**Enron Submission Free-Response Questions**

A critical part of machine learning is making sense of your analysis process, and communicating it to others.  The questions below will help us understand your decision-making process and allow us to give feedback on your project.  Please answer each question; your answers should be 1-2 paragraphs per question.  If you find yourself writing much more than that, take a step back and see if you can simplify your response!

When your coach evaluates your responses, he or she will use a specific list of rubric items to assess your answers.  Here is the link to that rubric: [Link to the rubric](https://docs.google.com/a/knowlabs.com/document/d/17-JwNQH1aRxtqMkJ6zpCL_68kh5F6uSbDXcJS26vZWY/pub)

Each question has one or more specific rubric items associated with it, so before you submit an answer, take a look at that the rubric.  If your response does not meet expectations, you will be asked to resubmit.

Once you’ve submitted your responses, your coach will take a look and ask a few more focused follow-up questions on one or more of your answers.

We can’t wait to see what you’ve put together for this project!

1. Summarize for us the goal of this project and how machine learning is useful in trying to accomplish it.  As part of your answer, give some background on the dataset and how it can be used to answer the project question.  Were there any outliers in the data when you got it, and how did you handle those?  [relevant rubric items: “data exploration”, “outlier investigation”]

The goal of the project is to identify person’s of interest (POI) among Enron employees who may have committed fraud or broken the law. To do this, I will build a POI identifier that uses email and financial data released as part of the court case. Machine learning is useful for building a model that can determine whether someone is a POI. I will use Machine Learning techniques to build a classifier, to select the most important features to use for classification, and to test the accuracy of my classifier. Initial data discovery uncovered the following (see ../datasets\_questions/explore\_num\_people.py):

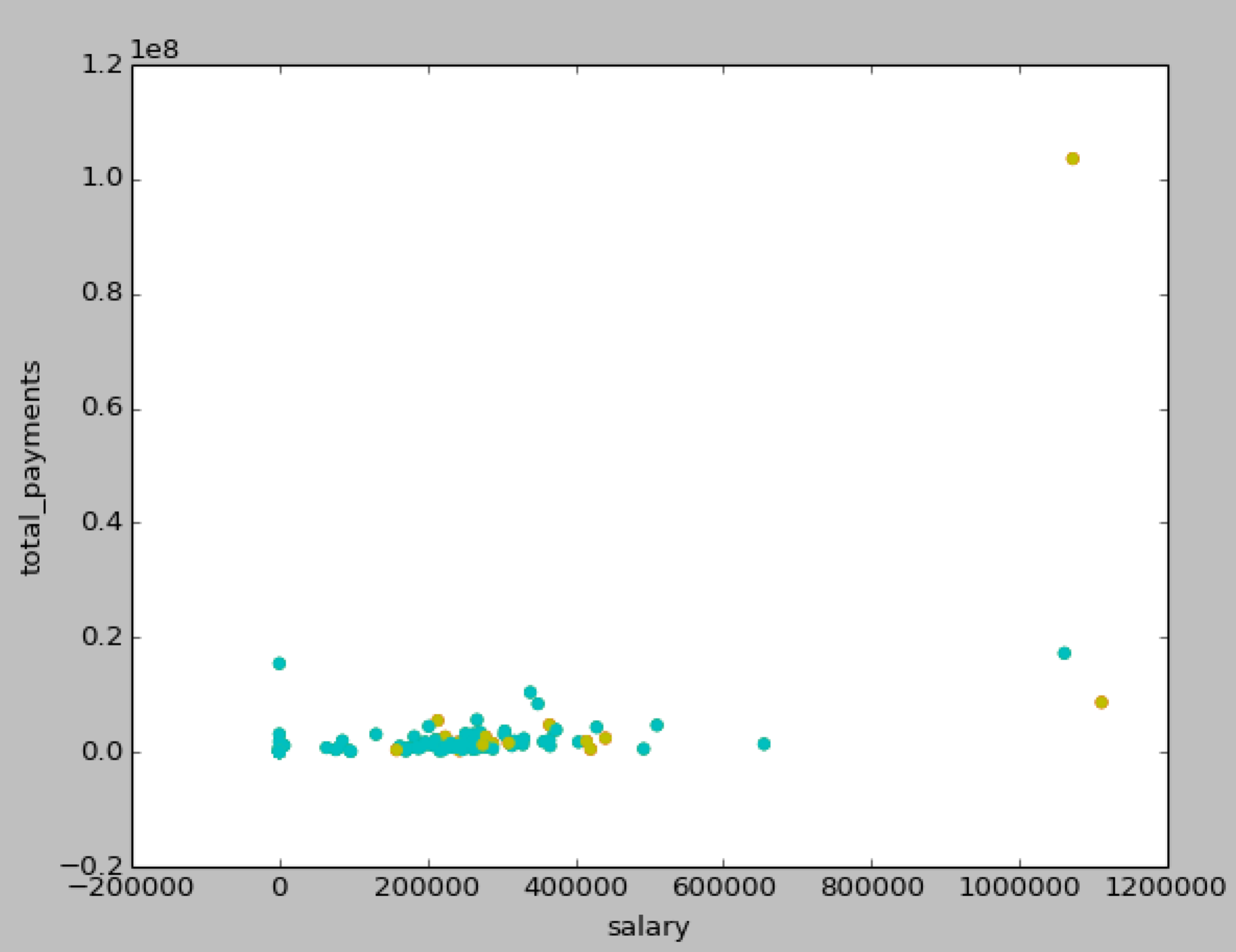
* total no. of data points = 146 people in the data set, POI and non-POI
* allocation across classes (POI/non-POI) = 18 POI, 128 non-POI
* number of features = 21 features are available
* are there features with many missing values? Yes. Features with most number of missing values (where value is NaN) = loan\_advances, restricted\_stock\_deferred, director\_fees. Features with moderate number of missing values (where occurrence of NaN > 73) = deferral\_payments, long\_term\_incentive, deferred\_income

financial features: ['salary', 'total\_payments', , 'bonus', 'total\_stock\_value', 'expenses', 'exercised\_stock\_options', 'other', 'restricted\_stock']

email features: ['to\_messages', 'email\_address', 'from\_poi\_to\_this\_person', 'from\_messages', 'from\_this\_person\_to\_poi', 'poi', 'shared\_receipt\_with\_poi']

POI label: [‘poi’] (boolean, represented as integer)

Outlier found to be POIs except for FREVERT MARK A but did not remove him.



1. What features did you end up using in your POI identifier, and what selection process did you use to pick them?  Did you have to do any scaling?  Why or why not?  As part of the assignment, you should attempt to engineer your own feature that doesn’t come ready-made in the dataset--explain what feature you tried to make, and the rationale behind it.  (You do not necessarily have to use it in the final analysis, only engineer and test it.)  If you used an algorithm like a decision tree, please also give the feature importance of the features that you use.  [relevant rubric items: “create new features”, “properly scale features”, “intelligently select feature”]

From the original feature set, I began by selecting features based on my intuition. I chose ‘salary’, ‘bonus’, and ‘total\_payments’ but the accuracy, precision, and recall were very low using those features. Therefore, I used the SelectKBest algorithm to determine which features were most important as this seemed more systematic. The features 'total\_payments', 'restricted\_stock\_deferred', and 'director\_fees’ were the financial features with the highest scores. I decided on these by taking the top 8 highest scoring features, and running them through the Gaussian Naïve Bayes classifier one-by-one. I omitted the ones that did not increase the accuracy, precision and recall or actually decrease those metrics. The reason was to avoid overfitting and remove features that hurt my classification model. In the end I had the 3 mentioned above.

Features: “salary”

Performance: Accuracy: 0.25560 Precision: 0.18481 Recall: 0.79800 F1: 0.30011 F2: 0.47968 Total predictions: 10000 True positives: 1596 False positives: 7040 False negatives: 404 True negatives: 960

Features: “total\_payments”

Performance: Accuracy: 0.76308 Precision: 0.06938 Recall: 0.04350 F1: 0.05347 F2: 0.04701 Total predictions: 13000 True positives: 87 False positives: 1167 False negatives: 1913 True negatives: 9833

Features: “bonus”

Performance: Accuracy: 0.37800 Precision: 0.22229 Recall: 0.72000 F1: 0.33970 F2: 0.49731 Total predictions: 9000 True positives: 1440 False positives: 5038 False negatives: 560 True negatives: 1962

'poi', 'shared\_receipt\_with\_poi', 'loan\_advances', 'director\_fees','to\_messages', 'from\_poi\_to\_this\_person', 'from\_messages'

pseudo code engineer new feature: take each email and create text vectorizer. Dump vectorizer into pickle as it will take a long time to run. Label which emails are from POI (tuple). Run the vectorizers in k-means to cluster. Create clusters of similar documents. Calculate ratio of POIs in each cluster. Add number of emails from POI in cluster as a feature.

1. What algorithm did you end up using?  What other one(s) did you try? [relevant rubric item: “pick an algorithm”]
2. What does it mean to tune the parameters of an algorithm, and what can happen if you don’t do this well?  How did you tune the parameters of your particular algorithm?  (Some algorithms don’t have parameters that you need to tune--if this is the case for the one you picked, identify and briefly explain how you would have done it if you used, say, a decision tree classifier). [relevant rubric item: “tune the algorithm”]
3. What is validation, and what’s a classic mistake you can make if you do it wrong?  How did you validate your analysis?  [relevant rubric item: “validation strategy”]
4. Give at least 2 evaluation metrics, and your average performance for each of them.  Explain an interpretation of your metrics that says something human-understandable about your algorithm’s performance. [relevant rubric item: “usage of evaluation metrics”]