Enron POI Detector Project Assignment

--- Project Overview

In 2000, Enron was one of the largest companies in the United States. By 2002, it had collapsed into bankruptcy due to widespread corporate fraud. In the resulting Federal investigation, there was a significant amount of typically confidential information entered into public record, including tens of thousands of emails and detailed financial data for top executives. In this project, you will play detective, and put your new skills to use by building a person of interest identifier based on financial and email data made public as a result of the Enron scandal. To assist you in your detective work, we've combined this data with a hand-generated list of persons of interest in the fraud case, which means individuals who were indicted, reached a settlement, or plea deal with the government, or testified in exchange for prosecution immunity.

--- Resources Needed ---

You should have python and sklearn running on your computer, as well as the starter code (both python scripts and the Enron dataset) that you downloaded as part of the first mini-project. The starter code can be found in the `final_project` directory of the codebase that you downloaded for use with the mini-projects. Some relevant files:

- `poi_id.py` : starter code for the POI identifier, you will write your analysis here
- `final_project_dataset.pkl` : the dataset for the project, more details below
- `tester.py` : when you turn in your analysis for evaluation by your Udacity coach, you will submit the algorithm, dataset and list of features that you use (these are created automatically in poi_id.py). That coach will then use this code to test your result, to make sure we see performance that's similar to what you report. You don't need to do anything with this code, but we provide it for transparency and for your reference.
- `emails_by_address` : this directory contains many text files, each of which contains all the messages to or from a particular email address. It is for your reference, if you want to create more advanced features based on the details of the emails dataset.

--- Steps to Success ---

We will provide you with starter code, that reads in the data, takes your features of choice, then puts them into a numpy array, which is the input form that most sklearn functions assume. Your job is to engineer the features, pick and tune an algorithm, test, and evaluate your identifier. Several of the mini-projects were designed with this final project in mind, so be on the lookout for ways to use the work you've already done.

Your submission will have 2 parts. First, and most importantly, you should document your final project steps in a series of focused questions on the Udacity website. Your responses should be about 1 paragraph for each question; taken together, they will be the way that you report to another person (most notably your Udacity coach) what you tried, and what you eventually ended up using as your final analysis strategy. We advise that you keep notes as you work through the project; your thought process is, in many ways, more important than your final project and we will by trying to probe your thought process in these questions.

Second, you will create three pickle files (`my_dataset.pkl, my_classifier.pkl, my_feature_list.pkl`) and turn those in to your Udacity coach when you submit your writeup; your coach will test them using the `tester.py` script. This step is just to check that you've created useable machine learning code, and to verify the performance and parameters of your algorithm. These pickle files are already set up to be made at the end of `poi_id.py`, you don't need to add any code to create them.

As you are probably already aware, as preprocessing to this project, we've combined the Enron email and financial data into a dictionary, where each key-value pair in the dictionary corresponds to one person. The dictionary key is the person's name, and the value is another dictionary, which contains the names of all the features and their values for that person. The features available to you are the following:

['salary', 'to_messages', 'deferral_payments', 'total_payments', 'loan_advances', 'bonus', 'email_address', 'restricted_stock_deferred', 'deferred_income', 'total_stock_value', 'expenses', 'from_poi_to_this_person', 'exercised_stock_options', 'from_messages', 'other', 'from_this_person_to_poi', 'poi', 'long_term_incentive', 'shared_receipt_with_poi', 'restricted_stock', 'director_fees']

These fall into three major types of features, namely financial features, email features and POI labels.

- financial features: ['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'] (all units are in US dollars)
- email features: ['to_messages', 'email_address', 'from_poi_to_this_person', 'from_messages', 'from_this_person_to_poi', 'poi', 'shared_receipt_with_poi'] (units are generally number of emails messages; notable exception is ‘email_address’, which is a text string)
- POI label: [‘poi’] (boolean, represented as integer)

You are encouraged to make, transform or rescale new features from the starter features. If you do this, you should store the new feature to my_dataset, and if you use the new feature in the final algorithm, you should also add the feature name to my_feature_list, so your coach can access it during testing. For a concrete example of a new feature that you could add to the dataset, refer to the lesson on Feature Selection.

--- What We Provide, and What You Should Do ---

--- Background Info on the Data Sources and Conventions ---

The financial data comes from the enron61702insiderpay.pdf source, which is included for your reference in the tools directory. The email data comes from the Enron email corpus, which we introduced in Lesson 5 on datasets and questions; you should have downloaded and unzipped this dataset as part of the code setup process. The email features in final_project_dataset.pkl are aggregated from the email dataset, and they record the number of messages to or from a given person/email address, as well as the number of messages to or from a known POI email address and the number of messages that have shared receipt with a POI.

To help you navigate the messages on your own, if that’s something you want to do, we've provided the `final_project/emails_by_address` directory, which contains lists of all the messages to and from each email address in the corpus--so for example, if you want to read all the emails from Sara Shackleton, you can find them listed in "from_sara.shackleton@enron.com.txt"

Not all features have values for all people in the dataset. When the value for a particular feature is unknown, "NaN" appears for that person/feature. When the data is transformed into a numpy array, "NaN" is converted to 0 by default.