Enron Machine Learning Project
Bryony Miles

Question: Which Enron Employees may have committed fraud based on the public Enron financial and email dataset.

1. Data Exploration

First I listed the 21 available features:

Email related	**Financial**	POI
1. email_address 2. to_messages 3. from_messages 4. shared_receipt_with_poi 5. from_this_person_to_poi 6. from_poi_to_this_person	1. deferred_income 2. salary 3. director_fees 4. total_payments 5. long_term_incentive 6. expenses 7. exercised_stock_options 8. restricted_stock_deferred 9. restricted_stock 10. loan_advances 11. bonus 12. total_stock_value 13. deferral_payments 14. other	true(1) or false(0)

I then used *explore_enron_data.py* as a starting point to confirm what I knew already:

```
number of players: 146
total POIs: 18
```

I then had a look at the null values, for everyone and just for POI's. Stats gleaned as follows:

Field	No.of Nulls	POI null values
email address	35	
email address but **no messages**	60	`4: ['FASTOW ANDREW S', 'KOPPER MICHAEL J', 'HIRKO JOSEPH', 'YEAGER F SCOTT']`
salary	51	`1: ['HIRKO JOSEPH']`
restricted stock	36	`1: ['HIRKO JOSEPH']`
bonus	64	`2: ['YEAGER F SCOTT', 'HIRKO JOSEPH']`
long term incentive	80	`6: ['RIEKER PAULA H', 'YEAGER F SCOTT', 'BELDEN TIMOTHY N', 'COLWELL WESLEY', 'SHELBY REX', 'HIRKO JOSEPH']`
director fees	129	`no POIs have values`
restricted_stock_deferred	128	`no POIs with values`
exercised stock options	44	`6: ['KOPPER MICHAEL J', 'COLWELL WESLEY', 'FASTOW ANDREW S', 'BOWEN JR RAYMOND M', 'CALGER CHRISTOPHER F', 'CAUSEY RICHARD A']`
3 finance fields: director_fees, restricted_stock_deferred, exercised_stock_options,	29	`6: see above`

deferred income	97	7: ['SKILLING JEFFREY K', 'YEAGER F SCOTT', 'GLISAN JR BEN F', 'HIRKO JOSEPH', 'DELAINEY DAVID W', 'KOPPER MICHAEL J', 'KOENIG MARK E']
deferral payments	107	All null except 5: ['RIEKER PAULA H'],['LAY KENNETH L'],['BELDEN TIMOTHY N'],['COLWELL WESLEY'],['HIRKO JOSEPH']
loan advances	142	All null except 1: ['LAY KENNETH L']
other	53	all POIs have values
expenses	51	all POIs have values
total stock value	20	all POIs have values
total payments	21	all POIs have values

Thoughts and Questions at this stage:

- All POI's have expenses, total stock options, total payments and exercised stock options
- There are no POI directors (if we assume you only receive the fees if you are one)
- Only 4 people, including one POI had a loan advance
- Deferral payments (39/146), Restricted Stock Deferred (18/146) and Director Fees (17/146) are rare.
- All POIs have email addresses but 4 have no messages
- Why no salary? Freelance?

Joe Hirko and Scott Yeager have a lot of null values. Are they important?

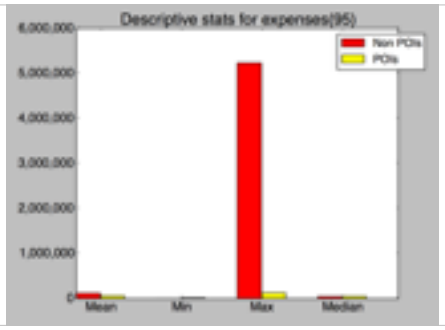
```
{'email_address': 'joe.hirko@enron.com',
   'deferral_payments': 10259,
   'expenses': 77978,
   'exercised_stock_options': 30766064,
   'total_stock_value': 30766064,
   'poi': True}

{'email_address': 'scott.yeager@enron.com',
   'other': 147950,
   'salary': 158403,
   'total_payments': 360300,
   'expenses': 53947,
   'restricted_stock': 3576206,
   'exercised_stock_options': 8308552,
   'total_stock_value': 11884758,
   'poi': True}
```

The data they do have shows they could definitely be POIs!

Decision: convert all null values to 0 for the moment.

2. Outliers

To look at potential outlers, I drew descriptive stats graphs for each field. All the fields were behaving really oddly with very high values for non POIs such as expenses below.



I searched for expenses over 1000000 and returned the following:

```
TOTAL : expenses are  5235198
```

That's the first outlier to remove!

I deleted TOTAL and ran the graphs again and noticed some non-POIs with high values (see below). For this project I'm to assume that means they were exonerated. They might be useful later on for comparisons with POIs though:

```
HORTON STANLEY C : exercised_stock_options are  5210569
DERRICK JR. JAMES V : exercised_stock_options are  8831913
CHRISTODOULOU DIOMEDES : exercised_stock_options are  5127155
THORN TERENCE H : exercised_stock_options are  4452476
FREVERT MARK A : salary are  1060932, deferral_payments are  6426990, deferred_income are  -3367011,
exercised_stock_options are  10433518, other are  7427621
DIMICHELE RICHARD G : exercised_stock_options are  8191755
MARTIN AMANDA K : long_term_incentive are  5145434
WALLS JR ROBERT H : exercised_stock_options are  4346544
MCCLELLAN GEORGE : expenses are  228763
OVERDYKE JR JERE C : exercised_stock_options are  5266578
REDMOND BRIAN L : exercised_stock_options are  7509039
BAXTER JOHN C : exercised_stock_options are  6680544
ELLIOTT STEVEN : exercised_stock_options are  4890344
REYNOLDS LAWRENCE : exercised_stock_options are  4160672
URQUHART JOHN A : expenses are  228656
BANNANTINE JAMES M : exercised_stock_options are  4046157
ALLEN PHILLIP K : deferred_income are  -3081055
LAVORATO JOHN J : bonus are  8000000, exercised_stock_options are  4158995
PAI LOU L : exercised_stock_options are  15364167, total_stock_value are  23817930
WHITE JR THOMAS E : restricted_stock are  13847074
```

En route I also noticed the name "THE TRAVEL AGENCY IN THE PARK". Searching for this I noticed there are only other payments. This is outlier number 2, remove it. I checked all the other names manually and they all look like genuine people.

```
{'restricted_stock_deferred': 0, 'from_poi_to_this_person': 0, 'from_this_person_to_poi': 0,
'exercised_stock_options': 0, 'total_payments': 362096, 'long_term_incentive': 0,
'restricted_stock': 0, 'deferral_payments': 0, 'other': 362096, 'to_messages': 0,
'total_stock_value': 0, 'salary': 0, 'email_address': 0, 'loan_advances': 0, 'expenses': 0,
'shared_receipt_with_poi': 0, 'poi': False, 'from_messages': 0, 'bonus': 0, 'director_fees': 0,
'deferred_income': 0}
```

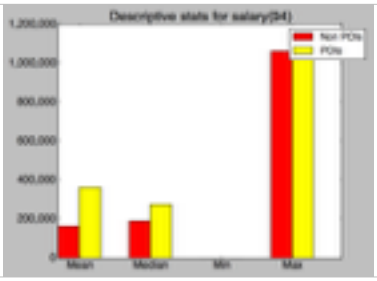
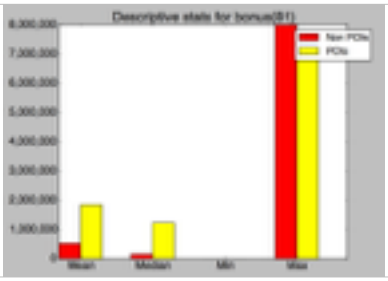
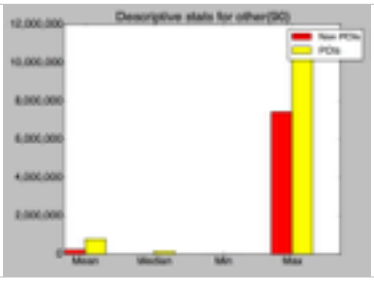
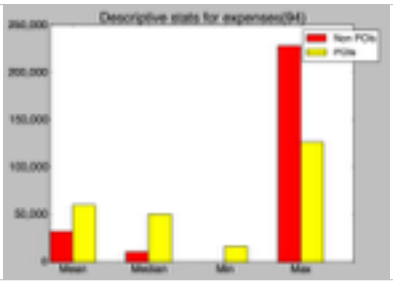
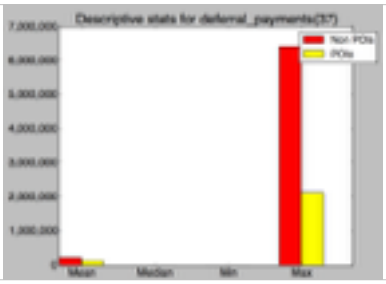
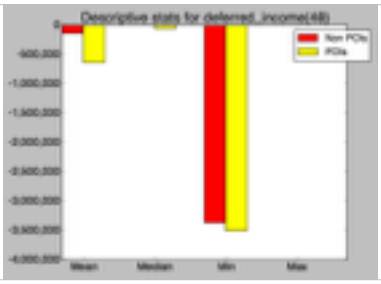
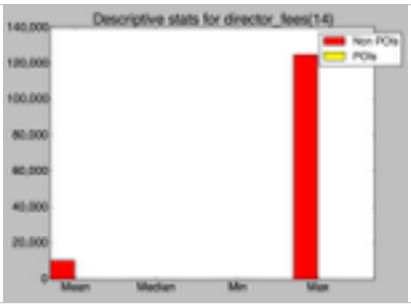
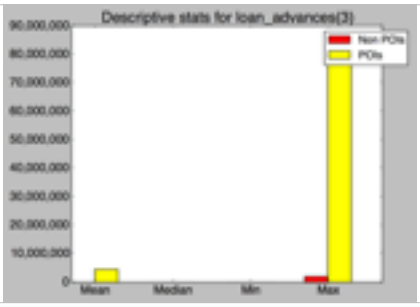
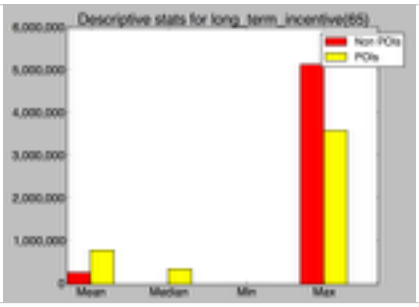
I then decided to look at the total payments field. After a bit of exploration, I discovered it was a sum of the certain financial fields. However there were two:

BHATNAGAR SANJAY **total_payments : 15456290 (137864)** deferred_income : 0 salary : 0 director_fees : 137864 **(0)** long_term_incentive : 0 expenses : 0 **(137864)** exercised_stock_options : 2604490 **(15456290)** restricted_stock_deferred : 15456290 **(-2604490)** restricted_stock : -2604490 **(2604490)** loan_advances : 0 bonus : 0 total_stock_value : 0 **(15456290)** deferral_payments : 0 other : 137864 **(0)**	BELFER ROBERT **total_payments : (3285)** deferred_income : 0 **(-102500)** salary : 0 director_fees : **(102500)** long_term_incentive : 0 expenses : 0 **(3285)** exercised_stock_options : 3285 **(0)** restricted_stock_deferred : 44093 **(-44093)** restricted_stock : 0 **(44093)** loan_advances : 0 bonus : 0 total_stock_value : -44093 **(0)** deferral_payments : -102500 **(0)** other : 0

I went through various options and then discovered the pdf *"enron61702insiderpay.pdf".* There seemed to be a data entry problem. The actual values are in brackets in purple. I therefore updated them manually in the code.

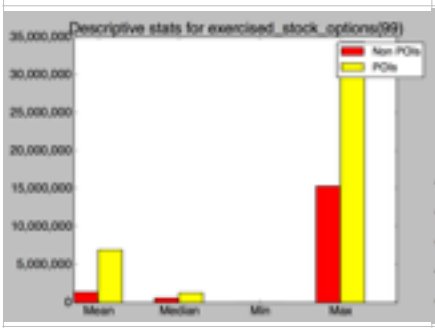
3. Create new features

I used the graphs to help me decide on new features.

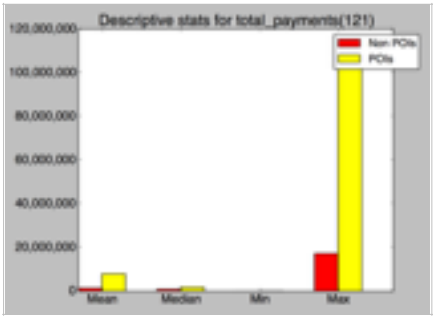
Salary: clearly a key feature.	**Bonus:** check relationship with salary.	**Other:** disproportionately high values.
		
Expenses: not significant amounts. **(don't use this feature at this stage)**	**Deferral Payments:** high values again.	**Deferred Income:** compare with payments?
		
Director Fees: Relates to non-employee directors. low amounts and no POIs. **(don't use this feature at this stage)**	**Loan advances:** some dodgy here!	**Long term incentive:** Relates to employee retention which I would argue could put this in the same category as stocks..
		

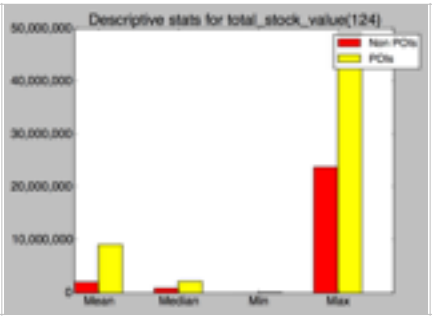
New payment related features:

- *key_payments:* salary + bonus + other
- *deferral_balance:* deferral_payments + deferred_ income

Exercised SO: amount paid out	**Restricted_stock** and **Restricted_stock_deferred:** Only nonPOIs have deferred. The amounts concerned for deferred are also low compared to restricted stock (which includes deferred). However, as it applies to non-POIs I would like to include it as a total variable.
	

No new variables here.





Finally two new features linked to totals:

retention incentives: long_term_incentive + total_stock_value

total_of_totals: total_payments + total_stock_value

4. Intelligently select features

Firstly, I decided to miss out two features based on the descriptive stats graphs:

- expenses - the amounts involved (240,000 or less) were insignificant compared to the other figures.
- director_fees - no POI's involved so potentially less relevant.

I then ran SelectKBest and SelectPercentile on the remaining fields + my new features but the results were very inconclusive with all my features:

```
old_features_list = ['poi','salary','bonus','other','deferral_payments',
'deferred_income','loan_advances','long_term_incentive',
'exercised_stock_options','restricted_stock_deferred','restricted_stock',
'key_payments','deferral_balance', 'total_payments', 'total_stock_value',
'retention_incentives','total_of_totals']
```

and a select few:

```
features_list = ['poi','loan_advances',
'key_payments','deferral_balance', 'total_payments', 'total_stock_value',
'long_term_incentive','total_of_totals']
```

I am looking for information, conclusion and insights. I also want to make sure that I don't have too many features and overfit the data. Therefore for my first round I'm going to go with five features which together capture the remaining finance features:

- **key_payments** - bonus + salary + other
- **deferral_balance** - deferred_payments + deferred_income
- **loan_advances**
- **retention incentives** - long_term_incentive + total_stock_value
- **total_of_totals** - total_payments + total_stock_value

I can always come back and try again later. I have not looked at email data at this stage.

5. Properly scale features

About the features
- they all the same type - integer
- they could all be related
- there are a lot of outliers which are potentially key
- the numbers vary considerably between features

I therefore applied a MinMaxScaler.

6. Pick an algorithm

Now for the fun bit! Finding an algorithm and playing around with the features.

SVC

I started with this. This was causing confusion as it returned a suspiciously high accuracy score and a Divide by zero error. It was throwing out 0 true and false positives - i.e ignoring all the POI's.

With a little help from the forum I made sure the labels were working correctly and still got the same error. It turns out this has caused other students trouble. In one post I was directed to it was solved by setting C to 1,000,000 but I decided in the end to test out some other algorithms instead. I suspect changing the kernel would have made a difference too but as there are still lots to choose from I decided to look at some other algorithms.

I tested five: GaussianNB, LinearSVC, Decision Tree, Logistic Regression and Random Forest looking at precision and recall.

The best results were from Random Forest. Precision levels were pretty high (around 0.6) but recall wasn't quite there (around 0.23).

It took a lot of tweaking: changing the features, sample size, algorithm parameters. At one point I decided to add a new composite feature:

poiemails - boolean, true if values in any of the following features: shared_receipt_with_poi, from_this_person_to_poi, from_poi_to_this_person exist.

In the end I got a better result without it. Here's the final algorithm on a 70/30 train/test split:

```
feature_list = ['poi','poi_emails','key_payments','deferral_balance',
'retention_incentives','total_of_totals','loan_advances']

clf = RandomForestClassifier(n_estimators=100, min_samples_split=4,
max_features=None,oob_score=True)

RandomForestClassifier(bootstrap=True, class_weight=None, criterion='gini',
            max_depth=None, max_features=None, max_leaf_nodes=None,
            min_samples_leaf=1, min_samples_split=6,
            min_weight_fraction_leaf=0.0, n_estimators=100, n_jobs=1,
            oob_score=True, random_state=None, verbose=0, warm_start=False)
      Accuracy: 0.86947 Precision: 0.51789     Recall: 0.30400   F1: 0.38311       F2: 0.33137
      Total predictions: 15000     True positives:  608   False positives:  566   False negatives:
1392  True negatives: 12434
```

As a matter of interest, here are the results for the other algorithms (same features and test size).

Algorithm	Precision	Recall
Linear SVC	0.18067	0.27100
Gaussian NB	0.23362	0.25850
Logistic Regression	0.18201	0.17700
Decision Tree	0.27778	0.02250

Just for fun I experimented with deleting the nonPOIs with high values found earlier to see how that affected the results…

```
RandomForestClassifier(bootstrap=True, class_weight=None, criterion='gini',
            max_depth=None, max_features=None, max_leaf_nodes=None,
            min_samples_leaf=1, min_samples_split=4,
            min_weight_fraction_leaf=0.0, n_estimators=100, n_jobs=1,
            oob_score=True, random_state=None, verbose=0, warm_start=False)
      Accuracy: 0.87092 Precision: 0.62404     Recall: 0.40500   F1: 0.49121       F2: 0.43558
      Total predictions: 13000     True positives:  810   False positives:  488   False negatives:
1190  True negatives: 10512
```

This is a fantastic score but it is not a valid algorithm. It discounts 20 records from the data (14%) and assumes that anyone who was paid over a certain amount is automatically guilty. Suspicious though!

This has been a fabulous project. I would like to come back to the data again and have a dig into the email content linked to POIs.

The code is found in four python files:

- *explore_data.py*
- *feature_selection.py (commented out in this case)*
- *algorithms.py*
- *poi_id.py*