Identifying “Fake News” with Supervised and Unsupervised Machine Learning

In the lecture, we learned about how network-level features of spammers can be used to distinguish legitimate email senders from spammers. You learned about both supervised learning techniques and clustering techniques, and how they can be applied to email spam and scams.

- **Rulefit / Decision Tree** for identifying network-level properties of spam senders
- **PCA / SVD** for identifying clusters of spammers with similar sending patterns

In this laboratory, you will begin to explore whether similar techniques can be applied to Twitter posts on “fake news”. The R markdown file at [NOTE: URL] will get you started in identifying various Tweets.

The basic idea is that some various trending topics on Twitter that are related to news events might have URLs that correspond to “fake news”.

Can you discover features of these URLs/posts that are more characteristic of “fake news”?

Part 1 (Day 1). Data Download and Feature Extraction. Identify a set of relevant Tweets that you would like to study. The R Markdown file and accompanying Python skeleton file has some ideas to get you started. Today, you will tackle two problems: (1) getting a “news” dataset that you want to study; (2) finding ways to extract relevant features.

One idea for identifying a news event would be to take a trending topic on Twitter for a news topic (or trending public figure or topic).

After you have identified the set of Tweets you would like to analyze, the next step will be to extract the relevant features. The recommended first step will be to **extract the URLs from the Tweets** that you select. From there you can analyze the URLs further. Here are some ideas of things to try:

- How many times did the domain name in the URL appear in Tweets?
- When were these domain names registered?
- Where are the domain names hosted?
- Are there specific words or characters that are common in the domain names?
- Does the website use HTTP or HTTPS?
- What are the other properties of the website?

After you identify some features of the domain names that you would like to analyze, write a script (recommended: R or Python) to extract these features. You may find the included R markdown or iPython notebook [NOTE: URL] useful as a starting point.

Part 2 (Day 2). Supervised or Unsupervised Learning. Choose one of the methods that we learned about in lecture to analyze the data/features that you gathered from Part 1. If you use a supervised learning method, you will want to figure out a way to label “fake news”.

The following resources may be helpful:

- Decision Trees: R, SciKitLearn
- PCA: R, SciKitLearn
- kNN: R: k-Nearest Neighbour Classification, SciKitLearn