当前实验思路总结

1. 前提假设

我们假设正常企业中，员工满意度分布符合正态分布，即绝大部分用户的满意度位于一个N-(均值μ，标准差σ）的[μ-2σ, μ+2σ]的区间范围内，合计概率约0.9544。

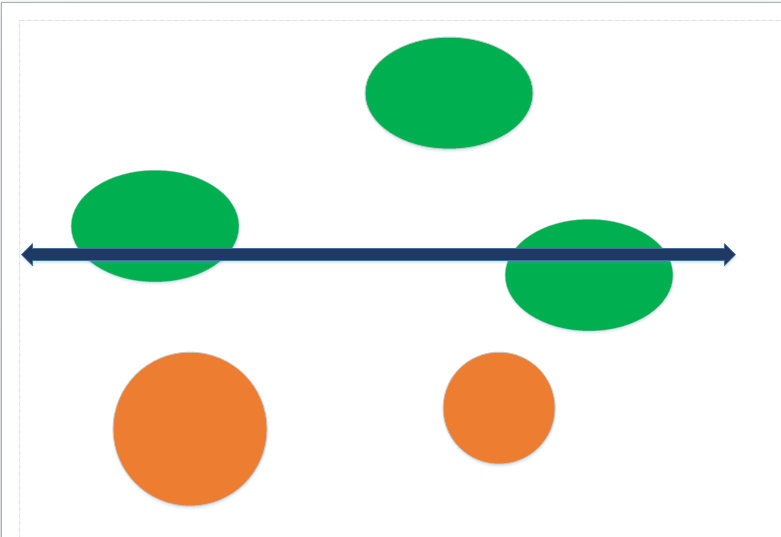
故假定对于落入过高与过低区间的用户的小概率事件的概率合计为5%，则满意度过低的用户群体概率为2.5%，故至少应选择全部用户满意度最低的前2.5%，作为高危用户。

1. 实验思路

思路1：基于提取的JS特征直接计算用户的满意度分数（Job Satisfaction Scores, JSS）,然后根据分数从低到高排序，查看前2.5%的用户（约100个用户）中包含多少个Insiders，即计算Recall，如果能够发现CERT6.2的全部5个恶意用户全部在前100个用户中，则验证了我们的假设（也可以用CERT5.2的数据再次计算其JSS，看98个模拟的Insiders有多少落在了前100个用户中）

思路2：基于KMeans与OCSVM的组合，其主要步骤为：

1. 提取用户的JS特征；
2. 对CERT6.2的4000个用户直接进行KMeans聚类，以发现满意度相近的用户群。（根据轮廓系数衡量不同K值的凝聚度与离散度，从而在{2,10]中确定最优的K值）
3. 根据所得用户群的中心的特征数值，计算相应的JS分数，并根据JS分数将用户群从高到低排序，选择中位数（减小极端值对均值的影响）以上的用户群，添加到OCSVM训练集；
4. 对于上步中剩下的用户群，选取其中JS分数高于全体用户JS分数均值的用户，补充到OCSVM训练集，剩余的用户群用户组成OCSVM测试集；
5. 根据得到的OCSVM训练集训练OCSVM分类器；
6. 使用OCSVM分类器分类测试集，得到内部高危用户集合，并计算这些高危用户的JS分数，排序；作为输出等待异常检测结果进行融合；



思路3：基于OCSVM分类器与JS分数顾虑相结合，其主要步骤为：

1. 提取用户的JS特征；
2. 基于CERT5.2数据集中的用户数据，2000个用户（98个Insiders），构造训练集与测试集（测试集中包含98个Insiders与200个正常用户），训练OCSVM，优先确保Recall，在此基础上考虑优化误报率FP；
3. 将训练好的OCSVM直接用于CERT6.2的4000个用户，得到标签[-1]的用户集合；
4. 针对上述用户集合，计算其JS分数，从低到高排序，保留前100个用户作为内部高危用户（2.5%的高危率）
5. 候选JSS计算公式：



