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”】

- 对数据集进行了探索，其中总共包括了 146 个相关的字典项，其中包含了 20 个特征项目，和一个 POI 特征项目。
- 发现在 146 个字典项中包含了 18 个 poi 嫌疑人和 128 个非 poi 嫌疑人
- 逐个打印特征项目的缺失值 NaN 结果如下下表所示

特征名称	NaN 数量
salary	51
deferral_payments	107
total_payments	21
loan_advances	142
bonus	64
restricted_stock_deferred	128
deferred_income	97
total_stock_value	20
expenses	51
exercised_stock_options	44
other	53
long_term_incentive	80
restricted_stock	36
director_fees	129
to_messages	60

from_poi_to_this_person	60
from_messages	60
from_this_person_to_poi	60
shared_receipt_with_poi	60

从上表看出，其中财务特征中 deferral_payments、loan_advances、restricted_stock_deferred、deferred_income 、director_fees 都出现了缺省值 NaN 过多，在考虑特征选择的时候可以先把这些特征排除。

- 剔除上述特征后，逐个打印所有剩下的特征项目，发现其中的问题其中的 key="TOTAL" 的字典项目中存在明显财务指标过大，从而删除这个字典项。Key="THE TRAVEL AGENCY IN THE PARK" 并不是一个人的名字，从而删除这个字典项。Key="LOCKHART EUGENE E"所有的特征项都是 NaN，对这个字典条目进行删除。

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"】

- 由于财务特征和邮件特征不在一个数量级上面，对以下所有的特征 'salary', 'total_payments', 'bonus', 'total_stock_value', 'expenses', 'exercised_stock_options', 'other', 'long_term_incentive', 'restricted_stock', 'to_messages', 'from_poi_to_this_person', 'from_messages', 'from_this_person_to_poi', 'shared_receipt_with_poi' 进行了特征缩放，把所有特征都缩放到 0—1 之间的数量。
- 添加了一个新的变量 total_messages,在没有添加变量之前通过打印得到指标变化如下表所示：（并未经过测试集）

指标名称	添加 total_message 之前	添加 total_messages 之后
Accuracy	0.79	0.81060
precision	0.375	0.26337
recall	0.23	0.22700

通过添加 total_messages 之后，看出添加 total_messages 之后再 precision、recall 指标上有提升，而 accuracy 指标有下降。

现在在全部的特征中使用特征自动选择函数 SelectBest 在 15 个特征中选择。随着选择特征参数 K 值的变化，打印除三个指标在测试集中变化情况如下表所示：

指标名称	K=1	K=2	K=3	K=4	K=5	K=6	K=7	K=8	K=9	K=10	K=11	K=12	K=13	K=14	K=15
Accuracy	0.79593	0.79673	0.82333	0.80613	0.80673	0.79820	0.79773	0.80020	0.81900	0.82167	0.80547	0.80327	0.81593	0.24750	0.81120
Precision	0.23672	0.22496	0.35334	0.27854	0.27998	0.24261	0.23268	0.23777	0.30581	0.31668	0.25027	0.24801	**0.28270**	0.26934	0.26524
recall	0.23850	0.21450	0.39150	0.28550	0.28600	0.24200	0.22500	0.22600	0.28150	0.29150	0.23000	0.23400	0.24750	0.23150	0.23500
sum	1.27115	1.23619	1.56817	1.37017	1.37271	1.28281	1.25541	1.26397	1.40631	1.42985	1.28574	1.28528	1.34613	0.74834	1.31144

从上表可以看出在 k=3 的时候各项指标的总和最高，我们选择 k=3 的时候指标合最大，

通过 selectBestK 函数，我们选择的三个特征为

bouns	total_stock_value	exercised_stock_options

我们使用决策时分类器，在默认参数的情况下，得到的分类器评分 Accuracy: 0.82273 Precision: 0.34989 Recall: 0.38400

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”】

   最终使用决策树算法，在决策树算法中，测试性能参数为 Accuracy: 0.82273 Precision: 0.34989 Recall: 0.38400

   尝试使用朴素贝叶斯算法，对上述的特征进行了分析，测试性能参数为 Accuracy: 0.85327 Precision: 0.43619 Recall: 0.34350

   不同算法的模型性能有差别：两个算法的差别，在同样选择上述三个参数的情况下，分类器的评分指标如上面所示，在数据中，尽可能多的识别出 poi 至关重要，而朴素贝叶斯算法虽然在准确率、精确度上面优于决策树算法，但是在查全率上面低于决策树算法。

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”】

   模型调参主要是为了寻找算法中的最优参数，提高模型的性能。

   如果不这样做可能会发生模型过拟合或者欠拟合的情况。

   在决策树算法中，我们使用了 GridSearchCV（）函数选择最佳的决策树参数，我们调整了其中的三个参数，如下表所示：

criterion':('gini','entropy'),	'min_samples_split':[2,10],	'presort':(1,0)

   打印最优秀参数{'min_samples_split': 2, 'presort': 1, 'criterion': 'gini'}

   把最有参数代入到分类器中,得到了最佳的决策树分类器，可以对决策树模型进行调整

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”】

   验证是把原数据集拆分成训练数据和测试数据，通过这些数据来测试分类器的性能，使得分类器可以独立与数据集的存在，并且可以作为过拟合的检查方式。

   在没有正确执行的情况下出现的典型错误包括：分类器性能达不到预想，分类器过拟合，分类器欠拟合。

   train_test_split 交叉验证原理将总的数据集进行了拆分，train_test_split(features, labels, test_size=0.3, random_state=42) 其中 30%的数据为测试数据，70%为训练数据。

   KFold 交叉验证：kf=KFold(len(features),4)，我们把数据集分成 K=4 个大小不同的容器，并进行 4 次交叉验证，通过这种方式，4 次交叉验证之后，全部数据集都成为了数据集和测试集。

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”】

   这里使用了三个参数对结果进行评估，一个是准确率，一个是精确度，还有一个是召回率，我们得到的使用选定特征，使用决策树算法得到：

   准确率 accuracy:0.82:这个表明了在这个分类器中,一个新特征进入该分类器后，正确的分出这个人是否 poi 的概率。

   精确度：percision: 0.34989 表明了在测试数据集中，该分类器识别出一个人为嫌疑人，有 0.32 的概率这个人为真正的嫌疑人

   查全率：recall=0.38400 表明该分类器在测试集中，能够识别全部嫌疑人数据中的 0.3

参考文献：

http://scikit-learn.org/stable/modules/generated/sklearn.tree.DecisionTreeClassifier.html

http://scikit-learn.org/stable/modules/generated/sklearn.model_selection.GridSearchCV.html