[library](https://rdrr.io/r/base/library.html)("quanteda")

In this vignette we show how the **quanteda** package can be used to replicate the text analysis part (Chapter 5.1) from Kosuke Imai’s book [*Quantitative Social Science: An Introduction*](http://qss.princeton.press) (Princeton: Princeton University Press, 2017).

**Download the Corpus**

To get the textual data, you need to install and load the **qss** package first that comes with the book.

devtools::[install\_github](https://rdrr.io/pkg/devtools/man/reexports.html)("kosukeimai/qss-package", build\_vignettes = TRUE)

**Section 5.1.1: The Disputed Authorship of ‘The Federalist Papers’**

First, we use the **readtext** package to import the Federalist Papers as a data frame and create a **quanteda** corpus.

[library](https://rdrr.io/r/base/library.html)("readtext")

# use readtext package to import all documents as a dataframe

corpus\_texts <- [readtext](https://rdrr.io/pkg/readtext/man/readtext.html)([system.file](https://rdrr.io/r/base/system.file.html)("extdata/federalist/", package = "qss"))

# create docvar with number of paper

corpus\_texts$paper\_number <- [paste](https://rdrr.io/r/base/paste.html)("No.", [seq\_len](https://rdrr.io/r/base/seq.html)([nrow](https://rdrr.io/r/base/nrow.html)(corpus\_texts)), sep = " ")

# transform to a quanteda corpus object

corpus\_raw <- [corpus](https://quanteda.io/reference/corpus.html)(corpus\_texts, text\_field = "text", docid\_field = "paper\_number")

# create docvar with authorship (used in Section 5.1.4)

[docvars](https://quanteda.io/reference/docvars.html)(corpus\_raw, "paper\_numeric") <- [seq\_len](https://rdrr.io/r/base/seq.html)([ndoc](https://quanteda.io/reference/ndoc.html)(corpus\_raw))

# create docvar with authorship (used in Section 5.1.4)

[docvars](https://quanteda.io/reference/docvars.html)(corpus\_raw, "author") <- [factor](https://rdrr.io/r/base/factor.html)(NA, levels = [c](https://rdrr.io/r/base/c.html)("madison", "hamilton"))

[docvars](https://quanteda.io/reference/docvars.html)(corpus\_raw, "author")[[c](https://rdrr.io/r/base/c.html)(1, 6:9, 11:13, 15:17, 21:36, 59:61, 65:85)] <- "hamilton"

[docvars](https://quanteda.io/reference/docvars.html)(corpus\_raw, "author")[[c](https://rdrr.io/r/base/c.html)(10, 14, 37:48, 58)] <- "madison"

# inspect Paper No. 10 (output suppressed)

[texts](https://quanteda.io/reference/texts.html)(corpus\_raw)[10] %>%

stringi::[stri\_sub](https://rdrr.io/pkg/stringi/man/stri_sub.html)(1, 240) %>%

[cat](https://rdrr.io/r/base/cat.html)()

## AMONG the numerous advantages promised by a well-constructed Union, none

## deserves to be more accurately developed than its tendency to break and

## control the violence of faction. The friend of popular governments never

##

**Section 5.1.2: Document-Term Matrix**

Next, we transform the corpus to a document-feature matrix. dfm\_prep (used in sections 5.1.4 and 5.1.5) is a dfm in which numbers and punctuation have been removed, and in which terms have been converted to lowercase. In dfm\_papers, the words have also been stemmed and a standard set of stopwords removed.

# transform corpus to a document-feature matrix

dfm\_prep <- [dfm](https://quanteda.io/reference/dfm.html)(corpus\_raw, remove\_numbers = TRUE, tolower = TRUE,

remove\_punct = TRUE, verbose = TRUE)

## Creating a dfm from a corpus input...

## ... lowercasing

## ... found 85 documents, 8,630 features

## ... created a 85 x 8,630 sparse dfm

## ... complete.

## Elapsed time: 0.376 seconds.

# remove stop words and stem words

dfm\_papers <- [dfm](https://quanteda.io/reference/dfm.html)(dfm\_prep, stem = TRUE, remove = [stopwords](https://rdrr.io/pkg/stopwords/man/stopwords.html)("english"))

# inspect

dfm\_papers

## Document-feature matrix of: 85 documents, 4,859 features (89.1% sparse).

# sort into alphabetical order of features, to match book example

dfm\_papers <- dfm\_papers[, [order](https://rdrr.io/r/base/order.html)([featnames](https://quanteda.io/reference/featnames.html)(dfm\_papers))]

# inspect some documents in the dfm

[head](https://rdrr.io/r/utils/head.html)(dfm\_papers, nf = 8)

## Document-feature matrix of: 6 documents, 8 features (97.9% sparse).

## 6 x 8 sparse Matrix of class "dfm"

## features

## docs 1st 2d 3d 4th 5th abandon abat abb

## No. 1 0 0 0 0 0 0 0 0

## No. 2 0 0 0 0 0 0 0 0

## No. 3 0 0 0 0 0 0 0 0

## No. 4 0 0 0 0 0 0 0 0

## No. 5 1 0 0 0 0 0 0 0

## No. 6 0 0 0 0 0 0 0 0

The **tm** package considers features such as “1st” to be numbers, whereas **quanteda** does not. We can remove these easily using a wildcard removal:

dfm\_papers <- [dfm\_remove](https://quanteda.io/reference/dfm_select.html)(dfm\_papers, "[0-9]", valuetype = "regex", verbose = TRUE)

## removed 5 features

[head](https://rdrr.io/r/utils/head.html)(dfm\_papers, nf = 8)

## Document-feature matrix of: 6 documents, 8 features (91.7% sparse).

## 6 x 8 sparse Matrix of class "dfm"

## features

## docs abandon abat abb abet abhorr abil abject abl

## No. 1 0 0 0 0 0 0 0 1

## No. 2 0 0 0 0 0 1 0 0

## No. 3 0 0 0 0 0 0 0 2

## No. 4 0 0 0 0 0 0 0 1

## No. 5 0 0 0 0 0 0 0 0

## No. 6 0 0 0 0 0 0 0 0

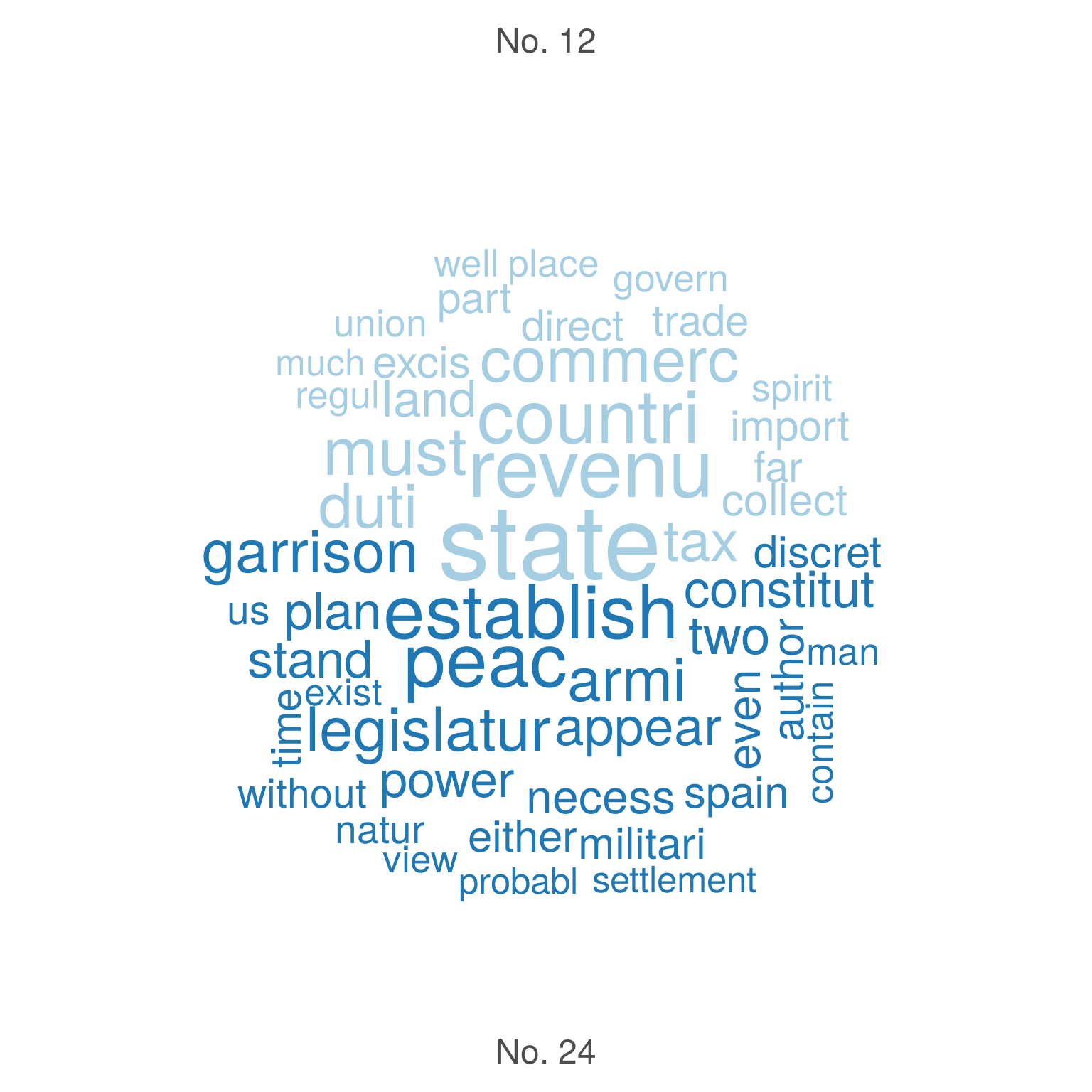
**Section 5.1.3: Topic Discovery**

We can use the [textplot\_wordcloud()](https://quanteda.io/reference/textplot_wordcloud.html) function to plot word clouds of the most frequent words in Papers 12 and 24.

[set.seed](https://rdrr.io/r/base/Random.html)(100)

[textplot\_wordcloud](https://quanteda.io/reference/textplot_wordcloud.html)(dfm\_papers[[c](https://rdrr.io/r/base/c.html)("No. 12", "No. 24"), ],

max.words = 50, comparison = TRUE)



Since **quanteda** cannot do stem completion, we will skip that part.

Next, we identify clusters of similar essay based on term frequency-inverse document frequency (*tf-idf*) and apply the *k*

-means algorithm to the weighted dfm.

# tf-idf calculation

dfm\_papers\_tfidf <- [dfm\_tfidf](https://quanteda.io/reference/dfm_tfidf.html)(dfm\_papers, base = 2)

# 10 most important words for Paper No. 12

[topfeatures](https://quanteda.io/reference/topfeatures.html)(dfm\_papers\_tfidf[12, ], n = 10)

## revenu contraband patrol excis coast trade

## 19.42088 19.22817 19.22817 19.12214 16.22817 15.01500

## per tax cent gallon

## 14.47329 13.20080 12.81878 12.81878

# 10 most important words for Paper No. 24

[topfeatures](https://quanteda.io/reference/topfeatures.html)(dfm\_papers\_tfidf[24, ], n = 10)

## garrison dock-yard settlement spain armi frontier

## 24.524777 19.228173 16.228173 13.637564 12.770999 12.262389

## arsenal western post nearer

## 10.818782 10.806108 10.228173 9.648857

We can match the clustering as follows:

k <- 4 # number of clusters

# subset The Federalist papers written by Hamilton

dfm\_papers\_tfidf\_hamilton <- [dfm\_subset](https://quanteda.io/reference/dfm_subset.html)(dfm\_papers\_tfidf, author == "hamilton")

# run k-means

km\_out <- stats::[kmeans](https://rdrr.io/r/stats/kmeans.html)(dfm\_papers\_tfidf\_hamilton, centers = k)

km\_out$iter # check the convergence; number of iterations may vary

## [1] 2

[colnames](https://rdrr.io/r/base/colnames.html)(km\_out$centers) <- [featnames](https://quanteda.io/reference/featnames.html)(dfm\_papers\_tfidf\_hamilton)

for (i in 1:k) { # loop for each cluster

[cat](https://rdrr.io/r/base/cat.html)("CLUSTER", i, "\n")

[cat](https://rdrr.io/r/base/cat.html)("Top 10 words:\n") # 10 most important terms at the centroid

[print](https://rdrr.io/r/base/print.html)([head](https://rdrr.io/r/utils/head.html)([sort](https://rdrr.io/r/base/sort.html)(km\_out$centers[i, ], decreasing = TRUE), n = 10))

[cat](https://rdrr.io/r/base/cat.html)("\n")

[cat](https://rdrr.io/r/base/cat.html)("Federalist Papers classified: \n") # extract essays classified

[print](https://rdrr.io/r/base/print.html)([docnames](https://quanteda.io/reference/docnames.html)(dfm\_papers\_tfidf\_hamilton)[km\_out$cluster == i])

[cat](https://rdrr.io/r/base/cat.html)("\n")

}

## CLUSTER 1

## Top 10 words:

## armi militia revenu militari war trade taxat upon

## 6.271473 5.204072 4.634529 4.413837 4.349851 4.222969 4.065847 4.049374

## land tax

## 4.007387 3.889522

##

## Federalist Papers classified:

## [1] "No. 6" "No. 7" "No. 8" "No. 11" "No. 12" "No. 15" "No. 21"

## [8] "No. 22" "No. 24" "No. 25" "No. 26" "No. 29" "No. 30" "No. 34"

## [15] "No. 35" "No. 36"

##

## CLUSTER 2

## Top 10 words:

## juri trial court crimin admiralti equiti

## 218.20102 84.74567 62.47940 42.06871 40.87463 38.24428

## chanceri common-law probat civil

## 37.86574 27.04695 27.04695 26.77843

##

## Federalist Papers classified:

## [1] "No. 83"

##

## CLUSTER 3

## Top 10 words:

## court appel jurisdict suprem juri tribun cogniz

## 69.68857 35.27513 25.46591 24.79126 22.16104 21.27125 19.12214

## inferior appeal re-examin

## 18.76875 16.21098 13.52348

##

## Federalist Papers classified:

## [1] "No. 81" "No. 82"

##

## CLUSTER 4

## Top 10 words:

## senat presid claus upon court governor offic appoint

## 5.459987 4.228173 3.530536 3.456783 3.254136 3.134332 3.071945 2.823580

## impeach nomin

## 2.737110 2.658907

##

## Federalist Papers classified:

## [1] "No. 1" "No. 9" "No. 13" "No. 16" "No. 17" "No. 23" "No. 27"

## [8] "No. 28" "No. 31" "No. 32" "No. 33" "No. 59" "No. 60" "No. 61"

## [15] "No. 65" "No. 66" "No. 67" "No. 68" "No. 69" "No. 70" "No. 71"

## [22] "No. 72" "No. 73" "No. 74" "No. 75" "No. 76" "No. 77" "No. 78"

## [29] "No. 79" "No. 80" "No. 84" "No. 85"

**Section 5.1.4: Authorship Prediction**

In a next step, we want to predict authorship for the Federalist Papers whose authorship is unknown. As the topics of the Papers differs remarkably, Imai focuses on 10 articles, prepositions and conjunctions to predict authorship.

# term frequency per 1000 words

tfm <- [dfm\_weight](https://quanteda.io/reference/dfm_weight.html)(dfm\_prep, "prop") \* 1000

# select words of interest

words <- [c](https://rdrr.io/r/base/c.html)("although", "always", "commonly", "consequently",

"considerable", "enough", "there", "upon", "while", "whilst")

tfm <- [dfm\_select](https://quanteda.io/reference/dfm_select.html)(tfm, words, valuetype = "fixed")

# average among Hamilton/Madison essays

tfm\_ave <- [dfm\_group](https://quanteda.io/reference/dfm_group.html)([dfm\_subset](https://quanteda.io/reference/dfm_subset.html)(tfm, ![is.na](https://rdrr.io/r/base/NA.html)(author)), "author") /

[as.numeric](https://rdrr.io/r/base/numeric.html)([table](https://rdrr.io/r/base/table.html)([docvars](https://quanteda.io/reference/docvars.html)(tfm, "author")))

# bind docvars from corpus and tfm to a data frame

author\_data <- [data.frame](https://rdrr.io/r/base/data.frame.html)([docvars](https://quanteda.io/reference/docvars.html)(corpus\_raw), tfm)

## Warning: 'as.data.frame.dfm' is deprecated.

## Use 'convert(x, to = "data.frame")' instead.

## See help("Deprecated")

# create numeric variable that takes value 1 for Hamilton's essays,

# -1 for Madison's essays and NA for the essays with unknown authorship

author\_data$author\_numeric <- [ifelse](https://rdrr.io/r/base/ifelse.html)(author\_data$author == "hamilton", 1,

[ifelse](https://rdrr.io/r/base/ifelse.html)(author\_data$author == "madison", -1, NA))

# use only known authors for training set

author\_data\_known <- [na.omit](https://rdrr.io/r/stats/na.fail.html)(author\_data)

hm\_fit <- [lm](https://rdrr.io/r/stats/lm.html)(author\_numeric ~ upon + there + consequently + whilst,

data = author\_data\_known)

hm\_fit

##

## Call:

## lm(formula = author\_numeric ~ upon + there + consequently + whilst,

## data = author\_data\_known)

##

## Coefficients:

## (Intercept) upon there consequently whilst

## -0.2729 0.2240 0.1278 -0.5596 -0.8378

hm\_fitted <- [fitted](https://rdrr.io/r/stats/fitted.values.html)(hm\_fit) # fitted values

[sd](https://rdrr.io/r/stats/sd.html)(hm\_fitted)

## [1] 0.7175527

**Section 5.1.5: Cross-Validation**

Finally, we assess how well the model fits the data by classifying each essay based on its fitted value.

# proportion of correctly classified essays by Hamilton

[mean](https://rdrr.io/r/base/mean.html)(hm\_fitted[author\_data\_known$author == "hamilton"] > 0)

## [1] 1

# proportion of correctly classified essays by Madison

[mean](https://rdrr.io/r/base/mean.html)(hm\_fitted[author\_data\_known$author == "madison"] < 0)

## [1] 1

n <- [nrow](https://rdrr.io/r/base/nrow.html)(author\_data\_known)

hm\_classify <- [rep](https://rdrr.io/r/base/rep.html)(NA, n) # a container vector with missing values

for (i in 1:n) {

# fit the model to the data after removing the ith observation

sub\_fit <- [lm](https://rdrr.io/r/stats/lm.html)(author\_numeric ~ upon + there + consequently + whilst,

data = author\_data\_known[-i, ]) # exclude ith row

# predict the authorship for the ith observation

hm\_classify[i] <- [predict](https://rdrr.io/r/stats/predict.html)(sub\_fit, newdata = author\_data\_known[i, ])

}

# proportion of correctly classified essays by Hamilton

[mean](https://rdrr.io/r/base/mean.html)(hm\_classify[author\_data\_known$author == "hamilton"] > 0)

## [1] 1

# proportion of correctly classified essays by Madison

[mean](https://rdrr.io/r/base/mean.html)(hm\_classify[author\_data\_known$author == "madison"] < 0)

## [1] 1

disputed <- [c](https://rdrr.io/r/base/c.html)(49, 50:57, 62, 63) # 11 essays with disputed authorship

tf\_disputed <- [dfm\_subset](https://quanteda.io/reference/dfm_subset.html)(tfm, [is.na](https://rdrr.io/r/base/NA.html)(author)) %>%

[convert](https://quanteda.io/reference/convert.html)(to = "data.frame")

author\_data$prediction <- [predict](https://rdrr.io/r/stats/predict.html)(hm\_fit, newdata = author\_data)

author\_data$prediction # predicted values

## [1] 0.73682138 -0.13810334 -1.35072048 -0.03823549 -0.27285552

## [6] 0.71537231 1.33178340 0.19414036 0.37516964 -0.20369832

## [11] 0.67584796 0.99478176 1.39348825 -1.18908379 1.20446180

## [16] 0.63999272 0.91390560 -0.38341254 -0.62471327 -0.12453888

## [21] 0.91552544 1.08097849 0.88528696 1.01192444 0.08226598

## [26] 0.72338300 1.08177034 0.71354396 1.47718825 1.53672871

## [31] 1.85572895 0.71920518 1.07664097 1.26163462 0.90800379

## [36] 1.06339810 -0.81826934 -0.56126822 -0.27285552 -0.10309170

## [41] -0.47343432 -0.17492007 -0.60654100 -0.86944061 -1.67765248

## [46] -0.91487748 -0.13268399 -0.13500254 -0.96784742 -0.06907196

## [51] -1.46790700 -0.27285552 -0.54229524 -0.54529610 0.04081584

## [56] -0.56418213 -1.18177840 -0.41902525 0.55200092 0.98676075

## [61] 0.59237415 -0.97795312 -0.21203157 -0.18295716 1.15640459

## [66] 1.12403237 0.78348571 0.11197467 1.12824159 0.70981781

## [71] 0.34751104 0.84257351 1.24182206 0.91612261 0.50276872

## [76] 1.05480578 1.05926761 0.90585578 0.54292995 0.64035410

## [81] 0.91773469 0.31189155 0.80869221 0.73649069 1.00895939

Finally, we plot the fitted values for each Federalist paper with the **ggplot2** package.

author\_data$author\_plot <- [ifelse](https://rdrr.io/r/base/ifelse.html)([is.na](https://rdrr.io/r/base/NA.html)(author\_data$author), "unknown", [as.character](https://rdrr.io/r/base/character.html)(author\_data$author))

[library](https://rdrr.io/r/base/library.html)(ggplot2)

[ggplot](https://ggplot2.tidyverse.org/reference/ggplot.html)(data = author\_data, [aes](https://ggplot2.tidyverse.org/reference/aes.html)(x = paper\_numeric,

y = prediction,

shape = author\_plot,

colour = author\_plot)) +

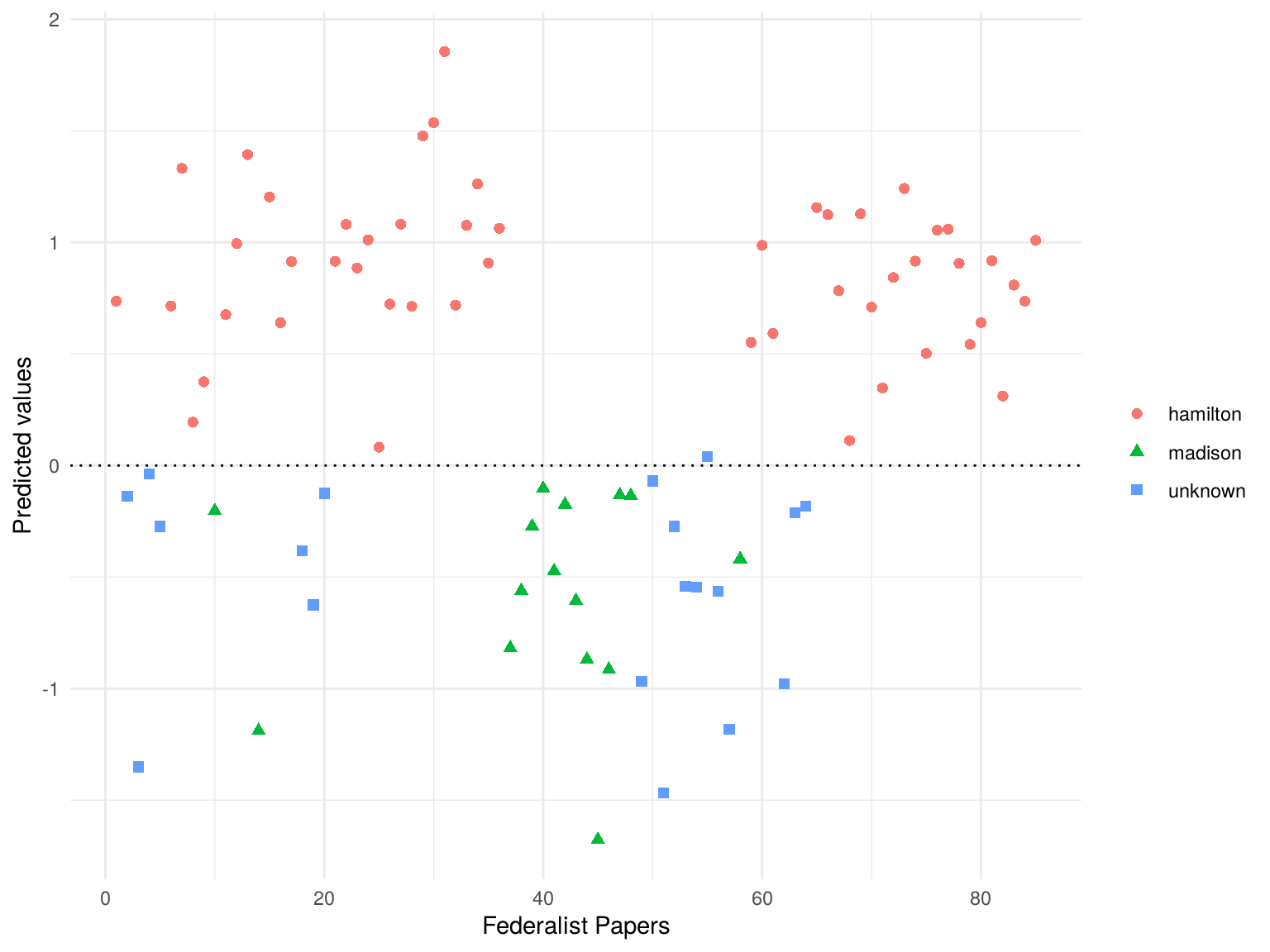
[geom\_point](https://ggplot2.tidyverse.org/reference/geom_point.html)(size = 2) +

[geom\_hline](https://ggplot2.tidyverse.org/reference/geom_abline.html)(yintercept = 0, linetype = "dotted") +

[labs](https://ggplot2.tidyverse.org/reference/labs.html)(x = "Federalist Papers", y = "Predicted values") +

[theme\_minimal](https://ggplot2.tidyverse.org/reference/ggtheme.html)() +

[theme](https://ggplot2.tidyverse.org/reference/theme.html)(legend.title=[element\_blank](https://ggplot2.tidyverse.org/reference/element.html)())



**Contents**

* [Download the Corpus](https://quanteda.io/articles/pkgdown/replication/qss.html#download-the-corpus)
* [Section 5.1.1: The Disputed Authorship of ‘The Federalist Papers’](https://quanteda.io/articles/pkgdown/replication/qss.html#section-5-1-1-the-disputed-authorship-of-the-federalist-papers)
* [Section 5.1.2: Document-Term Matrix](https://quanteda.io/articles/pkgdown/replication/qss.html#section-5-1-2-document-term-matrix)
* [Section 5.1.3: Topic Discovery](https://quanteda.io/articles/pkgdown/replication/qss.html#section-5-1-3-topic-discovery)
* [Section 5.1.4: Authorship Prediction](https://quanteda.io/articles/pkgdown/replication/qss.html#section-5-1-4-authorship-prediction)
* [Section 5.1.5: Cross-Validation](https://quanteda.io/articles/pkgdown/replication/qss.html#section-5-1-5-cross-validation)

Developed by Kenneth Benoit, Kohei Watanabe, Haiyan Wang, Paul Nulty, Adam Obeng, Stefan Müller, Akitaka Matsuo, Jiong Wei Lua, Patrick O. Perry, Jouni Kuha, Benjamin Lauderdale, William Lowe, European Research Council.