# 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

说明：你可以在这里下载此文档的英文版本。

机器学习的一个重要部分就是明确你的分析过程，并有效地传达给他人。下面的问题将帮助我们理解你的决策过程及为你的项目提供反馈。请回答每个问题；每个问题的答案长度应为大概 1 到 2 段文字。如果你发现自己的答案过长，请看看是否可加以精简！

当评估员审查你的回答时，他或她将使用特定标准项清单来评估你的答案。下面是该标准的链接：评估准则。每个问题有一或多个关联的特定标准项，因此在提交答案前，请先查阅标准的相应部分。如果你的回答未满足所有标准点的期望，你将需要修改和重新提交项目。确保你的回答有足够的详细信息，使评估员能够理解你在进行数据分析时采取的每个步骤和思考过程。

提交回答后，你的导师将查看并对你的一个或多个答案提出几个更有针对性的后续问题。

我们期待看到你的项目成果！

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”】

   此项目的目标即从安然公司的员工数据中识别出其中的嫌疑人，包括被起诉的，已经定罪的和通过供认其他人而获得免罪的员工或者相关人士。机器学习主要起到了特征处理如特征降维，特征缩放等，利用算法进行分类和验证的作用。

   数据集包含 144 个数据点，从 feature 角度，在查看 Insider pay 时发现 loan advances, director fees 和 restricted stock deferred 缺失值较多，因此首先排除这些特征；从 labels 角度，有 18 个 POI，126 个非 POI，在项目中的后续影响如下：

   由于数据集样本点较少，因此可以在参数调整时使用 grid search；
   由此看见数据集不平衡，因此在选择算法评估指标时 accuracy 并不是很好的评估指标，选择 precision 和 recall 或者结合了两者的 f1 和 f2 更好一些；
   由于数据的不平衡，在交叉验证时，使用 stratified shuffle split 的来对数据集进行划分。

   在可视化 salary 和 bonus 特征时，发现一个明显的异常点，这个点是财务表的 total，不能作为个人财务特征使用，因此删除。

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”】

依靠背景调查信息，初步使用以下特征：

'poi','salary','bonus','long_term_incentive','deferred_income','deferral_payments','total_payments','exercised_stock_options','restricted_stock','restricted_stock_deferred','total_stock_value','fraction_from_poi','fraction_to_poi'

最终使用的特征：首先利用 PCA.explained_variance_ratio_，可以得出，当 n_component=2 时，explained_variance 已经达到 95%以上，因此可选择 n=2；另外，利用 selectKBest 获得各项特征的重要性得分如下：

```
array([ 18.57570327,  21.06000171,  10.07245453,  11.59554766,
         0.21705893,   8.86672154,  25.09754153,   9.34670079,
         0.06498431,  24.46765405,   3.21076192,  16.64170707])
```

选择 10 分以上的，初步确定 k=7；最后利用 feature Union 将两者结合，作为最终特征。

缩放特征对于 logistic regression 是有必要的，因为算法会受到数据集范围的影响，缩放特征后算法能起到更好的效果。

设计特征：从邮件信息来说，自行设计的特征包括 fraction_to_poi 和 fraction_from_poi，即从邮件往来数量上来说，个人发送到嫌疑人的邮件占总发送邮件的比例和个人收到的嫌疑人邮件占总接受邮件的比例。

选择特征：上述已阐明进行 pca analysis 时为什么 n=2；而在 selectKBest 中选择 k=7 是因为将 score 的阈值手动设置为 10，但是在后续 hyper parameter tuning 时，算法自动计算得出，对于朴素贝叶斯当 k=9 时效果最好，对于决策树 k=1 时效果最好。

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”】

最终使用了朴素贝叶斯算法，还尝试了逻辑回归，决策树，随机森林和简单的算法融合。

算法性能比较：

| | Accuracy | Precision | Recall | F1 | F2 |
|---|---|---|---|---|---|
| GaussianNB | 0.848 | 0.410 | 0.314 | 0.356 | 0.329 |
| LogisticRegression | 0.855 | 0.401 | 0.172 | 0.241 | 0.195 |
| DecisionTree | 0.805 | 0.277 | 0.288 | 0.282 | 0.285 |
| RandomForest | 0.853 | 0.373 | 0.147 | 0.211 | 0.167 |
| VotingClassifier | 0.849 | 0.371 | 0.194 | 0.255 | 0.214 |
| GaussianNB_best | 0.855 | 0.450 | 0.385 | 0.415 | 0.396 |
| DecisionTree_best | 0.850 | 0.413 | 0.290 | 0.340 | 0.308 |

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”】

   如果不进行参数的调整，算法会在默认参数下进行计算，而默认参数往往不是最好的组合，因此需要进行调整。本项目中主要利用 grid search 进行算法参数的调整，即设定参数范围，让算法在所有可能的参数组合中进行计算，效果最好的参数组合将被作为最后使用的参数。

   本项目中首先利用默认参数进行计算，然后选出得分最高的两种算法，即朴素贝叶斯和决策树分类器，在 pipeline 中进行 grid search，因为在朴素贝叶斯中可调参数较少，因此 pipeline 中调整的主要是特征选择也就是 selectKBest 的参数，并得到最佳的 k=9；在决策树的 pipeline 中调整的参数包括 selectKBest, max_depth 和 min_samples_split，并得出最佳组合参数为 1,4,9.

   由上表可知，经过参数调整之后，算法的五项得分均有所提高。

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”】

   验证就是再次评估算法在未知数据集上的性能。在没有正确执行验证时，典型错误时在训练集上性能好，在测试集上性能差，即算法的泛化能力差。

   因为数据集的不平衡，使用 stratified shuffle split 方法对数据集进行划分。 Stratified shuffle split 是 stratified k fold 和 shuffle split 的结合，将数据集划分成 k 份之后，保持各个数据集与原始数据集中各类比例相同，

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”】

   主要度量指标为 F1 和 F2，推导过程如下：

| | | PREDICTED | |
|---|---|---|---|
| | | P | N |
| ACTUAL | P | TRUE POSITIVE_tp | FALSE NEGATIVE_fn |
| | N | FALSE POSITIVE_fp | TRUE NEGATIVE_tn |

Total = tp+fp+fn+tn
Accuracy = (tp+tn)/total 所有正确分类的数量占总数量的比例
Precision = tp/(tp+fp) 即所有预测为正的数量中预测对的比例
Recall = tp/(tp+fn) 即实际为正的数量中预测对的比例
F1 = 2*Precision*Recall/(Precision+Recall)
F2 = (1+2.0*2.0) * Precision*Recall/(4*Precision + Recall)

F1 和 F2 结合了 Precision 和 Recall，更好的平衡这两个度量指标

以此项目中效果最好的朴素贝叶斯算法为例，预测出来的 POI 有 1708，实际 POI 有 769.

| | | PREDICTED | |
|---|---|---|---|
| | | P | N |
| ACTUAL | P | 769 | 1231 |
| | N | 939 | 12061 |

优达学城
2016 年 9 月