データの分析と知識発見

Introduction to Data Analysis

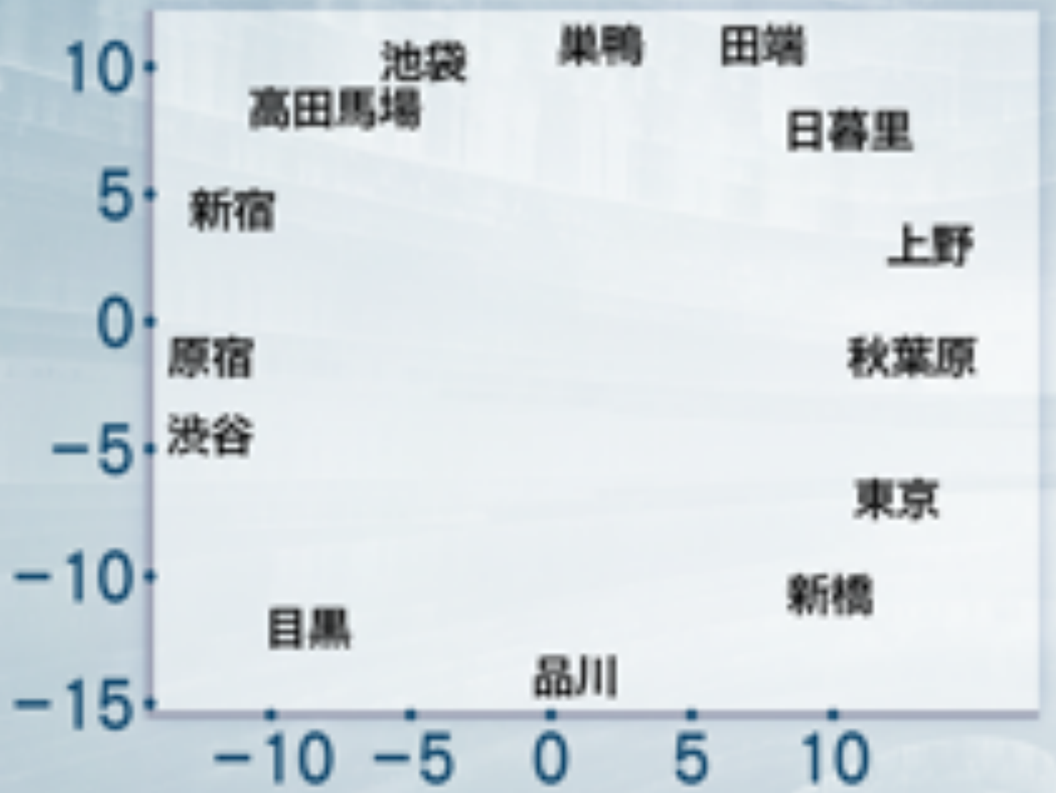
今回の構成

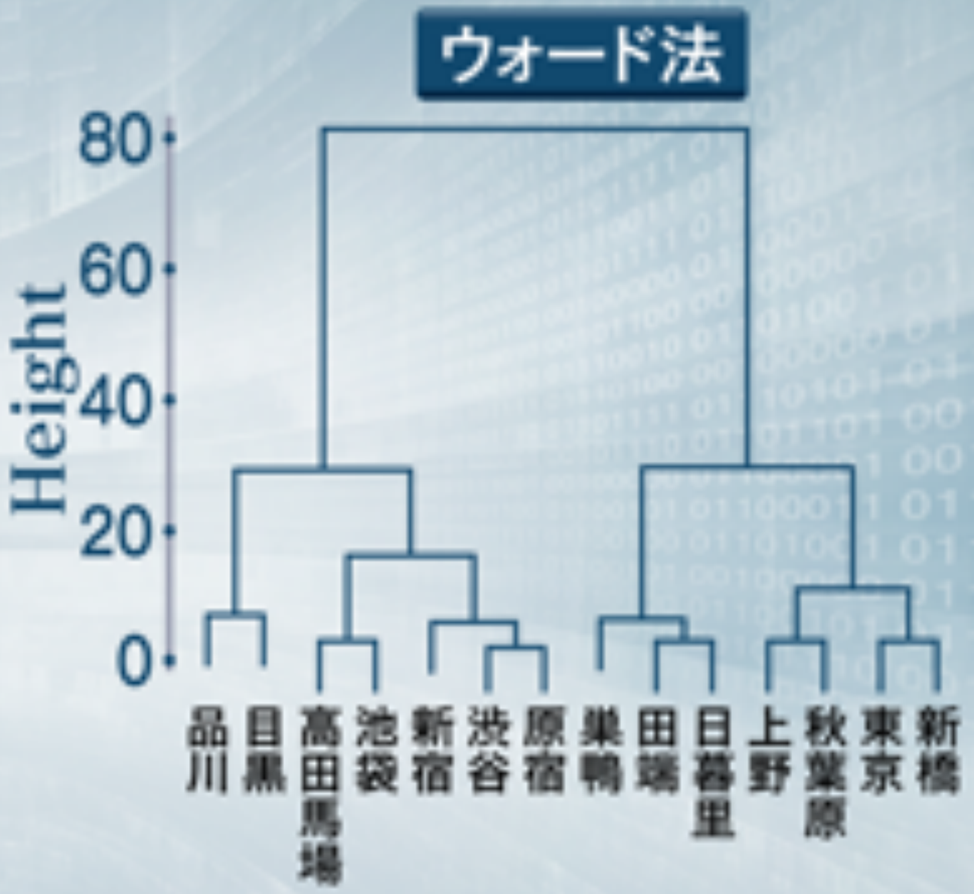
階層的クラスター分析の計算手順を学ぶ

k平均法の計算の手順について学ぶ

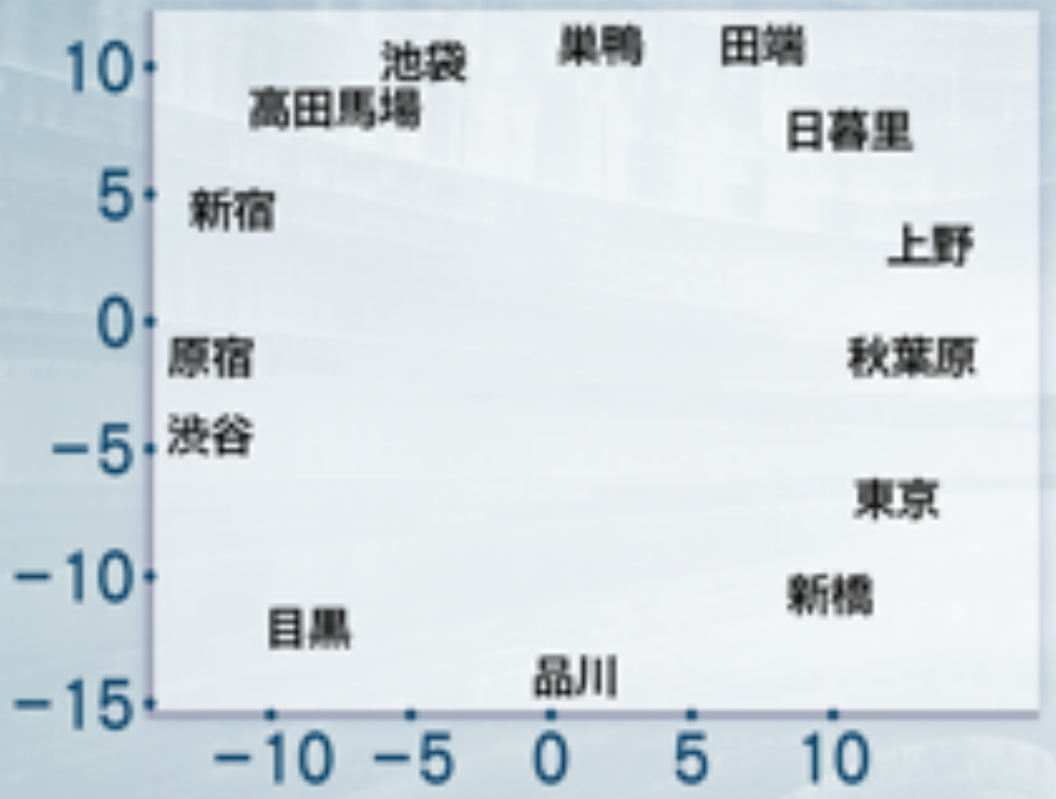
Rを用いてクラスター分析を行う

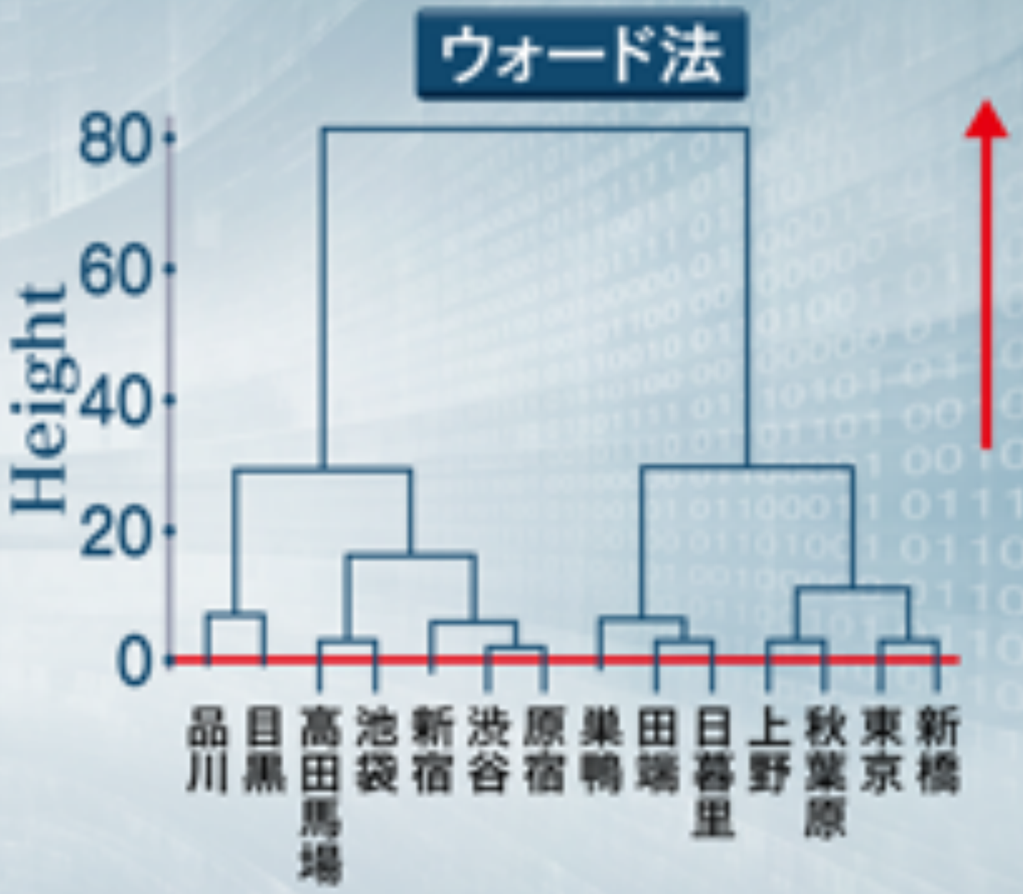
階層的クラスター分析


池袋
巣鴨
田端
高田馬場
日暮里
新宿
上野
原宿
秋葉原
渋谷
東京
新橋
目黒
品川
10
5
0
−5
−10
−15
−10
−5
0
5
10

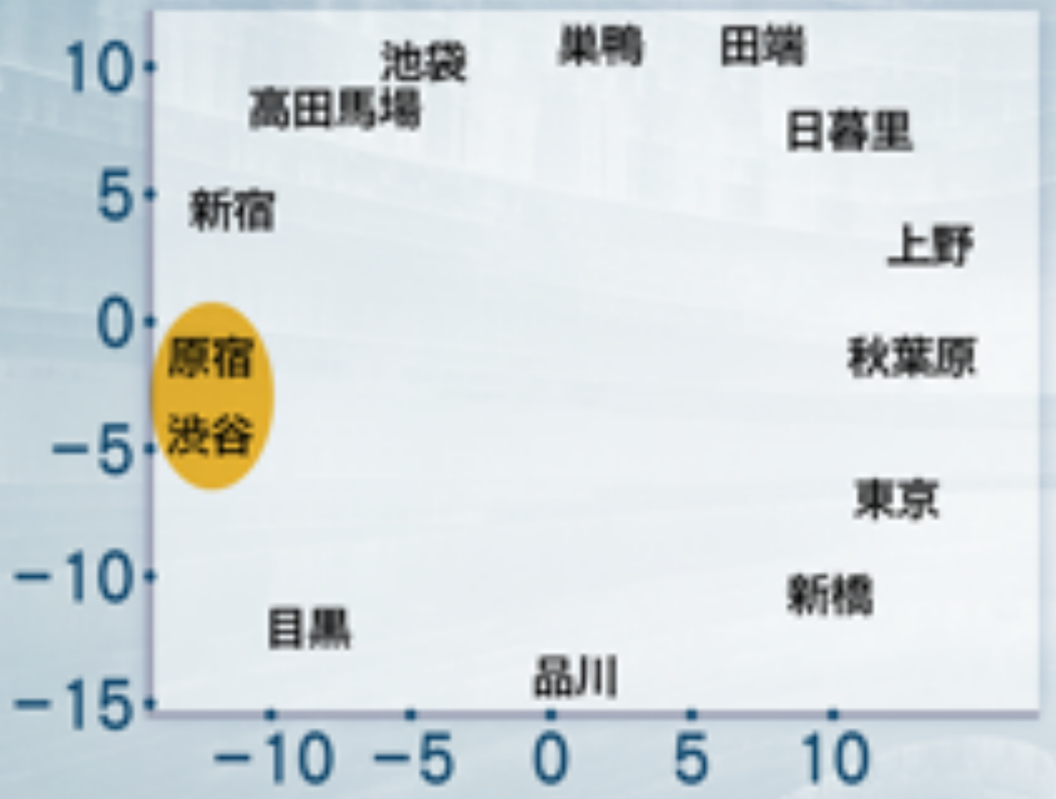

ウォード法
Height
80
60
40
20
0
品川
目黒
高田馬場
池袋
新宿
渋谷
原宿
巣鴨
田端
日暮里
上野
秋葉原
東京
新橋

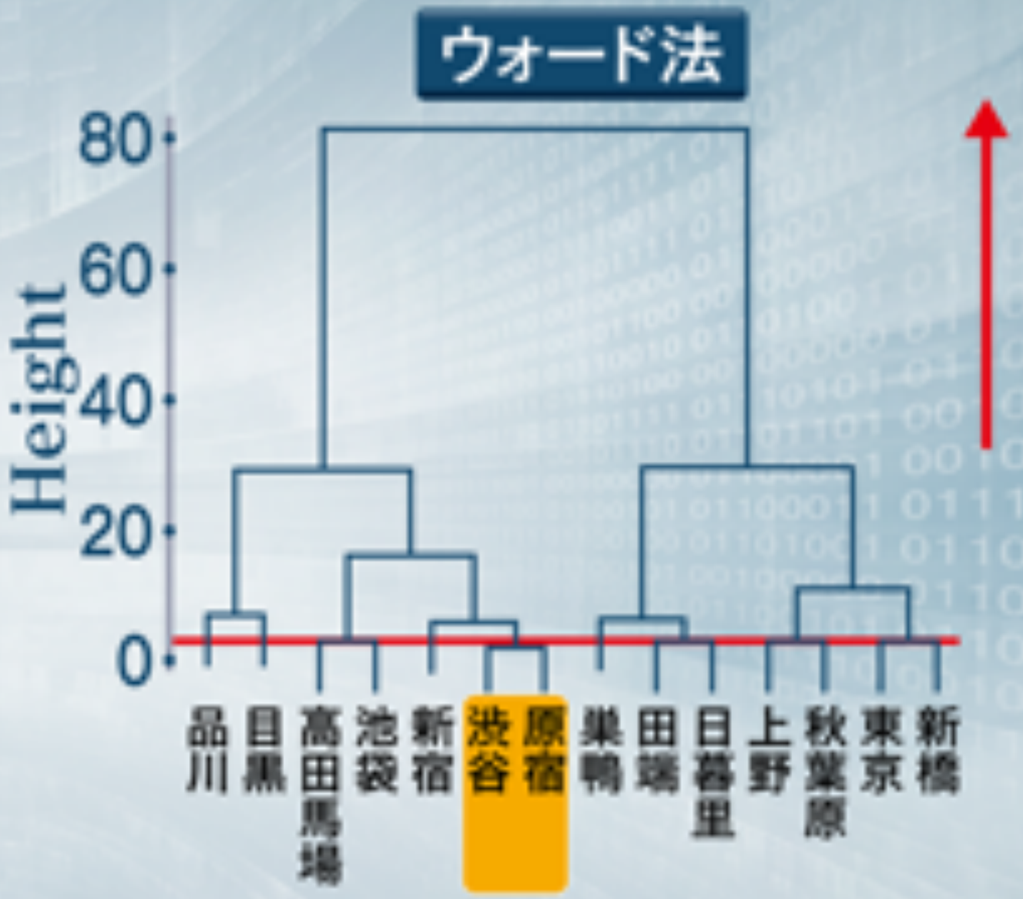
階層的クラスター分析


池袋
巣鴨
田端
高田馬場
日暮里
新宿
上野
原宿
秋葉原
渋谷
東京
新橋
目黒
品川
10
5
0
−5
−10
−15
−10
−5
0
5
10

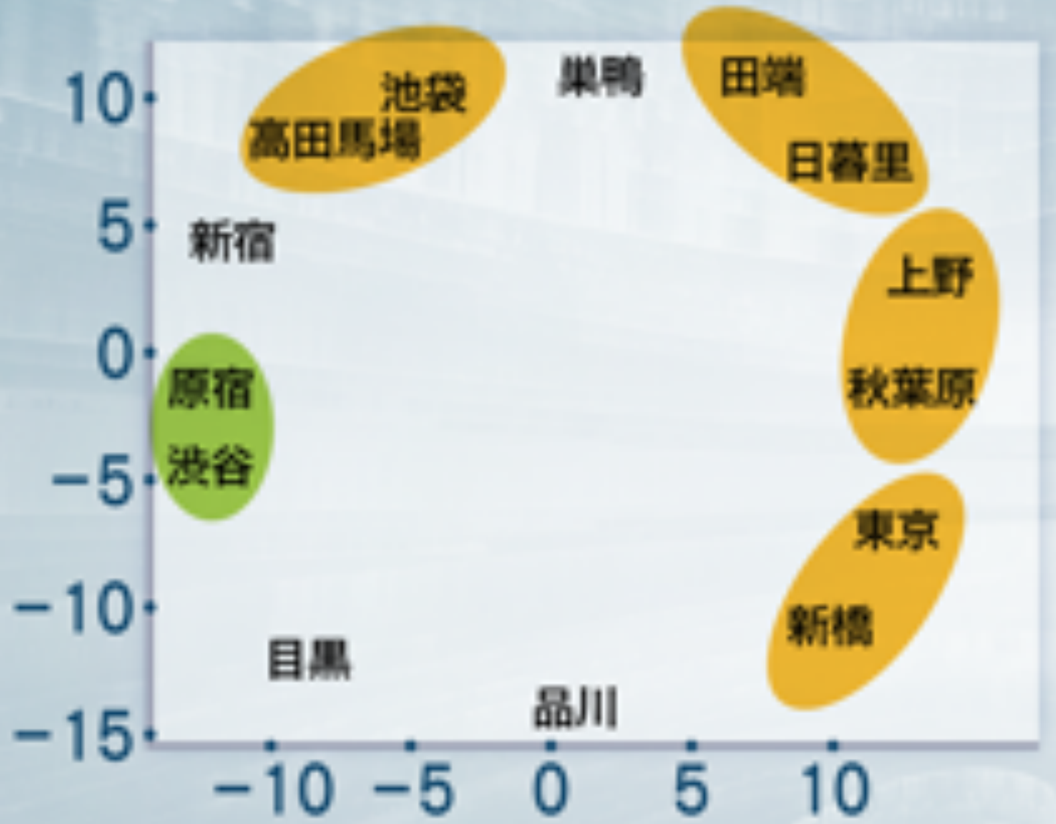

ウォード法
Height
80
60
40
20
0
品川
目黒
高田馬場
池袋
新宿
渋谷
原宿
巣鴨
田端
日暮里
上野
秋葉原
東京
新橋

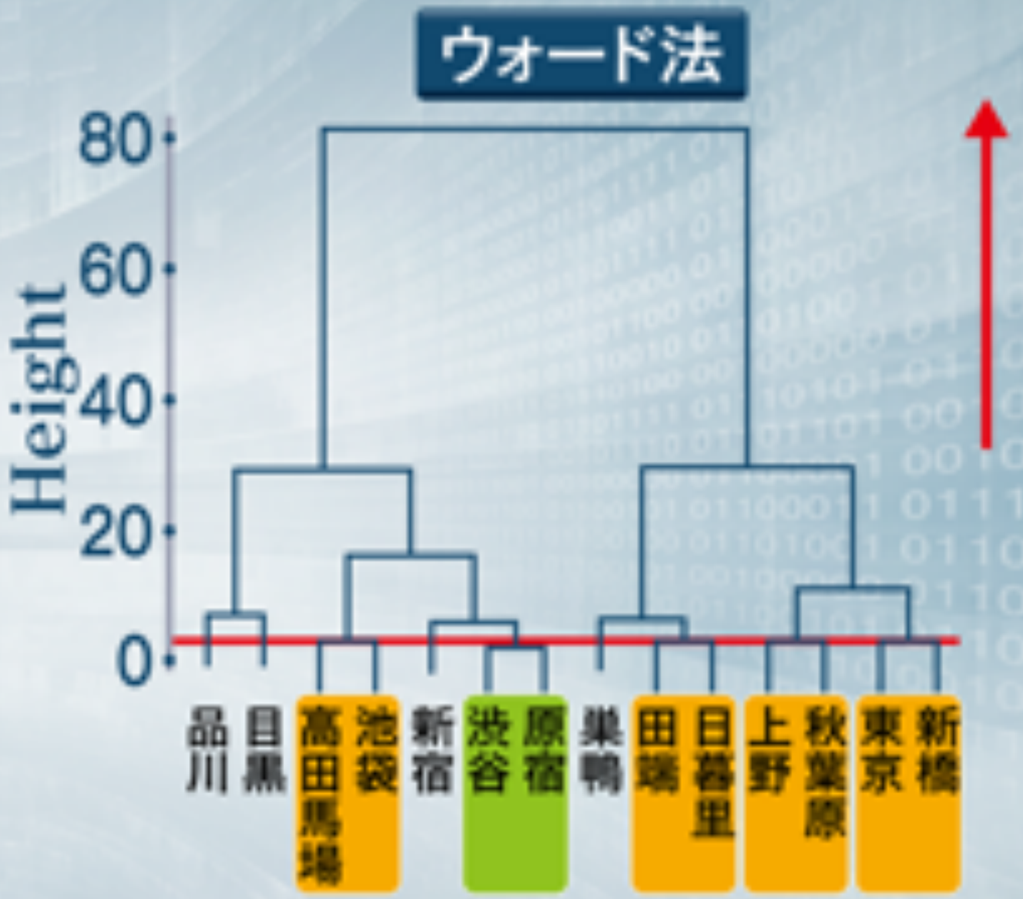
階層的クラスター分析


池袋
巣鴨
田端
高田馬場
日暮里
新宿
上野
原宿
渋谷
秋葉原
東京
新橋
目黒
品川
10
5
0
−5
−10
−15
−10
−5
0
5
10

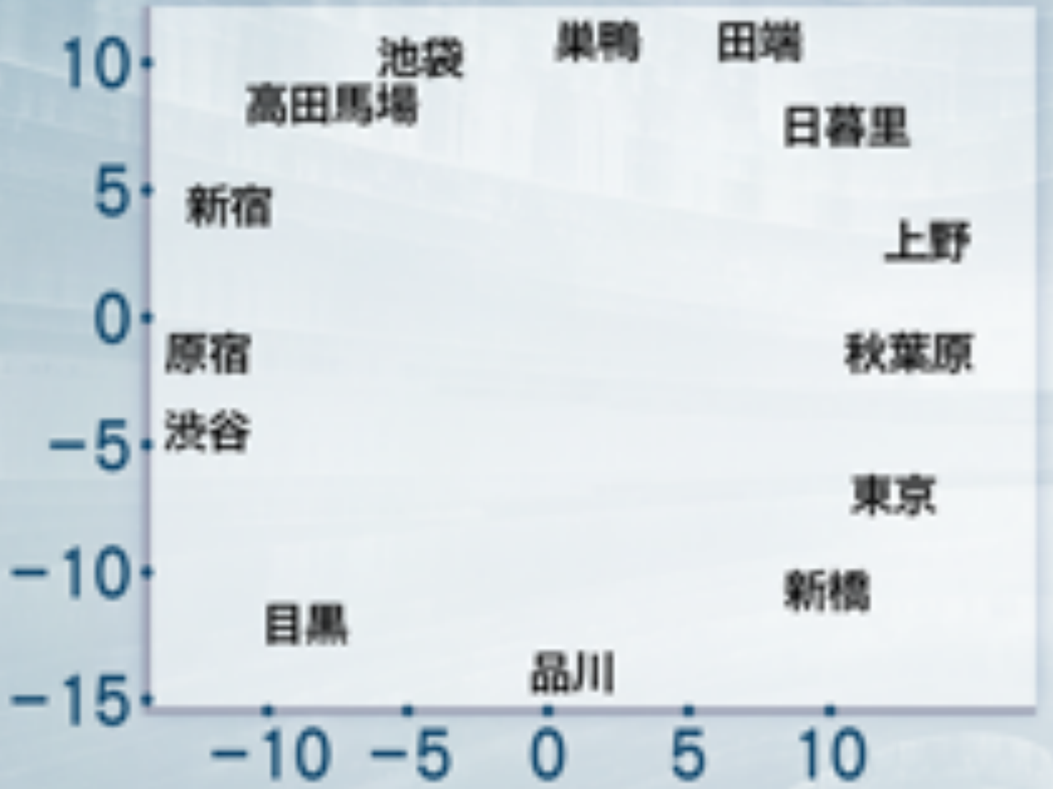

ウォード法
Height
80
60
40
20
0
品川
目黒
高田馬場
池袋
新宿
渋谷
原宿
巣鴨
田端
日暮里
上野
秋葉原
東京
新橋

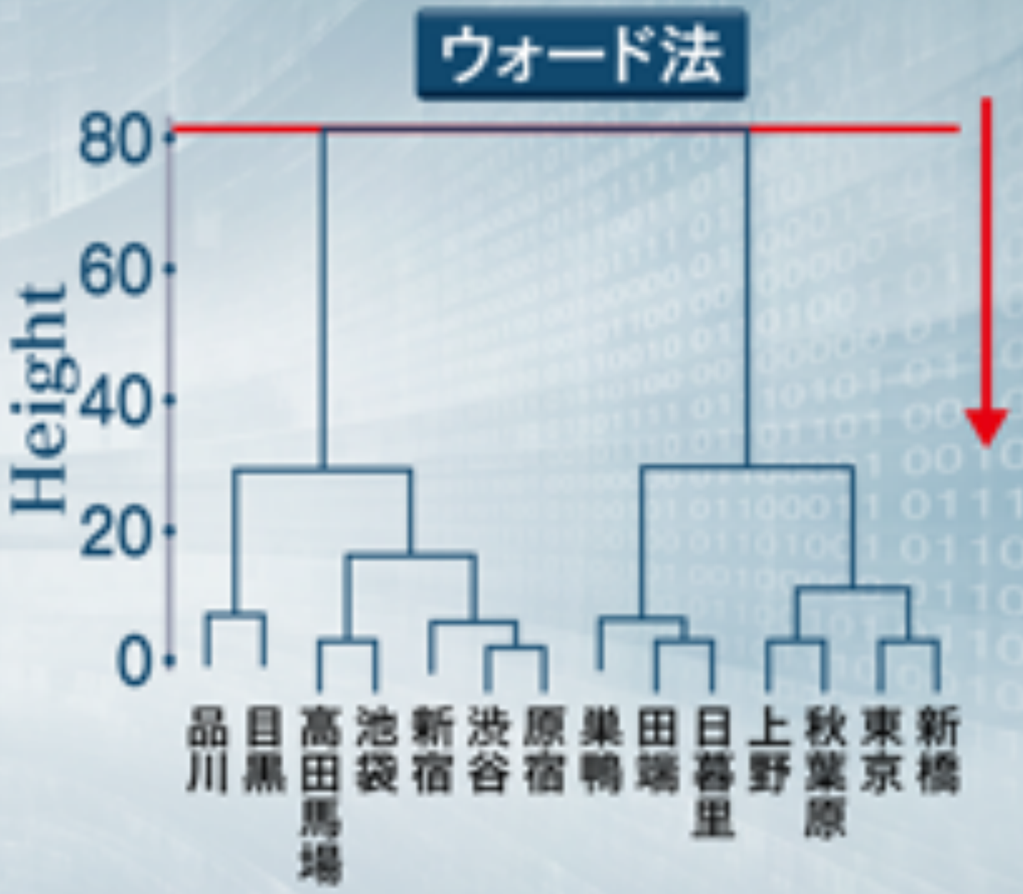
階層的クラスター分析


池袋
高田馬場
巣鴨
田端
日暮里
新宿
上野
秋葉原
原宿
渋谷
東京
新橋
目黒
品川
10
5
0
−5
−10
−15
−10
−5
0
5
10

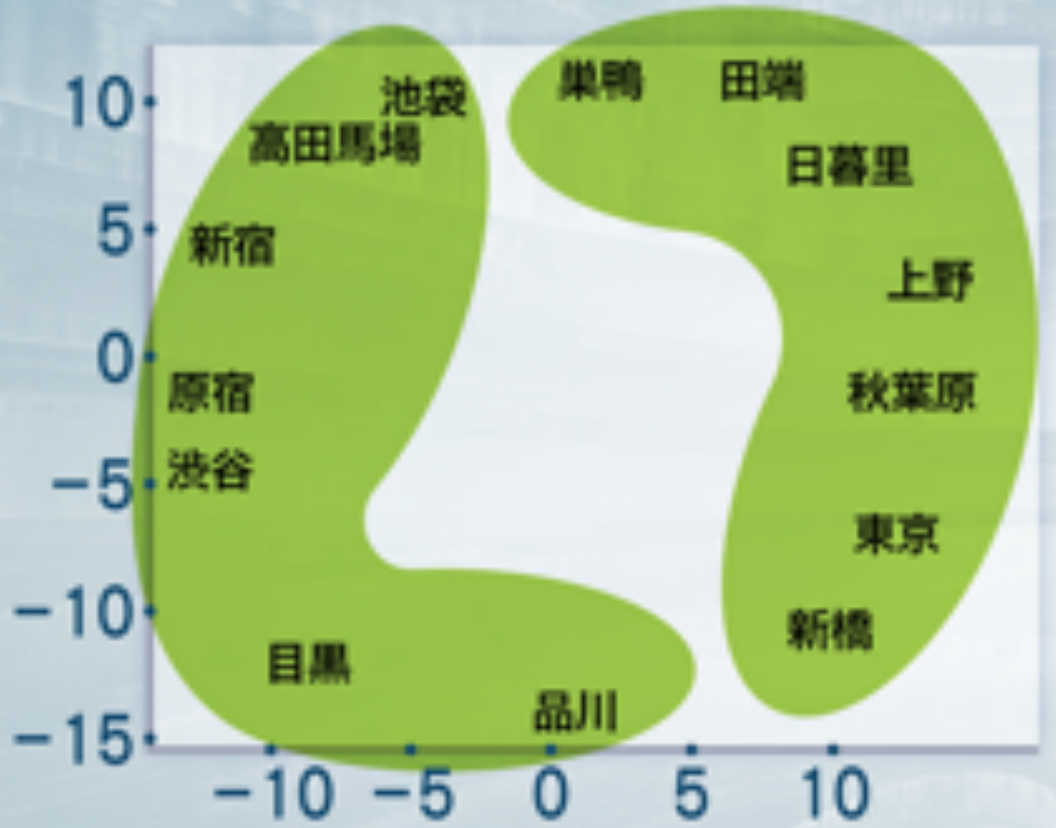

ウォード法
Height
80
60
40
20
0
品川
目黒
高田馬場
池袋
新宿
渋谷
原宿
巣鴨
田端
日暮里
上野
秋葉原
東京
新橋

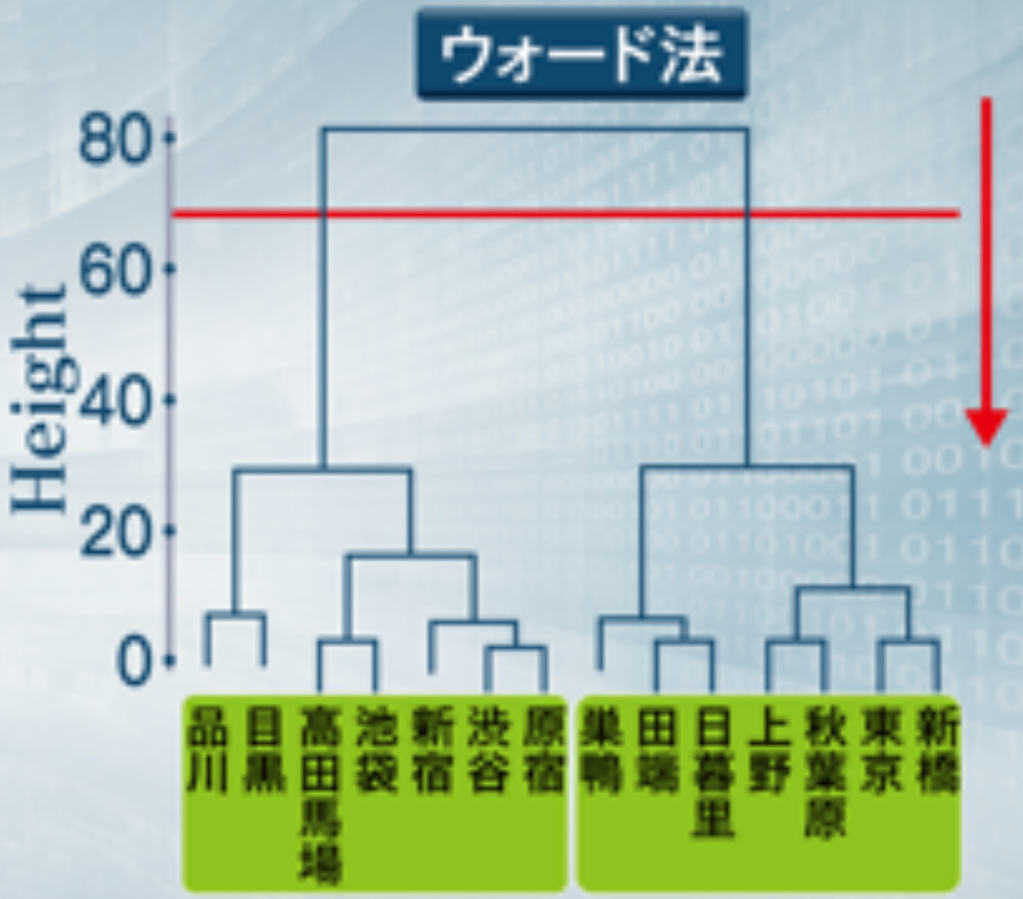
階層的クラスター分析


池袋
巣鴨
田端
高田馬場
日暮里
新宿
上野
原宿
秋葉原
渋谷
東京
新橋
目黒
品川
10
5
0
−5
−10
−15
−10
−5
0
5
10

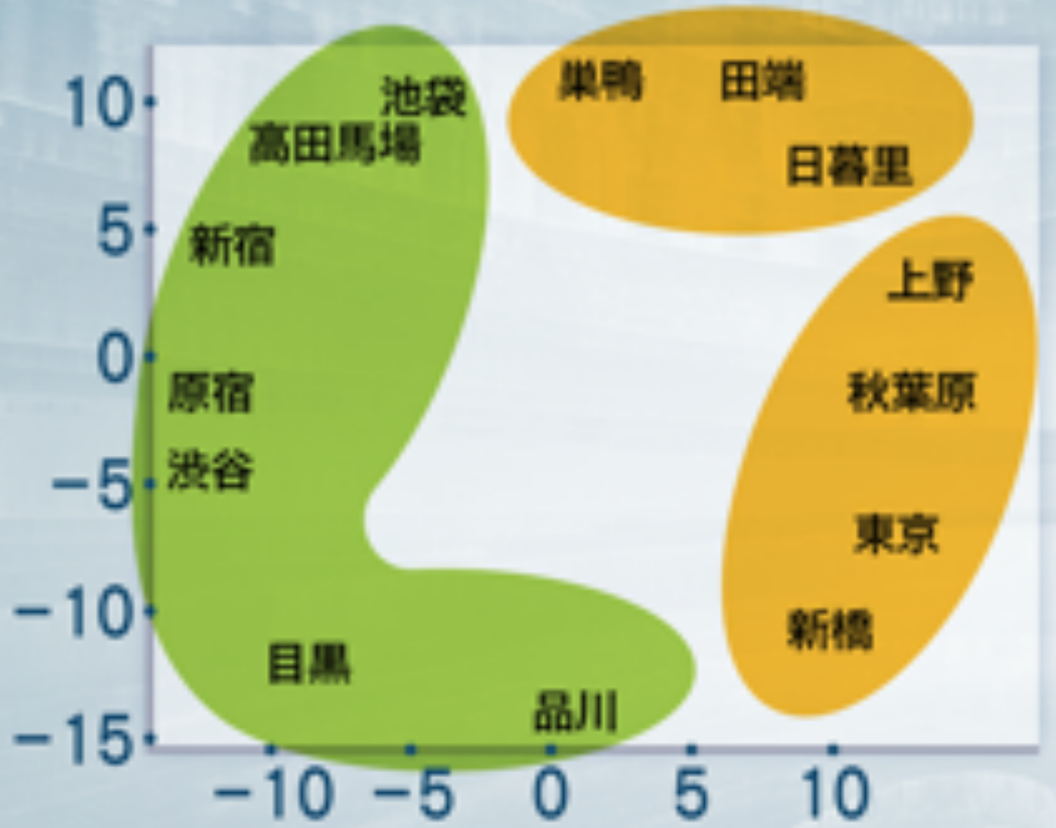

ウォード法
Height
80
60
40
20
0
品川
目黒
高田馬場
池袋
新宿
渋谷
原宿
巣鴨
田端
日暮里
上野
秋葉原
東京
新橋

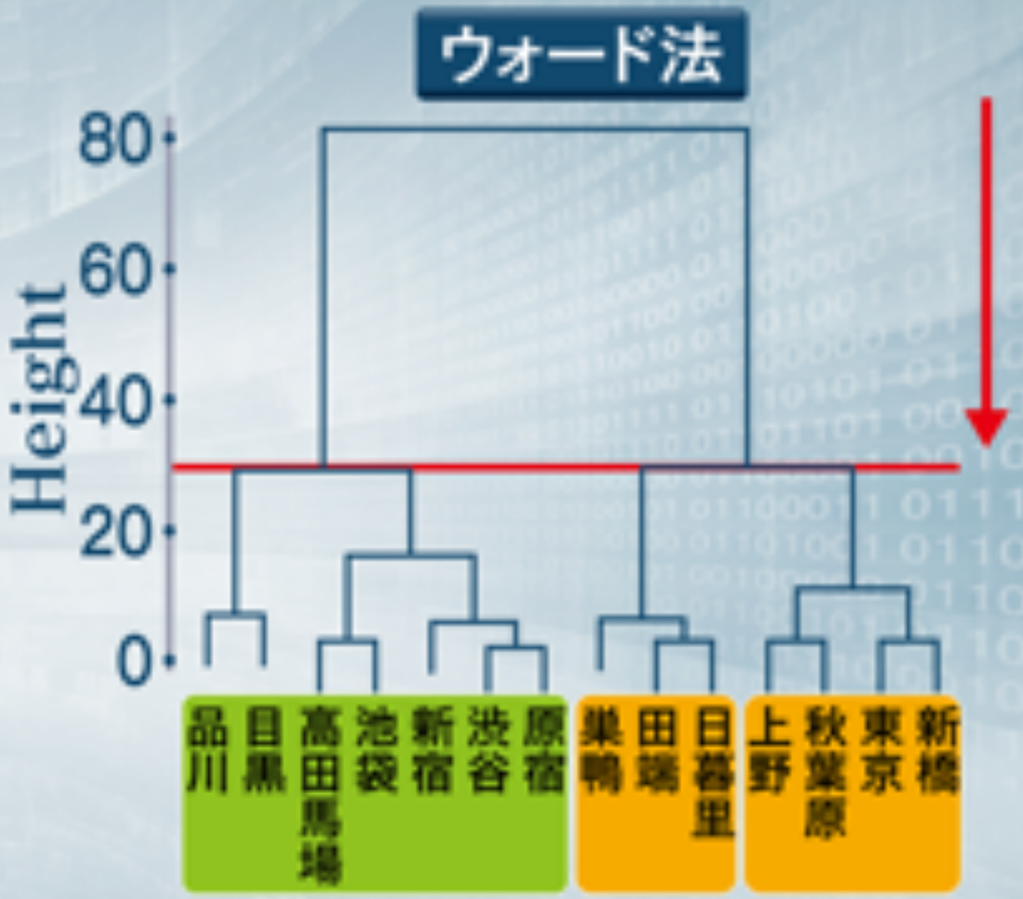
階層的クラスター分析


池袋
高田馬場
新宿
原宿
渋谷
目黒
品川
巣鴨
田端
日暮里
上野
秋葉原
東京
新橋
10
5
0
−5
−10
−15
−10
−5
0
5
10

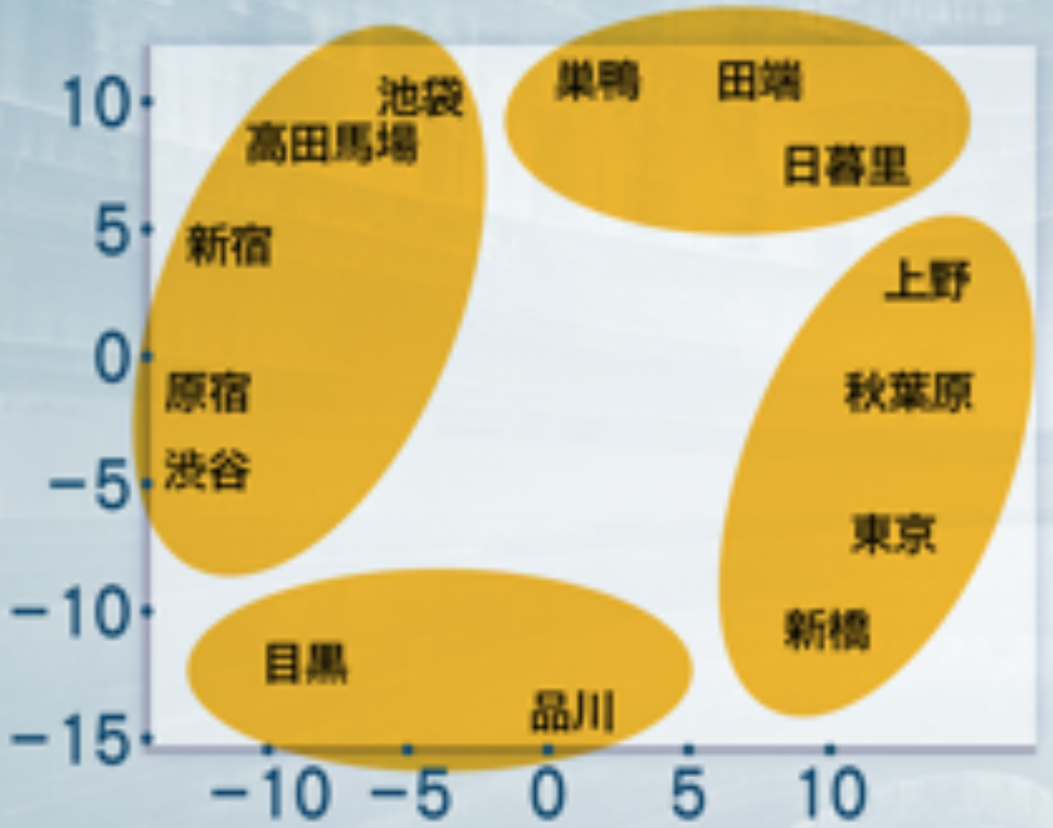

ウォード法
Height
80
60
40
20
0
品川
目黒
高田馬場
池袋
新宿
渋谷
原宿
巣鴨
田端
日暮里
上野
秋葉原
東京
新橋

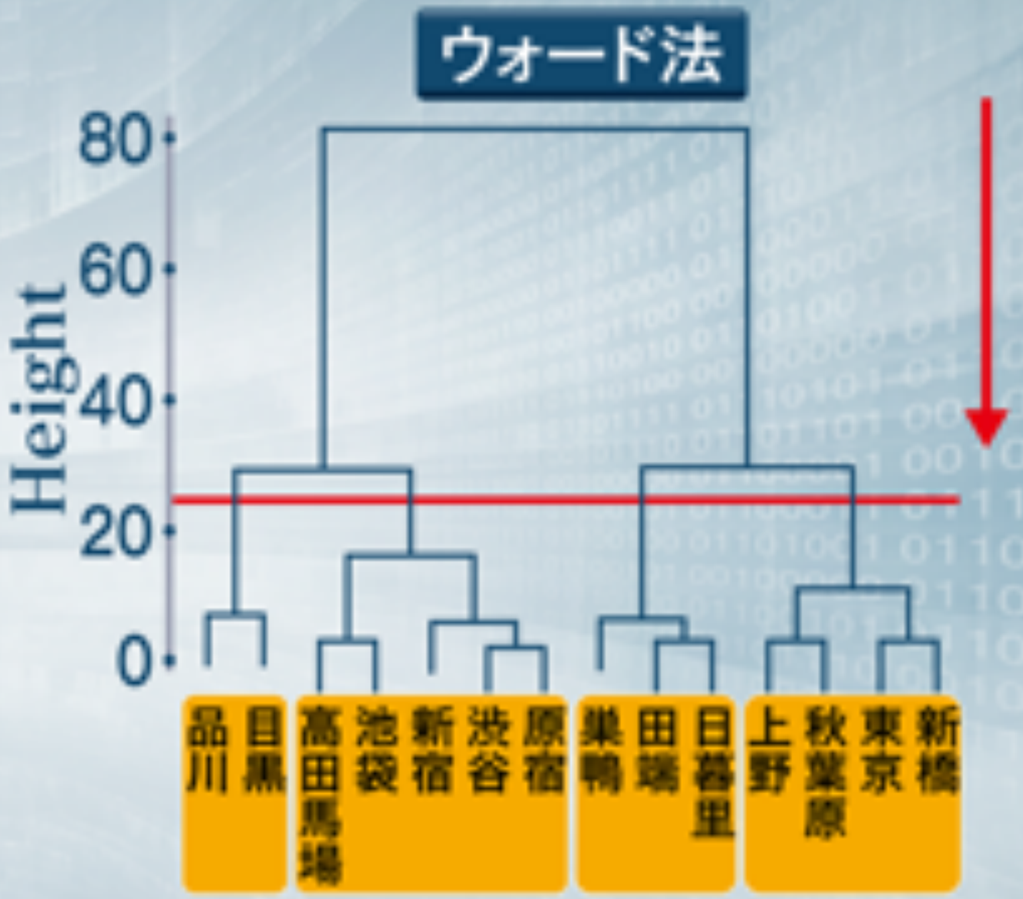
階層的クラスター分析


池袋
高田馬場
新宿
原宿
渋谷
目黒
品川
巣鴨
田端
日暮里
上野
秋葉原
東京
新橋
10
5
0
−5
−10
−15
−10
−5
0
5
10


ウォード法
Height
80
60
40
20
0
品川
目黒
高田馬場
池袋
新宿
渋谷
原宿
巣鴨
田端
日暮里
上野
秋葉原
東京
新橋

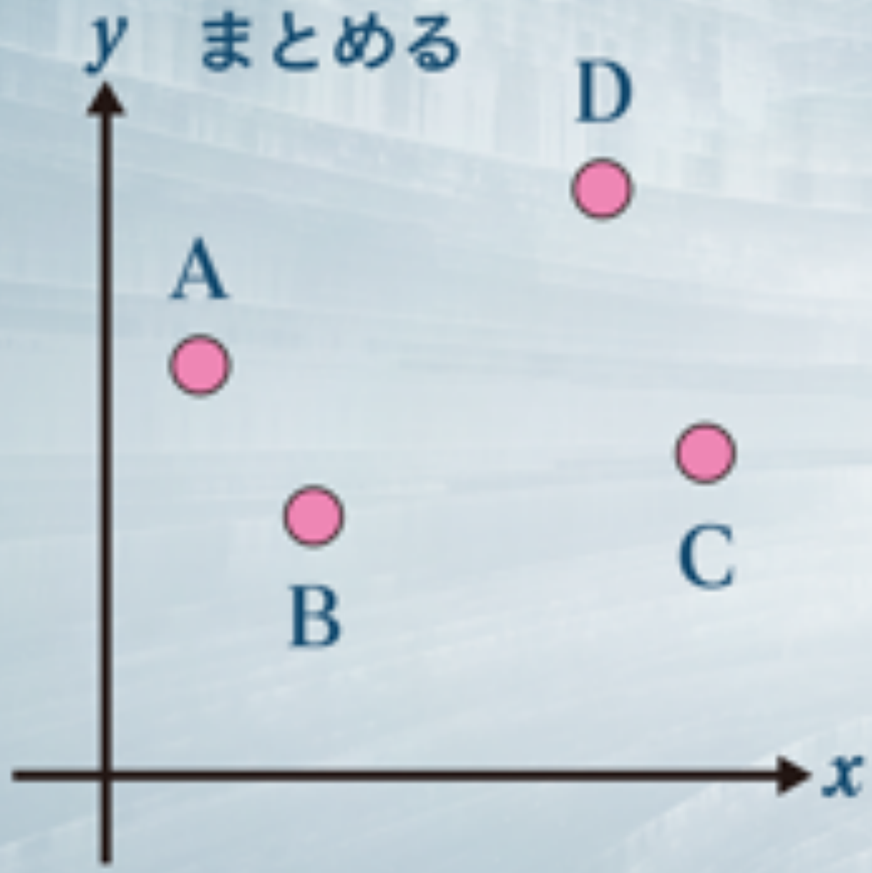
階層的クラスター分析


池袋
高田馬場
新宿
原宿
渋谷
巣鴨
田端
日暮里
上野
秋葉原
東京
新橋
目黒
品川
10
5
0
−5
−10
−15
−10
−5
0
5
10


ウォード法
Height
80
60
40
20
0
品川
目黒
高田馬場
池袋
新宿
渋谷
原宿
巣鴨
田端
日暮里
上野
秋葉原
東京
新橋

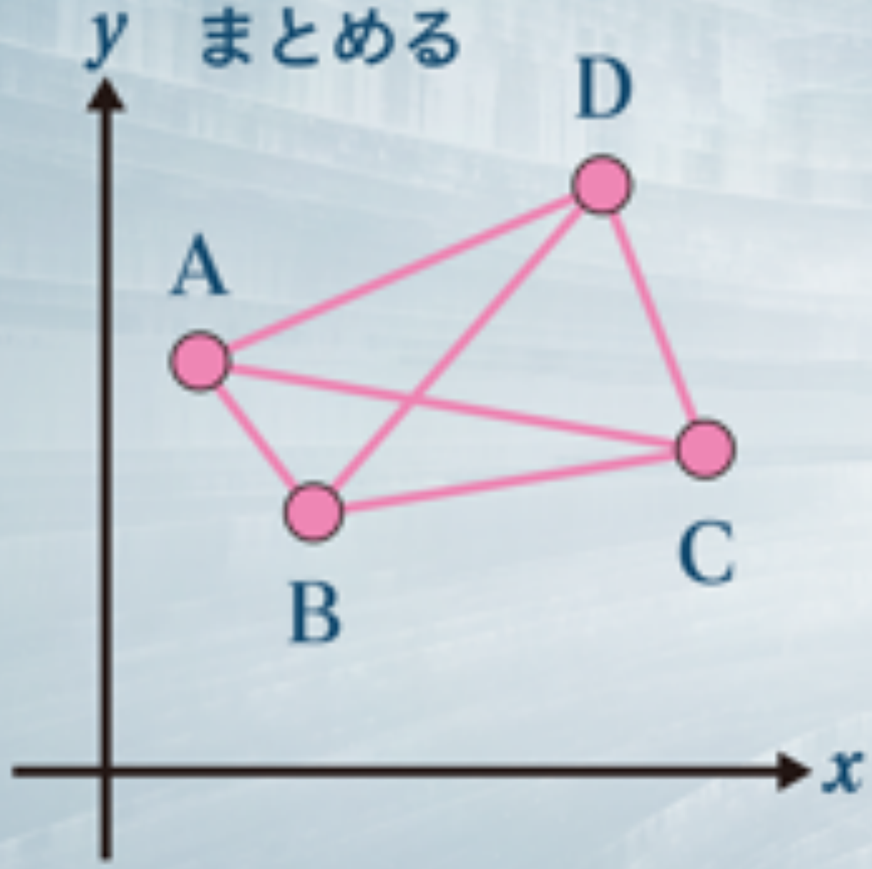
階層的クラスター分析

距離の近い2つの点を
まとめる



階層的クラスター分析

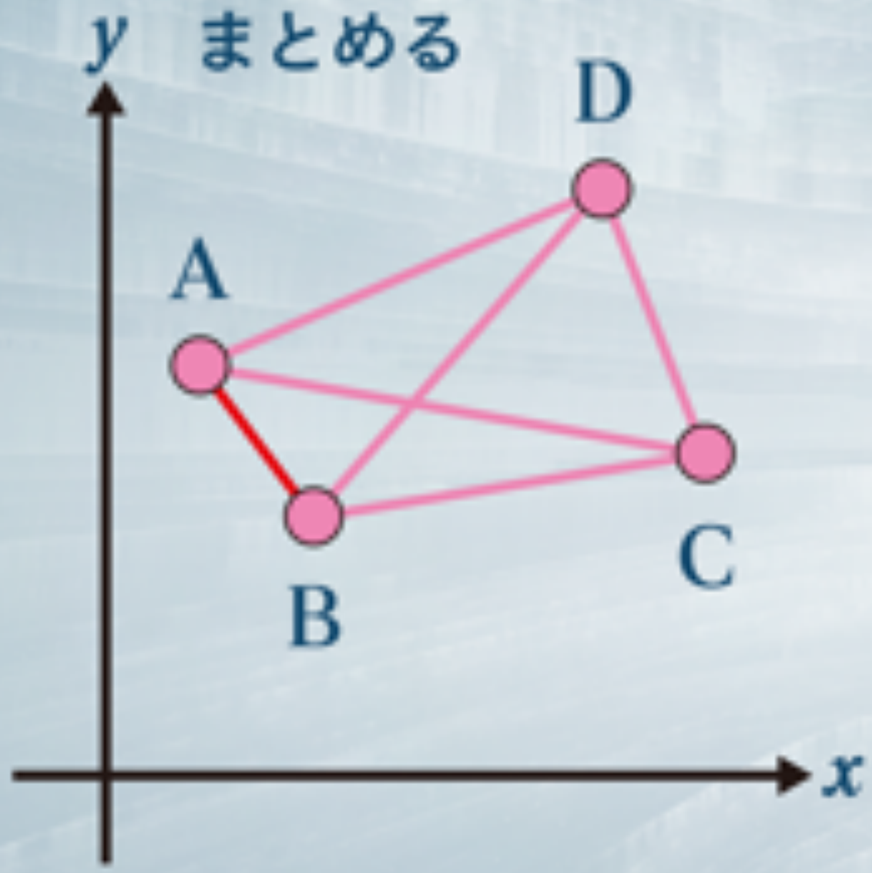
距離の近い2つの点を
まとめる



- それぞれ2点間の距離が
  与えられているものとする

階層的クラスター分析

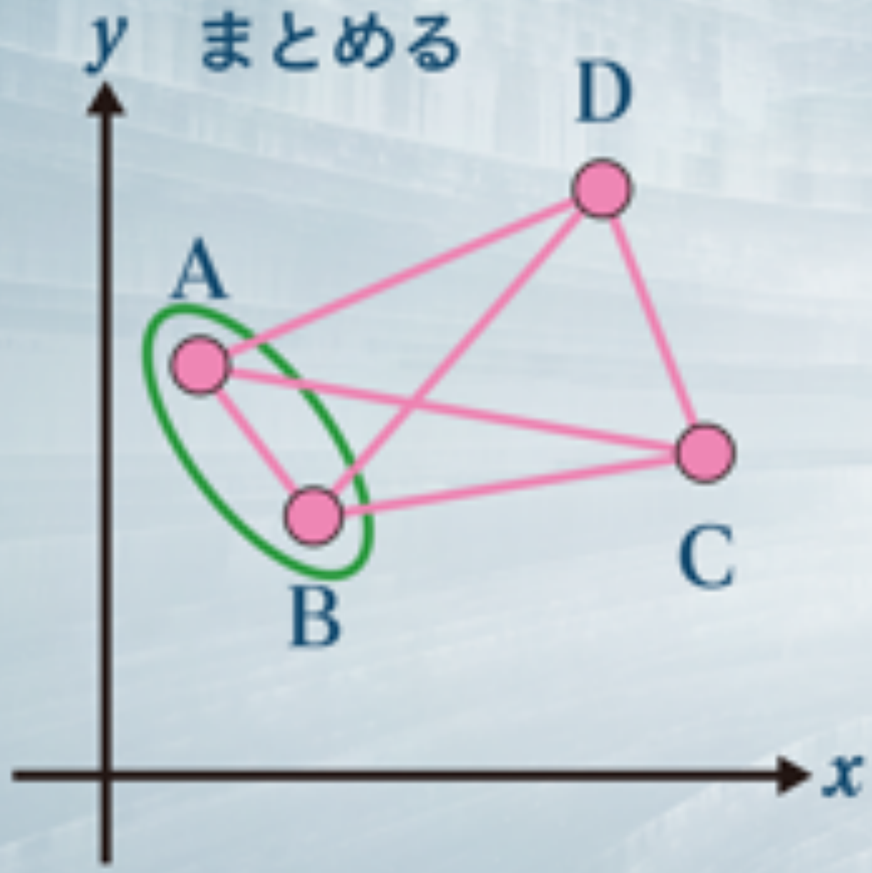
距離の近い2つの点を
まとめる



- 距離が最短のものを選ぶ

階層的クラスター分析

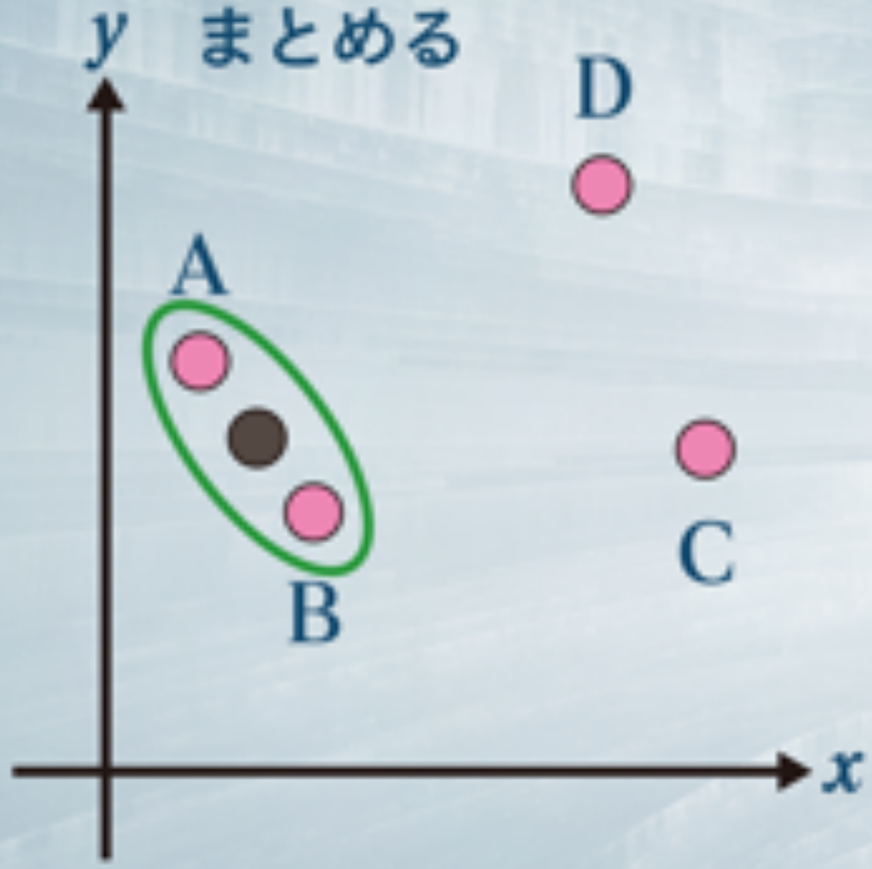
距離の近い2つの点を
まとめる



- 一番距離の短いものを1つのグループにする

階層的クラスター分析

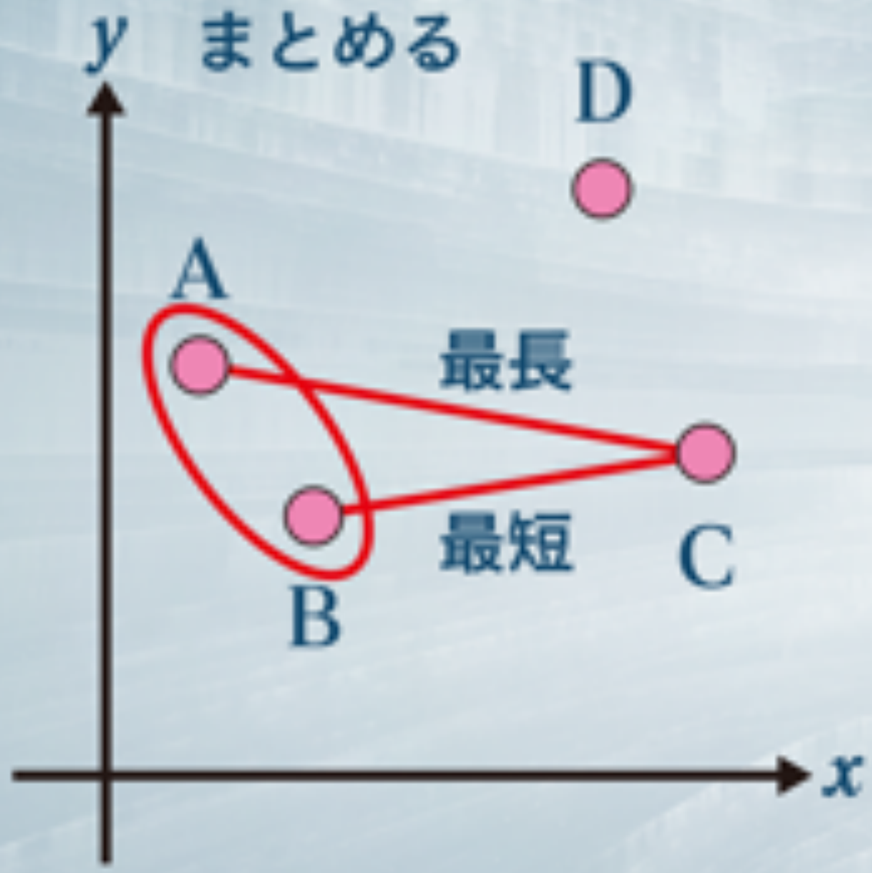
距離の近い2つの点を
まとめる



- まとめた点と他の点との距離を
  計算し直す（いくつか方法がある）

階層的クラスター分析

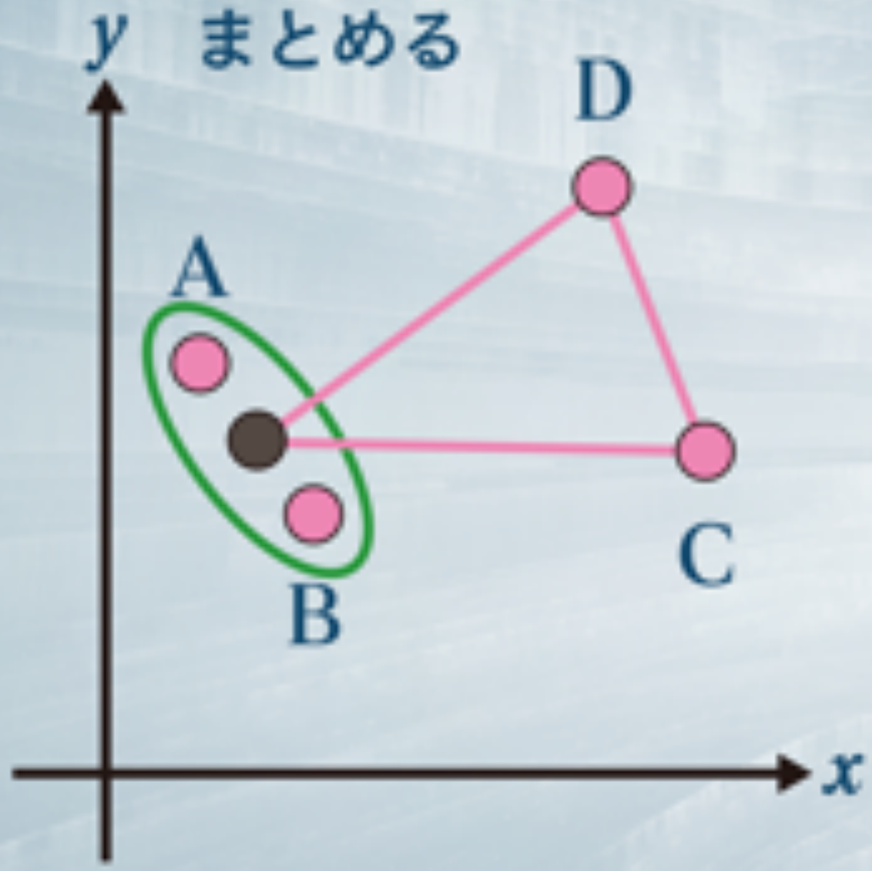
距離の近い2つの点を
まとめる



- まとめた点と他の点との距離を
  計算し直す（いくつか方法がある）

階層的クラスター分析

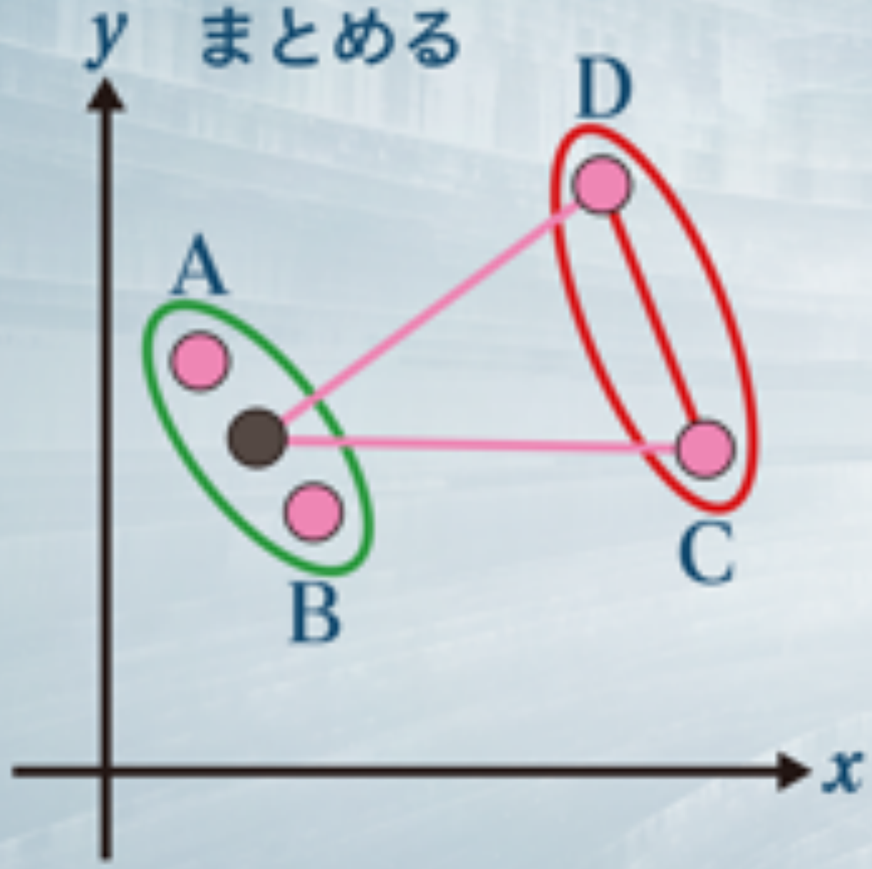
距離の近い2つの点を
まとめる



- 距離を再計算することで
  点が1つ少なくなったと
  考えることができる

階層的クラスター分析

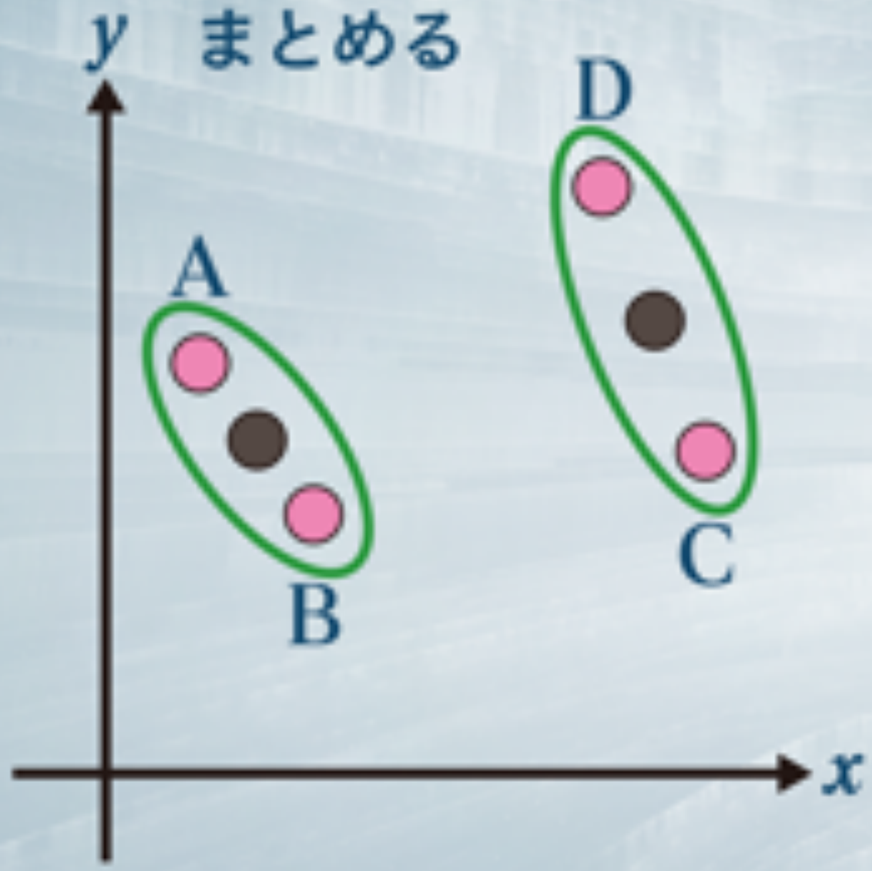
距離の近い2つの点を
まとめる



- 先ほどと同様に、距離の一番短い点を1つの点としてまとめる

階層的クラスター分析

距離の近い2つの点を
まとめる

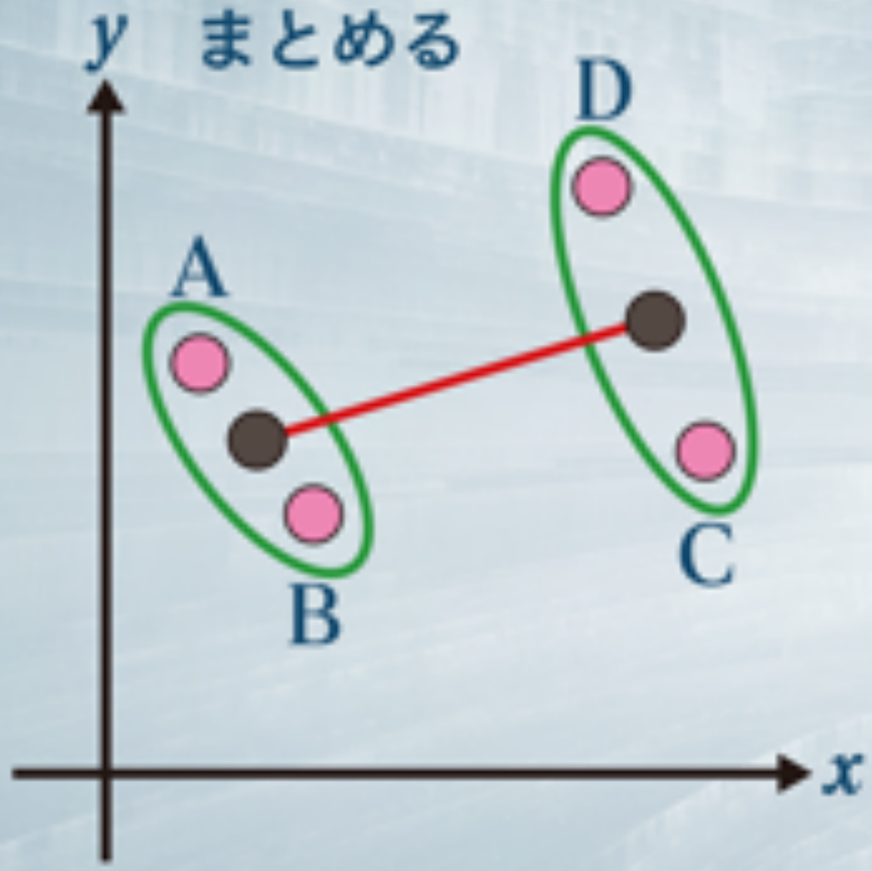


- 距離の短いものを選ぶ
- それを1つの点として考える
- 距離を計算し直す

という作業を繰り返す

階層的クラスター分析

距離の近い2つの点を
まとめる



- 距離の短いものを選ぶ
- それを1つの点として考える
- 距離を計算し直す

という作業を繰り返す

最後は一点にまとまって終了となる

距離の再計算

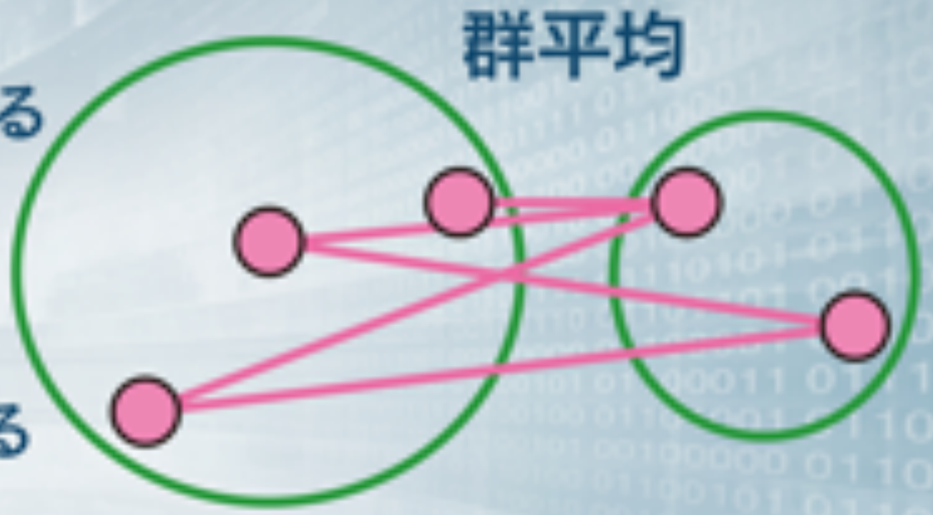
1 最短距離法　一番短い距離を利用する

2 最長距離法　一番長い距離を利用する

3 群平均法　距離の平均を求める

4 重心法　重心との距離を計算する

5 ウォード法　クラスターが巨大にならないように
グループの中の距離がどれだけ増えるかを計算する
それが一番最小となる点を選ぶ



階層的クラスター分析の計算(hclust)

所要時間データ

	1	2	3	・・・	12	13	14
1. 品川	0	7	12	・・・	14	10	7
2. 目黒	7	0	5	・・・	21	17	14
3. 渋谷	12	5	0	・・・	26	22	19
4. 原宿	14	7	2	・・・	28	24	21
・・・				・・・			
12. 秋葉原	14	21	26	・・・	0	4	7
13. 東京	10	17	22	・・・	4	0	3
14. 新橋	7	14	19	・・・	7	3	0

階層的クラスター分析の計算(hclust)

所要時間データ

CSVファイル

,品川,目黒,渋谷,原宿,新宿,高田馬場･･･
品川, 0 , 7 ,12,14,18,22,･･･
目黒, 7 , 0 , 5 , 7 ,11,15,･･･
渋谷,12, 5 , 0 , 2 , 6 ,10,･･･
原宿,14, 7 , 2 , 0 , 4 , 8 ,･･･
新宿,18,11, 6 , 4 , 0 , 4 ,･･･
高田馬場,22,15,10, 8 , 4 , 0 ,･･･
池袋,25,18,13,11, 7 , 3 ,･･･
⋮

階層的クラスター分析の計算(hclust)

```
> yamate0 <- read.csv("ch111.csv", header=T, row.names=1)
> yamate1 <- as.dist(yamate0)
```

```
> yamatewa <- hclust(yamate1, method="ward.D2")
> yamatewa
```

```
> plot(yamatewa)
```

階層的クラスター分析の計算(hclust)

```
> yamate0 <- read.csv("ch111.csv", header=T, row.names=1)
> yamate1 <- as.dist(yamate0)
```

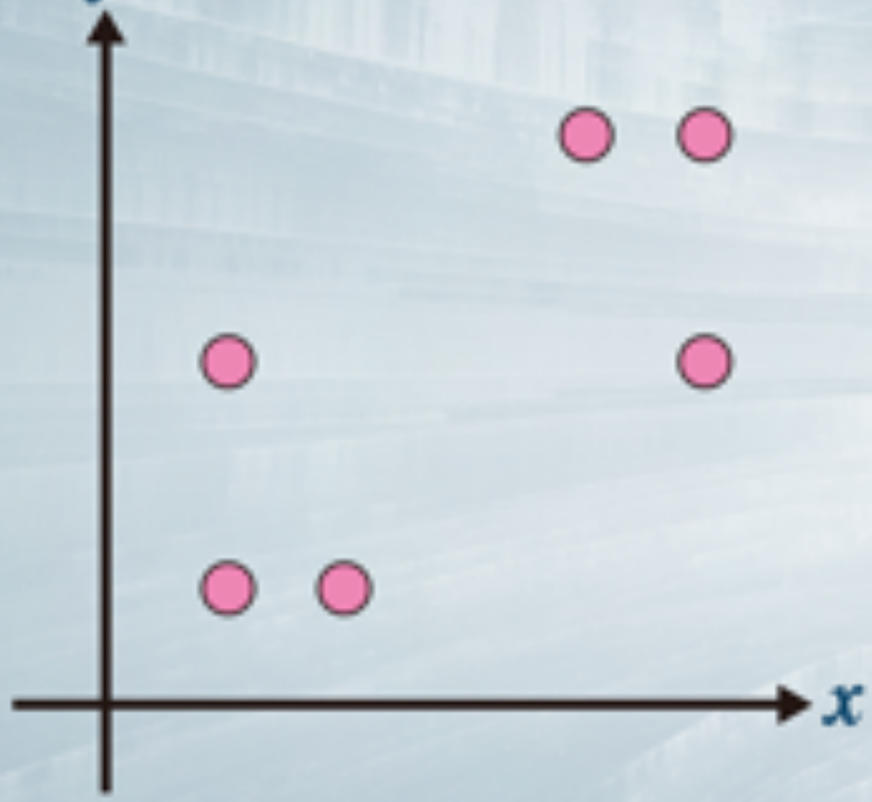
```
> yamatewa <- hclust(yamate1, method="ward.D2")
> yamatewa
```

```
> plot(yamatewa)
```

hclust (距離データ, method = 距離の方法)
最短距離法 single、最長距離法 complete、
群平均法 average、重心法 centroid、
ウォード法 ward.D2

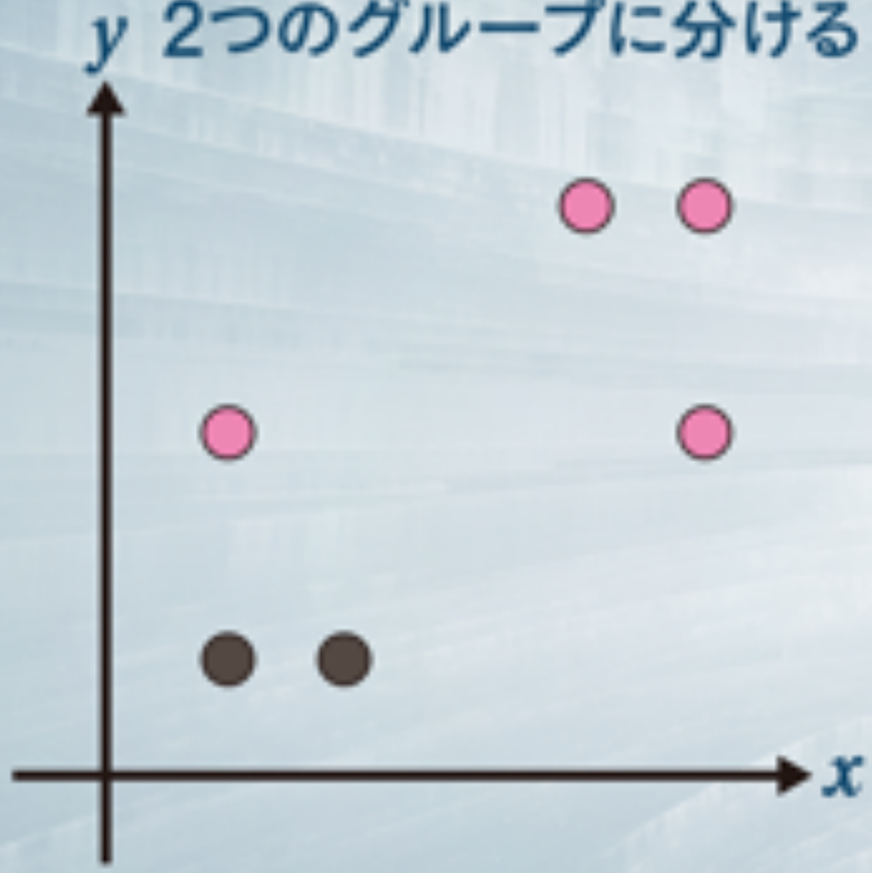
k平均法

6個の点を
2つのグループに分ける



k平均法

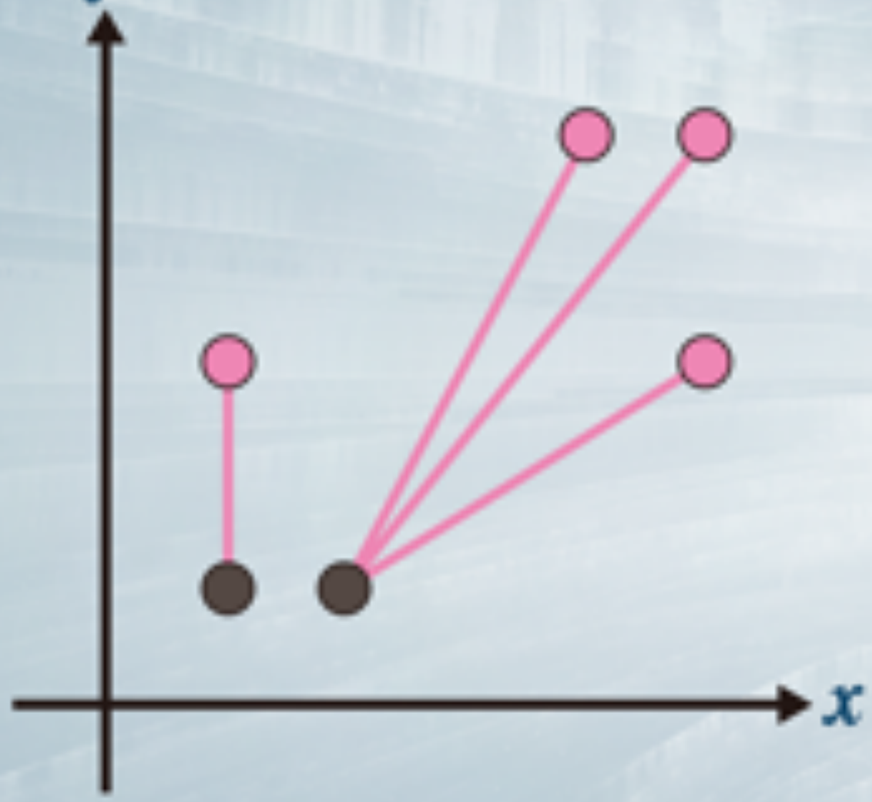
6個の点を
2つのグループに分ける



- 2個の点を選び、その点を
  2つのグループの代表者とする

k平均法

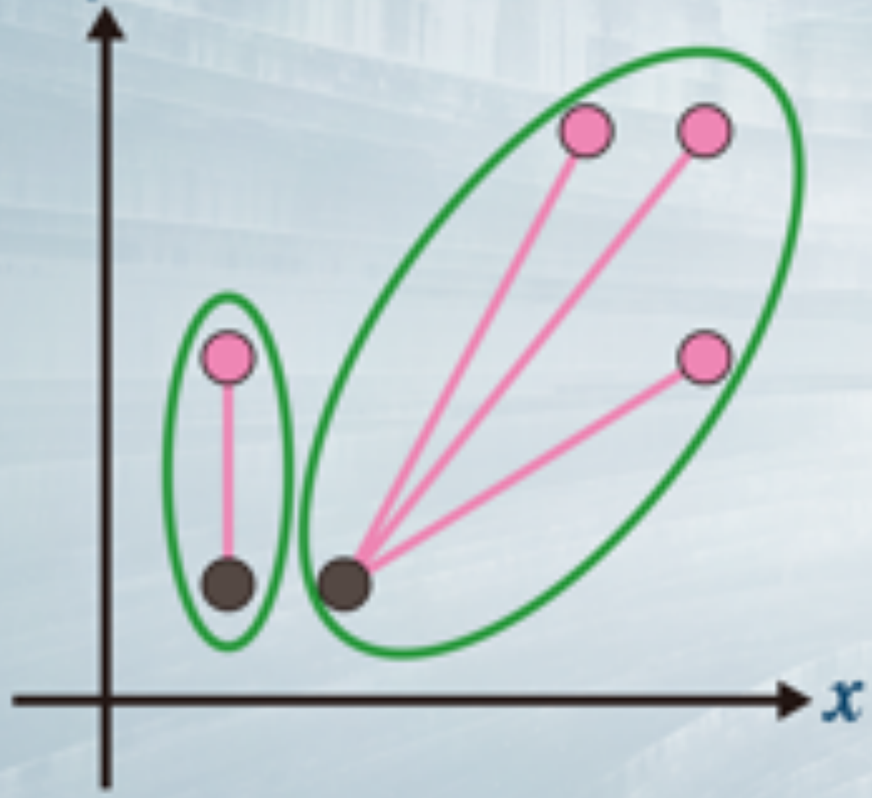
6個の点を
2つのグループに分ける



- その2点との距離を計算する

k平均法

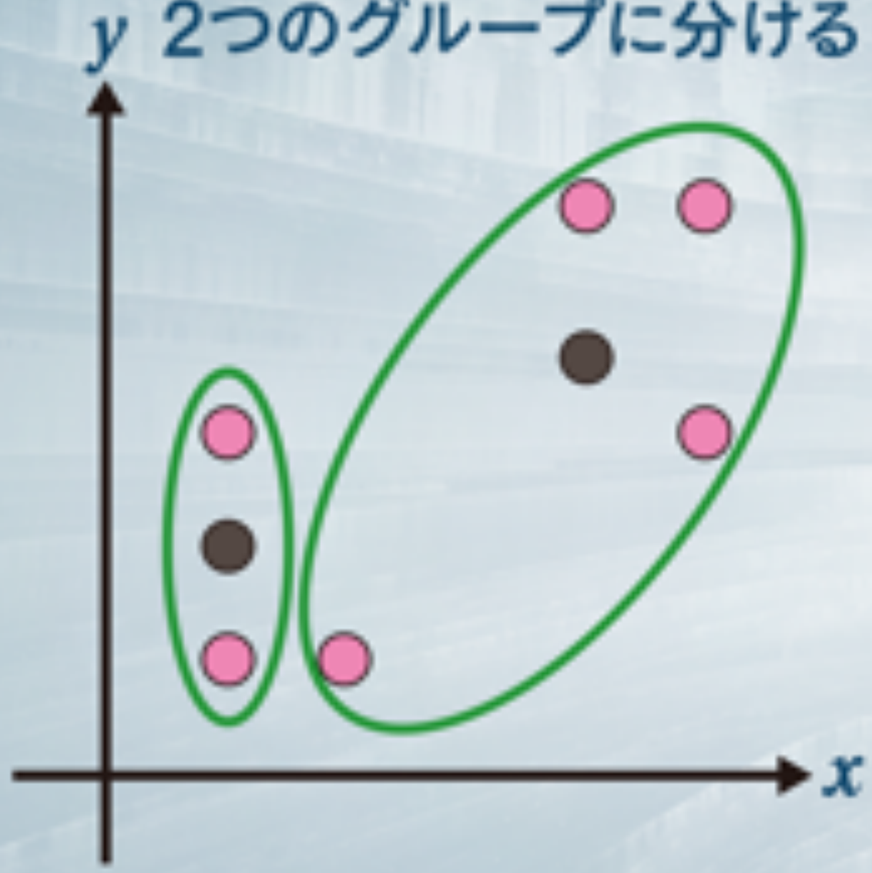
6個の点を
2つのグループに分ける



- それぞれの点は、近い方の代表点のグループに属することにする

k平均法

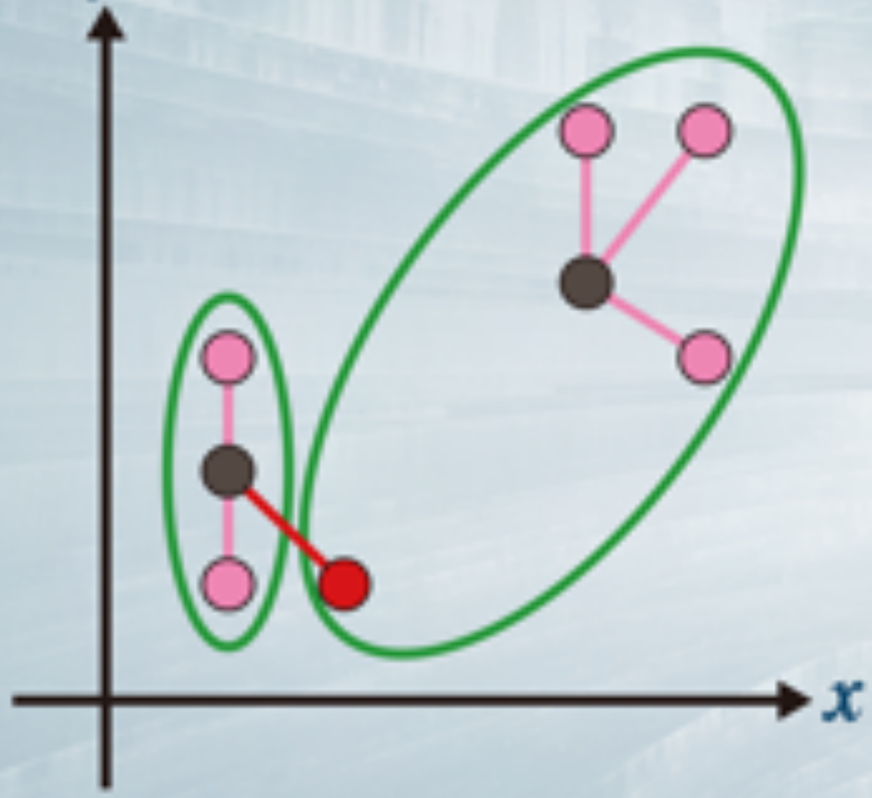
6個の点を
2つのグループに分ける



- それぞれのグループの重心を
  新たな代表の点とする

k平均法

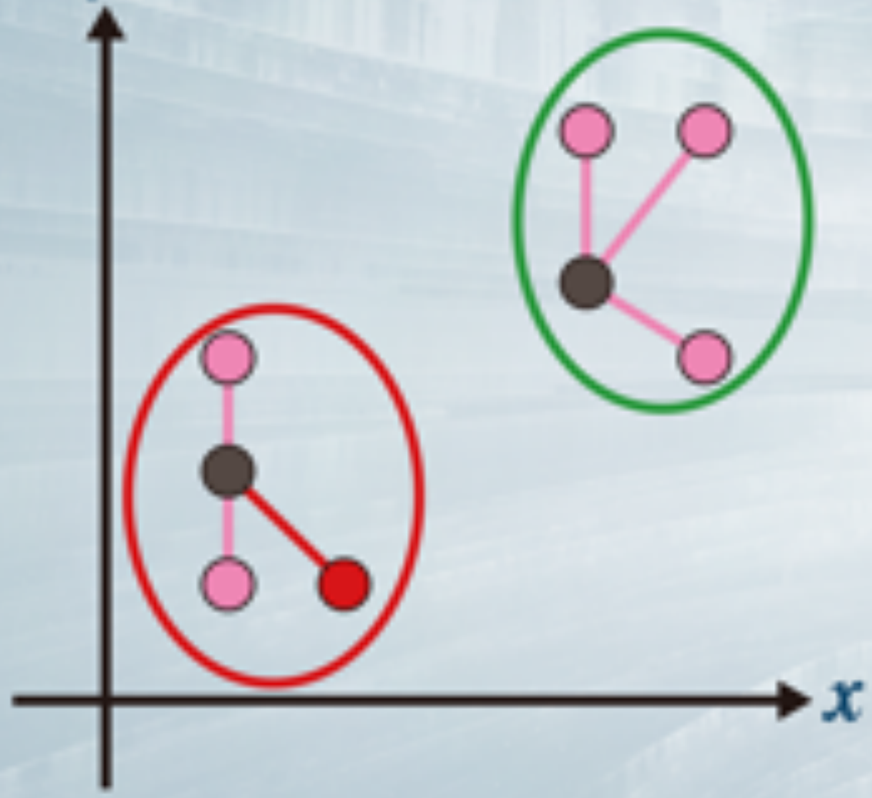
6個の点を
2つのグループに分ける



- 新たな代表点と
  それぞれの距離を計算する

k平均法

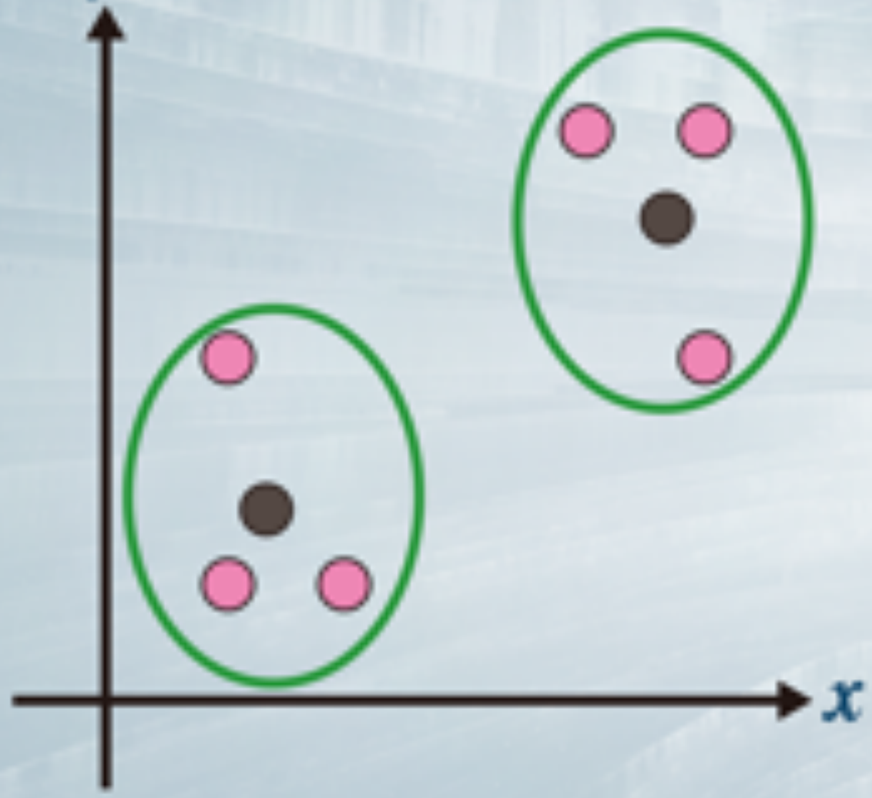
6個の点を
2つのグループに分ける



- 先ほどと同様に、近い方の代表点と同じグループとして、分類する

k平均法

6個の点を
2つのグループに分ける

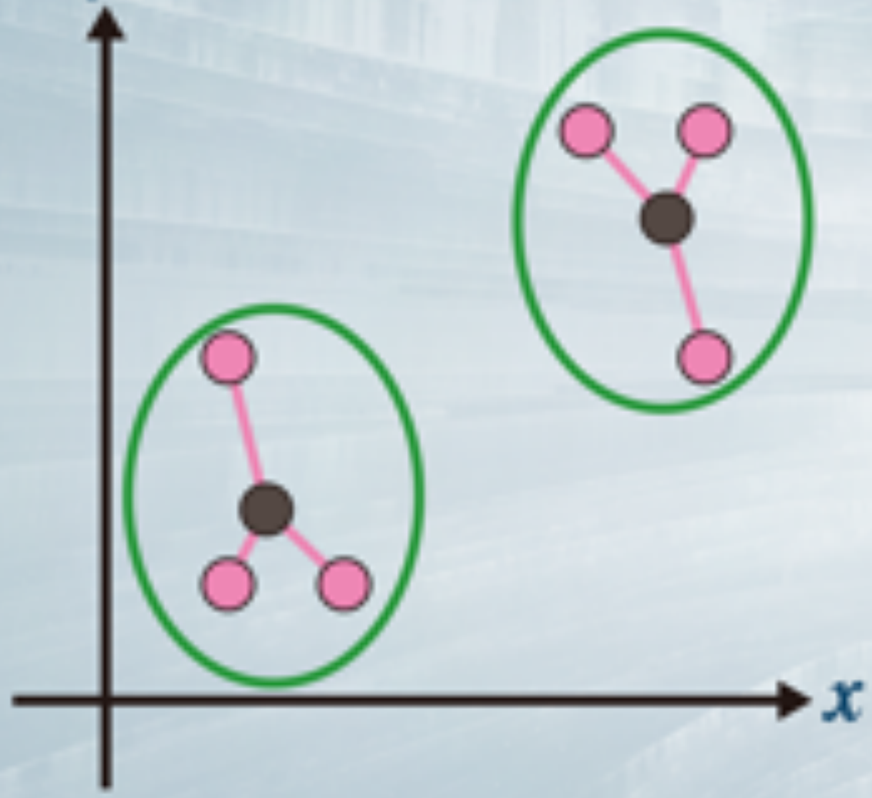


- 代表となる点を選び直す
- 距離を計算する
- 近い方のグループの点とする

という作業を繰り返す

k平均法

6個の点を
2つのグループに分ける

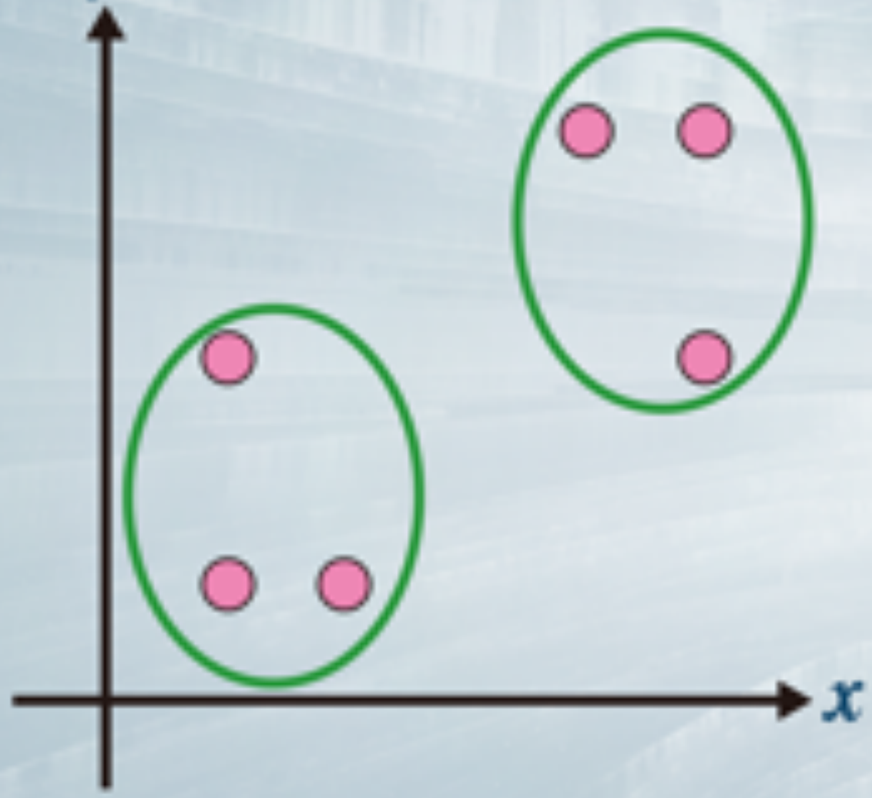


- 代表となる点を選び直す
- 距離を計算する
- 近い方のグループの点とする

という作業を繰り返す

k平均法

6個の点を
2つのグループに分ける



- 代表となる点を選び直す
- 距離を計算する
- 近い方のグループの点とする

という作業を繰り返す
グループに変化がなければ終了とする

k平均法の計算(kmeans)

6点の座標データ

name	x	y
C1	1	1
C2	2	1
C3	1	3
C4	4	5
C5	5	5
C6	5	3

Rによる計算（kmeans）

```
> km1 <- read.table("ch112.dat", header=T, row.names=1)
> km1
```

```
> km2<- kmeans (km1, 2)
> km2
```

```
> plot (km1, pch=km2$cluster)
> points (km2$centers, pch=8)
```