Text as Data

Matthew Gentzkow
Stanford

Bryan T. Kelly
Yale and AQR Capital Management

Matt Taddy
Chicago Booth

Abstract

An ever increasing share of human interaction, communication, and culture is recorded as digital text. We provide an introduction to the use of text as an input to economic research. We discuss the features that make text different from other forms of data, offer a practical overview of relevant statistical methods, and survey a variety of applications.

1 Introduction

New technologies have made available vast quantities of digital text, recording an ever increasing share of human interaction, communication, and culture. For social scientists, the information encoded in text is a rich complement to the more structured kinds of data traditionally used in research, and recent years have seen an explosion of empirical economics research using text as data.

To take just a few examples: in finance, text from financial news, social media, and company filings is used to predict asset price movements and study the causal impact of new information. In macroeconomics, text is used to forecast variation in inflation and unemployment, and estimate the effects of policy uncertainty. In media economics, text from news and social media is used to study the drivers and effects of political slant. In industrial organization and marketing, text from advertisements and product reviews is used to study the drivers of consumer decision making. In political economy, text from politicians' speeches is used to study the dynamics of political agendas and debate.

The most important way that text differs from the kinds of data often used in economics is that text is inherently high-dimensional. Suppose that we have a sample of documents, each of which is w words long, and suppose that each word is drawn from a vocabulary of p possible words. Then the unique representation of these documents has dimension p^w. A sample of 30-word Twitter messages that use only the 1,000 most common words in the English language, for example, has roughly as many dimensions as there are atoms in the universe.

A consequence is that the statistical methods used to analyze text are closely related to those used to analyze high-dimensional data in other domains, such as machine learning and computational biology. Some methods, such as lasso and other penalized regressions, are applied to text more or less exactly as they are in other settings. Other methods, such as topic models and multinomial inverse regression, are close cousins of more general methods adapted to the specific structure of text data.

In all of the cases we consider, the analysis can be summarized in three steps:

1. Represent raw text $\mathcal{D}$ as a numerical array $\mathbf{C}$

2. Map $\mathbf{C}$ to predicted values $\hat{\mathbf{V}}$ of unknown outcomes $\mathbf{V}$

3. Use $\hat{\mathbf{V}}$ in subsequent descriptive or causal analysis

In the first step, the researcher must impose some preliminary restrictions to reduce the dimensionality of the data to a manageable level. Even the most cutting-edge high-dimensional techniques can make nothing of 1000^{30}-dimensional raw Twitter data. In almost all the cases we discuss, the elements of $\mathbf{C}$ are counts of *tokens*: words, phrases, or other pre-defined features of text. This step may involve filtering out very common or uncommon words; dropping numbers, punctuation, or proper names; and restricting attention to a set of features such as words or phrases that are likely to be especially diagnostic. The mapping from raw text to $\mathbf{C}$ leverages prior information about the structure of language to reduce the dimensionality of the data prior to any statistical analysis.

The second step is where high-dimensional statistical methods are applied. In a classic example, the data is the text of emails, and the unknown variable of interest $\mathbf{V}$ is an indicator for whether the email is spam. The prediction $\hat{\mathbf{V}}$ determines whether or not to send the email to a spam filter. Another classic task is sentiment prediction (e.g., Pang et al. 2002), where the unknown variable $\mathbf{V}$ is the true sentiment of a message (say positive or negative), and the prediction $\hat{\mathbf{V}}$ might be used to identify positive reviews or comments about a product. A third task is predicting the incidence of local flu outbreaks from Google searches, where the outcome $\mathbf{V}$ is the true incidence of flu.

In these examples, and in the vast majority of settings where text analysis has been applied, the ultimate goal is prediction rather than causal inference. The interpretation of the mapping from $\mathbf{V}$ to $\hat{\mathbf{V}}$ is not usually an object of interest. Why certain words appear more often in spam, or why certain searches are correlated with flu is not important so long as they generate highly accurate predictions. For example, Scott and Varian (2014, 2015) use data from Google searches to produce high-frequency estimates of macroeconomic variables such as unemployment claims, retail sales, and consumer sentiment that are otherwise available only at lower frequencies from survey data. Groseclose and Milyo (2005) compare the text of news outlets to speeches of congresspeople in order to estimate the outlets' political slant. A large literature in finance following Antweiler and Frank (2004) and Tetlock (2007) uses text from the Internet or the news to predict stock prices.

In many social science studies, however, the goal is to go further and, in the third step, use text to infer causal relationships or the parameters of structural economic models. Stephens-Davidowitz (2014) uses Google search data to estimate local areas' racial animus, then studies the causal effect

of racial animus on votes for Obama in the 2008 election. Gentzkow and Shapiro (2010) use congressional and news text to estimate each news outlet's political slant, then study the supply and demand forces that determine slant in equilibrium. Engelberg and Parsons (2011) measure local news coverage of earnings announcements, then use the relationship between coverage and trading by local investors to separate the causal effect of news from other sources of correlation between news and stock prices.

In this paper, we provide an overview of methods for analyzing text, and a survey of current applications in economics and related social sciences. The methods discussion is forward-looking, providing an overview of methods that are currently applied in economics as well as those that we expect to have high value in the future. Our discussion of applications is selective and necessarily omits many worthy papers. We highlight examples that illustrate particular methods, as well as use text data to make important substantive contributions even if they do not apply methods close to the frontier.

A number of other excellent surveys have been written in related areas. See Evans and Aceves (2016) and Grimmer and Stewart (2013) for related surveys focused on text analysis in sociology and political science respectively. For methodological survey, Bishop (2006), Hastie et al. (2009), and Murphy (2012) cover contemporary statistics and machine learning in general while Jurafsky and Martin (2009) overviews methods from computational linguistics and natural language processing. The Spring 2014 issue of *Journal of Economic Perspectives* contains a symposium on "big data," which surveys broader applications of high-dimensional statistical methods to economics.

In Section 2 we discuss representing text data as a manageable (though still high-dimensional) numerical array $\mathbf{C}$; in Section 3 we discuss methods from data mining and machine learning for predicting $\mathbf{V}$ from $\mathbf{C}$. Section 4 then provides a selective survey of text analysis applications in social science, and Section 5 concludes.

2 Representing text as data

When humans read text, they do not see a vector of dummy variables, nor a sequence of unrelated tokens. They interpret words in light of other words, and extract meaning from the text as a whole.

It might seem obvious that any attempt to distill text into meaningful data must similarly take account of complex grammatical structures and rich interactions among words.

The field of computational linguistics has made tremendous progress in this kind of interpretation. Most of us have mobile phones that are capable of complex speech recognition. Algorithms exist to efficiently parse grammatical structure, disambiguate different senses of words, distinguish key points from secondary asides, and so on.

Yet virtually all analysis of text in the social sciences, like much of the text analysis in machine learning more generally, ignores the lion's share of this complexity. Raw text consists of an ordered sequence of language elements: words, punctuation, and whitespace. To reduce this to a simpler representation suitable for statistical analysis, we typically make three kinds of simplifications: dividing the text into individual documents i, reducing the number of language elements we consider, and limiting the extent to which we encode dependence among elements within documents. The result is a mapping from raw text $\mathcal{D}$ to a numerical array $\mathbf{C}$. A row $\mathbf{c}_i$ of $\mathbf{C}$ is a numerical vector with each element indicating the presence or count of a particular language token in document i.

2.1 What is a document?

The first step in constructing $\mathbf{C}$ is to divide raw text $\mathcal{D}$ into individual documents $\{\mathcal{D}_i\}$. In many applications, this is governed by the level at which the attributes of interest $\mathbf{V}$ are defined. For spam detection, the outcome of interest is defined at the level of individual emails so we want to divide text that way too. If $\mathbf{V}$ is daily stock price movements which we wish to predict from the prior day's news text, it might make sense to divide the news text by day as well.

In other cases, the natural way to define a document is not so clear. If we wish to predict legislators' partisanship from their floor speeches (Gentzkow et al. 2016) we could aggregate speech so a document is a speaker-day, a speaker-year, or all speech by a given speaker during the time she is in Congress. When we use methods that treat documents as independent (which is true most of the time), finer partitions will typically ease computation at the cost of limiting the dependence we are able to capture. Theoretical guidance for the right level of aggregation is often limited, so this is an important dimension along which to check the sensitivity of results.

2.2 Feature selection

To reduce the number of features to something manageable, a common first step is to strip out elements of the raw text other than words. This might include punctuation, numbers, HTML tags, proper names, and so on.

It is also common to remove a subset of words that are either very common or very rare. Very common words, often called "stop words," include articles ("the," "a"), conjunctions ("and," "or"), forms of the verb "to be," and so on. These words are important to the grammatical structure of sentences, but they typically convey relatively little meaning on their own. The frequency of "the" is probably not very diagnostic of whether an email is spam, for example. Common practice is to exclude stop words based on a pre-defined list.[1] Very rare words do convey meaning, but their added computational cost in expanding the set of features that must be considered often exceeds their diagnostic value. A common approach is to exclude all words that occur fewer than k times for some arbitrary small integer k.

An approach that excludes both common and rare words and has proved very useful in practice is filtering by "term-frequency-inverse-document-frequency" (tf-idf). For a word or other feature j in document i, term frequency (tf_{ij}) is the count c_{ij} of occurrences of j in i. Inverse document frequency (idf_j) is the log of one over the share of documents containing j: $\log\left(n/d_j\right)$ where $d_j = \sum_i \mathbb{1}_{[c_{ij}>0]}$ and n is the total number of documents. The object of interest tf-idf is the product $tf_{ij} \times idf_j$. Very rare words will have low tf-idf scores because tf_{ij} will be low. Very common words that appear in most or all documents will have low tf-idf scores because idf_j will be low. (Note that this improves on simply excluding words that occur frequently because it will keep words that occur frequently in some documents but do not appear in others; these often provide useful information.) A common practice is to keep only the words within each document i with tf-idf scores above some rank or cutoff.

A final step that is commonly used to reduce the feature space is *stemming*: replacing words with their root, such that, e.g., "economic," "economics," "economically" are all replaced by the stem "economic." The Porter stemmer (Porter 1980) is a standard stemming tool for English

[1] There is no single stop word list that has become a standard. How aggressive one wants to be in filtering stop words depends on the application. The web page http://www.ranks.nl/stopwords shows several common stop word lists, including the one built into the database software SQL and the list claimed to have been used in early versions of Google search. (Modern Google search does not appear to filter any stop words.)

language text.

All of these cleaning steps reduce the number of unique language elements we must consider and thus the dimensionality of the data. This can provide a massive computational benefit, and it is also often key to getting more interpretable model fits (e.g., in topic modeling). However, each of these steps requires careful decisions about the elements likely to carry meaning in a particular application. [2] One researcher's stop words are another's subject of interest. Dropping numerals from political text means missing references to "the first 100 days" or "September 11." In online communication, even punctuation can no longer be stripped without potentially significant information loss `:-(`.

2.3 *N*-grams

Producing a tractable representation also requires that we limit dependence among language elements. A fairly mild step in this direction, for example, might be to parse documents into distinct sentences, and encode features of these sentences while ignoring the order in which they occur. The most common methodologies go much further.

The simplest and most common way to represent a document is called *bag-of-words*. The order of words is ignored altogether, and $\mathbf{c}_i$ is a vector whose length is equal to the number of words in the vocabulary and whose elements c_{ij} are the number of times word j occurs in document i. Suppose that the text of document i is

Good night, good night! Parting is such sweet sorrow.

After stemming, removing stop words, and removing punctuation, we might be left with "good night good night part sweet sorrow." The bag-of-words representation would then have $c_{ij} = 2$ for $j \in \{good, night\}$, $c_{ij} = 1$ for $j \in \{part, sweet, sorrow\}$, and $c_{ij} = 0$ for all other words in the vocabulary.

This scheme can be extended to encode a limited amount of dependence by counting unique phrases rather than unique words. A phrase of length n is referred to as an n-gram. For example, in our snippet above the count of 2-grams (or "bigrams") would have $c_{ij} = 2$ for $j = good.night$,

[2] Denny and Spirling (2018) discuss the sensitivity of unsupervised text analysis methods such as topic modeling to preprocessing steps.

$c_{ij} = 1$ for j including *night.good, night.part, part.sweet,* and *sweet.sorrow*, and $c_{ij} = 0$ for all other possible 2-grams. The bag-of-words representation then corresponds to counts of 1-grams.

Counting n-grams of order $n > 1$ yields data that describes a limited amount of the dependence between words. Specifically, the n-gram counts are sufficient for estimation of an n-order homogeneous Markov model across words (i.e., the model that arises if we assume that word choice is only dependent upon the previous n words). This can lead to richer modeling. In analysis of partisan speech, for example, single words are often insufficient to capture the patterns of interest: "death tax" and "tax break" are phrases with strong partisan overtones that are not evident if we look at the single words "death," "tax," and "break" (see, e.g., Gentzkow and Shapiro 2010).

Unfortunately, the dimension of $\mathbf{c}_i$ increases exponentially quickly with the order n of the phrases tracked. The majority of text analyses consider n-grams up to two or three at most, and the ubiquity of these simple representations (in both machine learning and social science) reflects a belief that the return to richer n-gram modeling is usually small relative to the cost. Best practice in many cases is to begin analysis by focusing on single words. Given the accuracy obtained with words alone, one can then evaluate if it is worth the extra time to move on to 2-grams or 3-grams.

2.4 Richer representations

While rarely used in the social science literature to date, there is a vast array of methods from computational linguistics that capture richer features of text and may have high return in certain applications. One basic step beyond the simple n-gram counting above is to use sentence syntax to inform the text tokens used to summarize a document. For example, Goldberg and Orwant (2013) describe syntactic n-grams where words are grouped together whenever their meaning depends upon each other, according to a model of language syntax.

An alternative approach is to move beyond treating documents as counts of language tokens, and to instead consider the *ordered* sequence of transitions between words. In this case, one would typically break the document into sentences, and treat each as a separate unit for analysis. A single sentence of length s (i.e., containing s words) is then represented as a binary $p \times s$ matrix $\mathbf{S}$, where the nonzero elements of $\mathbf{S}$ indicate occurrence of the row-word in the column-position within the sentence, and p is the length of the vocabulary. Such representations lead to a massive increase in

the dimensions of the data to be modeled, and analysis of this data tends to proceed through *word-embedding*: the mapping of words to a location in $\mathbb{R}^K$ for some $K \ll p$, such that the sentences are then sequences of points in this K dimensional space. This is discussed in detail in Section 3.3.

2.5 Other practical considerations

It is worth mentioning two details that can cause practical social science applications of these methods to diverge a bit from the ideal case considered in the statistics literature. First, researchers sometimes receive data in a pre-aggregated form. In the analysis of Google searches, for example, one might observe the number of searches containing each possible keyword on each day, but not the raw text of the individual searches. This means documents must be similarly aggregated (to days rather than individual searches), and it also means that the natural representation where c_{ij} is the number of occurrences of word j on day i is not available. This is probably not a significant limitation, as the missing information (how many times per search a word occurs conditional on occurring at least once) is unlikely to be essential, but it is useful to note when mapping practice to theory.

A more serious issue is that researchers sometimes do not have direct access to the raw text and must access it through some interface such as a search engine. For example, Gentzkow and Shapiro (2010) count the number of newspaper articles containing partisan phrases by entering the phrases into a search interface (e.g., for the database ProQuest) and counting the number of matches they return. Baker et al. (2016) perform similar searches to count the number of articles mentioning terms related to policy uncertainty. Saiz and Simonsohn (2013) count the number of web pages measuring combinations of city names and terms related to corruption by entering queries in a search engine. Even if one can automate the searches in these cases, it is usually not feasible to produce counts for very large feature sets (e.g., every two-word phrase in the English language), and so the initial feature selection step must be relatively aggressive. Relatedly, interacting through a search interface means that there is no simple way to retrieve objects like the set of all words occurring at least 20 times in the corpus of documents, or the inputs to computing tf-idf.

3 Statistical methods

This section considers methods for mapping the document-token matrix $\mathbf{C}$ to predictions $\hat{\mathbf{V}}$ of an attribute $\mathbf{V}$. In some cases, the observed data is partitioned into submatrices $\mathbf{C}^{train}$ and $\mathbf{C}^{test}$, where the matrix $\mathbf{C}^{train}$ collects rows for which we have observations $\mathbf{V}^{train}$ of $\mathbf{V}$ and the matrix $\mathbf{C}^{test}$ collects rows for which $\mathbf{V}$ is unobserved. The dimension of $\mathbf{C}^{train}$ is $n^{train} \times p$ and the dimension of $\mathbf{V}^{train}$ is $n^{train} \times k$, where k is the number of attributes we wish to predict.

Attributes in $\mathbf{V}$ can include observable quantities such as the frequency of flu cases, the positive or negative rating of movie reviews, or the unemployment rate, about which the documents are informative. There can also be latent attributes of interest, such as the topics being discussed in a congressional debate or in news articles.

Methods to connect counts $\mathbf{c}_i$ to attributes $\mathbf{v}_i$ can be roughly divided into four categories. The first, which we will call *dictionary-based* methods, do not involve statistical inference at all: they simply specify $\hat{\mathbf{v}}_i = f(\mathbf{c}_i)$ for some known function $f(\cdot)$. This is by far the most common method in the social science literature using text to date. In some cases, researchers define $f(\cdot)$ based on a pre-specified dictionary of terms capturing particular categories of text. In Tetlock (2007), for example, $\mathbf{c}_i$ is a bag-of-words representation and the outcome of interest $\mathbf{v}_i$ is the latent "sentiment" of *Wall Street Journal* columns, defined along a number of dimensions such as "positive," "optimistic," and so on. The author defines the function $f(\cdot)$ using a dictionary called the General Inquirer, which provides lists of words associated with each of these sentiment categories.[3] The elements of $f(\mathbf{c}_i)$ are defined to be the sum of the counts of words in each category. (As we discuss below, the main analysis then focuses on the first principal component of the resulting counts.) In Baker et al. (2016), $\mathbf{c}_i$ is the count of articles in a given newspaper-month containing a set of pre-specified terms such as "policy," "uncertainty," and "Federal Reserve," and the outcome of interest $\mathbf{v}_i$ is the degree of "policy uncertainty" in the economy. The authors define $f(\cdot)$ to be the raw count of the pre-specified terms divided by the total number of articles in the newspaper-month, averaged across newspapers. We do not provide additional discussion of dictionary-based methods in this section, but we return to them in Section 3.5 below and in our discussion of applications in Section 4.

[3] http://www.wjh.harvard.edu/~inquirer/

The second and third groups of methods are distinguished by whether they begin from a model of $\mathrm{p}(\mathbf{v}_i|\mathbf{c}_i)$ or a model of $\mathrm{p}(\mathbf{c}_i|\mathbf{v}_i)$. In the former case, which we will call *text regression* methods, we directly estimate the conditional outcome distribution, usually via the conditional expectation $\mathbb{E}[\mathbf{v}_i|\mathbf{c}_i]$ of attributes $\mathbf{v}_i$. This is intuitive: if we want to predict $\mathbf{v}_i$ from $\mathbf{c}_i$, we would naturally regress the observed values of the former ($\mathbf{V}^{train}$) on the corresponding values of the latter ($\mathbf{C}^{train}$). Any generic regression technique can be applied, depending upon the nature of $\mathbf{v}_i$. However, the high-dimensionality of $\mathbf{c}_i$, where p is often as large as or larger than n^{train}, requires use of regression techniques appropriate for such a setting, such as penalized linear or logistic regression.

In the latter case, we begin from a *generative model* of $\mathrm{p}(\mathbf{c}_i|\mathbf{v}_i)$. To see why this is intuitive, note that in many cases the underlying causal relationship runs from outcomes to language rather than the other way around. For example, Google searches about flu do not cause flu cases to occur; rather, people with flu are more likely to produce such searches. Congresspeople's ideology is not determined by their use of partisan language; rather, people who are more conservative or liberal to begin with are more likely to use such language. From an economic point of view, the correct "structural" model of language in these cases maps from $\mathbf{v}_i$ to $\mathbf{c}_i$, and as in other cases familiar to economists modeling the underlying causal relationships can provide powerful guidance to inference and make the estimated model more interpretable.

Generative models can be further divided by whether the attributes are observed or latent. In the first case of *unsupervised* methods, we do not observe the true value of $\mathbf{v}_i$ for any documents. The function relating $\mathbf{c}_i$ and $\mathbf{v}_i$ is unknown, but we are willing to impose sufficient structure on it to allow us to infer $\mathbf{v}_i$ from $\mathbf{c}_i$. This class includes methods such as topic modeling and its variants (e.g., latent Dirichlet allocation, or LDA). In the second case of *supervised* methods, we observe training data $\mathbf{V}^{train}$ and we can fit our model, say $f_{\boldsymbol{\theta}}(\mathbf{c}_i;\mathbf{v}_i)$ for a vector of parameters $\boldsymbol{\theta}$, to this training set. The fitted model $f_{\hat{\boldsymbol{\theta}}}$ can then be *inverted* to predict $\mathbf{v}_i$ for documents in the test set and can also be used to interpret the structural relationship between attributes and text. Finally, in some cases, $\mathbf{v}_i$ includes both observed and latent attributes for a *semi-supervised* analysis.

Lastly, we discuss *word embeddings*, which provide a richer representation of the underlying text than the token counts that underlie other methods. They have seen limited application in economics to date, but their dramatic successes in deep learning and other machine learning domains suggest they are likely to have high value in the future.

We close in Section 3.5 with some broad recommendations for practitioners.

3.1 Text regression

Predicting an attribute $\mathbf{v}_i$ from counts $\mathbf{c}_i$ is a regression problem like any other, except that the high-dimensionality of $\mathbf{c}_i$ makes OLS and other standard techniques infeasible. The methods in this section are mainly applications of standard high-dimensional regression methods to text.

3.1.1 Penalized linear models

The most popular strategy for very high-dimensional regression in contemporary statistics and machine learning is the estimation of *penalized* linear models, particularly with L_1 penalization. We recommend this strategy for most text regression applications: linear models are intuitive and interpretable, fast high-quality software is available for big sparse input matrices like our $\mathbf{C}$. For simple text-regression tasks with input dimension on the same order as the sample size, penalized linear models typically perform close to the frontier in terms of out-of-sample prediction.

Linear models in the sense we mean here are those in which $\mathbf{v}_i$ depends on $\mathbf{c}_i$ only through a linear index $\eta_i = \alpha + \mathbf{x}_i'\boldsymbol{\beta}$, where $\mathbf{x}_i$ is a known transformation of $\mathbf{c}_i$. In many cases, we simply have $\mathbb{E}[\mathbf{v}_i|\mathbf{x}_i] = \eta_i$. It is also possible that $\mathbb{E}[\mathbf{v}_i|\mathbf{x}_i] = f(\eta_i)$ for some known link function $f(\cdot)$, as in the case of logistic regression.

Common transformations are the identity $\mathbf{x}_i = \mathbf{c}_i$, normalization by document length $\mathbf{x}_i = \mathbf{c}_i/m_i$ with $m_i = \sum_j c_{ij}$, or the positive indicator $x_{ij} = \mathbb{1}_{[c_{ij}>0]}$. The best choice is application-specific, and may be driven by interpretability; does one wish to interpret β_j as the added effect of an extra count for token j (if so, use $x_{ij} = c_{ij}$) or as the effect of the presence of token j (if so, use $x_{ij} = \mathbb{1}_{[c_{ij}>0]}$)? The identity is a reasonable default in many settings.

Write $l(\alpha, \boldsymbol{\beta})$ for an unregularized objective proportional to the negative log likelihood, $-\log \mathrm{p}(\mathbf{v}_i|\mathbf{x}_i)$. For example, in Gaussian (linear) regression, $l(\alpha, \boldsymbol{\beta}) = \sum_i (\mathbf{v}_i - \eta_i)^2$ and in binomial (logistic) regression, $l(\alpha, \boldsymbol{\beta}) = -\sum_i [\eta_i \mathbf{v}_i - \log(1 + e^{\eta_i})]$ for $\mathbf{v}_i \in \{0, 1\}$. A penalized estimator is then the solution to

$$\min \left\{ \; l(\alpha, \boldsymbol{\beta}) + n\lambda \sum_{j=1}^{p} \kappa_j(|\beta_j|) \; \right\}, \tag{1}$$

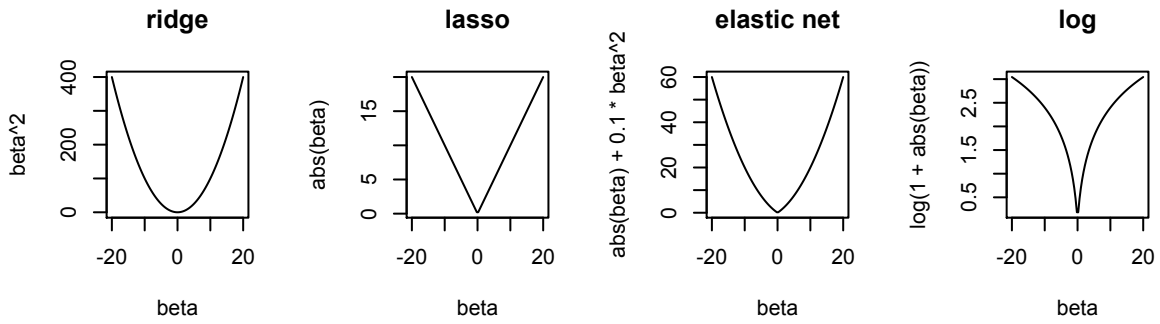


Figure 1: From left to right, L_2 costs (ridge, Hoerl and Kennard 1970), L_1 (lasso, Tibshirani 1996), the "elastic net" mixture of L_1 and L_2 (Zou and Hastie 2005), and the log penalty (Candes et al. 2008).

where $\lambda > 0$ controls overall penalty magnitude and $\kappa_j(\cdot)$ are increasing "cost" functions that penalize deviations of the β_j from zero.

A few common cost functions are shown in Figure 1. Those that have a non-differentiable spike at zero (lasso, elastic net, and log) lead to sparse estimators, with some coefficients set to exactly zero. The curvature of the penalty away from zero dictates the weight of shrinkage imposed on the nonzero coefficients: L_2 costs increase with coefficient size; lasso's L_1 penalty has zero curvature and imposes constant shrinkage, and as curvature goes towards $-\infty$ one approaches the L_0 penalty of subset selection. The lasso's L_1 penalty (Tibshirani 1996) is extremely popular: it yields sparse solutions with a number of desirable properties (e.g., Bickel et al. 2009; Wainwright 2009; Belloni et al. 2013; Bühlmann and Van De Geer 2011), and the number of nonzero estimated coefficients is an unbiased estimator of the regression degrees of freedom (which is useful in model selection; see Zou et al. 2007).[4]

Focusing on L_1 regularization, re-write the penalized linear model objective as

$$\min \left\{ l(\alpha, \boldsymbol{\beta}) + n\lambda \sum_{j=1}^{p} \omega_j |\beta_j| \right\}. \tag{2}$$

A common strategy sets ω_j so that the penalty cost for each coefficient is scaled by the sample standard deviation of that covariate. In text analysis, where each covariate corresponds to some

[4]Penalties with a bias that diminishes with coefficient size, such as the log penalty in Figure 1 (Candes et al. 2008), the smoothly clipped absolute deviation (SCAD) of Fan and Li (2001), or the adaptive lasso of Zou (2006), have been promoted in the statistics literature as improving upon the lasso by providing consistent variable selection and estimation in a wider range of settings. These diminishing bias penalties lead to increased computation costs (due to a non-convex loss), but there exist efficient approximation algorithms (see, e.g., Fan et al. 2014; Taddy 2017b).

transformation of a specific text token, this type of weighting is referred to as "rare feature up-weighting" (e.g., Manning et al. 2008) and is generally thought of as good practice: rare words are often most useful in differentiating between documents.[5]

Large λ leads to simple model estimates in the sense that most coefficients will be set at or close to zero, while as $\lambda \to 0$ we approach maximum likelihood estimation (MLE). Since there is no way to define an optimal λ *a priori*, standard practice is to compute estimates for a large set of possible λ and then use some criterion to select the one that yields the best fit.

Several criteria are available to choose an optimal λ. One common approach is to leave out part of the training sample in estimation and then choose the λ that yields the best out-of-sample fit according to some criterion such as mean squared error. Rather than work with a single leave-out sample, researchers most often use K-fold cross-validation (CV). This splits the sample into K disjoint subsets, and then fits the full regularization path K times excluding each subset in turn. This yields K realizations of the mean squared error or other out-of-sample fit measure for each value of λ. Common rules are to select the value of λ which minimizes the average error across these realizations, or (more conservatively) to choose the largest λ with mean error no more than one standard error away from the minimum.

Analytic alternatives to cross-validation are Akaike's information criterion (AIC; Akaike 1973) and the Bayesian information criterion (BIC) of Schwarz (1978). In particular, Flynn et al. (2013) describe a bias-corrected AIC objective for high-dimensional problems that they call AICc. It is motivated as an approximate likelihood maximization subject to a degrees of freedom (df_λ) adjustment: $AICc(\lambda) = 2l(\alpha_\lambda, \boldsymbol{\beta}_\lambda) + 2df_\lambda \frac{n}{n-df_\lambda-1}$. Similarly, the BIC objective is $BIC(\lambda) = l(\alpha_\lambda, \boldsymbol{\beta}_\lambda) + df_\lambda \log n$, and is motivated as an approximation to the Bayesian posterior marginal likelihood in Kass and Wasserman (1995). AICc and BIC selection choose λ to minimize their respective objectives. The BIC tends to choose simpler models than cross-validation or AICc. Zou et al. (2007) recommend BIC for lasso penalty selection whenever variable selection, rather than predictive performance, is the primary goal.

[5]This is the same principle which motivates "inverse-document frequency" weighting schemes, such as tf-idf.

3.1.2 Dimension reduction

Another common solution for taming high dimensional prediction problems is to form a small number of linear combinations of predictors and to use these derived indices as variables in an otherwise standard predictive regression. Two classic dimension reduction techniques are *principal components regression* (PCR) and *partial least squares* (PLS).

Penalized linear models use shrinkage and variable selection to manage high dimensionality by forcing the coefficients on most regressors to be close to (or, for lasso, exactly to) zero. This can produce suboptimal forecasts when predictors are highly correlated. A transparent illustration of this problem would be a case in which all of the predictors are equal to the forecast target plus an i.i.d. noise term. In this situation, choosing a subset of predictors via lasso penalty is inferior to taking a simple average of the predictors and using this as the sole predictor in a univariate regression. This predictor averaging, as opposed to predictor selection, is the essence of dimension reduction.

PCR consists of a two-step procedure. In the first step, principal components analysis (PCA) combines regressors into a small set of K linear combinations that best preserve the covariance structure among the predictors. This amounts to solving the problem

$$\min_{\Gamma,B} \text{trace}\left[(C-\Gamma B')(C-\Gamma B')'\right], \text{ s.t. } \text{rank}(\Gamma) = \text{rank}(B) = K \tag{3}$$

The count matrix $\mathbf{C}$ consists of n rows (one for each document) and p columns (one for each term). PCA seeks a low rank representation $\Gamma B'$ that best approximates the text data $\mathbf{C}$. This formulation has the character of a factor model. The $n \times K$ matrix Γ captures the prevalence of K common components, or "factors," in each document. The $p \times K$ matrix B describes the strength of association between each word and the factors. As we will see, this reduced-rank decomposition bears a close resemblance to other text analytic methods such as topic modeling and word embeddings.

In the second step, the K components are used in standard predictive regression. As an example, Foster et al. (2013) use PCR to build a hedonic real estate pricing model that takes textual content of property listings as an input.[6] With text data, where the number of features tend to vastly exceed

[6]See Stock and Watson (2002a,b) for development of the PCR estimator and an application to macroeconomic

the observation count, regularized versions of PCA such as predictor thresholding (e.g., Bai and Ng 2008) and sparse PCA (Zou et al. 2006) help exclude the least informative features to improve predictive content of the dimension-reduced text.

A drawback of PCR is that it fails to incorporate the ultimate statistical objective—forecasting a particular set of attributes—in the dimensionality reduction step. PCA condenses text data into indices based on the covariation *among* the predictors. This happens prior to the forecasting step and without consideration of how predictors associate with the forecast target.

In contrast, partial least squares performs dimension reduction by directly exploiting covariation of predictors with the forecast target.[7] Suppose we are interested in forecasting a scalar attribute v_i. PLS regression proceeds as follows. For each element j of the feature vector $\mathbf{c}_i$, estimate the univariate covariance between v_i on c_{ij}. This covariance, denoted φ_j, reflects the attribute's "partial" sensitivity to each feature j. Next, form a single predictor by averaging all attributes into a single aggregate predictor $\hat{v}_i = \sum_j \varphi_j c_{ij} / \sum_j \varphi_j$. This forecast places the highest weight on the strongest univariate predictors, and the least weight on the weakest. In this way, PLS performs its dimension reduction with the ultimate forecasting objective in mind. The description of $\hat{v}_i$ reflects the $K = 1$ case, i.e. when text is condensed into a single predictive index. To use additional predictive indices, both v_i on c_{ij} are orthogonalized with respect to $\hat{v}_i$, the above procedure is repeated on the orthogonalized dataset, and the resulting forecast is added to the original $\hat{v}_i$. This is iterated until the desired number of PLS components K is reached. Like PCR, PLS components describe the prevalence of K common factors in each document. And also like PCR, PLS can be implemented with a variety of regularization schemes to aid its performance in the ultra-high dimensional world of text. Section 4 discusses applications using PLS in text regression.

PCR and PLS share a number of common properties. In both cases, K is a user-controlled parameter which, in many social science applications, is selected ex ante by the researcher. But, like any hyperparameter, K can be tuned via cross-validation. And neither method is scale invariant—the forecasting model is sensitive to the distribution of predictor variances. It is therefore common to variance-standardize features before applying PCR or PLS.

forecasting with a large set of numerical predictors.

[7] See Kelly and Pruitt (2013, 2015) for the asymptotic theory of PLS regression and its application to forecasting risk premia in financial markets.

3.1.3 Nonlinear text regression

Penalized linear models are the most widely applied text regression tools due to their simplicity, and because they may be viewed as a first-order approximation to potentially nonlinear and complex data generating processes. In cases where a linear specification is too restrictive, there are several other machine learning tools that are well suited to represent nonlinear associations between text $\mathbf{c}_i$ and outcome attributes $\mathbf{v}_i$. Here we briefly describe four such nonlinear regression methods—generalized linear models, support vector machines, regression trees, and deep learning—and provide references for readers interested in thorough treatments of each.

GLMs and SVMs. One way to capture nonlinear associations between $\mathbf{c}_i$ and $\mathbf{v}_i$ is with a generalized linear model (GLM). These expand the linear model to include nonlinear functions of $\mathbf{c}_i$ such as polynomials or interactions, while otherwise treating the problem with the penalized linear regression methods discussed above.

A related method used in the social science literature is the support vector machine, or SVM (Vapnik 1996). This is used for text classification problems (when $\mathbf{V}$ is categorical), the prototypical example being email spam filtering. A detailed discussion of SVMs is beyond the scope of this review, but from a high level, the SVM finds hyperplanes in a basis expansion of $\mathbf{C}$ that partition the observations into sets with equal response (i.e., so that $\mathbf{v}_i$ are all equal in each region).[8]

GLMs and SVMs both face the limitation that, without a priori assumptions for which basis transformations and interactions to include, they may overfit and require extensive tuning (Hastie et al. 2009; Murphy 2012). For example, multi-way interactions increase the parameterization combinatorially and can quickly overwhelm the penalization routine, and their performance suffers in the presence of many spurious "noise" inputs (Hastie et al. 2009).[9]

Regression Trees. Regression trees have become a popular nonlinear approach for incorporating multi-way predictor interactions into regression and classification problems. The logic of trees differs markedly from traditional regressions. A tree "grows" by sequentially sorting data

[8] Hastie et al. (2009 chap.12) and Murphy (2012 chap. 14) provide detailed overviews of GLMs and SVMs. Joachims (1998) and Tong and Koller (2001) (among others) study text applications of SVMs.

[9] Another drawback of SVMs is that they cannot be easily connected to the estimation of a probabilistic model and the resulting fitted model can sometimes be difficult to interpret. Polson and Scott (2011) provide a pseudo-likelihood interpretation for a variant of the SVM objective. Our own experience has led us to lean away from SVMs for text analysis in favor of more easily interpretable models. Murphy (2012 chap. 14.6) attributes the popularity of SVMs in some application areas to an ignorance of alternatives.

observations into bins based on values of the predictor variables. This partitions the dataset into rectangular regions, and forms predictions as the average value of the outcome variable within each partition (Breiman et al. 1984). This structure is an effective way to accommodate rich interactions and nonlinear dependencies.

Two extensions of the simple regression tree have been highly successful thanks to clever regularization approaches that minimize the need for tuning and avoid overfitting. Random forests (Breiman 2001) average predictions from many trees that have been randomly perturbed in a bootstrap step. Boosted trees (e.g., Friedman 2002) recursively combine predictions from many oversimplified trees.[10]

The benefits of regression trees—nonlinearity and high-order interactions—are sometimes lessened in the presence of high-dimensional inputs. While we would generally recommend tree models, and especially random forests, they are often not worth the effort for simple text regression. Often times, a more beneficial use of trees is in a final prediction step after some dimension reduction derived from the generative models in Section 3.2.

Deep Learning. There is a host of other machine learning techniques that have been applied to text regression. The most common techniques not mentioned thus far are neural networks, which typically allow the inputs to act on the response through one or more layers of interacting nonlinear basis functions (e.g., see Bishop 1995). A main attraction of neural networks is their status as *universal approximators*, a theoretical result describing their ability to mimic general, smooth nonlinear associations.

In high-dimensional and very noisy settings, such as in text analysis, classical neural nets tend to suffer from the same issues referenced above: they often overfit and are difficult to tune. However, the recently popular "deep" versions of neural networks (with many layers, and fewer nodes per layer) incorporate a number of innovations that allow them to work better, faster, and with little tuning, even in difficult text analysis problems. Such deep neural nets (DNNs) are now the state-of-the-art solution for many machine learning tasks (LeCun et al. 2015).[11] DNNs are now employed in many complex natural language processing tasks, such as translation (Sutskever et al.

[10]Hastie et al. (2009) provide an overview of these methods. In addition, see Wager et al. (2014) and Wager and Athey (2017) for results on confidence intervals for random forests, and see Taddy et al. (2015) and Taddy et al. (2016) for interpretation of random forests as a Bayesian posterior over potentially optimal trees.

[11]Goodfellow et al. (2016) provide a thorough textbook overview of these "deep learning" technologies, while Goldberg (2016) is an excellent primer on their use in natural language processing.

2014; Wu et al. 2016) and syntactic parsing (Chen and Manning 2014), as well as in exercises of relevance to social scientists—for example, Iyyer et al. (2014) infer political ideology from text using a DNN. They are frequently used in conjunction with richer text representations such as word embeddings, described more below.

3.1.4 Bayesian regression methods

The penalized methods above can all be interpreted as posterior maximization under some prior. For example, ridge regression maximizes the posterior under independent Gaussian priors on each coefficient while Park and Casella (2008) and Hans (2009) give Bayesian interpretations to the lasso. See also the horseshoe of Carvalho et al. (2010) and the double Pareto of Armagan et al. (2013) for Bayesian analogues of diminishing bias penalties like the log penalty on the right of Figure 1.

For those looking to do a full Bayesian analysis for high-dimensional (e.g., text) regression, an especially appealing model is the spike-and-slab introduced in George and McCulloch (1993). This models the distribution over regression coefficients as a mixture between two densities centered at zero—one with very small variance (the spike) and another with large variance (the slab). This model allows one to compute posterior variable inclusion probabilities as, for each coefficient, the posterior probability that it came from the slab and not spike component. Due to a need to integrate over the posterior distribution, e.g., via Markov chain Monte Carlo (MCMC), inference for spike-and-slab models is much more computationally intensive than fitting the penalized regressions of Section 3.1.1. However, Yang et al. (2016) argue that spike-and-slab estimates based on short MCMC samples can be useful in application, while Scott and Varian (2014) have engineered efficient implementations of the spike-and-slab model for big data applications. These procedures give a full accounting of parameter uncertainty, which we miss in a quick penalized regression.

3.2 Generative language models

Text regression treats the token counts as generic high-dimensional input variables, without any attempt to model structure that is specific to language data. In many settings it is useful to instead propose a generative model for the text tokens to learn about how the attributes influence

word choice and account for various dependencies between words and between attributes. In this approach, the words in a document are viewed as the realization of a generative process defined through a probability model for $\mathrm{p}(\mathbf{c}_i|\mathbf{v}_i)$.

3.2.1 Unsupervised generative models

In the unsupervised setting we have no direct observations of the true attributes $\mathbf{v}_i$. Our inference about these attributes must therefore depend entirely on strong assumptions that we are willing to impose on the structure of the model $\mathrm{p}(\mathbf{c}_i|\mathbf{v}_i)$. Examples in the broader literature include cases where the $\mathbf{v}_i$ are latent factors, clusters, or categories. In text analysis, the leading application has been the case in which the $\mathbf{v}_i$ are topics.

A typical generative model implies that each observation $\mathbf{c}_i$ is a conditionally independent draw from the vocabulary of possible tokens according to some document-specific token probability vector, say $\mathbf{q}_i = [q_{i1} \dots q_{ip}]'$. Conditioning on document length, $m_i = \sum_j c_{ij}$, this implies a *multinomial* distribution for the counts

$$\mathbf{c}_i \sim \mathrm{MN}(\mathbf{q}_i, m_i). \tag{4}$$

This multinomial model underlies the vast majority of contemporary generative models for text.

Under the basic model in (4), the function $\mathbf{q}_i = q(\mathbf{v}_i)$ links attributes to the distribution of text counts. A leading example of this link function is the *topic model* specification of Blei et al. (2003),[12] where

$$\mathbf{q}_i = v_{i1}\boldsymbol{\theta}_1 + v_{i2}\boldsymbol{\theta}_2 + \ldots + v_{ik}\boldsymbol{\theta}_k = \boldsymbol{\Theta}\mathbf{v}_i. \tag{5}$$

Many readers will recognize the model in (5) as a *factor model* for the vector of normalized counts for each token in document i, $\mathbf{c}_i/m_i$. Indeed, a topic model is simply a factor model for multinomial data. Each *topic* is a probability vector over possible tokens, denoted $\boldsymbol{\theta}_l$, $l = 1, \ldots, k$ (where $\theta_{lj} \geq 0$ and $\sum_{j=1}^{p} \theta_{lj} = 1$). A topic can be thought of as a cluster of tokens that tend to appear in documents. The latent attribute vector $\mathbf{v}_i$ is referred to as the set of *topic weights* (formally, a distribution over topics, $v_{il} \geq 0$ and $\sum_{l=1}^{k} v_{il} = 1$). $\mathbf{v}_{il}$ describes the proportion of language in document i devoted

[12] Standard least-squares factor models have long been employed in "latent semantic analysis" (LSA; Deerwester et al. 1990), which applies PCA (i.e., singular value decompositions) to token count transformations such as $\mathbf{x}_i = \mathbf{c}_i/m_i$ or $x_{ij} = c_{ij}\log(d_j)$ where $d_j = \sum_i \mathbb{1}_{[c_{ij}>0]}$. Topic modeling and its precursor, probabilistic LSA, are generally seen as improving on such approaches by replacing arbitrary transformations with a plausible generative model.

to the l^{th} topic. We can allow each document to have a mix of topics, or we can require that one $v_{il} = 1$ while the rest are zero, so that each document has a single topic. [13]

Since its introduction into text analysis, topic modeling has become hugely popular.[14] (See Blei 2012 for a high-level overview.) The model has been especially useful in political science (e.g., Grimmer 2010), where researchers have been successful in attaching political issues and beliefs to the estimated latent topics.

Since the $\mathbf{v}_i$ are of course latent, estimation for topic models tends to make use of some alternating inference for $\mathbf{V}|\boldsymbol{\Theta}$ and $\boldsymbol{\Theta}|\mathbf{V}$. One possibility is to employ a version of the expectation-maximization algorithm (EM) to either maximize the likelihood implied by (4) and (5) or, after incorporating the usual Dirichlet priors on $\mathbf{v}_i$ and $\boldsymbol{\theta}_l$, to maximize the posterior; this is the approach taken in Taddy (2012; see this paper also for a review of topic estimation techniques). Alternatively, one can target the full posterior distribution $\mathrm{p}(\boldsymbol{\Theta}, \mathbf{V} \mid \mathbf{c}_i)$. Estimation, say for $\boldsymbol{\Theta}$, then proceeds by maximization of the estimated *marginal* posterior, say $\mathrm{p}(\boldsymbol{\Theta} \mid \mathbf{c}_i)$.

Due to the size of the datasets and dimension of the models, posterior approximation for topic models usually uses some form of variational inference (Wainwright and Jordan 2008) that fits a tractable parametric family to be as close as possible (e.g., in Kullback-Leibler divergence) from the true posterior. This variational approach was used in the original Blei et al. (2003) paper and in many applications since. Hoffman et al. (2013) present a *stochastic variational inference* algorithm that takes advantage of techniques for optimization on massive data; this algorithm is used in many contemporary topic modeling applications. Another approach, which is more computationally intensive but can yield more accurate posterior approximations, is the MCMC algorithm of Griffiths and Steyvers (2004). Alternatively, for quick estimation without uncertainty quantification, the posterior maximization algorithm of Taddy (2012) is a good option.

The choice of k, the number of topics, is often fairly arbitrary. Data-driven choices do exist: Taddy (2012) describes a model selection process for k that is based upon Bayes factors, Airoldi et al. (2010) provide a cross-validation (CV) scheme, while Teh et al. (2006) use Bayesian non-

[13] Topic modeling is alternatively labeled as "latent Dirichlet allocation," (LDA) which refers to the Bayesian model in Blei et al. (2003) that treats each $\mathbf{v}_i$ and $\boldsymbol{\theta}_l$ as generated from a Dirichlet-distributed prior. Another specification that is popular in political science (e.g., Quinn et al. 2010) keeps $\boldsymbol{\theta}_l$ as Dirichlet-distributed but requires each document to have a single topic. This may be most appropriate for short documents, such a press releases or single speeches.

[14] The same model was independently introduced in genetics by Pritchard et al. (2000) for factorizing gene expression as a function of latent populations; it has been similarly successful in that field. Latent Dirichlet allocation is also an extension of a related mixture modeling approach in the latent semantic analysis of Hofmann (1999).

parametric techniques that view k as an unknown model parameter. In practice, however, it is very common to simply start with a number of topics on the order of ten, and then adjust the number of topics in whatever direction seems to improve interpretability. Whether this *ad hoc* procedure is problematic depends on the application. As we discuss below, in many applications of topic models to date the goal is to provide an intuitive description of text rather than inference on some underlying "true" parameters; in these cases, the *ad hoc* selection of the number of topics may be reasonable.

The basic topic model has been generalized and extended in variety of ways. A prominent example is the dynamic topic model of Blei and Lafferty (2006), which considers documents that are indexed by date (e.g., publication date for academic articles) and allows the topics, say $\mathbf{\Theta}_t$, to evolve smoothly in time. Another example is the supervised topic model of McAuliffe and Blei (2008), which combines the standard topic model with an extra equation relating the weights $\mathbf{v}_i$ to some additional attribute y_i in $\mathrm{p}(y_i|\mathbf{v}_i)$. This pushes the latent topics to be relevant to y_i as well as the text $\mathbf{c}_i$. In these and many other extensions, the modifications are designed to incorporate available document metadata (in these examples, time and y_i respectively).

3.2.2 Supervised generative models

In supervised models, the attributes $\mathbf{v}_i$ are observed in a training set and thus may be directly harnessed to inform the model of text generation. Perhaps the most common supervised generative model is the so-called naive Bayes classifier (e.g., Murphy 2012) which treats counts for each token as independent with class dependent means. For example, the observed attribute might be author identity for each document in the corpus with the model specifying different mean token counts for each author.

In naive Bayes, $\mathbf{v}_i$ is a univariate categorical variable and the token count distribution is factorized as $\mathrm{p}(\mathbf{c}_i|\mathbf{v}_i) = \prod_j \mathrm{p}_j(c_{ij}|\mathbf{v}_i)$, thus "naively" specifying conditional independence between tokens j. This rules out the possibility that by choosing to say one token (say, "hello") we reduce the probability that we say some other token (say, "hi"). The parameters of each independent token distribution are estimated, yielding $\hat{\mathrm{p}}_j$ for $j = 1 \ldots p$. The model can then be inverted for prediction,

with classification probabilities for the possible class labels obtained via Bayes's rule as

$$\mathrm{p}(\mathbf{V}|\mathbf{c}_i) = \frac{\mathrm{p}(\mathbf{c}_i|\mathbf{V})\pi_v}{\sum_a \mathrm{p}(\mathbf{c}_i|a)\pi_a} = \frac{\prod_j \mathrm{p}_j(c_{ij}|\mathbf{V})\pi_v}{\sum_a \prod_j \mathrm{p}_j(c_{ij}|a)\pi_a} \tag{6}$$

where π_a is the prior probability on class a (usually just one over the number of possible classes). In text analysis, Poisson naive Bayes procedures, with $\mathrm{p}(c_{ij}|\mathbf{V}) = \mathrm{Pois}(c_{ij};\ \theta_{vj})$ where $\mathbb{E}[c_{ij}|\mathbf{V}] = \theta_{vj}$, have been used as far back as Mosteller and Wallace (1963). Some recent social science applications use binomial naive Bayes, which sets $\mathrm{p}(c_{ij}|\mathbf{V}) = \mathrm{Bin}(c_{ij};\ \theta_{vj})$ where $\mathbb{E}[c_{ij}/m_i|\mathbf{V}] = \theta_{vj}$. The Poisson model has some statistical justification in the analysis of text counts (Taddy 2015a), but the binomial specification seems to be more common in off-the-shelf software.

A more realistic sampling model for text token counts is the multinomial model of (4). This introduces limited dependence between token counts, encoding the fact that using one token for a given utterance must slightly lower the expected count for all other tokens. Combining such a sampling scheme with generalized linear models, Taddy (2013b) advocates the use of multinomial logistic regression to connect text counts with observable attributes. The generative model specifies probabilities in the multinomial distribution of (4) as

$$q_{ij} = \frac{e^{\eta_{ij}}}{\sum_{h=1}^{p} e^{\eta_{ih}}}, \quad \eta_{ij} = \alpha_j + \mathbf{v}_i'\boldsymbol{\varphi}_j. \tag{7}$$

Taddy (2013a,b) applies these models in the setting of univariate or low dimensional $\mathbf{v}_i$, focusing on their use for prediction of future document attributes through an inversion strategy discussed below. More recently, Taddy (2015a) provides a distributed-computing strategy that allows the model implied by (7) to be fit (using penalized deviance methods as detailed in Section 3.1.1) for high-dimensional $\mathbf{v}_i$ on massive corpora. This facilitates language models containing a large number of sources of heterogeneity (even document-specific random effects), thus allowing social scientists to apply familiar regression analysis tools in their text analyses.

Application of the logistic regression text models implied by (7) often requires an *inversion* step for inference about attributes conditional on text—that is, to map from $\mathrm{p}(\mathbf{c}_i|\mathbf{v}_i)$ to $\mathrm{p}(\mathbf{v}_i|\mathbf{c}_i)$. The simple Bayes's rule technique of (6) is difficult to apply beyond a single categorical attribute. Instead, Taddy (2013b) uses the inverse regression ideas of Cook (2007) in deriving *sufficient projections* from the fitted models. Say $\boldsymbol{\Phi} = [\boldsymbol{\varphi}_1 \cdots \boldsymbol{\varphi}_p]$ is the matrix of regression coefficients

from (7) across all tokens j; then the token count projection $\mathbf{\Phi c}_i$ is a sufficient statistic for $\mathbf{v}_i$ in the sense that

$$\mathbf{v}_i \perp\!\!\!\perp \mathbf{c}_i \mid \mathbf{\Phi c}_i, \tag{8}$$

i.e., the attributes are independent of the text counts conditional upon the projection. Thus, the fitted logistic regression model yields a map from high-dimensional text to the presumably lower dimensional attributes of interest, and this map can be used instead of the full text counts for future inference tasks. For example, to predict variable $\mathbf{v}_i$ you can fit the low dimensional OLS regression of $\mathbf{v}_i$ on $\mathbf{\Phi c}_i$. Use of projections built in this way is referred to as multinomial inverse regression (MNIR). This idea can also be applied to only a subset of the variables in $\mathbf{v}_i$, yielding projections that are sufficient for the text content relevant to those variables *after* conditioning on the other attributes in $\mathbf{v}_i$. Taddy (2015a) details use of such sufficient projections in a variety of applications, including attribute prediction, treatment effect estimation, and document indexing.

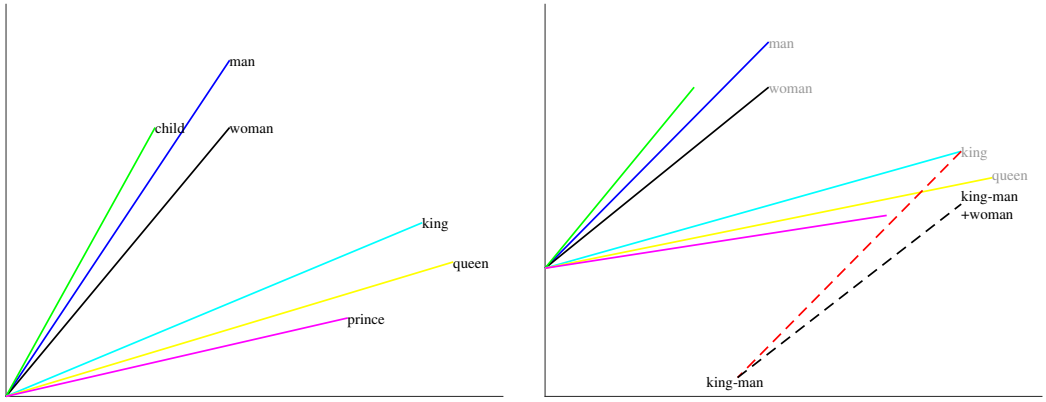
New techniques are arising that combine MNIR techniques with the latent structure of topic models. For example, Rabinovich and Blei (2014) directly combine the logistic regression in (7) with the topic model of (5) in a mixture specification. Alternatively, the structural topic model of Roberts et al. (2013) allows both topic content ($\boldsymbol{\theta}_l$) and topic prevalence (latent $\mathbf{v}_i$) to depend on observable document attributes. Such semi-supervised techniques seem promising for their combination of the strong text-attribute connection of MNIR with topic modeling's ability to account for latent clustering and dependency within documents.

3.3 Word embeddings

Throughout this article, documents have been represented through token count vectors, $\mathbf{c}_i$. This is a crude language summarization. It abstracts from any notion of similarity between words (such as *run, runner, jogger*) or syntactical richness. One of the frontiers of textual analysis is in developing new representations of text data that more faithfully capture its meaning.

Instead of identifying words only as an index for location in a long vocabulary list, imagine representing words as points in a large vector space, with similar words co-located, and an internally consistent arithmetic on the space for relating words to one another. For example, suppose our vocabulary consists of six words: $\{king, queen, prince, man, woman, child\}$. The vector space

Figure 2: A Graphical Example of Word Embeddings



representation of this vocabulary based on similarity of their meaning might look something like the left figure below.[15]

In the vector space, words are relationally oriented and we can begin to draw meaning from term positions, something that is not possible in simple bag of words approaches. For example, in the right figure, we can see that by subtracting the vector *man* from the vector *king*, and then adding to this *woman*, we arrive spatially close to *queen*. Likewise, the combination *king*$-$*man*$+$*child* lies in close proximity to the vector *prince*.

Such *word embeddings*, also known as distributed language representations, amount to a pre-processing of the text data to replace word identities—encoded as binary indicators in a vocabulary-length vector—with an embedding (location) of each vocabulary word in $\mathbb{R}^K$, where K is the dimension of the latent representation space. The dimensions of the vector space corresponds to various aspects of meaning that give words their content. Continuing from the simplified example vocabulary, the latent (and, in reality, unlabeled) dimensions and associated word embeddings might look like:

Dimension	king	queen	prince	man	woman	child
Royalty	0.99	0.99	0.95	0.01	0.02	0.01
Masculinity	0.94	0.06	0.02	0.99	0.02	0.49
Age	0.73	0.81	0.15	0.61	0.68	0.09
...						

[15]This example is motivated by https://blog.acolyer.org/2016/04/21/the-amazing-power-of-word-vectors/.

This type of text representation has long been applied in natural language processing (Rumelhart et al. 1986; Morin and Bengio 2005). The embeddings must be estimated and are chosen to optimize, perhaps approximately, an objective function defined on the original text (such as a likelihood for word occurrences). They form the basis for many deep learning applications involving textual data (see, e.g., Chen and Manning 2014; Goldberg 2016). They are also valuable in their own right for mapping from language to a vector space where we can compute distances and angles between words for fundamental tasks such as classification, and have begun to be adopted by social scientists as a useful summary representation of text data.

Some popular embedding technique are Word2Vec (Mikolov et al. 2013) and Global Vector for Word Representation (GloVe, Pennington et al. 2014). The key preliminary step in these methods is to settle on a notion of *co-occurrence* among terms. For example, consider a $p \times p$ matrix denoted $CoOccur$, whose (i, j) entry counts the number of times in your corpus that the terms i and j appear within, say, b words of each other. This is known as the *skip-gram* definition of co-occurrences.

To embed $CoOccur$ in a K-dimensional vector space, where K is much smaller than p (say a few hundred), we solve the same type of problem that PCA used to summarize the word count matrix in equation (3). In particular, we can find rank-K matrices Γ and B that best approximate co-occurrences among terms:

$$CoOccur \approx \Gamma B'.$$

The j^{th} rows of Γ and B (denoted γ_j and β_j) give a K-dimensional embedding of the j^{th} word, so co-occurrences of terms i and j are approximated as $\gamma_i \beta_j'$. This geometric representation of the text has an intuitive interpretation. The inner product of terms' embeddings, which measures the closeness of the pair in the K-dimensional vector space, describes how likely the pair is to co-occur.

Researchers are beginning to connect these vector-space language models with the sorts of document attributes that are of interest in social science. For example, Le and Mikolov (2014) estimate latent document scores in a vector space, while Taddy (2015b) develops an inversion rule for document classification based upon Word2Vec. In one especially compelling application, Bolukbasi et al. (2016) estimate the direction of *gender* in an embedding space by averaging the angles between female and male descriptors. They then show that stereotypically male and female jobs, for example, live at the corresponding ends of the implied gender vector. This information

is used to derive an algorithm for removing these gender biases, so as to provide a more "fair" set of inputs for machine learning tasks. Approaches like this, which use embeddings as the basis for mathematical analyses of text, can play a role in the next generation of text-as-data applications in social science.

3.4 Uncertainty Quantification

The machine learning literature on text analysis is focused on point estimation and predictive performance. Social scientists often seek to interpret parameters or functions of the fitted models, and hence desire strategies for quantifying the statistical uncertainty around these targets – that is, for statistical inference.

Many machine learning methods for text analysis are based upon a Bayesian modeling approach, where uncertainty quantification is often available as part of the estimation process. In MCMC sampling, as in the Bayesian regression of Scott and Varian (2014) or the topic modeling of Griffiths and Steyvers (2004), the software returns samples from the posterior and thus inference is immediately available. For estimators relying on variational inference—i.e., fitting a tractable distribution as closely as possible to the true posterior—one can simulate from the approximate distribution to conduct inference.[16]

Frequentist uncertainty quantification is often favored by social scientists, but analytic sampling distributions are unavailable for most of the methods discussed here. Some results exist for the lasso in stylized settings (especially Knight and Fu 2000), but these assume low dimensional asymptotic scenarios that may be unrealistic for text analysis. More promising are computation algorithms that approximate the sampling distribution, the most common being the familiar *nonparametric bootstrap* (Efron 1979). This repeatedly draws samples with replacement of the same size as the original dataset and re-estimates parameters of interest on the bootstrapped samples, with the resulting set of estimated parameters approximating the sampling distribution.[17]

Unfortunately, the nonparametric bootstrap fails for many of the algorithms used on text. For

[16] Due to its popularity in the deep learning community, variational inference is a common feature in newer machine learning frameworks; see for example Edward (Tran et al. 2016, 2017) for a python library that builds variational inference on top of the TensorFlow platform. Edward and similar tools can be used to implement topic models and the other kinds of text models that we discussed above.

[17] See, e.g., Horowitz (2003) for an overview.

example, it is known to fail for methods that involve non-differentiable loss functions (e.g., the lasso), and with-replacement resampling produces overfit in the bootstrap samples (repeated observations make prediction seem easier than it actually is). Hence, for many applications, it is better to look to methods more suitable for high-dimensional estimation algorithms. The two primary candidates are the parametric bootstrap and subsampling.

The *parametric bootstrap* generates new unrepeated observations for each bootstrap sample given an estimated generative model (or other assumed form for the data generating process).[18] In doing so, it avoids pathologies of the nonparametric bootstrap that arise from using the empirical sample distribution. The cost is that the parametric bootstrap is, of course, parametric: It makes strong assumptions about the underlying generative model, and one must bear in mind that the resulting inference is conditional upon these assumptions.[19]

An alternative method, *subsampling*, provides a nonparametric approach to inference that remains robust to estimation features such as non-differentiable losses and model selection. The book by Politis et al. (1999) gives a comprehensive overview. In subsampling, the data are partitioned into subsamples without replacement (the number of subsamples being a function of the total sample size) and the target parameters are re-estimated separately on each subsample. The advantage of subsampling is that, because each subsample is a draw from the true DGP, it works for a wide variety of estimation algorithms. However, since each subsample is smaller than the sample of interest, one needs to know the estimator's rate of convergence in order to translate between the uncertainty in the subsamples and in the full sample.[20]

Finally, one may consider *sample splitting* with methods that involve a model selection step (i.e., those setting some parameter values to zero, such as lasso). Model selection is performed on one 'selection' sample, then the standard inference is performed on the second 'estimation' sample

[18] See Efron (2012) for an overview that also makes interesting connections to Bayesian inference.

[19] For example, in a linear regression model, the parametric bootstrap requires simulating errors from an assumed, say Gaussian, distribution. One must make assumptions on the exact form of this distribution, including whether the errors are homoskedastic or not. This contrasts with our usual approaches to standard errors for linear regression that are robust to assumptions on the functional form of the errors.

[20] For many applications, doing this translation under the assumption of a standard $\sqrt{n}$ learning rate is a reasonable choice. For example, Knight and Fu (2000) shows conditions under which the $\sqrt{n}$ rate holds for the lasso. However, in many text-as-data applications the dimension of the data (the size of the vocabulary) will be large and growing with the number of observations. In such settings $\sqrt{n}$ learning may be optimistic, and more sophisticated methods may need to be used to infer the learning rate (see, e.g., Politis et al. 1999).

conditional upon the selected model.[21] Econometricians have successfully used this approach to obtain accurate inference for machine learning procedures. For example, Chernozhukov et al. (2018) use it in the context of treatment effect estimation via lasso, and Athey and Imbens (2016) use sample splitting for inference about heterogeneous treatment effects in the context of a *causal tree* model.

3.5 Some practical advice

The methods above will be compared and contrasted in our subsequent discussion of applications. In broad terms, however, we can make some rough recommendations for practitioners.

3.5.1 Choosing the best approach for a specific application

Dictionary-based methods heavily weight prior information about the function mapping features $\mathbf{c}_i$ to outcomes $\mathbf{v}_i$. They are therefore most appropriate in cases where such prior information is strong and reliable, and where information in the data is correspondingly weak. An obvious example is a case where the outcomes $\mathbf{v}_i$ are not observed for any i, so there is no training data available to fit a supervised model, and where the mapping of interest does not match the factor structure of unsupervised methods such as topic models. In the setting of Baker et al. (2016), for example, there is no ground truth data on the actual level of policy uncertainty reflected in particular articles, and fitting a topic model would be unlikely to endogenously pick out policy uncertainty as a topic. A second example is where some training data does exist, but it is sufficiently small or noisy that the researcher believes a prior-driven specification of $f(\cdot)$ is likely to be more reliable.

Text regression is generally a good choice for predicting a single attribute, especially when one has a large amount of labeled training data available. As described in Ng and Jordan (2002) and Taddy (2013c), supervised generative techniques such as naive Bayes and MNIR can improve prediction when p is large relative to n; however, these gains diminish with the sample size due to the asymptotic efficiency of many text regression techniques. In text regression, we have found that it is usually unwise to attempt to learn flexible functional forms unless n is much larger than p. When this is not the case, we generally recommend linear regression methods. Given the

[21] For example, when using lasso, one might apply OLS in the second stage using only the covariates selected by lasso in the first stage.

availability of fast and robust tools (gamlr and glmnet in R, and scikit-learn in Python), and the typically high dimensionality of text data, many prediction tasks in social science with text inputs can be efficiently addressed via penalized linear regression.

When there are multiple attributes of interest, and one wishes to resolve or control for interdependencies between these attributes and their effects on language, then one will need to work with a generative model for text. Multinomial logistic regression and its extensions can be applied to such situations, particularly via distributed multinomial regression. Alternatively, for corpora of many unlabeled documents (or when the labels do not tell the whole story that one wishes to investigate), topic modeling is the obvious approach. Word embeddings are also becoming an option for such questions. In the spirit of contemporary machine learning, it is also perfectly fine to combine techniques. For example, a common setting will have a large corpora of labeled documents as well as a smaller set of documents about which some metadata exist. One approach is to fit a topic model on the larger corpora, and to then use these topics *as well as* the token counts for supervised text regression on the smaller labeled corpora.

3.5.2 Model validation and interpretation

Ex ante criteria for selecting an empirical approach are suggestive at best. In practice, it is also crucial to validate the performance of the estimation approach *ex post*. Real research often involves an iterative tuning process with repeated rounds of estimation, validation, and adjustment.

When the goal is prediction, the primary tool for validation is checking out-of-sample predictive performance on data held out from the main estimation sample. In Section 3.1.1, we discussed the technique of cross-validation (CV) for penalty selection, a leading example. More generally, whenever one works with complex and high-dimensional data, it is good practice to reserve a test set of data to use in estimation of the true average prediction error. Looping across multiple test sets, as in CV, is a common way of reducing the variance of these error estimates. (See Efron 2004 for a classic overview.)

In many social science applications, the goal is to go beyond prediction and use the values $\hat{\mathbf{V}}$ in some subsequent descriptive or causal analysis. In these cases, it is important to also validate the accuracy with which the fitted model is capturing the economic or descriptive quantity of interest.

One approach that is often effective is *manual audits*: cross-checking some subset of the fitted

values against the coding a human would produce by hand. An informal version of this is for a researcher to simply inspect a subset of documents alongside the fitted $\hat{\mathbf{V}}$ and evaluate whether the estimates align with the concept of interest. A formal version would involve having one or more people manually classify each document in a subset and evaluating quantitatively the consistency between the human and machine codings. The subsample of documents does not need to be large in order for this exercise to be valuable—often as few as 20 or 30 documents is enough to provide a sense of whether the model is performing as desired.

This kind of auditing is especially important for dictionary methods. Validity hinges on the assumption that a particular function of text features—counts of positive or negative words, an indicator for the presence of certain keywords, etc.—will be a valid predictor of the true latent variable $\mathbf{V}$. In a setting where we have sufficient prior information to justify this assumption, we typically also have enough prior information to evaluate whether the resulting classification looks accurate. An excellent example of this is Baker et al. (2016), who perform a careful manual audit to validate their dictionary-based method for identifying articles that discuss policy uncertainty.

Audits are also valuable in studies using other methods. In Gentzkow and Shapiro (2010), for example, the authors perform an audit of news articles that their fitted model classifies as having right-leaning or left-leaning slant. They do not compare this against hand-coding directly, but rather count the number of times the key phrases that are weighted by the model are used straightforwardly in news text, as opposed to occurring in quotation marks or in other types of articles such as letters to the editor.

A second approach to validating a fitted model is inspecting the estimated coefficients or other parameters of the model directly. In the context of text regression methods, however, this needs to be approached with caution. While there is a substantial literature on statistical properties of estimated parameters in penalized regression models (see Bühlmann and Van De Geer (2011) and Tibshirani et al. (2015)), the reality is that these coefficients are typically only interpretable in cases where the true model is extremely sparse, so that the model is likely to have selected the correct set of variables with high probability. Otherwise, multicollinearity means the set of variables selected can be highly unstable.

These difficulties notwithstanding, inspecting the most important coefficients to see if they make intuitive sense can still be useful as a validation and sanity check. Note that "most important"

can be defined in a number of ways; one can rank estimated coefficients by their absolute values, or by absolute value scaled by the standard deviation of the associated covariate, or perhaps by the order in which they first become nonzero in a lasso path of decreasing penalties. Alternatively, see Gentzkow et al. (2016) for application-specific term rankings.

Inspection of fitted parameters is generally more informative in the context of a generative model. Even there, some caution is in order. For example, Taddy (2015a) finds that for MNIR models, getting an interpretable set of word loadings requires careful penalty tuning and the inclusion of appropriate control variables. As in text regression, it is usually worthwhile to look at the largest coefficients for validation but not take the smaller values too seriously.

Interpretation or story building around estimated parameters tends to be a major focus for topic models and other unsupervised generative models. Interpretation of the fitted topics usually proceeds by ranking the tokens in each topic according to token probability, θ_{lj}, or by token lift $\theta_{lj}/\bar{p}_j$ with $\bar{p}_j = \frac{1}{n}\sum_i c_{ij}/m_i$. For example, if the five highest lift tokens in topic l for a model fit to a corpus of restaurant reviews are *another.minute*, *flag.down*, *over.minute*, *wait.over*, *arrive.after*, we might expect that reviews with high v_{il} correspond to negative experiences where the patron was forced to wait for service and food (example from Taddy 2012). Again, however, we caution against the over-interpretation of these unsupervised models: the posterior distributions informing parameter estimates are often multimodal, and multiple topic model runs can lead to multiple different interpretations. As argued in Airoldi and Bischof (2017) and in a comment by Taddy (2017a), the best way to build interpretability for topic models may be to add some supervision (i.e., to incorporate external information on the topics for some set of cases).

4 Applications

We now turn to applications of text analysis in economics and related social sciences. Rather than presenting a comprehensive literature survey, the goal of this section is to present a selection of illustrative papers to give the reader a sense of the wide diversity of questions that may be addressed with textual analysis and to provide a flavor of how some of the methods in Section 3 are applied in practice.

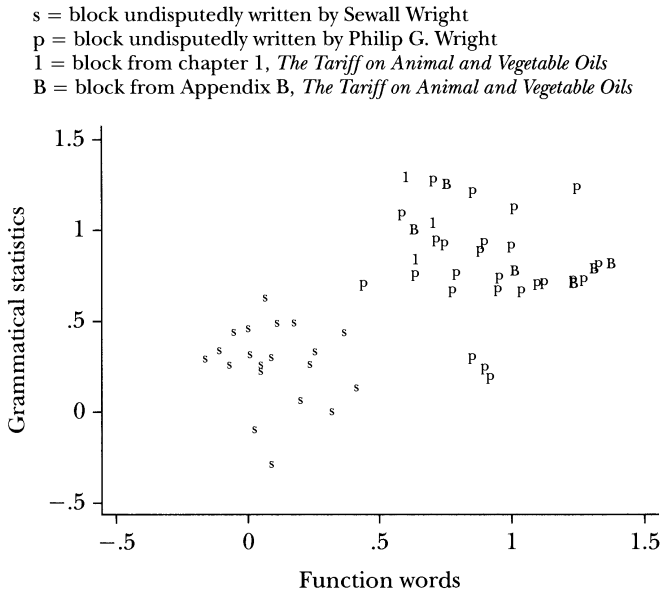
4.1 Authorship

A classic descriptive problem is inferring the author of a document. While this is not usually a first-order research question for social scientists, it provides a particularly clean example, and a good starting point to understand the applications that follow.

In what is often seen as the first modern statistical analysis of text data, Mosteller and Wallace (1963) use text analysis to infer the authorship of the disputed *Federalist Papers* that had alternatively been attributed to either Alexander Hamilton or James Madison. They define documents i to be individual *Federalist Papers*, the data features $\mathbf{c}_i$ of interest to be counts of function words such as "an," "of," and "upon" in each document, and the outcome v_i to be an indicator for the identity of the author. Note that the function words the authors focus on are exactly the "stop words" that are frequently excluded from analysis (as discussed in Section 2 above). The key feature of these words is that their use by a given author tends to be stable regardless of the topic, tone, or intent of the piece of writing. This means they provide little valuable information if the goal is to infer characteristics such as political slant or discussion of policy uncertainty that are independent of the idiosyncratic styles of particular authors. When such styles are the object of interest, however, function words become among the most informative text characteristics. Mosteller and Wallace (1963) use a sample of *Federalist Papers*, whose authorship by either Madison or Hamilton is undisputed, to train a naive Bayes classifier (a supervised generative model, as discussed in Section 3.2) in which the probabilities $\mathrm{p}\left(c_{ij}|v_i\right)$ of each phrase j are assumed to be independent Poisson or negative binomial random variables, and the inferences for the unknown documents are made by Bayes' rule. The results provide overwhelming evidence that all of the disputed papers were authored by Madison.

Stock and Trebbi (2003) apply similar methods to answer an authorship question of more direct interest to economists: who invented instrumental variables? The earliest known derivation of the instrumental variables estimator appears in an appendix to *The Tariff on Animal and Vegetable Oils*, a 1928 book by statistician Philip Wright. While the bulk of the book is devoted to a "painfully detailed treatise on animal and vegetable oils, their production, uses, markets and tariffs," the appendix is of an entirely different character, with "a succinct and insightful explanation of why data on price and quantity alone are in general inadequate for estimating either supply or demand;

Figure 3: Scatterplot of Authorship Predictions from PCA Method



Source: Stock and Trebbi (2003).

two separate and correct derivations of the instrumental variables estimators of the supply and demand elasticities; and an empirical application" (Stock and Trebbi 2003 pg. 177). The contrast between the two parts of the book has led to speculation that the appendix was not written by Philip Wright, but rather by his son Sewall. Sewall Wright was an economist who had originated the method of "path coefficients" used in one of the derivations in the appendix. Several authors including Manski (1988) are on record attributing authorship to Sewall; others including Angrist and Krueger (2001) attribute it to Philip.

In Stock and Trebbi's (2003) study, the outcome v_i is an indicator for authorship by either Philip or Sewall. The data features $\mathbf{c}_i = \left[\begin{array}{cc} \mathbf{c}_i^{func} & \mathbf{c}_i^{gram} \end{array} \right]$ are counts of the same function words used by Mosteller and Wallace (1963) plus counts of a set of grammatical constructions (e.g, "noun followed by adverb") measured using an algorithm due to Mannion and Dixon (1997). The training sample $\mathbf{C}^{train}$ consists of 45 documents known to have been written by either Philip or Sewall, and the test sample $\mathbf{C}^{test}$ in which v_i is unobserved consists of eight blocks of text from the appendix plus one block of text from chapter 1 of *The Tariff on Animal and Vegetable Oils* included as a validity check. The authors apply principal components analysis, which we can think of as an unsupervised cousin of the topic modeling approach discussed in Section 3.2.1. They extract the

first four principal components from $\mathbf{c}_i^{func}$ and $\mathbf{c}_i^{gram}$ respectively, and then run regressions of the binary authorship variable on the principal components, resulting in predicted values $\hat{v}_i^{func}$ and $\hat{v}_i^{gram}$.

The results provide overwhelming evidence that the disputed appendix was in fact written by Philip. Figure 3 plots the values $\left(\hat{v}_i^{func}, \hat{v}_i^{gram}\right)$ for all of the documents in the sample. Each point in the figure is a document i, and the labels indicate whether the document is known to be written by Philip ("P" or "1"), known to be written by Sewall ("S"), or of uncertain authorship ("B"). The measures clearly distinguish the two authors, with documents by each forming clear, non-overlapping clusters. The uncertain documents all fall squarely within the cluster attributed to Philip, with the predicted values $\hat{v}_i^{func}, \hat{v}_i^{gram} \approx 1$.

4.2 Stock prices

An early example analyzing news text for stock price prediction appears in Cowles (1933). He subjectively categorizes the text of editorial articles of Peter Hamilton, chief editor of the *Wall Street Journal* from 1902 to 1929, as "bullish," "bearish," or "doubtful." Cowles then uses these classifications to predict future returns of the Dow Jones Industrial Average. Hamilton's track record is unimpressive. A market-timing strategy based on his *Wall Street Journal* editorials underperforms a passive investment in the Dow Jones Industrial Average by 3.5 percentage points per year.

In its modern form, the implementation of text-based prediction in finance is computationally driven, but it applies methods that are conceptually similar to Cowles's approach, seeking to predict the target quantity $\mathbf{V}$ (the Dow Jones return, in the example of Cowles) from the array of token counts $\mathbf{C}$. We discuss three examples of recent papers that study equity return prediction in the spirit of Cowles: one relying on a pre-existing dictionary (as discussed at the beginning of Section 3), one using regression techniques (as discussed in Section 3.1), and another using generative models (as discussed in Section 3.2).

Tetlock (2007) is a leading dictionary-based example of analyzing media sentiment and the stock market. He studies word counts $\mathbf{c}_i$ in the *Wall Street Journal*'s widely read "Abreast of the Market" column. Counts from each article i are converted into a vector of sentiment scores $\hat{\mathbf{v}}_i$ in 77 different sentiment dimensions based on the Harvard IV-4 psychosocial dictionary.[22] The

[22]While it has only recently been used in economics and finance, the Harvard dictionary and associated General

time series of daily sentiment scores for each category ($\hat{\mathbf{v}}_i$) are condensed into a single principal component, which Tetlock names the "pessimism factor" due to the component's especially close association with the "pessimism" dimension of the sentiment categories.

The second stage of the analysis uses this pessimism score to forecast stock market activity. High pessimism significantly negatively forecasts one-day-ahead returns on the Dow Jones Industrial Average. This effect is transitory, and the short term index dip associated with media pessimism reverts within a week, consistent with the interpretation that article text is informative regarding media and investor sentiment, as opposed to containing fundamental news that permanently impacts prices.

The number of studies using dictionary methods to study asset pricing phenomena is growing. Loughran and McDonald (2011) demonstrate that the widely used Harvard dictionary can be ill-suited for financial applications. They construct an alternative, finance-specific dictionary of positive and negative terms and document its improved predictive power over existing sentiment dictionaries. Bollen et al. (2011) document a significant predictive correlation between Twitter messages and the stock market using other dictionary-based tools such as OpinionFinder and Google's Profile of Mood States. Wisniewski and Lambe (2013) show that negative media attention of the banking sector, summarized via *ad hoc* pre-defined word lists, Granger-causes bank stock returns during the 2007–2009 financial crisis and not the reverse, suggesting that journalistic views have the potential to influence market outcomes, at least in extreme states of the world.

The use of text regression for asset pricing is exemplified by Jegadeesh and Wu (2013). They estimate the response of company-level stock returns, v_i, to text information in the company's annual report (token counts, $\mathbf{c}_i$). The authors' objective is to determine whether regression techniques offer improved stock return forecasts relative to dictionary methods.

The authors propose the following regression model to capture correlations between occurrences of individual words and subsequent stock return realizations around regulatory filing dates

$$v_i = a + b\left(\sum_j w_j \frac{c_{ij}}{\sum_j c_{ij}}\right) + \varepsilon_i.$$

Inquirer software for textual content analysis dates to the 1960s and has been widely used in linguistics, psychology, sociology, and anthropology.

Documents i are defined to be annual reports filed by firms at the Securities Exchange Commission. The outcome variable v_i is a stock's cumulative four-day return beginning on the filing day. The independent variable c_{ij} is a count of occurrences of word j in annual report i. The coefficient w_j summarizes the average association between an occurrence of word j and the stock's subsequent return. The authors show how to estimate w_j from a cross-sectional regression, along with a subsequent rescaling of all coefficients to remove the common influence parameter b. Finally, variables to predict returns are built from the estimated weights, and are shown to have stronger out-of-sample forecasting performance than dictionary-based indices from Loughran and McDonald (2011). The results highlight the limitations of using fixed dictionaries for diverse predictive problems, and that these limitations are often surmountable by estimating application-specific weights via regression.

Manela and Moreira (2015) take a regression approach to construct an index of news-implied market volatility based on text from the *Wall Street Journal* from 1890-2009. They apply support vector machines, a non-linear regression method which we discuss in Section 3.1.3. This approach applies a penalized least squares objective to identify a small subset of words whose frequencies are most useful for predicting outcomes—in this case, turbulence in financial markets. Two important findings emerge from their analysis. First, the terms most closely associated with market volatility relate to government policy and wars. Second, high levels of news-implied volatility forecast high future stock market returns. These two facts together give insight into the types of risks that drive investors' valuation decisions.[23]

The closest modern analog of Cowles's study is Antweiler and Frank (2004), who take a generative modeling approach to ask: How informative are the views of stock market prognosticators who post on internet message boards? Similar to Cowles's analysis, these authors classify postings on stock message boards as "buy," "sell," or "hold" signals. But the vast number of postings, roughly 1.5 million in the analyzed sample, makes subjective classification of messages infeasible. Instead, generative techniques allow the authors to automatically classify messages.

The authors create a training sample of 1,000 messages, and form $\mathbf{V}^{train}$ by manually classifying messages into one of the three categories. They then use the naive Bayes method described

[23]While Manela and Moreira (2015) study aggregate market volatility, Kogan et al. (2009) and Boudoukh et al. (2016) use text from news and regulatory filings to predict firm-specific volatility. Chinco et al. (2017) apply lasso in high frequency return prediction using pre-processed financial news text sentiment as an explanatory variable.

in Section 3.2.2 to estimate a probability model that maps word counts of postings $\mathbf{C}$ into classifications $\hat{\mathbf{V}}$ for the remaining 1.5 million messages. Finally, the buy/sell/hold classification of each message is aggregated into an index that is used to forecast stock returns. Consistent with the conclusions of Cowles, message board postings show little ability to predict stock returns. They do, however, possess significant and economically meaningful information about stock volatility and trading volume.[24]

Bandiera et al. (2017) apply unsupervised machine learning—topic modeling (LDA)—to a large panel of CEO diary data. They uncover two distinct behavioral types that they classify as "leaders" who focus on communication and coordination activities, and "managers" who emphasize production-related activities. They show that, due to horizontal differentiation of firm and manager types, appropriately matched firms and CEOs enjoy better firm performance. Mismatches are more common in lower income economies, and mismatches can account for 13 percent of the labor productivity gap between firms in high- and middle/low-income countries.

4.3 Central bank communication

A related line of research analyzes the impact of communication from central banks on financial markets. As banks rely more on these statements to achieve policy objectives, an understanding of their effects is increasingly relevant.

Lucca and Trebbi (2009) use the content of Federal Open Market Committee (FOMC) statements to predict fluctuations in Treasury securities. To do this, they use two different dictionary-based methods (Section 3) —Google and Factiva semantic orientation scores—to construct $\hat{\mathbf{v}}_i$, which quantifies the direction and intensity of the i^{th} FOMC statement. In the Google score, $\mathbf{c}_i$ counts how many Google search hits occur when searching for phrases in i plus one of the words from a list of antonym pairs signifying positive or negative sentiment (e.g., "hawkish" versus "dovish"). These counts are mapped into $\hat{\mathbf{v}}_i$ by differencing the frequency of positive and negative searches and averaging over all phrases in i. The Factiva score is calculated similarly. Next, the central bank sentiment proxies $\hat{\mathbf{v}}_i$ are used to predict Treasury yields in a vector autoregression (VAR). They find that changes in statement content, as opposed to unexpected deviations

[24]Other papers that use naive Bayes and similar generative models to study behavioral finance questions include Buehlmaier and Whited (2016), Li (2010), and Das and Chen (2007).

in the federal funds target rate, are the main driver of changes in interest rates.

Born et al. (2014) extend this idea to study the effect of central bank sentiment on stock market returns and volatility. They construct a financial stability sentiment index $\hat{\mathbf{v}}_i$ from Financial Stability Reports (FSRs) and speeches given by central bank governors. Their approach uses a sentiment dictionary to assign optimism scores to word counts $\mathbf{c}_i$ from central bank communications. They find that optimistic FSRs tend to increase equity prices and reduce market volatility during the subsequent month.

Hansen et al. (2017) research how FOMC transparency affects debate during meetings by studying a change in disclosure policy. Prior to November 1993, the FOMC meeting transcripts were secret, but following a policy shift transcripts became public with a time lag. There are potential costs and benefits of increased transparency, such as the potential for more efficient and informed debate due to increased accountability of policymakers. On the other hand, transparency may make committee members more cautious, biased toward the status quo, or prone to groupthink.

The authors use topic modeling (Section 3.2.1) to study 149 FOMC meeting transcripts during Alan Greenspan's tenure. The unit of observation is a member-meeting. The vector $\mathbf{c}_i$ counts the words used by FOMC member m in meeting t, and i is defined as the pair (m,t). The outcome of interest, $\mathbf{v}_i$, is a vector that includes the proportion of i's language devoted to the K different topics (estimated from the fitted topic model), the concentration of these topic weights, and the frequency of data citation by i. Next, a difference-in-differences regression estimates the effects of the change in transparency on $\hat{\mathbf{v}}_i$. The authors find that, after the move to a more transparent system, inexperienced members discuss a wider range of topics and make more references to data when discussing economic conditions (consistent with increased accountability); but also speak more like Chairman Greenspan during policy discussions (consistent with increased conformity). Overall, the accountability effect appears stronger, as inexperienced members' topics appear to be more influential in shaping future deliberation after transparency.

4.4 Nowcasting

Important variables such as unemployment, retail sales, and GDP are measured at low frequency, and estimates are released with a significant lag. Others, such as racial prejudice or local government corruption are not captured by standard measures at all. Text produced online such as search queries, social media posts, listings on job websites, and so on can be used to construct alternative real-time estimates of the current values of these variables. By contrast with the standard exercise of forecasting future variables, this process of using diverse data sources to estimate current variables has been termed "nowcasting" in the literature (Banbura et al. 2013).

A prominent early example is the Google Flu Trends project. Zeng and Wagner (2002) note that the volume of searches or web hits seeking information related to a disease may be a strong predictor of its prevalence. Johnson et al. (2004) provide an early data point suggesting that browsing influenza-related articles on the website healthlink.com is correlated with traditional surveillance data from the Centers for Disease Control (CDC). In the late 2000s, a group of Google engineers built on this idea to create a product that predicts flu prevalence from Google searches using text regression.

The results are reported in a widely-cited *Nature* article by Ginsberg et al. (2009). Their raw data $\mathcal{D}$ consist of "hundreds of billions of individual searches from 5 years of Google web search logs." Aggregated search counts are arranged into a vector $\mathbf{c}_i$, where a document i is defined to be a particular US region in a particular week, and the outcome of interest v_i is the true prevalence of flu in the region-week. In the training data, this is taken to be equal to the rate measured by the CDC. The authors first restrict attention to the 50 million most common terms, then select those most diagnostic of an outbreak using text regression (Section 3.1), specifically a variant of partial least squares regression. They first run 50 million univariate regressions of $\log\left(v_i/\left(1-v_i\right)\right)$ on $\log\left(c_{ij}/\left(1-c_{ij}\right)\right)$, where c_{ij} is the share of searches in i containing search term j. They then fit a sequence of multivariate regression models of v_i on the top n terms j as ranked by average predictive power across regions for $n \in \{1,2,...\}$. Next, they select the value of n that yields the best fit on a hold-out sample. This yields a regression model with $n = 45$ terms. The model produces accurate flu rate estimates for all regions approximately 1–2 weeks ahead of the CDC's regular report publication dates.[25]

[25] A number of subsequent papers debate the longer term performance of Google Flu Trends. Lazer et al. (2014), for

Related work in economics attempts to nowcast macroeconomic variables using data on the frequency of Google search terms. In Choi and Varian (2012) and Scott and Varian (2014, 2015), search term counts are aggregated by week and by geographic location, then converted to location-specific frequency indices. They estimate spike and slab Bayesian forecasting models, discussed in Section 3.1.4 above. Forecasts of regional retail sales, new housing starts, and tourism activity are all significantly improved by incorporating a few search term indices that are relevant for each category in linear models. Their results suggest a potential for large gains in forecasting power using web browser search data.

Saiz and Simonsohn (2013) use web search results to estimate the current extent of corruption in US cities. Standard corruption measures based on surveys are available at the country and state level, but not for smaller geographies. The authors use a dictionary approach in which the index $\hat{v}_i$ of corruption is defined to be the ratio of search hits for the name of a geographic area i plus the word "corruption" divided by hits for the name of the geographic area alone. These counts are extracted from search engine results. As a validation, the authors first show that country-level and state-level versions of their measure correlate strongly with established corruption indicies and covary in a similar way with country- and state-level demographics. They then compute their measure for US cities, and study its observable correlates.

Stephens-Davidowitz (2014) uses the frequency of racially charged terms in Google searches to estimate levels of racial animus in different areas of the United States. Estimating animus via traditional surveys is challenging because individuals are often reluctant to state their true attitudes. The paper's results suggest Google searches provide a less filtered and therefore more accurate measure. The author uses a dictionary approach in which the index $\hat{v}_i$ of racial animus in area i is the share of searches originating in that area that contain a set of racist words. He then uses these measures to estimate the impact of racial animus on votes for Barack Obama in the 2008 election, finding a statistically significant and economically large negative effect on Obama's vote share relative to the Democratic vote share in the previous election.

example, show that the accuracy of the Google Flu Trends model—which has not been re-calibrated or updated based on more recent data—has deteriorated dramatically, and that in recent years it is outperformed by simple extrapolation from prior CDC estimates. This may reflect changes in both search patterns and the epidemiology of the flu, and it suggests a general lesson that the predictive relationship mapping text to a real outcome of interest may not be stable over time. On the other hand, Preis and Moat (2014) argue that an adaptive version of the model that more flexibly accounts for joint dynamics in flu incidence and search volume significantly improves real-time influenza monitoring.

4.5 Policy uncertainty

Among the most influential applications of text analysis in the economics literature to date is a measure of economic policy uncertainty (EPU) developed by Baker et al. (2016). Uncertainty about both the path of future government policies and the impact of current government policies has the potential to increase risk for economic actors and so potentially depress investment and other economic activity. The authors use text from news outlets to provide a high-frequency measure of EPU and then estimate its economic effects.

Baker et al. (2016) define the unit of observation i to be a country-month. The outcome v_i of interest is the true level of economic policy uncertainty. The authors apply a dictionary method to produce estimates $\hat{v}_i$ based on digital archives of ten leading newspapers in the US. An element of the input data c_{ij} is a count of the number of articles in newspaper j in country-month i containing at least one keyword from each of three categories defined by hand: one related to the economy, a second related to policy, and a third related to uncertainty. The raw counts are scaled by the total number of articles in the corresponding newspaper-month and normalized to have standard deviation one. The predicted value $\hat{v}_i$ is then defined to be a simple average of these scaled counts across newspapers.

The simplicity of the manner in which the index is created allows for a high amount of flexibility across a broad range of applications. For instance, by including a fourth, policy-specific category of keywords, the authors can estimate narrower indices related to Federal Reserve policy, inflation, and so on.

Baker et al. (2016) validate $\hat{v}_i$ using a human audit of 12,000 articles from 1900 to 2012. Teams manually scored articles on the extent to which they discuss economic policy uncertainty and the specific policies they relate to. The resulting human-coded index has a high correlation with $\hat{v}_i$.

With the estimated $\hat{v}_i$ in hand, the authors analyze the micro- and macro-level effects of EPU. Using firm-level regressions, they first measure how firms respond to this uncertainty, and find that it leads to reduced employment, investment, and greater asset price volatility for that firm. Then, using both US and international panel VAR models, the authors find that increased $\hat{v}_i$ is a strong predictor of lower investment, employment, and production.

Hassan et al. (2017) measure political risk at the firm level by analyzing quarterly earnings

call transcripts. Their measure captures the frequency with which policy-oriented language and "risk" synonyms co-occur in a transcript. Firms with high levels of political risk actively hedge these risks by lobbying more intensively and donating more to politicians. When a firm's political risk rises, it tends to retrench hiring and investment, consistent with the findings of Baker et al. (2016) at the aggregate level. Their findings indicate that political shocks are an important source of idiosyncratic firm-level risk.

4.6 Media slant

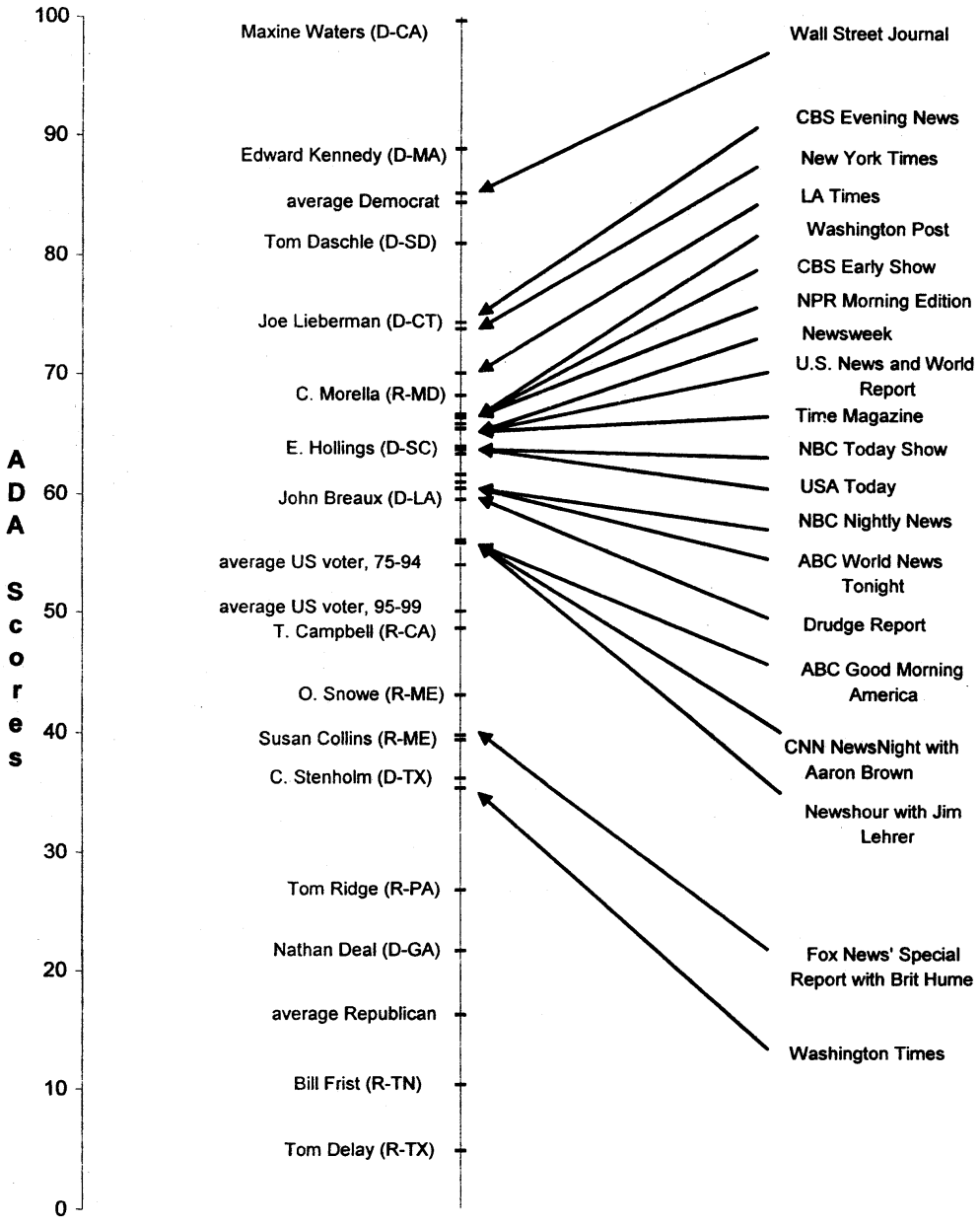
A text analysis problem that has received significant attention in the social science literature is measuring the political slant of media content. Media outlets have long been seen as having a uniquely important role in the political process, with the power to potentially sway both public opinion and policy. Understanding how and why media outlets slant the information they present is important to understanding the role media play in practice, and to informing the large body of government regulation designed to preserve a diverse range of political perspectives.

Groseclose and Milyo (2005) offer a pioneering application of text analysis methods to this problem. In their setting, i indexes a set of large US media outlets, and documents are defined to be the complete news text or broadcast transcripts for an outlet i. The outcome of interest v_i is the political slant of outlet i. To give this measure content, the authors use speeches by politicians in the US Congress to form a training sample, and define v_i within this sample to be a politician's Americans for Democratic Action (ADA) score, a measure of left-right political ideology based on congressional voting records. The predicted values $\hat{v}_i$ for the media outlets thus place them on the same left-right scale as the politicians, and answer the question "what kind of politician does this news outlet's content sound most similar to?"

The raw data are the full text of speeches by congresspeople and news reports by media outlets over a period spanning the 1990s to the early 2000s.[26] The authors dramatically reduce the dimensionality of the data in an initial step by deciding to focus on a particularly informative subset of phrases: the names of 200 think tanks. These think tanks are widely viewed as having clear

[26]For members of Congress, the authors use all entries in the *Congressional Record* from January 1, 1993 to December 31, 2002. The text includes both floor speeches and documents the member chose to insert in the record but did not read on the floor. For news outlets, the time period covered is different for different outlets, with start dates as early as January 1990 and end dates as late as July 2004.

Figure 4: Distribution of Political Orientation: Media Outlets and Members of Congress



Source: Groseclose and Milyo (2005).

political positions (e.g., the Heritage Foundation on the right and the NAACP on the left). The relative frequency with which a politician cites conservative as opposed to liberal think tanks turns out to be strongly correlated with a politician's ideology. The paper's premise is that the citation frequencies of news outlets will then provide a good index of those outlets' political slants. The features of interest $\mathbf{c}_i$ are a (1×50) vector of citation counts for each of 44 highly-cited think tanks

plus 6 groups of smaller think tanks.

The text analysis is based on a supervised generative model (Section 3.2.2). The utility that congress member or media firm i derives from citing think tank j is $U_{ij} = a_j + b_j v_i + e_{ij}$, where v_i is the observable ADA score of a congressmember i or unobserved slant of media outlet i, and e_{ij} is an error distributed type-I extreme value. The coefficient b_j captures the extent to which think tank j is cited relatively more by conservatives. The model is fit by maximum likelihood with the parameters (a_j, b_j) and the unknown slants v_i estimated jointly. This is an efficient but computationally intensive approach to estimation, and it constrains the authors' focus to 20 outlets. This limitation can be sidestepped using more recent approaches such as Taddy's (2013b) multinomial inverse regression.

Figure 4 shows the results, which suggest three main findings. First, the media outlets are all relatively centrist: they are all to the left of the average Republican and to the right of the average Democrat with one exception. Second, the ordering matches conventional wisdom, with the *New York Times* and *Washington Post* on the left, and *Fox News* and the *Washington Times* on the right.[27] Third, the large majority of outlets fall to the left of the average in congress, which is denoted in the figure by "average US voter." The last fact underlies the authors' main conclusion, which is that there is an overall liberal bias in the media.

Gentzkow and Shapiro (2010) build on the Groseclose and Milyo (2005) approach to measure the slant of 433 US daily newspapers. The main difference in approach is that Gentzkow and Shapiro (2010) omit the initial step that restricts the space of features to mentions of think tanks, and instead consider all phrases that appear in the 2005 *Congressional Record* as potential predictors, letting the data select those that are most diagnostic of ideology. These could potentially be think tank names, but they turn out instead to be politically charged phrases such as "death tax," "bring our troops home," and "war on terror."

After standard pre-processing—stemming and omitting stopwords—the authors produce counts of all 2-grams and 3-grams by speaker. They then select the top 1,000 phrases (500 of each length) by a χ^2 criterion that captures the degree to which each phrase is diagnostic of the speaker's party.

[27] The one notable exception is the *Wall Street Journal* which is generally considered to be right-of-center but which is estimated by Groseclose and Milyo (2005) to be the most left-wing outlet in their sample. This may reflect an idiosyncrasy specific to the way they cite think tanks; Gentzkow and Shapiro (2010) use a broader sample of text features and estimate a much more conservative slant for the *Wall Street Journal*.

This is the standard χ^2 test statistic for the null hypothesis that phrase j is used equally often by Democrats and Republicans, and it will be high for phrases that are both used frequently and used asymmetrically by the parties.[28] Next, a two-stage supervised generative method is used to predict newspaper slant v_i from the selected features. In the first stage, the authors run a separate regression for each phrase j of counts (c_{ij}) on speaker i's ideology, which is measured as the 2004 Republican vote share in the speaker's district. They then use the estimated coefficients $\hat{\beta}_j$ to produce predicted slant $\hat{v}_i \propto \sum_{j=1}^{1,000} \hat{\beta}_j c_{ij}$ for the unknown newspapers i.[29]

The main focus of the study is characterizing the incentives that drive newspapers' choice of slant. With the estimated $\hat{v}_i$ in hand, the authors estimate a model of consumer demand in which a consumer's utility from reading newspaper i depends on the distance between i's slant v_i and an ideal slant v^* which is greater the more conservative is the consumer's ideology. Estimates of this model using zipcode-level circulation data imply a level of slant that newspapers would choose if their only incentive was to maximize profits. The authors then compare this profit-maximizing slant to the level actually chosen by newspapers, and ask whether the deviations can be predicted by the identity of the newspaper's owner, or by other non-market factors such as the party of local incumbent politicians. They find that profit maximization fits the data well, and that ownership plays no role in explaining the choice of slant. In this study, $\hat{v}_i$ is both an independent variable of interest (in the demand analysis) and an outcome of interest (in the supply analysis).

Note that both Groseclose and Milyo (2005) and Gentzkow and Shapiro (2010) use a two-step procedure where they reduce the dimensionality of the data in a first stage and then estimate a predictive model in the second. Taddy (2013b) shows how to combine a more sophisticated generative model with a novel algorithm for estimation to estimate the predictive model in a single step using the full set of phrases in the data. He shows that this substantially increases the in-sample predictive power of the measure.

[28] The statistic is

$$\chi^2_j = \frac{f_{jr} f_{\sim jd} - f_{jd} f_{\sim jr}}{(f_{jr} + f_{jd})(f_{jr} + f_{\sim jd})(f_{\sim jr} + f_{jd})(f_{\sim jr} + f_{\sim jd})}$$

where f_{jd} and f_{jr} denote the number of times phrase j is used by Democrats or Republicans, respectively, and $f_{\sim jd}$ and $f_{\sim jr}$ denote the number of times phrases other than j are used by Democrats and Republicans, respectively.

[29] As Taddy (2013b) notes, this method (which Gentzkow and Shapiro 2010 derive in an *ad hoc* fashion) is essentially partial least squares. It differs from the standard implementation in that the variables v_i and c_{ij} would normally be standardized. Taddy (2013b) shows that doing so increases the in-sample predictive power of the measure from 0.37 to 0.57.

Greenstein et al. (2016) analyze the extent of bias and slant among Wikipedia contributors using similar methods. They find that contributors tend to edit articles with slants in opposition to their own slants. They also show that contributors' slants become less extreme as they become more experienced, and that the bias reduction is largest for those with the most extreme initial biases.

4.7 Market definition and innovation impact

Many important questions in industrial organization hinge on the appropriate definition of product markets. Standard industry definitions can be an imperfect proxy for the economically relevant concept. Hoberg and Phillips (2015) provide a novel way of classifying industries based on product descriptions in the text of company disclosures. This allows for flexible industry classifications that may vary over time as firms and economies evolve, and allows the researchers to analyze the effect of shocks on competition and product offerings.

Each publicly traded firm in the US must file an annual 10-K report describing, among other aspects of their business, the products that they offer. The unit of analysis i is a firm-year. Token counts from the business description section of the i^{th} 10-K filing are represented in the vector $\mathbf{c}_i$. A pairwise cosine similarity score, s_{ij}, based on the angle between $\mathbf{c}_i$ and $\mathbf{c}_j$, describes the closeness of product offerings for each pair i and j in the same filing year. Industries are then defined by clustering firms according to their cosine similarities. The clustering algorithm begins by assuming each firm is its own industry, and gradually agglomerates firms into industries by grouping a firm to the cluster with its nearest neighbor according to s_{ij}. The algorithm terminates when the number of industries (clusters) reaches 300, a number chosen for comparability to SIC and NAICS codes.[30]

After establishing an industry assignment for each firm-year, $\hat{v}_i$, the authors examine the effect of military and software industry shocks to competition and product offerings among firms. As an example, they find that the events of September 11, 2001 increased entry in high demand military markets and pushed products in this industry towards "non-battlefield information gathering and products intended for potential ground conflicts."

[30] Best et al. (2017) use a similar approach to classify products in their study of public procurement and organizational bureaucracy in Russia.

In a similar vein, Kelly et al. (2018) use cosine similarity among patent documents to create new indicators of patent quality. They assign higher quality to patents that are *novel* in that they have low similarity with the existing stock of patents and are *impactful* in that they have high similarity with subsequent patents. They then show that text-based novelty and similarity scores correlate strongly with measures of market value. Atalay et al. (2017) use text from job ads to measure task content, and use their measure to show that within-occupation task content shifts are at least as important as employment shifts across occupations in describing the large secular reallocation of routine tasks from humans to machines.

4.8 Topics in research, politics, and law

A number of studies apply topic models (Section 3.2.1) to describe how the focus of attention in a specific text corpus shifts over time.

A seminal contribution in this vein is Blei and Lafferty's (2007) analysis of topics in *Science*. Documents i are individual articles, the data $\mathbf{c}_i$ are counts of individual words, and the outcome of interest $\mathbf{v}_i$ is a vector of weights indicating the share of a given article devoted to each of 100 latent topics. The authors extend the baseline LDA model of Blei et al. (2003) to allow the importance of one topic in a particular article to be correlated with the presence of other topics. They fit the model using all *Science* articles from 1990-1999. The results deliver an automated classification of article content into semantically coherent topics such as evolution, DNA and genetics, cellular biology, and volcanoes.

Applying similar methods in the political domain, Quinn et al. (2010) use a topic model to identify the issues being discussed in the US Senate over the period 1997–2004. Their approach deviates from the baseline LDA model in two ways. First, they assume that each speech is associated with a single topic. Second, their model incorporates time-series dynamics that allow the proportion of speeches generated by a given topic to gradually evolve over the sample, similar to the dynamic topic model of Blei and Lafferty (2006). Their preferred specification is a model with 42 topics, a number chosen to maximize the subjective interpretability of the resulting topics.

Table 1 shows the words with the highest weights in each of twelve fitted topics. The labels "Judicial Nominations," "Constitutional," and so on are assigned by hand by the authors. The

Table 1: *Congressional Record* Topics and Key Words

Topic (Short Label)	Keys
1. Judicial Nominations	*nomine, confirm, nomin, circuit, hear, court, judg, judici, case, vacanc*
2. Constitutional	*case, court, attornei, supreme, justic, nomin, judg, m, decis, constitut*
3. Campaign Finance	*campaign, candid, elect, monei, contribut, polit, soft, ad, parti, limit*
4. Abortion	*procedur, abort, babi, thi, life, doctor, human, ban, decis, or*
5. Crime 1 [Violent]	*enforc, act, crime, gun, law, victim, violenc, abus, prevent, juvenil*
6. Child Protection	*gun, tobacco, smoke, kid, show, firearm, crime, kill, law, school*
7. Health 1 [Medical]	*diseas, cancer, research, health, prevent, patient, treatment, devic, food*
8. Social Welfare	*care, health, act, home, hospit, support, children, educ, student, nurs*
9. Education	*school, teacher, educ, student, children, test, local, learn, district, class*
10. Military 1 [Manpower]	*veteran, va, forc, militari, care, reserv, serv, men, guard, member*
11. Military 2 [Infrastructure]	*appropri, defens, forc, report, request, confer, guard, depart, fund, project*
12. Intelligence	*intellig, homeland, commiss, depart, agenc, director, secur, base, defens*

Source: Quinn et al. (2010).

results suggest that the automated procedure successfully isolates coherent topics of congressional debates. After discussing the structure of topics in the fitted model, the authors then track the relative importance of the topics across congressional sessions, and argue that spikes in discussion of particular topics track in an intuitive way the occurrence of important debates and external events.

Sim et al. (2015) estimate a topic model from the text of amicus briefs to the Supreme Court of the United States. They show that the overall topical composition of briefs for a given case, particularly along a conservative-liberal dimension, is highly predictive for how individual judges vote in the case.

5 Conclusion

Digital text provides a rich repository of information about economic and social activity. Modern statistical tools give researchers the ability to extract this information and encode it in a quantitative form amenable to descriptive or causal analysis. Both the availability of text data and the frontier of methods are expanding rapidly, and we expect the importance of text in empirical economics to continue to grow.

The review of applications above suggests a number of areas where innovation should proceed rapidly in coming years. First, a large share of text analysis applications continue to rely on ad hoc dictionary methods rather than deploying more sophisticated methods for feature selection

and model training. As we have emphasized, dictionary methods are appropriate in cases where prior information is strong and the availability of appropriately labeled training data is limited. Experience in other fields, however, suggests that modern methods will likely outperform ad hoc approaches in a substantial share of cases.

Second, some of the workhorse methods of text analysis such as penalized linear or logistic regression have still seen limited application in social science. In other contexts, these methods provide a robust baseline that performs similarly to or better than more complex methods. We expect the domains in which these methods are applied to grow.

Finally, virtually all of the methods applied to date, including those we would label as sophisticated or on the frontier, are based on fitting predictive models to simple counts of text features. Richer representations such as word embeddings (3.3), and linguistic models that draw on natural language processing tools have seen tremendous success elsewhere and we see great potential for their application in economics.

The rise of text analysis is part of a broader trend toward greater use of machine learning and related statistical methods in economics. With the growing availability of high-dimensional data in many domains—from consumer purchase and browsing behavior, to satellite and other spatial data, to genetics and neuro-economics—the returns are high to economists investing in learning these methods, and to increasing the flow of ideas between economics and fields such as statistics and computer science where frontier innovations in these methods are taking place.

References

Airoldi, E. M. and M. Bischof (2017). A regularization scheme on word occurrence rates that improves estimation and interpretation of topical content. *Journal of the American Statistical Association 111*(516), 1382–1403.

Airoldi, E. M., E. A. Erosheva, S. E. Fienberg, C. Joutard, T. Love, and S. Shringarpure (2010). Reconceptualizing the classification of PNAS articles. *Proceedings of the National Academy of Sciences 107*(49), 20899–20904.

Akaike, H. (1973). Information theory and an extension of the maximum likelihood principle. In *2nd International Symposium on Information Theory*, pp. 267–281. Akademiai Kiado.

Angrist, J. D. and A. B. Krueger (2001). Instrumental variables and the search for identification: from supply and demand to natural experiments. *Journal of Economic Perspectives 15*(4), 69–85.

Antweiler, W. and M. Z. Frank (2004). Is all that talk just noise? The information content of internet stock message boards. *Journal of Finance 59*(3), 1259–1294.

Armagan, A., D. B. Dunson, and J. Lee (2013). Generalized double Pareto shrinkage. *Statistica Sinica 23*(1), 119–143.

Atalay, E., P. Phongthiengtham, S. Sotelo, and D. Tannenbaum (2017). The evolving us occupational structure. Working Paper.

Athey, S. and G. Imbens (2016). Recursive partitioning for heterogeneous causal effects. *Proceedings of the National Academy of Sciences 113*(27), 7353–7360.

Bai, J. and S. Ng (2008). Forecasting economic time series using targeted predictors. *Journal of Econometrics 146*(2), 304–317.

Baker, S. R., N. Bloom, and S. J. Davis (2016). Measuring economic policy uncertainty. *The Quarterly Journal of Economics 131*(4), 1593–1636.

Banbura, M., D. Giannone, M. Modugno, and L. Reichlin (2013). Now-casting and the real-time data flow. *Handbook of economic forecasting 2*(A), 195–237.

Bandiera, O., S. Hansen, A. Prat, and R. Sadun (2017). CEO behavior and firm performance. *NBER Working Paper No. 23248*.

Belloni, A., V. Chernozhukov, and C. Hansen (2013). Inference for high-dimensional sparse econometric models. In *Advances in Economics & Econometrics: Tenth World Congress*, Volume 3, pp. 245–295. Cambridge University Press.

Best, M. C., J. Hjort, and D. Szakonyi (2017). Individuals and organizations as sources of state effectiveness, and consequences for policy. *NBER Working Paper No. 23350*.

Bickel, P. J., Y. Ritov, and A. B. Tsybakov (2009). Simultaneous analysis of lasso and Dantzig selector. *Annals of Statistics 37*(4), 1705–1732.

Bishop, C. M. (1995). *Neural networks for pattern recognition*. Oxford university press.

Bishop, C. M. (2006). *Pattern Recognition and Machine Learning*. Springer.

Blei, D. M. (2012). Probabilistic topic models. *Communications of the ACM 55*(4), 77–84.

Blei, D. M. and J. D. Lafferty (2006). Dynamic topic models. In *Proceedings of the 23rd International Conference on Machine Learning*, pp. 113–120. ACM.

Blei, D. M. and J. D. Lafferty (2007). A correlated topic model of *Science*. *Annals of Applied Statistics 1*, 17–35.

Blei, D. M., A. Y. Ng, and M. I. Jordan (2003). Latent Dirichlet allocation. *Journal of Machine Learning Research 3*, 993–1022.

Bollen, J., H. Mao, and X. Zeng (2011). Twitter mood predicts the stock market. *Journal of Computational Science 2*(1), 1–8.

Bolukbasi, T., K.-W. Chang, J. Y. Zou, V. Saligrama, and A. T. Kalai (2016). Man is to computer programmer as woman is to homemaker? Debiasing word embeddings. In *Advances in Neural Information Processing Systems*, pp. 4349–4357.

Born, B., M. Ehrmann, and M. Fratzscher (2014). Central bank communication on financial stability. *Economic Journal 124*(577), 701–734.

Boudoukh, J., R. Feldman, S. Kogan, and M. Richardson (2016). Information, trading, and volatility: Evidence from firm-specific news. Working Paper.

Breiman, L. (2001). Random forests. *Machine Learning 45*(1), 5–32.

Breiman, L., J. Friedman, C. Stone, and R. Olshen (1984). *Classification and Regression Trees*. The Wadsworth and Brooks-Cole statistics-probability series. Taylor & Francis.

Buehlmaier, M. M. and T. M. Whited (2016). Are financial constraints priced? Evidence from textual analysis. *Simon School Working Paper No. FR 14-11*.

Bühlmann, P. and S. Van De Geer (2011). *Statistics for high-dimensional data: methods, theory and applications*. Springer Science & Business Media.

Candes, E. J., M. B. Wakin, and S. P. Boyd (2008). Enhancing sparsity by reweighted L_1 minimization. *Journal of Fourier Analysis and Applications 14*(5-6), 877–905.

Carvalho, C. M., N. G. Polson, and J. G. Scott (2010). The horseshoe estimator for sparse signals. *Biometrika 97*(2), 465–480.

Chen, D. and C. Manning (2014). A fast and accurate dependency parser using neural networks. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pp. 740–750.

Chernozhukov, V., D. Chetverikov, M. Demirer, E. Duflo, C. Hansen, W. Newey, and J. Robins (2018). Double/debiased machine learning for treatment and structural parameters. *The Econometrics Journal 21*(1), C1–C68.

Chinco, A. M., A. D. Clark-Joseph, and M. Ye (2017). Sparse signals in the cross-section of returns. *NBER Working Paper No. 23933*.

Choi, H. and H. Varian (2012). Predicting the present with Google Trends. *Economic Record 88*(s1), 2–9.

Cook, R. D. (2007). Fisher lecture: dimension reduction in regression. *Statistical Science 22*(1), 1–26.

Cowles, A. (1933). Can stock market forecasters forecast? *Econometrica 1*(3), 309–324.

Das, S. R. and M. Y. Chen (2007). Yahoo! for Amazon: sentiment extraction from small talk on the web. *Management Science 53*(9), 1375–1388.

Deerwester, S., S. T. Dumais, G. W. Furnas, T. K. Landauer, and R. Harshman (1990). Indexing by latent semantic analysis. *Journal of the American Society for Information Science 41*(6), 391–407.

Denny, M. J. and A. Spirling (2018). Text preprocessing for unsupervised learning: why it matters, when it misleads, and what to do about it. *Political Analysis*, 1–22.

Efron, B. (1979). Bootstrap methods: Another look at the jackknife. *The Annals of Statistics 7*(1), 1–26.

Efron, B. (2004). The estimation of prediction error: covariance penalties and cross-validation. *Journal of the American Statistical Association 99*(467), 619–632.

Efron, B. (2012). Bayesian inference and the parametric bootstrap. *The Annals of Applied Statistics 6*(4), 1971–1997.

Engelberg, J. E. and C. A. Parsons (2011). The causal impact of media in financial markets. *Journal of Finance 66*(1), 67–97.

Evans, J. A. and P. Aceves (2016). Machine translation: Mining text for social theory. *Annual Review of Sociology 42*, 21–50.

Fan, J. and R. Li (2001). Variable selection via nonconcave penalized likelihood and its oracle properties. *Journal of the American Statistical Association 96*(456), 1348–1360.

Fan, J., L. Xue, and H. Zou (2014). Strong oracle optimality of folded concave penalized estimation. *Annals of Statistics 42*(3), 819–849.

Flynn, C., C. Hurvich, and J. Simonoff (2013). Efficiency for regularization parameter selection in penalized likelihood estimation of misspecified models. *Journal of the American Statistical Association 108*(503), 1031–1043.

Foster, D. P., M. Liberman, and R. A. Stine (2013). Featurizing text: Converting text into predictors for regression analysis. *The Wharton School of the University of Pennsylvania*.

Friedman, J. H. (2002). Stochastic gradient boosting. *Computational Statistics and Data Analysis 38*(4), 367–378.

Gentzkow, M. and J. M. Shapiro (2010). What drives media slant? Evidence from U.S. daily newspapers. *Econometrica 78*(1), 35–71.

Gentzkow, M., J. M. Shapiro, and M. Taddy (2016). Measuring polarization in high-dimensional data: method and application to congressional speech. *NBER Working Paper No. 22423*.

George, E. I. and R. E. McCulloch (1993). Variable selection via Gibbs sampling. *Journal of the American Statistical Association 88*(423), 881–889.

Ginsberg, J., M. H. Mohebbi, R. S. Patel, L. Brammer, M. S. Smolinski, and L. Brilliant (2009). Detecting influenza epidemics using search engine query data. *Nature 457*(7232), 1012–1014.

Goldberg, Y. (2016). A primer on neural network models for natural language processing. *Journal of Artificial Intelligence Research 57*, 345–420.

Goldberg, Y. and J. Orwant (2013). A dataset of syntactic-ngrams over time from a very large corpus of english books. In *Second Joint Conference on Lexical and Computational Semantics (* SEM)*, Volume 1, pp. 241–247.

Goodfellow, I., Y. Bengio, and A. Courville (2016). *Deep learning*. MIT Press.

Greenstein, S., Y. Gu, and F. Zhu (2016). Ideological segregation among online collaborators: Evidence from wikipedians. *NBER Working Paper No. 22744*.

Griffiths, T. L. and M. Steyvers (2004). Finding scientific topics. *Proceedings of the National Academy of Sciences 101*(suppl 1), 5228–5235.

Grimmer, J. (2010). A Bayesian hierarchical topic model for political texts: measuring expressed agendas in Senate press releases. *Political Analysis 18*(1), 1–35.

Grimmer, J. and B. M. Stewart (2013). Text as data: The promise and pitfalls of automatic content analysis methods for political texts. *Political Analysis 21*(3), 267–297.

Groseclose, T. and J. Milyo (2005). A measure of media bias. *Quarterly Journal of Economics 120*(4), 1191–1237.

Hans, C. (2009). Bayesian lasso regression. *Biometrika 96*(4), 835–845.

Hansen, S., M. McMahon, and A. Prat (2017). Transparency and deliberation within the fomc: a computational linguistics approach. *The Quarterly Journal of Economics 133*(2), 801–870.

Hassan, T. A., S. Hollander, L. van Lent, and A. Tahoun (2017). Firm-level political risk: Measurement and effects. *NBER Working Paper No. 24029*.

Hastie, T., R. Tibshirani, and J. Friedman (2009). *The Elements of Statistical Learning*. New York: Springer.

Hoberg, G. and G. M. Phillips (2015). Text-based network industries and endogenous product differentiation. *Journal of Political Economy 124*(5), 1423–1465.

Hoerl, A. and R. Kennard (1970). Ridge regression: biased estimation for nonorthogonal problems. *Technometrics 12*(1), 55–67.

Hoffman, M. D., D. M. Blei, C. Wang, and J. Paisley (2013). Stochastic variational inference. *Journal of Machine Learning Research 14*(1), 1303–1347.

Hofmann, T. (1999). Probabilistic latent semantic indexing. In *Proceedings of the 22 Annual International SIGIR Conference on Research and Development in Information Retrieval*, pp. 50–57.

Horowitz, J. L. (2003). The bootstrap in econometrics. *Statistical Science 18*(2), 211–218.

Iyyer, M., P. Enns, J. Boyd-Graber, and P. Resnik (2014). Political ideology detection using recursive neural networks. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*, Volume 1, pp. 1113–1122.

Jegadeesh, N. and D. Wu (2013). Word power: a new approach for content analysis. *Journal of Financial Economics 110*(3), 712–729.

Joachims, T. (1998). Text categorization with support vector machines: Learning with many relevant features. In *European conference on machine learning*, pp. 137–142. Springer.

Johnson, H. A., M. M. Wagner, W. R. Hogan, W. Chapman, R. T. Olszewski, J. Dowling, and G. Barnas (2004). Analysis of web access logs for surveillance of influenza. *Studies in Health Technology and Informatics 107*(Pt 2), 1202–1206.

Jurafsky, D. and J. H. Martin (2009). *Speech and language processing: An introduction to natural language processing, computational linguistics, and speech recognition*. Prentice Hall, Pearson Education International.

Kass, R. E. and L. Wasserman (1995). A reference Bayesian test for nested hypotheses and its relationship to the Schwarz criterion. *Journal of the American Statistical Association 90*(431), 928–934.

Kelly, B., D. Papanikolaou, A. Seru, and M. Taddy (2018). Measuring technological innovation over the long run. Working Paper.

Kelly, B. and S. Pruitt (2013). Market expectations in the cross-section of present values. *The Journal of Finance 68*(5), 1721–1756.

Kelly, B. and S. Pruitt (2015). The three-pass regression filter: A new approach to forecasting using many predictors. *Journal of Econometrics 186*(2), 294–316.

Knight, K. and W. Fu (2000). Asymptotics for lasso-type estimators. *The Annals of Statistics 28*(5), 1356–1378.

Kogan, S., D. Levin, B. R. Routledge, J. S. Sagi, and N. A. Smith (2009). Predicting risk from financial reports with regression. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pp. 272–280. Association for Computational Linguistics.

Lazer, D., R. Kennedy, G. King, and A. Vespignani (2014). The parable of Google Flu: traps in big data analysis. *Science 343*(6176), 1203–1205.

Le, Q. and T. Mikolov (2014). Distributed representations of sentences and documents. In *International Conference on Machine Learning*, pp. 1188–1196.

LeCun, Y., Y. Bengio, and G. Hinton (2015). Deep learning. *Nature 521*(7553), 436–444.

Li, F. (2010). The information content of forward-looking statements in corporate filings—a naïve Bayesian machine learning approach. *Journal of Accounting Research 48*(5), 1049–1102.

Loughran, T. and B. McDonald (2011). When is a liability not a liability? Textual analysis, dictionaries, and 10-Ks. *Journal of Finance 66*(1), 35–65.

Lucca, D. O. and F. Trebbi (2009). Measuring central bank communication: an automated approach with application to FOMC statements. *NBER Working Paper No. 15367*.

Manela, A. and A. Moreira (2015). News implied volatility and disaster concerns. *Journal of Financial Economics 123*(1), 137–162.

Manning, C. D., P. Raghavan, and H. Schütze (2008). *Introduction to Information Retrieval*. Cambridge university press.

Mannion, D. and P. Dixon (1997). Authorship attribution: the case of Oliver Goldsmith. *Journal of the Royal Statistical Society, Series D 46*(1), 1–18.

Manski, C. F. (1988). *Analog Estimation Methods in Econometrics*. New York: Chapman & Hall.

McAuliffe, J. D. and D. M. Blei (2008). Supervised topic models. In *Advances in neural information processing systems*, pp. 121–128.

Mikolov, T., I. Sutskever, K. Chen, G. S. Corrado, and J. Dean (2013). Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pp. 3111–3119.

Morin, F. and Y. Bengio (2005). Hierarchical probabilistic neural network language model. In *Aistats*, Volume 5, pp. 246–252.

Mosteller, F. and D. L. Wallace (1963). Inference in an authorship problem. *Journal of the American Statistical Association 58*(302), 275–309.

Murphy, K. P. (2012). *Machine Learning: a Probabilistic Perspective*. MIT Press.

Ng, A. Y. and M. I. Jordan (2002). On discriminative vs. generative classifiers: A comparison of logistic regression and naive bayes. In *Advances in neural information processing systems*, pp. 841–848.

Pang, B., L. Lee, and S. Vaithyanathan (2002). Thumbs up? Sentiment classification using machine learning techniques. In *Proceedings of the Empirical Methods in Natural Language Processing (EMNLP)*, pp. 79–86.

Park, T. and G. Casella (2008). The Bayesian lasso. *Journal of the American Statistical Association 103*(482), 681–686.

Pennington, J., R. Socher, and C. D. Manning (2014). GloVe: global vectors for word representation. In *Proceedings of the Empirical Methods in Natural Language Processing (EMNLP)*, pp. 1532–1543.

Politis, D. N., J. P. Romano, and M. Wolf (1999). *Subsampling*. Springer Science & Business Media.

Polson, N. G. and S. L. Scott (2011). Data augmentation for support vector machines. *Bayesian Analysis 6*(1), 1–23.

Porter, M. F. (1980). An algorithm for suffix stripping. *Program 14*(3), 130–137.

Preis, T. and H. S. Moat (2014). Adaptive nowcasting of influenza outbreaks using google searches. *Royal Society open science 1*(2), 140095.

Pritchard, J. K., M. Stephens, and P. Donnelly (2000). Inference of polulation structure using multilocus genotype data. *Genetics 155*(2), 945–959.

Quinn, K. M., B. L. Monroe, M. Colaresi, M. H. Crespin, and D. R. Radev (2010). How to analyze political attention with minimal assumptions and costs. *American Journal of Political Science 54*(1), 209–228.

Rabinovich, M. and D. M. Blei (2014). The inverse regression topic model. In *Proceedings of the 31st International Conference on International Conference on Machine Learning*, pp. 199–207.

Roberts, M. E., B. M. Stewart, D. Tingley, E. M. Airoldi, et al. (2013). The structural topic model and applied social science. In *Advances in Neural Information Processing Systems Workshop on Topic Models: Computation, Application, and Evaluation*.

Rumelhart, D., G. Hinton, and R. J. Williams (1986). Learning representations by back-propagating errors. *Nature 323*(6088), 533–536.

Saiz, A. and U. Simonsohn (2013). Proxying for unobservable variables with internet document-frequency. *Journal of the European Economic Association 11*(1), 137–165.

Schwarz, G. (1978). Estimating the dimension of a model. *Annals of Statistics 6*(2), 461–464.

Scott, S. and H. Varian (2014). Predicting the present with Bayesian structural time series. *International Journal of Mathematical Modeling and Numerical Optimisation 5*(1-2), 4–23.

Scott, S. and H. Varian (2015). Bayesian variable selection for nowcasting economic time series. In *Economic Analysis of the Digital Economy*, pp. 119–135. University of Chicago Press.

Sim, Y., B. R. Routledge, and N. A. Smith (2015). The utility of text: The case of amicus briefs and the supreme court. In *AAAI*, pp. 2311–2317.

Stephens-Davidowitz, S. (2014). The cost of racial animus on a black candidate: evidence using Google search data. *Journal of Public Economics 118*(C), 26–40.

Stock, J. H. and F. Trebbi (2003). Retrospectives: who invented instrumental variable regression? *Journal of Economic Perspectives 17*(3), 177–194.

Stock, J. H. and M. W. Watson (2002a). Forecasting using principal components from a large number of predictors. *Journal of the American statistical association 97*(460), 1167–1179.

Stock, J. H. and M. W. Watson (2002b). Macroeconomic forecasting using diffusion indexes. *Journal of Business & Economic Statistics 20*(2), 147–162.

Sutskever, I., O. Vinyals, and Q. V. Le (2014). Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*, pp. 3104–3112.

Taddy, M. (2012). On estimation and selection for topic models. In *Artificial Intelligence and Statistics*, pp. 1184–1193.

Taddy, M. (2013a). Measuring political sentiment on Twitter: factor optimal design for multinomial inverse regression. *Technometrics 55*(4), 415–425.

Taddy, M. (2013b). Multinomial inverse regression for text analysis. *Journal of the American Statistical Association 108*(503), 755–770.

Taddy, M. (2013c). Rejoinder: efficiency and structure in MNIR. *Journal of the American Statistical Association 108*(503), 772–774.

Taddy, M. (2015a). Distributed multinomial regression. *Annals of Applied Statistics 9*(3), 1394–1414.

Taddy, M. (2015b). Document classification by inversion of distributed language representations. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing*, Volume 2, pp. 45–49.

Taddy, M. (2017a). Comment: A regularization scheme on word occurrence rates that improves estimation and interpretation of topical content. *Journal of the American Statistical Association*. Forthcoming.

Taddy, M. (2017b). One-step estimator paths for concave regularization. *Journal of Computational and Graphical Statistics 26*(3), 525–536.

Taddy, M., C.-S. Chen, J. Yu, and M. Wyle (2015). Bayesian and empirical bayesian forests. In *Proceedings of the 32nd International Conference on International Conference on Machine Learning*, pp. 967–976.

Taddy, M., M. Gardner, L. Chen, and D. Draper (2016). A nonparametric bayesian analysis of heterogenous treatment effects in digital experimentation. *Journal of Business & Economic Statistics 34*(4), 661–672.

Teh, Y. W., M. I. Jordan, M. J. Beal, and D. M. Blei (2006). Hierarchical Dirichlet processes. *Journal of the American Statistical Association 101*(476), 1566–1581.

Tetlock, P. (2007). Giving content to investor sentiment: the role of media in the stock market. *Journal of Finance 62*(3), 1139–1168.

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society, Series B 58*(1), 267–288.

Tibshirani, R., M. Wainwright, and T. Hastie (2015). *Statistical learning with sparsity: the lasso and generalizations*. Chapman and Hall/CRC.

Tong, S. and D. Koller (2001). Support vector machine active learning with applications to text classification. *Journal of machine learning research 2*(1), 45–66.

Tran, D., M. D. Hoffman, R. A. Saurous, E. Brevdo, K. Murphy, and D. M. Blei (2017). Deep probabilistic programming. In *International Conference on Learning Representations*.

Tran, D., A. Kucukelbir, A. B. Dieng, M. Rudolph, D. Liang, and D. M. Blei (2016). Edward: A library for probabilistic modeling, inference, and criticism. Working Paper.

Vapnik, V. (1996). *The Nature of Statistical Learning Theory*. New York: Springer.

Wager, S. and S. Athey (2017). Estimation and inference of heterogeneous treatment effects using random forests. *Journal of the American Statistical Association*. Forthcoming.

Wager, S., T. Hastie, and B. Efron (2014). Confidence intervals for random forests: The jackknife and the infinitesimal jackknife. *The Journal of Machine Learning Research 15*(1), 1625–1651.

Wainwright, M. J. (2009). Sharp thresholds for high-dimensional and noisy sparsity recovery using L_1-constrained quadratic programming (lasso). *IEEE Transactions on Information Theory 55*(5), 2183–2202.

Wainwright, M. J. and M. I. Jordan (2008). Graphical models, exponential families, and variational inference. *Foundations and Trends in Machine Learning 1*(1–2), 1–305.

Wisniewski, T. P. and B. J. Lambe (2013). The role of media in the credit crunch: the case of the banking sector. *Journal of Economic Behavior and Organization 85*(C), 163–175.

Wu, Y., M. Schuster, Z. Chen, Q. V. Le, M. Norouzi, W. Macherey, M. Krikun, Y. Cao, Q. Gao, K. Macherey, et al. (2016). Google's neural machine translation system: bridging the gap between human and machine translation. Working Paper.

Yang, Y., M. J. Wainwright, M. I. Jordan, et al. (2016). On the computational complexity of high-dimensional Bayesian variable selection. *Annals of Statistics 44*(6), 2497–2532.

Zeng, X. and M. Wagner (2002). Modeling the effects of epidemics on routinely collected data. *Journal of the American Medical Informatics Association 9*(6), S17–S22.

Zou, H. (2006). The adaptive lasso and its oracle properties. *Journal of the American Statistical Association 101*(476), 1418–1429.

Zou, H. and T. Hastie (2005). Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society, Series B 67*(2), 301–320.

Zou, H., T. Hastie, and R. Tibshirani (2006). Sparse principal component analysis. *Journal of computational and graphical statistics 15*(2), 265–286.

Zou, H., T. Hastie, and R. Tibshirani (2007). On the "degrees of freedom" of the lasso. *Annals of Statistics 35*(5), 2173–2192.