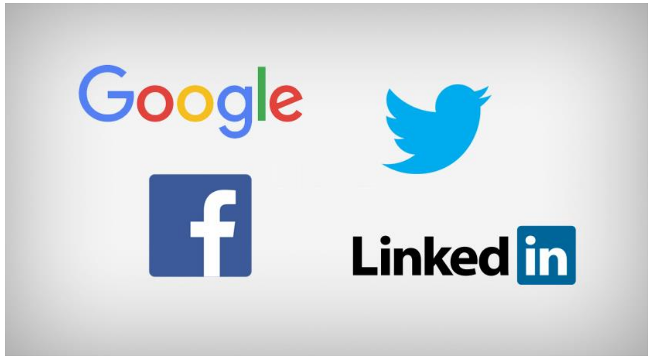
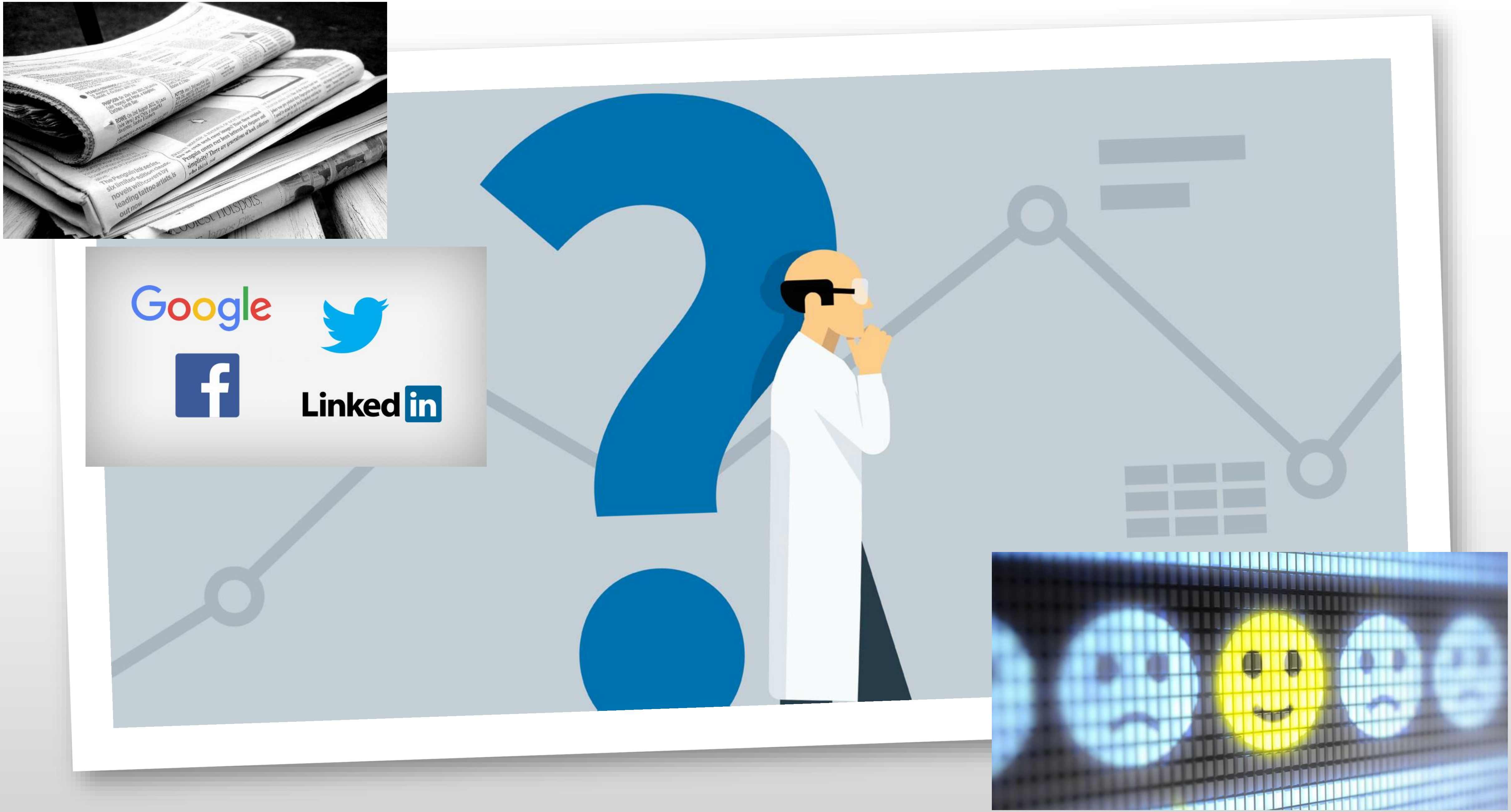


Textual data

Sentiment


Google
Linked in

The R package **sentometrics** to analyze textual sentiment

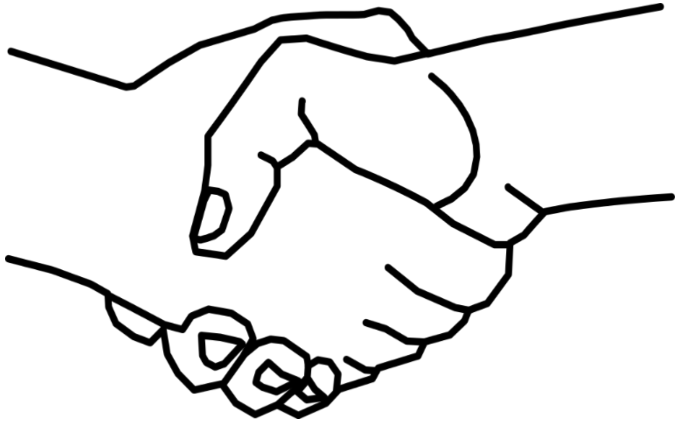
Compute

Aggregate

Predict

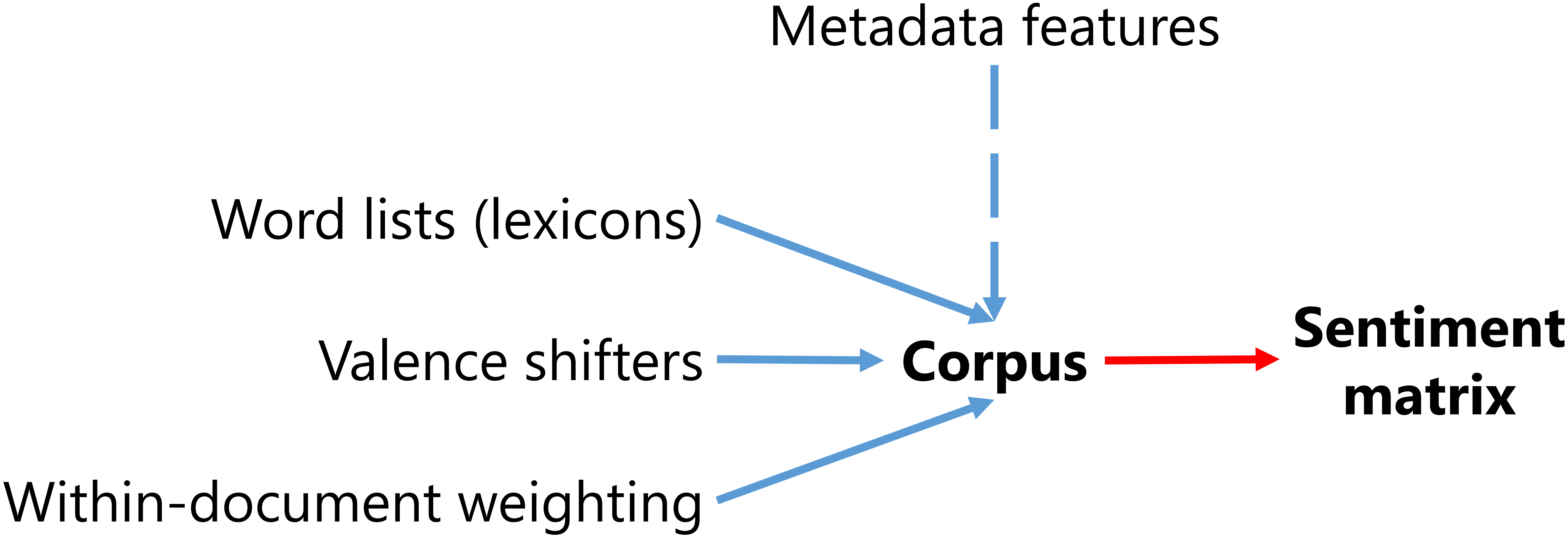
Sentometrics is about **econometrics meeting sentiment**!

sentometrics



Transformation of qualitative data into quantitative sentiment variables

Application in econometric analysis of the relationships between sentiment and other variables



Metadata features

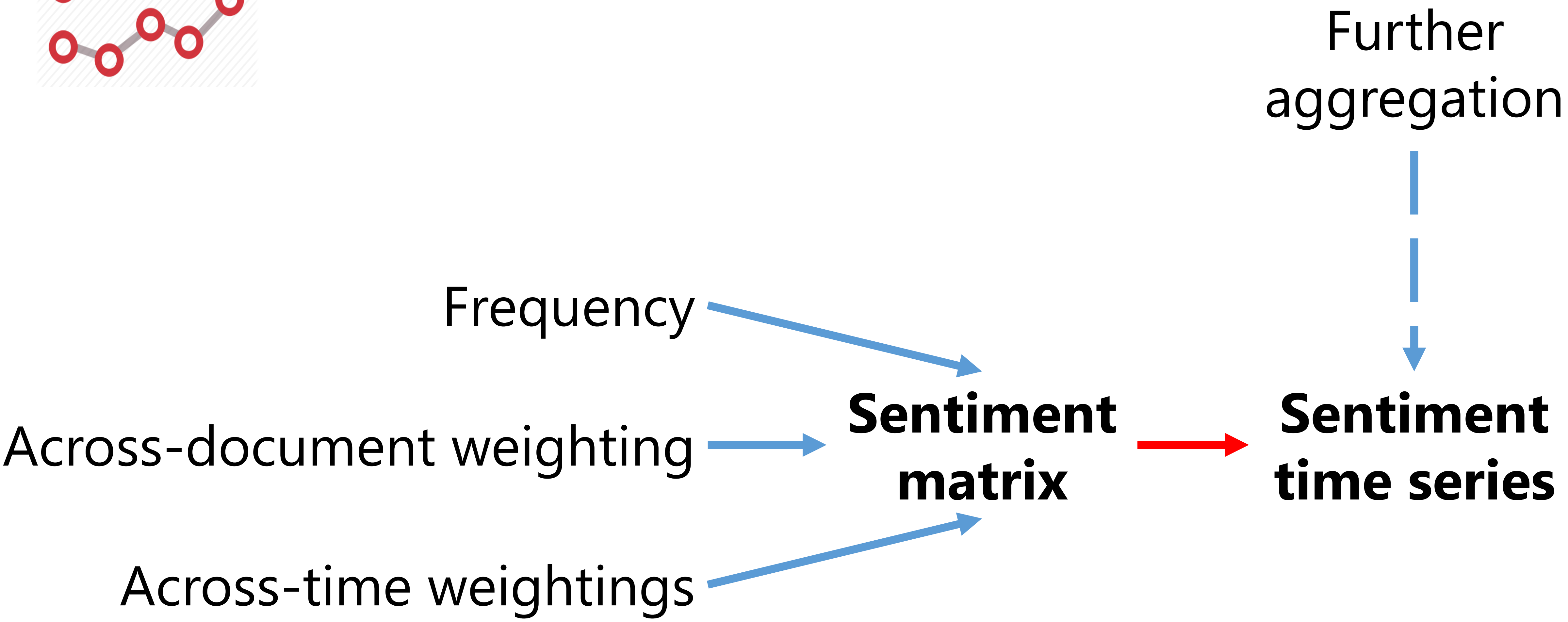
Word lists (lexicons)

Valence shifters

Within-document weighting

Corpus

Sentiment matrix



Further aggregation

Frequency

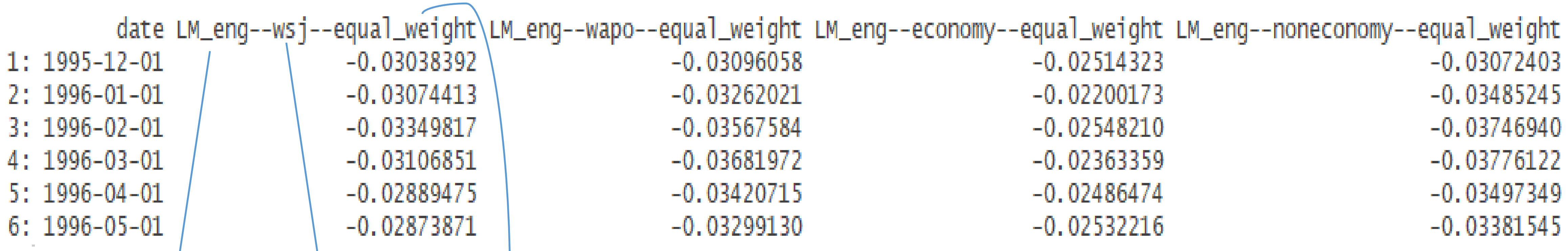
Across-document weighting

Across-time weightings

Sentiment matrix

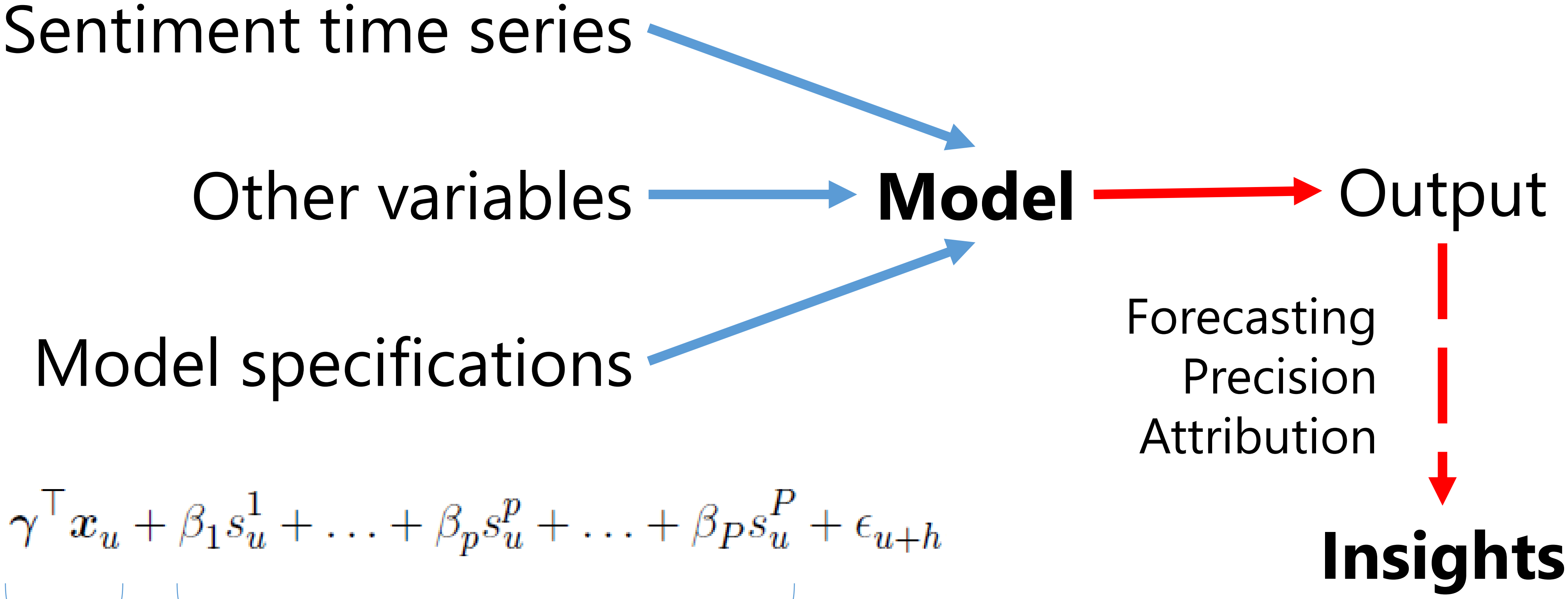
Sentiment time series

Many time series obtained across **lexicons**, **features** and **time aggregation schemes**



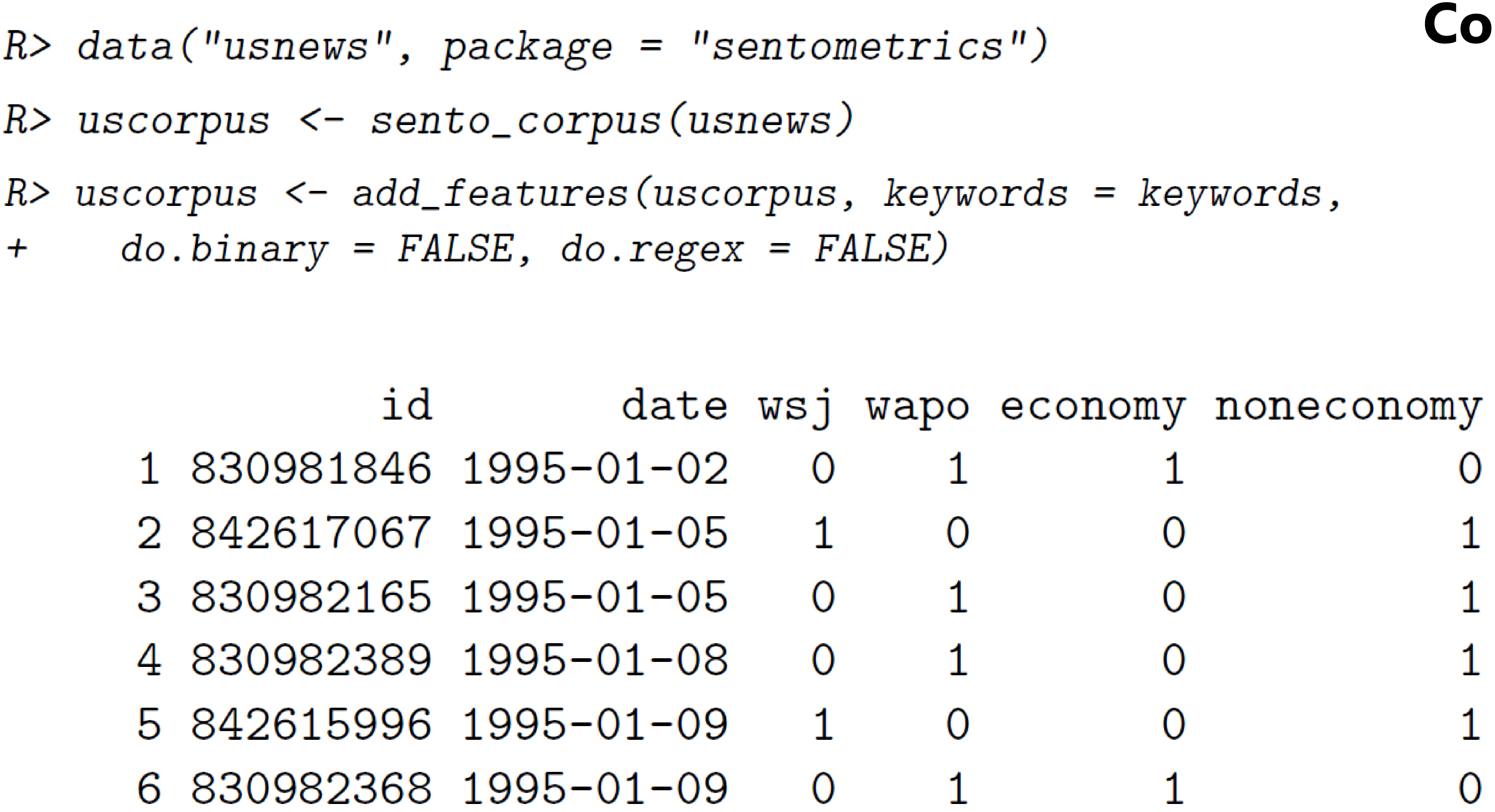
```
         date LM_eng--wsj--equal_weight LM_eng--wapo--equal_weight LM_eng--economy--equal_weight LM_eng--noneconomy--equal_weight
1: 1995-12-01               -0.03038392                -0.03096058                   -0.02514323                      -0.03072403
2: 1996-01-01               -0.03074413                -0.03262021                   -0.02200173                      -0.03485245
3: 1996-02-01               -0.03349817                -0.03567584                   -0.02548210                      -0.03746940
4: 1996-03-01               -0.03106851                -0.03681972                   -0.02363359                      -0.03776122
5: 1996-04-01               -0.02889475                -0.03420715                   -0.02486474                      -0.03497349
6: 1996-05-01               -0.02873871                -0.03299130                   -0.02532216                      -0.03381545
```

lexicon **feature** **time aggregation**



$$\underbrace{y_{u+h}}_{\text{target}} = \delta + \underbrace{\gamma^\top \boldsymbol{x}_u}_{\text{other variables}} + \underbrace{\beta_1 s_u^1 + \ldots + \beta_p s_u^p + \ldots + \beta_P s_u^P}_{\text{sentiment}} + \epsilon_{u+h}$$

Corpus

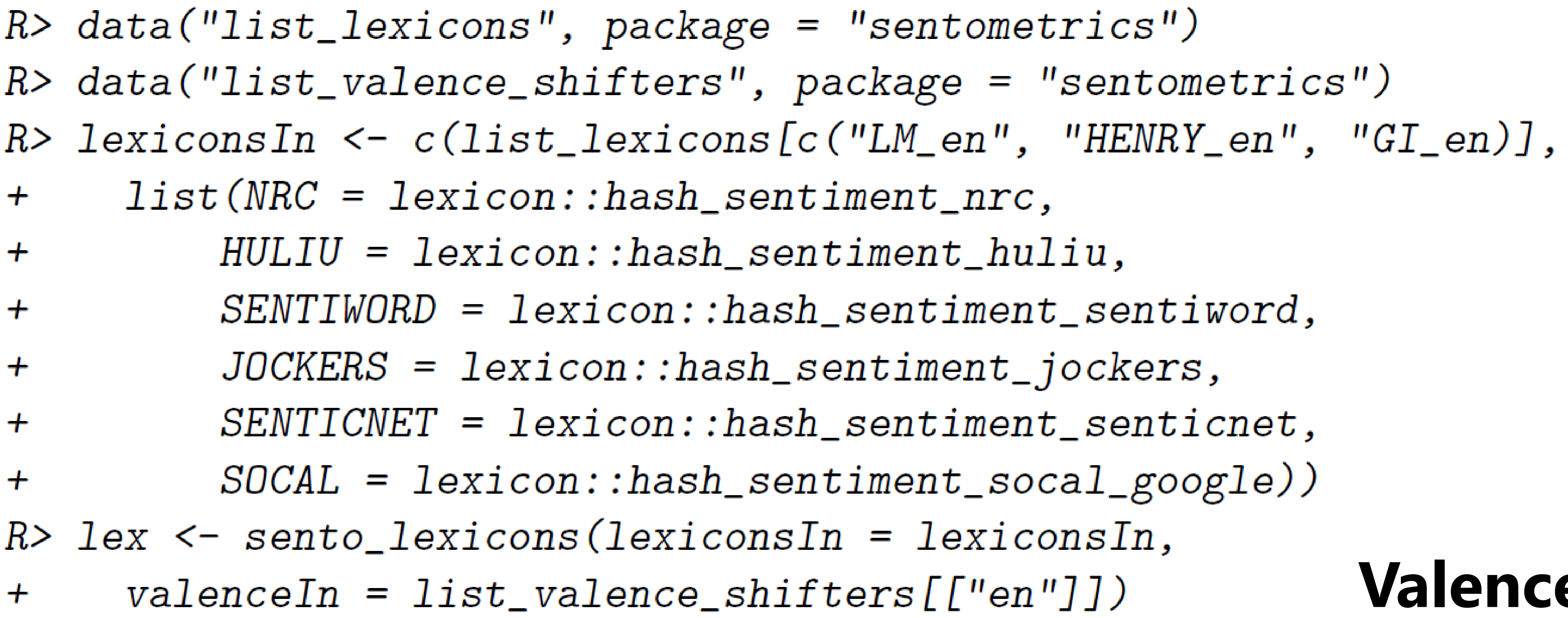


```
R> data("usnews", package = "sentometrics")

R> uscorpus <- sento_corpus(usnews)

R> uscorpus <- add_features(uscorpus, keywords = keywords,
+     do.binary = FALSE, do.regex = FALSE)
```

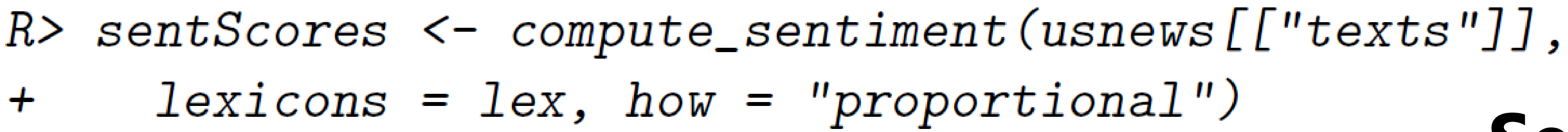
```
         id       date wsj wapo economy noneconomy
1 830981846 1995-01-02   0    1       1          0
2 842617067 1995-01-05   1    0       0          1
3 830982165 1995-01-05   0    1       0          1
4 830982389 1995-01-08   0    1       0          1
5 842615996 1995-01-09   1    0       0          1
6 830982368 1995-01-09   0    1       1          0
```



```
R> data("list_lexicons", package = "sentometrics")
R> data("list_valence_shifters", package = "sentometrics")
R> lexiconsIn <- c(list_lexicons[c("LM_en", "HENRY_en", "GI_en)],
+    list(NRC = lexicon::hash_sentiment_nrc,
+         HULIU = lexicon::hash_sentiment_huliu,
+         SENTIWORD = lexicon::hash_sentiment_sentiword,
+         JOCKERS = lexicon::hash_sentiment_jockers,
+         SENTICNET = lexicon::hash_sentiment_senticnet,
+         SOCAL = lexicon::hash_sentiment_socal_google))
R> lex <- sento_lexicons(lexiconsIn = lexiconsIn,
+    valenceIn = list_valence_shifters[["en"]])
```

Lexicons

Valence shifters



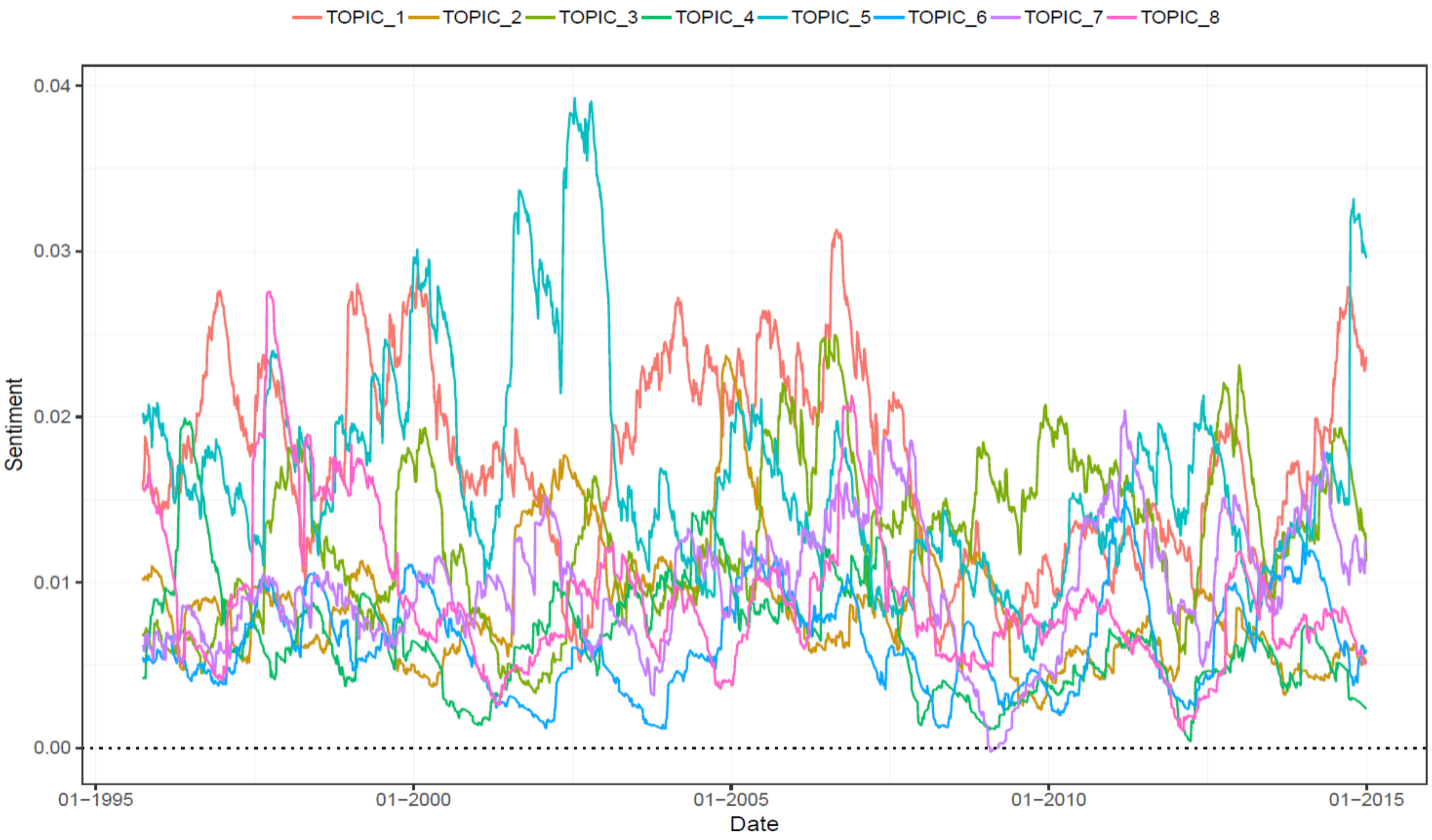
```
R> sentScores <- compute_sentiment(usnews[["texts"]],
+    lexicons = lex, how = "proportional")
```

Sentiment matrix

Weightings Frequency

```
R> ctrAggPred <- ctr_agg(howWithin = "proportionalPol",
+    howDocs = "equal_weight", howTime = "beta",
+    by = "day", fill = "latest", lag = 270, aBeta = 1:3, bBeta = 1:2)
R> sentMeasIn <- sento_measures(uscorpus, lexicons = lex, ctr = ctrAggPred)
```

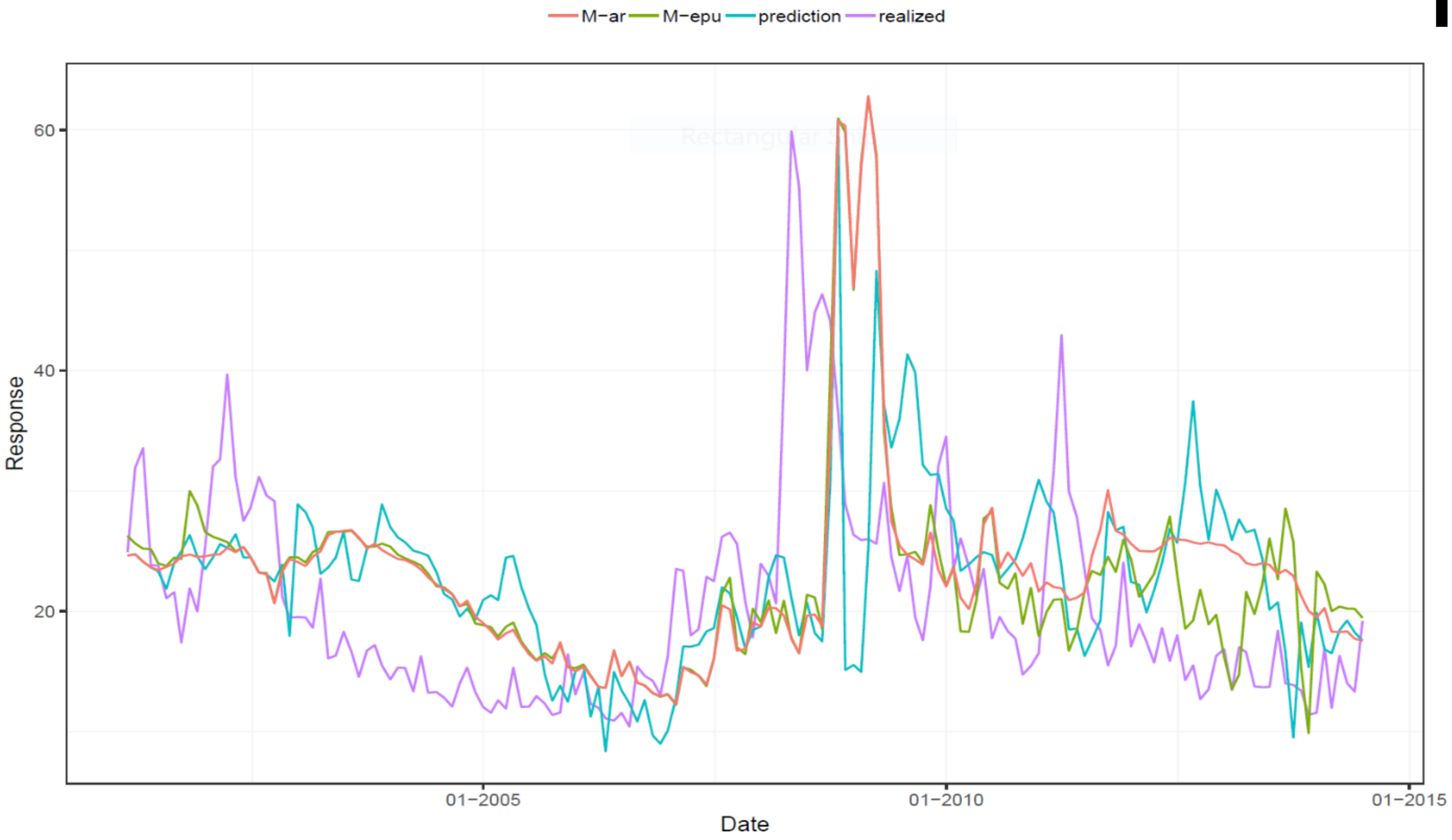
Sentiment time series



Specifications

```
R> ctrIter <- ctr_model(model = "gaussian",
+     type = "BIC", h = h, alphas = c(0, 0.1, 0.3, 0.5, 0.7, 0.9, 1),
+     do.iter = TRUE, oos = oos, nSample = M, nCore = 1)
R> fit <- sento_model(sentMeasIn, x = x[, "lag", drop = FALSE], y = y,
+     ctr = ctrIter)
```

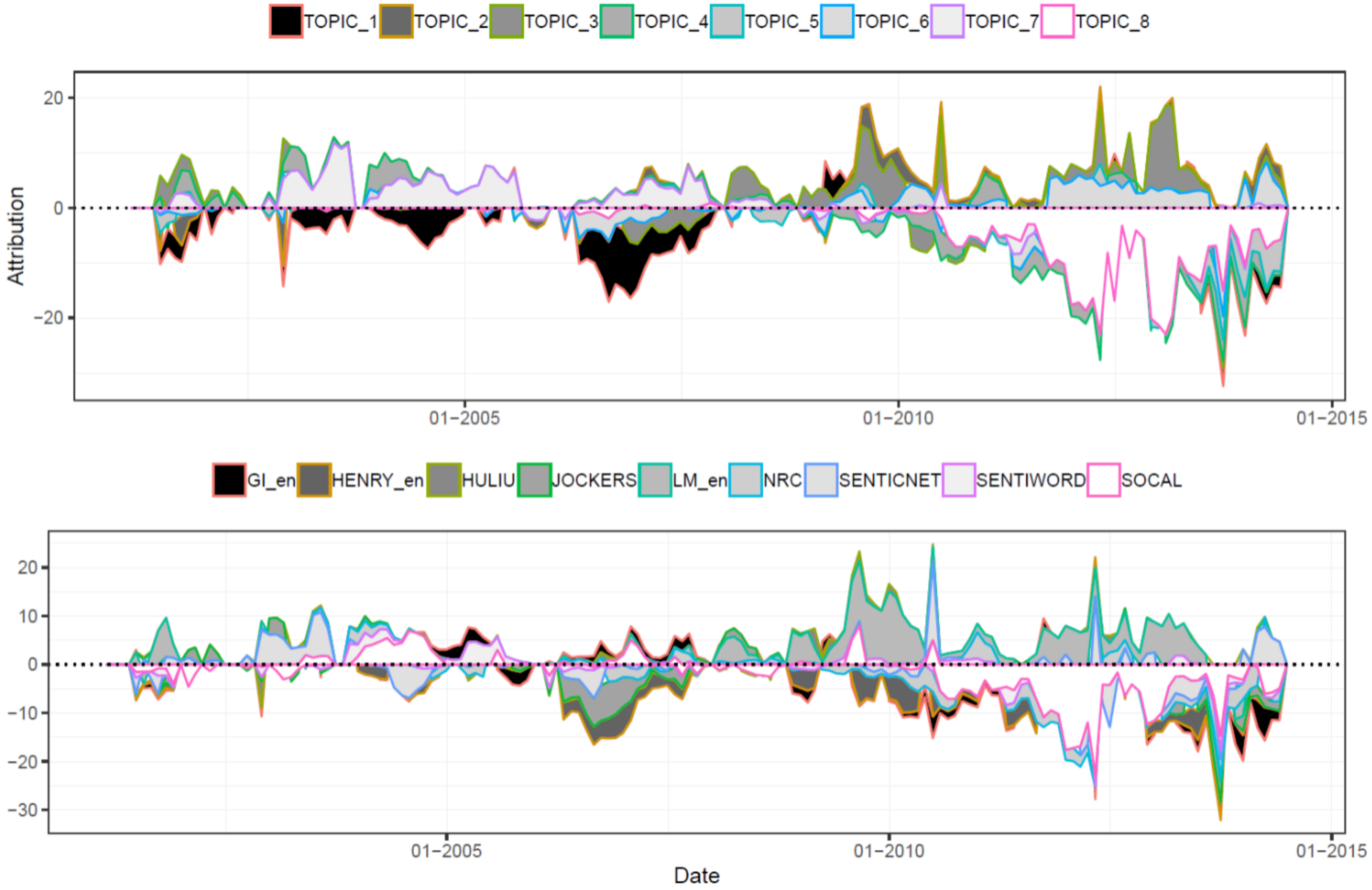
Target variable



Output

Attribution

```
R> attrFit <- attributions(fit, sentMeas)
```



Use the sentometrics R package!

To aggregate **textual data** into

sentiment variables…

… & **insights**

Samuel Borms

samuel.borms@unine.ch

Ardia, Bluteau, Borms and Boudt (2018). **"The R Package *sentometrics* to Compute, Aggregate and Predict with Textual Sentiment"**. Available at: http://dx.doi.org/10.2139/ssrn.3067734.

Repository: https://github.com/sborms/sentometrics.

Project website: https://www.sentometrics.org.