优达学城数据分析师纳米学位项目 P5

安然提交开放式问题

1. 向我们总结此项目的目标以及机器学习对于实现此目标有何帮助。作为答案的部分，提供一些数据集背景信息以及这些信息如何用于回答项目问题。你在获得数据时它们是否包含任何异常值，你是如何进行处理的？【相关标准项：“数据探索”，“异常值调查”】

■ **项目背景：**

2001 年美国能源公司安然公司被曝出企业欺诈丑闻并最终破产，一度成为美国历史上最大的破产案。安然的高管团队为谋取利益，通过许多高风险、不道德的商业行为和会计手段，推高公司股价，回避审计监管，从而获取丰厚的工资和奖金。

此项目旨在通过对已知嫌疑人（person of interest, POI）的安然邮件数据和财务数据进行机器学习，提取关键特征，从而基于这些特征判断安然案中其他嫌疑人。

■ **数据集：**

- **数据集中共包含 146 个数据点，其中 POI 18 个,占比约 0.12。**看得出来，这是一个分布非常**不平衡（unbalanced）的数据**，这在之后的评估指标和交叉验证上都需要注意。
- 评估指标：**precision 和 recall 比 accuracy 更好一些。**因为不平衡的数据中，正确判断出非 POI 的情况是较高的，这会拉高 accuracy，而我们要知道的是算法正确判断出 POI 的能力。
- 交叉验证分配训练集（train）和测试集（test）：**选择 Stratified Shuffle Split 比直接默认的 Split 方式更合适。**分层抽样可以更好地确保，每次测试集和训练集的

比例都接近数据集中 POI 比，即 0.12 。这对不平衡的数据特别重要，不然某一个训练集中可能只有一两个 POI，而训练集中有十几个。

- 除此以外，因为数据样本比较少，因此我们可以使用 GridSearchCV 来进行参数调整，如果较大的数据则会花费较长的时间，可以考虑使用 RandomizedSearchCV.

■ 异常值：

通过对两个关键的财务数据特征工资（salary）和 奖金（bonus）进行可视化，发现数据中有一个特别明显的异常值，对应到财务数据源，发现这是总值（total），并不是指向任何 POI。删除之后再次检验，余下有一些疑似异常值，都是安然公司的高管，因此予以保留。

除此以外在数据源中发现还有一个非自然人数据点：THE TRAVEL AGENCY IN THE PARK，和一个全部为 NaN 的数据点 LOCKHART EUGENE E 一并删除掉了。

2. 你最终在你的 POI 标识符中使用了什么特征，你使用了什么筛选过程来挑选它们？你是否需要进行任何缩放？为什么？作为任务的一部分，你应该尝试设计自己的特征，而非使用数据集中现成的——解释你尝试创建的特征及其基本原理。（你不一定要在最后的分析中使用它，而只设计并测试它）。在你的特征选择步骤，如果你使用了算法（如决策树），请也给出所使用特征的特征重要性；如果你使用了自动特征选择函数（如 SelectBest），请报告特征得分及你所选的参数值的原因。【相关标准项：“创建新特征”、“适当缩放特征”、“智能选择功能”】

■ 选择特征：

'salary','bonus','total_payments',

'exercised_stock_options','restricted_stock','total_stock_value',

'from_poi_to_this_person','from_this_person_to_poi',

'to_messages','from_messages',"ratio_from_poi","ratio_to_poi"

首先，检查数据集所有变量的缺失值，在缺失值相对较少的变量中，从财务数据中选了六个（现金收入和股票收入各三个），邮件数据中选取四个特征（收发邮件数量，和与 poi 之间直接进行手法的邮件数量），并根据收发邮件中 poi 的比例，添加了两个新特征。

然后，对于不同的算法我选择了不同的特征处理方式。

朴素贝叶斯：特征缩放，特征缩放+KBest，特征缩放+PCA。我希望通过这三种处理哪种降维的方式表现更好，最后的结果是特征缩放+ Kbest 表现最佳。

SVC: 特征缩放，特征缩放+KBest，特征缩放+PCA。 SVC 的表现非常不好，尤其是在正确判断 poi 这一点上，召回率都是 0，所以很快决定放弃这一算法。

决策树： 按特征重要性从高到低排序，从财务中选了 2 个，邮件数据中选了 3 个特征重要性高且和 poi 相关性搞得特征。

3. 你最终使用了什么算法？你还尝试了其他什么算法？不同算法之间的模型性能有何差异？【相关标准项：“选择算法”】

最终算法： 朴素贝叶斯

其他尝试：SVC 和决策树。

朴素贝叶斯 :Accuracy: 0.85533 ,Precision: 0.44444 ,Recall: 0.34000， F1: 0.38527

决策树： Accuracy: 0.78808， Precision: 0.32168， Recall: 0.34050， F1: 0.33082

SVC 判断 POI 的 precision 和 recall 在几次特征预处理之后一直是 0 ，所以没有进一步调参。

通过几个指标（尤其是 precision, recall）可以看出，经过特征处理和调参之后的朴素贝叶斯在正确判断 POI 的能力上比决策树更优秀。

4. 调整算法的参数是什么意思 ,如果你不这样做会发生什么？你是如何调整特定算法的参数的？（一些算法没有需要调整的参数 – 如果你选择的算法是这种情况，指明并简要解释对于你最终未选择的模型或需要参数调整的不同模型，例如决策树分类器，你会怎么做 ）。【相关标准项：“调整算法”】

调参：通过对算法中的一些参数进行调整，以获取更好的表现。

我对朴素贝叶斯和决策树都进行了调参，朴素贝叶斯通过对 Kbest 的 K 值进行不同尝试，而决策树对 clf 的 min_samples_split 进行不同尝试。通过 GridSearch 爬格子的方式，找到变现最好的参数。SVC 虽然没有使用，但如果需要可以对其 C 和 gamma 进行调参。

5. 什么是验证，未正确执行情况下的典型错误是什么？你是如何验证你的分析的？【相关标准项：“验证策略”】

验证可以尽可能的确保所得到的结果不是因为随机性，或因为调参对训练集过度拟合。

未正确验证的典型错误是过拟合，可能这个分类器对训练集解释力度很好，但对一个新的数据就表现糟糕。

我将验证放在调参的过程中进行，在 GridSearchCV 中将 cv 改为 StratifiedKFold(10)。因为安然数据中 poi 和 non-poi 比例非常失衡，总体 poi 的比例非常小。一开始将 GridSearchCV 设为默认 cv 时，虽然在 poi_id.py 中表现不错，但是在 tester.py 中表现很糟糕，原因在于没有分层抽样，而分层抽样之后稳健性有了很大的提高。

6. 给出至少 2 个评估度量并说明每个的平均性能。解释对用简单的语言表明算法性能的度量的解读。【相关标准项：“评估度量的使用”】

Precision: 0.44444　　Recall: 0.34000　F1: 0.38527

Precision 查准率: 指的是所有预测的 poi 中，正确出 poi 的比例。打个比方，有 100 杯咖啡，30 杯豆奶咖啡，70 杯普通咖啡。我猜其中 20 杯咖啡是豆奶咖啡，而这 20 杯中真正的豆奶咖啡只有 5 杯，那查准率就是 5/20=0.4.

Recall 召回率：在所有真正的 poi 中，被正确预测出的 poi 比例是多少。还是刚刚那个咖啡栗子，查准率是 5/30=0.6

F1: 准确率和召回率调和之后的平均值。