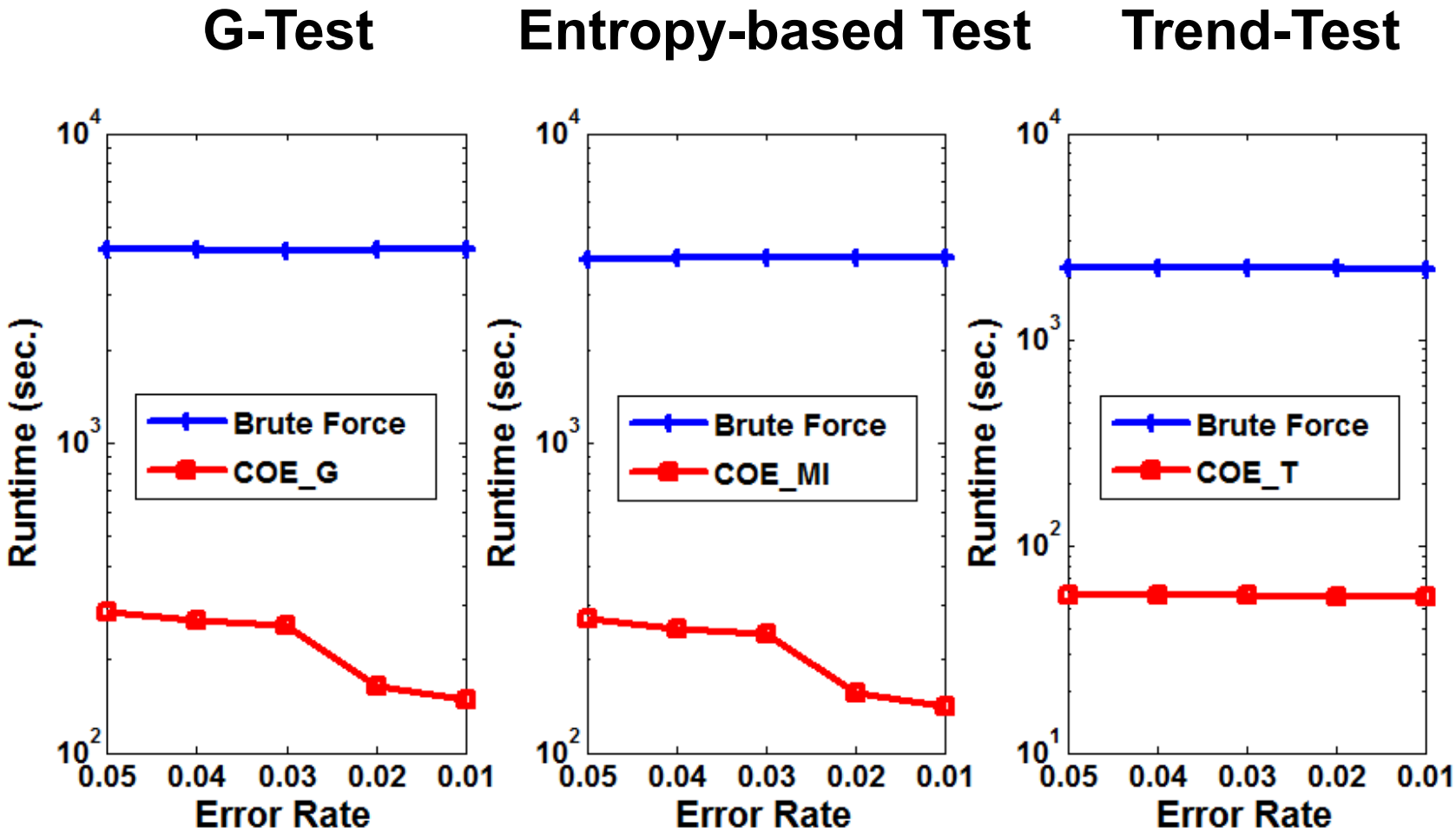


- **Theorem**: Convexity is a common property of many tests
- Determine the range of the *free variable* to get the upper bound

Brute Force vs. FastChi vs. COE



Brute Force vs. COE (on Various Tests)



Summary of the Algorithms

- Key ideas: bound, indexing, convexity

Algorithm	Supported Test
FastANOVA	ANOVA test
FastChi	Chi-square test
COE	Convex tests

- Designed for inbred mouse data: small sample size, binary SNPs

TEAM – Overview

	Previous	TEAM
SNPs	{0,1}	{0,1} & {0,1,2}
Sample size	Small	Large
Error Control	FWER	FWER & FDR
Test Statistic	With certain properties	Based on contingency table

The Computational Problem

- SNPs $\{X_1, X_2, \ldots, X_N\}$
- Phenotype Y, and permutations $\{Y_1, Y_2, \ldots, Y_k\}$

Problem: Computing **Test Values** for SNP-pairs

Computing **Contingency Tables**

Contingency Table

	$X_i = 0$			$X_i = 1$			$X_i = 2$		
	$X_j = 0$	$X_j = 1$	$X_j = 2$	$X_j = 0$	$X_j = 1$	$X_j = 2$	$X_j = 0$	$X_j = 1$	$X_j = 2$
$Y = 0$	a_1	a_2	a_3	b_1	b_2	b_3	e_1	e_2	e_3
$Y = 1$	c_1	c_2	c_3	d_1	d_2	d_3	f_1	f_2	f_3

Contingency Table

	$X_i = 0$			$X_i = 1$			$X_i = 2$		
	$X_j = 0$	$X_j = 1$	$X_j = 2$	$X_j = 0$	$X_j = 1$	$X_j = 2$	$X_j = 0$	$X_j = 1$	$X_j = 2$
$Y = 0$	a_1	a_2	a_3	b_1	b_2	b_3	e_1	e_2	e_3
$Y = 1$	c_1	c_2	c_3	d_1	d_2	d_3	f_1	f_2	f_3

$$\#(X_i, X_j, Y) = (1,1,1)$$

Contingency Table

	$X_i = 0$			$X_i = 1$			$X_i = 2$		
	$X_j = 0$	$X_j = 1$	$X_j = 2$	$X_j = 0$	$X_j = 1$	$X_j = 2$	$X_j = 0$	$X_j = 1$	$X_j = 2$
$Y = 0$	a_1	a_2	a_3	b_1	b_2	b_3	e_1	e_2	e_3
$Y = 1$	c_1	c_2	c_3	d_1	d_2	d_3	f_1	f_2	f_3

Only need to compute four variables

Contingency Table

	$X_i = 0$			$X_i = 1$			$X_i = 2$		
	$X_j = 0$	$X_j = 1$	$X_j = 2$	$X_j = 0$	$X_j = 1$	$X_j = 2$	$X_j = 0$	$X_j = 1$	$X_j = 2$
$Y = 0$	a_1	a_2	a_3	b_1	b_2	b_3	e_1	e_2	e_3
$Y = 1$	c_1	c_2	c_3	d_1	d_2	d_3	f_1	f_2	f_3

$$\#(X_i, X_j, Y) = (1,1,1)$$

Incremental Update

Y	X_1	X_2	X_3
0	0	0	1
0	0	0	0
0	1	1	0
0	1	0	0
0	1	0	1
1	1	1	0
1	0	1	1
1	1	0	1
1	1	1	1
1	0	1	0

$$\#(X_i, X_j, Y) = (1,1,1) \longleftrightarrow d_2$$

Incremental Update

Y	X_1	X_2	X_3
0	0	0	1
0	0	0	0
0	1	1	0
0	1	0	0
0	1	0	1
1	1	1	0
1	0	1	1
1	1	0	0
1	1	1	1
1	0	1	0

$$\#(X_i, X_j, Y) = (1,1,1) \longleftrightarrow d_2$$

$$(X_1, X_2, Y) \longleftrightarrow d_2 = 2$$

$$(X_1, X_3, Y) \longleftrightarrow d_2 = ?$$

Incremental Update

Y	X_1	X_2	X_3
0	0	0	1
0	0	0	0
0	1	1	0
0	1	0	0
0	1	0	1
1	1	1	0
1	0	1	1
1	1	0	0
1	1	1	1
1	0	1	0

$\#(X_i, X_j, Y) = (1,1,1) \longleftrightarrow d_2$

$(X_1, X_2, Y) \longleftrightarrow d_2 = 2$

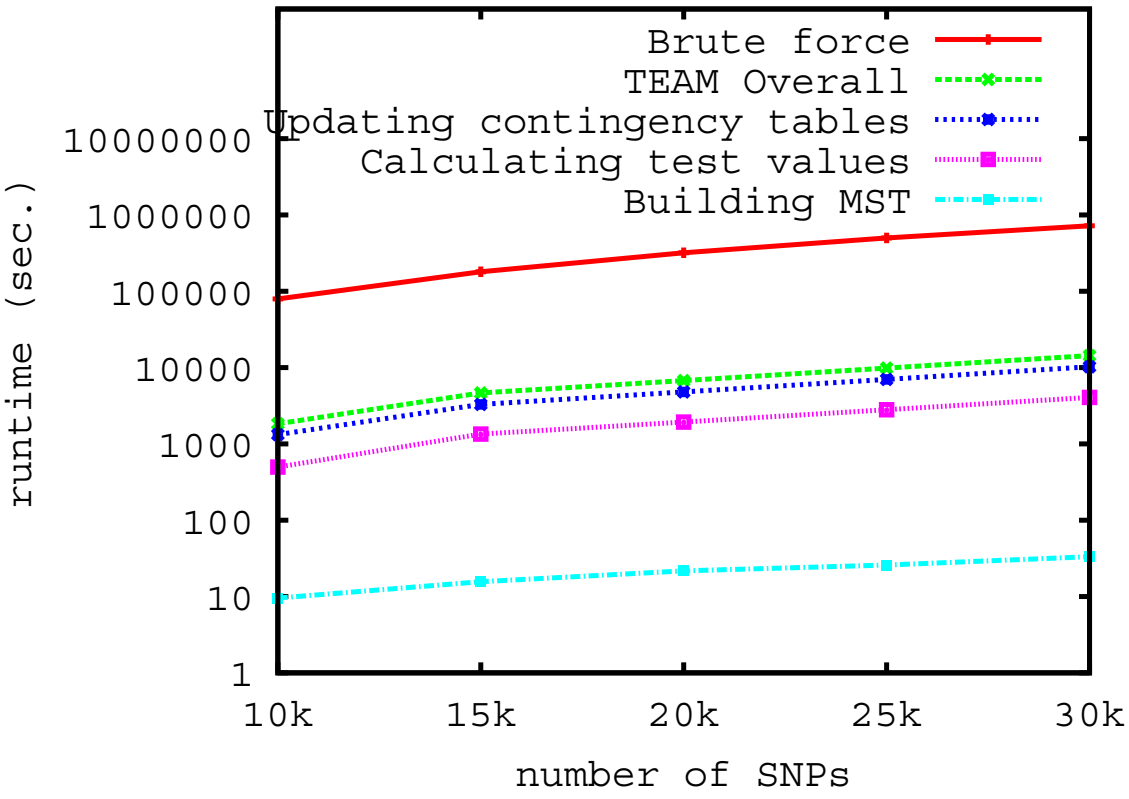
$(X_1, X_3, Y) \longleftrightarrow d_2 = 1$

No need to scan all individuals

Cost proportional to the difference

Updating order? – Minimal Spanning Tree

TEAM v.s. Brute Force (Human Data)



Data generated by Hapsample

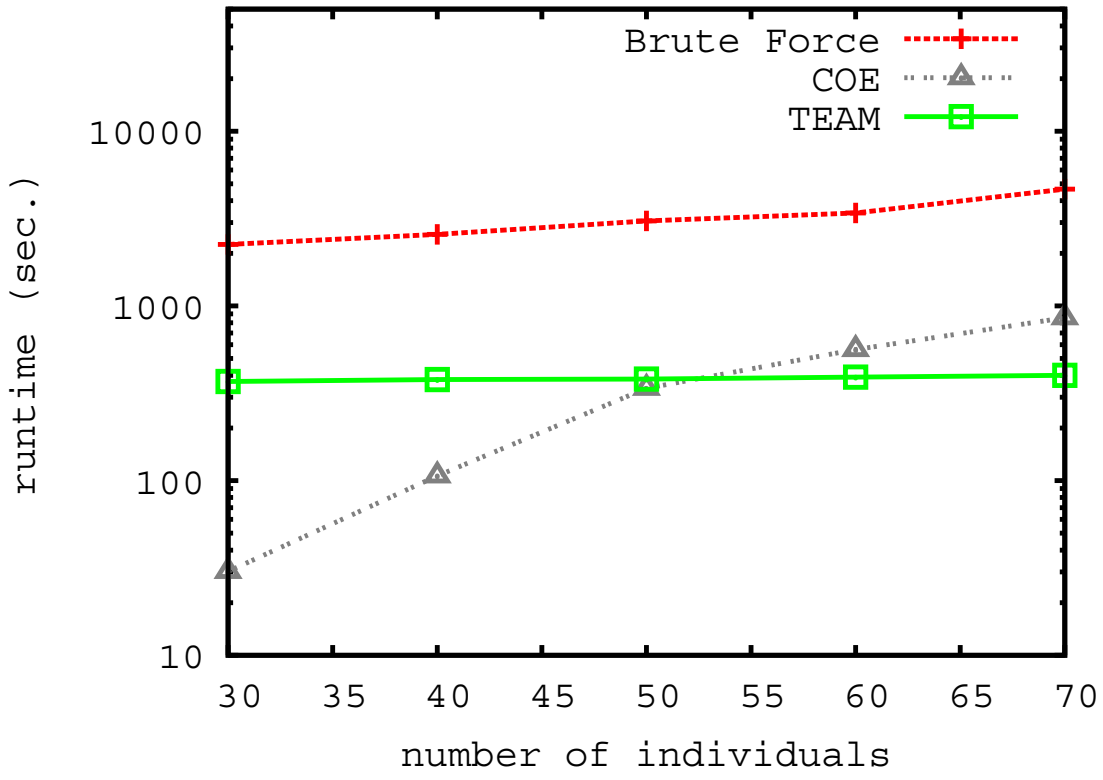
#SNPs = 100K

#Samples = 400

#Permutations = 100

Case/Control = 1

TEAM v.s. COE (Inbred Mouse Data)



Real Mouse Genotype Data (from Jackson Lab)

#SNPs = 10K

#Samples = 71

#Permutations = 100

Case/Control = 1

TEAM – Summary

- Designed for human study: large sample size, {0,1,2} SNPs
- Idea: incrementally update contingency tables

Overall Summary on Detecting Genetic Interactions

- Studying SNP-SNP interactions is important
- Challenges
  - Statistical: effective statistics
  - Computational: enormous search space
- We provide first solutions to
  - Efficiency and Optimality
  - Applicability

References

- **FastANOVA: an efficient algorithm for genome-wide association study, by Xiang Zhang, Fei Zou, and Wei Wang. *ACM SIGKDD*, pp. 821-829, 2008. (Best Research Paper)**
- **FastChi: an efficient algorithm for analyzing gene-gene interactions, by Xiang Zhang, Fei Zou, and Wei Wang. *PSB*, pp. 528-539, 2009.**
- **COE: a general approach for efficient genome-wide two-locus epistasis test in disease association study, by Xiang Zhang, Feng Pan, Yuying Xie, Fei Zou, and Wei Wang. *RECOMB*, pp. 253-269, 2009.**
- **TEAM: Efficient two-locus epistasis tests in human genome-wide association study, by Xiang Zhang, Shunping Huang, Fei Zou, and Wei Wang, *ISMB, Special Issue of Bioinformatics*, vol. 26, no. 12, pp. 217-227, 2010.**
- **Tools for efficient epistasis detection in genome-wide association study, by Xiang Zhang, Shunping Huang, Fei Zou, and Wei Wang. *Source Code for Biology and Medicine*, vol. 6, no. 1, pp. 1-3, 2011.**

ScAi
SCALABLE ANALYTICS INSTITUTE

THANK YOU

weiwang@cs.ucla.edu

Intelligent Data Exploration and Analysis Lab (IDEAL)