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?

答

安然公司（Enron Corporation），曾是一家位于美国的德克萨斯州休斯敦市的能源类公司。2000年披露的营业额达1010亿美元之巨。公司连续六年被《财富》杂志评选为“美国最具创新精神公司”，然而真正使安然公司在全世界声名大噪的，却是这个拥有上千亿资产的公司2002年在几周内破产，持续多年精心策划、乃至制度化、系统化的财务造假丑闻。

本项目的目标是根据公开的安然财务和电子邮件数据集，构建一个预测性算法，找出有欺诈嫌疑的安然员工。

本数据集中，最初包含有146条记录，其中包含了雇员的财务和邮件数据，数据中包含20个特征。根据程序输出的结果，发现共有18条记录可能是嫌疑人。

数据中的特征分为三大类，即财务特征、邮件特征和 POI 标签。

财务特征 : [‘salary’, ‘deferral\_payments’, ‘total\_payments’, ‘loan\_advances’, ‘bonus’, ‘restricted\_stock\_deferred’, ‘deferred\_income’, ‘total\_stock\_value’, ‘expenses’, ‘exercised\_stock\_options’, ‘other’, ‘long\_term\_incentive’, ‘restricted\_stock’, ‘director\_fees’] (单位均是美元）

邮件特征 : [‘to\_messages’, ‘email\_address’, ‘from\_poi\_to\_this\_person’, ‘from\_messages’, ‘from\_this\_person\_to\_poi’, ‘shared\_receipt\_with\_poi’]

POI 标签 : [‘poi’], boolean类型

通过数据可视化以及数据集打印和处理，我发现数据中存有NaN值和异常值。

名为Total的数据远远大于其他数据点，根据其名称，我们可以得知此值为合计值，在随后的数据分析中，我们将其移除。

此外还有LOCKHART EUGENE E这条数据，20个特征全为NaN值，我判断其为无效数据，故予以删除。

最后还有THE TRAVEL AGENCY IN THE PARK，此条数据，根据数据名称我们可以发现，这实际为一个旅行社的名称，与我们分析目标无关，故予以删除。

在对以上数据进行清理后，我们还剩余143条数据。

1. 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”】

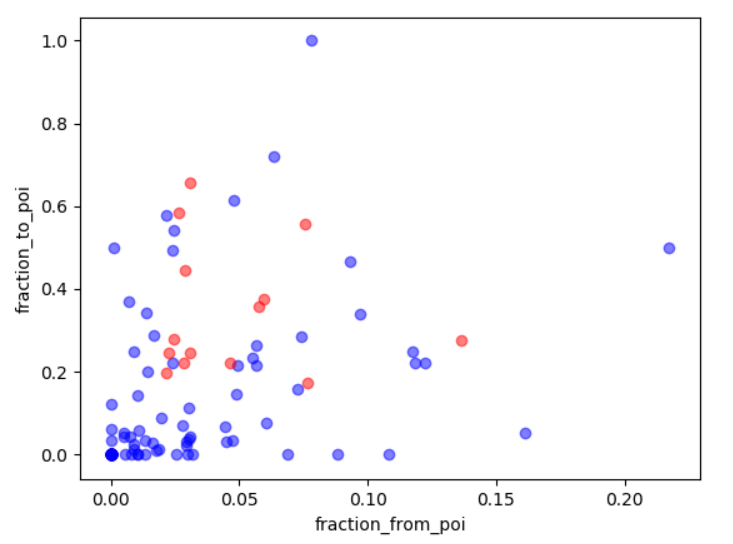
创建新的特征：

在本数据集中，from\_poi\_to\_this\_person和from\_this\_person\_to\_poi是该雇员接收到嫌疑人员和发送给嫌疑人员的邮件数目。

由于每个人处理邮件的数目不同，将数据转换为该人员发送和接收自嫌疑人员与自身总发送接收邮件数目的比例更有说服力。

fraction\_from\_poi = from\_poi\_to\_this\_person / to\_messages

fraction\_to\_poi = from\_this\_person\_to\_poi / from\_messages



根据上图， 我们可以发现，当两个特征值都小于某个值后，此人为POI的概率就几乎为0。这将为我们未来的算法提供便利。此外从SelectKBest的特征值得分我们也可以看出，fraction\_to\_poi是对创建模型起到一定作用的具有强特征的数据。通过下表可以看出，在与其他特征值进行组合后，提高了算法的性能。

特征选择：

本数据集中，一共有20个特征，加上刚刚创建的两个新特征共22个特征。在这些特征中，并不是每一个特征都对创建预测模型有用，在这里使用 Scikit-Learn中的SelectKBest算法，评判得分，选取最好的6个feature。

具体得分如下：

[('exercised\_stock\_options', 24.815079733218194), ('total\_stock\_value', 24.18289867856688), ('bonus', 20.792252047181535), ('salary', 18.289684043404513), ('fraction\_to\_poi', 16.4097125480358), ('deferred\_income', 11.458476579280369)]

随后我对选择特征对算法性能的影响进行了测试，结果如下：

|  |  |  |  |
| --- | --- | --- | --- |
| Features\_Selected | Accuracy | Precision | Recall |
| exercised\_stock\_options | 0.90409 | 0.46055 | 0.321 |
| exercised\_stock\_options', 'total\_stock\_value' | 0.84069 | 0.46889 | 0.2675 |
| exercised\_stock\_options', 'total\_stock\_value', 'bonus' | 0.843 | 0.48581 | 0.351 |
| exercised\_stock\_options', 'total\_stock\_value', 'bonus', 'salary' | 0.84677 | 0.50312 | 0.323 |
| exercised\_stock\_options', 'total\_stock\_value', 'bonus', 'salary', 'fraction\_to\_poi' | 0.84879 | 0.45558 | 0.3 |
| exercised\_stock\_options', 'total\_stock\_value', 'bonus', 'salary', 'fraction\_to\_poi', 'deferred\_income' | 0.85236 | 0.47723 | 0.351 |

由此表可以看出当特征为6个时候，三个性能指标达到了最优平衡。

特征缩放：

上述6个特征中，有5个是该雇员的收入，而最后一个是发送给POI雇员的邮件占比。6个特征的值相差巨大。故采用StandardScaler进行特征缩放，将值缩放到[0,1]之间。决策树和线性回归不受特征缩放的影响。决策树的分割是一系列的水平线和垂直线，不存在两者的交换。在考虑一个维度时，不需要考虑另一个维度的值。

1. 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”】

在本项目中，我共使用了三种算法，分别为：

朴素贝叶斯

决策树

随机森林

首先我观测了三种算法的运算时间：

朴素贝叶斯：0.0170001983643 s

决策树：1.802 s

随机森林：32.44 s

我们可以看出 三种算法的运算时间差距达到了百倍。

此外 三种算法的准确率如下：

朴素贝叶斯：0.857142857143

决策树：0.908163265306

随机森林：0.897959183673

但是根据SKlearn的文档，我们可以得知，Accuracy并不是一个很好的评判算法性能的参数，综合考虑precision, recall和F1score 是更为合理的选择。各个算法的数据如下：

|  |  |  |  |
| --- | --- | --- | --- |
|  | Precision | Recall | F1 Score |
| 朴素贝叶斯 | 0.47723 | 0.351 | 0.40449 |
| 决策树 | 0.48649 | 0.162 | 0.24306 |
| 随机森林 | 0.35695 | 0.2695 | 0.30712 |

根据上表我们可以看出，朴素贝叶斯的性能最好，提供最高的precision, recall和F1 score.

因此，我选择了朴素贝叶斯为最终的算法。

1. 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”】

调参是机器学习中非常重要的一部分，不同的算法函数和初始设定会对最终结果产生影响。为算法选择了错误的参数，可能会造成模型过拟合，准确率大大降低，算法性能下降等后果。

调参的最终目的是要使训练之后的模型检测物体更精确，向程序的方向更靠近一步的话，就是使得损失函数（例如SSD中的loss）尽量小。

本数据集数据样本比较少，故使用GridSearchCV来进行参数调整。在我的代码中，使用了GridSearchCV()来调试各个算法的参数。如果较大的数据则会花费较长的时间，可以考虑使用RandomizedSearchCV。另外通过test\_classifier()测试了算法并给出了判断结果。根据结果，可以发现经过调试最好的结果来自朴素贝叶斯分类器。

1. 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”】

验证是将训练出得模型，用测试数据进行评价的过程。验证中的典型错误是没有将数据分成训练和测试两部分，从而导致用训练的数据进行验证，得到虚高的准确率，模型过拟合，无法进行泛化使用。

我们都知道，当用于模型训练的数据量越大时，训练出来的模型通常效果会越好。所以训练集和测试集的划分意味着我们无法充分利用我们手头已有的数据，所以得到的模型效果也会受到一定的影响。 在本项目中，我使用了Cross-Validation方法，也就是交叉验证。通过对数据进行多次洗牌和分割，能够确保训练集和测试集中POI与非POI的比例，比较适合于该数据。最终验证结果如下：

朴素贝叶斯：

Accuracy: 0.85236 Precision: 0.47723 Recall: 0.35100 F1: 0.40449 F2: 0.37061

Total predictions: 14000 True positives: 702 False positives: 769 False negatives: 1298 True negatives: 11231

决策树：

Accuracy: 0.85586 Precision: 0.48649 Recall: 0.16200 F1: 0.24306 F2: 0.18694

Total predictions: 14000 True positives: 324 False positives: 342 False negatives: 1676 True negatives: 11658

随机森林：

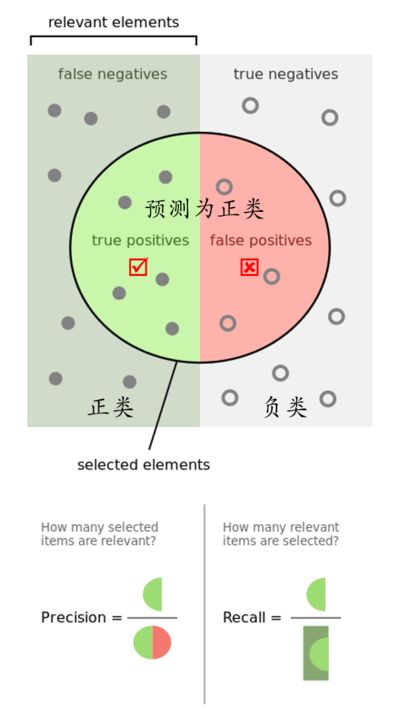
Accuracy: 0.82629 Precision: 0.35695 Recall: 0.26950 F1: 0.30712 F2: 0.28339

Total predictions: 14000 True positives: 539 False positives: 971 False negatives: 1461 True negatives: 11029

1. 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”】

基于准确率Precision,召回率Recall和综合评价指标F1等进行的判断。我最终选取的算法为朴素贝叶斯。

下图很好的解释了上个问题中，我们提到的True positives, False positives, False negatives, True negatives四个值以及recall和precision的意义。



关于精确率Precision:在本项目中，精确率指的是模型预测出的POI中，真正为POI的比率。

关于召回率Recall：在本项目中，指的是所有真正的POI雇员中，有多少被真正的识别出来了。

F1分数（F1 Score），是统计学中用来衡量二分类模型精确度的一种指标。它同时兼顾了分类模型的准确率和召回率。F1分数可以看作是模型准确率和召回率的一种加权平均，它的最大值是1，最小值是0。可见F1综合了P和R的结果，当F1较高时则能说明该模型效果不错，数据也显示朴素贝叶斯的F1为其中的最高值，为0.40449。

参考文献：

1. <https://baike.baidu.com/item/F1%E5%88%86%E6%95%B0/13864979?fr=aladdin>
2. <https://www.zhihu.com/question/19645541>
3. http://scikit-learn.org/stable/