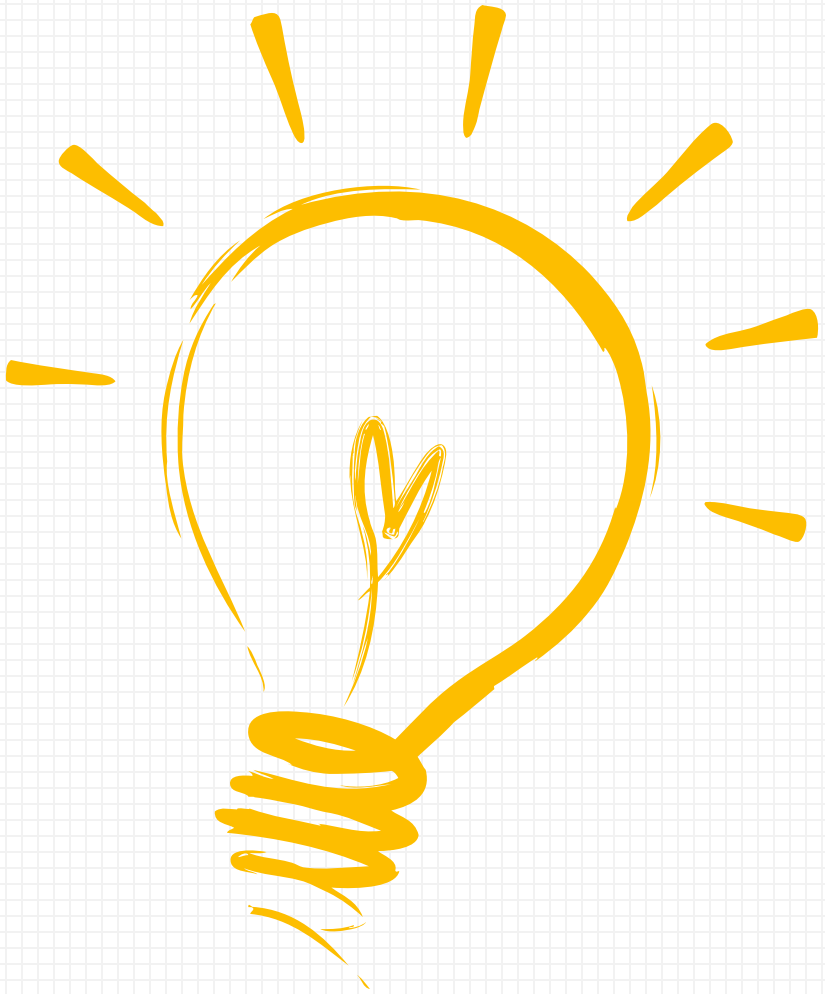
数据挖掘算法

K-Means聚类

CONTENTS



di yi zhang jie

第一章节

算法背景

1.1 前置知识

监督学习（英语：Supervised learning），又叫有监督学习，监督式学习，是机器学习的一种方法，可以由训练资料中学到或建立一个模式（函数 / learning model），并依此模式推测新的实例[1]。训练资料是由输入物件（通常是向量）和预期输出所组成。

常见的有监督学习:分类,回归分析等.

无监督学习（英语：unsupervised learning）是机器学习的一种方法，没有给定事先标记过的训练示例，自动对输入的资料进行分类或分群。

最常见的无监督学习:聚类.

简单来看,有监督学习就是”对数据进行预先学习”,然后再去解决任务.无监督学习就是没有预先的学习行为,直接解决任务.

1.2 什么是聚类

聚类是一种包括数据点分组的机器学习技术。

聚类是一种无监督学习的方法，是一种在许多领域常用的统计数据分析技术

给定一组数据点，我们可以用聚类算法将每个数据点分到特定的组中。 理论上，属于同一组的数据点应该有相似的属性或特征，而属于不同组的数据点应该有非常不同的属性或特征。

聚类算法有很多种,比如层次聚类,划分聚类,密度聚类等等。

K-Means聚类是可能是我们最熟知的聚类算法之一。它在很多介绍性的数据科学和机器学习课程中出现过。因为很容易理解并且容易用代码实现，所以在这里我们首先对该算法进行学习。

di yi zhang jie

第二章节

算法原理

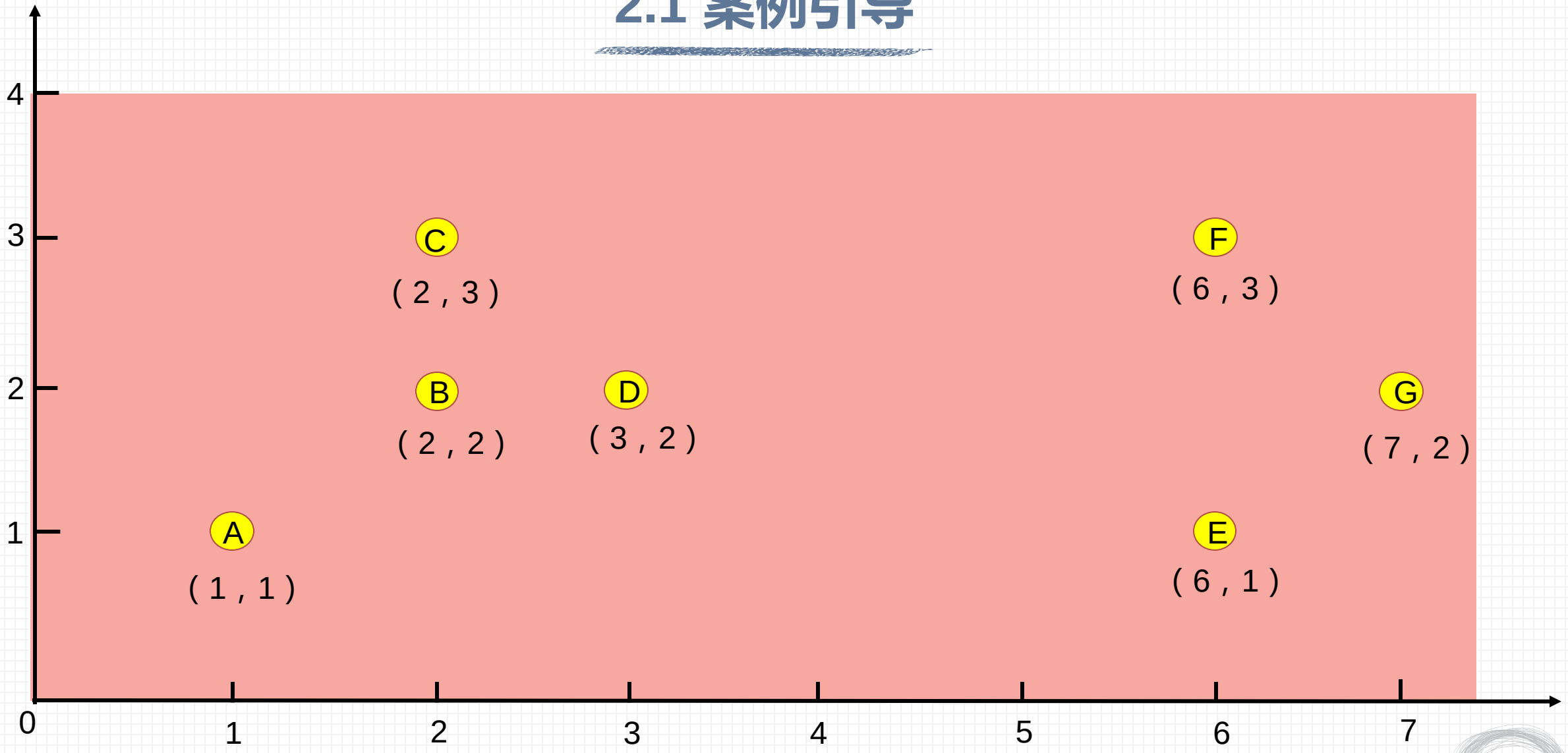
2.1 案例引导

研究人员对某种植物抽检其各项生长指标部分结果如下,共7条记录：

标号	茎长（cm）	叶宽（cm）	……
A	1	1	……
B	2	2	……
C	2	3	……
D	3	2	……
E	6	1	……
F	6	3	……
G	7	2	……

不失一般性，这里只研究两个属性，多个属性采用的算法逻辑一致。

2.1 案例引导



2.2 必要知识

欧氏距离： 在欧几里得空间，点$x = (x_1, x_2 \dots\dots x_n)$与点$y = (y_1, y_2 \dots\dots y_n)$之间的欧氏距离为

$$d(x,y) = \sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2 + \cdots + (x_n - y_n)^2}$$

它是一个纯数值。

显然在二维空间，点$x = (x_1, x_2)$与点$y = (y_1, y_2)$之间的欧式距离为

$$d(x,y) = \sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2}$$

2.2 必要知识

数据的标准化(normalization)：是将数据按比例缩放，使之落入一个小的特定区间。在某些比较和评价的指标处理中经常会用到，去除数据的单位限制，将其转化为无量纲的纯数值，便于不同单位或量级的指标能够进行比较和加权。

其中最典型的就是数据的归一化处理，即将数据统一映射到[0,1]区间上。这里只介绍一种归一化处理方法：min-max标准化。

$$x' = \frac{x - x_{min}}{x_{max} - x_{min}}$$

经处理后， 点x的坐标便会统一映射到[0,1]区间上。

2.3 算法流程

输入 ： 聚类个数k

Step1：**随机选择**k个样本作为初始聚类中心$C = \{c_1, c_2 \dots c_k\}$

Step2：针对数据集中每个样本x_i，分别计算该样本到K个聚类中心的距离，并将其分到距离最小的聚类中心所对应的类中

Step3：针对每个类别c_j，重新计算它的聚类中心

$$c_j' = \frac{1}{n}\sum_{x_i \in c_j} x_i$$

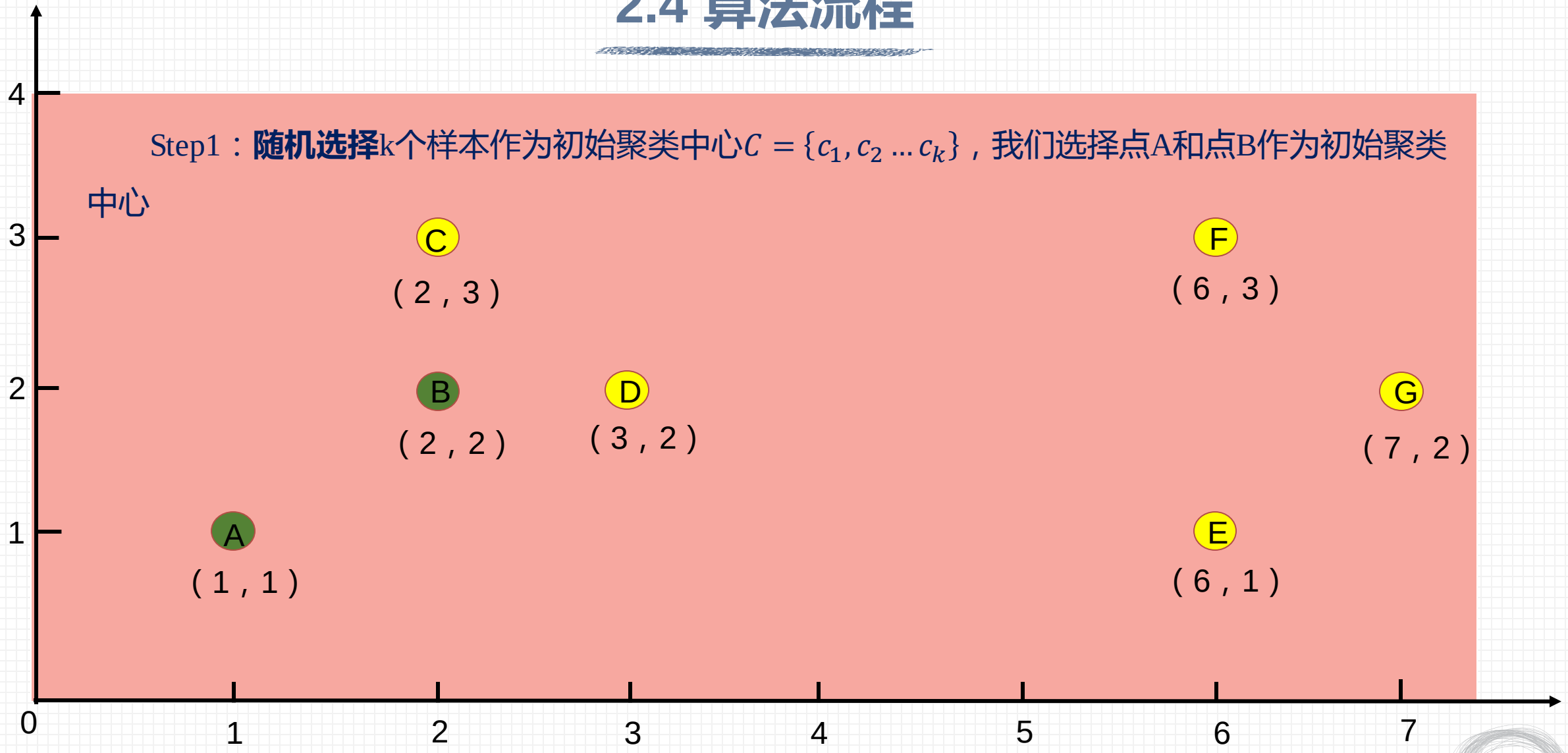
其中n为类别c_j中所包含数据点的个数

Step4：重复上述2,3两步操作，直到达到某个中止条件（迭代次数、聚类中心不再变化等等）

输出 ：聚类结果

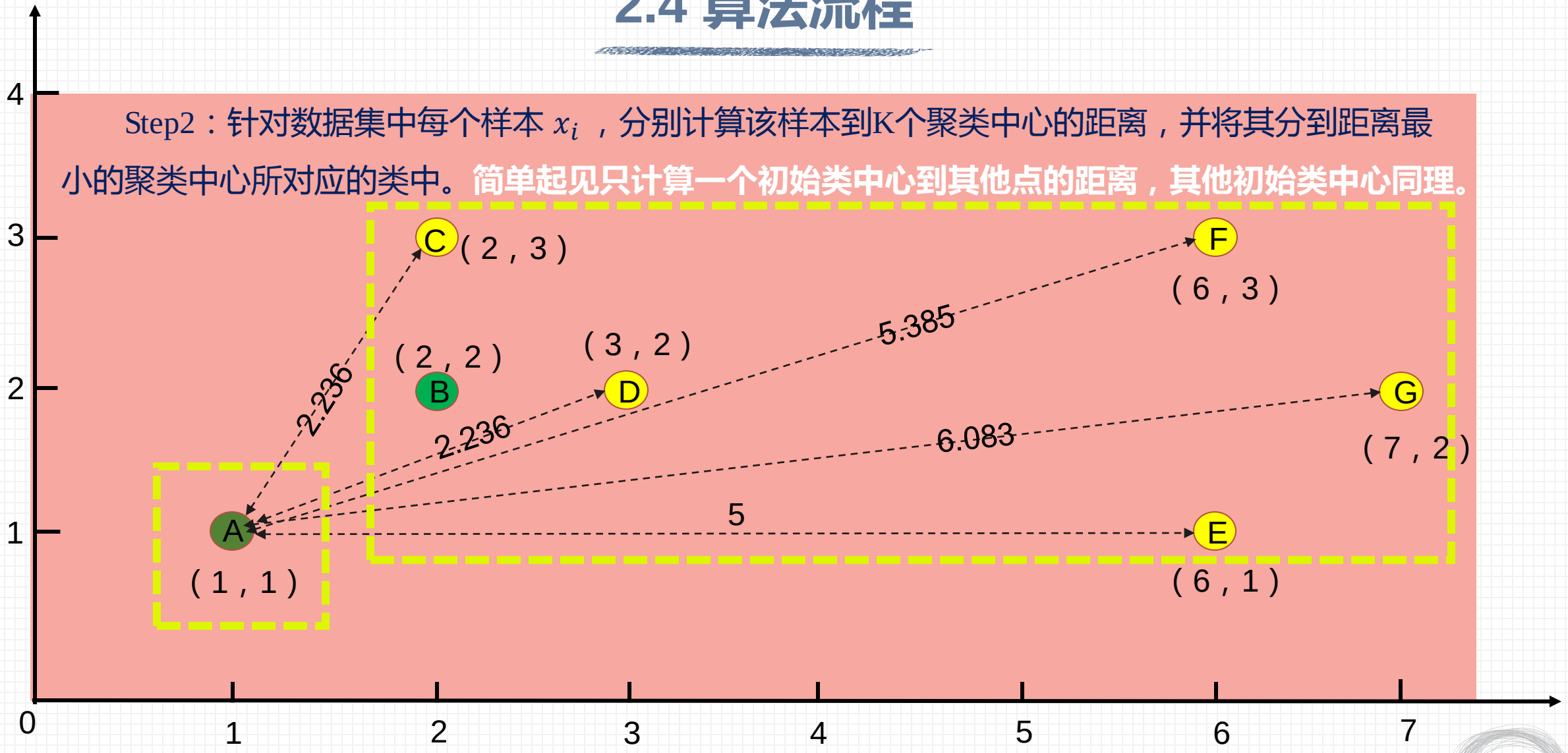
2.4 算法流程

Step1：**随机选择**k个样本作为初始聚类中心$C = \{c_1, c_2 \ldots c_k\}$，我们选择点A和点B作为初始聚类中心



2.4 算法流程

Step2：针对数据集中每个样本 x_i ，分别计算该样本到K个聚类中心的距离，并将其分到距离最小的聚类中心所对应的类中。**简单起见只计算一个初始类中心到其他点的距离，其他初始类中心同理。**

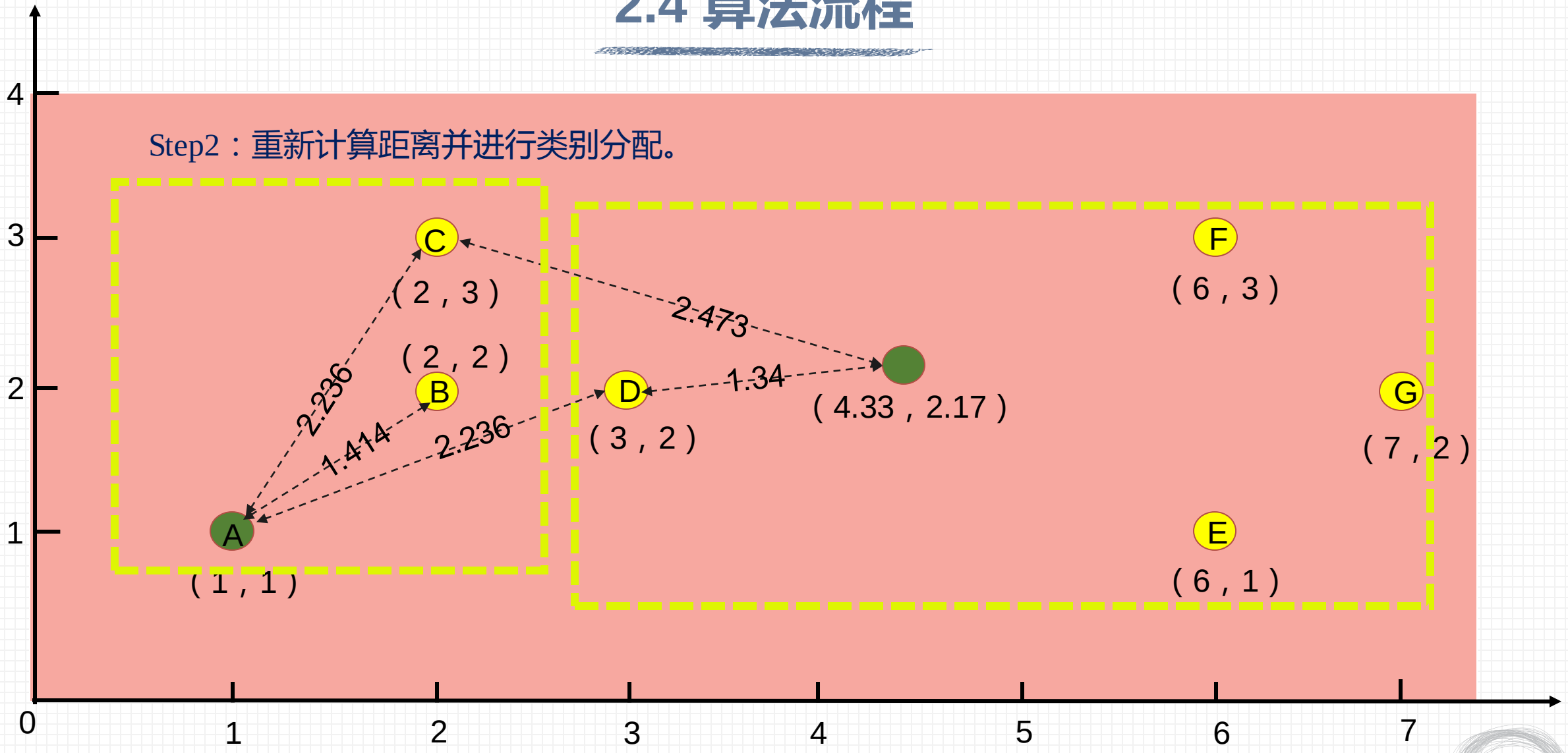


类别1：A　　类别2：BCDEFG

2.4 算法流程

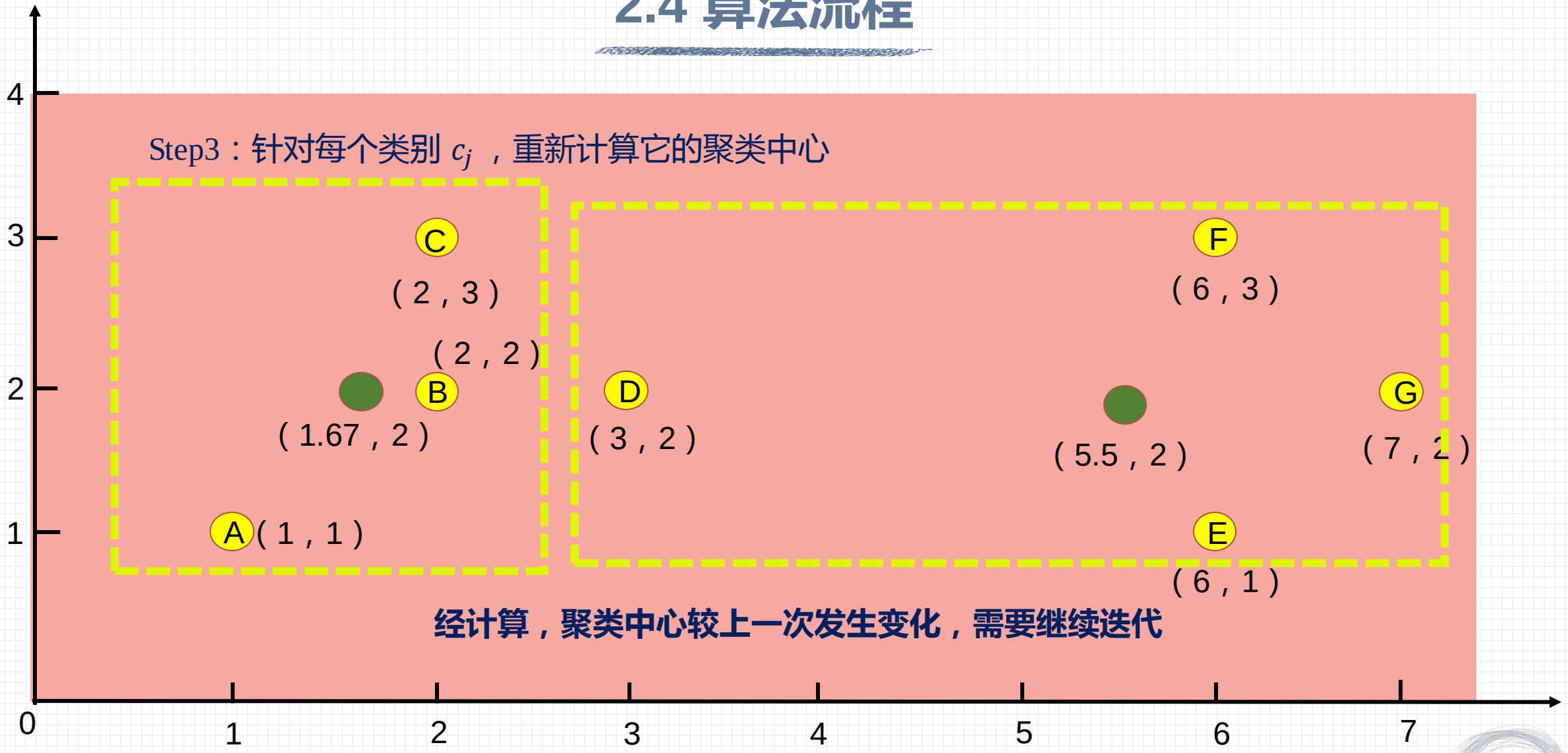
类别1：A　　类别2：BCDEFG

2.4 算法流程



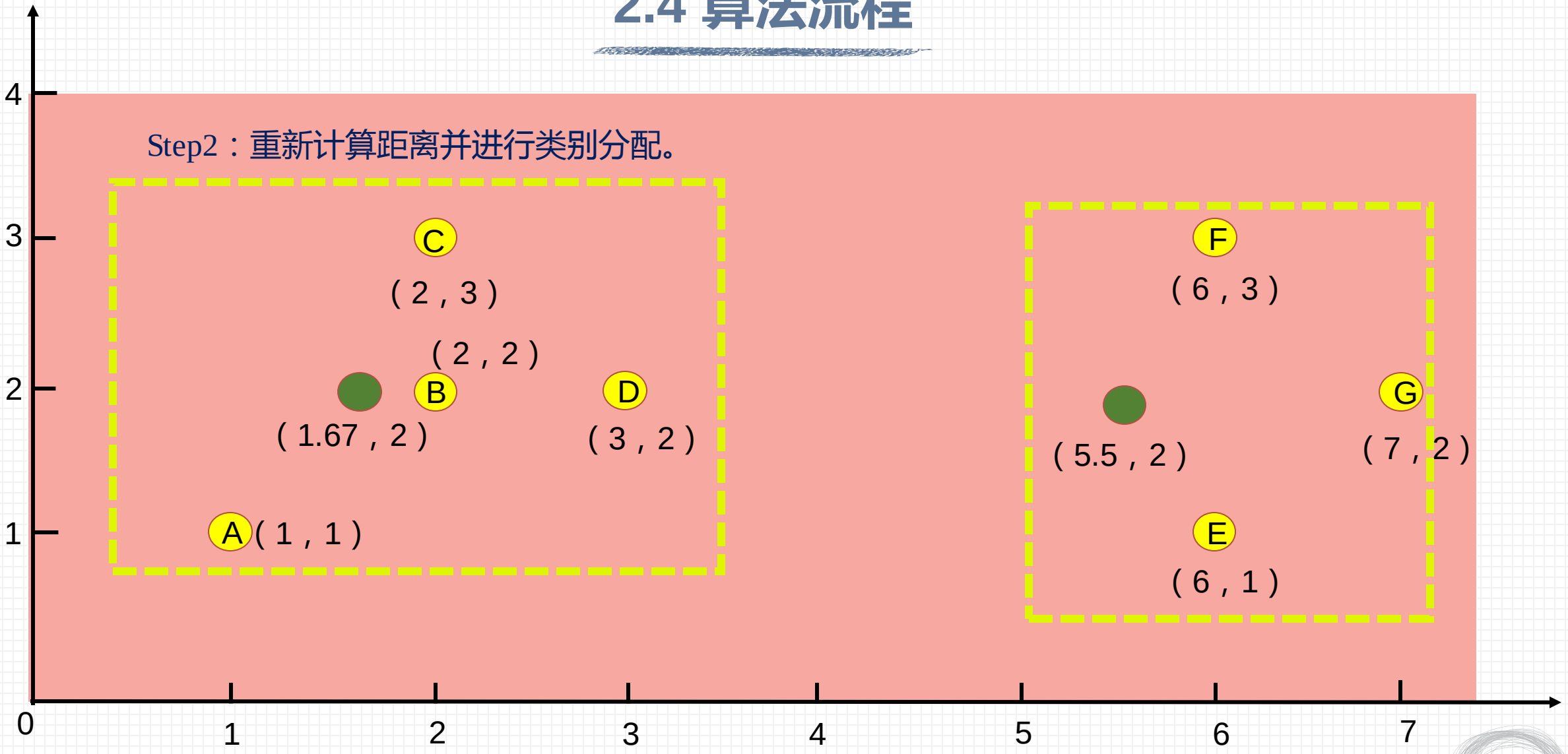
类别1：ABC　　类别2：DEFG

2.4 算法流程



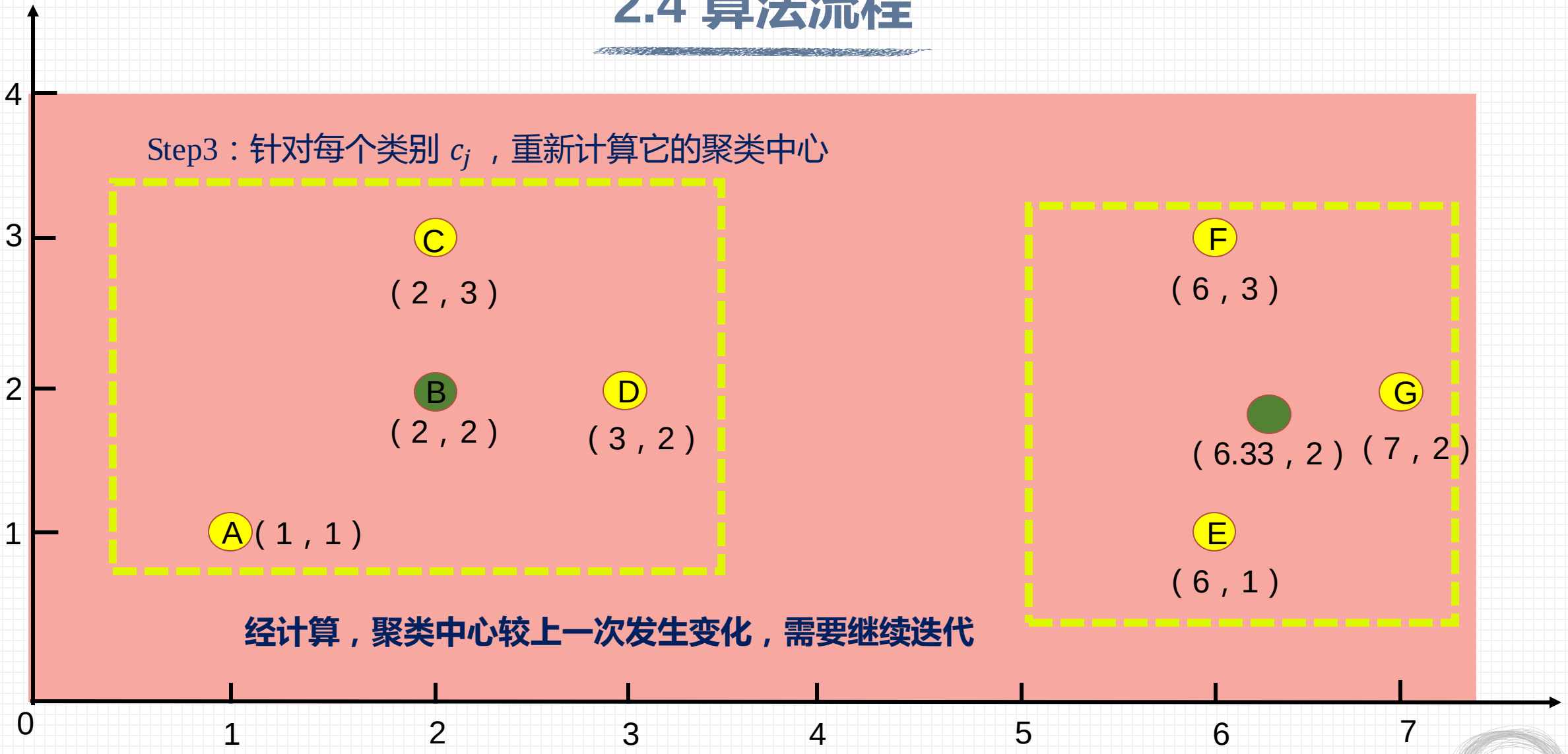
类别1：ABC　　类别2：DEFG

2.4 算法流程

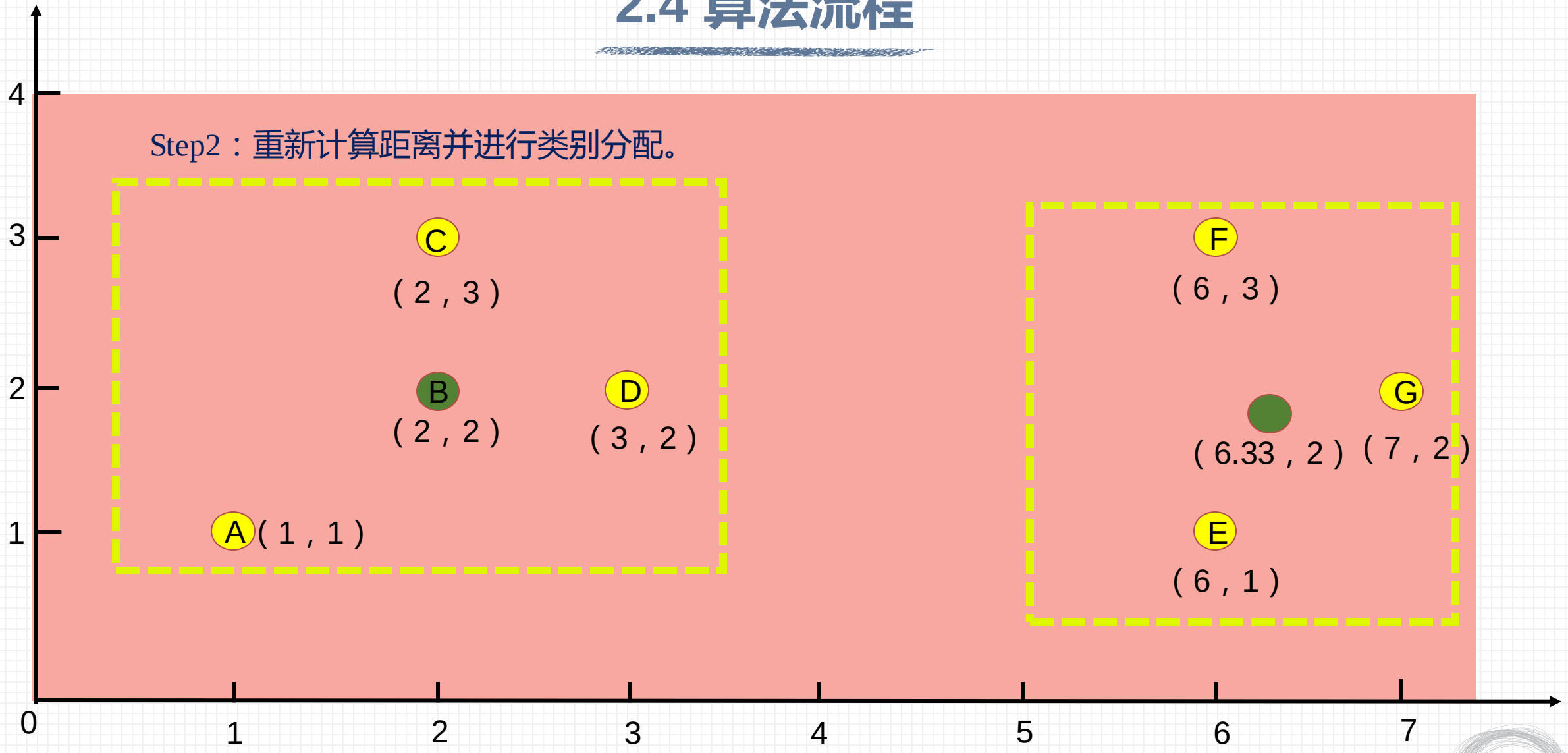


类别1：ABCD　　类别2：EFG

2.4 算法流程

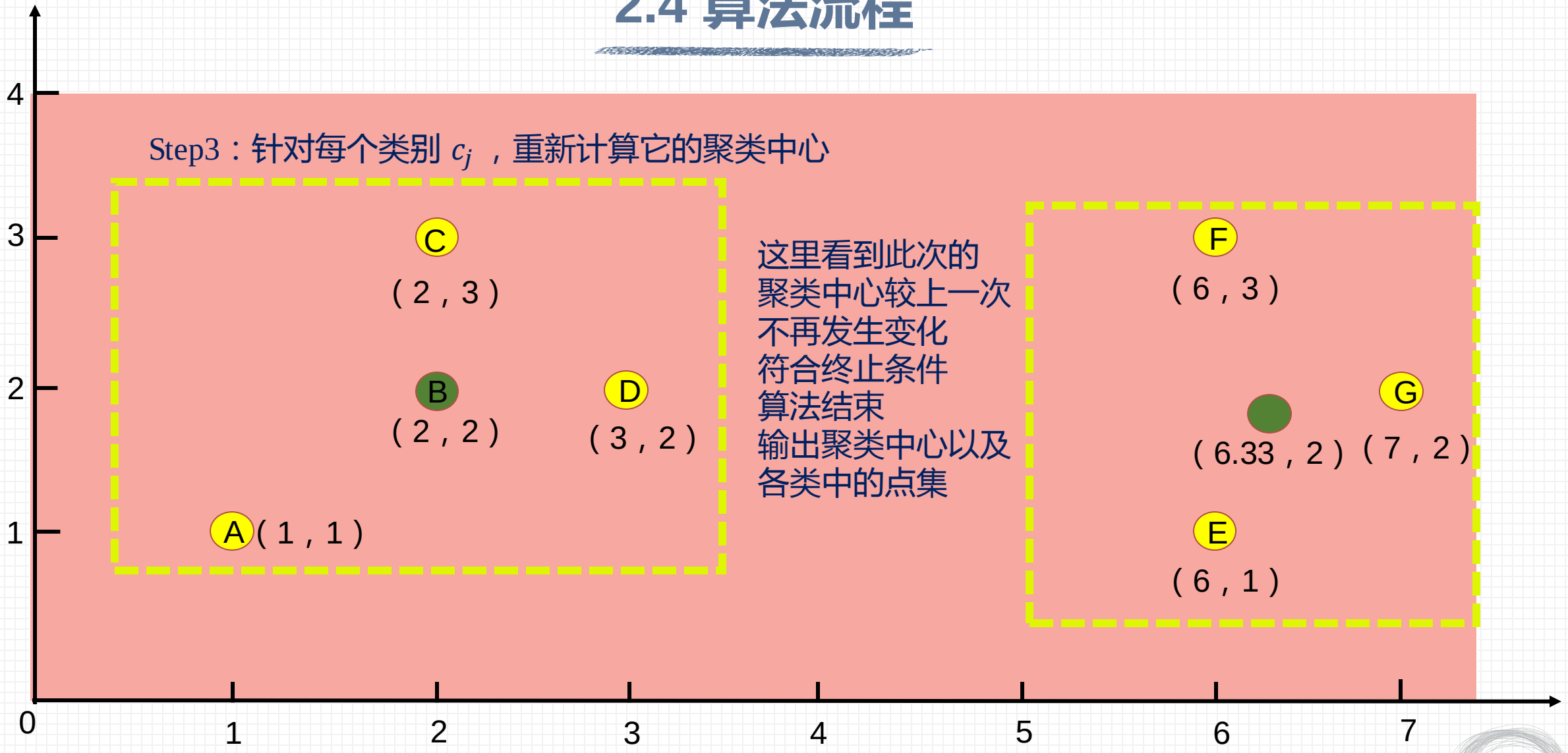


2.4 算法流程



类别1：ABCD　　类别2：EFG

2.4 算法流程



类别1：ABCD　　类别2：EFG

di san zhang jie

第三章节

算法分析

3.1 算法分析

优点：

1）原理比较简单，实现也是很容易，收敛速度快。

2）聚类效果较优。

3）算法的可解释度比较强。

4）主要需要调参的参数仅仅是簇数k。

缺点：

1）K值的选取不好把握。

2）初始聚类中心的选择

3）采用迭代方法，得到的结果不一定是全局最优解。

4）对噪音和异常点比较的敏感等

di si zhang jie

第四章节

算法拓展

4.1 算法拓展

针对经典K-Means算法的缺点，很多前辈提出或者改进出了更加高效的聚类算法，如在K-Means基础上的K-Means++，K-Medoids。基于密度聚类的DBSCAN算法，DPC算法。基于图论的Single-link，complete-link等等。

以下是几篇这方面的文章，大家感兴趣可以看看。

[1] https://doi.org/10.1016/j.knosys.2017.07.010

[2] https://doi.org/10.1016/j.fss.2009.06.012

[3] https://blog.csdn.net/tyh70537/article/details/76768802

[4] https://wenku.baidu.com/view/3396bb4d6294dd88d0d26bee.html

di wu zhang jie

第五章节

案例实操

5.1 案例实操

这一部分在下次视频中讲解，我们使用Python中经典机器学习第三方库scikit-learn，来操作一下。

需要做的准备是：

首先要有Python环境，最好可以安装上pycharm。

然后安装一下必要的库（以后也需要的）：如numpy，pandas，xlrd，scikit-learn库等

如果你现在没有Python环境以及pycharm，或者不清楚第三方库如何安装，以及pip文件如何设置，pycharm的一些设置等，可以看一下我之前的基于Python实现网络爬虫的视频的第一课：

https://www.bilibili.com/video/BV1WV411U7LQ

同时如果有一些其他问题，可以联系我

QQ：1366420642，Q群：1019030249

欢迎大佬萌新加入

参　考　资　料

【1】https://zh.wikipedia.org/wiki/监督学习

【2】https://zh.wikipedia.org/wiki/欧几里得距离

THANKS

谢谢观看