博士課程「ゲノム統計学大意」 修士課程「ゲノムサイエンス概論」の合同講義

受講に先駆けて

- ノートパソコンを持参
- R、Rstudioをインストール (https://qiita.com/hujuu/items/ddd66ae8e6f3f989f2c0)
- スライド上映形式の講義ではありません
- 必死でノートを取りましょう
- 公開資料 (https://github.com/ryamada22/atom/blob/master/KurumeLecture/KL2019/KL2019Final.pdf)をプリントアウトして、それに書き込むというのもよい方法です

スケジュール

90分　x　15コマ = 45分　x　30コマ 金 3-7限、土 1-2限

1-1 イントロダクション

1-2 分子遺伝学概論（１）

2-1 分子遺伝学概論（２）

2-2 分子遺伝学概論（３）

3-1 p値と検定

3-2 解析の再現性とRmarkdown

4-1 帰無仮説、データシミュレーション作成、一様分布

4-2 2x2表のカイ二乗検定とフィッシャーの正確確率検定

5-1 SNVだけで学ぶ統計学　イントロダクション

5-2 アレル、ディプロタイプ、ハーディワインベルグ平衡

6-1 ケースコントロール・スタディ 2x3表

6-2 ケースコントロール・スタディ 2x3表

7-1 確率と尤度

7-2 最尤推定と二項分布とベータ分布

8-1 ハプロタイプと連鎖不平衡

8-2 連鎖不平衡係数とEMアルゴリズム

9-1 多重検定とボンフェロニ補正

9-2 多重検定とパーミュテーション法(と正確確率法)

10-1 次世代シークエンサーデータとリードのマッピング

10-2 デプスとSNV

11-1 SNV以外で学ぶ統計学　イントロダクション

11-2 トランスクリプトームとヒートマップ

12-1 クラスタリング、ヒートマップ

12-2 系統樹というクラスタリング

13-1 階層型クラスタリングと非階層型クラスタリング

13-2 ネットワークというグラフ

14-1 非線形クラスタリング tSNEを自力で使う

14-2は予備。きっと時間が足りなくなるから

15-1,15-2 試験・試験

1-1 イントロダクション

- 自己紹介・受講目的確認
- 知識確認アンケート(基本的な知識 on 分子遺伝学・統計学/データサイエンス)
- 内容概観
- ゲノム・オミックス学
- ゲノム統計学

1-2 分子遺伝学概論（１）

2-1 分子遺伝学概論（２）

2-2 分子遺伝学概論（３）

- ゲノムとオミックス (http://omics.jp/about-omics)
- 統計学 (https://ja.wikipedia.org/wiki/%E7%B5%B1%E8%A8%88%E5%AD%A6)
- (機械)学習 (https://ja.wikipedia.org/wiki/%E6%A9%9F%E6%A2%B0%E5%AD%A6%E7%BF%92)

参考資料

- 学部１回生の講義資料　分子遺伝学 (https://github.com/ryamada22/MolGenetics)
- 京大MOOCの第１の第一セクション (https://www.edx.org/course/introduction-to-statistical-methods-for-gene-mapping)
- オーム社『遺伝統計学の基礎』ISBN 978-4-274-06822-5
- SNV解析資料(山田) (http://www.statgenet.med.kyoto-u.ac.jp/ra/indexold.html)

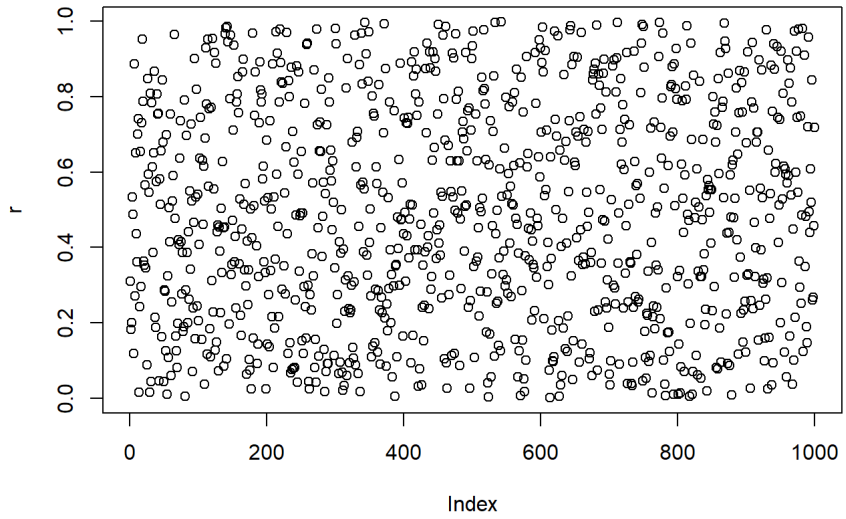
読み物

- 多型 (https://ja.wikipedia.org/wiki/%E5%A4%9A%E5%9E%8B)

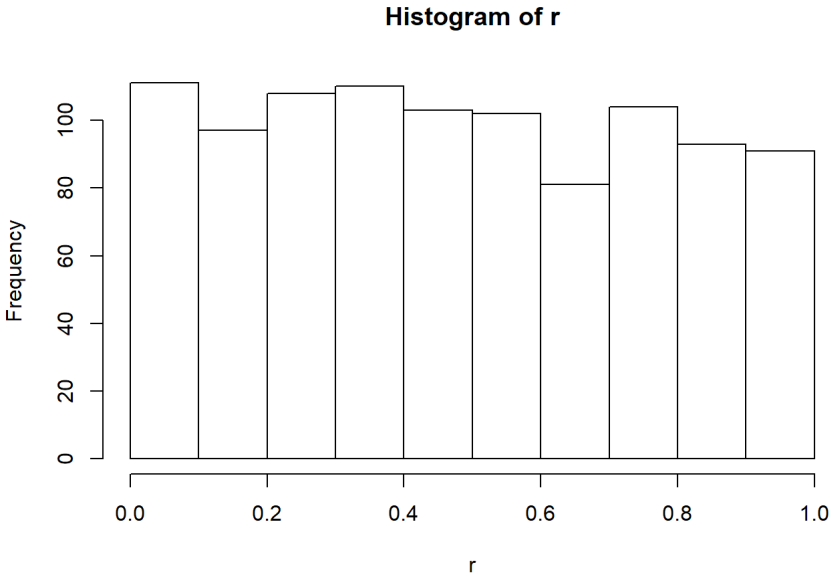
3-1 p値と検定

- 参考スライド (http://www.statgenet.med.kyoto-u.ac.jp/wiki_tokyo/index.php/%E6%97%A5%E6%9C%AC%E4%BA%BA%E9%A1%9E%E9%81%BA%E4%BC%9D%E5%AD%A6%E4%BC%9A%E6%9
(3-1～6-2)

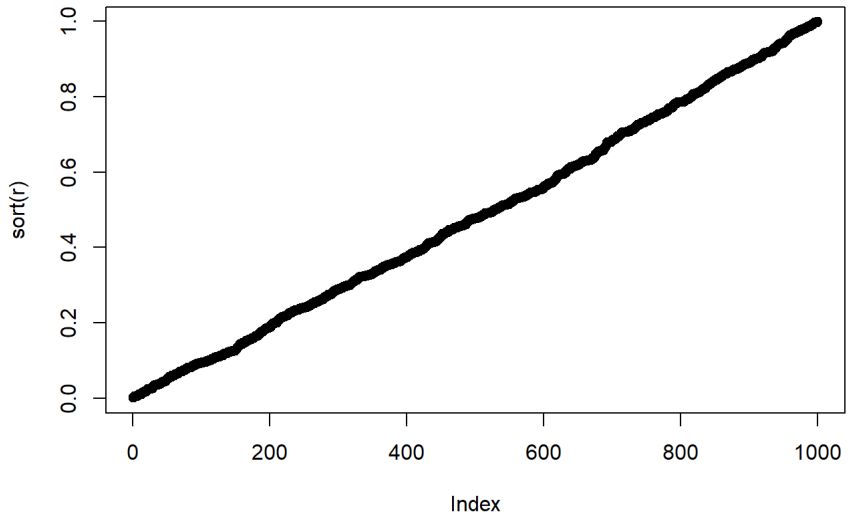
```
# 一様分布からの乱数発生
n <- 1000
r <- runif(n)
plot(r)
```




```
hist(r)
```




```
plot(sort(r))
```



3-2 解析の再現性とRmarkdown *

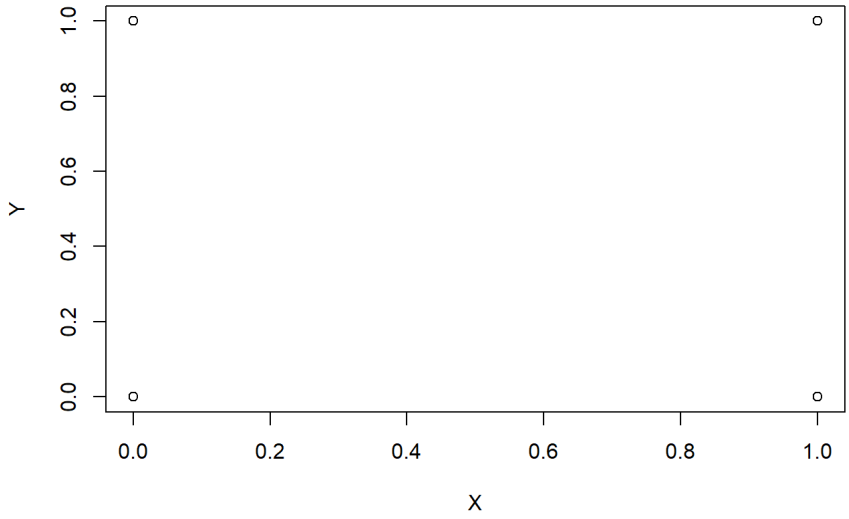
Rstudio (https://qiita.com/hujuu/items/ddd66ae8e6f3f989f2c0) * R,reproducibility,再現性でウェブ検索して情報を集めてみる * この講義のノートを書いてみる。うまく行かなければコピーペーストしてから考える

4-1 帰無仮説、データシミュレーション作成、一様分布

ランダムにn個の値(0か1)を発生させる。 それとは別に、やはりランダムにn個の値(0か1)を発生させる。

2x2表を作る。 カイ二乗検定をする。

```
n <- 10000
X <- sample(c(0,1),n,replace=TRUE)
Y <- sample(c(0,1),n,replace=TRUE)
plot(X,Y)
```




```
table(X,Y)
```

```
##    Y
## X     0    1
##   0 2466 2487
##   1 2507 2540
```

```
tab <- table(X,Y)
chisq.test(tab)
```

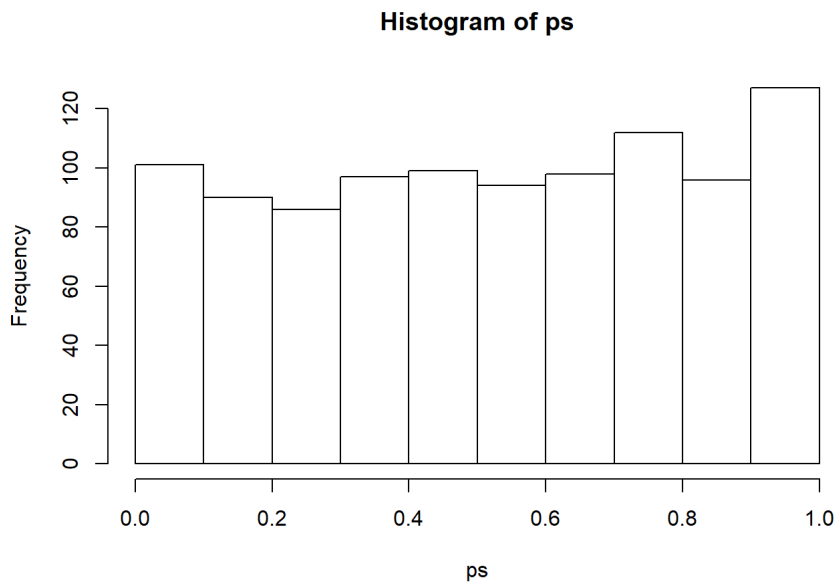
```
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  tab
## X-squared = 0.0090116, df = 1, p-value = 0.9244
```

- p値ってなんだっけ？
- カイ二乗値ってなんだっけ？
- 自由度ってなんだっけ？

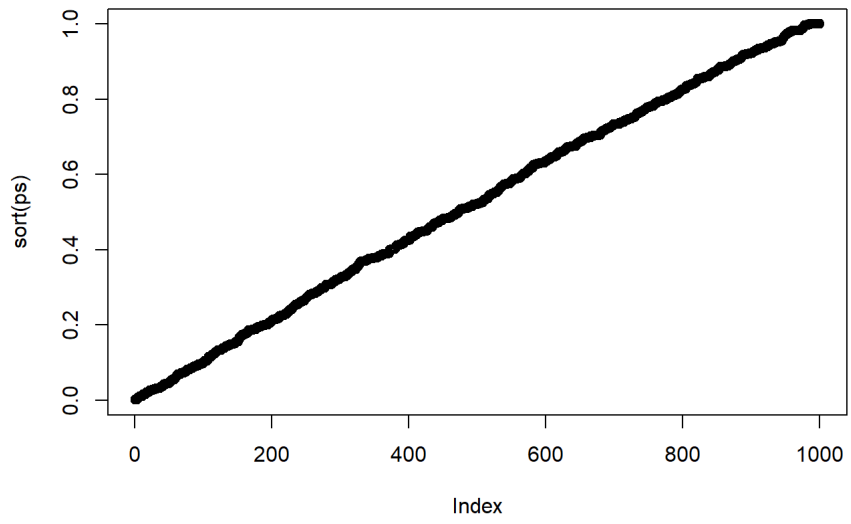
たくさんのp値を発生させてみる。

```
n.iter <- 1000
ps <- rep(0,n.iter)
for(i in 1:n.iter){
  X <- sample(c(0,1),n,replace=TRUE)
  Y <- sample(c(0,1),n,replace=TRUE)
  tab <- table(X,Y)
  chisqout <- chisq.test(tab)
  ps[i] <- chisqout$p.value
}
```

```
hist(ps)
```




```
plot(sort(ps))
```



4-2 2x2表のカイ二乗検定とフィッシャーの正確確率検定

黒板を使う講義

Wikipediaの記事
(https://ja.wikipedia.org/wiki/%E3%83%95%E3%82%A3%E3%83%83%E3%82%B7%E3%83%A3%E3%83%BC%E3%81%AE%E6%AD%A3%E7%A2%BA%E7

小さい表を使って、周辺度数を共有するテーブルを全列挙してみる。

正確生起確率を計算してみる。

```
my.exact.prob <- function(tab){
  a <- tab[1,1]
  b <- tab[1,2]
  c <- tab[2,1]
  d <- tab[2,2]
  ab <- a + b
  cd <- c + d
  ac <- a + c
  bd <- b + d
  n <- sum(tab)
  bunbo <- factorial(n) * factorial(a)*factorial(b)*factorial(c)*factorial(d)
  bunsi <- factorial(ab)*factorial(cd)*factorial(ac)*factorial(bd)
  prob <- bunsi / bunbo
  return(prob)
}
```

足し合わせてみる。

```
tab1 <- matrix(c(1,1,1,0),2,2)
tab1
```

```
##      [,1] [,2]
## [1,]    1    1
## [2,]    1    0
```

```
tab2 <- matrix(c(2,0,0,1),2,2)
my.exact.prob(tab1)
```

```
## [1] 0.6666667
```

```
my.exact.prob(tab2)
```

```
## [1] 0.3333333
```

自由度って...

```
fisher.test(tab)
```

```
##
##  Fisher's Exact Test for Count Data
##
## data:  tab
## p-value = 0.6456
## alternative hypothesis: true odds ratio is not equal to 1
## 95 percent confidence interval:
##  0.9410504 1.1025298
## sample estimates:
## odds ratio
##   1.018612
```

5-1 SNVだけで学ぶ統計学　イントロダクション

スライド (http://www.statgenet.med.kyoto-u.ac.jp/wiki_tokyo/index.php/%E6%97%A5%E6%9C%AC%E4%BA%BA%E9%A1%9E%E9%81%BA%E4%BC%9D%E5%AD%A6%E4%BC%9A%E6%95%99%

5-2 アレル、ディプロタイプ、ハーディワインベルグ平衡

- SNVは２アレル
- ディプロタイプは３タイプ
- アレル頻度
- ホモ接合体、ヘテロ接合体、その頻度

黒板を使う

- 赤球と白球の集まりからランダムに２個取り出す。
- 赤球と白球の集まりからランダムにペアを作る。
- できたペアは(赤、赤)、(赤、白)、(白、白)の３通り。
- その割合は？

6-1 ケースコントロール・スタディ 2x3表

スライド (http://www.statgenet.med.kyoto-u.ac.jp/wiki_tokyo/index.php/%E6%97%A5%E6%9C%AC%E4%BA%BA%E9%A1%9E%E9%81%BA%E4%BC%9D%E5%AD%A6%E4%BC%9A%E6%95%99%
6-2 ケースコントロール・スタディ 2x3表 引き続きスライド

7-1 確率と尤度

黒板を使う

確率

- 赤球と白球
- p:1-p
- n個取り出す
- k:n-kになる確率は？

$$n\_C\_k\ p^k\ (1-p)^{n-k}$$

尤度

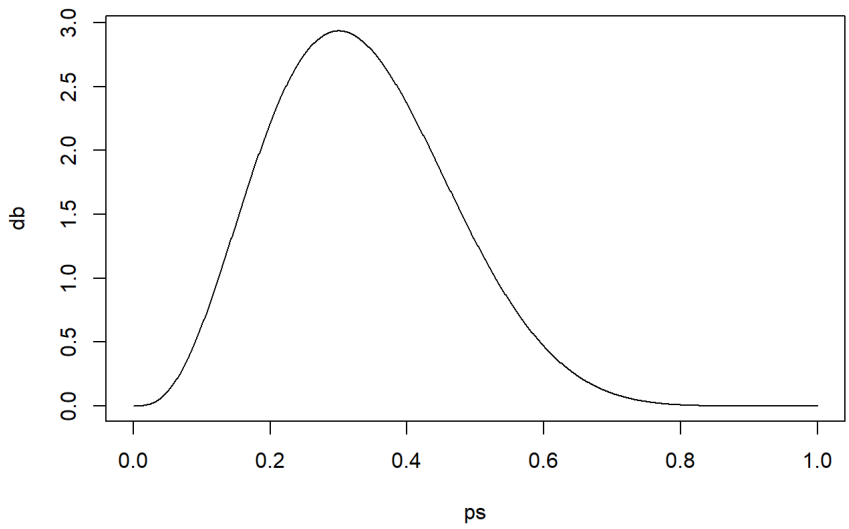
- n個のうちわけがk:n-kだったとする
- pが0.2と仮定したときにk:n-kになる確率は？
- pが0.3と仮定したときにk:n-kになる確率は？
- それぞれを、k:n-kという観測を見て想定する、pがxxxである尤度、と言う
- 尤度の計算は確率そのものだから、言い方・捉え方の違いがあるだけで同じもの

7-2 最尤推定と二項分布とベータ分布

- 尤度を考え合わせて、その総和が1になるように調整する
- それは確率密度(質量)分布
- 二項分布とベータ分布とは「確率」と「尤度」との相互関係
- あるパラメタの値が尤度を最大にする。そのような値を最尤推定値という

Rでやってみる。

```
ps <- seq(from=0,to=1,length=1000)
n <- 10
k <- 3
db <- dbeta(ps,k+1,n-k+1) # "+1"はおまじない
plot(ps,db,type="l")
```



- まだ何も観察されていないときの尤度はどうなる？
- n=1のときの尤度はどうなる
- 最尤推定値と期待値
- 最尤ではない推定値、分布推定、ベイズ

8-1 ハプロタイプと連鎖不平衡

2SNVsのハプロタイプ

黒板を使う

- 2SNVsハプロタイプの成立
- 変異
- 第二変異
- 交叉組み換え

8-2 連鎖不平衡係数とEMアルゴリズム

連鎖不平衡係数

- D'とr^2
- r^2 とカイ二乗値

連鎖平衡と独立と期待値

- ２つのSNV
- 相互に独立な（連鎖平衡な）２SNVの組み合わせハプロタイプ頻度と2x2表
- ２SNVのアレル頻度は同じで、4ハプロタイプ頻度を変える

複合ジェノタイプ

- 2SNVが作る複合ジェノタイプは9タイプ
- ２SNVハプロタイプのハーディ・ワインベルグ平衡

黒板を使う

4ハプロタイプ頻度と９複合ジェノタイプ頻度

黒板を使う

- 9複合ジェノタイプ観測データの下での、４ハプロタイプ頻度の最尤推定
- EMアルゴリズム
- 手作業でのEMアルゴリズム

アルゴリズムとは

- 統計学 vs. アルゴリズム（機械学習）
- 「計算的」か「機械的」か
- p値を気にするか、「答え」を気にするか
- 機械的に」答えがいつも一意に定まるアルゴリズム
- 「機械的に」答えが決まるが、条件・初期値依存であるアルゴリズム
- 「機械的に」答えが決まるが、乱数を使うので、答えが一定ではないアルゴリズム

9-1 多重検定とボンフェロニ補正

スライド (http://www.statgenet.med.kyoto-u.ac.jp/wiki_tokyo/index.php/%E6%97%A5%E6%9C%AC%E4%BA%BA%E9%A1%9E%E9%81%BA%E4%BC%9D%E5%AD%A6%E4%BC%9A%E6%95%99%

黒板を使う

- ボンフェロニ法の仮定
- 独立の仮定
- 独立ではない場合
- 独立ではない場合にボンフェロニ法を適用するとどうなるか

9-2 多重検定とパーミュテーション法(と正確確率法)

- 独立を仮定できない場合
- 連鎖不平衡にある複数のSNV（GWSAを含む）
- 1つの2x3表に３つの検定（相加・優性・劣性の３モデルで検定する場合)
- ３つの群を比較するために、３つのペアの検定をする場合
- パーミュテーション法
- 乱数を使う（モンテカルロ）
- 帰無仮説に基づいて、観測データをシャッフルする
- シャッフルを繰り返して検定をすれば、「帰無仮説の下で」多重検定をしたときに、得られるp値の分布と似たようなものが得られるのではないか？
- 全部の順列、フルのパーミュテーション法
- できうる限りのすべてのシャッフルパターンを１回ずつ行って、「帰無仮説の下で」の分布を得る
- 正確確率検定はこの方法

パーミュテーション法とリサンプリング方

- パーミュテーション法は得られた標本の順番を入れ替えるだけ
- リサンプリング法は、得られた標本から「リ・サンプル」して、新たな標本セットを作る方法。１回ではなく複数回、サンプリングされるかもしれない
- リサンプリング法の流儀は、いわゆる「機械学習」の手法として位置づけられている

10-1 次世代シークエンサーデータとリードのマッピング

そろそろ疲れが…

休憩代わりに、読み物(次世代シークエンサーデータ) (http://ryamada22.hatenablog.jp/entry/20140121/1390273733)

10-2 デプスとSNV

ゲノムシークエンス

黒板を使う

- SNVかどうかを調べるには

ヘテロ座位でのアレル別リード数

```
dp <- 40
prob <- c(0.5,0.5)
gs <- sample(0:1,dp,replace=TRUE,prob=prob)

table(gs)
```

```
## gs
##  0  1
## 21 19
```

ホモ座位でのアレル別リード数

全リードが同じアレルになる「はず」

ただし、エラーがなければ、の話

```
dp <- 40
# エラーを考慮
err <- 0.1
prob <- c(1-err,err)
gs <- sample(0:1,dp,replace=TRUE,prob=prob)

table(gs)
```

```
## gs
##  0  1
## 34  6
```

観測から、ホモ接合体かヘテロ接合体かを判断する

アレル別リード数が、4：1 のとき

- データは観測された
- 仮説１「ホモ接合体である」
- 仮説２「ヘテロ接合体である」
- 仮説１と仮説２の尤度を計算する
- 片方の尤度が圧倒的に高ければ、その仮説を信じればよい
- 圧倒的に高いとは？

トランスクリプトーム

- アレル特異的発現量解析
- ゲノム上、ヘテロであることがわかっているSNVに相当するmRNAシークエンスデータ
- ホモ接合体かもしれなかったらどうする…

一部の細胞だけが変異体である場合

- 癌
- 体細胞モザイク
- デプスの深さの重要性

11-1 SNV以外で学ぶ統計学　イントロダクション

- 博士課程「ゲノム統計学大意」
- 修士課程「ゲノムサイエンス概論」

SNVの統計学しかやってこなかったけど、大丈夫なのか？

- 大丈夫
- その１：遺伝情報の臨床利用の半分以上はSNV関連だから大丈夫
- その２：以下のことについての基本知識は、ほぼすべてのゲノム・オミックス統計解析の基礎になっているから大丈夫
  - 検定とp値
  - 尤度とベイズ的考え方
  - 独立と帰無仮説と期待値との関係
  - 多重検定、検定が相互に非独立であること
  - 推定とサンプル数の数の必要性（パワー）
  - パーミュテーション法、リサンプリング法、モンテカルロ法
  - 統計学的考え方と機械学習的な考え方の違いと重なり
- その３：以下のスキルを習得した
  - Rが使える、Rmdで「再現性のあるデータサイエンス」の記録が取れる

11-2 トランスクリプトームとヒートマップ

トランスクリプトームデータ解析のイントロダクション

- たくさんの個体・個別細胞
- たくさんの遺伝子
- 長方形データ（行列データ）
- 何が知りたい？

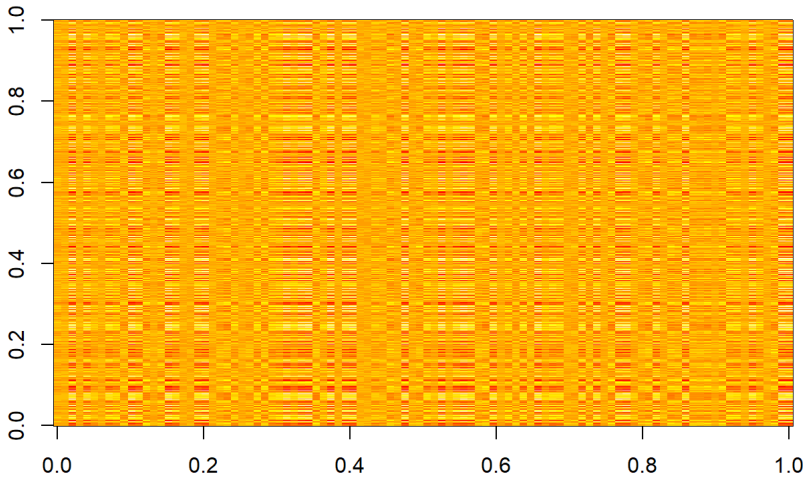
- 遺伝子をグループ分けしたい
- 個体・個別細胞をグループ分けしたい
- 個体・個別細胞の表現型と、多数の遺伝子が織りなす様子（signature シグナチャとか呼ばれる）を『把握したい』

『把握したい』という気持ちの本性

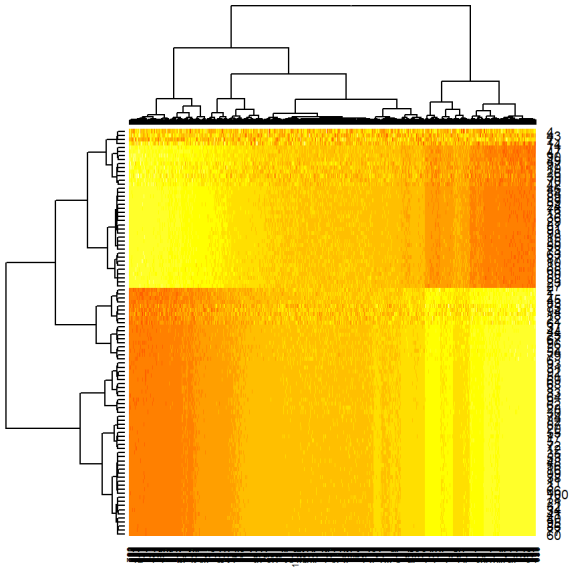
- 簡潔にしたい
- 少ない数値で分かったことにする
    - 平均値
    - 平均値と分散
    - ２つの数値で分かったことにする
- 『当てはめ』てわかったことにする
    - 平均値と分散とを、正規分布のそれ、とみなせば、『正規分布』を用いた『パラメトリック手法』に当てはめて簡潔にしている
- 次元を下げる
- 高次元データは「見えない」
- 「見えない」ものを人生で理解しようとしたことがないから
- 「見える」ようにする
- 簡潔にする　＝　すべてを理解することをあきらめる
- 知りたい側面のみを取り出す
- 主要な部分のみを取り出す
- 出力が「一般人的に」わかりやすいように取り出す
- 出力が「数学的・機械的に」わかりやすいように取り出す
- 簡潔にする方法はたくさんある
- それぞれの方法は、それぞれの方法の「気持ち」に基づいて「正しい」

12-1 クラスタリング、ヒートマップ

```
nsample <- 100
ngene <- 1000
S <- sin(sample(nsample))+0.4
G <- sin((sample(ngene))/10)^3
X <- matrix(S,ncol=1) %*% matrix(G,nrow=1)
X <- X + rnorm(length(X),0,sd(X)*0.1)
image(X)
```




```
heatmap(X)
```



12-2 系統樹というクラスタリング

近隣結合法 Neighbor-Joining法を手でやる

Wikipediaの記事 (https://en.wikipedia.org/wiki/Neighbor_joining)

図を見よう

- ある２点を選び、「まとめ」ている
- 「まとめ」ると、辺の長さの総和が短くなっている
- どの２点を「まとめる」かを決めるルールを決めれば、あとは繰り返し
- 繰り返しにするための処理を決める
- この「繰り返し」がアルゴリズムでは大事
- 少ないルールで着実に処理できるから

式を読もう

- Q(a,b)の式
- この式の値が最小となる(a,b)をペアとして採用する
- Q(a,b)は、どのようなときに小さくなるかを考えよう
- d(a,b)は２点の距離。これは短い方がQ(a,b)は小さくなる
- $\sum_{k=1}^{n} d(a,k)$ は、aから、他のすべての点への距離の総和。これが大きければQ(a,b)は小さくなる
- 結局、全体から離れている点が選ばれやすく
- ペアとしては、近い者同士が選ばれやすい

アルゴリズム

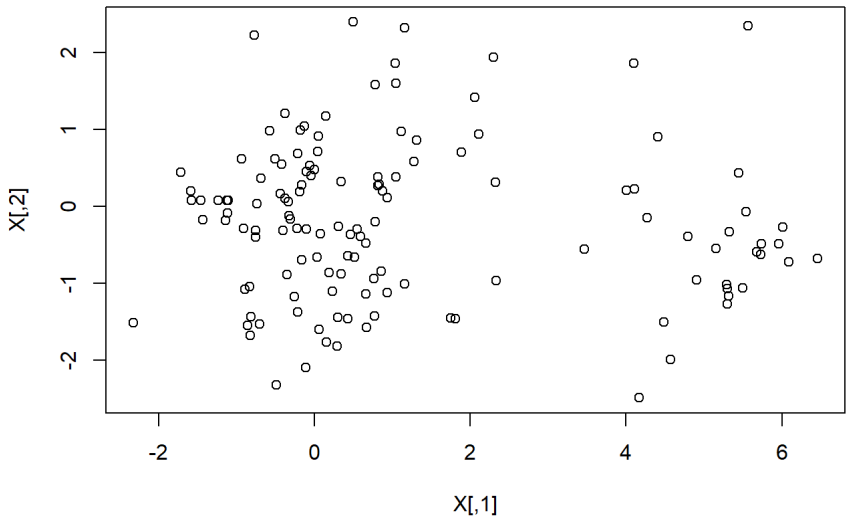
アルゴリズムとは、「手順」。手でやればできるけれど、機械にやらせることが多い： * 自分でやるのは面倒くさいから * やれるとは言っても、たくさんありすぎて手でやることは非現実的だから * 人がやると間違うけれど、機械は間違えないから

13-1 階層的クラスタリングと非階層的クラスタリング

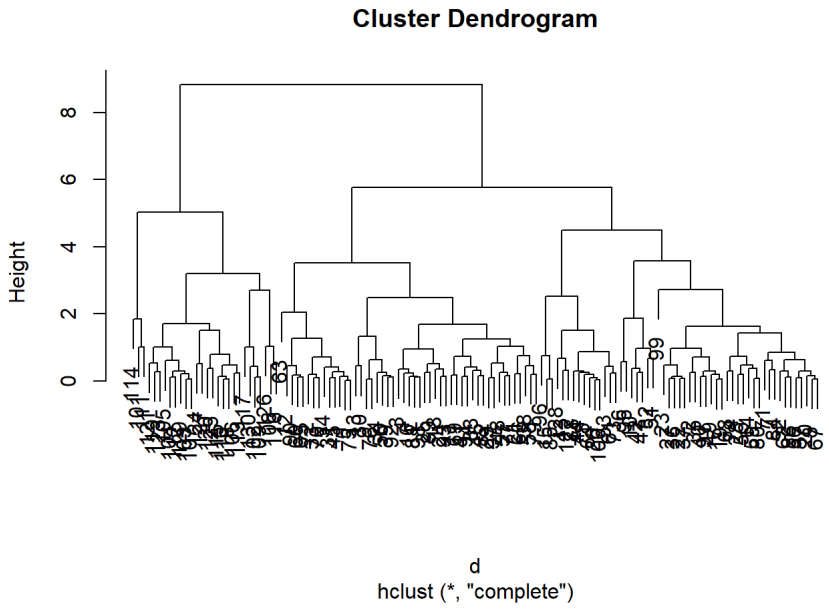
階層型的

```
n1 <- 100
n2 <- 30
nsample <- n1 + n2
X <- matrix(0,nsample,2)
X[1:n1,] <- cbind(rnorm(n1),rnorm(n1))
X[(n1+1):nsample,] <- cbind(rnorm(n2)+5,rnorm(n2+0.5))

plot(X)
```




```
d <- dist(X)
cl <- hclust(d)
plot(cl)
```



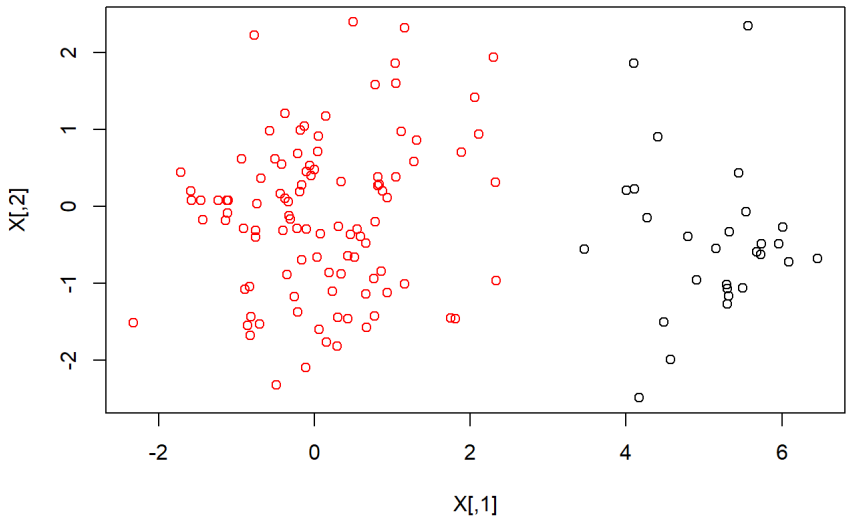
- いくつに分けられた？

非階層的

- いくつに分ける？

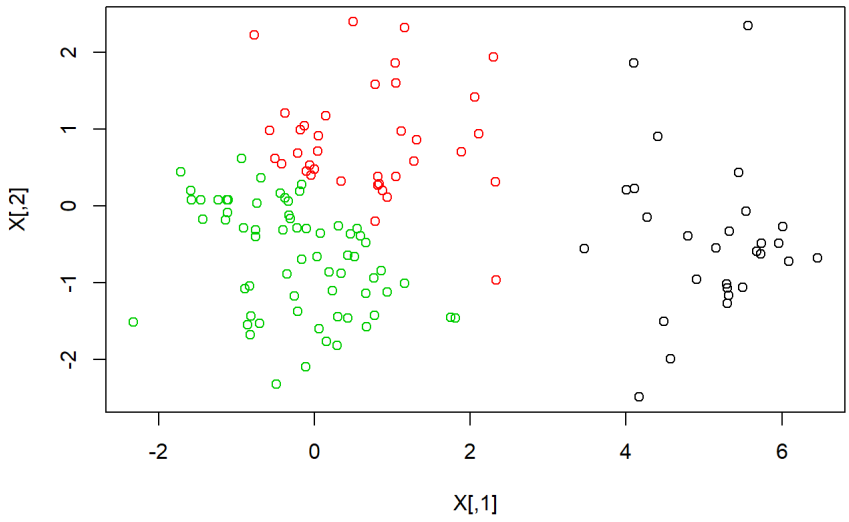
２つに分ける

```
n.gr <- 2
cl2 <- kmeans(X,centers=n.gr)
plot(X,col=cl2[[1]])
```



3つに分ける

```
n.gr <- 3
cl2 <- kmeans(X,centers=n.gr)
plot(X,col=cl2[[1]])
```



- 非階層的クラスタリングでは、いくつに分けるかを決めるのが課題になる
- 「見えていれば」決めやすいが、遺伝子数 = 1000のデータではどうする？

13-2 ネットワークというグラフ

https://ja.wikipedia.org/wiki/%E3%82%B0%E3%83%A9%E3%83%95_(%E3%83%87%E3%83%BC%E3%82%BF%E6%A7%8B%E9%80%A0) (グラフ)とは

特別なグラフ～木グラフ

- 階層的クラスタリングの出力は木グラフ
- 系統樹も木グラフ
- 木グラフは解釈が簡単
- 木グラフはコンピュータ処理も簡単

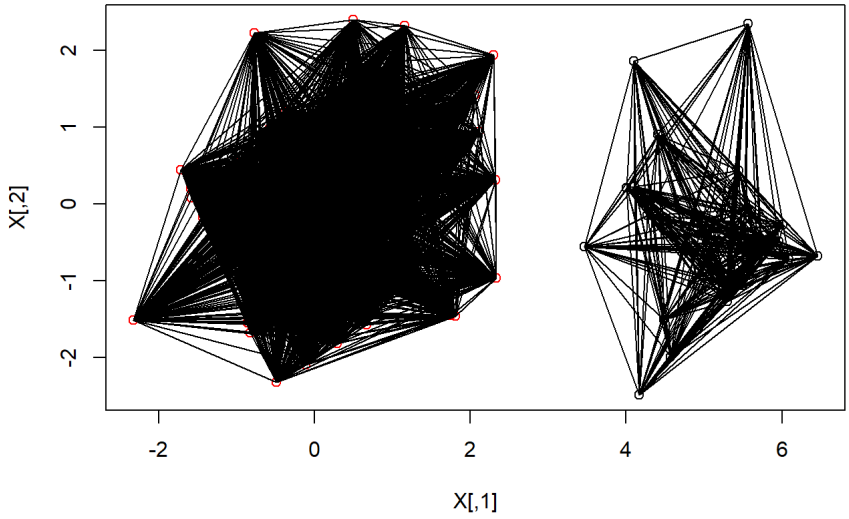
普通のグラフ～ネットワーク

- グラフは頂点と辺でできている
- どの頂点のペアに辺を引くか・引かないかで決まる
- たくさんの遺伝子があるとき、遺伝子ペアについて、「互いに近いペア」「そうでないペア」に分けて辺の引き分けルールにすれば、それだけで「遺伝子遠近グラフ」になる
- ネットワークもグラフ。『人的ネットワーク』とか『インターネットのネットワークの様子』というように、連携具合を意識した時に「ネットワーク」という単語を使うことが多くなる

2群に分けて、同じグループになったら辺で結ぶことにしてみる。

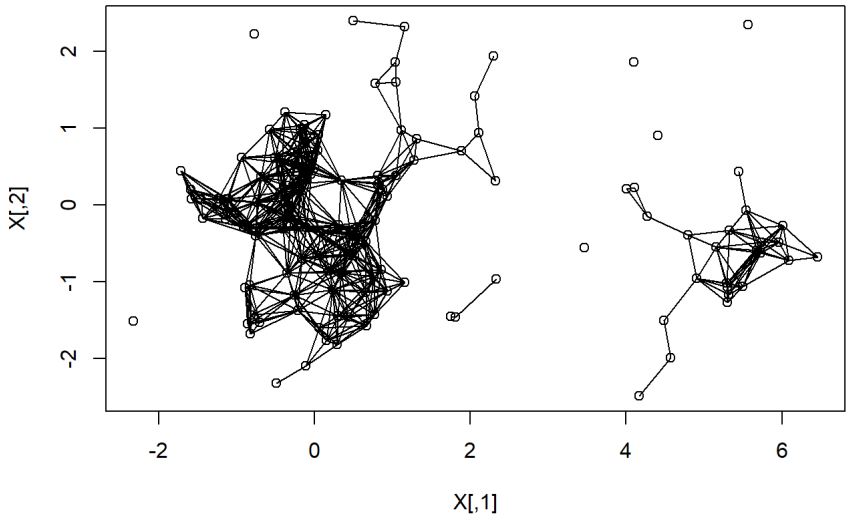
```
n.gr <- 2
cl2 <- kmeans(X,centers=n.gr)
plot(X,col=cl2[[1]])

for(i in 1:length(X[,1])){
  for(j in 1:length(X[,2])){
    if(cl2[[1]][i]==cl2[[1]][j]){
      segments(X[i,1],X[i,2],X[j,1],X[j,2])
    }
  }
}
```



距離が平均距離の1/4より短いペアを結んでみる。

```
plot(X)
d <- as.matrix(dist(X))
mean.d <- mean(d)
for(i in 1:(length(X[,1])-1)){
  for(j in (1+i):length(X[,1])){
    if(d[i,j] < mean(d)/4){
      segments(X[i,1],X[i,2],X[j,1],X[j,2])
    }
  }
}
```




```
library(igraph)
```

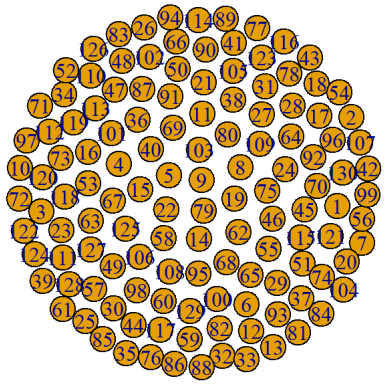
```
## Warning: package 'igraph' was built under R version 3.5.3
```

```
##
## Attaching package: 'igraph'
```

```
## The following objects are masked from 'package:stats':
##
##     decompose, spectrum
```

```
## The following object is masked from 'package:base':
##
##     union
```

```
d2 <- d
d2[which(d2 < mean.d/4)] <- 1
d2[which(d2 >= mean.d/4)] <- 0
g <- graph.adjacency(d2)
plot(g)
```



14-1は予備。きっと時間が足りなくなるから ## 14-2は予備。きっと時間が足りなくなるから ## 15-1,15-2 試験・試験