Inaugurating Data Science Skills by Studying Inaugural Addresses

Robert Michael Morrissey

May 9, 2017

# An Experiment: Soup to Nuts

Presidential addresses have drawn attention of historians and other scholars for years. In addition to scholars who focus on indiviual presidents’ inaugural speeches as texts fit for close reading, a good many scholars have viewed the inaugural address as a kind of genre, and they have analyzed *all* inaugural addresses as a group. Treating the inaugural addresses as a corpus, some have explored questions about “civil religion,” or discourses of nationalism, as they are expressed in the presidents’ speeches.[[1]](#footnote-22) Others have focused attention on rhetorical commonalities in the inaugural addresses, finding similar rhetorical “moves” in the speeches across time.[[2]](#footnote-24) Methodolgically, much of this work has relied on tools such as “content analysis,” a laborious and justly-critiqued method of manual keyword counting.[[3]](#footnote-27) Recently, scholars have begun to use tools of computational text analysis to explore the inaugural addresses algorithmically.[[4]](#footnote-28) My project was inspired by a desire to apply new methods of machine learning to the corpus of inaugural addresses. As a historian, I have been impressed at the applications of these new techniques for distant reading in literature, such as in the study of literary genre over time. I wondered whether machine learning techniques– especially the use of classifiers and other supervised learning techniques, or unsupervised learning techniques like clustering– could be applied to a “historical” data-set like the inaugural addresses to yield interesting results.[[5]](#footnote-30) I was especially interested in whether exploratory data analysis could provide ‘objects’ for interpretation– models and coefficients that might prompt interesting historical questions about difference, similarity and pattern in historical data. As a historian, perhaps my biggest interpretive interest was in seeing whether I could apply some of these methods to suss out the historian’s most important quarry: novel understandings (or simple observations) of change over time.

As I mentioned during my in-class presentation of my work, my priority in this final project was to try to employ several of the new tools in my EDA and machine learning toolkit. But I also wanted to practice a “soup to nuts” project design and execution. I take to heart– and I know from experience– that none of the machine learning methods are really worth much to somebody who has no useful datasets. A week prior to beginning the Spring 2017 semester, I happened on a new article by Andrew Goldstone about the challenges of teaching quantitative methods in the humanities.[[6]](#footnote-31) One of Goldstone’s central reflections about teaching this material to humanities students and scholars is that datasets are often a central problem. Even the most dedicated would-be digital humanist needs to have a good dataset or two. At the same time, it is pretty worthless if digital humanists confine themselves to questions that can be answered by querying the same handful of ready-made datasets that are available to researchers thanks to a library investment, a handy API, or some other unusual circumstance. After all, if I only gravitate towards those datasets that already exist, my research will by necessity be driven not by my intellectual questions, but rather by a more opportunistic consideration of what questions can likely be answered by the data at hand. Of course this is a dilemma that we always face in the discipline of history, even when we practice it in a completely analog fashion. But reflecting on this dilemma as it pertains to digital history, I decided that I would dedicate myself to creating (or at least finding) a new dataset for analysis. To the extent that digital humanists are joining the world of “makers,” it is probably just as important that we learn to “make” our datasets as it is that we learn to make some kind of code or app. Of course, I know for the purposes of this project that this commitment on my part involves a tradeoff; one of Ted’s syllabus entries for the week on data munging promised that munging data was “90% of the work,” so I knew that deciding to create a dataset by myself would leave me less time for interesting visualizations and analysis. Still, I decided that if I spent 90% of my work time in this project getting a dataset ready for analysis, I could live with that, as long as I succeeded in creating a dataset to which I could in the end apply the methods like dunnings or bayes.

In this connection, there’s one more reflection that I want to make by way of introduction. This is a project where the technical cart is pulling the intellectual horse, so to speak. Despite the fact that I dedicated myself to munging a dataset and doing this work soup to nuts, I still had to pick my topic based on what I knew about scrapable clean texts online, and that left me with only a few options (that I could think of) for a group of texts that could constitute corpora for analysis with the tools that we have learned and with the web-scraping and parsing techniques that I know so far. In other words, I view my “big data” historical work as a work in progress, to be realized in the future if ever. For now, I understand and accept my limitations. The good news is that I still find this fun even if it does not result in publishable work! Obviously at some point this will likely change, but for now, a project like this is just fine for me, despite its obvious weaknesses, and the fact that it is less a question-driven investigation and more a skills-driven proof of concept.

# Trial and Error: Munging

Since I was especially interested in classifiers, I began my work searching for data (i.e. digitized texts) for which an obvious problem was their comparability, or their differentiation from one another. I initially wanted to work with pamphlets from the AMerican REvolution, examining the extent to which pamphlets published in Britain were distinguishable from those published in the colonies, as well as the question of when that distinction became pronounced. Unfortunately, I could find no way to get a good corpus of texts for this purpose, and I abandoned it.

My next idea was to use the JSTOR API to examine historiography through machine learning in five historical journals over the last 25 years or so. I planned to use algorithms to examine article abstracts or even titles and see whether a method like bayes or another machine learning model could successfully classify abstracts from particular journals, or from particular periods of time. My thought was that the classifiers probably would fail to distinguish journals from one another, but might succeed at distinguishing decades, or perhaps subjects. If this proved true, and if then I could group my dataframes into decadal or 5-year subsets, perhaps I could notice a differential over time or journal in the models’ or algorithms’ ability to distinguish the data which might tell me something interesting about scholarly conversations over time. Unfortunately, this project never got off the ground. I requested datasets from the JSTOR [Data For Research](www.dfr.jstor.org/) site, and received the nearly 5000 records that JSTOR allows researchers to request. I began writing code to munge the xml and json files for one of the journals I was examining. Unfortunately, after succeeding in my efforts to parse the xml files for the *William and Mary Quarterly*, I realized that the file structure for the other journals in my corpus was slightly different, and so was the xml format for the citations. As a result, I could not– as I had hoped I could– apply my functions to the rest of the corpus to create a giant (n= 5000) dataframe. After banging my head against the wall for a while, I gave up. The good news is I now know how I can use the xml files from jstor much more efficiently, and I solved a few parsing challenges (such as how to deal with multiple authors while looping through an xml tree and parsing data into lists for a dataframe). But in the end I realized that the jstor data was going to be too much work for my time constraints with this project.

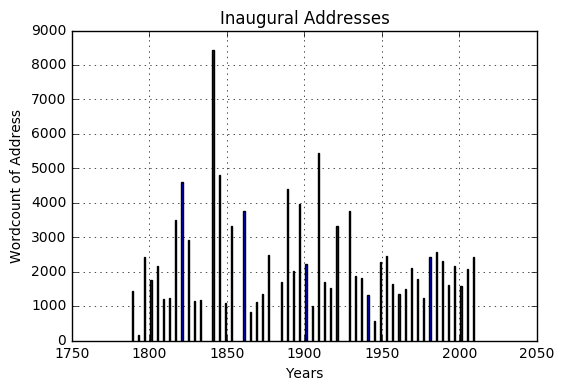
My fallback was the inaugural addresses. In contrast to the historiography problem that I started with, my investigation into the inaugural addresses did not start with even the most general historical question; this was *extremely* exploratory data analysis. But if I did not have a question, I knew intuitively that there might be a way to apply the classifier and clustering techniques to the inaugural addresses just to see what would happen. I knew that the dataset was very small– just shy of 60 texts. But I also knew that the munging and the parsing for this dataset would be satisfying and at the same time feasible. I decided that I could explore distinctions in party and use the algorithms to see if the presidents’ party affiliations would be reflected in a rhetorical or semantic distinction that the classifiers could pick up. If so, I wondered if that dinstinction increased over time, or was greater in one era than another. Even with a small dataset, I wondered if some of these issues could be explored. Furthermore, I wondered if a similar distinction was detectable between presidents who were “party incumbents”– i.e. entering the White House on the heels of a previous president from their own political party– and “party newcomers.” In the end, I was not able to execute the change over time dimension of the analysis, but I came to certain (although not surprising or significant, as we will see) answers on the rest of these questions.

# Munging Process

My munging process was, as predicted, the lion’s share of the work I performed on this project. I got the bulk of my data by scraping the inaugural addresses and other metadata from the [Yale Avalon Project](http://avalon.law.yale.edu/). You can see my munging in the attached jupyter notebook or in the one posted at [GitHub](https://github.com/historybman/EADA/blob/master/Inaugural_Munging.ipynb). The two biggest technical challenges in this part of the work were:

* First, I scraped the texts from a table at the Avalon project, using BeautifulSoup. Parsing the html links and years was no trouble, and I wound up with two lists for year and urls leading to the actual texts. The problem was when I went to create a list of presidents– I could not figure out how to duplicate multi-year president’s names. For instance, I needed FDR to appear in the list four times to match his four inaugural addresses. I had to hand code. Similarly, I needed to add political parties by hand.
* Secondly, I spent a long time trying to get all the texts in programatically, but some of them had to be added after the fact, by hand. For these, I wrote a small python program to clean the text (remove newlines and etc., so that I had just one long continuous string. That python program is in my folder with the rest of the data).
* Thirdly, I had to add the column for party incumbency, and for this I decided to simply write a function in excel– I am pretty fluent with writing simple functions in excel and I knew that I could accomplish my task much easier in excel than would have been possible inside pandas. I know I *can* accomplish the same thing with pandas, but I went with excel just for time’s sake.

The rest of the munging was pretty straightforward; I had a struggle removing some of the extraneous words that did not come out with my function to remove header and footers. I wrote some functions to count the words in the texts, which incidentally gave me a pretty nice little picture:



Word Counts

In the end, I was able to create a csv file for my analysis, called inaugurals\_complete.csv. It sits in a folder in my working directory called ./data/ and it has all my cleaned text and the metadata I need for the analysis. It definitely was 90% of the work to create this csv file, but I was pretty pleased that I was able to solve the various problems involved. I think I’ll be ready when it comes time to ‘scale up’ my analysis and make a truly large dataset.

# Analysis

As mentioned, my analysis was mostly exploratory, not question-driven. I decided that I wanted to run some of the classifiers to see how well they could distinguish the inaugural addresses across party, time, and party incumbency. In the end I dropped time becuase I realized first that I did not have enough data, and second that it would require a different approach, one that I am not sure I could perform with the limited amount of time I had for the rest of this project. In the end I did not write any new algorithms or adapt any modules from outside of class; rather I picked a few of what seemed like the most promising methods and I simply ran them mostly by cutting and pasting Ted’s code into my notebooks.

### Robinson’s Log Odds and Dunnings

In my [Inaugural Address Analysis notebook](https://github.com/historybman/EADA/blob/master/Inaugural_Address_Analysis.ipynb), I ran several exploratory methods on my dataset. Starting off with the text analysis methods from Week 4, I performed Robinson’s log-odds measure and Dunning’s log-likelihood. I used these to measure the most overrepresented and underrepresented words across two categories: political party and incumbency. The results were interesting. Republican presidential speeches overrepresented the following words, according to the signed dunnings method:

TOP VALUES:  
law 37.85900298929792  
business 32.87730435677244  
enforcement 27.661956717563825  
there 26.71258693822815  
no 21.023840498701418  
we 20.368952478920193  
freedom 19.949151468604846  
will 19.711108571025164  
make 19.27829134912094  
do 18.29887958039273  
negro 17.7213002579283  
amendment 17.662006465968325  
accept 17.01798088257794  
congress 16.84090924743422  
america 16.585253111580684  
method 14.88972168638792  
islands 14.88972168638792  
arbitration 14.88972168638792  
south 14.304357332097728  
prayer 13.7751544999991

Democratic presidents’ speeches featured these words disproportionately:

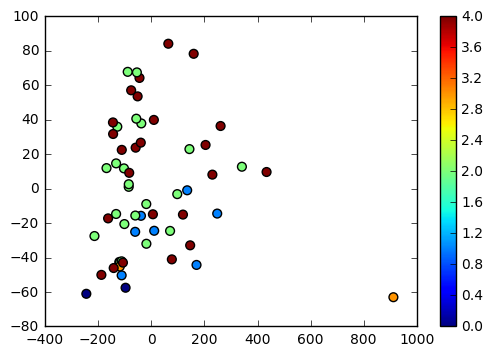
BOTTOM VALUES:  
which -25.587694865809624  
been -24.768251604369084  
powerful -24.257396715885317  
powers -20.639945618320855  
union -19.808186401517098  
myself -19.68520084932692  
me -19.212377774373138  
foreign -18.94148581831667  
my -18.87101149359202  
 -18.74433136968643  
on -18.413088656786343  
opinion -16.987689023401476  
happy -16.890688937556916  
fellow-citizens -16.81270986608169  
spirit -16.686359964373743  
period -16.515374263165533  
limits -16.276798224583224  
measures -15.846495944172565  
country's -15.213400850163701  
democracy -14.591557939972915

While there’s no real logic to these, one thing that does strike me as interesting is the use of the first person in the Democratic presidents’ speeches– me, myself, my. I wonder if that overrepresentation is a result of an idiosyncracy– FDR’s stylistic approach, for instance (becuase he’s definitely overrepresented in the sample). Or possibly Democratic presidents were more ‘familiar’ in their speeches. In any event, it was an interesting result.

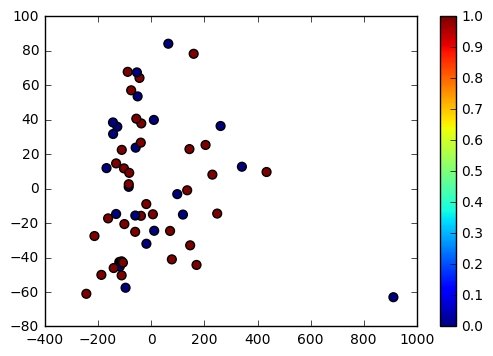
Running the same analyses across the incumbency category, I was struck that speeches by presidents from newcomer parties overrepresented words like “no,” “any,” and “used,” in addition to the ones that we might recognize as reflecting the “change” in political regime: “renewal,” “begin,” “intention.” The list of overrepresented words in the incumbent speeches do not surprise much, but they too suggest some interesting themes. I was struck that this kind of method could be a better approach to some of the questions that scholars have used “content analysis” to answer in the past.[[7]](#footnote-42)

### Unsupervised Learning

My next step was to use clustering to see if the algorithms could separate the speeches by party. Interestingly, the answer is probably no. My attempt to run unsupervised learning on the corpus seemed not to distinguish clusters either for political party (although there are some general clusters present here in the figure below) nor for incumbency. I frankly was not surprised to see this; after all, the themes and ideas that define political parties at any given time do not hold constant over time, and it is not surprising that an unsupervised learning method would not put Lincoln’s speeches togehter with Nixons, as indeed this method did not. Interestingly, re-coloring the dots as “incumbent” and “new party” makes it clear that the method does little better to distinguish those categories.



Unsupervised Learning: Political Parties



Unsupervised Learning: Incumbency

### Bayes

The last analytical approach I tried was [Bayesian analysis](https://github.com/historybman/EADA/blob/master/Inaugural_Bayes.ipynb). As mentioned above, I was not surprised when machine learning failed to cluster the documents together based on categories such as politcal party and incumbency. Experiencing this failure then prepared me for what happened with the Bayesian analysis. Not surprisingly, Bayes failed– and it failed pretty badly. I ran it just once – I did not do the fivefold verification for lack of time. But when I tried to run a Bayesian analysis to predict incumbent and non-incumbent in my test set, the model did horribly, predicting 2 rows correctly and 8 incorrectly. Assuming that the political parties would not improve on this given the logic above about changing ideology and shifting political principles and themes over time, I decided not to try it on the political parties. I was also very short on time.

# Conclusion

My final project for Data Science for the Humanities was ultimately not successful at producing a significant finding or what I would consider a noteworty conclusion. Nobody would be surprised to know that a machine learning project did not distinguish the content, semantics, or rhetoric of inaugural addresses on the basis of political party of incumbency– there are too many other variables that affect the content of inaugural addresses, including a long sweep of time during which language changed, idiom changed, context changed, ideology changed, and etc. The project was not well-designed to yield a significant result.

And yet, there are some interesting things that we can say about the non-result. The *failure* of the algorithms to distinguish REpublican from Democratic inaugural addresses probably does reflect how much the ideologies of “Democratic” and “Republican” parties have changed over time. For instance, it does not take a deep understanding of American history to know that 19th century Republicans were the liberals on many social questions of their day, and their relative liberalism on many questions lasted right up until the Civil Rights movement and arguably Nixon. It makes sense that machine learning and other algorithms cannot find the common thread between the parties’ rhetoric over time.

And there might be an interesting upshot to this point. For instance, is popular for modern-day political parties– especially in moments like inaugurations– to celebrate their ideological heritage and to gesture at a long and consistent political tradition. For instance, it is common to hear Republican candidates for office celebrate the GOP as “the party of Lincoln.”

I think the results of my analysis have something to say about this. If an unsupervised machine learning cannot cluster Lincoln’s ideas together with Nixon’s, then perhaps it’s fair to say that “data science” argues against the idea that today’s Republicans are the party of Lincoln, or that Obama is somehow the inheritor of a Wilsonian intellectual tradition, or whatever. Previous scholars of inaugural addresses have identified a “civil religion” present in the presidents’ rhetoric. But if certain themes are consistent in the inaugural addresses over time, much of the contents of political ideology and party identity is not. Machine learning the inaugural addresses helps us see the fluidity and ever-changing nature of political ideology and circumstances over time.

Fulmer, Hal W. “A Rhetoric of Oldspeak: Mythic Elements in Presidential Inaugural Addresses, 1960-1980.” *Rhetoric Society Quarterly* 16, no. 4 (1986): 299–312. <http://www.jstor.org/stable/3885546>.

Goldstone, Andrew. “Teaching Literary Data: What Makes It Hard · Andrew Goldstone.” *Andrew Goldstone*. Accessed January 6, 2017. <https://andrewgoldstone.com/blog/ddh2018preprint>.

McDiarmid, John. “Presidential Inaugural Addresses-A Study in Verbal Symbols.” *The Public Opinion Quarterly* 1, no. 3 (1937): 79–82. <http://www.jstor.org/stable/2744670>.

Toolin, Cynthia. “American Civil Religion from 1789 to 1981: A Content Analysis of Presidential Inaugural Addresses.” *Review of Religious Research* 25, no. 1 (1983): 39–48. doi:[10.2307/3511310](https://doi.org/10.2307/3511310).

Whissell, Cynthia, and Lee Sigelman. “The Times and the Man as Predictors of Emotion and Style in the Inaugural Addresses of U.S. Presidents.” *Computers and the Humanities* 35, no. 3 (2001): 255–72. <http://www.jstor.org/stable/30204857>.

1. Cynthia Toolin, “American Civil Religion from 1789 to 1981: A Content Analysis of Presidential Inaugural Addresses,” *Review of Religious Research* 25, no. 1 (1983): 39–48, doi:[10.2307/3511310](https://doi.org/10.2307/3511310). [↑](#footnote-ref-22)
2. John McDiarmid, “Presidential Inaugural Addresses-A Study in Verbal Symbols,” *The Public Opinion Quarterly* 1, no. 3 (1937): 79–82, <http://www.jstor.org/stable/2744670>, Hal W. Fulmer, “A Rhetoric of Oldspeak: Mythic Elements in Presidential Inaugural Addresses, 1960-1980,” *Rhetoric Society Quarterly* 16, no. 4 (1986): 299–312, <http://www.jstor.org/stable/3885546>. [↑](#footnote-ref-24)
3. Toolin, “American Civil Religion from 1789 to 1981.” [↑](#footnote-ref-27)
4. Cynthia Whissell and Lee Sigelman, “The Times and the Man as Predictors of Emotion and Style in the Inaugural Addresses of U.S. Presidents,” *Computers and the Humanities* 35, no. 3 (2001): 255–72, <http://www.jstor.org/stable/30204857>. [↑](#footnote-ref-28)
5. This work was performed for Ted Underwood’s course *Data Science in the Humanities*, taught in the Spring semester of 2017 at GSLIS. I am grateful to Ted for letting me take this course and for helping me see how some of these new methods might apply to historical questions and datasets. [↑](#footnote-ref-30)
6. Andrew Goldstone, “Teaching Literary Data: What Makes It Hard · Andrew Goldstone,” *Andrew Goldstone*, accessed January 6, 2017, <https://andrewgoldstone.com/blog/ddh2018preprint>. [↑](#footnote-ref-31)
7. Toolin, “American Civil Religion from 1789 to 1981.” [↑](#footnote-ref-42)