Segmentation

消费者细分

小王最近从重点班转到了普通班，结果两个班平均成绩都提高了，这是为什么？

Hospital Choice

You are choosing between two hospitals:

- Hospital A: Among each 1,000 patients, 900 survived.
- Hospital B: Among each 1,000 patients, 800 survived.

Which hospital are you going to choose?

医院的选择

你在选择两家医院：

- 医院A：每1000名病人中，有900人幸存。
- 医院B：每1000名病人中，有800人幸存。

你会选择哪家医院？

Fund Managers

Let us consider two fund managers.

- Fund manager A: Annualized rate of return 12%
- Fund manager B: Annualized rate of return 10%

Under what conditions should you choose manager A?

基金经理

让我们考虑两个基金公司。

- 基金公司A：年化收益率12%
- 基金公司B：年化收益率10%

在什么情况下你应该选择公司B?

Lesson Learned 教训总结

If you group your consumers, you may find something different!

如果你对消费者进行分组，你可能会发现一些不同的东西！

What is market segmentation? 什么是市场细分?

Market segmentation is the process of dividing consumers into groups with common characteristics and respond similarly to your marketing actions.

市场细分是将消费者根据共同特征划分为不同组别的过程，这些组别对你的营销活动反应相似。

Many new products fail because their creators use an ineffective market segmentation mechanism, according to HBS professor **Clayton Christensen**. It's time for companies to look at products the way customers do: as a way to get a job done.

Market segmentation allows you to target the right people with the right messaging at the right time. Segmentation enables you to learn more about your audience so you can better tailor your messaging to their preferences and needs.

市场细分使你能够在正确的时间以正确的信息针对合适的人群。细分让你更深入地了解你的受众，从而更好地根据他们的偏好和需求调整你的信息。

Examples of market segmentation: Coca-Cola

市场细分的例子：可口可乐

Regular coke

Diet coke

Caffeine-free drinks

Orange juice

Examples of market segmentation: Xiaomi
小米的市场细分

Number

Redmi Note

Mix

Civi

BlackShark

Market segmentation for Airlines:

Urgent travellers: Consumers have an urgent need to travel that is usually unexpected.

Business travellers: Consumers who visit different places for the needs of business.

Budget conscious: Holidaymakers who are price-sensitive.

航空公司的市场细分：

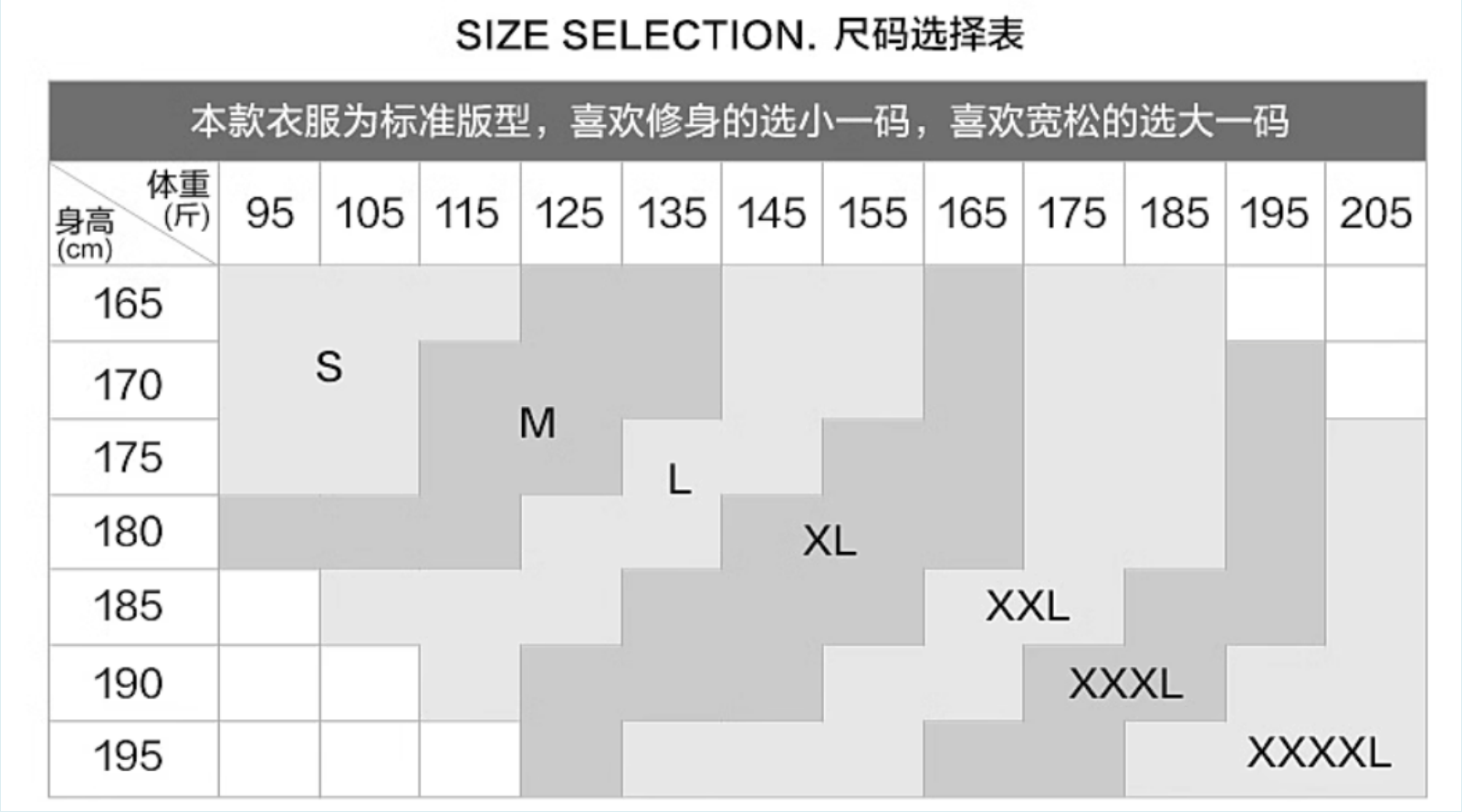
紧急旅行者：对旅行有紧迫需求的消费者，通常是意外产生的。
商务旅行者：因工作需要而前往不同地方的消费者。
注重预算的消费者：对价格敏感的度假者。

Clustering

聚类

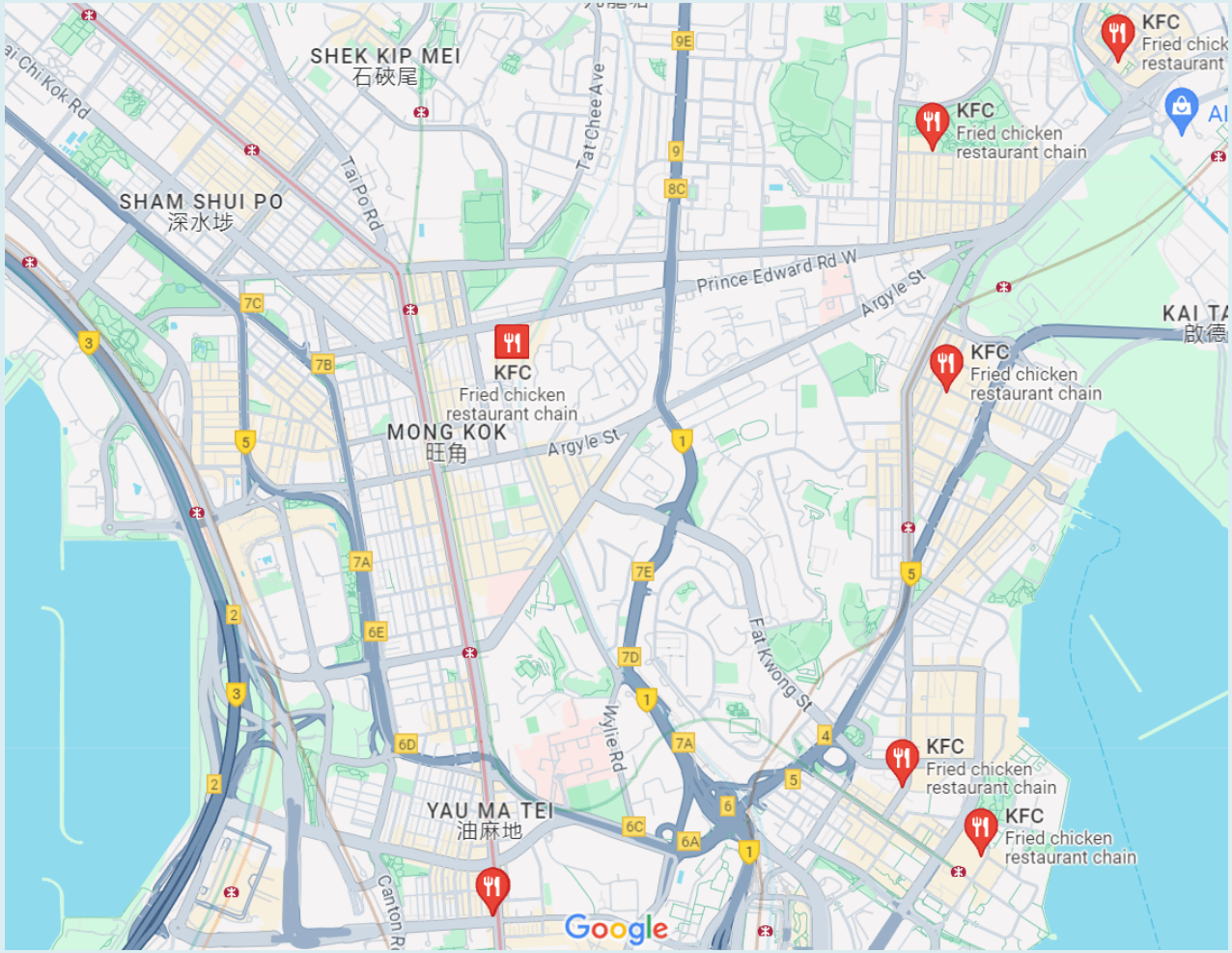
Imaging that you have designing T-shirt for consumers. You are looking at your consumers' weight and height, which would allow you to decide how many sizes to offer and which size fits a particular individual. Instead of classifying your consumers arbitrarily, you can use data to perform the task more precisely.

想象一下你在为消费者设计T恤。你会考虑消费者的体重和身高，这可以帮助你决定提供多少种尺码以及哪种尺码适合特定个体。通过数据分析，你可以更精确地完成这项任务，而不是随意分类消费者。



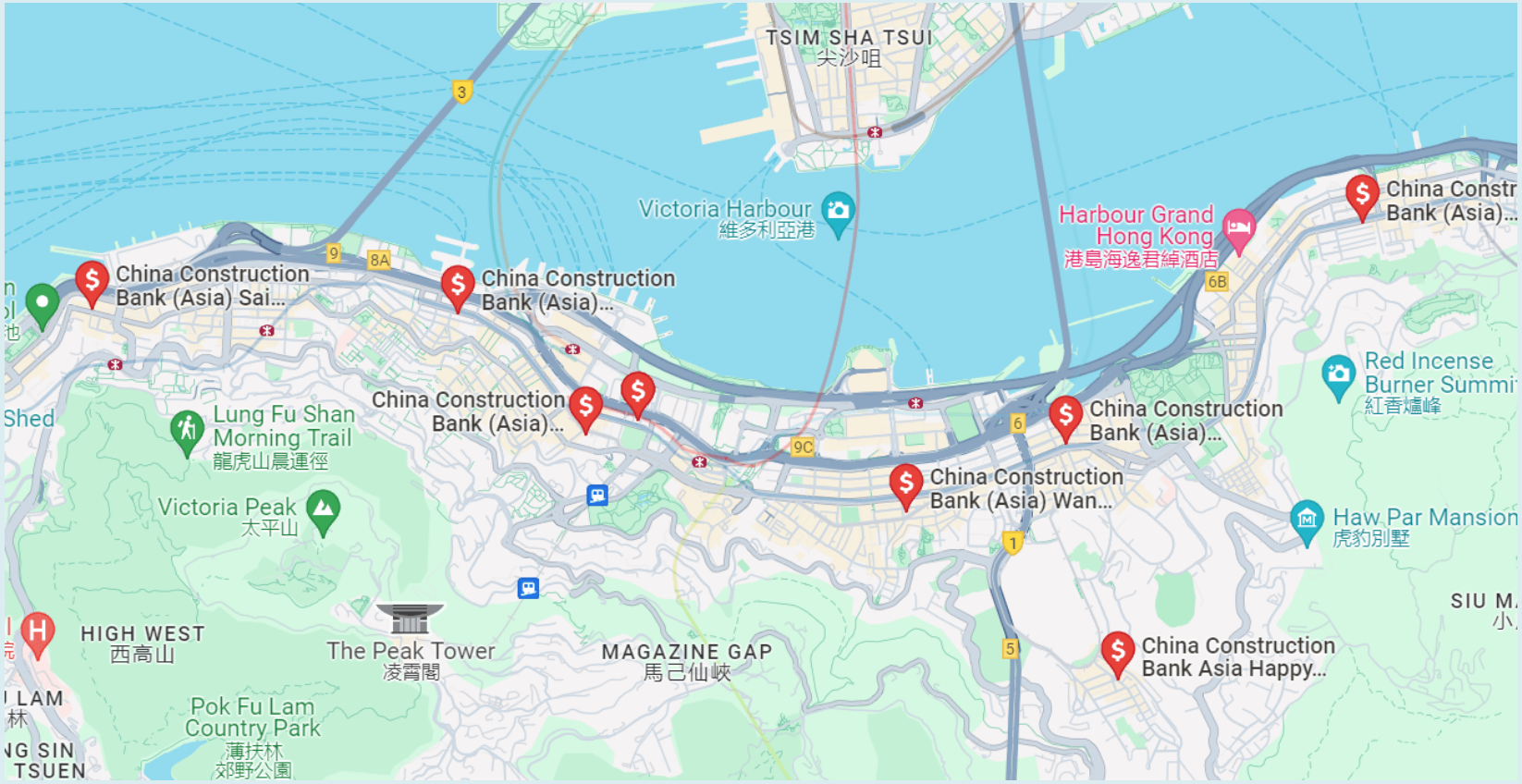
Why offer 7 sizes? Are these sizes optimal?

为什么提供7种尺码？这些尺码是否最优？



How are these locations decided?

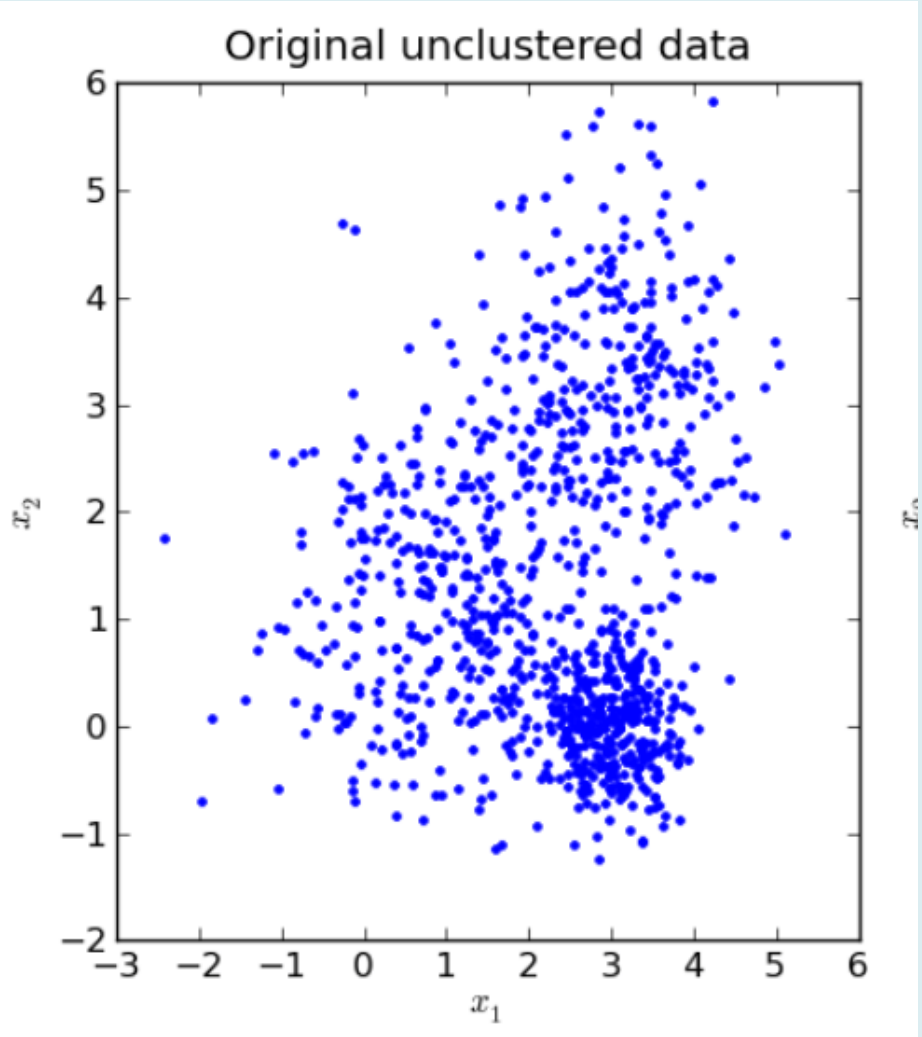
这些地点是如何决定的?

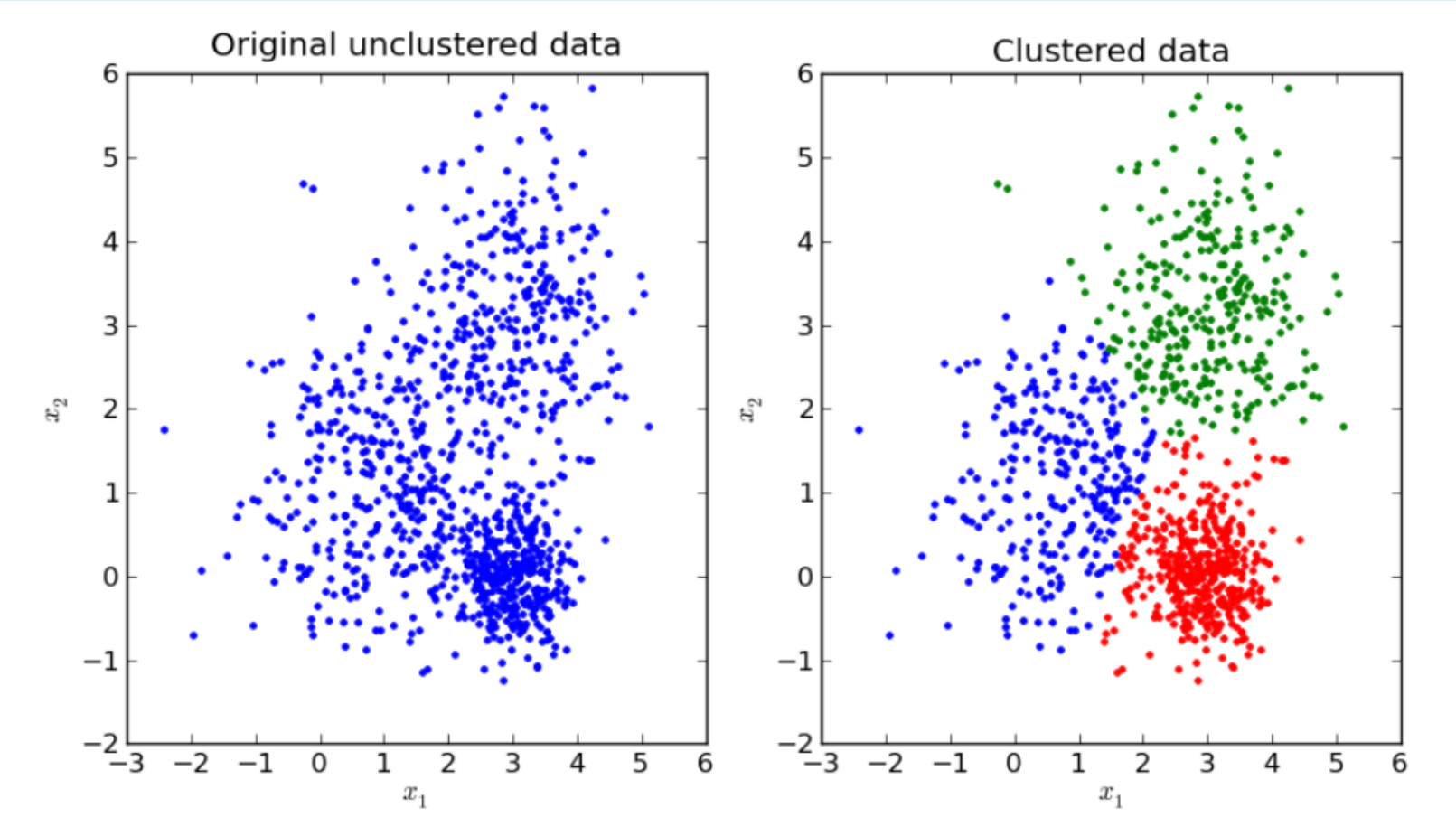

TSIM SHA TSUI
尖沙咀
Victoria Harbour
維多利亞港
Harbour Grand
Hong Kong
港島海逸君綽酒店
China Construction
Bank (Asia) Sai...
China Construction
Bank (Asia)...
China Construction
Bank (Asia)...
China Construction
Bank (Asia)...
China Construction
Bank (Asia) Wan...
China Construction
Bank Asia Happy...
Red Incense
Burner Summit
紅香爐峰
Haw Par Mansion
虎豹別墅
Lung Fu Shan
Morning Trail
龍虎山晨運徑
Victoria Peak
太平山
HIGH WEST
西高山
The Peak Tower
凌霄閣
MAGAZINE GAP
馬己仙峽
Pok Fu Lam
Country Park
薄扶林
郊野公園
3
9
8A
6
6B
9C
1
5

中國建設銀行(亞洲)

Here is consumer data. How would you classify them into groups?

这是消费者数据。你将如何将他们分类为不同的组?




Original unclustered data
Clustered data
x_1
x_2

The K-means Algorithm

The K-means algorithm an EM (expectation-maximization) algorithm commonly used for classifying objects.

Input: A number of observations $(X_1, X_2, \ldots, X_n)$ and k, the number of groups to be classified

Output: k mutually exclusive and collectively exhaustive groups containing all observations

K-均值算法

K- 均值算法是一种常用的EM（期望-最大化）算法，用于对对象进行分类

输入：一组观察值 $(X_1, X_2, \ldots, X_n)$ 和 k, 即要分类的组数。

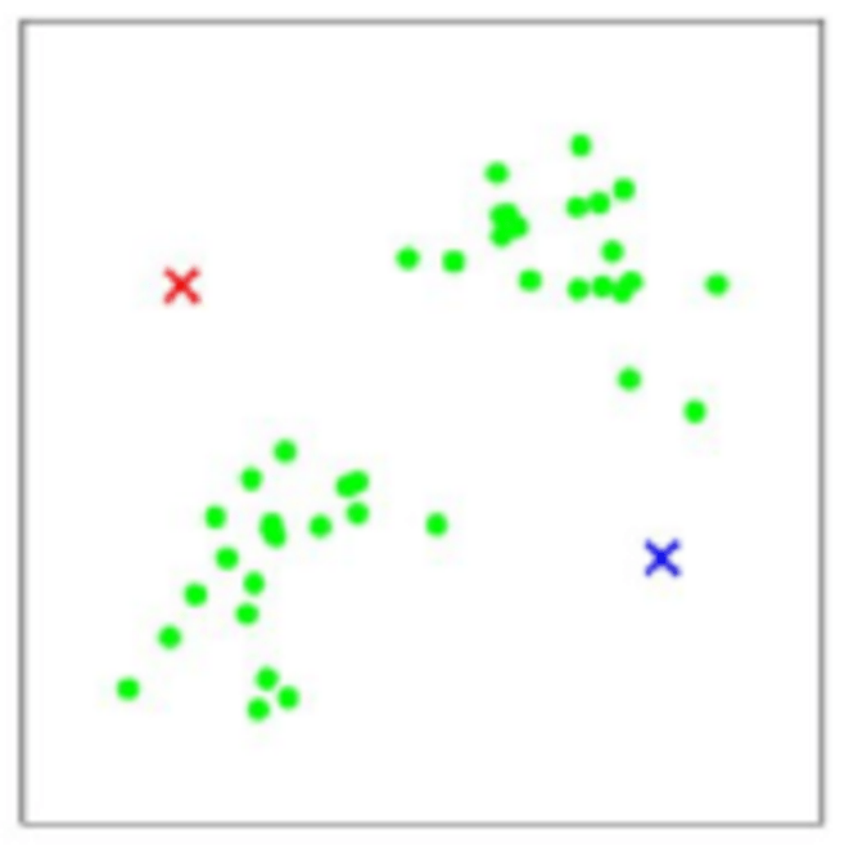
输出：k 个互斥且共同穷尽的组，包含所有观察值。

Classify the following observations into $k = 2$ groups:

将以下观察值分类为 $k = 2$ 组:

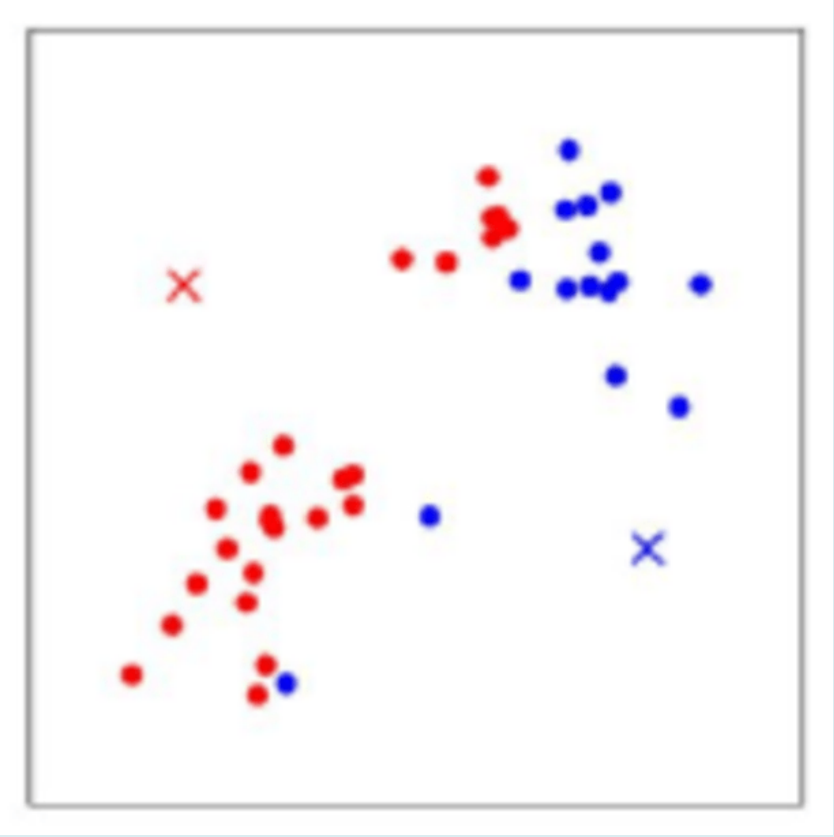
Step 1: Randomly choose $k = 2$ "centers" for your clusters.

步骤1：随机选择 $k = 2$ 个聚类的“中心”。



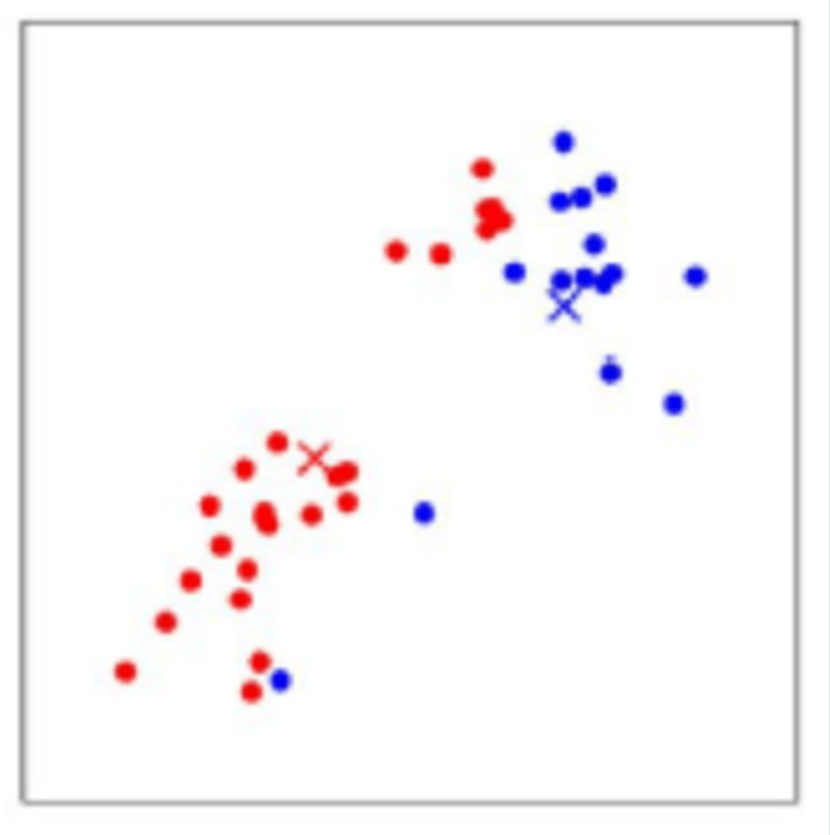
Step 2: Assign each observation to the nearest center.

步骤2：将每个观察值分配给最近的中心。



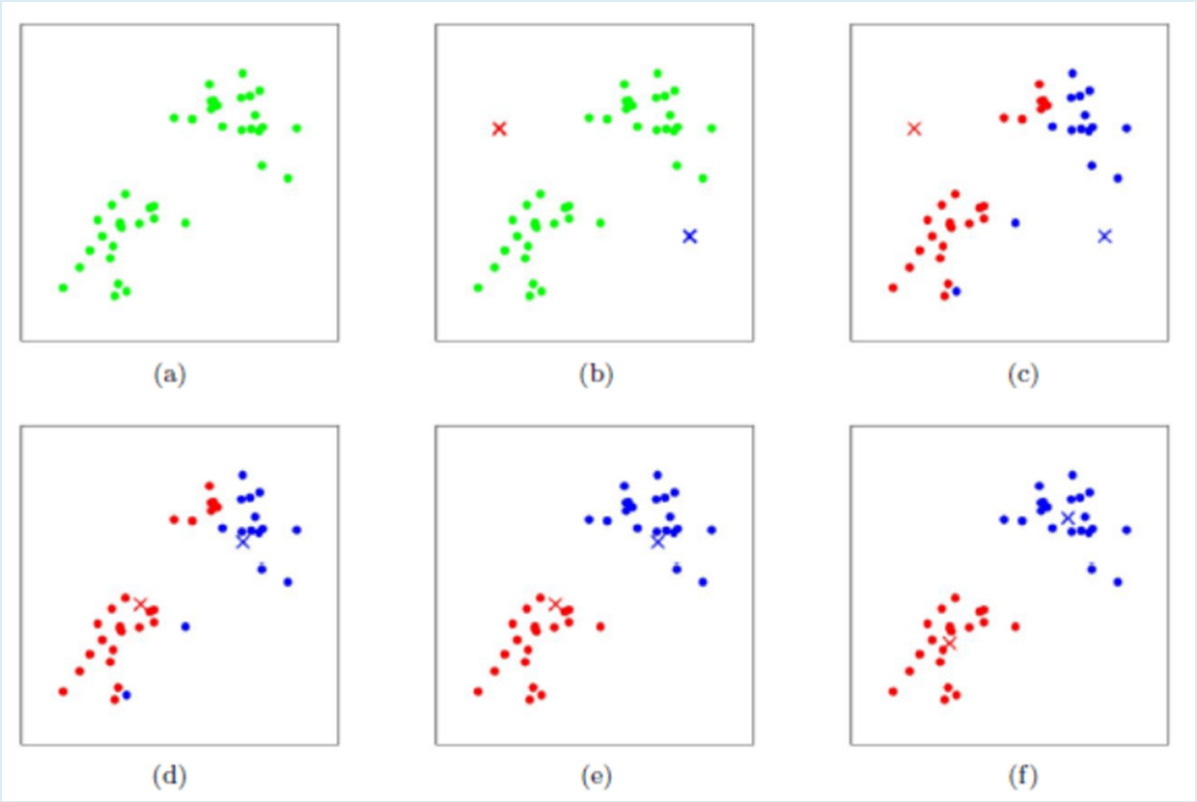
Step 3: Update the location the centers, which is given by the average location of all points in the corresponding cluster.

步骤3：更新中心的位置，其由相应聚类中所有点的平均位置给出。



Repeat the above process again and again until the centers no longer change.

重复上述过程，直到中心不再改变为止。



The K-means algorithm:

1. Select cluster centers $c_1, \ldots, c_k \in \mathbb{R}^d$ arbitrarily.
2. Assign every $x \in \mathcal{X}$ to the cluster $\mathcal{C}_i$ whose cluster center c_i is closest to it, i.e., $\|x - c_i\| \leq \|x - c_j\|$ for all $j \neq i$.
3. Set $c_i = \frac{1}{|\mathcal{C}_i|} \sum_{x \in \mathcal{C}_i} x$.
4. If clusters or centers have changed, goto 2. Otherwise, terminate.

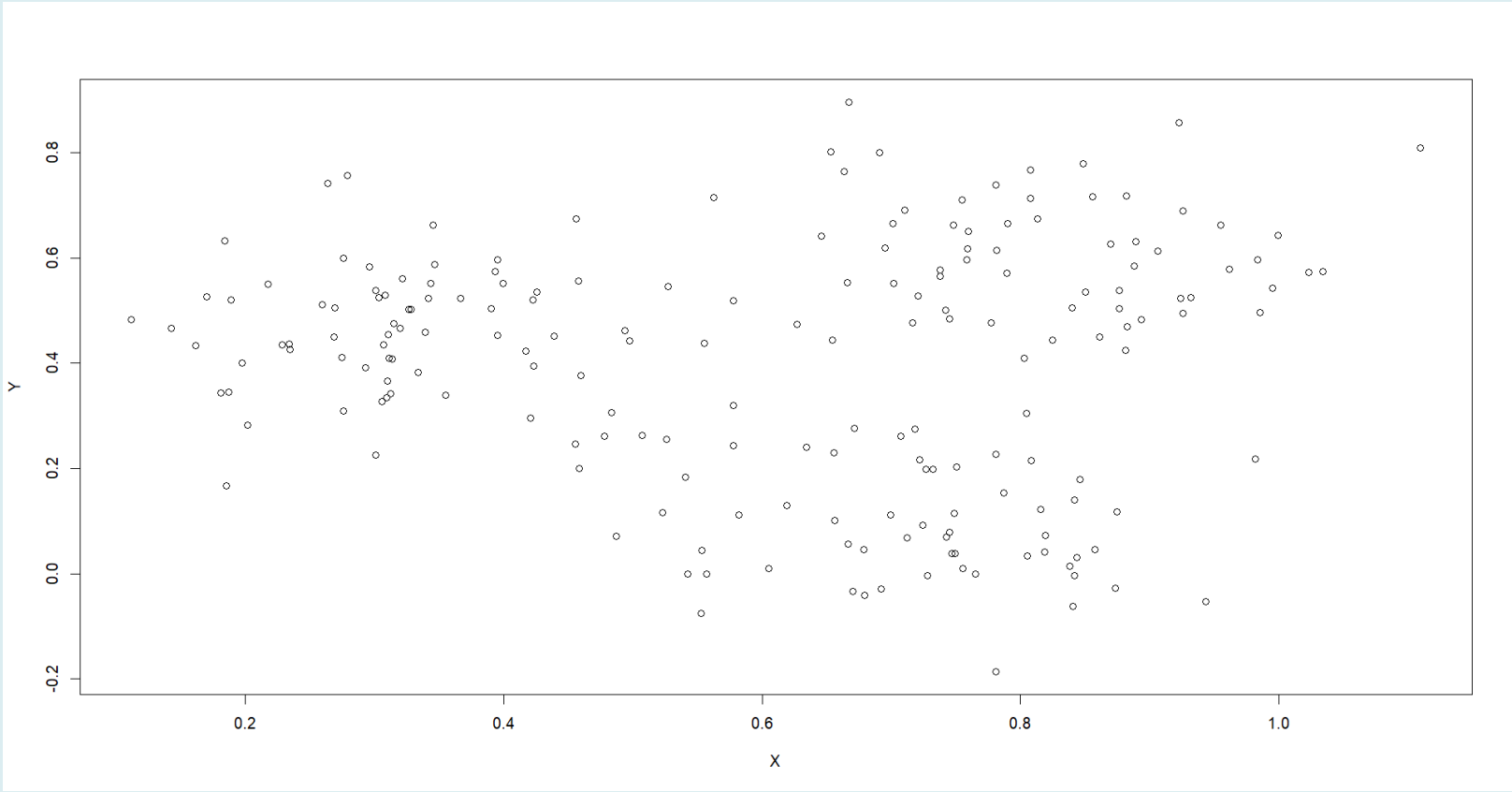
The K-means algorithm in R: 在 R 中实现 K-均值算法

```r
library(cluster)
library(factoextra)
mydata <- read_csv("https://ximarketing.github.io/data/clustering.csv")
head(mydata)
```

	X	Y
	<dbl>	<dbl>
1	0.627	0.474
2	0.563	0.714
3	0.296	0.583
4	0.816	0.122
5	0.345	0.662
6	0.234	0.436

Visualizing the data: 可视化数据

```
1 plot(mydata)
```



```
1 result <- kmeans(mydata, centers = 3, nstart = 25)
2 result
```

Here, centers = 3 means we want to classify the observations into three clusters. nstart = 25 means we randomly run the algorithm 25 times and pick the best results.

```
Cluster means:
          X         Y
1 0.8147681 0.6048477
2 0.3243930 0.4611514
3 0.7128718 0.1029810
```

These are the centers of the three clusters.

```
1 result <- kmeans(mydata, centers = 3, nstart = 25)
2 result
```

这里，centers = 3 意味着我们希望将观察值分类为三个聚类。nstart = 25 表示我们随机运行算法25次，并选择最佳结果。

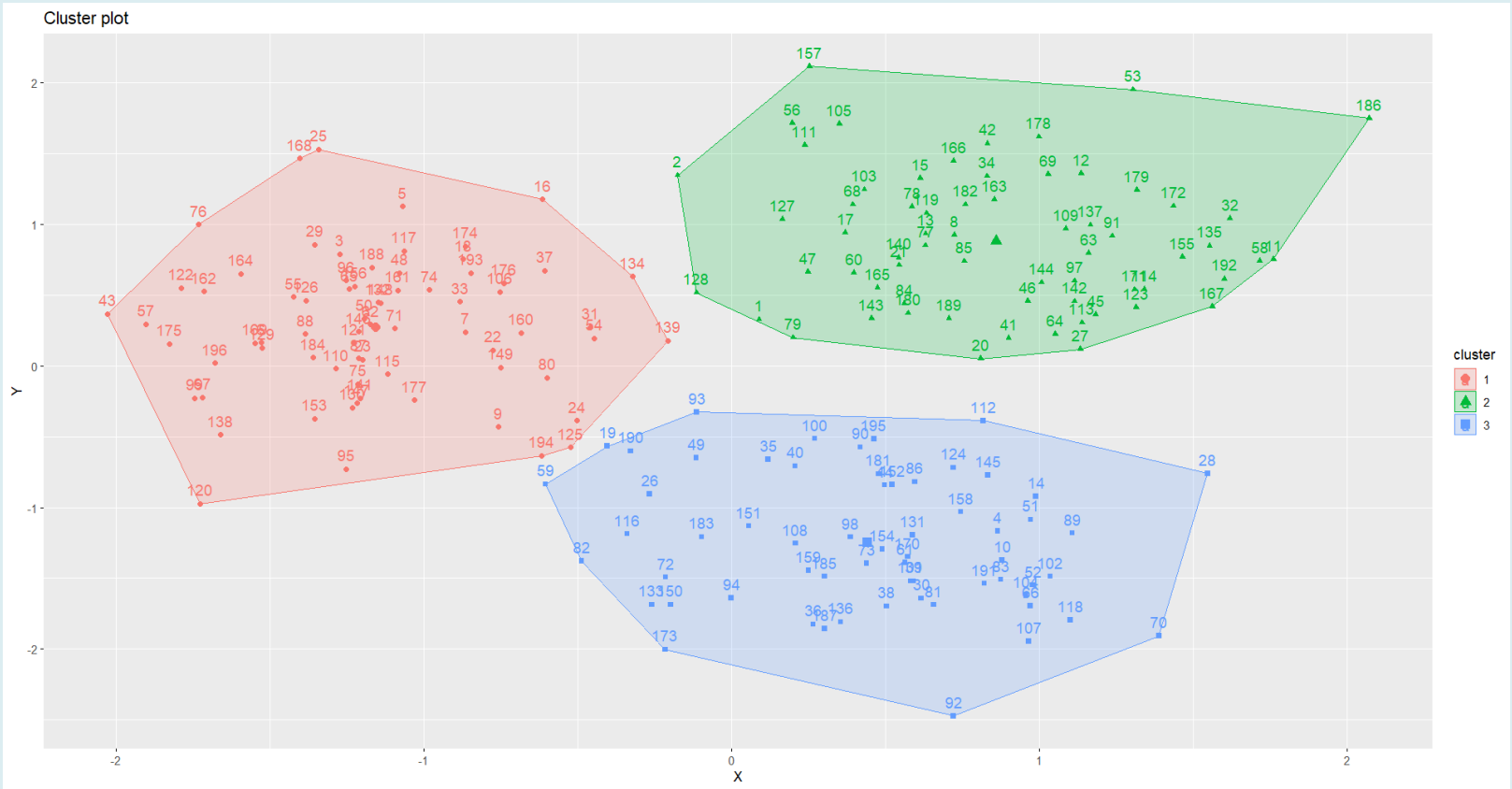
```
Cluster means:
          X         Y
1 0.8147681 0.6048477
2 0.3243930 0.4611514
3 0.7128718 0.1029810
```

这些是三个聚类的中心。

Lastly, we visualize the result:

最后，我们对结果进行可视化：

```
1 fviz_cluster(result, data = mydata)
```



How many groups to have?

There are several measures available, and one simple measure is the

$$\text{performance} = \frac{\text{between group sum of error}}{\text{total sum of error}}$$

The larger the value is, the better the performance is.

You can think it as

$1 - \text{performance} = \text{total distance from points to the nearest center}$

有多少个组合适?

有几种可用的度量方法，其中一种简单的度量是:

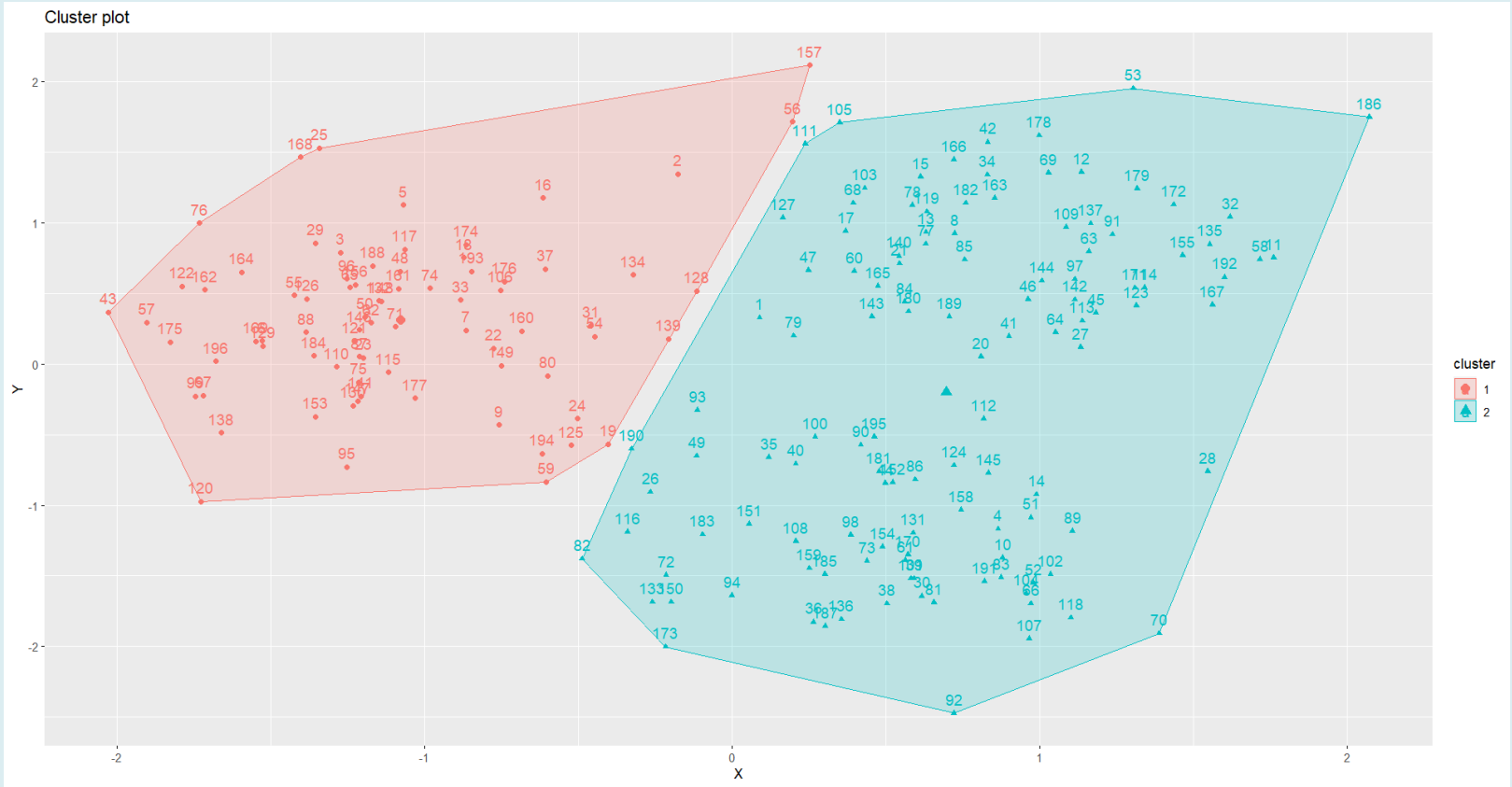
$$\text{performance} = \frac{\text{between group sum of error}}{\text{total sum of error}}$$

值越大，性能越好。你可以将其视为:

$$1 - \text{performance} = \text{total distance from points to the nearest center}$$

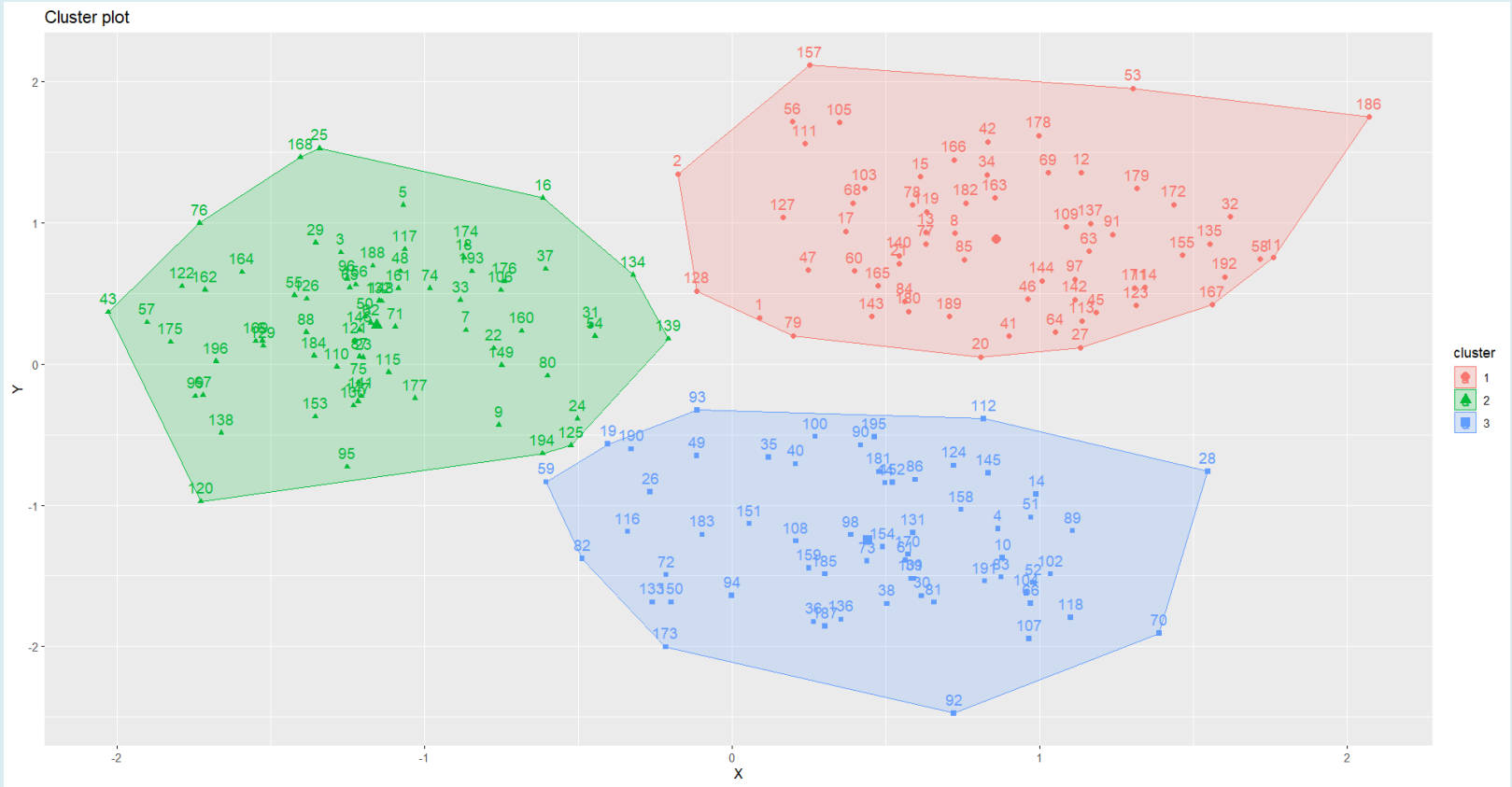
When $k = 2$, performance is 41.9%

当 $k = 2$ 时，性能为 41.9%



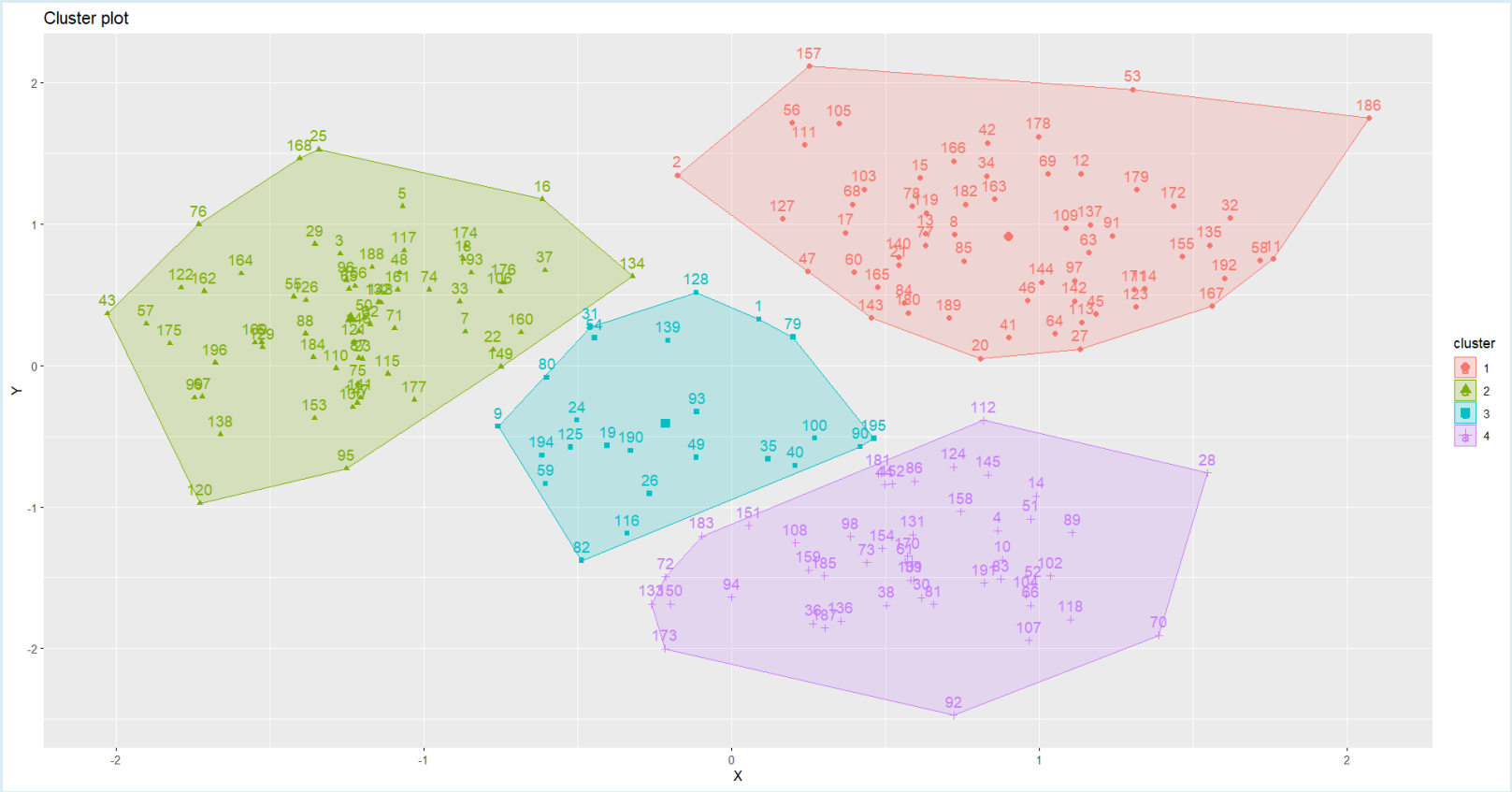
When $k = 3$, performance is 78.0%

当 $k = 3$ 时，性能为 78.0%



When $k = 4$, performance is 81.4%

当 $k = 4$ 时，性能为 81.4%



When $k = 5$, performance is 84.3%

当 $k = 5$ 时，性能为 84.3%

number of clusters	performance
2	41.9%
3	78.0%
4	81.4%
5	84.3%

While having more clusters always improve your performance, the improvement is low after $k = 3$. In this regard, having three clusters is considered enough.

聚类数量	性能
2	41.9%
3	78.0%
4	81.4%
5	84.3%

虽然增加聚类数量总是能提高性能，但在 $k = 3$ 之后，性能的提升幅度较小。在这种情况下，三个聚类被认为是足够的。

The complete code is here.

```
library(cluster)
library(factoextra)
mydata <- read_csv("https://ximarketing.github.io/data/clustering.csv")
result <- kmeans(mydata, centers = 3, nstart = 25)
result
fviz_cluster(result, data = mydata)
```

Latent Class Analysis

潜在类别分析

Suppose that you have a number of users, and you observe the behavior of each user, which is a binary variable. Here is an example:

For each consumer of financial institution, you observe:

1. Whether the consumer applied for a credit card.
2. Whether the consumer applied for a loan.
3. Whether the consumer defaulted in the past.
4. Whether the consumer transacted during the past week.
5. Whether the consumer issued a complaint before.

And so on... For each question, the answer is Yes or No.

假设你有多个用户，并观察每个用户的行为，这是一种二元变量。以下是一个示例：

对于每个金融机构的消费者，你观察到：

- 消费者是否申请了信用卡。
- 消费者是否申请了贷款。
- 消费者是否在过去违约。
- 消费者在过去一周是否进行了交易。
- 消费者是否曾提出过投诉。

等等……对于每个问题，答案都是“是”或“否”。

Now, we want to classify consumers into groups.

现在，我们希望将消费者分类为不同的组。

What does the word “latent” stand for?

Latent means hidden. In our context, it means there are hidden classes of consumers but we do not know what they are. On they other hand, we can also group individuals based on their observed characteristics such as gender, age group, which are not “latent.”

“Latent”这个词是什么意思？

“Latent”意为隐藏。在我们的语境中，它表示存在一些隐藏的消费者类别，但我们并不知道它们是什么。另一方面，我们也可以根据一些可观察的特征（如性别、年龄组）对个体进行分组，这些特征并不是“潜在的”。

Example 1: In a survey, respondents answered questions regarding their attitudes regarding social issues (e.g., Do you support capital punishment? Do you think the minimum wage should be increased?). Based on their answers, you can classify them into groups such as left-wing, right-wing, and central.

示例 1：在一项调查中，受访者回答了关于他们对社会问题态度的问题（例如：您支持死刑吗？您认为最低工资应该提高吗？）。根据他们的答案，您可以将他们分类为左翼、右翼和中间派等组别。

Example 2: A streaming APP observes the videos that consumers watch online, and classify consumers into groups based on their tastes. You will have groups that enjoy watching action movies, romantic movies, cartoons, etc.

示例 2：一个流媒体应用观察消费者在线观看的视频，并根据他们的喜好将消费者分类。您将会有喜欢观看动作片、爱情片、动画片等不同组别的用户。

Statistical Background (optional).

Suppose that there are two questions, A and B. Then, for an individual, the likelihood that the answers are X_A and X_B is given by

$$\sum_{\text{every class } j} \Pr[\text{Class j}] \times \Pr[X_A|\text{Class j}] \times \Pr[X_B|\text{Class j}]$$

Our goal is to find out the following values to maximize the likelihood:

$$\Pr[\text{Class } j], \Pr[X_A = \text{Yes}|\text{Class } j], \Pr[X_B = \text{Yes}|\text{Class } j].$$

统计背景（选学内容）.

假设有两个问题，A 和 B。那么，对于一个个体，答案为 X_A 和 X_B 的可能性为

$$\sum_{\text{every class } j} \Pr[\text{Class j}] \times \Pr[X_A|\text{Class j}] \times \Pr[X_B|\text{Class j}]$$

我们的目标是找出以下值，以最大化似然：

$$\Pr[\text{Class } j], \Pr[X_A = \text{Yes}|\text{Class } j], \Pr[X_B = \text{Yes}|\text{Class } j].$$

Statistical Background (optional).

How to find out these values? Typically, this is done by using the expectation-maximization (EM) method, which is beyond the scope of this class. However, if you are interested, you can find out the original paper on this topic:

Dempster, A.P., Laird, N.M. and Rubin, D.B., 1977. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society: Series B*, *39*(1), 1-22.

统计背景（选学内容）.

如何找到这些值？通常，这通过使用期望最大化（EM）方法来实现，但这超出了本课程的范围。然而，如果你感兴趣，可以查阅关于这个主题的原始论文：

Dempster, A.P., Laird, N.M. and Rubin, D.B., 1977. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society: Series B*, *39*(1), 1-22.

We ask whether a student likes the following subjects: English, History, Art, Mathematics, Physics, and Biology. For each subject, the answer is Yes or No.

When you run LCA in R, the class should be indicated as 2 or 1. So, we use 2 to denote Yes (2 = Yes) and 1 to denote No (1 = No).

我们询问学生是否喜欢以下科目：英语、历史、艺术、数学、物理和生物。对于每个科目，答案是“是”或“否”。

当你在 R 中运行 LCA 时，类别应标记为 2 或 1。因此，我们用 2 表示“是”（2 = 是），用 1 表示“否”（1 = 否）。

The data is as follows. 数据如下

	A	B	C	D	E	F	G
1	ID	english	history	art	math	physics	biology
2	1	1	1	1	2	2	1
3	2	1	1	1	1	1	1
4	3	2	1	1	2	2	2
5	4	1	2	2	2	1	2
6	5	1	1	2	1	1	1
7	6	2	1	1	2	1	2
8	7	2	1	1	2	2	2
9	8	1	1	1	2	1	1
10	9	1	1	1	1	1	2

First, we load data from the internet.

```
1 library(readr)
2 library(poLCA)
3 mydata <- read_csv("https://ximarketing.github.io/data/LCA.csv")
4 head(mydata)
```

Here, poLCA is the most commonly used R package for LCA.

	ID	english	history	art	math	physics	biology
	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
1	1	1	1	1	2	2	1
2	2	1	1	1	1	1	1
3	3	2	1	1	2	2	2
4	4	1	2	2	2	1	2
5	5	1	1	2	1	1	1
6	6	2	1	1	2	1	2

首先，我们从互联网加载数据。

```
library(readr)
library(poLCA)
mydata <- read_csv("https://ximarketing.github.io/data/LCA.csv")
head(mydata)
```

在这里，poLCA 是最常用的 R 包用于潜在类别分析（LCA）。

	ID	english	history	art	math	physics	biology
	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
1	1	1	1	1	2	2	1
2	2	1	1	1	1	1	1
3	3	2	1	1	2	2	2
4	4	1	2	2	2	1	2
5	5	1	1	2	1	1	1
6	6	2	1	1	2	1	2

```
1 f <- cbind(english, history, art, math, physics, biology)~1
```

In the above code, we use the cbind (column bind) function to merge the columns that we want to analyze. ~1 refers to the simple LCA model (there are complex versions of LCA).

在上述代码中，我们使用 cbind（列绑定）函数来合并我们想要分析的列。~1 指的是简单的 LCA 模型（还有复杂版本的 LCA）。

```
set.seed(5620)
LCA <- poLCA(f, data=mydata, nclass=4)
```

Lastly, we run LCA model with 4 latent classes.

The complete codes are as follows.

```
library(readr)
library(poLCA)
mydata <- read_csv("https://ximarketing.github.io/data/LCA.csv")
head(mydata)
f <- cbind(english, history, art, math, physics, biology)~1
set.seed(5620)
LCA <- poLCA(f, data=mydata, nclass=4)
```

```
set.seed(5620)
LCA <- poLCA(f, data=mydata, nclass=4)
```

最后，我们运行具有 4 个潜在类别的 LCA 模型。 完整的代码如下：

```
library(readr)
library(poLCA)
mydata <- read_csv("https://ximarketing.github.io/data/LCA.csv")
head(mydata)
f <- cbind(english, history, art, math, physics, biology)~1
set.seed(5620)
LCA <- poLCA(f, data=mydata, nclass=4)
```

```
$english
            Pr(1)  Pr(2)
class 1:  0.3416 0.6584
class 2:  0.8429 0.1571
class 3:  0.1409 0.8591
class 4:  0.6980 0.3020
```

Here, each class represents one segment.

With probability 65.8%, a person in class 1 likes English.

With probability 15.7%, a person in class 2 likes English.

With probability 85.9%, a person in class 3 likes English.

With probability 30.2%, a person in class 4 likes English.

```
$english
            Pr(1)  Pr(2)
class 1:  0.3416 0.6584
class 2:  0.8429 0.1571
class 3:  0.1409 0.8591
class 4:  0.6980 0.3020
```

在这里，每个类别代表一个细分。

- 在类别 1 中，有 65.8% 的概率一个人喜欢英语。
- 在类别 2 中，有 15.7% 的概率一个人喜欢英语。
- 在类别 3 中，有 85.9% 的概率一个人喜欢英语。
- 在类别 4 中，有 30.2% 的概率一个人喜欢英语。

```
$english
            Pr(1)  Pr(2)
class 1:  0.3416 0.6584
class 2:  0.8429 0.1571
class 3:  0.1409 0.8591
class 4:  0.6980 0.3020

$history
            Pr(1)  Pr(2)
class 1:  0.2684 0.7316
class 2:  0.8344 0.1656
class 3:  0.2345 0.7655
class 4:  0.9289 0.0711
```

```
$art
            Pr(1)  Pr(2)
class 1:  0.2617 0.7383
class 2:  0.9022 0.0978
class 3:  0.1612 0.8388
class 4:  0.7949 0.2051

$math
            Pr(1)  Pr(2)
class 1:  0.7348 0.2652
class 2:  0.3489 0.6511
class 3:  0.2235 0.7765
class 4:  0.8068 0.1932
```

```
$physics
            Pr(1)  Pr(2)
class 1:  0.8498 0.1502
class 2:  0.2735 0.7265
class 3:  0.2312 0.7688
class 4:  0.9952 0.0048

$biology
            Pr(1)  Pr(2)
class 1:  0.8868 0.1132
class 2:  0.2112 0.7888
class 3:  0.1678 0.8322
class 4:  0.8871 0.1129
```

Class 1: Like art/humanities but hate science ("文科型")

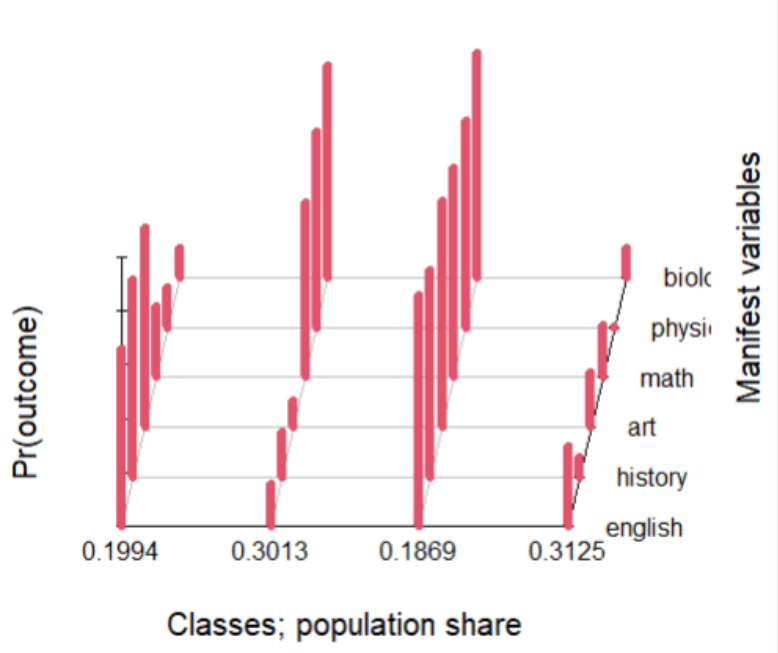
Class 2: Like science but hate art/humanities (“理科型”)

Class 3: Like all subjects ("学霸")

Class 4: Hate every subject (“学渣”)

We can further visualize the results:

```
1 plot(LCA)
```



```
predicted_class <- cbind(mydata, "Predicted LC" = LCA$predclass)
head(predicted_class)
```

Next, we want to the type (class) of each individual in the dataset. We can run the above code to see the results.

	ID	english	history	art	math	physics	biology	Predicted LC
1	1	1	1	1	2	2	1	2
2	2	1	1	1	1	1	1	4
3	3	2	1	1	2	2	2	2
4	4	1	2	2	2	1	2	3
5	5	1	1	2	1	1	1	4
6	6	2	1	1	2	1	2	2

```
predicted_class <- cbind(mydata, "Predicted LC" = LCA$predclass)
head(predicted_class)
```

接下来，我们想要查看数据集中每个个体的类别（类型）。我们可以运行上述代码来查看结果。

	ID	english	history	art	math	physics	biology	Predicted LC
1	1	1	1	1	2	2	1	2
2	2	1	1	1	1	1	1	4
3	3	2	1	1	2	2	2	2
4	4	1	2	2	2	1	2	3
5	5	1	1	2	1	1	1	4
6	6	2	1	1	2	1	2	2

Question: How many classes should be included?

问题：应该包含多少个类别?

Classes	2	3	4	5
AIC	15,559	14,859	14,732	14,731
BIC	15,632	14,971	**14,883**	14,921

Two most commonly used criteria are BIC and AIC, while BIC is more popular (Nylund et al. concluded that BIC tends to perform better than AIC especially when N is large). They two measures are largely aligned, and we choose the number of classes that minimize their BIC/AIC.

类别数	**2**	**3**	**4**	**5**
AIC	15,559	14,859	14,732	14,731
BIC	15,632	14,971	**14,883**	14,921

两种最常用的标准是 BIC 和 AIC，其中 BIC 更为流行（Nylund 等人得出结论，BIC 在样本量较大时通常表现优于 AIC）。这两个指标大体一致，我们选择使 BIC/AIC 最小化的类别数量。