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

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”】

这个项目的目的就是学生运用一些机器学习的知识来识别有欺诈嫌疑的安然雇员，在整个过程中，我首先是观察了数据，清理数据，转换特征，实现算法，在整个过程中有许多小细节让我做的不顺畅，例如查看 nan，找出异常值，数据的预处理，算法的选择以及实施。每一步都走的挺艰辛的，在观察数据这一部分可能是因为过于细节，而自己的能力不够，对于算法这部分更主要是自己不熟练，思维有些乱，整理思路以及实施耗时较长。但是整个项目做下来很明显感觉到自己的能力有进步，对更多细节有更多的了解，对这类型的项目已经有了一个思维框架。

该数据是由 146 行，21 个变量组成的，其中 20 个是特征，从对数据的探索可以发现，每一列每一行都具有缺失值。但是每一列的缺失值与非缺失值的比例是不同的，
通过对每一列每一行的缺失值的数值可以发现，发现 LOCKHART EUGENE E 这一行除了 poi 都是空值，TOTAL 和 THE TRAVEL AGENCY IN THE PARK，从名字就可以发现是异常值，total 是综合的意思，而 THE TRAVEL AGENCY IN THE PARK 不是一个个人的数据，不仅如此，从所做的图中可以看出 total 这个数值异常的大，所以这三行应当删除。

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”】

用了决策树算法，选出了特征。

```
accuracy 0.8
Feature Ranking:
1 feature salary (0.350693926975)
2 feature bonus (0.196967098082)
3 feature exercised_stock_options (0.172469008264)
4 feature deferred_income (0.143181818182)
5 feature from_messages (0.100892693952)
6 feature from_poi_to_this_person (0.0357954545455)
7 feature from_this_person_to_poi (0.0)
8 feature shared_receipt_with_poi (0.0)
9 feature expenses (0.0)
10 feature long_term_incentive (0.0)
11 feature restricted_stock (0.0)
```

即 features_list01 = ['poi','salary', 'bonus', 'exercised_stock_options', 'deferred_income', 'from_messages','expenses', 'long_term_incentive','from_poi_to_this_person', 'from_this_person_to_poi', 'shared_receipt_with_poi']

然而经过机器学习的模型效果不如意，因为 precision 和 recall 都特别低，而且这是一个类别失衡的数据集，所以 accuracy 并不是一个好的评估指标。
最后减少特征，选择了
features_list02=['poi', 'salary', 'bonus', 'exercised_stock_options', 'deferred_income', 'from_poi_to_this_person', 'from_this_person_to_poi', 'shared_receipt_with_poi']

在这里，首先对数据进行标准化，因为在没有标准化的情况下，机器学习算法不能正常工作。由于机器学习算法里运用到了 KNN，因此需要进行特征缩放。特征缩放最主要的目的其实是为了让机器学习算法工作的更好。 k-means, logistic regression, SVM, linear discriminant analysis, PCA 等需要特征缩放。

K 近邻算法中，分类器主要是计算两点之间的欧几里得距离，如果一个特征比其它的特征有更大的范围值，那么距离将会被这个特征值所主导。因此每个特征应该被归一化，比如将取值范围处理为 0 到 1 之间
本项目用的是 MinMaxScaler 方法来实现缩放。

以下用的特征是 features_list02=['poi', 'salary', 'bonus', 'exercised_stock_options', 'deferred_income', 'from_poi_to_this_person', 'from_this_person_to_poi', 'shared_receipt_with_poi']

Clf，方法	Accuracy	Precision	Recall	F1	F2
特征改变后，使用特征缩放 KNeighborsClassifier	0.84507	0.21341	0.05250	0.08427	0.06182

特征改变后，未使用特征缩放 KNeighborsClassifier	0.89129	0.78520	0.32900	0.46371	0.37226
特征未改变，未使用特征缩放，KNeighborsClassifier	0.86621	0.58687	0.21450	0.31417	0.24568

当进行特征缩放时，一定要非常小心，因为特征缩放很有可能会丢失掉有意义的信息。

我创建了一个新特征 coefficient_bonus_salary，这个特征是用 bonus 的值除以 salary 的值，设置的特征的原因是我认为如果要找出欺诈嫌疑人，那么肯定是跟收入的钱有关，而且最直接相关的应该是 salary 和 bonus。

```
for i in range(len(newdataset_scaled['salary'])):
    if newdataset_scaled['salary'][i] > 0:
        dataset['coefficient_bonus_salary'][i] = \
        1.0 * newdataset_scaled['bonus'][i] / newdataset_scaled['salary'][i]
```

My_dataset3(my_dataset+新特征)
['poi', 'salary', 'bonus', 'exercised_stock_options', 'deferred_income', 'from_poi_to_this_person', 'from_this_person_to_poi', 'shared_receipt_with_poi','coefficient_bonus_salary']
结果如下：

Clf	Accuracy	Precision	Recall	F1	F2
AdaBoostClassifier	0.81387	0.28194	0.25600	0.26834	0.26080
RandomForestClassifier	0.86633	0.49196	0.07650	0.13241	0.09205
GaussianNB	0.85547	0.44622	0.34850	0.39135	0.36446
KNeighborsClassifier	0.89793	0.77173	0.33300	0.46525	0.37572
QuadraticDiscriminantAnalysis	0.83527	0.35490	0.28800	0.31797	0.29928

可以看出，还是 KNN 算法效果最好，但是其数值只是比非加入新特征的数值大了一点点而已。

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”】

我尝试了 AdaBoostClassifier，RandomForestClassifier，GaussianNB，neighbors.KNeighborsClassifier，QuadraticDiscriminantAnalysis 这五种，最终取得效果最好的是 KNeighborsClassifier

效果如下：
My_dataset
['poi', 'salary', 'bonus', 'exercised_stock_options', 'deferred_income', 'from_poi_to_this_person', 'from_this_person_to_poi', 'shared_receipt_with_poi']

Clf	Accuracy	Precision	Recall	F1	F2
AdaBoostClassifier	0.81543	0.32473	0.27050	0.29514	0.27985
RandomForestClassifier	0.85464	0.45700	0.09300	0.15455	0.11062
GaussianNB	0.84514	0.44459	0.33700	0.38339	0.35414
KNeighborsClassifier	0.89129	0.78520	0.32900	0.46371	0.37226
QuadraticDiscriminantAnalysis	0.82829	0.36515	0.27350	0.31275	0.28796

不同算法之间的 accuracy 大致在同一个区间内(>0.8)，但是该项目不应该用 accuracy 来衡量结果，然而各个算法的 precision 和 recall 都不一样，相差较大。

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”】

模型可以有很多参数，算法调整的目标是找到这些指标的最佳组合。
调整算法非常重要，因为不同的设置会对其性能产生深远的影响。

第三题已经回答，效果最好的模型是 KNeighborsClassifier，在这里对 KNeighborsClassifier 用 GridSearchCV 进行参数调整。
最优参数为：

```
Fitting 10 folds for each of 240 candidates, totalling 2400 fits
('best params are:', "{'n_neighbors': 3, 'weights': 'uniform', 'leaf_size': 1, 'algorithm': 'auto'}")
```

用该函数调整后：

```
['poi', 'salary', 'bonus', 'exercised_stock_options', 'deferred_income', 'from_poi_to_this_pe
rson', 'from_this_person_to_poi', 'shared_receipt_with_poi']
KNeighborsClassifier(algorithm='auto', leaf_size=1, metric='minkowski',
           metric_params=None, n_jobs=1, n_neighbors=3, p=2,
           weights='uniform')
        Accuracy: 0.87743       Precision: 0.61873      Recall: 0.37000 F1: 0.46308     F2:
0.40235
        Total predictions: 14000        True positives:  740    False positives:  456   False
negatives: 1260 True negatives: 11544
```

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”】

验证是一种用于检查我们的模型如何与数据集的其余部分进行概括的技术。 在这种情况下常见的错误是过度拟合模型在训练集上表现良好，但在测试集上的结果却相当低。

在这个项目中，我使用本项目自带的 test_classifier（）来验证。

参数 folds 的意思是模型每次用不同的测试集来运行 1000 次，用 StratifiedSuffleSplit 来在原始数据集上划分这些测试集。

StratifiedSuffleSplit 函数会首先对数据进行打乱，然后才划分，以免出现过拟合的情况。
其中 n_iter 表示重新洗牌和分裂迭代次数；
test_size 如果是 float 类型的数据, 这个数应该介于 0-1.0 之间，代表 test 集所占比例. 如果是 int 类型, 代表 test 集的数量. 如果为 None, 值将自动设置为 train 集大小的补集；
train_size :如果是 float 类型的数据 应该介于 0 和 1 之间，并表示数据集在 train 集分割中所占的比例 如果是 int 类型, 代表 train 集的样本数量. 如果为 None, 值将自动设置为 test 集大小的补集；
random_state 用于随机抽样的伪随机数发生器状态。

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”】

accuracy=正确预测为 POI 的人数与数据集中所有人的数量的比值
recall=被正确识别出 POI 人的个数与测试集中所有 POI 的个数的比值
precision=在预测为 POI 的样本中，真正是 POI 的样本与总样本所占的比例

在最后的算法中，这些度量值具有以下值：精度=89.129%，查准率=78.520%，查全率=32.9%。

在实践中，这意味着我对我的预测 POI 的准确度为 89.129%，其中在算法那预测出来的 POI 人里面有 78.520%真的是 POI，并且在所有 POI 人中我们算法找出了 32.9%的人是 POI。