问题 1：kmeans 算法流程中：输入 n_clusters = k, Data = $\left\{x^{(1)}, x^{(2)}, ..., x^{(m)}\right\}$

step1. Initialize **cluster centroids** $\mu_1, \mu_2, ..., \mu_k$ randomly （or use kmean++ method）;

step2. Repeat until convergence: {

For every i， set

$$c^{(i)} := \arg\min_j \left\| x^{(i)} - \mu_j \right\|$$

For each j, set

$$\mu_j := \frac{\sum_{i=1}^{m} 1\left\{c^{(i)} = j\right\} x^{(i)}}{\sum_{i=1}^{m} 1\left\{c^{(j)} = j\right\}}$$

}

当样本量大的时候，每次迭代需要花费大量的时间计算样本点与 k 个中心的距离。由于在计算 $x^{(i)}$与μ_1， μ_2， . . . , μ_k的距离时，$x^{(i+1)}$可以同时进行相同的计算。因此，可以把 step2 中黄色的部分分给多台电脑处理。

思路：

1、 将原始数据随机分为几部分；

2、 每台电脑负责处理一部分数据的计算：

在每台电脑上，利用本机的局部数据初始化 **k 个中心**，每台电脑用本地的局部数据计算出一个新的 **centroids** 的 vector；

向其他的电脑广播新的 centroids，接收其他电脑广播的新的 centroids，对 S 个新的 centroids 取均值，得到新的 centroids：$\mu_j^{(\text{new})} = \underset{s \in S}{\text{mean}}(\mu_j^{(\text{s\_new})})$， s 表示第 s 台电脑，

$\mu_j^{(\text{s_new})}$是第 s 台电脑更新得到的 μ_j。

repeat until converge:

a、 将更新 centroids 作为初值(此时每台电脑的 centroids 初值都是一样的)，在本地执行 step2，得到新的 centroids.

b、 向其他的电脑广播新的 centroids，接收其他电脑广播的新的 centroids，对 S 个新的 centroids 取均值，得到新的 centroids。

My-kmeans 代码中：

1、 初始化用的是 kmeans++的方法，func 为 _ini_cent(X, n_clusters, n_local_trials =None)；
中间需要计算距离，包括矩阵的 Euclidean norm，row_norms，矩阵间的向量的距离等。

2、 my_kmeans 中寻找最近的质心用的是简单版，需要计算样本点与每个质心的距离。

问题 2：

可以考虑用做一个分类问题：对于每个商家 i，预测过去三个月所有跟该商家有互动（包括浏览，收藏，推荐，购买等）的顾客，在未来一个月会去是否会去该商家购物。

1、 数据处理：

A. priority_data(user_info, shop_info): 由前三个月的数据组成；

(1) 提取所有与商家 *i* 有互动的顾客集合 Customer_set(i)；
(2) 构造一些统计特征：
a、用户特征，如该用户平均每周购买几件商品，用户平均每周总的浏览(收藏行为)次数，用户的购买/浏览率，该用户过去三个月购买某品牌(某种类)的商品的数目，该用户过去三个月一共在多少家商家有购买行为，等等。
b、商家特征，如该商家过去三个月所有的购买条数，该商家过去三个月(一个月，一周)被浏览(收藏、推荐)的次数，该商家的购买/浏览率，该商家的主要品牌（商品种类），该商家的评分，等等。
c、用户-商家特征，用户历史在该商家的购物件数，该用户平均每月在该商家购物的数目，该用户过去三个月（一个月）浏览该商家的次数，等等。

B. Training set: 第四个月的数据。
(1) 属性：user_id, shop_id, label；
每个 user_id, shop_id 都是唯一，label 为 0-1.
对于每个 shop_id，对于所有 A 中（1）里的 顾客集合中的 user_id，如果在第四个月在该商家购物，label 为 1，否则为 0.

2、 由于这种问题很可能是不均衡分类，因此考虑将目标定为最小化平均 F1（对于每个 shop_id 都计算 F1）

3、 可以选用 xgboost 这类模型进行拟合，预测。

4、 如果数据量足够多的话，可以将 training set 分成三份，分别作为 training/validation/test，可以 7/21/1 划分，也可以 85/10/5 的比例划分。

5、 调参的话，可以用贪心的策略选出一个局部最优，先确定一个，接着再确定下一个；确定每个参数时，用 gridsearchCV 的方式进行。

6、 预测时，从概率转为 0-1 需要设置一个阈值，预测出概率后，可以用 maximum F1 expectation 进行确定每个商家对应的阈值。对每个 shop_id 对应为 1 的 user_id，生成一个 users 的 set.

7、 各商家可以对对应的 users set 推送一些活动消息。

其他问题：
a、 技术类的书籍: 主要是基础知识的书籍，如统计学习方法，机器学习，应用优化技术；
针对性的问题，一般是找一些 paper。
查阅相应的文档，https://stackoverflow.com/
http://www.csdn.net/，github 上寻找一些参考。