grant_text_mining

Steve Frenk

July 29, 2017

Science costs money. Lots of money. And if you want to get money to fund your awesome science, you have to apply for grants.

Writing successful grants is something of an art. In order to compete with the hundreds of other applicants, you really have to sell your ideas, engaging the review panel with claims to the effect of "this is the most novel and important thing ever" while also reassuring them that "this will definitely work, based on all this stuff we already know".

Universities now have entire departments dedicated to helping people with the grant application process, and if you're working as a researcher in academia, your inbox is likely to be inundated with regular invitations to grant writing workshops and the like. However, much the art of grant writing appears to be passed down throughout the generations in a folklore-esque manner from PI to postdoc/student.

The NIH website contains data on all succesful grant applications since 1985. I thought it would be interesting to do some text mining on the abstract data.

Data input and processing

The total amount of data available is pretty massive, so I took abstracts from one year of each decade with the intention of getting a wide representation of time frames. I wrote a shell script to download the data and do a bit of processing/tidying (see download_files.sh in the github repository). The end result is a file for each year, consisting of one abstract per line. To speed things up, I took a sample of 1000 abstracts for each year.

The abstract data needs to be put through a series of processing steps before we can do analysis. The wonderful tm (text mining) and SnowballC ("stemming" - see bellow) packages provide the functions for us to do this. For more details, check out this great blog entry.

```
library(tm)
library(SnowballC)

# Fetch data
corpus <- Corpus(DirSource("./data/R"))

# Remove stopwords (super common words such as "the" "and" "is")
corpus <- tm_map(corpus, removeWords, stopwords("english"))

# Eliminate extra whitespace
corpus <- tm_map(corpus, stripWhitespace)

# Convert text to lower case
corpus <- tm_map(corpus, content_transformer(tolower))

# Remove numbers
corpus <- tm_map(corpus, removeNumbers)

# Remove punctuation
corpus <- tm_map(corpus, removePunctuation)
```

```
# Stemming (collapse words to root eg learning -> learn)
corpus <- tm_map(corpus, stemDocument)

# Make term-document matrix
tdm <- TermDocumentMatrix(corpus)
```

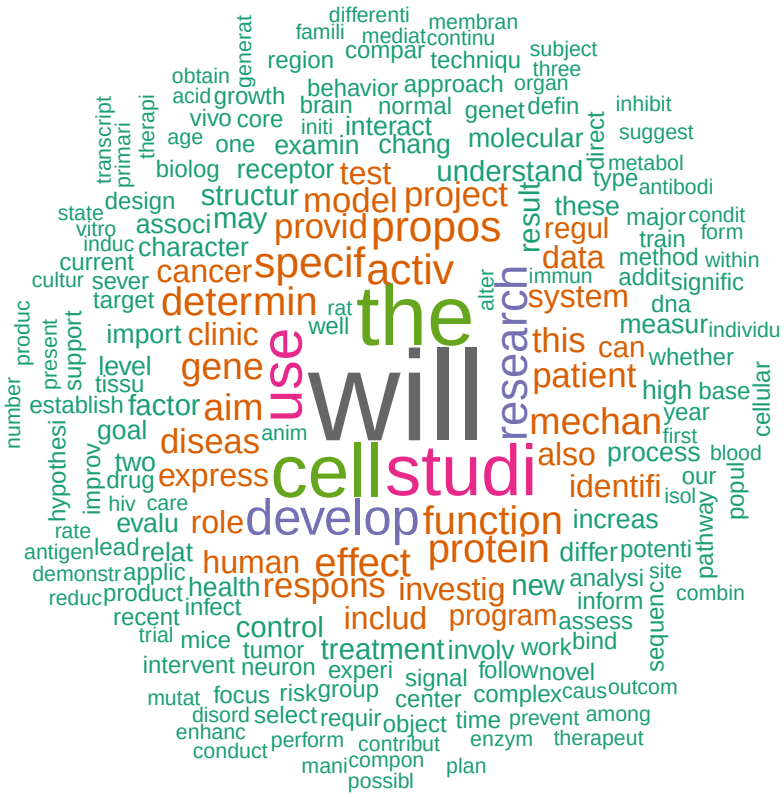
ANALYSIS

We now have a term document matrix - essentially a table containing the frequency of every word in the dataset in our samples. This can be twiddled a bit to make it easier to work with.

```
library(stringi)

# Convert tdm to dataframe and sort by word frequency
m <- as.data.frame(as.matrix(tdm))

# R doesn't like columns names that begin with numbers, so I'm adding a "y" at the beggining of each ye
colnames(m) <- sapply(colnames(m), function(x) stri_replace_all_regex(x, "([0-9]+).*", "y$1"))
m <- m[order(-rowSums(m)),]

# Normalize the word counts by the average total word count for all samples
average_sum <- mean(colSums(m))

for (i in 1:ncol(m)){
    m[i] <- (m[i]/colSums(m)[i]) * average_sum
}
```

Let's look at the most commonly used words in all the grant abstracts.

```
library(wordcloud)

wordcloud(words = rownames(m), freq = rowSums(m), min.freq = 1, max.words = 200, random.order = FALSE, 
```

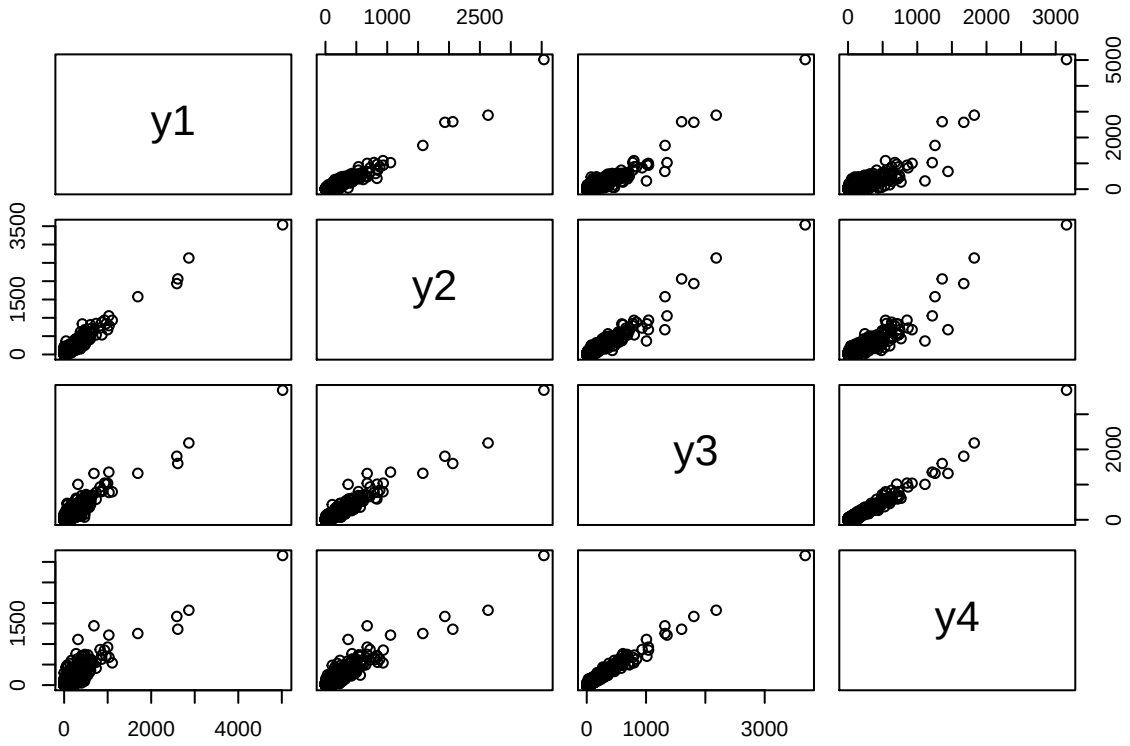


Turns out the most common word is "will". This is perhaps not suprising - when writing a grant, you tend to focus more on the stuff you're planning on doing in the future rather than what's already been done before.

Most of the common words after that tend to be sciencey: "cell", "develop", "studi" (note the stemming).

Let's see how much correlation there is between the different years.

```
pairs(m)
```



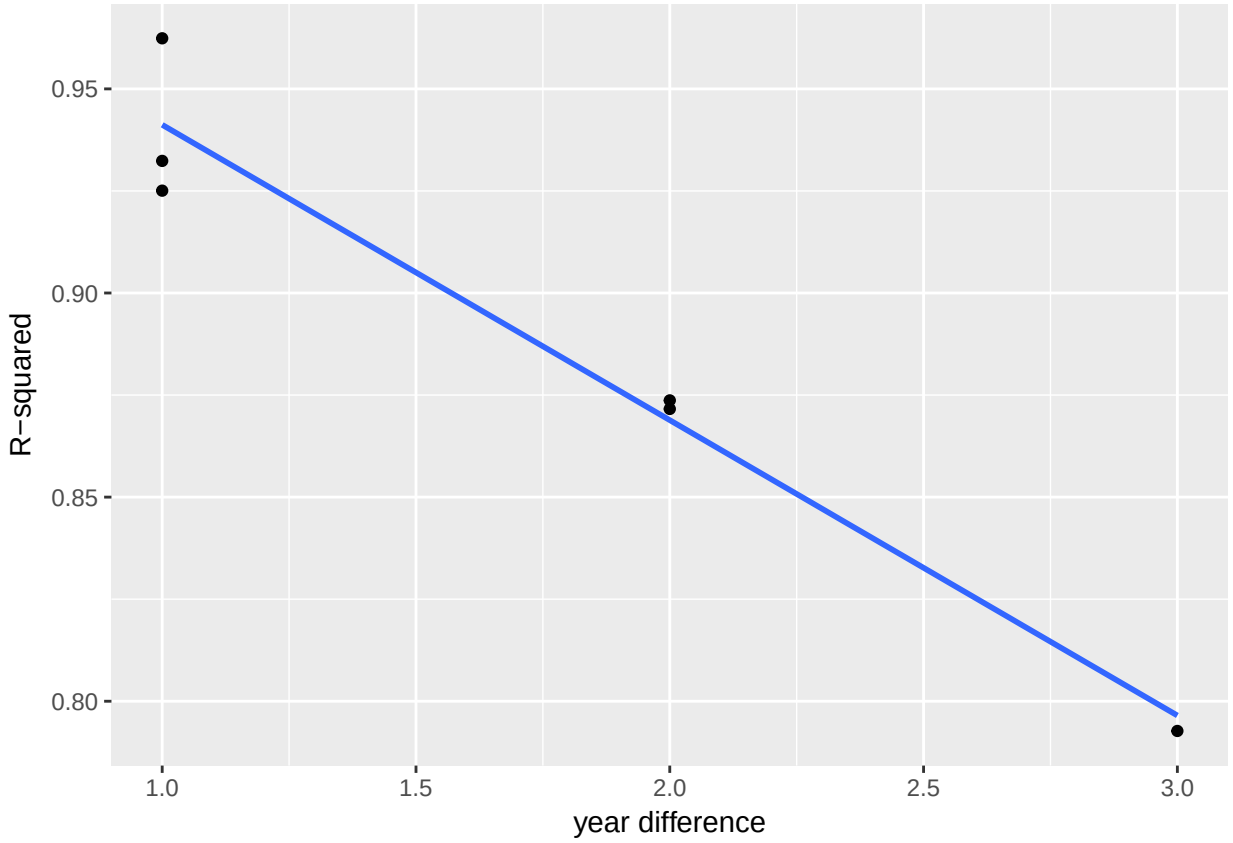
There does seem to be a fairly high amount of correlation between the different years, at least among the really common words. Years that are closer together time-wise seem to be more highly correlated than those that are further apart. Looking at comparrisons with 2016, abstracts seem to look more and more different as we go further back in time. We can look at this more quantitatively.

```
# Get the R-Squared value for each comparison
rsq_df <- data.frame("year_1" = character(), "year_2" = character(), "Rsq" = numeric())

for (i in 1:3){
    for (j in (i+1):4){
        row <- data.frame("year_1" = colnames(m[i]), "year_2" = colnames(m[j]), "Rsq" = summary(lm(m[,i
        rsq_df <- rbind(rsq_df, row)
    }
}

rsq_df
```

```
##   year_1 year_2       Rsq
## 1     y1     y2 0.9323593
## 2     y1     y3 0.8736950
## 3     y1     y4 0.7927186
## 4     y2     y3 0.9250686
## 5     y2     y4 0.8716003
## 6     y3     y4 0.9623908
```

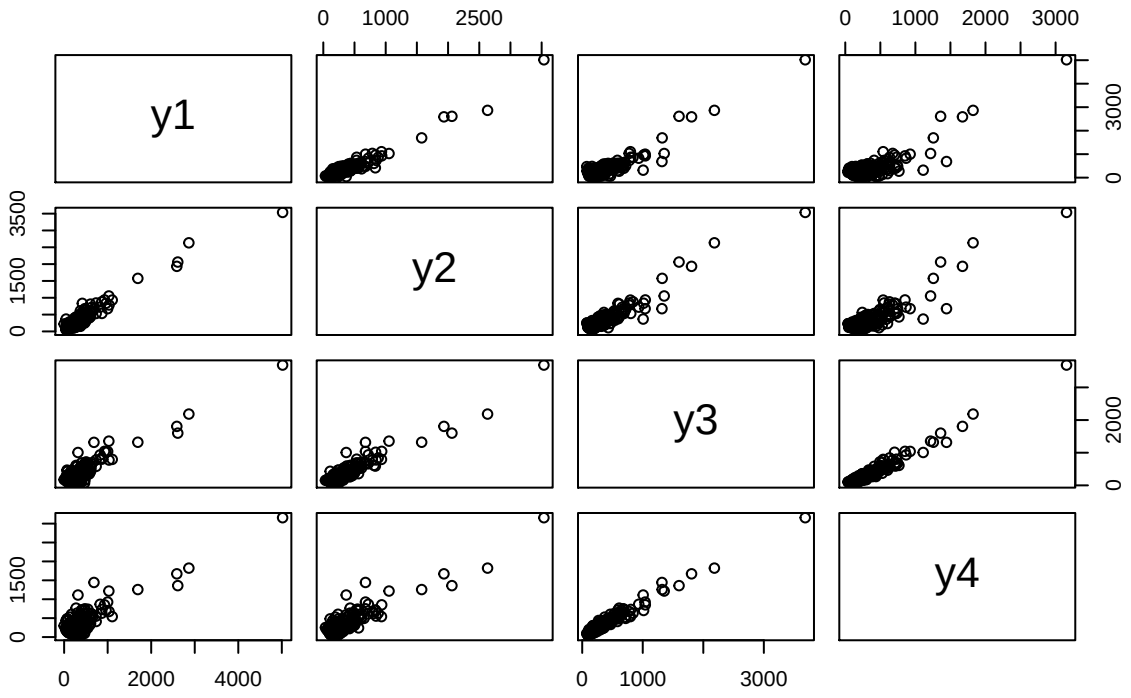
```
# Plot R-Squared against time difference
rsq_df$year_dif <- mapply(function(x,y) abs(as.numeric(gsub("y", "", x))-as.numeric(gsub("y", "", y))),

library(ggplot2)
ggplot(rsq_df, aes(x = year_dif, y = Rsq))+
    geom_point()+
    geom_smooth(method='lm', se = FALSE)+
```

```
    xlab("year difference")+
    ylab("R-squared")
```

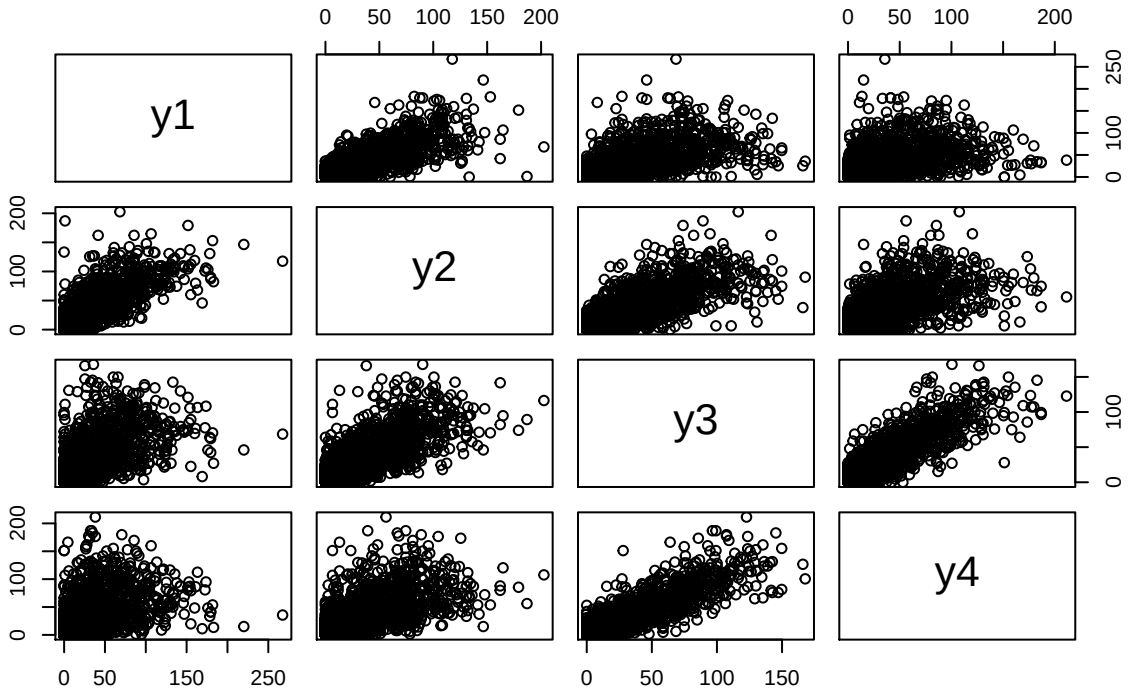


While messing around with the data, I noticed that while the top 1% of words by frequency are highly correlated, the remaining 99% are much less so.

```
pairs(m[1:300,], main = "top 1%")
```

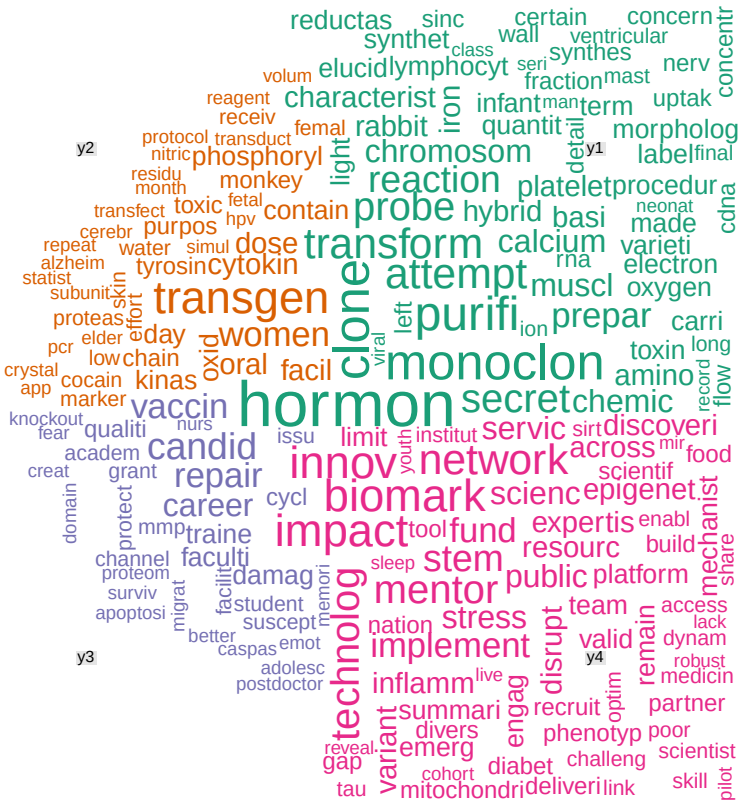



```
pairs(m[301:nrow(m),], main = "bottom 99%")
```



The top 1% presumably consists of the "core" words the will be used in most abstracts. The less frequent words are more interesting, as they may reveal how grant abstracts have changed over the years.

```
bottom_99 <- m[301:nrow(m),]
comparison.cloud(bottom_99, title.size = 0.5, scale = c(2,0.2))
```



Here are a few observations beased on the comparison wordcloud:

- Hormones, monoclonal antibodies and cloning were big in the eighties but have somewhat gone out of fashion since
- The 90s saw "transgenic" being mentioned a lot, presumably building off the pioneering work performed in the 70s and 80s.
- It's more common for succesful grants nowadays to talk about "innovation" and "impact". Networks and biomarkers are also big right now.

It will be interesting to see how these trends develop over the next decade.