1. (100 points) For the dataset given in file `chem.dataset`, implement a boosting approach to learn to classify the given chemical reaction systems.

- Given the handout of Mon, Oct 16, 2006, first gain an overall understanding of the dataset. Write some basic code to parse a chemical reaction signature and infer simple properties of it (that you will then use as input features to your learning algorithms).

- Next, implement three machine learning algorithms: a decision tree algorithm with information gain as the tree growing metric, a decision stump algorithm, and an alternating decision tree algorithm. As described in class, you really only have to implement the first, the second and third are minor variants thereof. The decision stump algorithm merely learns only one node (i.e., the root). The alternating decision tree algorithm is simply a way to compress multiple decision trees into a single structure, again via boosting. See: [Y. Freund and L. Mason, The Alternating Decision Tree Learning Algorithm, in Proceedings of the Sixteenth International Conference on Machine Learning (ICML 1999), pages 124-133, 1999.] If you read this paper, you will notice that this algorithm already comes with boosting built in (see next point below).

- Finally, implement the boosting algorithm. Use the AdaBoost algorithm given in your textbook for the decision tree and decision stump approaches. For the alternating decision tree approach, the boosting is multiplexed with the tree learning, as explained in the [Freund and Mason, 1999] paper.

- Observe that the dataset has nearly 36,000 instances, but the distribution of positive and negatives is rather skewed. For instance, there are only about 10% positive examples! (What this means is that, by default, you can always predict negative and you get a weak learner!) You should split the given dataset yourself into training and test sets. This split must be made so as to keep the distribution of positives/negatives the same across training and test sets (only then the conditions of PAC learning theory apply).

Learning from such skewed datasets is a common theme in practical machine learning. You might be interested in reading [M.V. Joshi, R.C. Agarawal, and V. Kumar, Predicting Rare Classes: Can Boosting make any Weak Learner Strong?, in Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pages 297-306, 2002] just to understand the underlying context, not for the gory mathematical details.

Track the performance of boosting as a function of the number of weak learners combined. Explain if/how the performance of boosting correlates with the choice of the underlying learning algorithm.