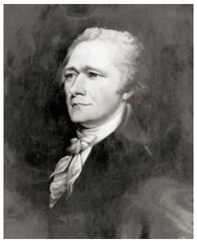
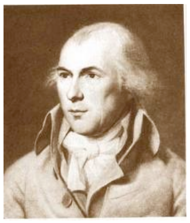
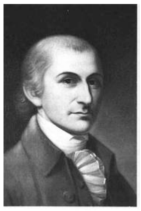
Authorship Attribution and the Federalist Papers

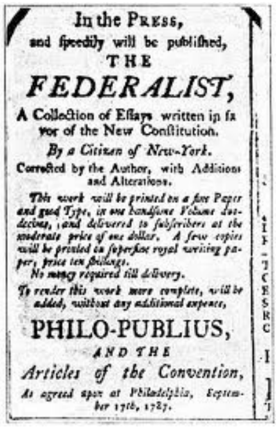

Hamilton


Madison


Jay

Summary:

We will replicate and extend the authorship attribution experiments of Jockers and Witten (2010) on the Federalist Papers. The **Federalist Papers** are “a series of 85 articles or essays advocating the ratification of the United States Constitution”(Wikipedia, 2011). In particular, we will use a number of machine learning classification techniques to gather evidence as to who *may have* written the twelve disputed papers and the three co-authored papers.


In the Preſs,
and ſpeedily will be publiſhed,
THE
FEDERALIST,
A Collection of Eſſays written in fa vor of the New Conſtitution.
By a Citizen of New-York.
Corrected by the Author, with Additions and Alterations.

This work will be printed on a fine Paper and good Type, in one handſome Volume duodecimo, and delivered to ſubſcribers at the moderate price of one dollar. A few copies will be printed on ſuperfine royal writing paper, price ten ſhillings.
No money required till delivery.
To render this work more complete, will be added, without any additional expence,
PHILO-PUBLIUS,
AND THE
Articles of the Convention,
As agreed upon at Philadelphia, September 17th, 1787.

Part I – The Beginning Game

(0) Fetch Federalist Papers.

(a) Project Gutenberg is always a good place to start:
http://www.gutenberg.org/

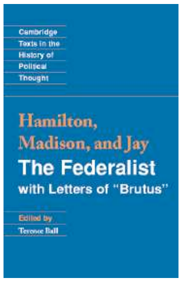
(b) *or* you know someone who has ‘em; in this case Matt Jockers, now at University Nebraska-Lincoln, shared his XML version of the text(s): one file, each of the separate Papers put in `<div><text>… </text></div>` tags.

(1) Parse into separate texts (I’ve done these steps and will share the code)

(2) Read up on XML (especially if you haven’t encountered this yet).
w3schools.com is best place to start …
http://www.w3schools.com/xml/

(3) Read up on the Document Object Model (DOM):
http://www.w3schools.com/dom/default.asp

A DOM (Document Object Model) defines a standard way for accessing and manipulating documents. The XML DOM defines a standard way for accessing and manipulating XML documents. The XML DOM views an XML document as a tree-structure. All elements can be accessed through the DOM tree. Their content (text and attributes) can be modified or deleted, and new elements can be created. The elements, their text, and their attributes are all known as nodes.


Cambridge
Texts in the
History of
Political
Thought

Hamilton,
Madison, and Jay
The Federalist
with Letters of “Brutus”

Edited by
Terence Ball

(4) Use Python’s miniDOM module to help parse DOM tree into objects:
http://docs.python.org/library/xml.dom.minidom.html

(5) Check out Starter Kit for parsing solution:
Run `parseFederalistPapers.py` using the XML version, `gutenfeder.xml`

(6) Read Matt Jockers and Daniela Witten's 2010 paper (see onCourse). Complete the set of guided questions (see addendum to this spec). These questions will help you to understand the paper, but as important, the guide provides you with *an example* of how I read and reread sections as I took notes to isolate and analyze the important parts of the work.

(7) Use single word tokens. Ignore whitespace (newlines, tabs, runs of spaces); convert everything to lowercase; remove any "strange" character within a token by replacing it with nothing (`''`), that is substitute [nothing] for a character that is not a letter or a digit; remove any punctuation *except* keep apostrophes, so for example we'd keep "`we'd`" as one token). Create the feature set for all the Federalist Papers, clearly a hash table of tokens and their respective counts of the number of times it occurred. Produce a matrix, 85xN, where there are 85 texts and N unique words. Also show the total number of words.

Note: If you wish, I will entertain the use of bigrams, that is, you could word pairs. Thus in the sentence, "Does your dog bite?" the features would be: "does your", "your dog", "dog bite".

> Be prepared to answer and discuss your results at this point. Do you have the same number of features (words) as Jockers and Witten? Why not? *Note: You may want (or need) to modify your list of features. For example, I believe Mallet will complain if you have entries with zero (0). Also, as you noticed from Jocker's paper, he removed those words that do not appear in all the texts, right?*

(8) Create a word cloud of all the words in the Federalist Papers (see: `http://www.wordle.net/`). Use this image somewhere in your final report (e.g., see mine at the top of this spec).

Part II – The "Middle Game"[1]

(9) Run at least two machine learning algorithms using the entire feature (word) list of each author.

References

Federalist Papers. http://en.wikipedia.org/wiki/Federalist_papers Accessed: 03/08/2011

Jockers, M.L. and Witten, D.M. (2010). A comparative study of machine learning methods for authorship attribution. *Literary and Linguistic Computing, v25(2), p215-223.*

[1] When I lived in Australia during 2004-2005, I met Dr. John Burrow's who devised the Delta authorship attribution technique. John was very modest about his computational techniques, calling them *only* middle game techniques. That is, there is much scholarship to do at the start, we then perform computational experiments, but then there is still much scholarship to do afterwards.

These questions are a guide to reading Jockers and Witten (2010). The answers to these questions appear in the text in (roughly) the order that they appear. I am assuming (and would like to believe) that you will have read the paper once all the way through first, skipping over the details that you know you will need to return to later when you attempt to really understand the experimental methods.

Submit *typed* answers to these questions.

Q0: What do Jockers and Witten say are the two most important factors from a machine learning perspective in authorship attribution?

Q1: What (types of features) are considered to provide the most reliable results in authorship attribution?

Q2: What is considered to be the best classifier?

Q3: Define “benchmarking” (you’ll have to look elsewhere for definitions).

Q4: Define “tuning parameters”.

Q5: Define “closed set”.

Q6: What are the three objectives of this study?

Q7: Make a table showing for each of the five methods the tuning parameters used.

Q8: Explain the difference between the “raw features” and the “pre-processed features”.

Q9: Explain the meanings behind the sizes of the three matrices of [Texts *x* Features] space.

Q10: Define “cross-validation”.

Q11: What are the training and test sets?

Q12: What texts and how many are in the Disputed and Co-Authored groups?

Q13: What is the general agreement with prior attribution work regarding Madison vs. Hamilton on the disputed papers?

Q14: When discussing the NSC model, the authors use a cautious way to describe the results. How do they recommend stating the results?