无监督学习_聚类_吴恩达_机器学习笔记

1、无监督学习

数据不带有任何标签。

聚类算法：通过算法找到隐含的数据结构形成的簇。

聚类算法的用途：市场用户分群、社交网络分析（找到关系密切的群体）、计算机集群重新布局网络和资源分配、天文学中的星系分类。

评估聚类算法除了算法本身：还可以将原本带标签的数据先去掉标签，然后通过聚类评估的分类，与原来的标签进行比对评估，以此来看聚类算法的效能。

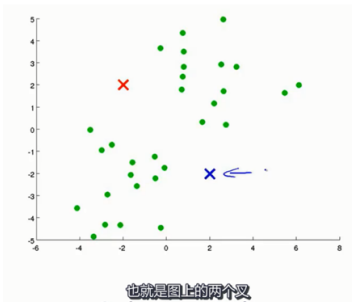
2、K_means算法

2.1 K_means算法步骤

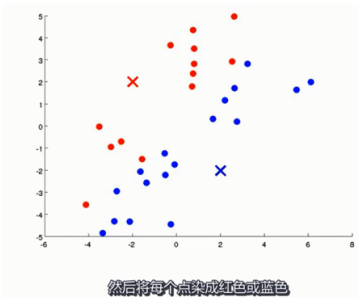
聚类算法中K均值算法是最常见的一种，下面将通过案例说明K均值算法的具体过程：

- 第一步：随机生成多个聚类中心。聚类中心随机生成的个数取决于前期定义的聚类数量。
- 第二步：K均值算法会经过两个重要步骤，一个是簇分配（遍历每个样本并计算距离，然后根据每个样本离初始聚类中心的距离将其分配到其中某一个簇）。
- 第三步：是移动聚类中心。根据新分配的簇，计算所有点的距离的平均值，然后就会找到每个簇新的聚类中心点。
- 第四步：计算新的聚类中心点和所有的点的距离，并重新分配簇，然后循环将簇的新中心计算出来。
- 循环迭代的停止条件：规定的迭代次数或者聚类中心的距离变化量低于阀值。阀值可以是0或者一个很小的实数。

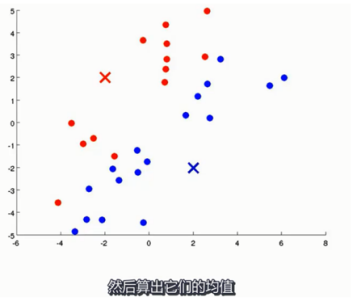
K均值算法的核心步骤



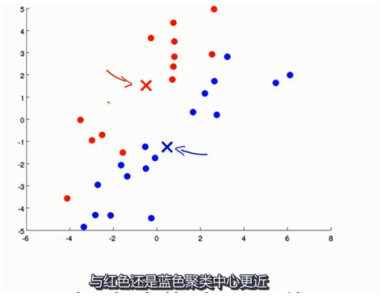
1.随机生成初始化的聚类中心点



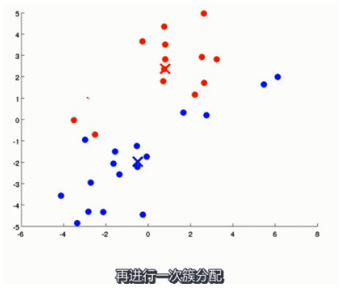
2.计算每个点与聚类中心的距离，然后将每个点分配给更近的聚类中心点并形成各自的簇。



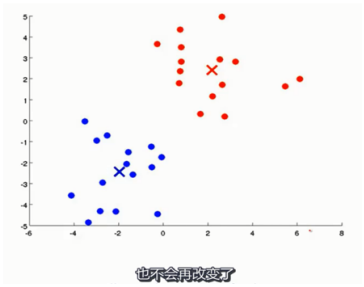
3.计算所有各自形成簇的点的均值，并将这个均值点定义为新的聚类中心。红色点、蓝色点的各自平均值。



4.新的聚类中心再次与所有数据点计算距离，将每个点重新分配给更近的聚类中心。



5.再根据新的簇分配计算所有簇内的点的向量均值，然后形成新的数据中心。如此循环。



6.迭代停止条件：规定迭代次数或者聚类中心距离变化量低于阀值。

2.2 算法步骤规划

#输入选择

1. K值选择
2. 不含label的数据集

x属于n维实数向量，不是n+1维实数向量

```
# 随机选择K个聚类中心u1,u2.....uK
# 定义for循环嵌套如下

#######簇分配步骤########
fro i = 1 in m：
    # c(i)表示第1到K个距离某个聚类中心x(i)最近的训练样本，c(i)是一个1到K的数
    # 分配训练样本到各自的簇

#######移动聚类中心########
for k=1 in K：
     uk:计算簇内所有点的均值
```

两种计算簇分配的方法:

1. 第一个方法是计算某个样本距离所有的初始聚类中心谁更近，然后就开始分配每一个样本到最近的聚类中心并形成簇 $min_k \left\| x^i - \mu_k \right\|$ ，求某个样本到所有聚类中心的最小的那个距离，作为簇分配依据。

2. 第二个方法是计算距离的平方。 $min_k \left\| x^i - \mu_k \right\|^2$ 。然后将最小的聚类中心的标签分配给该样本。一般第二个办法更常用。

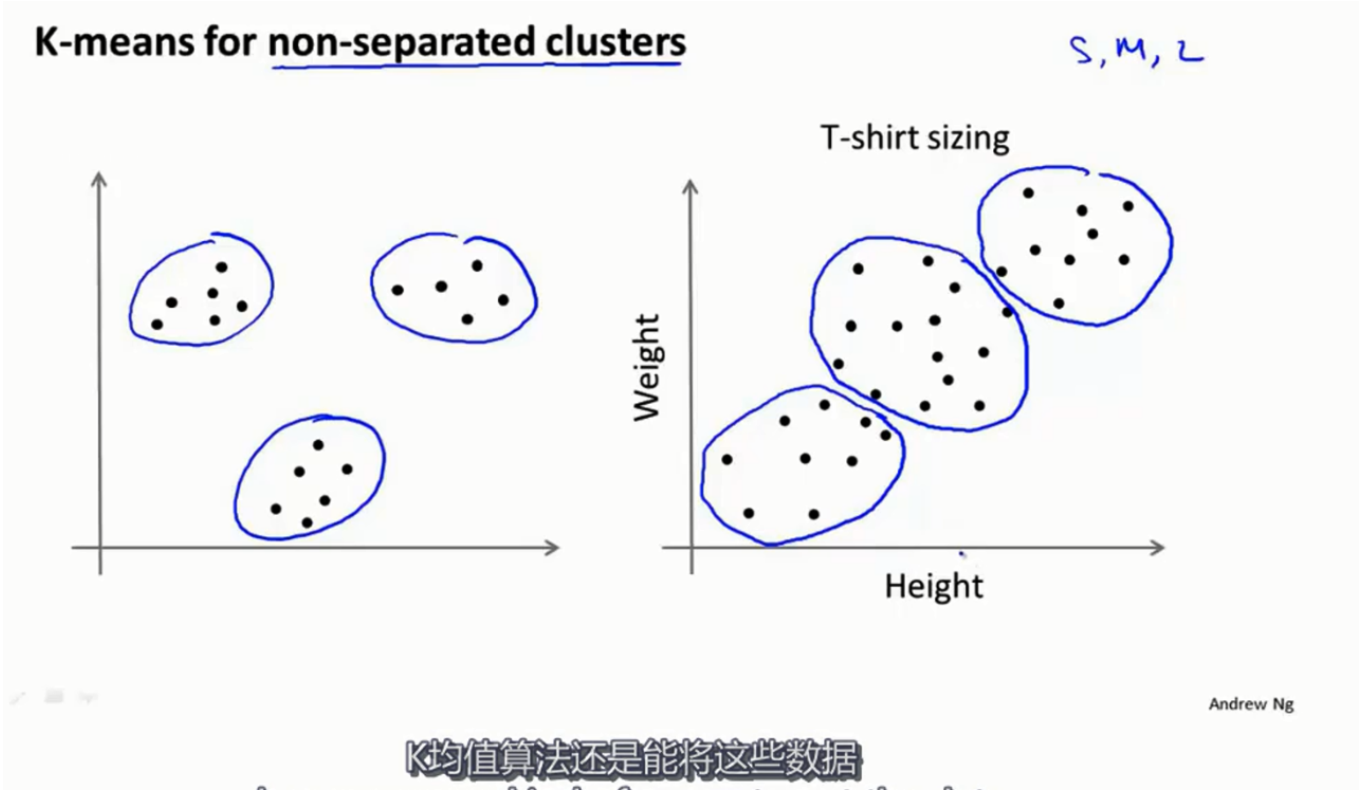
计算均值的步骤如下：

- 假设现有 $x_{(1)}, x_{(3)}, x_{(5)}, x_{(8)}$ 4个样本均为n维实数向量。
- 通过簇分配以后，他们的簇标签均为 $c^1 = 2, c^3 = 2, c^5 = 2, c^8 = 2$。
- 计算多个向量的平均值，其中平均值也是一个向量，这个就是新的聚类中心。

$$\mu_2 = \frac{1}{4}\left(x^{(1)} + x^{(3)} + x^{(5)} + x^{(8)}\right), x^{(i)} \in \mathbb{R}^n$$

问题：假设聚类中心计算出来以后，并不在该簇的中心点，有可能在簇的外面，这时候通常的做法是重新随机初始化新的聚类中心。或者是移除那个没有点的聚类中心。

2.3 聚类中心的业务场景选择



有时候，聚类中心并不像左图那样明确的把数据分成非常明显的簇。但是根据业务场景和需求，仍然可以将数据分成不同的簇。这非常取决于对业务场景的理解。

3、优化目标

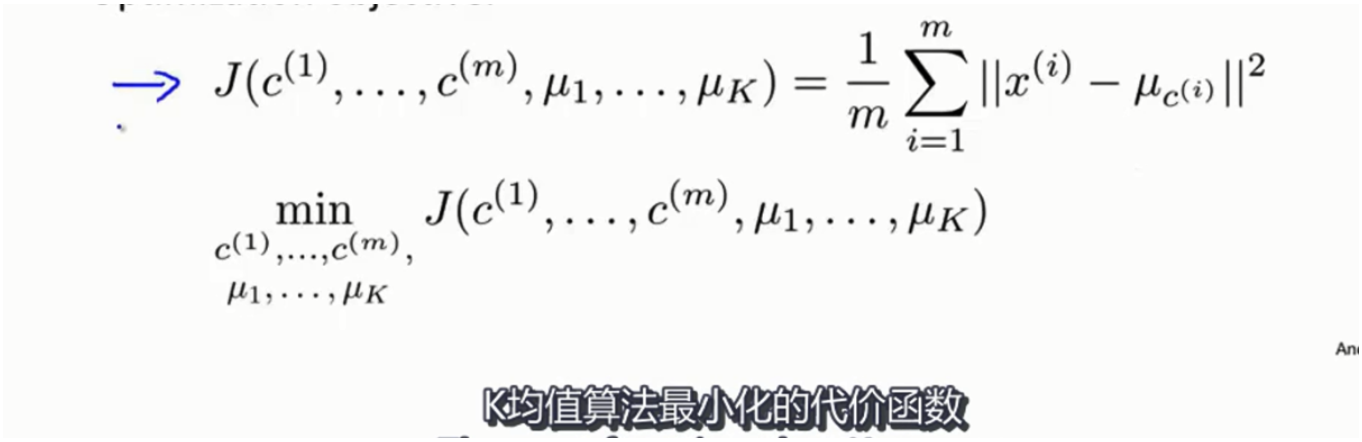
优化目标函数的作用：帮助K均值算法找到最优的簇，并且避免局部最优解。

K均值代价函数表示如下：

三个重要参数：

1. $c^{(i)}$ 表示当前样本所属的那个簇的索引或序号。大写的K表示簇的数量，小写的k表示聚类中心的下标。因此 $k \in \{1,2 \ldots,K\}$。
2. u_k 表示第k个聚类中心的位置。
3. $u_{c^{(i)}}$ 表示x(i)所属的那个簇的聚类中心。 假设x(i)划分到了第5个簇，那么c^(i) =5，即 $u_{c^{(i)}} = u_5$。这里指的就是第5个簇的聚类中心。

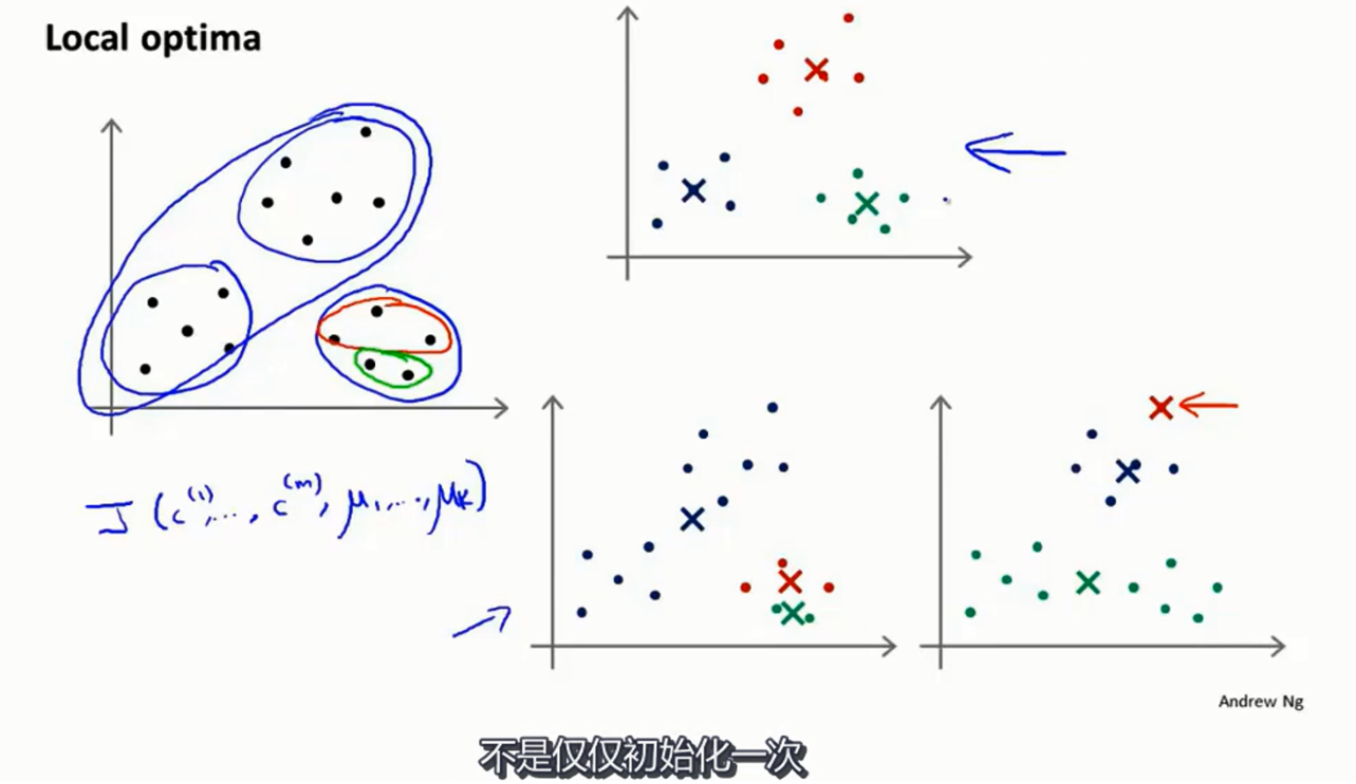
K均值算法的代价函数表示如下：



即每个样本到自己的聚类中心的距离的平方的平均值。通过这个代价函数，需要找到的参数就是每个样本归属最终簇的索引c(i)和每个聚类中心的位置u(i) 。这些参数能够最小化代价函数。

4、多次随机初始化的场景选择

随机初始化的错误选择，将会导致代价函数陷入局部最优而非全局最优，因此，随机初始化需要选择一套固定的方法来避免掉入局部最优。



上图的三个图很好的解释了什么是局部最优，什么是全局最优。

解决这个问题的方法就是：

做多次的随机初始化，然后进行聚类，选择最好的聚类结果。多次的随机初始化，可能选择的结果是50-1000次的范围。用每次随机初始化后最终的聚类结果的代价函数做选择，选择代价函数最小的作为最终的聚类结果。

```
a = 100
for i in a:
 # 随机初始化聚类中心100次
 # 计算每一次的聚类结果的代价函数J1，J2，…… ，J100
 # 比较J的最小值
 # 选择最小的代价函数对应的聚类算法的参数作为最终的算法参数
```

特别注意：当K值选择很大的时候，没必须随机初始化100次以上，有可能10次以内就会收敛一个相对很稳定的结果。再随机初始化也不会有多大变化。

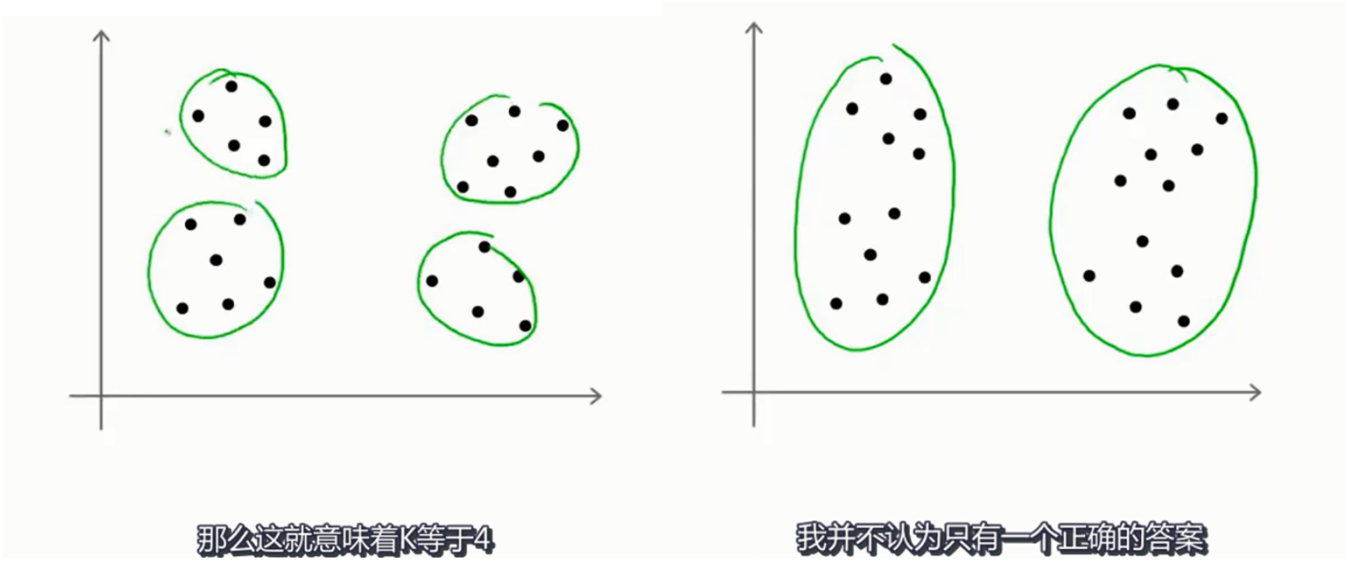
当聚类个数相对较小，比如在2-10之间的时候，那么大量的随机初始化的效果才会非常明显。

5、选择聚类数量

在K均值过程中，选择聚类数量，实际上是一个和业务具体环境高度相关的问题。观察可视化的聚类过程和聚类算法的输出，是两个常用的办法。

比如如下图，到底是选择四个聚类还是两个聚类，看起来都是对的：

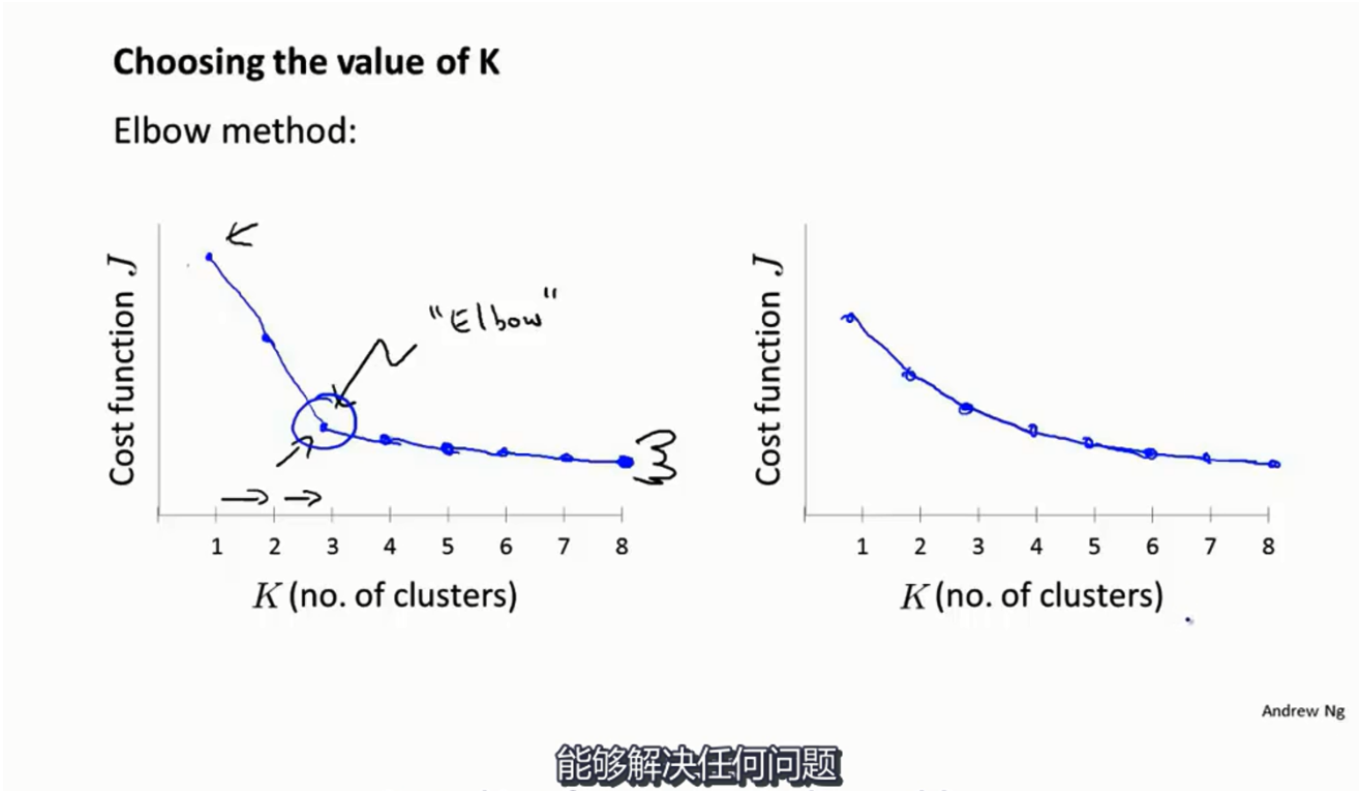
K均值算法在聚类个数的选择上不明确



5.1 肘部法则选择聚类数量

‘肘部法则’ 是一个相对比较好的可视化的观察聚类效果的方法：

这个法则实质上是通过可视化K值的不同选择和对应的代价函数（或者是其他聚类评价指标如互信息、轮廓系数等）的变化情况来判断一个陡降趋于平缓的曲线位置对应的K值位置：

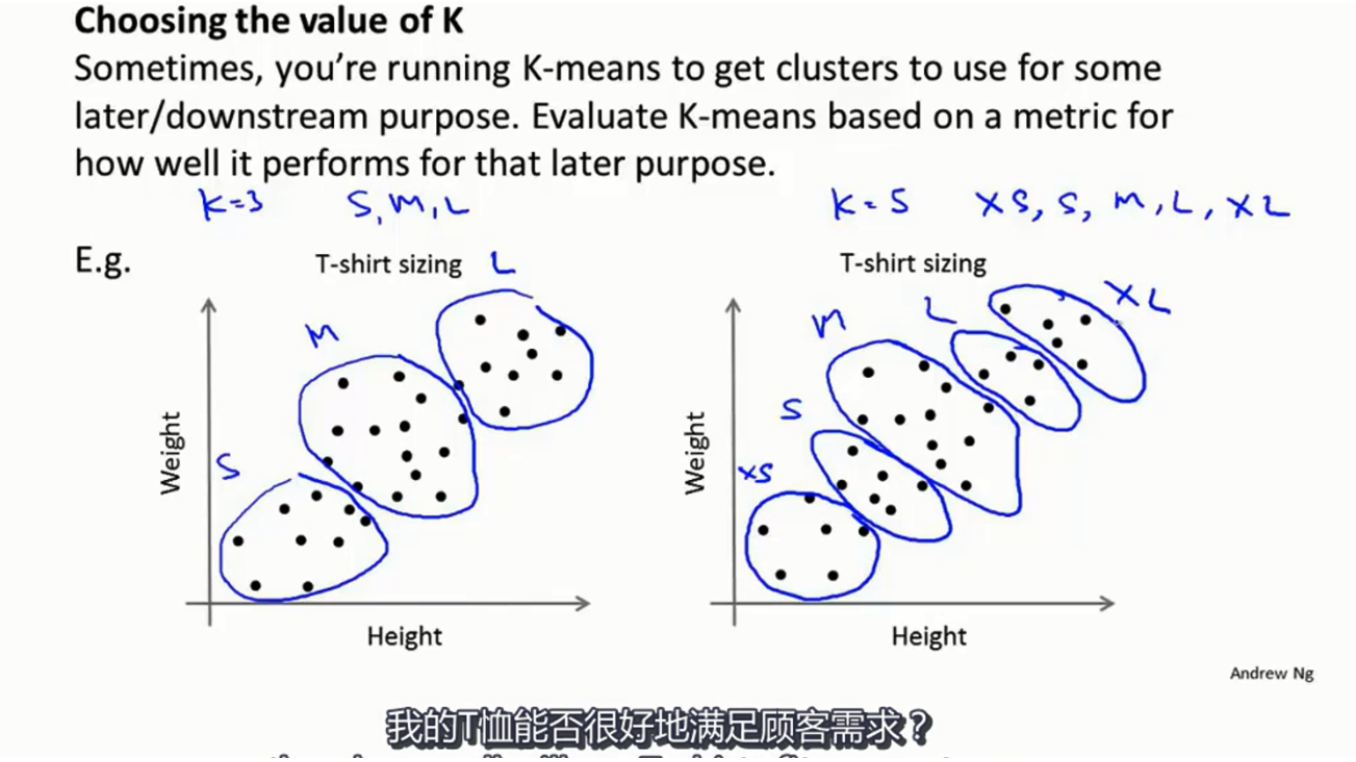


如上图左图，K=3就是类似于手的肘部位置，这个位置以后的代价函数变化趋于平缓的下降，因此建议选择聚类个数为3。这个也称为这条曲线的拐点。

但是，如果曲线作图以后的状态是右边这样的趋势，并不能找到一个好的肘部，整个变化趋势是平缓的，并没有拐点出现。这就是肘部法则的缺点，这个办法不能帮助我们解决所有问题。因此是一种可选的尝试方案。

5.2 基于业务场景的聚类数量选择

以制作T恤为案例，采集了一批不同身高、体重等身体指标的用户数据做无监督聚类算法，试图评估到底应该聚类多少个聚类中心作为不同尺寸的T恤的型号标准。



- 左图中，目标是为了制造三类的T恤型号，S,M,L即可。那么聚类数量可以设置为3。
- 右图中，目标是为了制造三类的T恤型号，XS,S,M,L,XL即可。那么聚类数量可以设置为5。

尺码多了，生成的配套设施和流程增多，库存型号增多，业务成本上升，是否足以支撑？这些都是可以思考的一个角度。