一、K-means

1.1 简单介绍一下K-means算法的原理和工作流程

K-means算法以k为参数，把n个对象分为k个簇，使得簇内具有较高的相似度，而簇间的相似度较低。

处理过程的简述：

1. 随机地选择k个对象，每个对象初始地代表了一个簇的平均值或中心；
2. 对剩余每个对象，根据其与各簇中心的距离，将它赋给最近的簇；
3. 重新计算每个簇的平均值。
4. 不断重复上述过程，直到准则函数收敛。

具体步骤：

1. 从数据D中随机选取k个元素，作为k个簇各自的中心。
2. 分别计算剩下的元素到k个簇中心的相异度，将这些元素分别划归到相异度最低的簇。
3. 根据聚类结果，重新计算k个簇各自的中心，计算方法是取簇中所有元素各自维度的算术平均数。
4. 将数据D中全部元素按照新的中心重新聚类，重复上述步骤，直到聚类结果不再变化。

1.2 K-means中常用的中心距离的度量有哪些？

1. 欧几里得距离 $$d = \sum_{i=1}^N\sqrt{(x_{1i}-x_{2i})^2}$$
2. 余弦相似度 $$\cos(\theta)=\frac{a\cdot b}{||a||\times ||b||}$$

1.3 K-means中的k值如何选取？

1. 场景选定法(屁话)
2. 随机法。随机n次，得到最优结果k，避免局部最优解。
3. **手肘法**。 核心指标为SSE误差平方和：$SSE=\sum_{i=1}^k\sum_{p\in{C_i}}|p-m_i|^2$
   核心思想：随着聚类数k的增大，样本划分会更加精细，每个簇的聚合程度会逐渐提高，那么误差平方和SSE自然会逐渐变小。变小分为2个过程：
   1. 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。
   2. 当k达到真实聚类数时，再增加k所得到的聚合程度回报会迅速变小，所以SSE下降幅度会骤减。
   3. 而真实聚类数就是SSE的转折点。
4. 轮廓系数法。使用轮廓系数评估分类质量，选择分类质量最好的k。
5. 稳定性方法
6. 与层次聚类结合
7. Canopy Method

1.4 K-means算法中的初始点的选择对最终结果有影响吗？

有影响，K-means选择的初始点不同获得的最终分类结果也可能不同，随机选择的重心会导致K-Means陷入局部最优解。

1.5 K-means聚类中的每个类别中心的初始点应如何选择？

1. 随机法

2. 选择个批次距离尽可能元的k个点。
3. 层次聚类或者Canopy预处理

1.6 K-means中空聚类的处理

如果所有的点在指派步骤都未分配到某个簇，就会得到空簇。

1. 选择一个距离当前任何质心最远的点。
2. 从具有最大SSE的簇中选择一个替补的质心。

1.7 如何快速收敛数据量超大的K-means？

Mini Batch KMeans：

1. 从数据集中随机抽取一些数据形成一个Mini Batch，将他们分配给最近的质心。
2. 更新质心。对于每一个小批量，通过计算平均值得到更新质心，并把小批量里的数据分配给该质心。

1.8 K-means的优缺点？

优点：这是一个凸优化函数，一定能收敛。 缺点：对噪声和孤立点敏感。

1.9 怎么评估K-means聚类效果？

轮廓系数：类的密集与分散程度的评价指标。越大越好。

1. 对于第i个对象，计算它到所属簇中所有其他对象的平均距离，记为凝聚度：a_i
2. 对于第i个对象和不包含该对象的任何簇，计算该对象到给定簇中所有对象的平均距离，记为分离度：b_i
3. 因此第i个对象的轮廓系数为 $$s_i = (b_i-a_i)/max(a_i, b_i)$$

二、KNN

2.1 简单介绍一下KNN算法的原理

KNN算法，即k最近邻分类算法。所谓k最近邻，就是最接近的k个邻居，即每个样本都可以由k个邻居表示。

核心思想：
在一个含未知样本的空间，可以根据离这个样本最邻近的k个样本的数据类型来确定样本的数据类型。该算法的三个主要因素：**分类决策规则，距离和相似度的衡量，k的大小**

分类决策规则：分类时为：多数表决法，即训练集里和预测的样本特征最近的k个样本，预测为里面有最多类别数的类别。回归时为：选择平均法，即最近的k个样本的样本输出的平均值作为回归预测值。
距离的度量：最常用的是欧式距离。Pearson距离，余弦相似度等都是可以考虑的。 **k值得选择：**过小容易过拟合，过大容易欠拟合。因此使用交叉验证法。

2.2 KNN的优缺点

优点:

1. 既可以分类也可以回归

2. 可用于非线性分类
3. 训练时间复杂度为O(n)
4. 准确度高。对异常值不敏感。

缺点:

1. 计算量大
2. 样本不平衡问题
3. 需要大量内存

2.3 如果样本不平衡对KNN会造成什么样的影响？怎么解决？

KNN在做分类时，最大的问题在于分类决策规则的设计：多数表决法。如果样本不平衡，有一个类的样本容量很大，而其他类的样本数量很小，很可能会导致当输入一个未知样本时，该样本的K个邻居中大数据量类的样本占多数。这样会导致分类错误。
解决办法：采用权值的方法来改进。距离近的样本权重大，距离远的样本权重小。

2.4 如何解决KNN算法计算量过大的问题？

可以使用**分组快速搜索近邻法**的方式进行计算。
基本思想：将样本集按近邻关系分解为组，给出每组质心位置，以质心作为代表点，和未知样本计算距离，选出距离最近的一个或若干个组，再在组的范围内应用一般的KNN算法。

2.5 什么是欧式距离和曼哈顿距离？

1. 欧氏距离： $$d = \sum_{i=1}^N\sqrt{(x_{1i}-x_{2i})^2}$$
2. 曼哈顿距离： $$\sum_p||l_1^p-l_2^p||$$

2.6 KNN中的K如何选取的？

K值一般取一个比较小的数值，例如采用交叉验证法(简单来说，就是一部分样本做训练集，一部分做测试集。)来选择最优的K值。

2.7 KNN和K-means有什么区别？

1. KNN是分类算法，K-means是聚类算法
2. KNN是监督学习，K-means是非监督学习
3. KNN和K-means的K的含义不同。

三、KD树