K-means算法实现及参数讨论

本次作业针对聚类算法中最经典的K-means聚类算法展开，将K-means聚类用于MNIST数据集，并对K-means算法中如何选择k值和如何初始化等问题进行了解决分析。

# K-means算法描述

K-means聚类算法是经典的数据挖掘算法之一，是一种属于划分方法的聚类算法，它通过计算对象间不同特征值距离来进行分类。其基本思想是：随机选取数据集中的 k个点作为初始聚类中心，根据数据集中的各个样本到k个中心的距离将其归到距离最小的类中，然后计算所有归到各个类中的样本的平均值，更新每个类中心，直到平方误差准则函数稳定在最小值。具体的数学推到过程如下。

K-means聚类算法把n个向量()分成k个簇，每一簇中所有的样本形成一组，使得非相似性指标的目标函数达到最小。非相似性指标有多种，这里以欧式距离作为非相似性指标进行推导，后续实验中会比较不同非相似指标的优缺点。当欧式距离作为非相似指标时，目标函数如下式：

.

其中代表第k个簇，表示第k个簇内的聚类中心。划分过的簇可以用一个k×n的二维隶属矩阵来定义，如果第i个数据点属于第j个簇，那么U中的元素为1，否则为0。一旦确定聚类中心，可以推出使目标函数达到最小的。

对每一个，如果：，那么，否则。

聚类中心是每一簇中所有向量的均值，如此反复迭代计算，直到目标函数小于某个阈值或者聚类中心不再发生变化，这样就完成了K-means算法的所有步骤，整体流程如下所示：

1、给出k个初始聚类中心；

2、repeat：

      把每一个数据对象重新分配到k个聚类中心处，形成k个簇；

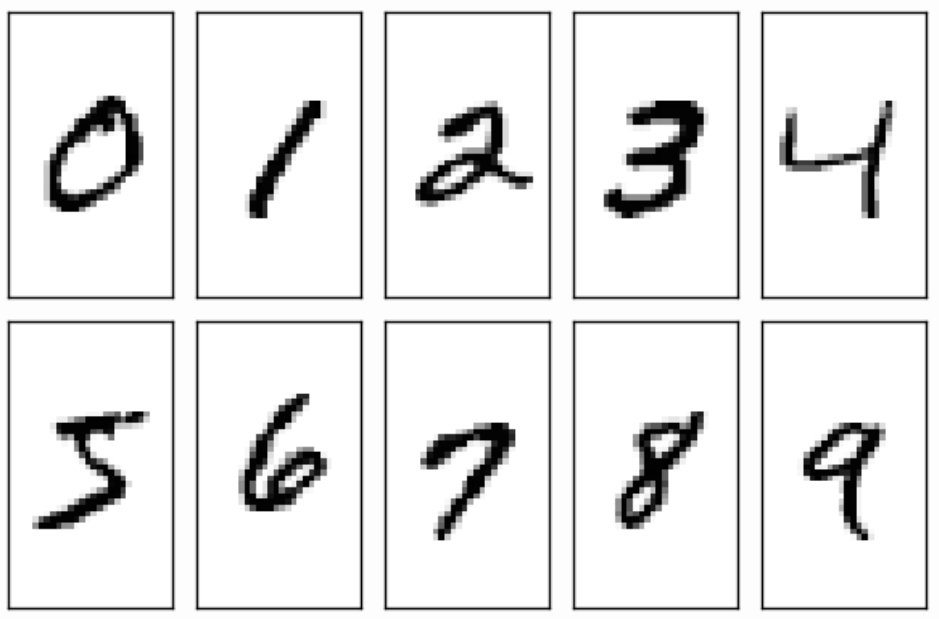
      重新计算每一个簇的聚类中心；

3、until  聚类中心不在发生变化。

下面介绍在MNIST数据集上进行K-means算法的实战。

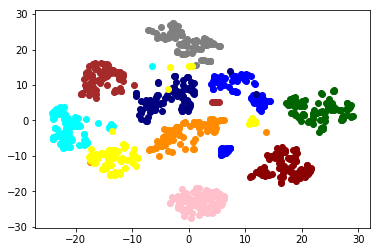
# 数据集描述及预处理

Mnist数据集是机器学习领域经典的数据集，MNIST 数据集来自美国国家标准与技术研究所，训练集由来自 250 个不同人手写的数字构成，其50%是高中学生，50%来自人口普查局的工作人员。测试集也是同样比例的手写数字数据，数据形式如下图所示：



**图1** **MNIST数据格式**

为了简化计算，我们在训练集中随机选取1000个数据用于聚类，所有数据的标签总共有10个，分别代表数字0到9。为了便于可视化和聚类分析，将这些高维的数据利用PCA降维到二维，并将其保存为pickle文件，这是K-means聚类算法所用的数据集，最终数据集的维度为1000×2，数据集的散点分布图如图2所示：



**图2 散点分布图**

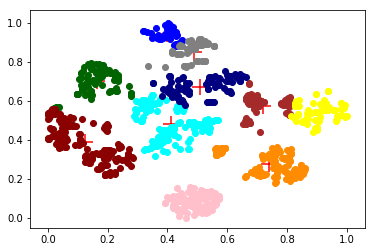
通过数据可视化可以很直观的观察出数据的分布，可以看出所有的数据大致分为10个簇。数据集中不存在缺省值，但由于不同特征之间的数据的取值范围不同，数据之间不具有可比性，在求数据点之间的相似性指标（如欧式距离）时，不同的特征对相似性指标的影响不同，会出现较大的偏差。为了解决此问题，我们利用数值归一化将所有的数值转化为0到1之间，公式如下：



其中Min为样本数据的最小值，Max为样本数据的最大值。

# 实验分析

将K-means聚类算法用于用于MNIST数据集，在迭代17次后，聚类中心不再改变，用时1.25s，聚类完成后的散点分布图如图3所示，可以看出将数据聚类成了10个簇，总体与原数据分类情况相同。



**图2 聚类后散点分布图**

K-means算法原理简单，容易实现，但需要事先确定一个k值，这个k值的选定是非常难以估计的。很多时候，事先并不知道给定的数据集应该分成多少个类别才最合适。另外再k-means聚类时首先需要根据初始聚类中心来确定一个初始划分，然后对初始划分进行优化。这个初始聚类中心的选择对聚类结果有较大的影响，一旦初始值选择的不好，可能无法得到有效的聚类结果，并且会陷入局部最优解。下面针对上述两个问题用不同的方法解决。

# 聚类个数k的确定

一般情况下可以有多种方法确定聚类个数k，例如按需选择法、观察法、手肘法以及Gap Statistic方法等。

按需选择法简单来说就是按照建模的需求和目的来选择聚类的个数。例如在上述实验中，我们已经知道要聚类的个数为10。按需选择法虽然合理，但是未必能保证在做K-means时能够得到清晰的分界线，并且很多时候并不事先知道需要分类的个数。

观察法就是将数据可视化，然后用肉眼去观察这些数据点大概分成几堆。这个方法有个缺点，那就是数据维度不能太高，一般是二维或者三维，否则数据无法可视化出来。

手肘法本质上也是一种间接观察的方法，手肘法的核心指标为误差平方和(SSE, sum of the square errors)，公式如下：



其中代表第i个簇，p是中的样本点，是的质心（即中所有样本的均值），k代表簇的个数。SSE是所有样本的聚类误差，代表了聚类效果的好坏。随着聚类系数k的逐渐增大，样本划分会更加精细，每一个簇的聚合程度会逐渐增加，那么误差平方和SSE 会逐渐减小。当k小于真实聚类数时，由于k的增大会大幅增加每个簇的聚合程度，故SSE的下降幅度会很大，而当k到达真实聚类数时，再增加k所得到的聚合程度回报会迅速变小，所以SSE的下降幅度会骤减，然后随着k值的继续增大而趋于平缓，也就是说SSE和k的关系图是一个手肘的形状，而这个肘部对应的k值就是数据的真实聚类数。

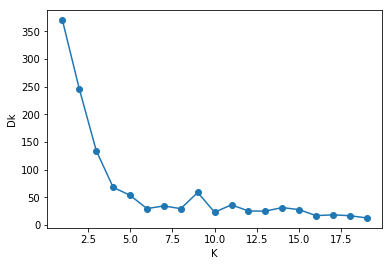
Gap Statistic方法相较于手肘法来说不需要肉眼的判断，而只需要找出使Gap Statistic最大的k值即可，因此，Gap Statistic方法适用于“批量化作业”。Gap Statistic 定义为：



其中指的是的期望，这个数值通常通过蒙特卡洛模拟产生，我们在样本所在的矩形区域中（高维的话就是立方体区域）按照均匀分布随机地产生和原始样本数一样多的随机样本，并对这个随机样本做K-means聚类，从而得到一个。如此往复多次，通常为20次，可以得到20个。对这20个数值求平均值，就得到了的近似值，最终可以计算Gap Statisitc，而Gap statistic取得最大值所对应的k就是最佳的k。

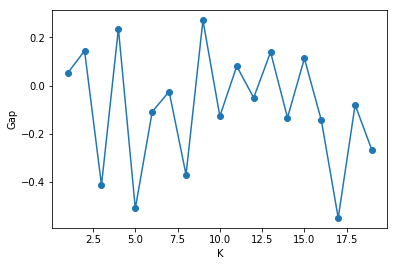
本文主要实现手肘法和Gap Statisitc方法。将k的取值范围设置为0到20之间，经过 K-means 算法后我们会得到不同的中心点和数据点所属的簇，从而得到不同的度量。将聚类个数k作为横坐标，作为纵坐标我们可以得到如图3所示的图像。从图像中我们可以观察到，k=9到k=10下降很快，k=10之后趋于平稳，因此 k=10 是一个拐点，手肘法认为这个拐点就是最佳的聚类个数k。

手肘法是一个经验方法，观察结果因人而异，特别是遇到拐点位置不是很明显时，例如在图3中，k=6也可以认为是一个不明显的拐点。相比于前面的观察法，该方法的优点在于其适用于高维度的数据。



**图3 聚类误差折线图**

下面利用Gap Statisitc方法实现对聚类数目k的选择，样本的所在区域选为所有数据的最小值和所有数据的最大值之间，然后在这个区域内均匀分布随机地产生和原始样本数一样多的随机样本，并对这个随机样本做K-means 聚类，重复20次可以近似计算出 ，的图像如图4所示，从图中可以看出当k=9时，取最大值，即9为最佳聚类簇数，这也和实际的聚类数10接近。



**图4 Gap(k)折线图**

# 初始聚类中心的确定

上述实验中K-means聚类算法的初始聚类中心是随机生成的，这样可能会造成聚类准确率低且结果不稳定，甚至可能陷入局部最优。针对K-means聚类算法对初始聚类中心敏感的特点，尝试用遗传算法去选择合适的初始聚类中心。

遗传算法（Genetic Algorithm, GA）是一种模拟自然界进化、自然选择法则的随机、并行的搜索算法。一个完整的遗传算法主要包括几个步骤：基因编码，种群初始化，选择操作，交叉操作，变异操作，结束条件判断等。针对选择K-means聚类算法的初始聚类中心的问题，下面详细介绍遗传算法的设计过程。

（1）编码方式

在使用遗传算法解决任何问题前，对该问题的所有可行解都应该以计算机能够处理的某种方式进行编码。针对该问题，我们采用浮点数编码的方式，该方式是用某一个范围内的浮点数来表示染色体中的基因值。初始聚类中心的个数应为10，每一个聚类中心的维度为2，我们在数据样本空间中随机生成维度为2的10个初始聚类中心作为种群中的一个个体。

（2）初始种群

遗传算法的初始种群是问题的所有可行解的集合，它是算法的起点。多个已编码的个体构成种群。种群的规模根据具体的问题来确定，其大小对遗传算法的效果具有较大影响，既要考虑到算法的时间复杂度，也要保证种群足够大，以确保个体的多样性。种群由若干个遗传个体组成，每一个遗传个体就是一个可能的解。在此实验中，我们选取种群的大小为40，初始种群的个体通过随机化生成。

（3）适应度函数

适应度函数是一种特殊类型的目标函数，它用来度量一个设计的解决方案接近目标方案的程度，即评价染色体“好坏”的度量标准。适应度函数对染色体进行定量的评价，并对每个染色体返回一个适应度值。根据本实验的特点，适应度函数为所有数据点到种群中个体距离和的最小值，当适应度函数越小，说明初始聚类中心划分的效果越好，该个体有更大的可能性进入下一代。

（4）选择操作

选择算子对种群中的个体进行基于适应度的优胜劣汰的操作，若个体性质优良，则将能被选中；否则，将被淘汰。本实验中采用轮盘赌的选择策略，即被选择的概率正比于该个体适用度值，适应度越高的个体，有更大的机会将被选择。这可以用一个轮盘游戏来表示，每个个体都有一块轮片，适应度高的轮片较大，反之，轮片较小。轮片越大，被选中的概率越高。

（5）交叉操作

交叉的目的是为了在下一代产生新的个体，是遗传算法获得新优良个体的最重要的手段。一旦适应度较高的个体被选择了，这些个体必须能发生随机的变化以提高它们下一代的个体适应度。这里我们采用单点交叉的方式，即两个父代个体以一定的概率（交叉概率）随机选择一个位置作为交叉点，交换交叉点之后的基因段，生成两个新的个体，最后比较父代和子代4个个体的适应度大小，保留适应度最高的两个个体进入下一代。

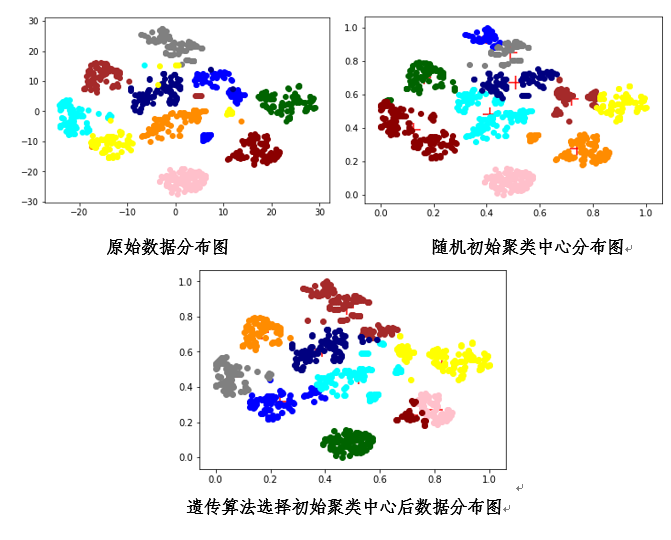
（6）变异操作

变异是使一个种群能够保持多样性的操作，变异操作能较小程度地改变个体的信息，它对局部搜索是非常重要的。本实验的变异概率为0.0001，当种群中的个体被选中变异时，我们随机选取种群其他个体的基因值来组成一个新的个体，然后比较变异个体和原个体的适应度大小，变异个体适应度高将替代原个体，否则变异无效。

（7）停止准则

停止准则是指定特征选择在什么条件下能够结束，当评价准则没有显著的改善时，需要停止准则来避免对搜索空间耗时的穷尽、完全搜索。本实验中可以采用以下的停止准则：预先设定一个最大的迭代次数，当达到最大迭代次数时停止迭代。这时选取末代种群中适应度最大的个体，就可以得到K-means的初始聚类中心。

利用遗传算法得到的初始聚类中心进行K-means聚类，在迭代15次后，聚类中心不再改变，用时1.01s，相对而言迭代次数减少。用遗传算法得到的初始聚类中心与最终的聚类中心更近。对比分析聚类后的散点分布图，如图5所示，可以明显的看出用遗传算法选择初始聚类中心后再进行聚类的效果要更好，与原有的数据簇的分布更加接近。对比实验表明，用遗传算法选择初始聚类中心，能够反映出数据的实际分布，聚类结果稳定，迭代次数更少。



**图5 对比分析图，**