You are to implement logistic regression and AdaBoost for predicting that email’s are spam. A link to the data is provided on the class web site.

**Base task:**
For logistic regression, regularize with two-norm squared and do batch gradient descent. You need to tune the amount of regularization with cross validation.

For AdaBoost convert the 0/1 data to ±1 data and use the features as weak learners. Note that since the features are ±1, \( Z(w) = \tilde{Z}(w) \) and no line search is needed (See lecture 9). You need to tune the number of iterations with cross validation.

For the base task do cross validation according to the following scheme:

```plaintext
for i=1 to 10
    permute data
    split into 3/4 training and 1/4 testing set
    use 1/5 of your training set as a validation set
    for choosing your parameter
    Report the average performance of the 10 runs
    as well as the standard deviation
```

For logistic regression minimize the following objective with batch gradient descent:

\[
\lambda \|w\|_2^2 + \frac{1}{T} \sum_{t=1}^{T} \text{Loss}(y_t, \sigma(w \cdot x_t)).
\]

Stop when the 1-norm of the gradient of the above objective is \( \leq 10^{-4} \).

**Full credit:** Do at least one of the following

1. Implement the following fancier version of cross validation:

```plaintext
for i=1 to 10
    permute data
    split into 3/4 training and 1/4 testing set
    Now do 5 fold cross validation to determine
    the best model parameters:
    - partition training set into 5 parts
```
- for each of the 5 holdouts
  * train all models on the 4/5 part
  * record average logistic loss on 1/5 part
- The best model is chosen as the one with the best
  average over the 5 holdouts
Evaluate the best model by computing the average logistic
loss on the 1/4 test set
Report the average performance of the best model on the 10 runs
as well as the standard deviation

2. For Boosting, implement one of the fancier Boosting algorithms such
   as AdaBoost* or LPBoost.

3. Compare 1-norm to squared 2-norm regularization for logistic regres-
   sion. Train all models to the same precision of the objectives.

4. Regularize logistic regression with early stopping. Use stochastic gra-
   dient descent for doing this:
   
   Do $P$ passes over your data.
   - Always do a gradient descent step for the current single example.
   - At pass $i$ use learning rate $\eta_0 \alpha^{i-1}$.
   
   Good choices are $P = 100$, $\eta_0 = .2$, $\alpha = .95$.
   Use cross validation to tune for the number of passes $P$. You might
   also tune $\eta_0$ or $\alpha$.

5. For logistic regression try a version of Shrink/Stretch. The model
   parameters would be various choices for the interval $[a, b]$.
   
   Train each model with $P = 100$, $\eta_0 = .2$, $\alpha = .95$ or some other rea-
   sonable choice until the total gradient of the objective
   $\sum_{t=1}^{T} \frac{\text{Loss}(y_t, \hat{y}_t)}{T}$ is
   $\leq 10^{-4}$.

Write a short report summarizing what you did including interesting
visualizations.

More helpful suggestions/hints:

- If your algorithms are too expensive to experiment with, then
  work on a subset of the data set first.

- Look at the talk presented in Lecture 10. Try to reproduce
  the same type of experimental reasoning.