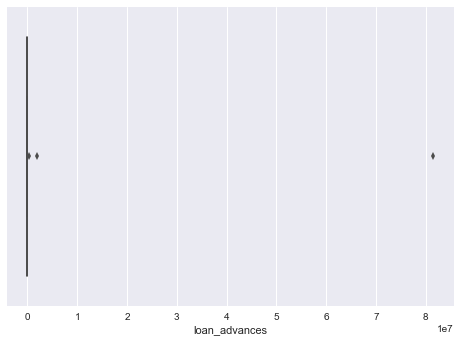
# Data Understanding

## Data Exploration

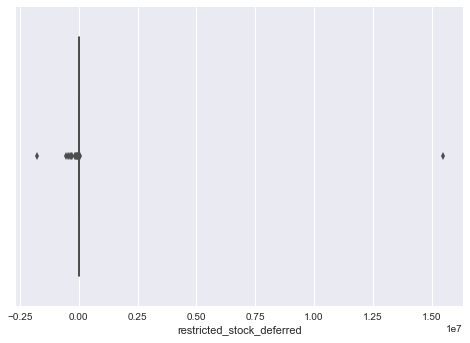
## Outliers

The first step is to detect any outliers and to see if they need to be removed from the dataset. The TOTAL entry is removed, as it details the sum over all persons included in the dataset. For the remaining entries, boxplots are created, as this allows detecting outliers easily.

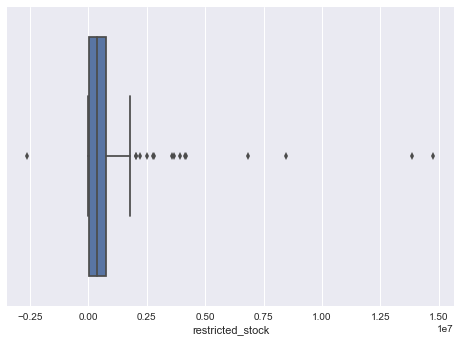
Three parameters show outliers that require further investigation:



loan\_advances: Only few data points are available for this parameter. One extreme point (81,525,000) sticks out. As this data point belongs to Ken Lay, it should be considered as valid.



restricted\_stock\_deferred: Only few data points available, most of them negative. The positive value requires further checking.



restricted\_stock: one negative value whereas the remaining data points are positive.

The parameters total\_payments and total\_stockvalue are the sum of all payments (salary, bonus, etc) and stock values (exercised stock option, restricted stock option, and restricted stock option deferred). So, either all other values are excluded and only the sums are included or vice versa.

Lastly, it can be seen that a lot of parameters contain zeros (i.e. NaNs in the original dictionary). As this may influence the performance of the algorithm, all features which contain more than 75% of zeros are removed.

The retained parameters are:

['poi',

'salary',

'deferral\_payments',

'bonus',

'deferred\_income',

'expenses',

'exercised\_stock\_options',

'other',

'long\_term\_incentive',

'restricted\_stock',

'to\_messages',

'from\_poi\_to\_this\_person',

'from\_messages',

'from\_this\_person\_to\_poi',

'shared\_receipt\_with\_poi']

# Algorithm Selection

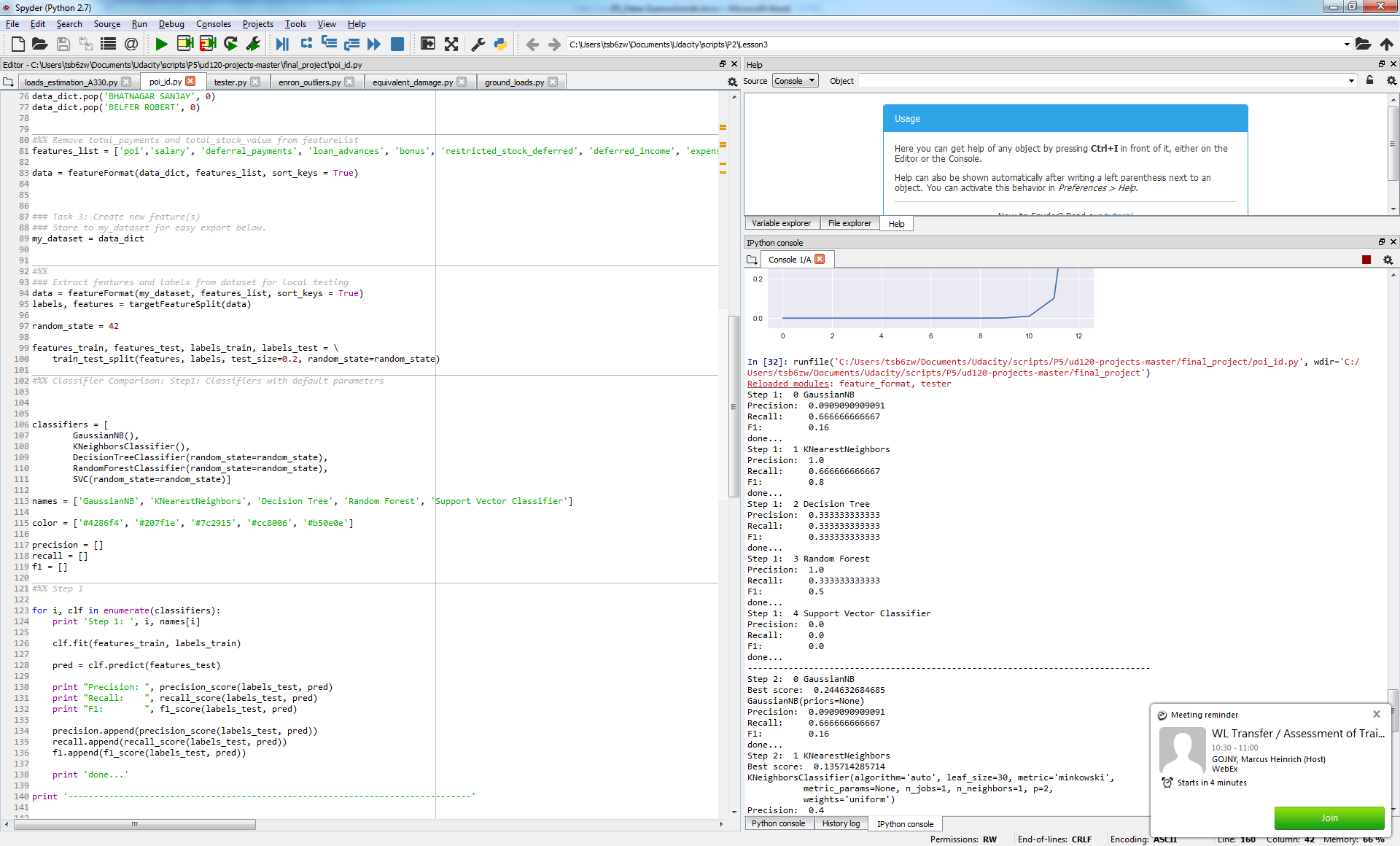
Algorithm selection is done in several steps:

1. Split dataset into a training and a test set. Here, this is done with train\_test\_split; the size of the test set is .2
2. Train classifiers with default parameters (no tuning) on the training set
3. Calculate precision, recall, and f1 score based on test set
4. No feature selection at this point (except the ones that were removed in the previous step)

Here 5 different algorithms are selected:

1. Gaussian Naives Bayes
2. K Nearest Neighbors
3. Decision Tree
4. Random Forest
5. Support Vector Classifier

The results of this first run are shown in the following figure:



# Feature Selection

# Final Validation

In the previous section it was seen that the results in terms of precision and recall differ when simply using train\_test\_split and when using StratifiedShuffleSplit.

This shows the importance of the validation, as the performance may depend a lot on the selected test set.

The metrics used to evaluate the performance are precision and recall. What this means for this project is as follows:

* Low Recall: Number of False Negatives is too high, i.e. a lot of employees are predicted not to be a POI whereas in reality they were
* Low Precision: Number of False Positive is too high, i.e. a lot of employed are predicted to be a POI whereas in reality they were not.
* In both cases the number of True Positives may also be too low, i.e. the number of correctly identified POIs

It is arguably which one is more important but I would tend to achieve a better recall than precision, as it seems better to falsely identify someone as POI (and to exonerate them later) than to miss a potential POI. On the other hand, … So f1