Cluster Analysis of Twitter stuff

Demo for Jon

What cluster analysis tells you

- Cluster analysis, or “unstructured learning AI problems”, are terms used for describing relationships between data items when you don’t have a formal “response variable” Y you’re trying to predict or know anything about.
- Not: “What effect does X have on Y?”
  - There is no Y.
- Instead: “Tell me how many similar/related subgroups within X there are.”
  - Neat for showing hidden patterns, sometimes hard to interpret, inscrutable.

Example with TwitteR

1. Using R, parsed the last 1500 tweets in the hashtags #inflation and #unemployment
2. Break each message in the sample into a “bag of words.” ie, grammar doesn’t matter, just **count** the occurrence of words
3. Check the correlation of the count of each word against others and see if they fall into discernible groups.
4. Graph it as a pretty tree

Get Data

1. Using R, parsed the last 1500 tweets in the hashtags #inflation and #unemployment

```
# paginate to get more tweets
for (page in c(1:15))
{
    # search parameter
    twitter_q <- URLencode('#inflation OR #unemployment')
    # construct a URL
    twitter_url =
paste('http://search.twitter.com/search.atom?q=',twitter_q,'&rpp=100&page=
', page, sep='')
    # fetch remote URL and parse
    mydata.xml <- xmlParseDoc(twitter_url, asText=F)
    # extract the titles
    mydata.vector <- xpathSApply(mydata.xml, '//s:entry/s:title',
xmlValue, namespaces =c('s'='http://www.w3.org/2005/Atom'))
    # aggregate new tweets with previous tweets
    mydata.vectors <- c(mydata.vector, mydata.vectors)
}
```

Process Data

2. Break each message in the sample into a "bag of words." ie, grammar doesn't matter, just count occurrence of words

```
# build a corpus
mydata.corpus <- Corpus(VectorSource(mydata.vectors))

# make each letter lowercase
mydata.corpus <- tm_map(mydata.corpus, tolower)

# remove punctuation
mydata.corpus <- tm_map(mydata.corpus, removePunctuation)

# remove generic and custom stopwords
my_stopwords <- c(stopwords('english'), 'unemployment', 'inflation')
#my_stopwords <- c(stopwords('english'))
mydata.corpus <- tm_map(mydata.corpus, removeWords, my_stopwords)

# build a term-document matrix
mydata.dtm <- TermDocumentMatrix(mydata.corpus)

# remove sparse terms to simplify the cluster plot
mydata.dtm2 <- removeSparseTerms(mydata.dtm, sparse=0.965)

# convert the sparse term-document matrix to a standard data frame
mydata.df <- as.data.frame(inspect(mydata.dtm2))
```

Analyze data

3. Check the correlation of each word against others and see if they fall into discernible groups.

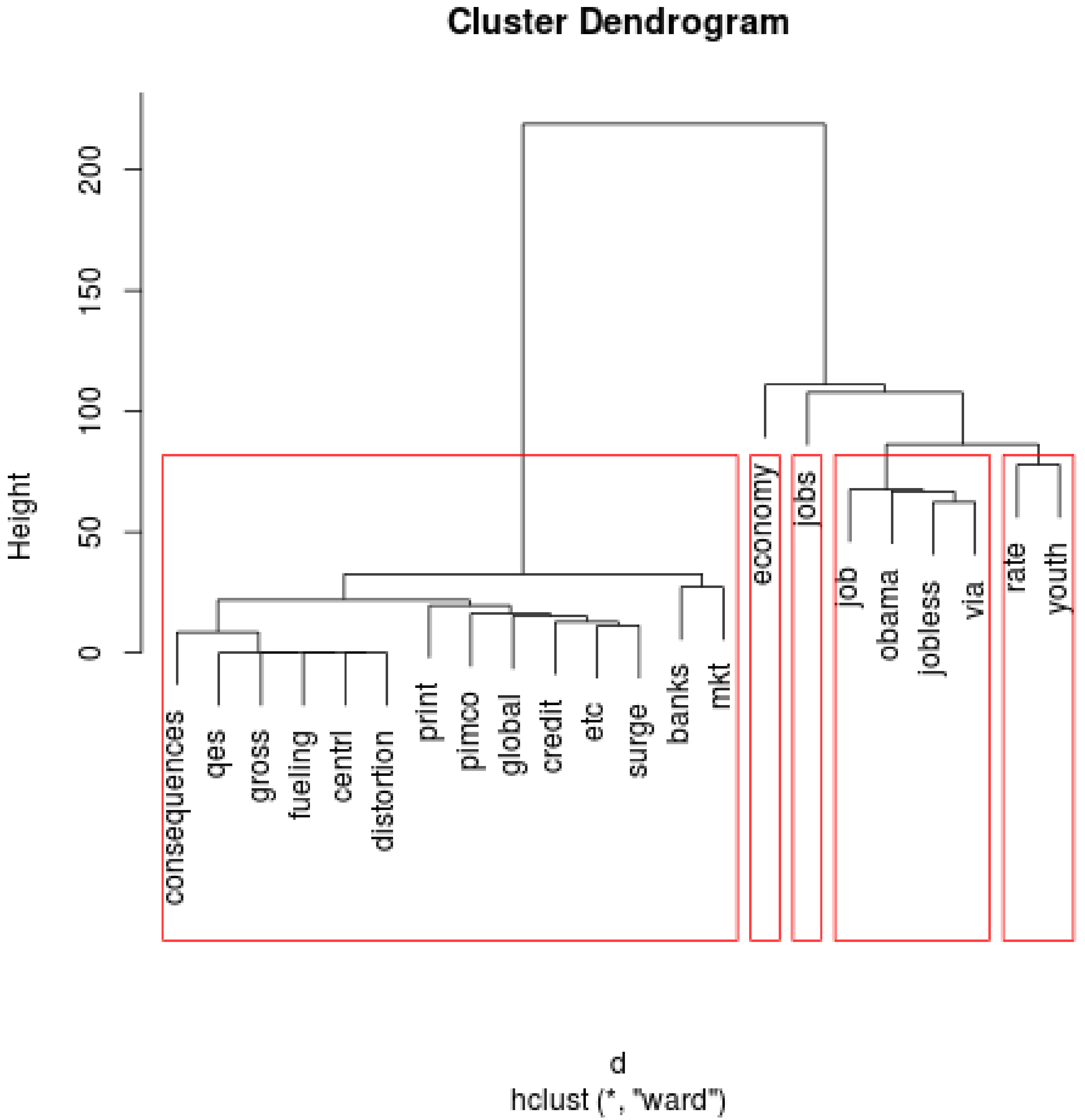
```
mydata.df.scale <- scale(mydata.df)
d <- dist(mydata.df.scale, method = "euclidean") # distance matrix
fit <- hclust(d, method="ward")
```

Visualize Data

4. Graph it as a pretty tree

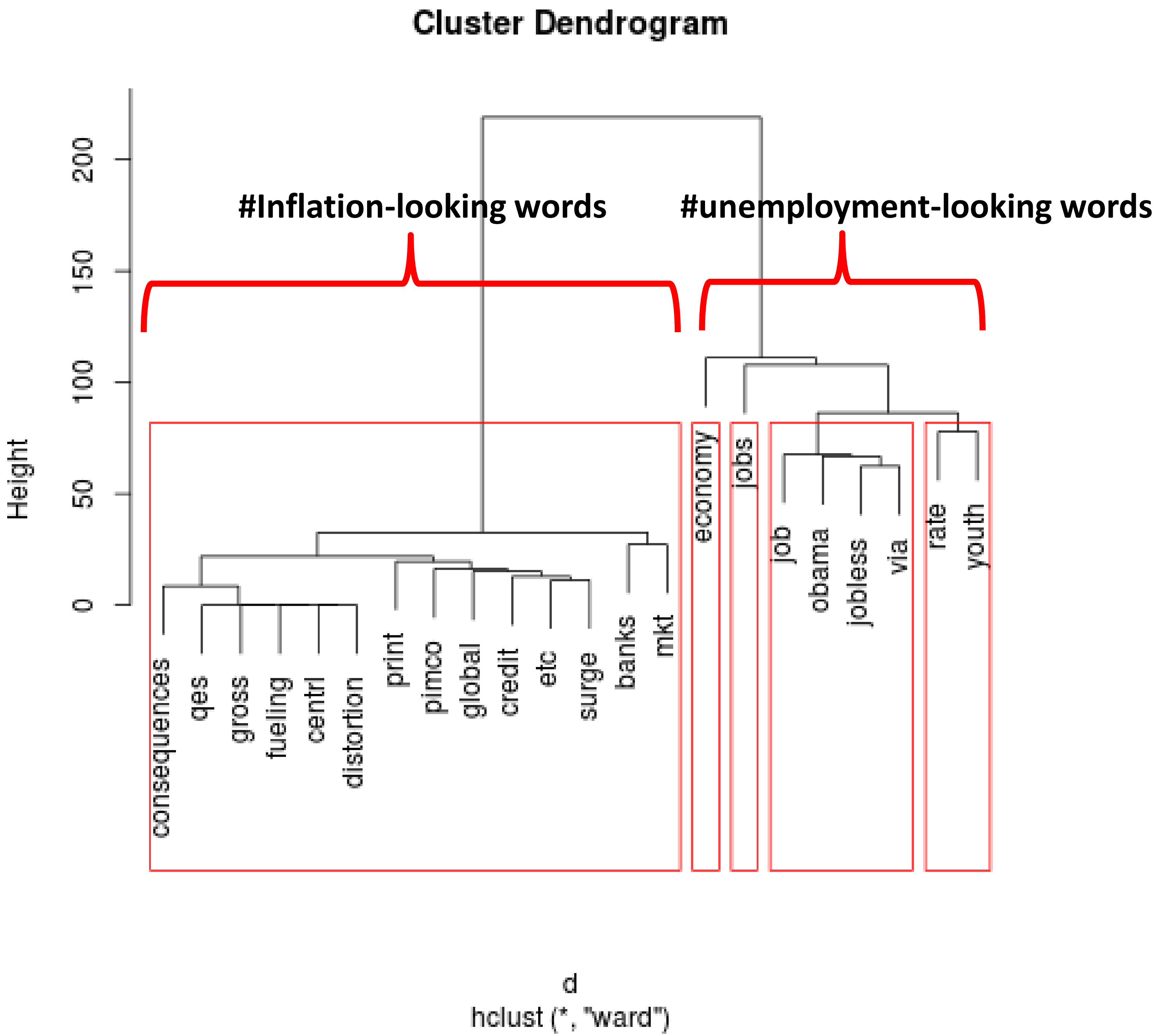
```
png('inflation_and_unemployment.png')
plot(fit) # display dendogram?
groups <- cutree(fit, k=5) # cut tree into 5 clusters
rect.hclust(fit, k=5, border="red")
dev.off()
```

Pretty Tree


Cluster Dendrogram
Height
0
50
100
150
200
consequences
qes
gross
fueling
centrl
distortion
print
pimco
global
credit
etc
surge
banks
mkt
economy
jobs
job
obama
jobless
via
rate
youth
d
hclust (*, "ward")

Interpret Data

- It looks like the algorithm was able to split the tweet data related to the “inflation” tag easily from the tweet data related to the “unemployment” tag
  - Left side of the tree is inflation
  - Right side is unemployment
- We know those were actually different data sources, so it’s good it was able to make that split without us telling it to.



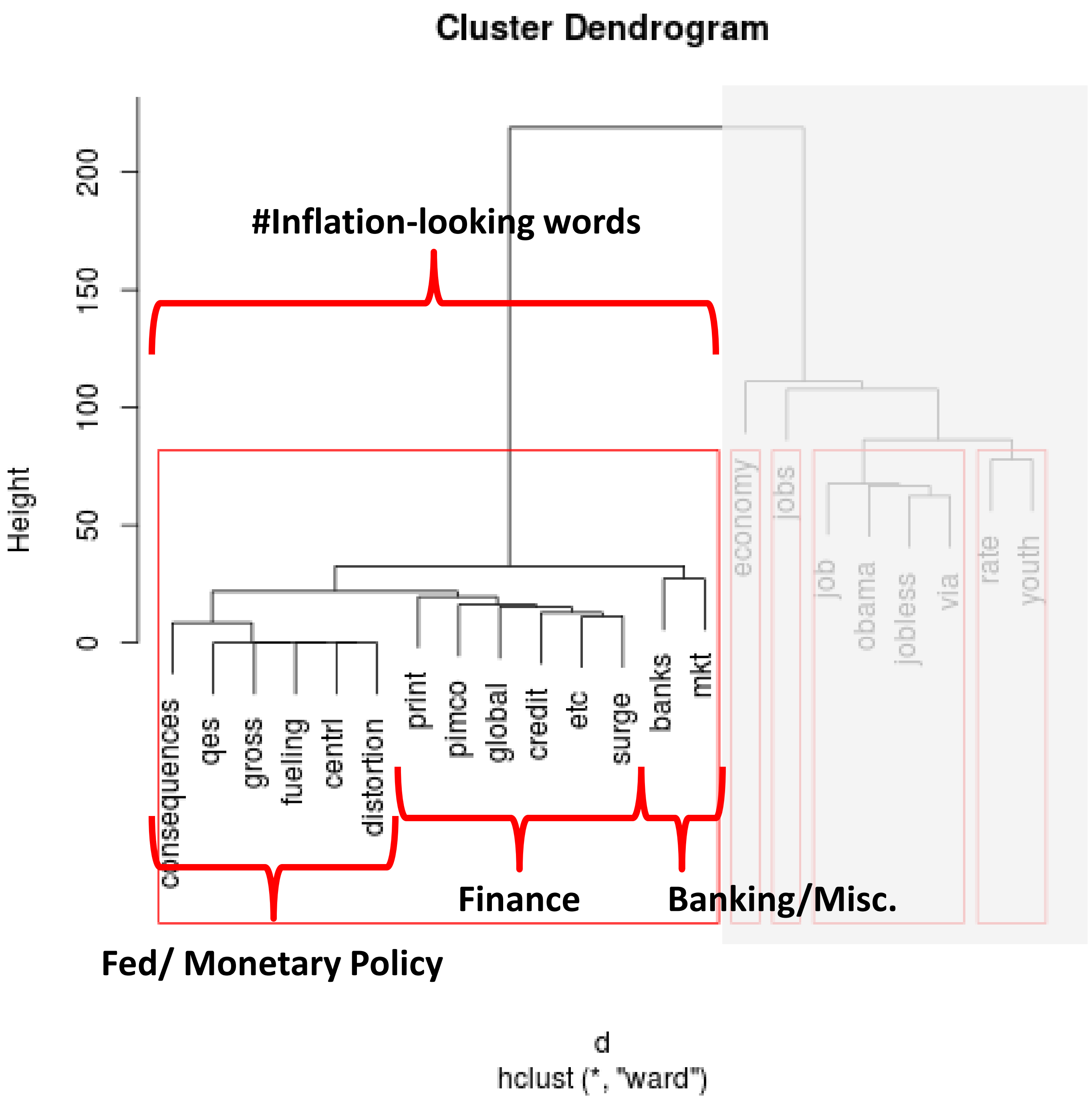
#Inflation-looking words

#unemployment-looking words

Split in two

Within Inflation

- There appear to be 3 categories:
  - Fed/monetary policy
    - Things related to monetary policy
      - Distortion
      - QE
      - Central (bank)
  - Finance
    - Things that have impacted the financial markets today
      - PIMCO
      - Global
      - Credit
  - Banking, miscellany
    - Banking as an institution

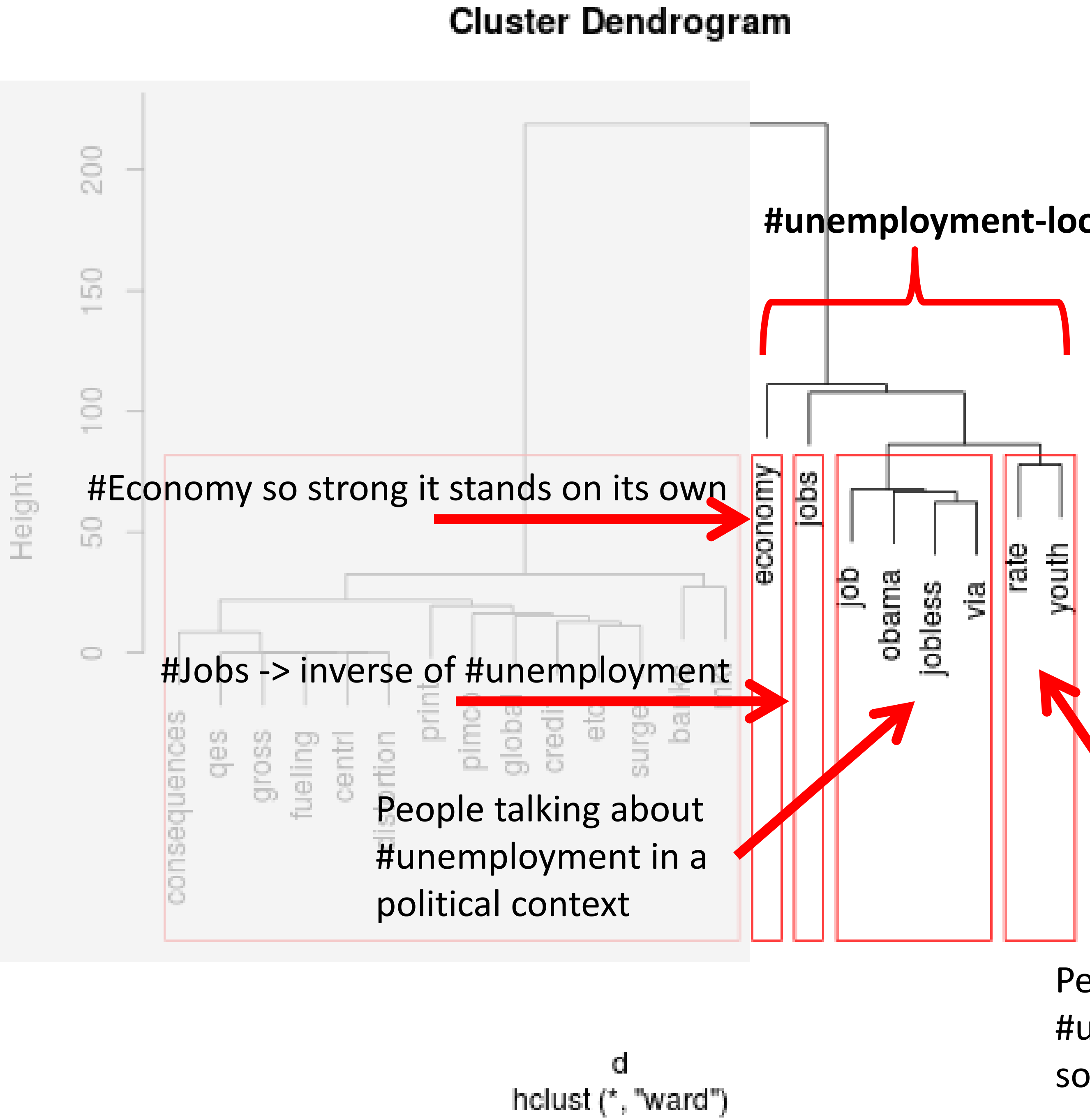


Inflation side

Within Unemployment

- There appear to be 4 categories:
  - Economic
    - “Economy”
  - Employment
    - Just the inverse of the term. Will be highly correlated with the term “unemployment”---people will talk about them together. Could be dropped in bona fide analysis.
  - Political implications
    - “Obama”
    - “Jobless”
  - Social, Sociological implications
    - “Rate”
    - “Youth”

Unemployment side



#unemployment-looking words

#Economy so strong it stands on its own

#Jobs -> inverse of #unemployment

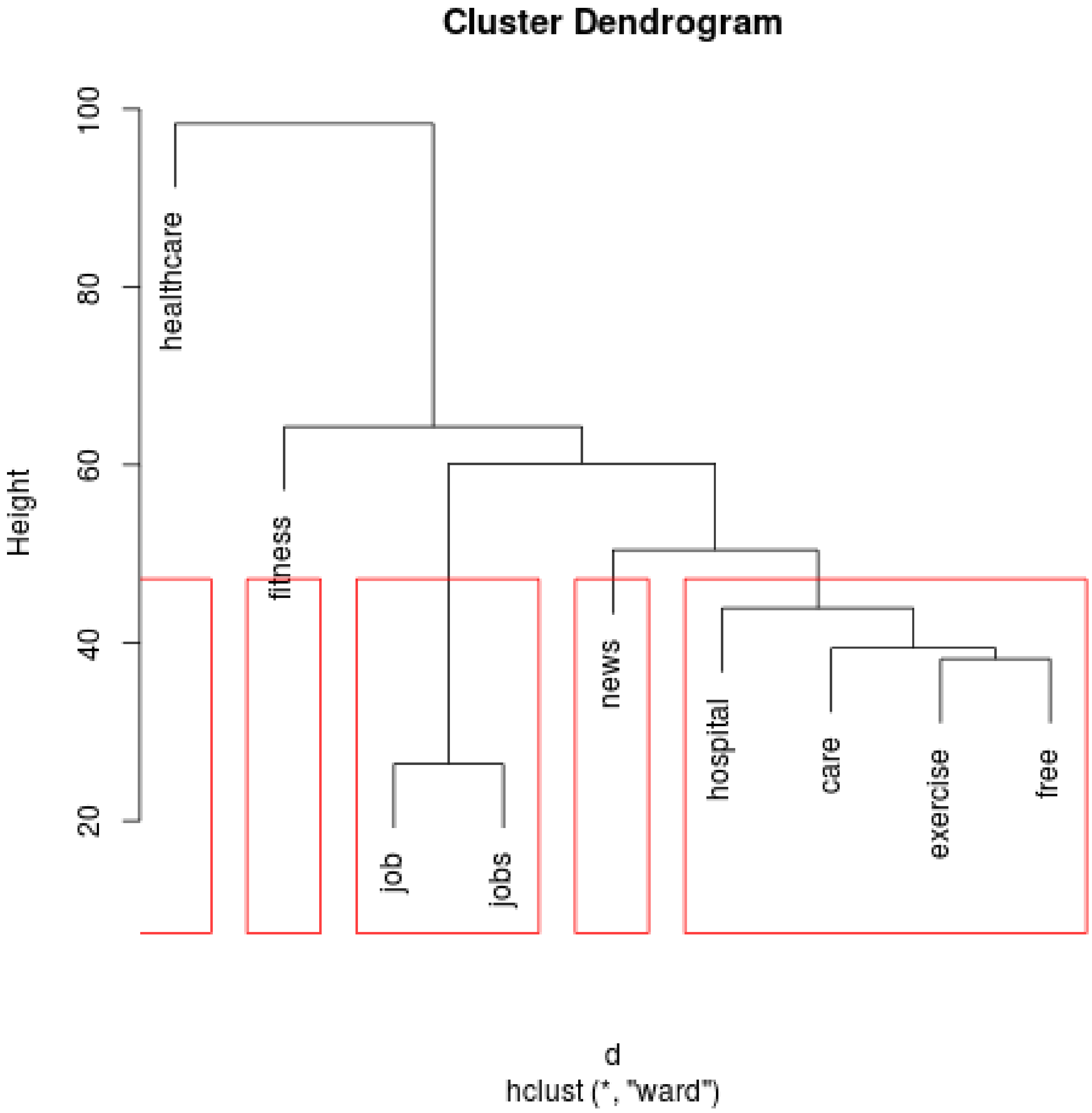
People talking about #unemployment in a political context

People talking about #unemployment in a sociological context

Tweets are dumb

- But plentiful source of correlated text data.
- Applied to medicine, we could see all kinds of different groups.
- This was super cursory, quick and dirty.
- Art and a science
  - You use the algorithm to split the groups which measure “distinct” on a correlation scale, but still need to interpret them.
  - However, there are also good quantitative methods for helping you interpret them...

Here’s one for #Health



Source

- Example adapted from this one, twitter-mining political terms:
    - http://heuristically.wordpress.com/2011/04/08/text-data-mining-twitter-r/