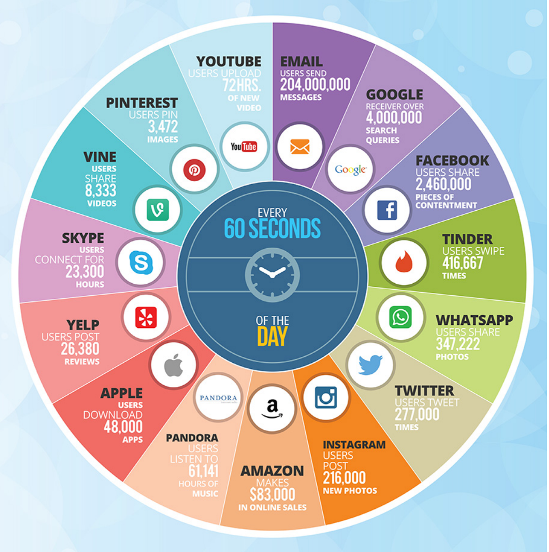
# SMI 606: Discovery Using Text As Data

Dr. Todd Hartman
Sheffield Methods Institute

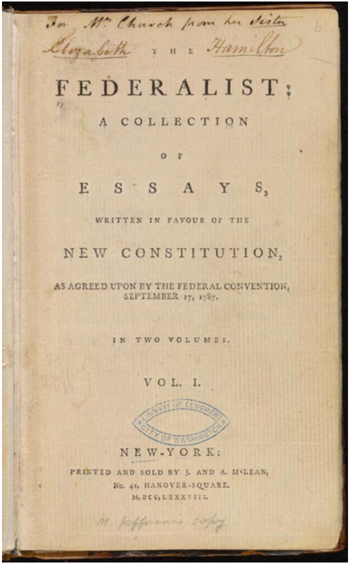
# The Digital Revolution



# Text As Data

- The digital revolution has created huge troves of textual data
  - Digitized books, articles, plays, etc.
  - Film, television, and radio transcripts
  - News media coverage
  - Speeches, press releases, and other forms of direct communication
  - Social media (Facebook, Twitter, blogs, etc.)

- 85 essays written by Alexander Hamilton, John Jay, and James Madison
- Encouraged Americans to ratify the newly drafted US Constitution
- Foundational document (reveals the 'framers' intentions)
- Long-standing debate about who authored 11 of the essays



For Mrs Church from her Sister Elizabeth Hamilton

THE FEDERALIST:

A COLLECTION OF ESSAYS,

WRITTEN IN FAVOUR OF THE NEW CONSTITUTION,

AS AGREED UPON BY THE FEDERAL CONVENTION, SEPTEMBER 17, 1787.

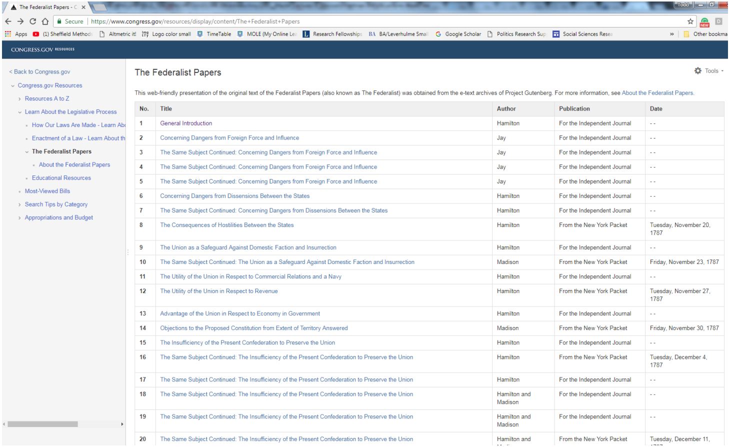
IN TWO VOLUMES.

VOL. I.

NEW-YORK:

PRINTED AND SOLD BY J. AND A. M'LEAN, No. 41, HANOVER-SQUARE. M,DCC,LXXXVIII.

*Web scraping* involves using a program – in our case, *R* – to automatically collect data from websites



CONGRESS.GOV RESOURCES

< Back to Congress.gov
- Congress.gov Resources
  - Resources A to Z
  - Learn About the Legislative Process
    - How Our Laws Are Made - Learn Ab
    - Enactment of a Law - Learn About th
    - The Federalist Papers
      - About the Federalist Papers
    - Educational Resources
  - Most-Viewed Bills
  - Search Tips by Category
  - Appropriations and Budget

The Federalist Papers

Tools

This web-friendly presentation of the original text of the Federalist Papers (also known as The Federalist) was obtained from the e-text archives of Project Gutenberg. For more information, see About the Federalist Papers.

| No. | Title | Author | Publication | Date |
|---|---|---|---|---|
| 1 | General Introduction | Hamilton | For the Independent Journal | - - |
| 2 | Concerning Dangers from Foreign Force and Influence | Jay | For the Independent Journal | - - |
| 3 | The Same Subject Continued: Concerning Dangers from Foreign Force and Influence | Jay | For the Independent Journal | - - |
| 4 | The Same Subject Continued: Concerning Dangers from Foreign Force and Influence | Jay | For the Independent Journal | - - |
| 5 | The Same Subject Continued: Concerning Dangers from Foreign Force and Influence | Jay | For the Independent Journal | - - |
| 6 | Concerning Dangers from Dissensions Between the States | Hamilton | For the Independent Journal | - - |
| 7 | The Same Subject Continued: Concerning Dangers from Dissensions Between the States | Hamilton | For the Independent Journal | - - |
| 8 | The Consequences of Hostilities Between the States | Hamilton | From the New York Packet | Tuesday, November 20, 1787 |
| 9 | The Union as a Safeguard Against Domestic Faction and Insurrection | Hamilton | For the Independent Journal | - - |
| 10 | The Same Subject Continued: The Union as a Safeguard Against Domestic Faction and Insurrection | Madison | From the New York Packet | Friday, November 23, 1787 |
| 11 | The Utility of the Union in Respect to Commercial Relations and a Navy | Hamilton | For the Independent Journal | - - |
| 12 | The Utility of the Union in Respect to Revenue | Hamilton | From the New York Packet | Tuesday, November 27, 1787 |
| 13 | Advantage of the Union in Respect to Economy in Government | Hamilton | For the Independent Journal | - - |
| 14 | Objections to the Proposed Constitution from Extent of Territory Answered | Madison | From the New York Packet | Friday, November 30, 1787 |
| 15 | The Insufficiency of the Present Confederation to Preserve the Union | Hamilton | For the Independent Journal | - - |
| 16 | The Same Subject Continued: The Insufficiency of the Present Confederation to Preserve the Union | Hamilton | From the New York Packet | Tuesday, December 4, 1787 |
| 17 | The Same Subject Continued: The Insufficiency of the Present Confederation to Preserve the Union | Hamilton | For the Independent Journal | - - |
| 18 | The Same Subject Continued: The Insufficiency of the Present Confederation to Preserve the Union | Hamilton and Madison | For the Independent Journal | - - |
| 19 | The Same Subject Continued: The Insufficiency of the Present Confederation to Preserve the Union | Hamilton and Madison | For the Independent Journal | - - |
| 20 | The Same Subject Continued: The Insufficiency of the Present Confederation to Preserve the Union | Hamilton and | From the New York Packet | Tuesday, December 11, |

# Loading Text Data in R

Load the raw corpus

```
## 'federalist' is the folder in the working directory, pattern 'fp' is the structure of the
 file names
corpus.raw <- Corpus(DirSource(directory = "federalist", pattern = "fp"))

## Check the content of the Federalist Paper No. 10
head(content(corpus.raw[[10]]))
```

```
## [1] "AMONG the numerous advantages promised by a well-constructed Union, none "
## [2] "        deserves to be more accurately developed than its tendency to break and "
## [3] "        control the violence of faction. The friend of popular governments never "
## [4] "        finds himself so much alarmed for their character and fate, as when he "
## [5] "        contemplates their propensity to this dangerous vice. He will not fail, "
## [6] "        therefore, to set a due value on any plan which, without violating the "
```

# Natural Language Processing (NLP)

Text data usually requires a lot of preprocessing before it can be properly analyzed; for example:

- Transform to lower case
- Eliminate extra white space between words
- Remove prefixes and suffixes (a.k.a. stemming)
- Remove unwanted punctuation, characters, and/or words
- Remove 'stop' words (e.g., 'a', 'the', etc.)

Process the Federalist Papers data for analysis

```r
## Make all of the textual data lower case
corpus.prep <- tm_map(corpus.raw, content_transformer(tolower))

## Remove whitespace in the data
corpus.prep <- tm_map(corpus.prep, stripWhitespace)

## Remove punctuation
corpus.prep <- tm_map(corpus.prep, removePunctuation)

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

## Remove stop words
head(stopwords("english"))
```

```
## [1] "i"      "me"     "my"     "myself" "we"     "our"
```

# Removing 'Stop' Words

```r
corpus <- tm_map(corpus.prep, removeWords, stopwords("english"))

## Reduce words to their root form
corpus <- tm_map(corpus, stemDocument)

## Check the content of the processed Federalist Paper No. 10
head(content(corpus[[10]]))
```

```
## [1] "among  numer advantag promis   wellconstruct union none"
## [2] " deserv    accur develop   tendenc  break "
## [3] " control  violenc  faction  friend  popular govern never"
## [4] " find   much alarm   charact  fate     "
## [5] " contempl  propens   danger vice  will  fail"
## [6] " therefor  set  due valu   plan  without violat "
```

# Who Wrote *The Federalist Papers?*

Who wrote the Federalist Papers supporting the passage of The Constitution?

- Publius **3 + ... 11?**
- James Madison **15**
- John Jay **5**
- Alexander Hamilton **51**

# Forensic linguistics

- Authorship identification is a task in *forensic linguistics*
- To determine authorship (ideally) we need a source of textual information that is
  - unrelated to the *topic* of a text
  - difficult to *be strategic with*
  - *idiosyncratic*

  i.e., a sample of writing *style*

# Writing Sample 1

Hamilton on the militia (29)

> *the power of regulating the militia and of commanding its services in times of insurrection and invasion are natural incidents to the duties of superintending the common defense and of watching over the internal peace of the confederacy it requires no skill in the science of war to discern that uniformity[...]*

# Writing Sample 2

Madison on the judiciary (49)

> *The several departments being perfectly co-ordinate by the terms of their common commission, none of them, it is evident, can pretend to an exclusive or superior right of settling the boundaries between their respective powers[. . . ]*

# Term Frequency (TF) - 'Bag of Words' Analysis

Count of each word in the corpus

- Document-Term Matrix (rows = documents; cols = words)
- Term-Document Matrix (rows = words; cols = documents)

```
dtm1 <- as.matrix(dtm)
dtm1[1:4,10:14]

          Terms
Docs        address admit adopt adoption advantages
  fp01.txt        1     1     1        2          1
  fp02.txt        0     1     2        0          0
  fp03.txt        0     0     1        0          0
  fp04.txt        1     1     0        0          2
```

# Document-Term Matrix (DTM)

Sparsity is the proportion of zero entries in the dtm

- Most documents are sparse (i.e., most terms only appear in a small number of documents)
- In *The Federalist Papers*, 89% of the elements of the dtm are 0

Create a Document-Term Matrix with docs on the rows and terms on the cols

```
dtm <- DocumentTermMatrix(corpus)

## Summary information about the D-T Matrix
dtm
```

```
## <<DocumentTermMatrix (documents: 85, terms: 4849)>>
## Non-/sparse entries: 44917/367248
## Sparsity           : 89%
## Maximal term length: 18
## Weighting          : term frequency (tf)
```

# Classification: Hamilton or Madison?

- *Identify the dependent variable*
- *Find predictors*
- Fit the model
- Evaluate the model

With text analysis, the hardest steps are often the first two

- Turn the text into something we can use a regression model to do prediction with

Let's start with the dependent variable. . .

# Classification: Outcome

```r
hamilton <- c(1, 6:9, 11:13, 15:17, 21:36, 59:61, 65:85)
madison <- c(10, 14, 37:48, 58)
hamilton.madison <- 18:20
jay <- c(2:5, 64)
contested <- c(49:57, 62, 63)

author <- rep(NA, 85)
author[hamilton] <- 'hamilton'
author[madison] <- 'madison'
author[hamilton.madison] <- 'hamilton.madison'
author[jay] <- 'jay'
author[contested] <- 'contested'
```

# Classification: Outcome to predict

```
table(author)

author
       contested         hamilton hamilton.madison
              11               51                3
             jay          madison
               5               15
```

# Classification: Raw materials for predictors

```
require(tm)

Loading required package:  tm
Loading required package:  NLP

corpus.raw <- Corpus(DirSource(directory='data/federalist',
                               pattern='fp'))
corpus <- tm_map(corpus.raw, content_transformer(tolower))
corpus <- tm_map(corpus, stripWhitespace)
corpus <- tm_map(corpus, removePunctuation)
corpus <- tm_map(corpus, removeNumbers)
dtm <- DocumentTermMatrix(corpus)
```

# Classification: The 'bag of words'

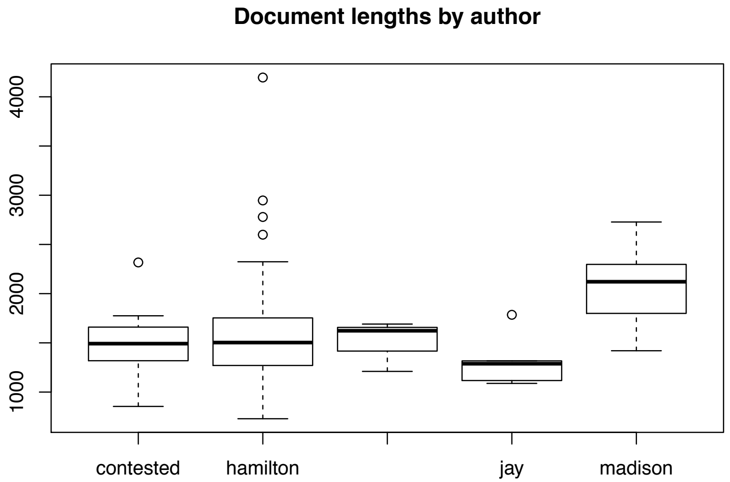
```
dtm1 <- as.matrix(dtm)
dtm1[1:4,10:14]

          Terms
Docs        address admit adopt adoption advantages
  fp01.txt        1     1     1        2          1
  fp02.txt        0     1     2        0          0
  fp03.txt        0     0     1        0          0
  fp04.txt        1     1     0        0          2
```

But raw word counts are not reliable predictors

- Operationalization: We should also *transform* them...

# Classification: Predictors



# Classification: Predictors

Turn word counts into *word rates*

- proportion = word count / document length
- rate = proportion × 1000

```
dtm1 <- dtm1 / rowSums(dtm1) * 1000
```

# Classification: Predictors

How many possible predictors?

```
ncol(dtm1)

[1] 8594
```

Too many! We'll use just

```
words <- c('although', 'always', 'commonly', 'consequently',
           'considerable', 'enough', 'there', 'upon',
           'while', 'whilst')
```

# Classification: Dependent variable

Our dependent variable is the identity of the author (if we know it)

```
table(author)

author
       contested         hamilton hamilton.madison
              11               51                3
             jay          madison
               5               15
```

# Classification: Data

Organise the data

```
dat <- data.frame(number=1:nrow(dtm1),
                  author=author,
                  dtm1[, words])
```

# Classification: Training

We will distinguish a **training set** of observations where the author is known

- This is the **sample**
- Predictions are **in-sample** (we'll call them *fitted values*)

Focus on distinguishing Hamilton from Madison

```
train.dat <- subset(dat, author=='madison' | author=='hamilton')
```

We need a *numerical representation* of this distinction

```
train.dat$score <- ifelse(train.dat$author=='hamilton', 1, -1)
```

# Classification: Test

And a **test set** where author is unknown.

- Predictions are **out-of-sample**

```
test.dat <- subset(dat, author=='contested')
```

# Classification: Fit model

We use 4 of the 8594 possible word rates to predict `score`

```
hm.mod <- lm(score ~ upon + there + consequently + whilst,
             data=train.dat)
coef(hm.mod)

 (Intercept)          upon         there  consequently
 -0.26288400    0.16677891    0.09493894   -0.44012341
      whilst
 -0.65875088
```

# Classification: Evaluation (in sample)

Remember we have distinguished `score` from `author`.

- Compare: distinguishing predicting a vote margin from winning a state

```
pred.authors <- fitted(hm.mod) > 0
table(pred.authors, train.dat$score)
```

```
pred.authors -1  1
       FALSE 15  0
       TRUE   0 51
```

# Classification: Evaluation (out of sample)

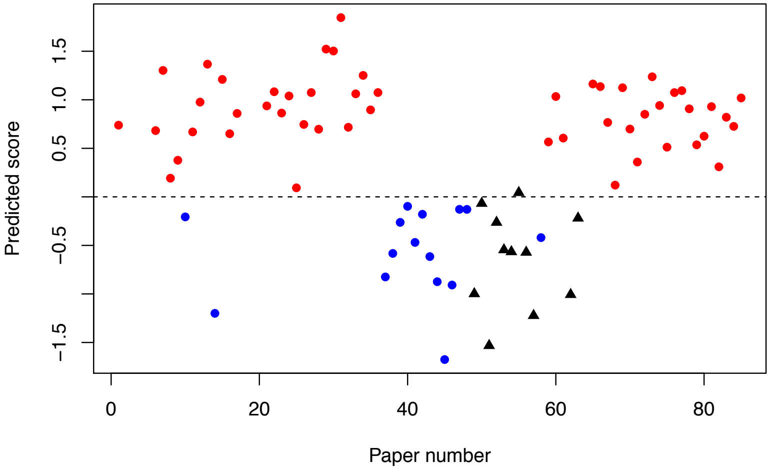
The real test of a classification model is to predict data it *has not* seen before

```
preds <- predict(hm.mod, newdata=test.dat)
preds > 0

fp49.txt fp50.txt fp51.txt fp52.txt fp53.txt fp54.txt
   FALSE    FALSE    FALSE    FALSE    FALSE    FALSE
fp55.txt fp56.txt fp57.txt fp62.txt fp63.txt
    TRUE    FALSE    FALSE    FALSE    FALSE
```

Who probably wrote the contested Federalist papers?

# Classification



# Classification: Evaluation (even more out of sample)

What about those joint authored papers?

```
hamad.dat <- dat[dat$author=='hamilton.madison',]
predict(hm.mod, newdata=hamad.dat)

  fp18.txt   fp19.txt   fp20.txt
-0.3853885 -0.6102727 -0.1250502
```

# Classification: Evaluation (even more out of sample)

Does *John Jay* write more like Hamilton or more like Madison?

```
jay.dat <- dat[dat$author=='jay',]
predict(hm.mod, newdata=jay.dat)

   fp02.txt    fp03.txt    fp04.txt    fp05.txt    fp64.txt
-0.13624854 -1.35995768 -0.04175293 -0.26288400 -0.19032925
```

# Faking 'out of sample': Testing generalization

How good could we *expect* to be out of sample (before we even try)
We can get an idea of what out of sample performance would be like with **cross-validation**

- Remove some of the training data
- Test on what we left out
- See how well we did

*In the book* we check how well we expect to able to predict authorship using the training set
Let's do some other useful things with cross-validation

# Looking ahead

If history had been a little different,

- our coefficients would have been a little different

If our coefficients were a little different

- our predictions are a little different

But *how* different?
Figuring out how different *without changing the world to see* is one of the things **statistics** studies

# Words as predictors

We simply chose 4 words because we thought (correctly) that they would work well for author identification

What if we didn't know how to choose them?

Some alternatives

- Choose a random subset
- Choose the most common words
- Choose the *least* common words
- Choose the words that we are sure we'll see in most documents
- Bundle up words to form topic vocabulary and count *that*

# Words as predictors

A systematic approach from computer science:
Good informative predictor words are

- Frequent (but not *too* frequent)
- Occur in fewer documents (so are more informative about them)

One transformation that balances these requirements is **tf-idf**

- 'tf' is the log of the *term frequency* (word count)
- 'idf' is the log of the **inverse document frequency** (the inverse of the proportion of documents containing the word)

# Words as predictors: tf-idf

How many times does ‘man’ occur in Paper 73?

```
dtm1[73, 'man']

[1] 1.702611
```

Log of this is 0.5321628

# Words as predictors: tf-idf

The **document frequency** of ‘man’

```
sum(dtm1[, 'man'] > 0) / nrow(dtm1)

[1] 0.5882353
```

so the *inverse* is

```
nrow(dtm1) / sum(dtm1[, 'man'] > 0)

[1] 1.7
```

and the log is 0.5306283
So the tf-idf score of ‘man’ in Paper 73 is 0.2823806

# Authorship attribution (again)

Politicians in social media

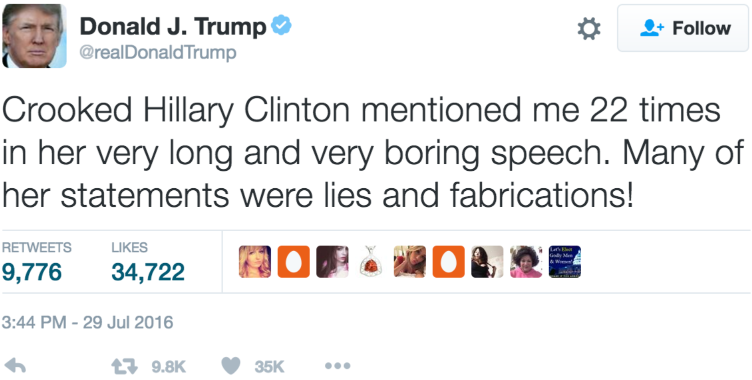



Figure:

# Authorship attribution (again)