ParaHaploの名前の由来
paraHaplo = Parallel + Haplotype

いくつかの名前の候補のうち、googleでヒットしなかったものを選択

ParaHaplo

- 目的
  - 疾患関連遺伝子を発見し、病気の治療や予防に貢献する。
- 特徴
  - ゲノムワイド・ハプロタイプ関連解析を行う
  - 従来法より多くの疾患関連遺伝子を発見できる
  - 京速コンピュータ「京」などの並列計算機で高速解析
- 開発者
  - 三澤計治、長谷川亜樹、角田達彦、鎌谷直之

本日の講義内容

1. ゲノムワイド関連解析とは
2. ハプロタイプ関連解析とは
3. ParaHaploの概要
4. 解析例の紹介と速度比較

本日の講義内容

1. ゲノムワイド関連解析とは
2. ハプロタイプ関連解析とは
3. ParaHaploの概要
4. 解析例の紹介と速度比較

遺伝子が病気を引きおこす

- 乳がんの例
  - 日本人女性の約16人に一人がかかる
  - 欧米では約8人に一人

乳がんの遺伝的変異

- 乳がんの遺伝的変異
  - 母娘や姉妹に乳がん患者がいると乳がんの可能性が高い

    Peto et al. (2000) Nat Genet 26: 411-4.
  - 乳がんを引き起こす遺伝子はいくつかわかってきている
    - 例 : BRCA1 Miki et al. (1994) Science **266:** 66-71

遺伝子が病気を引きおこす

- アンジェリーナ・ジョリーさんの例
  - 母親を乳がんで亡くしている
  - 叔母も乳がんにかかり、遺伝子検査を受けた
  - BRCA1に変異があり、乳がんになる確率が87%と推定された
  - 乳房切除
- 病気の原因遺伝子がわかれば対策ができる
  - 病気のなりやすさに影響を与えている遺伝子を、疾患関連遺伝子という

先日、乳がんおよび卵巣がんになるリスクを少しでも低くするために両乳房切除手術を行ったアンジェリーナ・ジョリーさん

関連解析

- 目的
  - 疾患関連遺伝子を見つけること
- アイディア
  - 病気を引き起こす遺伝子に関しては、患者さんたちと健康な人達では、疾患関連遺伝子の頻度が違うはず
- ありふれた疾患には共通の変異がある
  - Common disease common variant hypothesis

関連解析の手法

- 患者さんたちと健康な人達を集めて来る
  - 患者さん･･･case
  - 健康な人達･･･control
- caseとcontrolの間で遺伝子頻度を統計検定する
- 有意差があればそこを疾患関連遺伝子と推測

一塩基多型(SNP)

- ある生物種集団のゲノム塩基配列中に一塩基が変異した多様性が見られ、その変異が集団内で1%以上の頻度で見られる時、これを一塩基多型と呼ぶ。
- 英語では、Single Nucleotide Polymorphism、略してSNPと呼ばれる
- 乳がんと関連していると考えられるBRCA1上のSNP rs1799950 Johnson et al. (2007) *Hum Mol Genet* **16:** 1051-7.

関連解析における検定

対立仮説

- 疾患関連遺伝子である
- Caseとcontrolで遺伝子頻度に**差がある**

帰無仮説

- 疾患関連遺伝子でない
- Caseとcontrolで遺伝子頻度に**差が無い**

SNP rs1799950 における乳がん患者(case)とcontrolで観察された数
Johnson et al. (2007) *Hum Mol Genet* **16:** 1051-7.より改変

	A	G
Case	861	83
Control	4656	264

カイ二乗検定に使う検定統計量
ピアソンスコア

	A	G
Case	a	c
Control	b	d

$$S = \sum_{i=1}^{k} \frac{\left(\text{観測値} - \text{期待値}\right)^2}{\text{期待値}}$$

$$= \frac{(a+b+c+d)(ad-bc)^2}{(a+b)(c+d)(a+c)(b+d)}$$

先程の例では、
S = 16.7

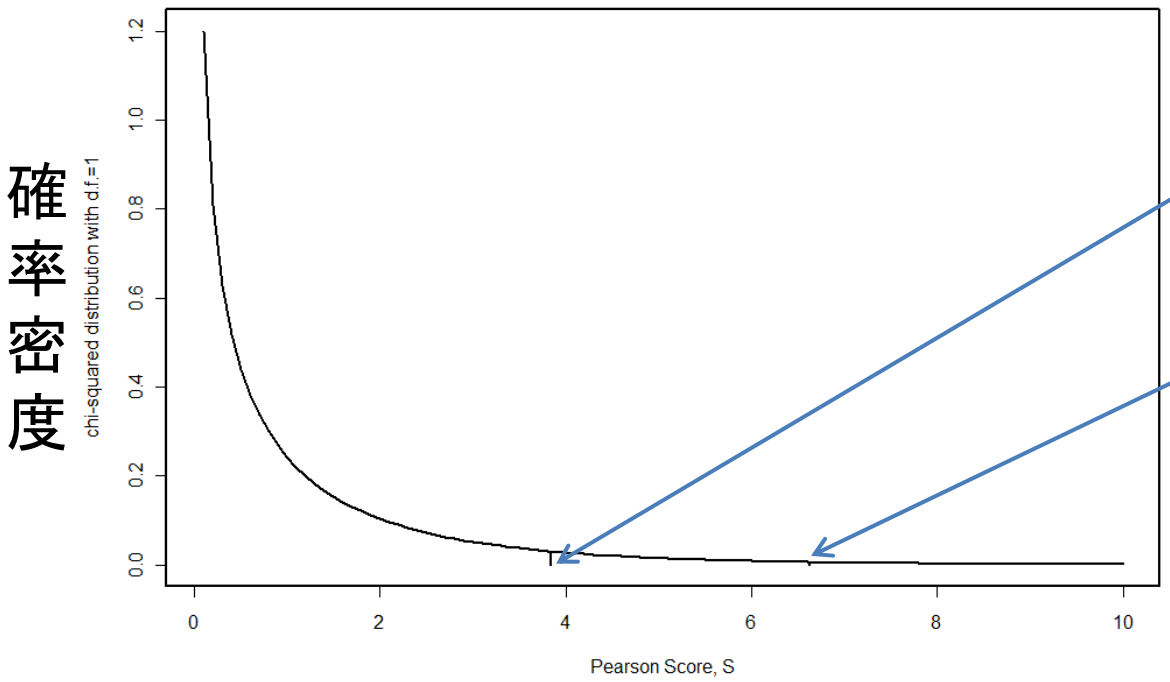
統計検定とエラー

- Type IとType IIのエラーがある
- Type I error
  - 帰無仮説が正しいのに棄却するエラー
  - 疾患に関連していない遺伝子を、関連していると言ってしまうエラー
- Type II error
  - 対立仮説が正しいのに帰無仮説を棄却しないエラー
  - 疾患に関連している遺伝子を、関連していないと言ってしまうエラー
- どちらも減らした方が良い

有意水準　Significance level

- Type I errorが起こる可能性をどの程度許容するかを有意水準と言う
- 有意水準が低いほど、厳しい検定となる
- 1%の有意水準→type I errorが起こる確率が1%未満
- 5%の有意水準→type I errorが起こる確率が5%未満

自由度1のカイ二乗分布



*S*が3.84より
大きくなる確率は
5%

*S*が6.63より
大きくなる確率は
1%

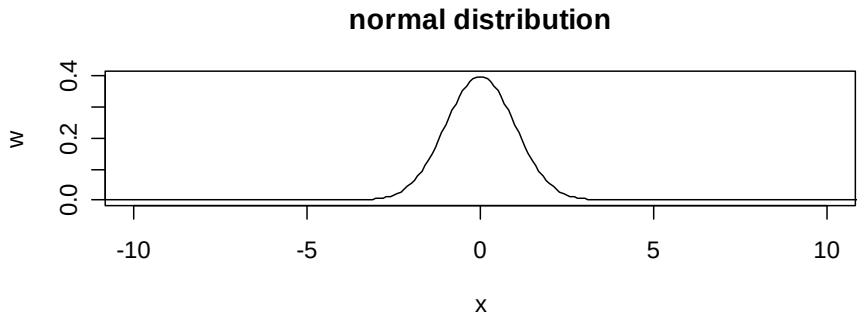
閾値(Threshold)

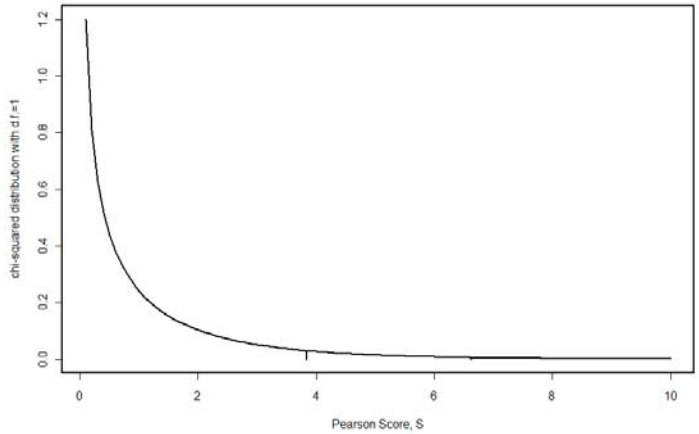
- 有意水準を5%に設定した時はSの閾値は3.84
- 有意水準を1%に設定した時はSの閾値は6.63
- 16.7は1%レベルの閾値を超えているので、
1%レベルで帰無仮説は棄却される

カイ二乗分布とは

- 標準正規分布に従う確率変数をk個取ってきたときに、
- そのk個の確率変数の二乗の和が、自由度kのカイ二乗分布に従う

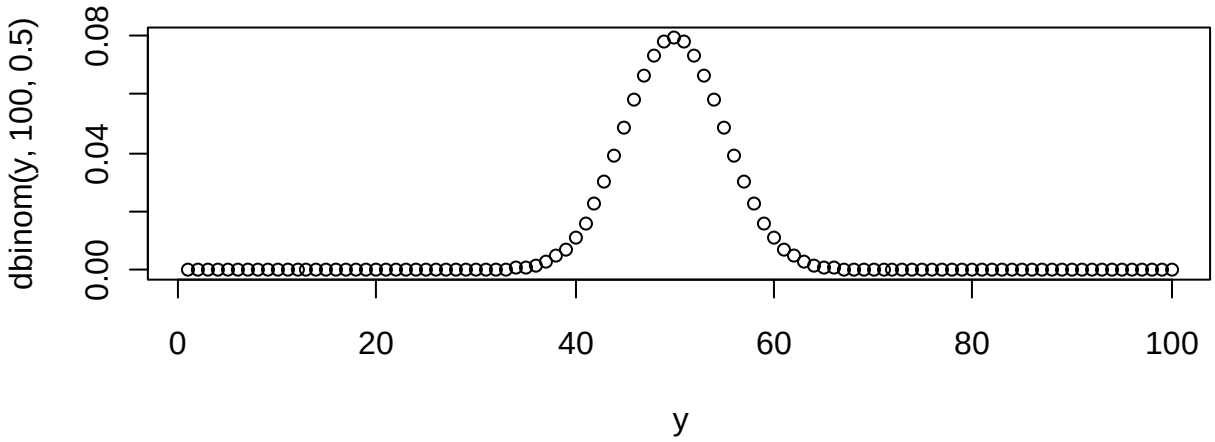
$$w(x) = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{x^2}{2}\right)$$





二項分布とカイ二乗分布

- 観察された中に塩基が何個カウントされるかは、二項分布に従う。
- 期待値が大きい時に、二項分布は正規分布に似ているため、帰無仮説の下でのSの分布にはカイ二乗分布が使える



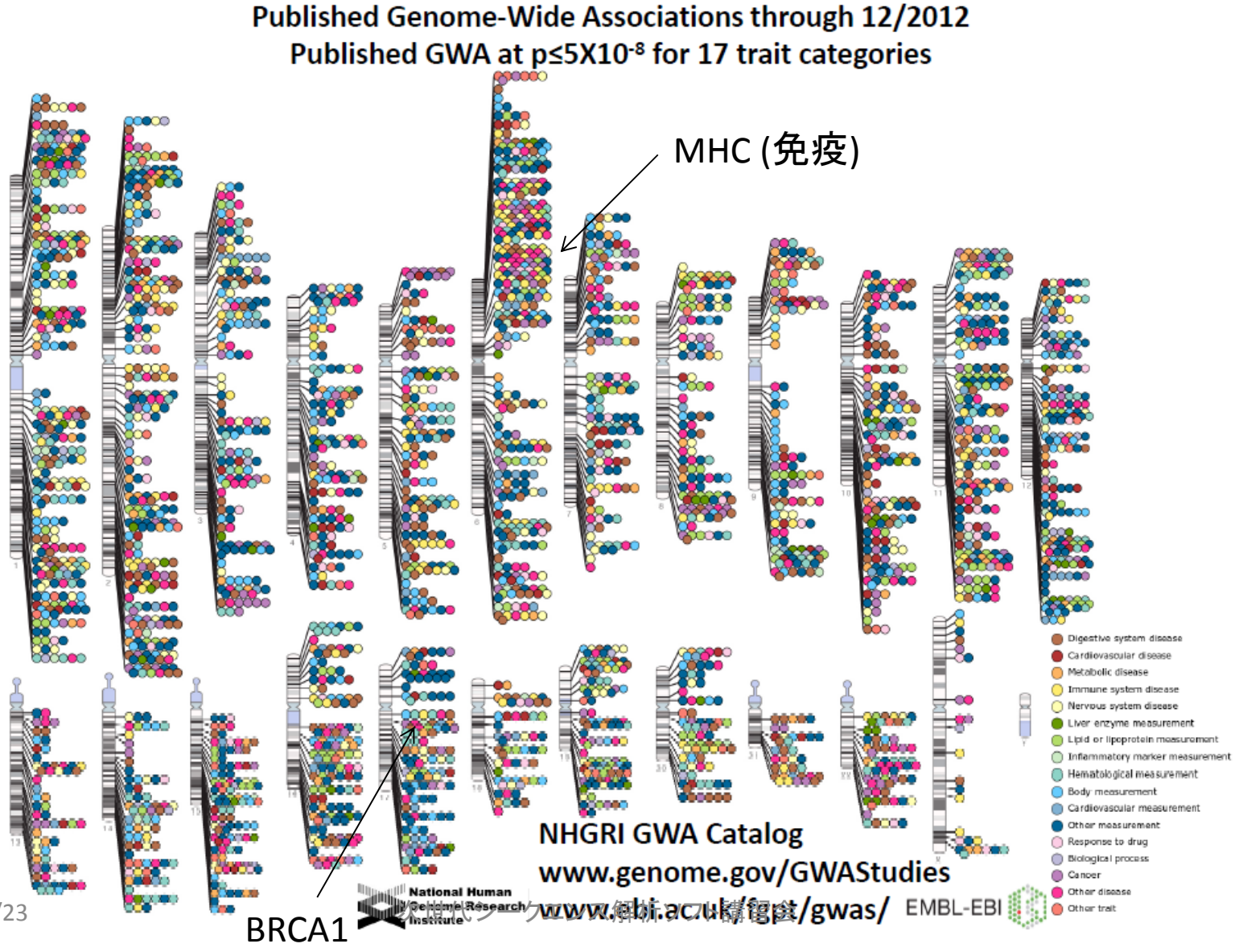
ゲノムワイド関連解析　GWAS

- Genome-wide association study, 略してGWAS
- 関連解析を全ゲノムに対して行い、疾患関連遺伝子を網羅的に見つける方法
- Type I errorに気をつけなくてはいけない
- Bonferroniの補正
  - N回テストがあり、有意水準をpとするときに、一つ一つのテストの有意水準をp/Nにする方法
- 世界に先駆けて理研のグループで行われた
  - Ozaki, K., Ohnishi, Y., Iida, A., Sekine, A., Yamada, R., Tsunoda, T., Sato, H., Hori, M., Nakamura, Y. and Tanaka, T.　(2002) *Nat Genet* **32:** 650-4.

ゲノムワイド関連解析　GWAS

- 調べたいサイトに関して、個人個人の遺伝子型を高速に、そして安価に検出する実験方法が開発されている。

ゲノムワイド関連解析の成功例



従来の関連解析の問題点

- 従来の関連解析
  - 従来のGWASでは、各SNPを独立と考えて、関連解析を行う
- 問題点
  - 近くにあるSNPは連鎖していることが多く、独立していない
  - 従来の方法ではtype II errorが多くなり、疾患関連遺伝子を見逃している可能性が高い

本日の講義内容

1. ゲノムワイド関連解析とは
2. ハプロタイプ関連解析とは
3. ParaHaploの概要
4. 解析例の紹介と速度比較

ハプロタイプ関連解析

- 一つ一つのSNPを独立と考えずに、連鎖を考慮して、ハプロタイプ単位で行う関連解析
- ハプロタイプ単位で行うGWASのtype I errorの確率を計算し、適切な有意水準を設定することで、いままで見つかっていなかった疾患関連遺伝子を見つけることができる

連鎖 Linkageと
連鎖不平衡 Linkage Disequilibrium

- ２つのalleleが同じ染色体に乗っていて、位置も近い場合は、親から子に渡るときにセットで渡る場合が多い。これを連鎖と言う
- 連鎖が壊れることを組み換え(recombination)という
- 十分な世代が経ち、組み換えが沢山起こった後、連鎖が見られなくなった場合を、連鎖平衡(linkage equilibrium)という
- そこまで至らない場合は連鎖不平衡(linkage disequilibrium; LD)

ハプロタイプ（Haplotype）とは

- “haploid genotype”（半数体の遺伝子型）の略で、生物がもっている単一の染色体上のDNA配列のタイプのこと

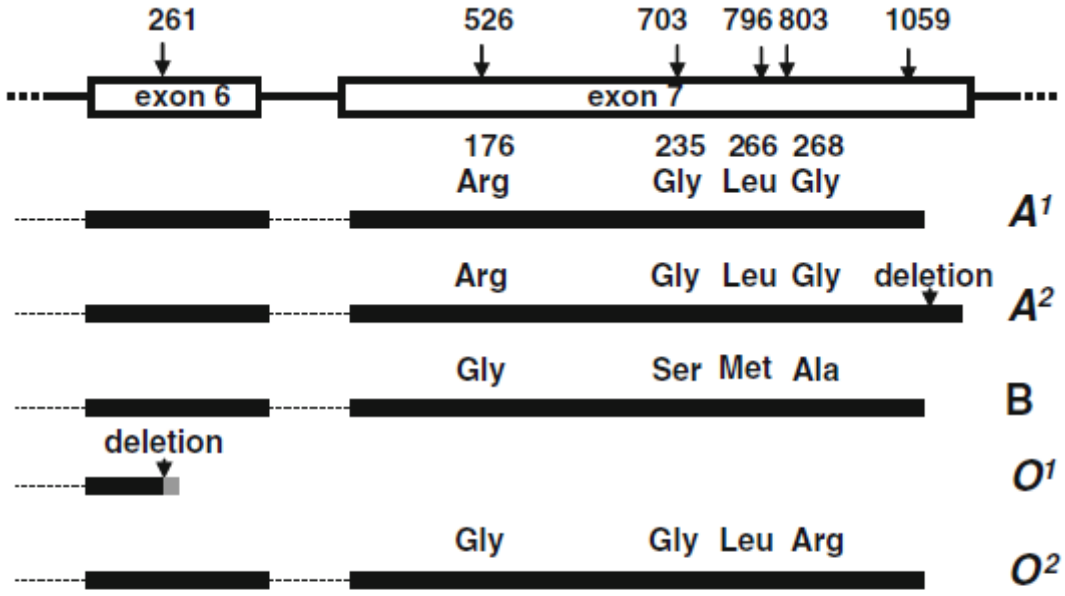


Fig. 2 Exons 6 and 7 of the *ABO* gene, showing the position of the nucleotide deletions responsible for the common form of O (O^1) (exon 6) and for A_2 (exon 7), and the positions of the four nucleotide changes in exon 7 responsible the amino acid residues that are characteristic of the A and B transferases. Below are representations of the encoded transferases

ハプロタイプの例、血液型遺伝子
Daniels G (2009) Hum Genet **126:**729-742

SNPをばらばらに考えるより、haplotype単位で考えた方が遺伝学的には意味がある

ハプロタイプ関連解析

帰無仮説
ハプロタイプ頻度がcaseとcontrolで同じ

```
Haplotype 0 ATAATTTAC 頻度h[0]
Haplotype 1 ACGGCCGGT 頻度h[1]
Haplotype 2 GTAATTTAT 頻度h[2]
Haplotype 3 ATAATTTAT 頻度h[3]
Haplotype 4 ATAATTTAC 頻度h[4]
```

```
Case
一人目 ATAATTTAC
       ACGGCCGGT
二人目 GTAATTTAT
       ATAATTTAT
...
```

```
Control
一人目 ATAATTTAC
       ATAATTTAC
二人目 GTAATTTAT
       ATAATTTAC
...
```

確率は多項分布に従う
Multinomial distribution

ハプロタイプ関連解析のtype I errorの確率

帰無仮説
ハプロタイプ頻度がcaseとcontrolで同じ

検定の手順

- サンプルされた配列に関して、一つ一つのサイトそれぞれにPearson Scoreを求める
- もっとも高いPearson ScoreをSとする。
- 帰無仮説の下でのSの確率分布を計算し、求めたい有意水準に対する閾値も設定する。
- 観測されたゲノム多型データに対して、Sを計算した時に、Sがその閾値を超えていれば、帰無仮説は棄却される

正確法による確率計算

各サイトのPearson Scoreがあるthresholdを越えた時に1,
それ以外のときに0となる関数*f*を考える

$$P[f(X_{11}, X_{12}, \ldots, X_{1,L-1}, X_{21}, X_{22}, \ldots, X_{2,L-1}) = 1]$$

$$= \sum_{x_{11}=0}^{2n_1} \sum_{x_{12}=0}^{2n_1-x_{11}} \sum_{x_{13}=0}^{2n_1-x_{11}-x_{12}} \cdots \sum_{x_{1,L-1}=0}^{2n_1-x_{11}-x_{12}-\cdots-x_{1,L-2}}$$

$$\times \sum_{x_{21}=0}^{2n_2} \sum_{x_{22}=0}^{2n_2-x_{21}} \sum_{x_{23}=0}^{2n_2-x_{21}-x_{22}} \cdots \sum_{x_{2,L-1}=0}^{2n_2-x_{21}-x_{22}-\cdots-x_{2,L-2}}$$

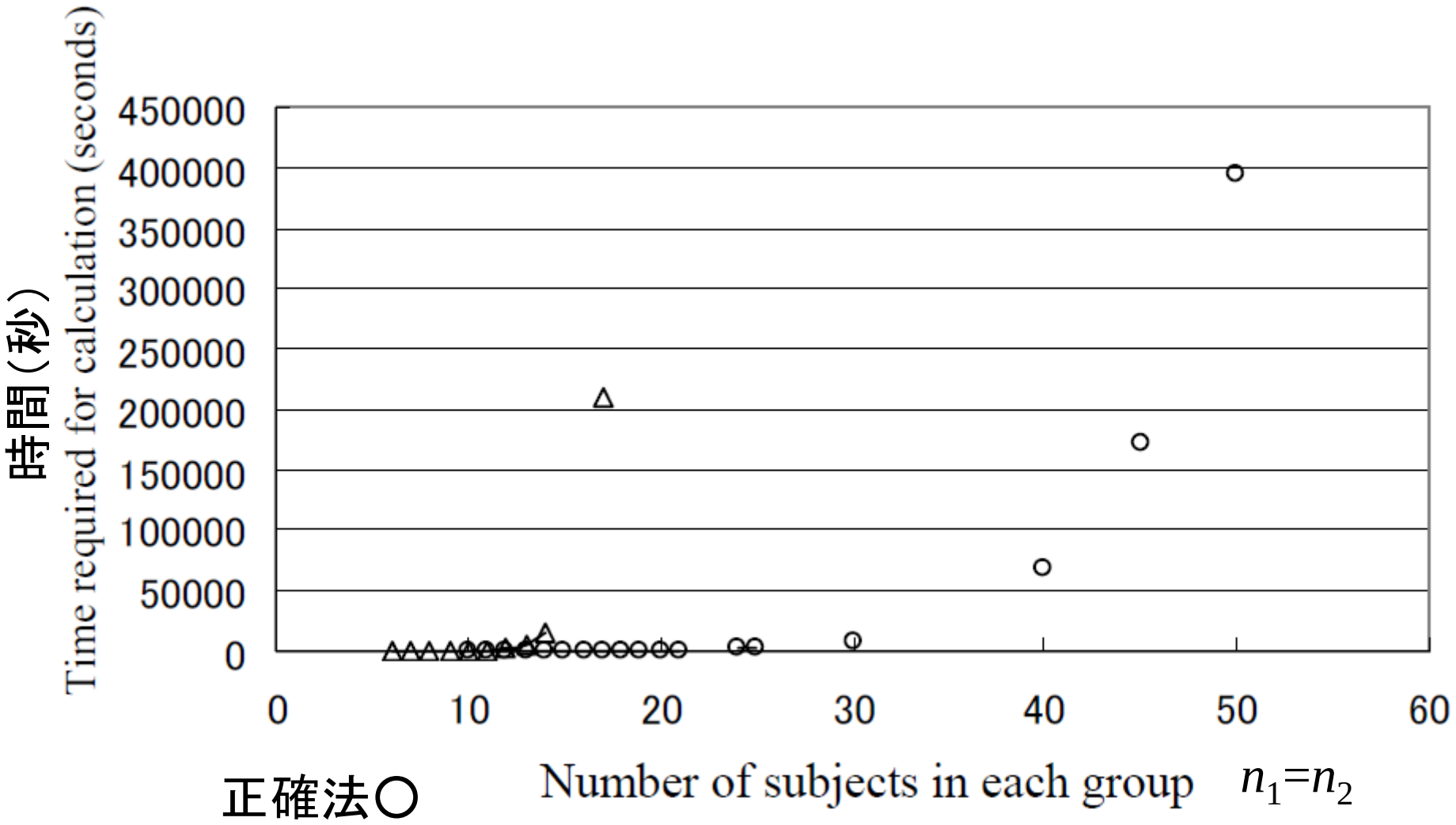
$$\times f(x_{11}, x_{12}, \ldots, x_{1,L-1}, x_{21}, x_{22}, \ldots, x_{2,L-1}) \frac{(2n_1)!(2n_2)!}{\prod_{i=1}^{L}\prod_{j=1}^{2} x_{ji}!} \prod_{i=1}^{L} h_i^{\sum_{j=1}^{2} x_{ji}},$$

where $x_{1L} = 2n_1 - \sum_{i=1}^{L-1} x_{1i}$ and $x_{2L} = 2n_2 - \sum_{i=1}^{L-1} x_{2i}$.

非常に計算に時間がかかる

Misawa *et al.* (2008) *J Hum Genet* ***53: 789-801.***

正確法の問題点：時間がかかる



正確法○

100人9座位でPCクラスタで5日間

5千人50万座位ではスパコンでも10^{1000}年くらいかかる

本日の講義内容

1. ゲノムワイド関連解析とは
2. ハプロタイプ関連解析とは
3. ParaHaploの概要
4. 解析例の紹介と速度比較

MCMC法(Markov Chain Monte Carlo)

多項分布に従う確率で配列をサンプリングしてくるアルゴリズム

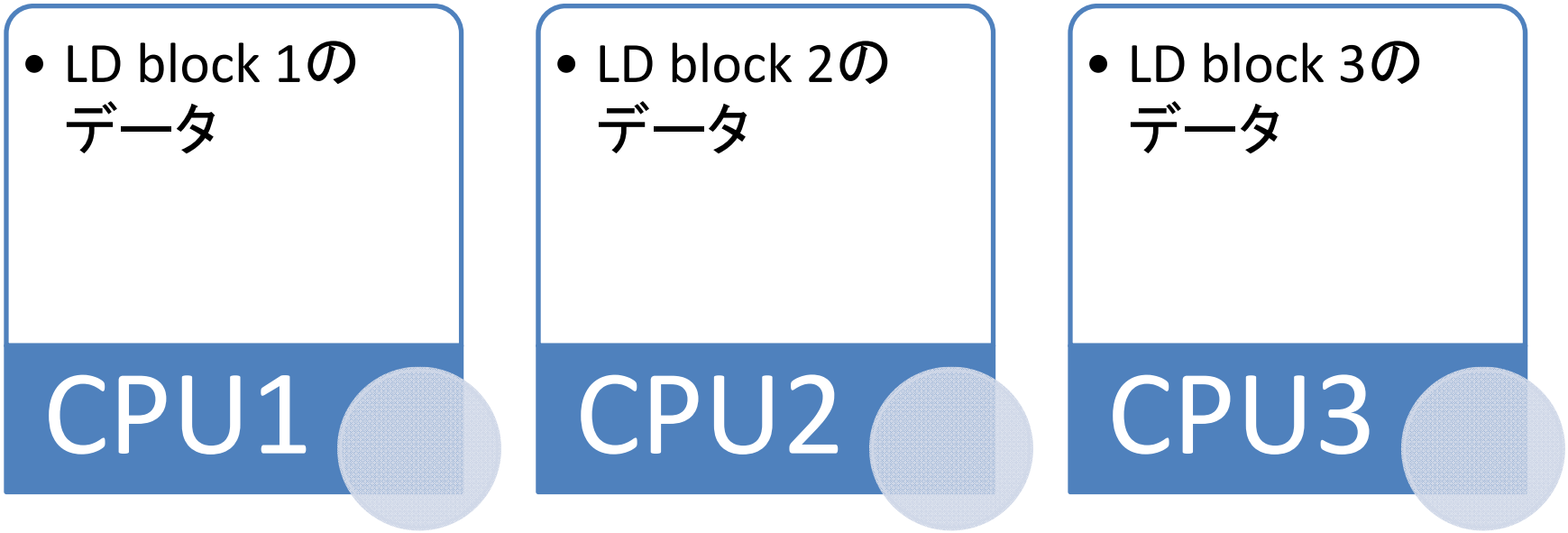
1. x_{ji}を適当な数でスタート。ただし合計がサンプルサイズになるようにする。
2. CaseかControlかどちらかを選ぶ。Caseならj=0, Controlならj=1
3. Haplotype vを選び、その数を調べる。これをx_{jv}とする。
4. x_{jv}がゼロでなければ、もう一つ別のHaplotype uを選び、この数をx_{ju}とする
5. x_{jv}を一つ減らし、x_{ju}を一つ増やすことを考える
6. $c = h_v x_{ju} / h_u (x_{jv}+1)$を計算し、$c \geq 1$なら必ず、$c<1$なら確率$c$で次の状態にする。それ以外なら状態に変化なし。
7. Caseとcontrolで差があるかどうかを示す関数fを計算し、f=1となる回数をカウント
8. 2.から繰り返す

直接確率法よりましだけど、それでも結構時間がかかる

Misawa *et al.* (2008) *J Hum Genet* ***53: 789-801.***

並列化

LD Blockごとに別のCPUに計算させる



- 連鎖が強いSNP同士をLD blockとしてまとめる
- LD block内でhaplotypeを推定してparaHaploで解析
- LD block間は連鎖が弱いので別のnodeで解析
- Bonferroni補正でglobalなP値を求める。

古い関数

```
static void TypeIMarkovOld(int[][] X, double[] h, int L)
{
    int j = 0;
    int /*long*/ u = 0;
    int /*long*/ v = 0;
    int XjuStar = 0;
    int XjvStar = 0;

    double c = 0;
    double tmp = 0;

    /* procedure 3 */
    j = TypeIZeroOne();

    /* procedure 4 */
    u = (int)(myRand() * L);

    if (0 == X[j][u]){ /* condition 5 */
        /* invariant */
    }
    else{ /* condition 6 */
        v = 0;
        int count=0;
      v = (int)(myRand() * L);
        do{
            v = (int)(myRand() * L);
        } while(u == v);

        /* new candidate */
        XjuStar = X[j][u] - 1;
        XjvStar = X[j][v] + 1;

        /* transition probability */
        c = h[v] * X[j][u] / (h[u] * (X[j][v] +1));

        if (c >= 1){ /* condition 8 */
            X[j][u] = XjuStar;
            X[j][v] = XjvStar;
        }
        else{
            tmp = myRand();
            if (tmp < c){
                X[j][u] = XjuStar;
                X[j][v] = XjvStar;
            }
        }

        /* procedure 10 */
    }
}
```

4回とちょっと乱数を呼び出してる

新しい関数

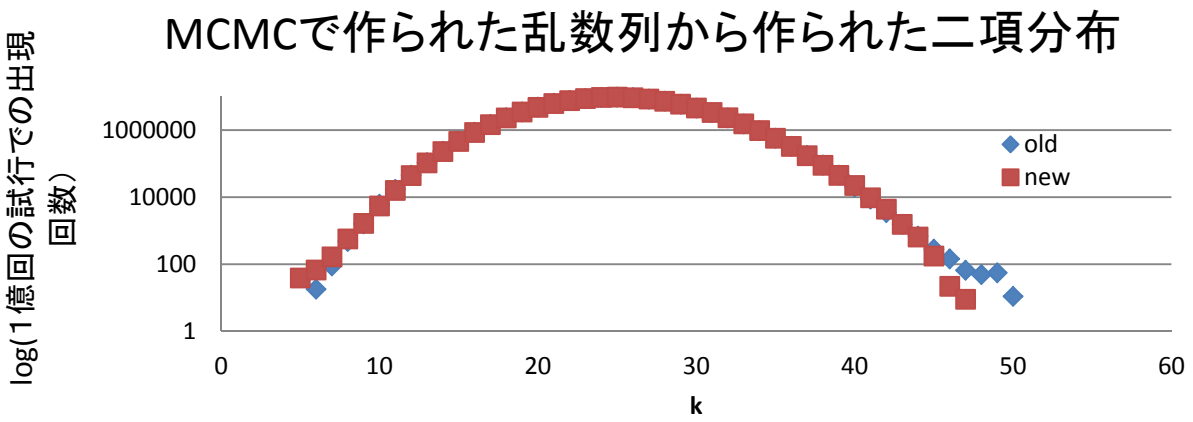
```
static void TypeIMarkov(int[][] X, double[] h, int L)
{
    /* procedure 3 */
    double tmp=myRand()*2*L*(L-1);//乱数は重いので使い回し
    int v=(int)(tmp);
    tmp-=v;              /* 0<=tmp<1 */
    int j=(v&1);          /* 0 or 1 */
    v>>=1;
    int u=(v%L);           /* 0,...,L-1 */
    v/=L;                /* 0,...,L-1 */
    if(v>=u) v++;
    int Xjv=X[j][v];
    int Xju=X[j][u];
    if(0<Xju){
        if (tmp*(h[u]*(Xjv+1))<h[v]*Xju){
            Xju--;
            Xjv++;
        }
        X[j][v]=Xjv;
        X[j][u]=Xju;
    }
}
```

乱数1回に

ひとつの乱数から4つの乱数を作る

この方法は精度が出てるか

- Double precision SIMD-oriented Fast Mersenne Twister (dSFMT)を利用
- 修正BSDライセンス
- ParaHaploではダウンロードしコンパイルしリンクして使用（ソース内に直接は入っていない）
- ParaHaploではその他の疑似乱数発生ルーチンも対応（Numerical Recipes in Cなど）
- 使っているのは0から1まで[0, 1)の倍精度の一様乱数
- (仕様書を信じるなら)4つの乱数を取り出しても十分精度はあるはず



開発環境

- ハードウエア
  - Quest
  - RICC
  - 京速コンピュータ「京」
- 言語
  - C
    - GCC, Intel C, 富士通C
      - 非並列化版
      - MPI並列化版
        - » OpenMPIによるコア並列
  - Java
    - 普及用、非並列化版

コア数の確認

```
 int threadNum = 1;          /* スレッド数 */
 int iam = 0;                /* スレッド番号 */
#ifdef _OPENMP
    /* 使用可能なプロセッサ数を取得 */
    threadNum = omp_get_num_procs();
#endif
/* 各スレッド専用のメモリ領域を確保 */
    X = (int***)malloc1Dim(sizeof(int**),
threadNum);
```

#pragma omp parallel文

- // OpenMP スレッド並列開始
- #pragma omp parallel private(m, iam)
- {
-
- #ifdef _OPENMP
- /* 自分のスレッド番号を取得 */
- iam = omp_get_thread_num();
- #endif
- private後の変数は、個々のスレッド、つまり個々のコアでプライベートに使われる変数の名前。それ以外の変数は共有に使われる

#pragma omp for 文

- #pragma omp for private(j, flag, diff, S)
- for (n = 0; n < repeat; n++){
- diff = 0;
- for (m = 0; m < gen; m++){
- total++;
- TypeIMarkov(X[iam], freq, L);
- (略)
- #pragma omp forの次の行のfor文を分割してスレッド並列で実行
- privateの後の変数は、個々のスレッド、つまり個々のコアでプライベートに使われる変数の名前。repeatとgenは定数

MPI並列化アルゴリズム

1. データを読み込む
   - Haplotypeデータ
   - Haplotype block データ
2. Haplotype blockごとに、Haplotypeデータを各計算機に分ける
   - MPI_Bcast
3. 統計検定をする
   - 直接確率検定(Misawa et al. 2008)
   - STP, RAT (Kimmel and Shamir 2006)
4. 結果を集める
   - MPI_Gather
5. 結果を出力する

本日の講義内容

1. ゲノムワイド関連解析とは
2. ハプロタイプ関連解析とは
3. ParaHaploの概要
4. 解析例の紹介と速度比較

ParaHaploの実行

- Wellcome Trust Case-Control Consortium (WTCCC)からのデータ提供を受けた
- 1型糖尿病、2型糖尿病、高血圧などの解析を進めている
- 京速コンピュータ「京」で解析

Flow chart of analysis by using paraHaplo

Genotype data of case and control populations

↓

Quality Check

↓

LD blocks were obtained by using Gabriel et al.'s (2002) algorithm

↓

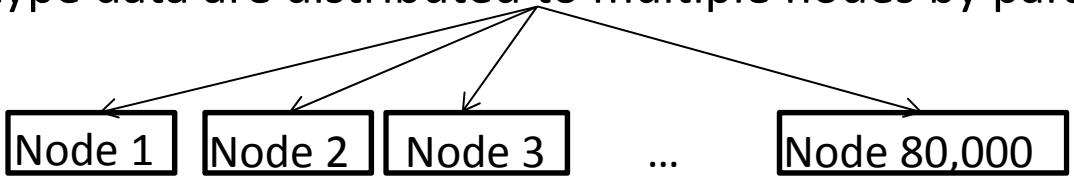
Haplotype phasing by using Mach 1.0

↓

Haplotype data are cut into LD blocks

↓

Haplotype data are distributed to multiple nodes by paraHaplo



Haplotype-based GWAS will be conducted on each node by using paraHaplo

JPTとCHBの間のhaplotype-based GWASを22番染色体で行ったときにかかった時間とCPUの数の関係

Table 2. Elapsed times and speedups obtained with ParaHaplo on the HapMap 3 JPT data and CHB of chromosome 22

Number of Processing Units	Calculation Time	Speed Ratio [a]
1	1時間 19分 58秒	1
64	3分 41秒	22
128	2分 1秒	40
256	1分 25秒	56
512	53秒	91
768	47秒	101
1536	41秒	116

[a] Ratio of Computational Time of Single Processor to Computational Time of Multiple Processors

ParaHaploの名前の由来
paraHaplo = Parallel + Haplotype

いくつかの名前の候補のうち、googleでヒットしなかったものを選択

paraHaploでgoogle検索すると論文とソースが出てきます

ParaHaploに関する論文

- ParaHaplo
  - http://www.scfbm.org/content/4/1/7
- ParaHaplo 2.0
  - http://www.scfbm.org/content/5/1/5
- ParaHaplo 3.0
  - http://www.scfbm.org/content/6/1/10

ソース

- http://sourceforge.jp/projects/parallelgwas/

ISLiMのサイトからもリンクしています

現在および今後の進展

- Wellcome Trust Case-Control Consortium (WTCCC)からのデータ提供を受けた
  - 1型糖尿病、2型糖尿病、高血圧の解析を行っている
    - それぞれ約1500人
  - 乳がんも予定している
  - Controlは約2000人
- HPCIという機構を通じて日本全国から「京」が利用できるようになる

謝辞

- 京での計算に関しては京速コンピュータ京の試験利用、および本年3月での特別運用での結果です。また、PCクラスタでの性能計測に関しては理化学研究所情報基盤センターのRICCを使用しています。