Tim Siwula & Zhe Xu
Dr. Maria Daltayanni
Data Mining CS 451
December 6, 2016

# Project Report

1) Project title: Kaggle Click Prediction
2) Team members: Tim Siwula & Zhe Xu
3) Timeline:

| Description | to do by | Status |
|---|---|---|
| Meet with team, decide on topic, features and target variables, split tasks | 9/26 | completed |
| Download data, setup git repo with team, install R packages | 9/30 | completed |
| Data exploration/preprocessing | 10/03 | completed |
| Project Proposal Due | 10/03 | completed |
| Data preprocessing | 10/10 | completed |
| Feature selection/creation | 10/17 | completed |
| Run models, evaluate accuracy, plot accuracy result (Round 1) | 10/24 | completed |
| Update models, evaluate accuracy, plot accuracy results (Round 2) | 10/31 | completed |
| Round 3 (with significance of results always tested) | 11/03 | completed |
| Project Progress Report Due | 11/03 | completed |
| Plot comparisons: models (m1, m2, m3, ...), baselines (random), compute % improvements | 11/7 | completed |
| Some improvements | 11/14 | completed |
| Some improvements | 11/21 | completed |
| Some improvements | 11/28 | completed |
| Project Presentation | 12/05 | completed |
| Project Report and Code Due | 12/07 | completed |

4) Problem Definition: To predict the click probability for a given article displayed to a user online. The target variable is if a user clicked on a article or not. We used a supervised learning approach by using logistic regression, SVM and Random Forests.

5) Data Preprocessing: For the data preprocessing, I had to set up a database server and load all of the csv files into the PostgreSQL server. Thereafter I created a new table consisting of several predictors selected across several of the csv files. The data then needed to be sampled in increments of 500K so as not to crash the in memory RStudio bottleneck. Data skewness was also prevent and assed using the e0171 library. After finding out, I balanced the dataset on the response variable called clicked. This created an even 50/50 of clicked/not-clicked data set. Missing values were also present so I replaced those as well with 0's, though they were relatively sparse. In my code you can see explicitly how this is handled.

6) Data Analysis: The features I used were topic_id, confidence_level and document_id. The were informative and affect the target variable clicked because they were all classified as significant in the logistic regression and random forest results. These three predictors all had a statical importance.

7) Training / Testing Setup: The training set was 75% and the test set was 25%. It took a few times get the data in the correct format to perform the regression analysis. I split by sampling the size of the data subset that was at 500K rows. However after omitting empty values and balancing it came to around 10K between the train (7.5K) and test (2.5K) set.

8) Models / Prediction Algorithms: The two types of models I used were logistic regression (glm) and Random Forests. I choose to use these because they were the best in their class for probability prediction. I also tried using Decision Trees that did fairly well.

9) Prediction accuracy: For prediction accuracy the Random Forest model performed the best. I used 400 trees to build it using just topic_id and confidence_level as features and it obtained an accuracy of 69%. The other two models were logistic regression models using different combinations of features and those two received and an accuracy of 61% and 62%.

10) Challenges: Some changes I faced was the up hill battle against lost time after entertaining another project that involved time series analysis. The data integration and custom table creation took a lot of time to set up and get right but once it was done it was more reliable to work with. RStudio kept crashing several times and was quite limited to how much of the data it handle so I was quite restricted to a small sample set of the data.