Assignment1 writeup

Santina Lin (87325149)

Wed, October 1, 2014

Explantion on each part of the R script and answering some questions for the assignment1.

1 load data

I loaded the data as well as several libraries - library(tm) for doing data processing - library(dplyr) for easier code inspection for the TAs :) - library(proxy) for cosine similarity

A helper method R script is also sourced. This is for increasing readability of the assignment script by putting all the methods into another R script file.

2 Clearn the data

I followed the tutorial script and processed the Corpus by removing spare terms, URL, puncuations, etc.

3 hierarchal clustering

Using the TM package, I made a distance matrix measured with Euclidean distance. Entropy (methods in helper_functions.R) was calculated for each k clustering (k = 1, k = 2, ... , k = 6) and the result is recorded in a list, from which I could retrieve the minimum entropy and the corresponding k value.

4 extending the feature to IDF

The wikipedia article has an great explanation on it. It based off the intuition that the rarer the terms occur in a document, the more information they give when they do appear. Therefore, these terms are weighted for in the classification criteria than those that appear more frequently.

I used 'weightTfIdf' function in the tm package for this.

5 Entropy analysis on IDF

All the entropy values calculated based on the IDF martrix are smaller than the one that uses just the term frequency matrix. In #3 where the term frequency matrix was used to classify, the lowest entropy was 0.4667768 for k = 6... entropy decreases as the k value increase. This could be an indication of overfitting, but in theory as k increases the purity would decrease in each group.

In using IDF, the the best k is 5, with much smaller minimum entropy (0.155134). All the other entropy values for different k are also smaller than those corresponding k values in #3. So this could be a better input for our classification method given that the purity in each group, implied by the entropy values, is higher.

6. Cosine similarity

I used the dist() in the proxy package calculate a new distance matrix. By running classification method on this distance matrix and evaluating the results with entropy analysis, I found that this distance matrix performs yields better results than the previously used method (distance matrix calculated by euclidean method). The best k in this case is 6, with entropy = 0.3738984. I think it performed better than in #3 but not as well as using IDF as an input to a distance matrix function using Euclidean. (thougts: would using IDF and Cosine similiarity to calculate a distance matrix yield the best result? Though not a requirement, I will test this in my own time)

Comparison and words of thoughts:

Different methods indeed yielded different results, not just the best choice for k but also the entropy, thus the quality of the classifications. From the result we obtan in ths assignment, it seems like the method that utilizes Inverse document frequency matrix provides the best answer in classifying our documents. In theory it makes sense . Given a small glimpse at the documents and the term frequency matrix even after we remove the sparse term, we see that there are many words that fall words that can appear in any of three of the category. As our human mind can easily distinguish what terms are relevant, data mining would requires mapping our intuition onto our computer programs and sequences of workflow as close as possible.