Download the liblinear package (http://www.csie.ntu.edu.tw/~cjlin/liblinear/) and install it on your machine. Liblinear has stand-alone executables but you can also use the Matlab interface provided in the package.

The following training options are available. You will use option \( -s \) 5.

- \( -s \) type : set type of solver (default 1)
  - for multi-class classification
    - 0 -- L2-regularized logistic regression (primal)
    - 1 -- L2-regularized L2-loss support vector classification (dual)
    - 2 -- L2-regularized L2-loss support vector classification (primal)
    - 3 -- L2-regularized L1-loss support vector classification (dual)
    - 5 -- L1-regularized L2-loss support vector classification
    - 6 -- L1-regularized logistic regression
    - 7 -- L2-regularized logistic regression (dual)

Use the most recent version of the competition data set for the following experiments. First split the competition data into two as train (first 200K rows) and test (remaining rows). Normalize each feature to have range \([0 \ 1]\) or zero mean, unit variance.

Part 1. Using \( s=5 \) and all of the features fill out the following table. F1 score is computed on the test data.

<table>
<thead>
<tr>
<th>C</th>
<th>F1 score</th>
<th>Training time</th>
<th># of support vectors</th>
<th># of nonzero coefficients in ( w ) (averaged over all ( w )’s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.01</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Independently investigate the correlation (if any) between the following pairs of variables.

a. F1 score vs. # of support vectors
b. F1 score vs. # of nonzero coefficients in w
c. F1 score vs. C
d. C vs. training time
e. C vs. # of support vectors
f. C vs. # of nonzero coefficients in w

Part 2. Use the C value that achieves the best F1 score from Part 1 and run your classifier with the top K words ranked according to TF-IDF scores (try K=100, 500, 1000, 2000, 5000). Do you outperform your results from Part 1 for any K?

Part 3. Use your best configuration to make a Kaggle submission.