Scaling lab

Texas A&M 2015

US Senate Speeches

Let's take a look at US Senate debate on partial birth abortion. We want to use the speeches to estimate individual positions based on Senatorial rhetoric.

Open *R*. Let's first start by installing and loading the package `austin` that allows us to run *Wordscores* or *Wordfish*:

```
library(austin)


## Loading required package: numDeriv
```

If this did not work you may need to install it as follows:

```
install.packages('austin', repos="http://r-forge.r-project.org", type="source")
```

Now change the working directory to the one with the csv file you just created. Now, read in the term document matrix (you might have called yours something else)

```
data <- read.csv('us-abortion-debate.csv', row.names=1)
```

First, let's check that we have the term document matrix correctly. First we chck the dimensions

```
dim(data)


[1]   12 2705
```

The matrix should contain 12 documents (rows) and 2705 words (columns). Typing

```
data[1:5,1:5]


                  mr presid   i thank the
sessionsREP.txt    4      4  31     2 102
santorumREP.txt    3      6 102     5 574
lautenbergDEM.txt  2      5  32     2  87
hatchREP.txt       1      4  15     0  57
harkinDEM.txt      2      1  46     2 134
```

shows the first five documents and the first five words. It does indeed look like a term document matrix.

Before fitting a Wordfish, we need to do some final processing of this matrix. First, we will tell the program that it is a word frequency matrix with the words in columns (it could also look the other way round, that is why we need to tell the computer). Let's write this into a new object that we call senate. This way, we preserve the original data object in the memory, in case we want to go back.

```
senate <- wfm(data, word.margin=2)
```

Now we fit the model. Remember that the scaling procedure is completely based on differences in word frequencies and generates estimates in one dimension. We simply need to fix two documents to tell the program how it should anchor this dimension (we can always aribitrarily rotate the dimension by multiplying all estimates by -1).

By default the first document will a smaller position value than the tenth document. You can alter this by setting the dir parameter (see ?wordfish for details). We estimate and write the output into a new object:

```
senate.res <- wordfish(senate)
```

This should take not very long to estimate. Let's take a look at the estimation results:

```
summary(senate.res)


Call:
wordfish(wfm = senate)

Document Positions:
                  Estimate Std. Error   Lower   Upper
sessionsREP.txt     0.5907    0.04630  0.4999  0.6814
santorumREP.txt     0.5719    0.02215  0.5285  0.6153
lautenbergDEM.txt  -0.5027    0.05817 -0.6168 -0.3887
hatchREP.txt        0.9025    0.04795  0.8085  0.9965
harkinDEM.txt      -0.5572    0.05351 -0.6621 -0.4523
feinsteinDEM.txt   -1.9309    0.01517 -1.9606 -1.9011
ensignREP.txt       0.7371    0.05458  0.6302  0.8441
durbinDEM.txt      -0.2895    0.06349 -0.4139 -0.1651
dewineREP.txt       1.1125    0.03415  1.0455  1.1794
brownbackREP.txt    1.2620    0.01972  1.2233  1.3006
boxerDEM.txt       -1.4280    0.01528 -1.4579 -1.3980
allardREP.txt      -0.3501    0.08199 -0.5108 -0.1894
```

This table shows an overview of the position estimates, the standard errors, and confidence intervals. This is hard to read, so we can plot it instead, as in Fig 1. What story do the estimates tell about the debate?

The analysis suggests that there is a partisan divide, with Democrats having position estimates that are smaller than Republicans. Let's take a look at the word estimates and see how they line up on the dimension. Let's plot the slope estimates for some likely looking word stems. But first we have to extract them from all the other parameters.

```
word.coefs <- coef(senate.res, 'poisson')$words ## just take the word parameters
wds <- c("life","unborn", "choic","her","woman","health","born","babi","defenseless","gruesom","kill")
word.coefs[wds,]


                  beta        psi
```

```
plot(senate.res)
```

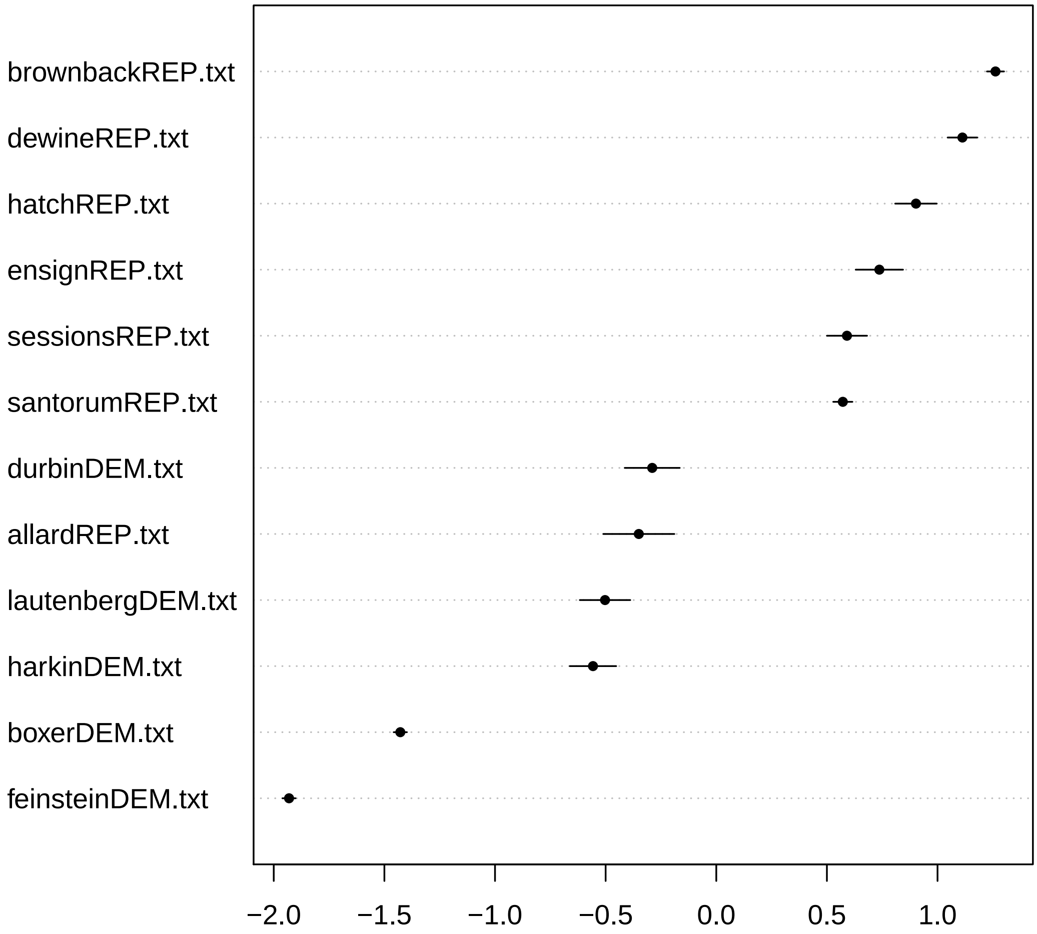

Figure 1: A chart of speaker positions, with uncertainty.

```
life        -0.2930615  2.1350161
unborn       0.3706191 -0.9095174
choic       -0.7901168  0.5999905
her         -0.5903537  1.8643746
woman       -1.1258828  1.2423738
health      -1.0917917  2.3641468
born         0.3985378  0.8163271
babi         0.5370816  1.9659210
defenseless  2.5680198 -4.2917566
gruesom      1.1676885 -1.0422918
kill         0.1140630  1.1792684
```

Do the estimates make sense? Again, it's easier to see if we plot them, as in Fig. 2.

```
dotchart(word.coefs[wds, 'beta'], wds)
```

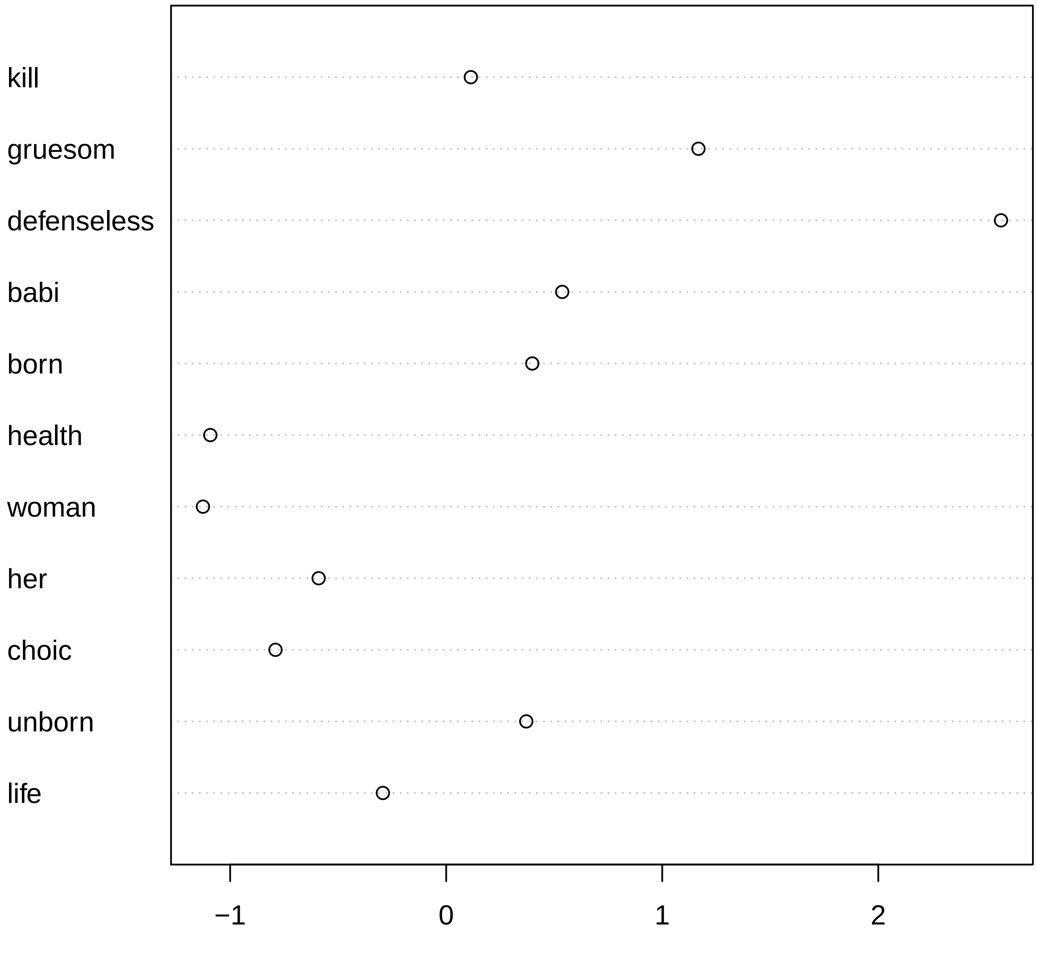

Figure 2: A chart of word slopes (sensitivity to ideological position) for the word stems.

If we were being thorough about these words we'd check they do what we think they do by looking at them in context.