Chapter 4: Discovery*

Justin Grimmer† Margaret E. Roberts ‡ Brandon M. Stewart §

March 19, 2018

Abstract

Social science research is often presented as if the basic concepts—the way we organize the empirical world—are given. Consider some examples from recent research. McGhee et al. (2014) examine how moving from a closed primary—where only members of a party can vote—to an open primary—where any eligible voter can cast a ballot—affects ideological polarization in state legislatures. On its face, this question is clear and corresponds with deep questions about how to reform American politics. But it also implies a particular organization of the world. Primary elections have to be categorized according to who is eligible to vote and members of the legislature have to be placed in an ideological space.

Examples of the central role of concepts are numerous. Consider the research from King, Pan, and Roberts (2015) (KPR) on Chinese censorship that we discussed in Chapter 2. In that article, KPR examine what determines whether a social media post is censored. This requires viewing social media posts as either censored or not. And then requires a second organization based on the topic of the posts. Other research streams require a similar organization. For example, a major debate in the study of the incidence of war examines

*Incomplete and preliminary draft from forthcoming book manuscript. Please do not cite or distribute.

†Associate Professor, Department of Political Science, University of Chicago

‡Assistant Professor, Department of Political Science, University of California at San Diego

§Assistant Professor, Department of Sociology, Princeton University

how bargaining in front of an audience affects the prevalence of conflict (Schultz, 1998). This theory imposes an organization on negotiations as either occurring in public and whether two countries are at war.

In each example the concepts form the entire structure of the research project. They define what measures are necessary to construct, what causal questions can be asked, and what conclusions can be reached about the world. Yet, quantitative researchers have traditionally spent too little time inquiring about their conceptualizations, how they arrived at them, and interrogating how they might consider new ways to organize the world. This lack of attention is surprising for two reasons. First, it is surprising because all of our research agenda depends upon the basic concepts that we use when interrogating the world. Therefore, it is surprising that quantitative scholars are not more self conscious about where their concepts come from or work to develop methods to facilitate new concepts. Second, it is all the more surprising because there is a long tradition in qualitative research in careful development of new concepts. For example, grounded theory provides a general methodology that guides research from their initial field notes and interviews towards the generation of concepts and new hypotheses (Corbin and Strauss, 1990; Nelson, 2017).

In this chapter we provide a methodology for discovering and applying new ways of organizing our observations. In particular, we show how the combination of texts, quantitative methods for discovery, and careful reading can facilitate uncovering new concepts, reemphasize the importance of existing organizational schema, and facilitate new theoretical innovations. The new organization leads to new quantities to measure, causal relationships to infer, refine theories and even offer policy prescriptions. This process can be iterated repeatedly to not only provide new evidence on long-standing theoretical questions of interest in the social sciences, but also used to consider new questions as well.

In the process of building this methodology for discovery we introduce four text as data methods that will be useful across numerous other applications: unsupervised clustering

methods, topic models, low-dimensional embeddings, and methods for discovering separating words. Each of the methods provide new and insightful ways to look at a collection of text data. The output from the methods, coupled with the careful reading from analysts, provides the opportunity to see new ways of organizing data. And, as a result, provides the opportunity to discover potentially new concepts that can become the basis for further research. Each of the methods we introduce to facilitate discovery could be the subject of its own large manuscript (and several of the mentioned methods have several books dedicated to them). Rather than provide a comprehensive introduction to each all four large groups of methods, we focus instead on the intuition for each method, explain how the particular method fits within the process of conceptual discovery, and emphasize the features of the methods that are common and the characteristics that are distinct. We take this introductory approach because our goal is to help the reader understand how broad groups of methods fit into the process of social science. Once this intuition is obtained, it can be easily generalized to any particular method that the reader chooses—and we attempt to provide guidance on the literature in the citations below.

As we discuss in Chapter 2 we take an unapologetically sequential approach to inference. That is, we believe that the best scientific inferences and theories develop only after repeated tests, revisions of hypotheses, acquisition of new data, and new theorizing. This sequential approach to research—and developing concepts in particular—is often shocking to researchers who have been taught that the best research is deductive and involves all theorizing before looking at the data. As we emphasize throughout the text, most of the rules of "good" scientific research came from a time when data were sparse, but thinking was cheap. The absolute cost of thought has not changed, but data are now more readily available. The readily available data allows us to use some data to develop concepts and then discard data used to develop the concepts before developing measures and testing causal relationships. This approach avoids critiques of data mining—that it will lead to circular

definitions, overfitting, or fail to be meaningful (Armstrong, 1967)—as we explain below, because once we have a conception in hand it is ours and it is irrelevant how it was developed. What matters is if it provides an insightful way to look at the data. Further, we anticipate that many of the objections made about the use of factor analysis will be made about text as data approaches to conceptualization, and we explain why many of the pathologies that plagued factor analysis applications in earlier literatures are avoided with careful research design and gathering fresh data.

We begin this chapter further clarifying how a computational approach to conceptualization—where quantitative methods guide the discover of organizations—fits within the process of social scientific research. We introduce three principles for discovery to make this clarification: text as data models facilitate discovery and complement theory and substantive knowledge, there is no ground truth conceptualization, and how you discovered a conceptualization does not affect its usefulness.

Text as Data Models Facilitate Discovery and Complement Theory and Substantive Knowledge—It Does Not Replace Them The methods in this chapter are designed to aide the researcher—to suggest new ways of organizing data, or to confirm that existing organizations are present in data. The methods that we present in this chapter are not, however, a substitute for substantive knowledge. And we should be clear from the outset: there is no replacement for careful study and deep knowledge of social institutions. This might be surprising if we take seriously claims from overly optimistic futurists who have asserted in breathless articles about "The End of Theory: The Data Deluge Makes the Scientific Method Obsolete" (Anderson, 2008).

These sorts of overly optimistic declarations overstate the power of machine learning algorithms and understate the importance of qualitative human analysis. Before applying any model we have to identify interesting data to analyze. Interpreting the output of any of

the models we discuss in this book requires deep substantive knowledge of the case at hand. And knowing what to make of the findings from any model—and how those findings revise our understanding of the world—still requires that we have a theory in mind and know a great deal about the particular empirical example. For example, it is well known that in simple problems basic reasoning can trump machine learning based methods (Armstrong, 1967).

Rather than viewing the methods in this chapter as supplanting traditional theorizing, we argue that the methods we introduce are best thought of as complements to the traditional social scientific theory building process. The unification of traditional theorizing and computational approaches to searching for conceptualizations enables social scientists to both develop careful theories while also making use of massive data sets and computational power for theory development. By unifying the theorizing and exploration, we can develop better theories based on categories that we discover using computational methods. Or, we might refine our theories in light of discoveries about types of behavior. The methods in this chapter therefore contribute to the sequence of the scientific process, where the goal is theory development and deep substantive knowledge is a prerequisite for inference. And, it is important to note, the methods in this chapter can facilitate new knowledge about a substantive case not previously known and highlight tensions in previous theories.

There is No Ground Truth Conceptualization A normal goal when applying statistical procedures is to discover the true value of some parameter or to recover the population value of an estimand. This goal, however, is nonsensical when it comes to conceptualizations. It makes little sense to discuss the true conceptualizations because different organizations of our data merely imply different ways of viewing the world. There is no sense in which there is a true or false way of grouping our observations. For example, we might organize texts based on their topic, organize texts for another study based on tone, and a third study

might organize documents based on their author. Each of those organizations are correct for their particular application. It is difficult to ask which conceptualization is "right" because different organizational schemes are more and less useful for different applications.

While there is no true way to group our data, it does not mean that all organizations of the data are equally useful. Some organizations will be more useful for research. For example, we would expect that documents organized by topic will be more useful than documents organized by the third letter in the first word of each document. The usefulness of the organization will depend on how it can be applied and how well it can lead to new insights from data.

We can also objectively evaluate how well labels that we apply to categories in a conceptualization actually fit those categories. Once we organize documents into categories, or place our observations into a lower dimensional space we often want to label the categories and dimensions so that we understand what information the conceptualization is conveying. Once the labels have been applied to categories or dimensions, there are objective criteria to evaluate the quality of the conceptualization. We can ask if the label we apply to the categories accurately summarizes the distinguishing features of the category and if the mapping from the features of documents to the category ensures that the same kind of documents are classified into the category out of sample.

Once You Have A Conceptualization, It is Yours, You Can Use it How You Want, and It Doesn't Matter Where It Came From This chapter introduces methods for discovering new ways of organizing texts into categories, placing texts into a space, or discovering words that separate documents. A common concern when applying any statistical method is that our procedures will lead to biased inferences or that we might "overfit" our data. This concern is that we spend so much time and effort analyzing our data that we end up modeling the random noise in our sample, rather than the systematic features of the

texts. A related problem is that discovering conceptualizations through automatic procedures might appear to be atheoretical or undermine the theoretical basis for our results.

We are also less concerned about overfitting and where a conceptualization comes from because there is no meaningful sense in which a conceptualization can be biased, because as we just discussed, there is no true single conceptualization of a corpus. We should be clear about what this point means. Given any conceptualizations we can obtain biased measures or biased causal inferences. But there is no sense in which the explicit organization we discover could be biased, because there is no single true conceptualization in the world.

This is because a conceptualization provides a way of organizing the data and there are an incredibly large number of ways to organize a collection of observations. So, there is no sense in which it can be right or wrong—it can just be more or less useful *for the particular task at hand*. This means that the process that gave rise to the conceptualization will matter little when determining the value of the new conceptualization. Regardless of where the organization comes from, once the researcher has the organizational schema in mind, she can use it to create measurements, test hypotheses, and update theories appropriately. Further, this implies that the only meaningful way to evaluate a conceptualization is based on its usefulness for the particular application. For example, we might ask if the organization helps us to better understand the primary contours of conflict in a legislature, the primary goal of censors when limiting posts on social media, or the content of war negotiations. But there is no meaningful sense in which we can make application independent declaration that organizing legislators according to ideology is "better" than organization according to tenure in the legislator. The value comes from the particular application.

Given that all that matters for a conceptualization is its usefulness to any application, there is no reason to prioritize where the idea for the organization comes from. That is because the way to evaluate a conceptualization is how useful it is for the inferences you make. On the one hand, this frees the researcher to do all the exploration she wants when

deciding how to organize her data. On the other hand, though, it means that evaluating the methodology for discovery is extremely difficult. There are no proofs that we could write down that would show some methods are better at discovering organizations of data than others. And there are no analogous monte carlo simulations that could show that one method does better on average than another.

This all implies that when attempting to discover new research conceptualizations, researchers should be willing to explore many different methods and many different ways of organizing the data. Statistical methods and computational algorithms based on clear and precise assumptions about the underlying organization can yield insightful ways of organizing data. But there are many ways to implement similar ideas about what constitutes a good organization and many intuitive properties we might want conceptualizations to have. Varying these assumptions is essential to try and uncover more interesting and useful ways of organizing the data. And it also means that there is no real sense in which the assumptions of the model much matter—other than their ability to produce useful organizations. Once researchers have a conceptualization, researchers can use it for whatever purpose they choose and apply it to whatever data set they would like to use.

In many instances we will use methods that automatically discover categories and then classify all the documents into those categories. When evaluating these conceptualizations for their usefulness it is essential that we use data that are not contained in the original data set used to form the conceptualizations. The use of external data ensures that the categories we discover are not merely artifacts of our particular data set and the labels that we place on the categories or dimensions mean what we claim they do. In other instances, we will recommend using distinct data sets to first discover an organizational schema and then use a distinct data set for measuring prevalence of the categories and testing hypotheses related to the categories. This is particularly true when using text as data methods to make causal inferences. If we fail to divide the data into two distinct data sets we run the risk of

violations of assumptions that we need to hold to make valid causal inferences.

Ideally, we would use fresh data. But even if this is impossible we can use the train/test split that we discuss in Chapter 2 to facilitate the analysis. That is, we can discover an organization of our texts on a subset of data and then analyze the prevalence of those concepts, test causal relationships between the two of them, or reach descriptive conclusions on a fresh data set.

1 Conceptualizing the US Congress

To motivate how statistical models can facilitate new insights, inferences, and theories in text, we turn to a discussion of how similar methods have facilitated a research agenda based on Congressional roll call voting data. In particular, we explain how a new conceptualization, generated using a statistical model, lead to new insights into the behavior of how legislators in the US Congress and how this conceptualization leads directly to new research questions, new theories, and ultimately a better understanding for society about how the US Congress behaves.

1.1 Conceptualizing Conflict in the US Congress

Scholars of American political institutions develop theories to explain how legislators' diverse preferences are aggregated to reach decisions, in order to understand when and how Congress shapes public policy. In order to study how Congress works, scholars have contributed numerous ways of conceptualizing its members, the votes taken in the institution, and the salient dimensions of conflict. Each conceptualization contributes a distinct organization of members, votes, and dimensions of conflict, which itself implies a particular view on how Congress creates public policy. For example, MacRae (1965) argues that underlying Congressional roll call votes are 6-8 voting blocs that emerge depending upon the legislative

content of a piece of legislation. Other conceptualizations emphasize ephemeral coalitions that emerge occasionally, such as the "conservative coalition" that would emerge around civil rights issues in the 1960's. There is also a long tradition in American politics to describe politicians along an ideological spectrum. Politicians are often assessed, declare that their opponents are, or declare themselves to be on the "far-left liberal", "liberal", "moderate" "conservative", or "far-right conservative". And others allege that candidates are "fascists", "communists", or "socialists". Each label corresponds to a location on the ideological spectrum.

The most consequential conceptualization of US legislators' voting records comes from the VoteView project (Poole and Rosenthal, 1997). Beginning with seminal work published in the early 1980s Keith Poole, Howard Rosenthal, and collaborators developed low-dimensional measures of where legislators fall on an ideological spectrum (Poole and Rosenthal, 1985). The organization was viewed as audacious when it was first introduced: there was deep skepticism that a single dimension could capture the salient dimensions of conflict within Congress (Poole and Rosenthal, 1997) or the origins of the dimensions (Snyder Jr, 1992). The evidence for this organization and assertion came from a discovery of where legislators fell in the ideological space. This particular conceptualization emphasized a representative's "ideal point" but suppressed a large number of other features of Congressional conflict and compressed the entire roll call voting record to a single point on a dimension. Not surprisingly, numerous scholars objected to the conceptualization as too simplistic to understand how Congressional conflict worked.

And yet, after over 30 years of analysis, the VoteView project, often in the form of NOMINATE scores, has become the default conceptualization of members of Congress. It is utilized in nearly every empirical paper about the US Congress and forms the basis for theoretical models of how Congress works. It has become the subject of its own theoretical literature, which seeks to explain why the voting in the US Congress is so low-dimensional and

inspired attempts to recreate the literature in legislatures outside of the US (Snyder Jr, 1992). This organization of the US Congress has inspired a large methodological literature that seeks to extend the particular organization of members of Congress to candidates for office (Shor and McCarty, 2011), donors (Bonica, 2013), social media users (Bond and Messing, 2015; Barberá, 2014), and voters (Bafumi and Herron, 2010).

The organization of legislators that comes from the voteview project is developed inductively, but it has been instrumental to further develop deductive theories of the US Congress. That is, even though the dimensions of DW-NOMINATE are not specified *a priori* it has still become a vital tool for the work of theoretical studies of Congress. For example, it is used to assess theoretical innovations about the organization of Congress (Krehbiel, 1998), the structure of interbrach bargaining (Cameron, 2000), and the nature of political representation (Canes-Wrone, Brady and Cogan, 2002). In turn, the use of DW-NOMINATE to assess theoretical models has pushed scholars of other legislatures to develop analogous measure of legislators' ideological positions using a variety of data sources, including political speech (Laver, Benoit and Garry, 2003).

Of course, there are other important ways to organize members of Congress and their actions—organizations that suggest different research questions and different inferences about the way Congress operates. Consider, for example, conceptualizations that are used in the institution. Legislators are organized into leadership, they are placed on committees that have more or less power, or they are members of party caucuses. Organizing legislators in this way to lead to different questions and key measures. For example, Berry and Fowler (2015) ask whether legislators who are on the Appropriations committee are able to deliver more money to their district. Other work includes a number of conceptualizations on legislator's behavior. We might ask how leaders came to be powerful (Powell, N.d.). A rich literature asks how representative Congressional committees are of the institution—questions that depend on organizing legislators based on the committees they sit on and their place in the

ideological spectrum (Krehbiel, 1990).

Still other conceptualizations are based on the way legislators communicate with their constituents. Fenno (1978) organized legislators based on the kind of issues they engaged with their constituents. Grimmer (2013) employs a similar organization of senators, placing them on a spectrum ranging from senators who focused their rhetoric on broad national issues to senators who focus on claiming credit for money delivered to their district. This organization leads to other important measurements—where do legislators fall on the pork/policy spectrum—and questions that lead to inferences about why legislators adopt particular styles, how those styles affect the way constituents evaluate their elected officials, and how differences in who adopts those styles affects contributions to debates.

Each of the examples also demonstrates that all social science inferences depend upon our conceptualizations. To study the origins of polarization, we have to assume that legislators' can be located in an ideological space and that this space is defined by the organization of legislators in it. Likewise, to study how legislator's appeals affect constituents' evaluations, we need a map for organizing what legislators say and how they say it. The particular organization that we assume facilitates certain hypotheses, while also making other questions impossible or nonsensical. This is why the particular organization is essential: all of our conclusions depend upon how we decide to organize the world from the start.

Conceptualizations have a pervasive influence in research, and yet, very little quantitative work engages methods for developing new conceptualizations. Even in the literature on the US Congress the application of DW-NOMINATE scores has far outpaced attempts to develop new organizations to explain the structure of Congressional conflict. The lack of attention to developing new conceptualizations is problematic because it severely limits what we study when study Congress—and when we study any other social science research. It is limiting because the lack of methods, or the anxiety about producing new schema, means that scholars will adopt organizations that prior researchers have used, or adopt conceptualizations that

are well established from the institution. Certainly, it is important that scholars accumulate evidence and share a common perspective. But this will also necessarily limit the questions that we ask and therefore limit the inferences we can make from our data.

1.2 Four Methods for Discovery

In this chapter we describe methods that use text data to facilitate discovery of conceptualizations. Our goal throughout will be to find conceptualizations that help us pose new questions, develop new measures, estimate causal effects, with the ultimate goal of new theoretical insights into social sciences. We organize the chapter around four broad ways to discover new organizations of texts: unsupervised clustering analysis, topic models, low-dimensional embeddings, and word separating algorithms. Each class of methods could be its own book and research is quickly developing in each area. Therefore, our focus will be on clarifying the goals of inference and providing exemplars of each method group. Before engaging in this discussion of the methods, we first summarize each method briefly, describe the particular kind of g it employs to compress the data, and how the method facilitates discovery.

Unsupervised Clustering Analysis Unsupervised clustering analysis methods produce a *partition* of the texts: every document is assigned to one of K categories. But rather than assume the categories are known before hand, unsupervised clustering analysis methods discover the categories, attempting to assign documents to the same cluster that are similar and assign distinct documents to different categories. Cluster analysis functions therefore map from documents to categories, where the categories that are discovered depend on the entire collection of texts. Clustering methods facilitate discovery by suggesting new organizations of texts, which can alter the way researchers view their text collections, encouraging new insights into texts.

Topic Models Topic models are closely related to unsupervised clustering analysis methods. Like clustering methods, topic models discover a set of K categories. Unlike clustering methods, topic models assume each document is created as an admixture of the underlying categories. This key conceptual difference implies that topic models are able to inform the relative prevalence of the categories within documents, which is particularly useful for corpora with documents that contain several themes. Topic models, therefore, map from a document to the $K-1$ dimensional simplex—a K component vector with each entry greater than 0 and the sum of the vector is equal to 1. Like clustering methods, topic models suggest new organizations to facilitate discovery, while also providing insights into the relative prevalence of topics across a collection of documents.

Low-Dimensional Embeddings Low-dimensional embeddings represent high dimensional texts with a low-dimensional summary. The general goal is to use fewer pieces of information to preserve as much information as possible. For example, DW-NOMINATE embeds the entire roll call voting history for a member of Congress in a session into a single number. The g for low-dimensional embeddings maps from a high-dimensional document to values in a much smaller $K-$dimensional space. The methods facilitate discovery by extracting underlying systematic features of the texts, highlighting the most salient correlations between documents.

Word-Separating Algorithms Word-separating algorithms identify words that separate two pre-determined categories. The goal is to find words that are indicative of a particular category: used disproportionately often by documents in one particular category. Rather than provide a lower-dimensional representation of each text, the g for word-separating algorithm maps each word to a particular value, where the value depends on the entire collection of documents. Word-separating algorithms identify salient words that distinguish two groups. This enables researchers to discover the language associated with each category.

To understand how text as data methods can facilitate discovery, we first turn to unsupervised clustering analysis.

2 Unsupervised Clustering Analysis

The goal of unsupervised clustering analysis is to estimate a set of K categories and place the documents into those K categories or partition the data. This facilitates discovery because the organizations are discovered as part of the process and along the way features that characterize the categories are also estimated. For each observation we will estimate its cluster assignment $\boldsymbol{\pi}_i = (\pi_{i1}, \pi_{i2}, \ldots, \pi_{iK})$, where each π_{ik} corresponds to the share of document i that is assigned to cluster k. If $\pi_{ik} \in \{0, 1\}$, then we will say that the partition is *hard* and if $\pi_{ik} \in [0, 1]$ then we will say that the partition is *soft*. The basic goal of both hard and soft clustering methods is to partition documents so that similar documents are in the same cluster or category and dissimilar documents are assigned to different clusters.

This basic task seems straightforward, but there is inherent ambiguity in several key steps, leading to a proliferation of methods for partitioning text data. The largest group of clustering methods are Fully Automated Clustering (FAC) methods (Grimmer and King, 2011): clustering methods that take as an input a collection of texts (or other observations) and automatically output a set of categories and documents assigned to those categories. FAC methods only involve the researcher after the model is fit in order to label the categories and to assess their interpretability. In contrast, a much smaller class of methods called Computer Assisted Clustering (CAC) methods involve the researcher throughout the clustering process, exploring many different organizations of the text, with a final partition emerging only after exploring and considering many potential partitions. Certainly each approach has its advantages, but those advantages depend on the context the methods are applied and the researcher's basic goals. We begin our discussion of clustering methods with FAC methods,

because CAC methods depend on FAC methods and FAC methods are far more standard in the literature on unsupervised clustering.

To focus our intuition and to provide a reference for our general discussion of fully automated clustering algorithms, we begin with the canonical k-means clustering algorithm. We then describe general properties of fully automated clustering algorithms and describe two more recently developed algorithms: affinity propagation and mixtures of von Mises-Fisher distributions.

2.1 K-Means Clustering

As we described in Chapter 3, we suppose that we have preprocessed our N texts, so that each text is a $J \times 1$ count vector, $\boldsymbol{W}_i$. Our goal is to partition our observations into a set of K categories, where documents that are similar to each other are assigned to the same cluster and dissimilar documents assigned to different categories. We will suppose that each of the K categories has a $J \times 1$ mean $\boldsymbol{\mu}_k = (\mu_{1k}, \mu_{2k}, \ldots, \mu_{Jk})$. We can think of $\boldsymbol{\mu}_k$ as the center of the k^{th} cluster and μ_{jk} will describe the average rate that documents that belong to the k^{th} cluster use the j^{th} feature. Our goal, restated, will be to find a set of cluster centers $\boldsymbol{\mu}$ and a partition of our documents $\boldsymbol{\pi}$ so that documents are close to their assigned cluster centers.

We can make this intuition precise. We suppose that we will measure the dissimilarity between a document and the cluster center as the squared Euclidean distance:

$$d(\boldsymbol{W}_i, \boldsymbol{\mu}_k) \;=\; \sum_{j=1}^{J} (W_{ij} - \mu_{jk})^2 \tag{2.1}$$

Using this measure of dissimilarity, we can assess the quality of any partition $\boldsymbol{\pi}$ and any set of cluster centers $\boldsymbol{\mu}$. The k-means algorithm makes *hard* assignments, so that each document is either assigned to a category or not. Formally, $\pi_{ik} \in \{0, 1\}$. We can use the

fact that K-means makes hard assignment to write the objective function that evaluates the quality of any proposed solution as :

$$f(\boldsymbol{\pi}, \boldsymbol{\mu}, \boldsymbol{W}) = \sum_{i=1}^{N}\sum_{k=1}^{K}\sum_{j=1}^{J} \overbrace{\pi_{ik}}^{\text{Cluster indicator}} \underbrace{(W_{ij} - \mu_{jk})^2}_{\text{dissimilarity measure}} \tag{2.2}$$

In words, the objective function measures the dissimilarity of documents from their assigned cluster centers, using the definition of dissimilarity from Equation 2.1.

Given this objective function, there is a well defined best partition and corresponding set of cluster centers. Unfortunately, optimizing Equation 2.2 with respect to the cluster assignments and cluster centers is far from straightforward. Because cluster assignments are discrete, the most familiar optimization methods (involving solving first-order conditions) are not applicable. Instead, the K-Means algorithm relies upon an iterative algorithm to optimize Equation 2.2. We begin with a set of random starting values for a subset of our parameters. In our case, we begin with a random initialization of the cluster centers. We will call this collection of parameters $\boldsymbol{\mu}^0$. Given this initial set of cluster centers, we then obtain the optimal cluster assignments: each document is assigned to the cluster center it is closest to. We call this set of cluster assignments $\boldsymbol{\pi}^1$. Then, given those new cluster assignments, we update the cluster centers. A straightforward derivation shows that update value for the k^{th} center, μ_k^1 is:

$$\mu_k^1 = \sum_{i=1}^{N} \frac{\pi_{ik}^1 \boldsymbol{W}_i}{\pi_{ik}^1}$$

or the average of the documents assigned to the k^{th} cluster center. We continue updating the parameters until the change in the objective function, Equation 2.2, drops below a small

threshold. The algorithm then returns estimates of the optimal cluster centers and partition $\boldsymbol{\mu}^*$, $\boldsymbol{C}^*$. Table 1 provides pseudocode for the algorithm.

Table 1: Pseudocode for the K-Means Algorithm

- Initialize a set of K cluster centers, $\boldsymbol{\mu}^0$
- While the change in the objective function remains above the threshold,
    - Set $\pi_{ik} = 1$ if document i is closest to center k, set to 0 otherwise
    - Set $\mu_k^t = \sum_{i=1}^{N} \frac{\pi_{ik} \boldsymbol{W}_i}{\pi_{ik}}$
- Return $\boldsymbol{\mu}^*$ and $\boldsymbol{\pi}^*$.

While this algorithm will often provide useful partitions, there is no guarantee that it will provide the overall optimal solution. Rather, the algorithm we just described is an approximation of the optimal partition. This has important implications for our analysis: the iterative algorithm we just described will often get stuck in local optima, causing instability in the solutions across repeated runs of the algorithm. The instability arises because different initializations of the cluster centers imply different local optima that the algorithm will settle in. To mitigate the influence of this instability, there are numerous approaches to initialization that result in less instability and better solutions—as measured by the objective function (Pena, Lozano and Larranaga, 1999; Celebi, Kingravi and Vela, 2013).

2.2 Clustering *Cluster* Papers from Archive

In order to show how K-Means performs when applied to a real example we analyze 10,000 articles from the *ArXiv* server that use the word *cluster*. Specifically, we used the *ArXiv* API to download the 10,000 most recently posted papers that have the word *cluster*. After downloading the papers, we preprocessed their *ArXiv* provided summary using some of the techniques we discussed in Chapter 3. That is, for each of the summaries we discarded

Table 2: Applying K-Means to the *ArXiv* Summaries

Cluster Label	Words	Proportion of Documents
	clusters,globular,globular_clusters,star,star_clusters	0.23
	mass,0,galaxies,galaxy,ray	0.39
	clustering,data,algorithm,based,algorithms	0.12
	cluster,algebras,algebra,cluster_algebras,cluster_algebra	0.03
	cluster,star,galaxies,states,state	0.23

punctuation, made all words lower case, and discarded stop words. We then represented the texts as a document-term matrix, $\boldsymbol{W}$, using the 1,500 most used unigrams and the 500 most used bigrams across the articles.

With this representation of the texts we apply K-Means clustering to the texts. Specifically, we use the `R` function `kmeans` in order to find five clusters. Before applying the algorithm we remove the influence of document length. To do this, we normalize each row of the dtm. Specifically, for each document $\boldsymbol{x}_i$ we obtain the normalized version by dividing by the count of the number of words in the document

$$\boldsymbol{W}_i^* = \frac{\boldsymbol{W}_i}{\sum_{j=1}^{J} W_{ij}}$$

which ensures that the conclusions of our algorithm are not dependent on the number of words used. We can then collect all 10,000 normalized summaries into the matrix $\boldsymbol{W}^*$. We then apply the K-Means algorithm in `R`, using the `kmeans` function. We use the default settings: the Hartigan-Wong algorithm to optimize the objective function.

In Table 2 we provide a brief summary of the clusters from applying the K-Means algorithm. We label the clusters manually by reading summaries assigned to the documents, attempting to identify the common distinctive theme that characterizes documents assigned to the particular cluster. Below we discuss methods to do this sort of labeling—both manually and automatically—more extensively.

2.3 Fully Automated Clustering Methods

While K-Means is a straightforward and intuitive solution to the unsupervised clustering problem, the inherent ambiguity involved in forming a "good" partition has given rise to a massive literature on FAC methods, with substantial differences in how the individual methods are justified, what assumptions the models and algorithms make about the underlying structure of the data, and how well the methods scale to larger data sets. In spite of the differences, there are three components that all FAC methods share: a notion of document (dis)similarity, an objective function to measure the quality of a proposed partition, and a method for optimizing over the set of partitions. In this section we describe each feature of FAC methods, relate those features back to the K-Means algorithm we just described and then explain how they matter for the partitions that are obtained. We then apply different algorithms to the *ArXiv* data set and contrast the different clusterings to the K-Means clustering.

Table 3: Three Features of FAC Methods

1) Document (Dis)Similarity
2) Measure of Partition Quality
3) Optimization Algorithm

2.3.1 Feature 1: Document Dissimilarity

The first component of FAC methods is a measure of document similarity, or a measure of the distance between two documents. The measure of document similarity makes precise the intuition that partitions should capture documents that are similar. In the K-Means algorithm, dissimilarity is measured using the squared-Euclidean distance. Other methods can make use of a much broader set of functions, like those that we discussed in Chapter

3. For example, we might use a measure of cosine similarity between documents, calculate the Manhattan distance between documents, or use a kernel to measure the similarity of a pair of documents. Using *Affinity Propagation*, researchers are able to use any similarity metric when clustering observations (Dueck and Frey, 2007). There also kernel based methods that enable kernel k-means, or kernel versions of other canonical clustering methods (Spirling, 2012*a*). And for other clustering methods the notion of dissimilarity is built into basic assumptions of the model. This is most evident in statistical models for unsupervised clustering procedures. For example a mixture of von-Mises Fisher distributions implicitly adopts a measure of cosine similarity (Banerjee et al., 2005), a mixture of Normal distributions measure dissimilarity as a function of squared-Euclidean distance (Fraley and Raftery, 2002), and a mixture of multinomial distribution which is based on the probability the count values were generated from a particular multinomial distribution.

Choosing the similarity metric or distance metric is one of the most challenging tasks in text analysis. Intuitively, it seems easy to envision features of pairs of documents that would make them more of less similar. But implementing this intuitive notion of similarity into a metric that can be applied to pairs of documents is often much more difficult. The difficulty arises because a researcher's notions of what makes pairs of documents similar or dissimilar might be difficult to implement in a metric or difficult to reduce to the information a computer might have available. After all, humans are accustomed to reasoning about language in the context of a conversation, but we are providing computers with a very different representation of the information. Because unsupervised methods are often fast and easy to run, rather than carefully considering the metric beforehand, it will often be easiest to run several methods and compare the output after the fact.

While running several models and picking the output you like most might raise alarm bells for researchers more familiar with causal inference, this procedure is not problematic at all when discovering conceptualizations. This is because the purpose of discovery is to find

some useful representation of the observations and it does not matter how this discovery occurs. So long as fresh data are used to assess the prevalence of categories, there is no analogous problem of overfitting or "fishing" (Humphreys, Sanchez de la Sierra and Van der Windt, 2013).

2.3.2 Feature 2: Measure of Partition Quality

Given a notion of document (dis)similarity we can measure the quality of a partition. As we described above with K-Means, intuitively we know that a good partition of our documents will tend to group together documents that are similar and separate documents that are different. Making this intuition concrete, however, requires assumptions about what features of a partition that we will measure and which features of a partition are less important. For example, with K-Means we measure partition quality by summing up each document's dissimilarity from its cluster center. This achieves part of our intuitive objective—grouping together similar documents. But it does not include information about the distinctiveness of different clusters. Other objective functions will be based on a generative statistical model used to generate the documents. For example, with a mixture of multinomial distributions a "good" partition is one that is relatively likely given the observed data—or at the mode of the posterior distribution if including a prior to do Bayesian analysis.

Objective functions provide us with a clear standard for measuring the quality of partitions, but they do not provide us with an absolute measure of cluster quality that we can use to make comparisons across different FAC methods. Further, we can never suppose that a clustering is optimal because it comes from an FAC method. It is worth reemphasizing that, without clear knowledge about what is in the dataset or about the general goals of an analysis, it is *impossible* to distinguish between two partitions. This is because the objective functions are on different scales, are often based on different notions of document similarity, and might be useful for researchers engaged in particular kinds of projects. What objective

functions do provide are relative measures of cluster quality, which are perfect for selecting the "best" partition given a specific notion of best. Adjudicating between the different objective functions will be an essential clustering task.

2.3.3 Feature 3: Optimization Algorithm

The final feature of a FAC method is an algorithm to optimize according to the objective function. Optimization algorithms are necessary because finding the best partition according to an objective function is almost never straightforward. One reason optimization is difficult is that finding the best partition requires searching over discrete partitions, where the usual helpful rules from calculus are not applicable. The second reason is that the number of ways to partition even small sets of objects is massive. For example, the number of ways to partition 100 documents is greater than 4.75×10^{115}. This means that it is impossible to manually search over all potential solutions to choose the best one.

Finding the partition that is the global maximum, then, would require such a monumental effort that it would render FAC methods useless. Instead, the methods use approximate optimization approaches. The approaches approximate the optimization problem in different ways and then obtain the best solution according to that optimization procedure. While this means we give up on the guarantee of a best solution, it does mean that clustering methods will be useful for our research (a trade off that we find reasonable). There are a wide variety of approximation methods that include coordinate ascent methods like that used for K-Means (and the related EM algorithm), approximations based on graph-theory, and even variational approximations that generalize the EM algorithm. A vast literature introduces new methods for optimization and throughout the book we will introduce the algorithms as they are necessary to present the material we introduce.

The use of approximation algorithms often come at more cost than just giving up on the globally optimal solution. Approximation algorithms will often result in unstable solutions—

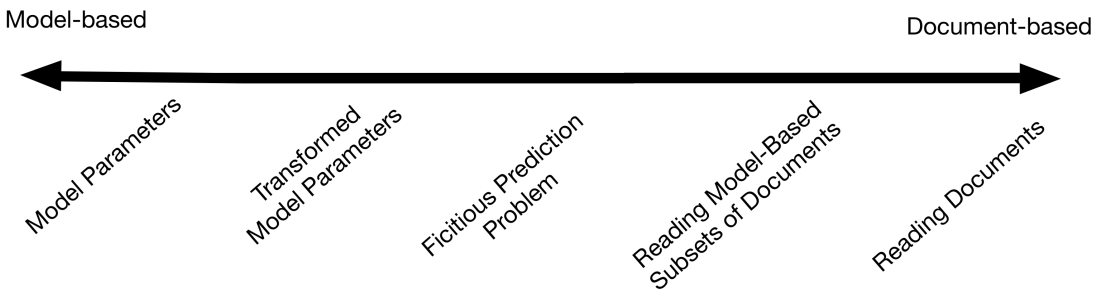


Figure 1: A spectrum of labeling methods for repurposed discovery methods.

running the same algorithm twice will sometimes result in different solutions. We can stabilize the approximations with careful starting values (Pena, Lozano and Larranaga, 1999; Celebi, Kingravi and Vela, 2013). But, as we discuss below, this instability is less problematic when we're using clustering methods for discovery. This is because when discovering new concepts implies that we are looking for interesting new ways of organizing our data. Even if a partition is only the optimal solution to approximate problem, it can still be useful for conceptualization.

2.4 Interpreting the Output of Clustering Methods

In this section we describe how to begin interpreting the output of FAC methods and labeling the components of the clusters. Labeling is ultimately a human exercise that involves careful consideration by the analyst. There are some useful heuristics though to assist the user in examining the learned structure. These different approaches exist along a continuum from model-based to document-based (depicted in Figure 1). Model-based approaches directly use the parameters of the model. While easy to use, these approaches may be blind to key considerations external to the model. Document-based approaches return to the original documents which make them more difficult and costly to employ; however, by avoiding dependence on the model, these approaches are often central to understanding what the g function is coding in practice.

.

Labeling is particularly important for FAC methods, As Quinn et al. (2010) argue, because unsupervised learning methods require little time investment upfront, but substantial work on interpretation. The most straightforward strategy, as described in the discovery chapter, is to simply use the estimated model parameters. In the case of FAC models, the estimated model parameters will often be over the word vocabulary. The usual procedure might be to select word parameters that are particularly large for a category. While we could define principled means of choosing the number of words—such as setting a cutoff value and reporting all word parameters bigger than that value—in practice and for the sake of symmetry, most software simply presents an arbitrary number of words without the associated weights.

Directly reading off the most probable words can be effective, but it will tend to favor words that are very common. Language is often filled with words that are extremely common but not very discriminating in terms of meaning. FAC models have to represent these words but they co-occur with everything and are thus spread across the clusters somewhat evenly. One way of dealing with this problem is to remove the frequent and uninformative words at the outset of the modeling. This is one of the motivations of stemming which we discussed back in Chapter 3. Originally, the popular consensus was that removing stopwords was both good for the model and for the labeling when they were not obviously related to the primary quantity of interest (e.g. as in Mosteller and Wallace 1963). A recent line of work by Schofield and Mimno suggests that this conclusion may have been premature as larger document collections seem to benefit from lighter preprocessing like leaving in stopwords and not stemming (Schofield and Mimno, 2016; Schofield, Magnusson and Mimno, 2017; Schofield et al., N.d.). Even when we remove stopwords from a pre-set list, there are often corpus-specific words which are highly frequent but relatively uninformative.

One solution which is still on the model-based side of the spectrum is to use some function

of the model parameters which is more informative. Roberts, Stewart and Tingley (2017) propose one metric based on the earlier work of Bischof and Airoldi (2012) and Airoldi and Bischof (2016) called FREX. FREX captures both the **Fr**equency and **Ex**clusivity of the words associated with a given topic by using the weighted harmonic mean between frequency of a word within a topic as well as exclusivity to the topic (both functions of the parameters under the model). By penalizing words which are frequent but not exclusive to the cluster, we get a better picture of what the topics are about even though the underlying model has not changed.

To interpret the output of clustering methods, we build on suggestions from Quinn et al. (2010) to label and interpret the output from clustering methods. Our algorithms provide us with a clear mapping from the features of a document to particular clusters. The goal at this stage is to translate this algorithmic rule into a substantive rule: the kind of rule that we could easily explain to manual coders. Indeed, we will argue later in this book that the best evaluation of clustering methods for many tasks will be to confirm that we can independently replicate the conceptualization from the unsupervised clustering documents using hand coding or supervised learning methods, discussed in Chapter 5.

Method 1: Identifying Distinctive Words While we can use model parameters to identify distinctive words, there are limitations to this approach. As we have already discussed, this might prioritize common words. And some FAC methods, like Affinity Propagation (which we discuss below), do not have an analogous parameter over the words because they do not estimate parameters on the vocabulary. And even when the parameters are available, a different method for calculating distinctive words ensures that we vary the assumptions we use when labeling the output of a clustering method.

A closely related idea, is to identify labels based on words which are good predictors of the topics, a strategy we call the fictitious prediction problem below. This leads to a lot of

useful metrics that we cover below, including χ^2 statistics, variance weighted log-odds ratios, and mutual information.

This strategy of running a predictive model on top of our unsupervised model might seem counterintuitive, but it can be an invaluable strategy when the original unsupervised model is quite complex. In a project seeking to understand the evolution of U.S. government treaties with Native Americans, Spirling (2012*b*) a string kernel PCA method to analyze the treaties. This is an important approach because it captures word order, which is important for the legal language of the treaties, but also is able to work on relatively small document collections (there are only a few hundred treaties). Unfortunately this leaves the task of interpreting the underlying dimension and the transformation in the string kernel means that the model parameters are, for all practical purposes, completely uninterpretable. Spirling (2012*b*) uses a random forest to predict the new dimension on the basis of the more interpretable original word count features. This provides a set of the most informative words for the dimension which provides a starting basis for interpretation.

To provide initial intuition and a tool for us to use now when labeling the output of clustering methods, we describe a simple approach: the t-statistic for testing the null that a regression coefficient is equal to zero. This approach builds on insights from Monroe, Colaresi and Quinn (2008), while providing computational ease.

Suppose that we have document term matrix $\boldsymbol{W}$ and we have a matrix of cluster assignments $\boldsymbol{\pi}$ and recall that $\pi_{ik} = 1$ if document i is assigned to the k^{th} cluster and $\pi_{ik} = 0$ otherwise. For each feature j we regress $\boldsymbol{W}_j$ on π_k, creating regression coefficient $\hat{\beta}_{jk}$ and standard error $\hat{\sigma}_{jk}$. We then create our score for each word $\text{score}_{jk} = \frac{\hat{\beta}_{jk}}{\hat{\sigma}_{jk}}$, which corresponds to the t-statistic for the null that the regression coefficient is equal to zero.

Given the scores we then identify the largest scores to summarize a particular model. For example, it is common to provide either the top 5, 10, or 20 words in publications. When working with clusters we recommend identifying a larger share, in order to get a better sense

of the features that make a particular category distinct.

Method 2: Sampling Documents Assigned to Each Cluster Following the advice of Quinn et al. (2010), the second approach that we recommend for labeling the cluster components is sampling documents assigned to each cluster component, reading those documents, and then generating labels by hand. We usually recommend reading between 10-30 of the posts assigned to each category. Our usual method is to carefully read the documents and write down a set of notes as they are read. We then try to synthesize those notes into a coherent label. If no label is readily available (or we struggle to provide a label), then we have good evidence that this particular cluster (or overall partition) might not be useful for discovery purposes.[1]

2.5 How do we select the number of clusters?

We have so far assumed that we know the number of clusters to include in our analysis, but often this is a quantity that individuals want to discover (along with the content of the clusters). In this section we first review common methods used to set the number of clusters in a clustering and explain why these methods are insufficient. We then provide a different strategy, that relies upon both statistical guidance and human evaluations.

2.5.1 Common Strategies for Determining the Number of Clusters

At first glance it might appear that the objective function used to obtain clusterings provides information to set the number of clusters. It might be tempting to try and use the machinery of FAC methods to make this determination. For example, it might (intuitively) seem that we could use the objective function from K-Means to compare the partitions that we obtain from K-Means. Unfortunately, in-sample fit is unable to provide a guide on how many clusters to

[1]Reading the documents serves two purposes. It is primarily useful for labeling the clusters, but reading texts grouped together in a new way can lead to new insights into the documents themselves.

include. This is because the K-Means objective function, like many other statistical models, improves as more cluster components (parameters) are added.[2] If we follow the advice from the objective function alone we receive the unhelpful suggestion of placing every document into its own cluster.

There are numerous quantitative methods that attempt to provide guidance on the number of clusters to include (Fraley and Raftery, 2002). The core intuition of the numerical approaches to determining the number of clusters is that a penalty term can be used to balance two competing concerns. On the one hand, we would like to have a sufficient number of clusters to capture the major variation across our documents and to find substantively interesting groups of texts. On the other hand, we would like to avoid too much model complexity: creating a large number of clusters that divide up very similar texts or create several clusters that group together essentially the same "type" of document. The statistical methods take several approaches to this problem. Perhaps one of the most prominent group of methods for determining the number of clusters builds on the objective function from clustering and embeds a penalty for additional cluster components. For example, when using a mixture model to cluster data there is easily available statistics such as the Akaike Information Criterion (AIC) or the Bayesian Information Criterion (BIC) (Fraley and Raftery, 2002), which penalize the addition of new clusters. This penalty ensures that to add a new clustering component we know that the improvement in model fit outweighs the penalty for additional parameters.

There are other penalties for model complexity that similarly try to encode a penalty.

[2]We can prove directly that the objective function for $K-$means is non-decreasing. Suppose there is an optimal clustering of a document collection with K clusters. Now suppose that we add one cluster and want to find a new solution with $K+1$ clusters. So long as there are distinct documents within each cluster, the objective function will improve. To see the objective function improve, take the observation that fits its cluster worst (and therefore contributes the most to the objective function) and move it to the new cluster. If the partition is perfect, in the sense that each document is identical within each clustering, then a new cluster component will not affect the objective function. While this may not be an optimal solution, moving one document either improves the clustering objective function or leaves it the same, so any optimal partition must make the objective function better.

Certainly the statistical approaches to determining the number of clusters can be useful, but they usually are very specific. That is, they rely upon specific models of the data generating process and specific asymptotic arguments. Even if we believe that this is a useful approach to selecting the number of clusters, we may have reason for concern about its performance in any one data set.

As an alternative, several FAC methods attempt to estimate the number of clusters as part of the estimation process. For example, affinity propagation does not require the researcher to set the number of clusters, but instead has a set of parameters that determine the number of clusters that are likely to emerge (Dueck and Frey, 2007). Specifically, the algorithm requires a specification of observations' self-similarity, which affects the propensity of that observation to be an exemplar, with more exemplars necessarily resulting in more clusters.

Nonparametric Bayesian methods provide a similar approach in statistical models. The most widely used nonparametric Bayesian prior, the Dirichlet Process Prior (DPP) or Chinese Restaurant Prior, is a prior over distributions, rather than parameters (Blackwell and MacQueen, 1973). The DPP has two components: a base measure, G_0 and a concentration parameter ξ. The concentration parameter exercises substantial influence over the number of clusters that are formed from the model (Wallach et al., 2010). Indeed, as the number of observations goes to infinity, the expected number of clusters from the DPP is $\xi \log(1 + \frac{N}{\xi})$ (Wallach et al., 2010). Further, the DPP assumes a particular process for cluster assignment that results in a "rich get richer" dynamic: a few clusters will have many documents assigned to it and many clusters will only have a few documents. Other nonparametric priors, like the Pitman-Yor prior, generalize the DPP and relax some features of the data-generating process. But the Pitman-Yor prior retains similar features, such as a few clusters receiving a large number of documents (Pitman and Yor, 1981). And still other nonparametric priors, such as the Uniform process prior (Wallach et al., 2010) provide a different data generating

process, but still make consequential modeling assumptions.

Both nonparametric and penalty-based approaches are applied to the full data set and attempt to correct the greediness of statistical models with a built in penalty. Other approaches to determining the number of clusters assesses how well additional clusters assist in predicting held out documents. For example, Computer Scientists use a variety of methods to measure how well clustering methods predict held out documents, including perplexity (Wallach et al., 2009). Adding additional clusters will always improve in sample fit, but too many clusters will result in overfitting, decreasing performance when predicting out of sample.

A shortcoming of quantitative approaches to model selection is that they are a blunt tool for selecting a final model and there can be only a weak relationship between the output from quantitative approaches to model selection and the most useful model for discovery perhaps. It is not surprising that there is only a blunt relationship between the statistics used for automatic model selection and the utility of the model for social science research. The objective function for FAC methods attempts to summarize the data "well" according to an objective function. This objective function can provide a useful organization of the texts, but it can be difficult to select between the model fits merely using a statistic. The problem is even more difficult, though, because there is only a weak relationship between the partitions automatic methods select and the partitions most useful for substantive research. Chang et al. (2009) show that, for a particular set of documents, there is a negative relationship between methods that receive a positive score from humans and the methods' score from automated evaluation methods. This is further exacerbated by the simplification of the text representation. Our preprocessing steps discard substantial information when we represent texts in a document-term matrix.

The shortcoming of automated methods is due, in large part, to the underspecified goal of discovery. The general goal—to find a useful clustering—is underspecified because we

are unsure about how to directly model "interesting". Further, it is generally impossible to define interesting before hand, even if we know what is interesting after the fact. The result is that merely using statistical procedures to discover categories will necessarily miss interesting organizations and will be insufficient to determine the number of categories.

2.5.2 Statistics, Experiments, and Careful Reading

Determining the number of clusters to include in a clustering necessarily requires the researcher to consider quantitative evidence, but also to think—to consider the particular problem she is confronting, the substantive goal of the project, and to evaluate distinct clusterings. This means that no one statistic is going to be sufficient to drive model selection, but statistics can still be useful. Rather, our preferred procedure will make use of statistics that ensure we choose clusterings that are as good as possible for a particular number of clusters, experiments that help us illicit credible human evaluations outside of the research team, and a manual deep inspection of different potential clusterings.

Statistics In addition to the statistics already described above, we introduce two additional statistics that are useful for selecting a particular clustering: cohesiveness and exclusivity (Roberts et al., 2014; Mimno et al., 2011). The focus on cohesiveness and exclusivity comes from our intuition about what makes a "good" clustering. A good clustering will identify groups of documents that have the same cohesive use of language: the content of documents in the same group are similar. Of course, as the number of clusters increases the groups of documents within each cluster will be more cohesive, but there might be several clusters that repeat the same basic content. Thus, a second property of a good clustering is that the clusters are exclusive: there are not several clusters that replicate the same basic content.

We follow discussions in the appendix of Roberts et al. (2014) and Mimno et al. (2011) to formalize this intuition. First, consider a definition of exclusivity. We will say that a

cluster is exclusive if the words that indicate membership in one cluster do not also indicate membership in other clusters. Specifically, suppose that each cluster has a center vector $\boldsymbol{\mu}_k = (\mu_{1k}, \mu_{2k}, \ldots, \mu_{Jk})$ where μ_{jk} describes the weight attached to the j^{th} word in cluster k. For each cluster we can select the M largest weights and collect the indices for the words that have the M largest weights into $\mathcal{M}$. For each word $m \in \mathcal{M}$ we can define the exclusivity as,

$$\text{Exclusivity}(m, k) \quad = \quad \frac{\mu_{m,k}}{\sum_{k=1}^{K} \mu_{m,k}}.$$

If a word is as exclusive as possible—it is only used in one cluster—then the exclusivity is 1. If the word is not exclusive at all and used equally across the cluster, then the score will be $\frac{1}{K}$. And if it is used more often in other clusters the score will be even smaller.

We can then aggregate up the exclusivity scores for the clusters by summing across the words and across clusters. For a particular clustering with K clusters, we can describe its average exclusivity as

$$\text{Exclusivity} \quad = \quad \sum_{m \in \mathcal{M}} \sum_{k=1}^{K} \frac{\text{Exclusivity}(m, k)}{K}$$

To measure cohesiveness we use the strategy adopted in Mimno et al. (2011) and examine the extent to which two words that indicate that a document belongs to a cluster actually co-occur in the documents that belong to that cluster. Call the function $D()$ a function that counts the number of times its arguments occur in documents. For example, if we provide the indices m_1 and m_2 then $D(m_1, m_2)$ will count the number of times the words m_1 and m_2 co-occur in documents, while $D(m_1)$ counts the number of documents in which the word m_1

appears. If we again collect all top M words into the set $\mathcal{M}$ we can then define cohesiveness of a cluster as,

$$\text{Cohesive} \quad = \quad \sum_{n=1}^{M} \sum_{l=1}^{l-1} \log \left(\frac{D(m_n, m_l)}{D(m_l)} \right) \tag{2.3}$$

And we can take the average across clusters to compute a clustering-level measure of cohesiveness.

As Roberts et al. (2014) note, we cannot compare cohesiveness and exclusivity across models with different numbers of clusters, because different clusters imply different constraints. Further, increasing the exclusivity necessarily will result in a drop of cohesiveness, so the statistics are unable to provide a specific recommendation on the single model to choose. But, the measures of cohesiveness and exclusivity can ensure that we end up on the cohesiveness/exclusivity *frontier*: the set of models that do the best job of managing the cohesiveness and exclusivity trade off.

Experiments Statistics are useful to guide decision making, but we can also incorporate credible human evaluations of the clustering. We examine two such experiments here: topic-intruder detection and overall cluster evaluation. Each attempts to inject human evaluation of the clustering while being attentive to cognitive limitations of humans as they engage with the content of clusters.

We consider first the topic-intruder experiment, first introduced in Chang et al. (2009). The intuition for the topic intruder experiment is that if a set of clusters is both cohesive and exclusive then we should be able to easily detect language that does not belong with a particular cluster. Suppose again that we have word weights that are indicative of a cluster $\boldsymbol{\mu}_k$ and suppose again that we have identified that top M words for each cluster. If a cluster is grouping together documents that have cohesive and exclusive language, then we should

expect that we could detect a top word from another category. To test this, we randomly select an intruder. Specifically, we randomly select a different cluster and then we randomly select one of the top words from that other cluster as the intruder. We then have a list that contains $M + 1$ words, M from our particular topic and one intruder word from the other topic.

The key to the topic-intruder experiment is asking experiment participants to identify the intruder word. The higher the proportion of topic intruder words detected the better the model performs under this human evaluation. The number of examples that are necessary to code to make meaningful comparisons depends on the topic-intrusion rate for the comparison models. In general, the lower the topic intrusion detection rate for clusterings the more examples that will need to be coded. This is because the variance of the topic intrusion rate is very low when the topic intrusion rate is very high.

Grimmer and King (2011) suggest a different approach to estimating the quality of a clustering and making comparisons. The intuition behind this cluster quality measure is that high-quality clusterings should group together pairs of documents that readers evaluate as similar and separate pairs of documents that are dissimilar. To evaluate this idea with human reading, the first step is to sample pairs of documents that are both assigned to the same cluster and pairs of documents that are assigned to different clusters. Then, evaluators are asked to rate the pairs of documents on a three point scale. Documents are given a 1 if they have no similarity, a 2 if there is some similarity, and 3 if the documents are very similar.[3] Using the evaluations from coders, the average evaluation for documents assigned to the same cluster are calculated and the average evaluation to different clusters is calculated. Finally, a cluster quality calculation is made:

[3]If a particular conceptual grouping is of interest more direction could be given to define similar, but we caution that if the analyst defines similar to bias selection to a particular model the value of doing the cluster quality evaluation is gone.

$$\text{Cluster Quality} \quad = \quad \text{Avg. Same Cluster} - \text{Avg. Different Cluster}$$

Clusterings will have higher cluster quality when documents assigned to the same cluster are evaluated higher than clusterings assigned to different clusters.[4]

Careful Reading Armed with measures of exclusivity, cohesiveness, and potentially experimental measures of particular clusterings we are close to ready to make a final selection of a particular number of clusters. Of course, the statistics alone are insufficient. This might be because the statistics and the experiments might offer contradictory recommendations. In particular, a subset of the clusterings might remain as potentially viable organizations. The only way to adjudicate between the clusterings is to careful consider the organizations, what they mean, and their implications for further analysis.

Reading the documents, guided by the organization, is essential for understanding the meaning of the clusterings. It is also an essential step in the discovery process. Closely engaging with the text and reading the documents in light of the organization tends to provide new insights and conceptualizations about how to observe the world.

2.6 The Wide Range of Clustering Models

In this section we describe the numerous clustering methods that exist. For example, there are groups of methods based on models (Fraley and Raftery, 2002), others based on algorithmic approaches (Dueck and Frey, 2007), and still others that appeal to graph-theory notions of community detection (Ng, Jordan and Weiss, 2002). Some methods search for a single

[4]If we are merely comparing two clusterings, then we only need to calculate cluster quality for pairs of documents where the two clusters disagree. This is because any pairs that are either placed in the same cluster in both or different clusters in both will not contribute to a final difference between the two evaluations. Further, note that any pair of documents can be used to calculate the cluster quality.

partitioning of the data, other methods build a hierarchy or tree-based clustering (Johnson, 1967). Some methods allow the center of a cluster to be estimated from the data and other methods constrain the center of the cluster to come from the data. The variety of the methods, and the number of algorithms, is astounding. In general, any attempt by us to pretend we could characterize this variation would be a fool's errand. It is foolish in part because the breadth of the field means that necessarily we will miss important models and fundamental distinctions that are made in that literature. It is also foolish because the rate of production of models is impressive. And finally, we think it is foolish because many of the arguments in favor of a particular class of clustering method are overstated, particularly when we used clustering methods to engage in discovery. Rather than list all the methods, then, instead we focus on the most prominent types of clustering methods, provide prominent examples, and explain why the different types of clustering approaches might matter for the clusterings that are discovered.

Model vs Algorithmic Perhaps the most salient division in clustering methods is between modeling and algorithmic approaches to clustering. Model based approaches define a probabilistic model to explain how the data are generated and then use a statistical procedure to infer the parameters of the model. The usual approach in model-based clustering methods is to use a mixture model, which has two components. First, there are the probabilistic distributions that form the components of the mixture. Second, there are weights attached to the distributions that describe the distribution's contribution to the mixture. For example, above we describe a mixture of multinomial distributions. Other common mixture models that are used include mixtures of normal distributions (Fraley and Raftery, 2002), mixtures of von mises Fisher distributions (Banerjee et al., 2005), and mixtures of Dirichlet distributions to characterize mixtures of proportions (Grimmer, 2013). Mixture models are useful for clustering because the estimation procedure assigns documents to a component of

the mixture, providing the cluster assignments, and then infer the features of the component distributions, which provides the characteristics of the clusters (McLachlan and Peel, 2004).

We will call clustering approaches algorithmic if they are not based on a probabilistic data generating process. Algorithmic models are often motivated with an appeal to intuition about what constitutes a useful clustering and then derive theorems based on metaphors about the clustering procedure. For example, spectral clustering methods have a close analogue to graph-cutting algorithms—procedures that attempt to find closely connected communities in a network (Ng, Jordan and Weiss, 2002). Similarly, affinity propagation uses message passing discover groups of high similarity documents (Frey and Dueck, 2007). And even the K-Means algorithm supposes that the data are divisible into a set of K clusters, based on their proximity to cluster centers. Each of the approaches to clustering implies objective functions on what constitutes a good clustering and an optimization procedure to improve that objective function.

Advocates of each type of clustering procedure highlight the relative advantage of the particular approach to clustering. For example, advocates for model-based clustering procedures "can provide a principled statistical approach to the practical questions that arise in applying clustering methods" (Fraley and Raftery, 2002, 611). and this is because "The problems of determining the number of clusters and of choosing an appropriate clustering method can be recast as statistical model choice problems, and models that differ in numbers of components and/or in component distributions can be compared" (Fraley and Raftery, 2002, 611). In contrast, algorithmic papers often tout their ability to solve difficult problems and prove theorems that demonstrate the conditions where the clustering algorithm will perform optimally (Fraley and Raftery, 2002).

We view the model and algorithmic distinctions as overwrought (Breiman, 2001). There are often close connections between modeling and algorithmic approaches to clustering. For example, K-Means can be thought of as a limiting version of a mixture of multivariate

normal distributions. Further, we show below that statistical and algorithmic methods often yield similar results when applied to the same data set. And many of the properties that are touted as advantages of statistical models have unclear application when applied to clustering methods. For example, it is unclear how to think about an outlier, rather than evidence that there is an insufficient number of clusters in the data set. Algorithmic approaches often have very clear theorems, but the assumptions necessary for the theorems to hold often require assumptions about ideal states of the world and it is hard to know how the algorithm performs as the assumptions are violated.

In short, as Breiman (2001) argues, there is a great deal to learn from both types of models. And we can learn a lot about the underlying content of documents by applying many different models to our data.

Soft vs Hard A separate dimension along with clustering algorithms differ is whether they provide a soft or hard partition of the data. A hard partition of the data assigns each document to one and only one cluster. Soft clustering, or fuzzy clustering, assigns a proportion of the document to a particular cluster. Algorithmic approaches tend to use hard clustering, though some use soft clustering. Model based methods are generally based on soft clustering, though they can be forced to make hard clustering decisions. In general, the difference between soft and hard clustering is not major. This is because almost all clustering methods suppose that every document truly belongs to one clustering, so the fuzzy methods tend to place most of a document into one cluster.

Means vs Mediods The way the center of a cluster is defined constitutes yet another difference across clustering methods. In some methods, like K-Means and mixture models, the cluster center is an average of the documents assigned to the cluster. For example, in a mixture of von-Mises Fisher distributions the cluster center is the weighted-average of the normalized documents, where the weights are determined by the probability of a document

belonging to a particular cluster. Similarly, in K-Means we saw that the center of the documents are the averages of documents assigned to the cluster. In contrast, in mediod methods the center of the cluster is constrained to be a document to assigned to the cluster. In K-Mediods the estimation procedure is similar to K-Means, but the cluster center is set as the document that minimizes the distance of documents assigned to the cluster. Similarly, in affinity propagation the cluster centers are specific documents.

The primary contrast between mean and mediod models, then, is in what constitutes an exemplar document. In mean methods the exemplar is an average of documents. This enables a much larger set of potential exemplars than in mediod methods, but has the disadvantage that there is no one document that can be read as representative of the cluster center (though it is always possible to select a document at the center of the cluster). In mediod methods the exemplar is a specific document. This makes it easier to read a representative document, but constrains the potential set of exemplars.

Flat vs Hierarchical Clustering methods also differ in whether they are flat or hierarchical. Flat clustering methods are the methods that we have considered so far—they produce a single clustering of the data, often times after conditioning on a specific number of clusters. Hierarchical clustering methods provide a nesting of observations. At the top of the hierarchy all the methods are grouped together, at the bottom the observations are in their own clusters. In between, hierarchical methods nest observations to create increasingly coarse clusters as we move up the tree.

While the two approaches to clustering may seem very different, they are actually quite similar. Every hierarchical method can be converted to a flat clustering by cutting the tree at a particular level. And every flat clustering method can be reestimated varying the number of clusters included in the clustering. And to make a direct hierarchy, flat clustering methods can be applied sequentially to particular clusters from a clustering. Both types of clustering

methods require similar assumptions that we have described before and require the careful analysis to determine the content.

Comparing Clustering Methods We have highlighted several prominently described features of clustering methods and explained why these divisions may not manifest in the actual partitions that the method produces. As we explain below, we recommend comparing clustering methods based on the partitions that are produced. And to compare partitions, we make comparisons based on the pairs of documents that are grouped together. To do this, we use a confusion matrix that enables us to compare the documents that are grouped together.

2.7 Comparing and Contrasting Different Clustering Methods

We now present the results from applying different clustering methods to the same data set: KMeans, affinity propagation, a mixture of von Mises-Fisher distributions, and spectral clustering applied to the data set of `ArXiv` papers that mention the word *cluster*. We summarize the results in two ways. First, we use a series of confusion matrices to compare the clusterings across methods. Second, we summarize the key words for each category using a simple method that we describe below. We have previously applied K-Means to the data above, so we turn to the two models that remain: Affinity Propagation and a Mixture of von Mises-Fisher distributions.

2.7.1 Affinity Propagation

Affinity propagation is an algorithmic, flat, mediod based clustering method that determines the number of clusters indirectly, rather than explicitly, the number of clusters Dueck and Frey (2007). To do this, the model takes an arbitrary *similarity* matrix: a matrix that measures the affinity between observations, similar to those we constructed in Chapter 3.

Table 4: Applying Affinity Propagation to the *ArXiv* Summaries

Exemplar	Words	Proportion of Documents
"On rooted cluster morphisms and cluster structures in 2 Calabi Yau triangulated categories"	cluster, algebras,algebra, cluster_algebras, cluster_algebra	0.20
"Sparse Convex Clustering"	clustering, algorithm,algorithms, data,cluster_algorithm	0.12
"LoCuSS Luminous infrared galaxies in the merging cluster Abell 1758 at z 0 28"	galaxies, galaxy,cluster,X0,ray	0.37
"In search of massive single population Globular Clusters"	clusters, globular_clusters, globular,star_clusters	0.31

The diagonal elements of the similarity matrix determines the likelihood that a particular document is selected as an exemplar. A "good" clustering according to this model is a clustering that groups together similar documents, with exemplars close to the center of each cluster. The model then optimizes cluster assignment by iteratively determining the fitness of a document as an exemplar and assessing which documents are likely to serve as an exemplar for other models. This message passing algorithm converges, providing a partition of the documents.

We use the `apcluster` package in `R` to fit Affinity Propagation to the `ArXiv` cluster data. We use cosine similarity to measure similarity between documents and set the self-similarity to -40. This results in 4 clusters. We summarize the output in Table 4 which shows a striking similarity between the clustering from Affinity Propagation and the clustering from K-Means. This similarity is particularly clear in Table 5, which shows the confusion between Affinity Propagation and K-Means. While there is substantial overlap, we see that some of the clusters are split, indicative of Affinity Propogation offering a slightly different partition of the texts.

Table 5: Confusion Table Comparing Affinity Propagation to K-Means

		Affinity Propagation			
		1	2	3	4
K-Means	1	6	8	217	2095
	2	72	75	2795	937
	3	6	1139	9	22
	4	316	0	0	0
	5	1645	17	630	11

2.8 Mixture of von Mises Fisher distributions

A mixture of von Mises-Fisher distribution is a statistical, flat, mean based clustering method, and the user explicitly sets the number of clusters. This clustering method assumes a process that generates the data (Banerjee et al., 2005). This model enables us to identify the key properties of the clustering method.

To apply a mixture of von Mises-Fisher distributions we first normalize the data such that $\boldsymbol{w}_i = \frac{\boldsymbol{w}_i}{||\boldsymbol{w}_i||}$. We suppose that each observation is assigned to a single cluster, $\pi_i \sim$ Multinomial($\boldsymbol{\pi}$) where $\boldsymbol{\pi}$ is a $K-$ element long vector that describes the population proportion of observations in each category, $\boldsymbol{\pi} = (\pi_1, \pi_2, \ldots, \pi_K)$. We suppose that the normalized texts are drawn from a category specific von Mises-Fisher distribution $\boldsymbol{w}_i|\pi_i = k \sim$ von Mises-Fisher($\phi, \boldsymbol{\mu}_k$). The von Mises-Fisher distribution is analogous to the normal distribution, but on a J-dimensional hypersphere. $\boldsymbol{\mu}_k$ is the center of the distribution and ϕ is analogous to precision of a distribution. The pdf for the von Mises-Fisher distribution $p(\boldsymbol{w}_i|\boldsymbol{\mu}_k, \phi) = c_J(\phi)\exp(\phi\boldsymbol{\mu}_k^{'}\boldsymbol{w}_i)$, where $c_J(\phi)$ is a function that normalizes the pdf to integrate to 1.

This data generating process provides the three key pieces of information to characterize the clustering method. The von Mises Fisher distribution measures the cosine of the angle between the document and the cluster center. The complete data generating process defines a likelihood model that provides the objective function. And there are numerous methods

Table 6: Applying Mixtures of von Mises Fisher Distributions to *ArXiv* Summaries

Cluster Label	Words	Proportion of Documents
	clusters,globular, globular_clusters,star,star_clusters	0.33
	X0,galaxies,X1, redshift, ray	0.16
	clustering, data,algorithm, based,algorithms	0.12
	mass,star,stars,stellar,cluster_mass	0.11
	cluster,algebras,algebra,cluster_algebras,quantum	0.28

Table 7: Confusion Matrix Comparing K-Means to Mixture of Von Mises-Fisher Distributions

		Mixture of vMF's				
		1	2	3	4	5
	1	2244	14	19	45	4
	2	1040	1510	42	912	375
K-Means	3	0	1	1174	0	1
	4	0	0	0	0	316
	5	14	92	14	113	2070

for optimizing the likelihood—with the EM-algorithm among the most widely used methods (Dempster, Laird and Rubin, 1977).

We implemented a model to cluster documents using a mixture of von Mises Fisher distributions in R. Table 6 describes 5 clusters from the model, while Table 7 compares the Mixture of von Mises-Fisher distributions to the clustering from K-Means.

2.9 Spectral Clustering

Spectral clustering is an algorithmic, flat, mean based clustering method where the user explicitly sets the number of clusters (Ng, Jordan and Weiss, 2002). Spectral clustering methods are based on intuition from graph-cutting algorithms, where a lower-dimensional representation of similarity is used to accentuate "natural" clusters within the data. We first suppose that we have some a similarity matrix between the documents. Then, a low-dimensional approximation of a transformed version of the matrix is made. And then K-Means is applied to this lower-dimensional approximation.

Table 8: Applying Spectral Clustering to *ArXiv* Summaries

Words	Proportion of Documents
bh,massive_star,mass_stars,low_mass,km_1	0.0016
clustering,algorithm,approach,problem,proposed	0.69
clusters,mass,0,cluster,1	0.28
accuracy,distributed,subspace_clustering,sensor,subspace	0.0006
star,star_clusters,mass,star_formation,clusters	0.0215

Table 9: Confusion Matrix Comparing K-Means to Spectral Clustering

		Spectral Clustering				
		1	2	3	4	5
	1	1	1589	670	0	66
	2	15	1953	1775	1	135
K-Means	3	0	1174	0	2	0
	4	0	313	3	0	0
	5	0	1886	400	3	14

To apply spectral methods we use the `R` package `kernlab`. We use the normal kernel to measure document similarity. The results are placed in Table 8 and the comparison to K-Means is found in Table 9.

2.10 Computer-Assisted Clustering

Applying different clustering methods to the *ArXiv* data set shows how different clustering methods can produce similar, though distinct, clustering methods when applied to the same data. Of course, we have barely scratched the surface of potential clustering methods. Other methods might specify different components of a mixture, implicitly changing the definition of similarity between the documents. Or, the different methods might use different optimization procedures to find a clustering, or even use different objective functions for defining a "good" clustering.

The potential set of models are numerous and growing quickly. Each of the new clustering methods are carefully derived, based on a clear set of assumptions, and rigorous derivations.

And often times the paper shows that the new clustering method is able "beat" existing methods at an important task, such as information retrieval or classification. This is also an active area of research with the number of clustering algorithms and their extensions growing rapidly.

The rigor in derivation and the growth of the field has not been met, however, with a critical examination of when to apply clustering algorithms and to what problems they should be applied to. In fact, there is little guidance from the literature on how to select a clustering method for a particular problem. Theoretical guidance based on theorems is particularly lacking. There are not any (to our knowledge) generally applicable theorems that demonstrate one particular clustering method is more effective than other clustering methods for discovering useful content. The literature also lacks papers that have a more modest goal: providing guidance on when to apply clustering methods based on the observable features of the data.

When considering the clustering literature, then, there is an implicit recognition that the right methods for a task will be difficult to identify before hand and the methods that one does end up using might be more about convenience than principle. This level of arbitrariness is a direct consequence of the vague goal of discovery–and the generally vague goal when using other unsupervised methods. The result is that we lack an easy to write down objective function.

To see why it is hard to write down the right objective function, consider the goal we started this chapter with: discovering some interesting organization of the texts. Certainly we know that something is interesting once we have seen it, but in general it is impossible to know if a clustering is interesting without human intervention. This makes automated search that excludes humans altogether impossible.

Given this limitation, Grimmer and King (2011) instead introduce a procedure that explicitly includes humans in the cluster selection process. Their procedure is based on the

insight that interesting clusterings are easy to spot once they have been spotted. With this in mind and assuming there were no cognitive or computational constraints, a reasonable approach to clustering would be enumerating all clusterings and then asking the user to find the most interesting clustering. Given this procedure is obviously impossible, Grimmer and King (2011) instead propose creating a geography of clusterings.

Their procedure contains the following six steps. First, Grimmer and King (2011) create a document term matrix of the text, potentially incorporating many ways texts could be pre-processed (Denny and Spirling, 2018). Second, they then apply as many clustering algorithms as available and many tuning parameters within those methods to generate a set of clusterings. Third, they use the distance metric from Meilă (2007) to create a distance matrix between the clusterings. Fourth, they project that distance matrix to two-dimensions using multidimensional scaling (see below). Fifth, they introduce a *local-cluster ensemble* to explore the two-dimensional project. A cluster ensemble aggregates different clustering methods to create a single clustering. A local-cluster ensemble uses different weights on the clusters to create an ensemble where clusters near a point in the two-dimensional projected space receive more weight. And finally, they use animated visualizations and related technology to make the space easier to explore. The steps are collected below and available in the software *consilience* at *consilience.com*.

We summarize the steps from Grimmer and King (2011) here

Grimmer and King (2011) apply their procedure to discover a conceptualization about speech in the US Congress. Generally, there are tradeoffs when applying FAC and CAC methods. FAC methods are able to provide a single and clear clustering. Further, it is relatively easy to build more complicated models with interpretable parameters (which we will see in later chapters). Yet, FAC methods will necessarily limit the set of assumptions we consider when clustering the data. CAC, methods, in contrast, enable us to explore the assumptions of clustering methods more completely than anyone FAC method could.

Table 10: Steps for Computer-Assisted Clustering Analysis

1) Create a dtm of the texts
2) Apply available clustering algorithms to the dtm, generating clusterings.
3) Calculate clustering level distance matrix, using the clustering distance metric from Meilă (2007)
4) Project the matrix to two-dimensions using multidimensional scaling (see below).
5) Use a local-cluster ensemble to average clusterings to create millions of new clusterings from the initial set of clusters.
6) Provide an animated visualization to facilitate exploration of the space.

Indeed, there will be many more clusterings considered. That said, using a CAC method can require a great deal of work from the analyst and it is impossible to automate. In the end, the choice comes down to both researcher preference and the type of basic problem the researcher is considering.

3 Latent Dirichlet Allocation and Vanilla Topic Models

Even though there are a wide array of clustering algorithms, at their core each share a common assumption that each documents is assigned to only one cluster. But documents may have more than one idea contained in them. For example, political speeches may cover a variety of key themes. Newspaper articles may cover a story from a variety of perspectives. And novels are often thought to contain several themes. When documents are a mixture of distinct ideas clustering methods will obscure this variation.

Topic models are a class of models that are closely related to clustering methods, but they make a fundamentally different assumption about the categories each document is assigned

to. Rather than assign each document to only one cluster, topic models assign each document to many categories. That is, topic models suppose that each document is a mixture across categories, which we will call a mixed membership model. As we will see, mixed membership models provide important insights often unavailable in clustering algorithms.

The first topic model is *Latent Dirichlet Allocation* (LDA) (Blei, Ng and Jordan, 2003).[5] Given the wide array of topic models that have emerged subsequently, we will call the original model *vanilla* LDA or a vanilla topic model.

LDA is a Bayesian hierarchical model that assumes a particular model of how an author generates a text. We first suppose that when writing a text the author draws a mixture of topics: a set of weights that will describe how prevalent the particular topics are. Given that set of weights, the author generates the actual text. For each word the author first draws the word's topic. Then, conditional on the topic, the actual word is drawn from a topic specific distribution. This topic-specific distribution is common across the categories and characterizes the rates words appear when discussing a particular topic.

Given this data generating process we can write down a specific statistical model for how the text are generated. For each document i ($i = 1, 2, \ldots, N$) we will suppose that we draw a $K \times 1$ vector of topic weights $\boldsymbol{\pi}_i = (\pi_{i1}, \pi_{i2}, \ldots, \pi_{iK})$. Suppose that we have the m^{th} word from a particular document ($m = 1, 2, \ldots, M_i$), which we will call w_{im}. We will suppose that each word has a corresponding topic indicator that is a draw from a Multinomial distribution $z_{im} \sim \text{Multinomial}(1, \boldsymbol{\pi_i})$. Then, conditional on $z_{imk} = 1$ we will assume that w_{im} is a draw from a multinomial distribution $w_{im}|z_{imk} = 1 \sim \text{Multinomial}(1, \boldsymbol{\mu}_k)$, where $\boldsymbol{\mu}_k = (\mu_{1k}, \mu_{2k}, \ldots, \mu_{Jk})$ is a $J \times 1$ vector where each μ_{jk} describes the probability of using

[5]Of course, there are many models that accomplish similar tasks to LDA that preceded it. The best example is Latent Semantic Indexing (LSI) which was essentially the application of Singular Value Decomposition (SVD) to a document term matrix. LSI was an important model and it remains used across several fields (Deerwester et al., 1990). The immediate predecessor to LDA was probabilistic Latent Semantic Indexing (Hofmann, 1999) which placed the method in a statistical modeling framework. That said, we focus on LDA because situating a similar model within a Bayesian framework has enabled extensions and modifications that would be difficult to situate in the original LSI and pLSI models.

word j when discussing topic k. We complete the data-generating process with priors, assuming that both $\boldsymbol{\pi}_i$ and $\boldsymbol{\mu}_k$ are drawn from Dirichlet distribution. While it is common to use a simple default prior for the model, other papers show that asymmetric priors can lead to more meaningful topic results (Mimno et al., 2011). The full posterior is described in Equation 3.1.

$$\begin{aligned} p(\boldsymbol{\pi}, \boldsymbol{\Theta}, \boldsymbol{T} | \boldsymbol{X}, \boldsymbol{\alpha}) &\propto p(\boldsymbol{\pi}|\boldsymbol{\alpha}) p(\boldsymbol{T}|\boldsymbol{\pi}) p(\boldsymbol{X}|\boldsymbol{\Theta}, \boldsymbol{T}) \\ &\propto \prod_{i=1}^{N} \left[p(\boldsymbol{\pi_i}|\boldsymbol{\alpha}) \prod_{m=1}^{M_i} p(\boldsymbol{\tau}_{im}|\boldsymbol{\pi}_i) p(x_{im}|\boldsymbol{\theta}_k, \tau_{imk} = 1) \right] \end{aligned} \tag{3.1}$$

For intuition about the sense in which LDA captures the topics in texts, consider a simple example about two different conversations. One conversation might involve US presidential politics. At the time of writing of this book, this might involve discussing "Donald Trump's statement" or "debate over the Russian election scandal". In contrast, we might have a conversation about formal models of international conflict. And there we might discuss "offensive-defensive balance" and "rationalist explanations for war".

The key is that there is one set of correlated vocabulary when we discuss the presidency and a second, relatively distinct, set of vocabulary when we discuss formal models of international conflict. We then learn a set of topics where one would assign relatively high probabilities to "presidential" words and a second topic that would allocate relatively high probabilities to "war" words. Of course, some words may receive a high weight in both topics. And both topics allocate some weight (often times a small weight) to all the words in the vocabulary.

LDA will work better, then, when there is a distinct vocabulary used when discussing different topics. The use of a mixture model can help uncover words that tend to occur together that otherwise might be difficult to uncover in a single membership model. This is

particular true if document are, in fact, discussing several topics.

For different intuition about why LDA can work for discovering an organization of texts, we can think about LDA as a model for compressing a document term matrix into a smaller set of topics (Reed, N.d.). To gain this intuition, we will focus on the representation of the prior in Equation 3.1. If we first focus on the component of the model for generating the text, $p(\boldsymbol{W}|\boldsymbol{\mu}, \boldsymbol{Z})$, we can note that we are going to try and find values of $\boldsymbol{\mu}$ that make the observed document term matrix more likely. This will be true when the components of $\boldsymbol{\mu}$ do a good job of approximating the original document term matrix. This will happen when there are groups of words that have a strong correlation with each other—or when a few topics can explain the variation in the original document term matrix.

There are several approaches to inference with a vanilla topic model, but there are two broad categories of estimation strategies: sampling based methods and variational approximations. One is a variety of Markov Chain, Monte Carlo (MCMC) methods, which include collapsed algorithms that marginalize over parameters that are often less of interest. This is the approach used for the popular `Mallet` software (McCallum, 2002). A variational approximation is a different approach, where the complex LDA posterior is approximated with a simpler distribution. This is the estimation strategy used in the structural topic model (STM) (Roberts et al., 2014). And an extension of this estimation strategy, online variational approximations, facilitates the application of topic models to extremely large collections of documents without requiring substantially more computational power (Vrehuuvrek and Sojka, 2011).

3.1 Example: Catalinac Work on Japanese Documents

[Summary of material to be added: Here we describe the data and Catalinac's approach in her book Catalinac (2016).]

3.2 Interpreting the Output of Topic Models

Similar to clustering methods, LDA is an extremely powerful tool for suggesting new organizations of documents. And just like clustering methods, LDA can be strongly dependent upon arbitrary tuning parameters. But this variation can be useful. Recall that when using LDA for discovery we are primarily interested in learning some new way of looking at our documents. Even if there is variability across of runs of the algorithm, so long as it provides a single useful way to look at the data the model has been useful.

In order to understand the organizations the model suggests, we can adopt methods we used to label and interpret the output of clustering models to label the topics and interpret their output. In the next chapter we describe several ways to compare the output of topic models and to assess their performance as measurement models. When making that assessment our goal is to assess their ability to credibly organize documents according to a particular organization. We can, however, make slight modifications to the procedures we used to validate the output of clustering methods to interpret the output of topic models. There are two primary methods that we recommend: careful reading of exemplar texts and quantitative procedures for identifying words that distinguish particular topics.

When labeling the output from clustering methods one approach we recommend is to closely read a random sample of texts assigned to a cluster. We make a similar recommendation with topic models, though we have to adopt a slight modification because documents have some "membership" in several categories. One approach is to select documents that have a large share assigned to a particular category. Specifically, we select the $M << N$ documents with the highest proportion of the document assigned to the particular category under consideration. We can then read those documents to assess their common facets and to interrogate whether a particular organization makes sense. We can also sample documents such that documents with a higher share allocated to a category are more likely to be selected. For example, we might set the probability of selecting anyone document as

$\tilde{\pi}_{i,k} = \frac{\pi_{i,k}}{\sum_{j=1}^{N} \pi_{j,k}}$. This has the advantage of insuring we select a variety of documents, but has the disadvantage of potentially selecting documents with little relationship to the category we are attempting to understand.

Just like with clustering methods, we can identify words that are indicative of a particular topic. The most straightforward method for obtaining these words is to select the top J words with the highest probability. While this is certainly useful, selecting the top words can obscure the distinctive features of a particular topic. This is because there may some words that have a high probability across topics because they are common. We explain in Section 5 below how to use methods for identifying separating words

Another similarity with clustering methods is that determining the number of topics to include in the model can be a vexing challenge. Similar to clustering methods there are numerous statistics that can be used and, as we discuss in the next chapter, there are a series of other more recently developed statistics that can be useful for determining the number of topics (Roberts, Stewart and Airoldi, 2016*a*). But it is impossible to determine the number of topics without knowing more about the specific application of the topic model in mind. This is because topic models of differing granularity can lead us to different sorts of insights. And, as we elaborate in the next chapter, there are numerous extensions of topic models that "nest" topics—facilitating the estimation of both granular and coarse topics for different levels of insights.

Labeling the Topics in Catalinac (2016) [Summary of material to be added: A description of the project and labeling topics in Catalinac (2016).]

3.3 Incorporating Structure into LDA: Structural Topic Models

Topic models build on a basic hierarchy within the data. At the top of the hierarchy there is a population level Dirichlet distribution that characterizes the distribution of topics for the

documents in the corpus. Each document's mixture of topics is drawn from this distribution. Then, conditional on the document's mixture of topics, topic labels for each word are drawn. These topic labels are used to determine which topic specific multinomial distribution is used to draw the actual word.

Vanilla LDA supposes that the documents are exchangeable in this hierarchy—that is, given the information available about the documents, we can permute their order and obtain the same posterior distribution. Of course, we might believe that there is additional information, document metadata, that explains differential prevalence (or absence) of topics across documents. For example, Grimmer (2013) argues that senators adopt a presentational style that informs the topics they emphasize in their press releases. To measure this presentational style, Grimmer (2013) includes information about a document's author in the topic model. In a different model, Quinn et al. (2010) examine the prevalence of topics in Senate debates. They argue that there is a temporal dependence across topics. They build a model that incorporates time, supposing that the topics in the US Senate evolve smoothly, with the prior day's topic informing the next day's topic distribution. Alternatively, we might suppose that characteristics about who is speaking could inform the way they discuss a topic. For example, in the United States we might expect that Republicans and Democrats would share vocabulary, but might discuss the same issue—such as tax policy—differently (Monroe, Colaresi and Quinn, 2008; Gentzkow, Shapiro and Taddy, 2016).

Incorporating the background information, then, implies that we alter the hierarchy, incorporating additional information to inform the type of distribution we draw the document topic distribution and the topics from. While numerous models exist that do specific versions of this—for example, Grimmer (2010) incorporates information about author and Quinn et al. (2010) and Blei and Lafferty (2006) incorporate information about time—Roberts, Stewart and Airoldi (2016*a*) provides a unified framework for incorporating an arbitrary set of covariates to explain both the prevalence of topics and how those topics are discussed.

To incorporate this information, Roberts, Stewart and Airoldi (2016*a*) alter the LDA data generating process, generalizing it to allow conditioning on covariates that explain the prevalence of topics and the way those topics are discussed. Their model is called the *Structural Topic Model* (STM), because it incorporates structural information about the document's covariates. To introduce the STM, suppose that we have a set of P covariates $\boldsymbol{X}_p$ that will be used to explain the topic prevalence and a separate set of $P^{'}$ covariates $\boldsymbol{X}_c$ that explains topical content. With this set of covariates we are able to specify the revised data generating process. To do this, Roberts, Stewart and Airoldi (2016*a*) build on the correlated topic model (Blei and Lafferty, 2007) and suppose the documents are generated according to the following hierarchical model:

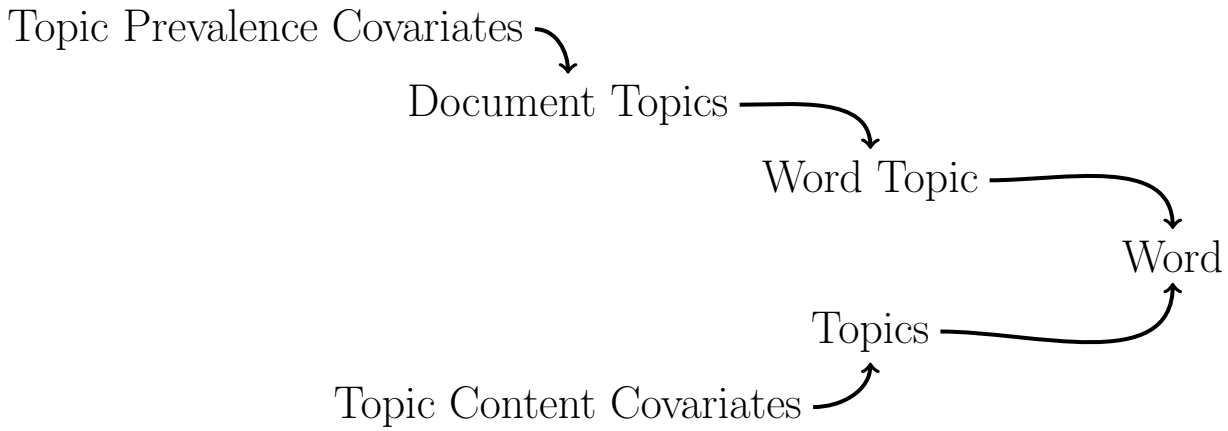
$$\begin{aligned}
\gamma_k &\sim \text{Normal}_P(0, \sigma^2 I_P) \\
\boldsymbol{\mu}_p &\sim \text{LogisticNormal}_{K-1}(\boldsymbol{\Gamma}^{'}\boldsymbol{x}_p^{'}, \boldsymbol{\Sigma}) \\
\boldsymbol{z}_{p,n} &\sim \text{Multinomial}_K(\boldsymbol{\mu}_p) \\
\boldsymbol{w}_{p,n} &\sim \text{Multinomial}_V(\boldsymbol{B}\boldsymbol{z}_{p,n}) \\
\beta_{p,k,v} &= \frac{\exp(m_v + \kappa^t_{k,v} + \kappa^c_{X_c,v} + \kappa^i_{X_c,k,v})}{\sum_v \exp(m_v + \kappa^t_{k,v} + \kappa^c_{X_c,v} + \kappa^i_{X_c,k,v})}
\end{aligned}$$

where $\boldsymbol{\Gamma}$ is a $P \times (K-1)$ is a matrix of coefficients for the topic prevalence model and $\{\kappa^t_{.,.}, \kappa^c_{.,.}, \kappa^i_{.,.}\}$ describe a set of rate deviations for the vocabulary and topics. The key idea is that each κ incorporates how the use of a feature varies across documents, how the use of a feature varies with the prevalence of a topic, and how the two interact for a particular topic. This provides a flexible method for measuring the presence/absence of a topic.

STM, then, enables researchers to encode the ways topic prevalence and content varies across documents based on the document's characteristics, or metadata. Figure 2 provides a heuristic view of the LDA hierarchy and how STM alters it. The reason Figure 2 is useful

is that it conveys exactly how STM alters the data generating process. The topic prevalance covariates inform our estimate of the prominence of topics in documents, while the topic content covariates alter how the topics are discussed.

Figure 2: The Hierarchy in LDA and STM



There are several reasons that including covariates can be helpful for the discovery process. Roberts, Stewart and Airoldi (2016*a*) show that including covariates helps STM identify more substantively interesting topics that are also more stable when there are few documents or a limited vocabulary. Figure 2 helps clarify a technical reason why STM can improve the topics that are discovered and the measured prevalence of those topics in documents. The prior on the document's topics enables the model to borrow information across documents to improve the estimation of topic prevalence. Without including covariates the model aggregates all the documents together into one prior, borrowing equal information across documents. Including covariates, however, enables the model to borrow information from documents with a similar topic distribution. Heuristically, this helps the model identify better topics, particularly when the included covariates explain linguistic differences.

STM conditions on a set of observed covariates in order to improve the estimated topic prevalence and the topical content. For example, Grimmer (2013) develops a model that assumes that there is a single, unobserved, covariate that partitions document distributions

into a set of categories. A different literature develops a model of nested topics, discovering covariates that group topics together based on their content. Called "Pachinko Allocation" after a popular Japanese arcade game (Li and McCallum, 2006), the model enables the estimation of a general set of coarse topics and then a more specific set of granular topics.

3.4 STM Example: CFPB and Topic Prevalence

[Summary of material to be added: We will include a short example applying the STM to data from the Consumer Financial Protection Bureau. This will include estimates of covariate relationships with topic prevalence.]

4 Low-Dimensional Embeddings

Discovery in text as data methods occurs as we use algorithmic or statistical models to distill the contents of texts and then use that distillation to learn about a way to organize documents. Thus far this organization has been in the form of groups: either clusters with documents assigned to only one or topics where documents are assigned to a mixture of them.

In this section we consider a low-dimensional representation of texts as a different way to distill and explore the contents of documents. By a low-dimensional representation we mean that we take the high-dimensional representation of each document as a $J \times 1$ vector and instead represent it with a $K \times 1$ vector where K is much smaller than J.

The goal when representing the texts using K rather than J dimensions is to focus attention on the salient underlying features that best explain the broad differences in the texts. Alternatively, we are looking for the K dimensions that best approximate the higher J dimensional space. Whether our goal is to capture the salient underlying features or to best approximate the higher dimensional document-term matrix, more details are necessary to

determine a specific "best" approximation. The different assumptions about what makes a "good" approximation and how the low-dimensional representation connects to the higher-dimensional data gives rise to a variety of distinct methods for finding low-dimensional representations.

In this section we preview several methods for obtaining low-dimensional representations of the texts, review methods for interpreting the output and labeling it, and emphasize the common components of the many methods. Like many of the other methods that we consider in this chapter, there is an inherent ambiguity in the application of the methods that comes from the goal being underspecified. We can write that our goal is to approximate the high-dimensional complexity of language with a low-dimensional representation that captures the salient features of language. But that sentence, while well formed, leaves ambiguous what it means to capture the content of language well and leaves ambiguous how we might adjudicate across low-dimensional representations.

As we maintain throughout the book, we will decide between representations based on the insights that they provide and the questions they lead us to ask. As a proximate measure of this we will discuss properties that we think embeddings will have that are useful. Of course, we will have no theorem to show that the properties that we posit actually correspond to the properties that lead to good discovery. For this reason, it will be hard to eliminate even seemingly simple models as not useful.

We begin with perhaps the most widely used method for generating low-dimensional approximations of high-dimensional observations: Principal Component Analysis (PCA).

4.1 Principal Component Analysis

The goal in Principal Component Analysis (PCA) is to discover a small set of underlying latent features—the principal components—that we will use to approximate the higher-dimensional data. We will continue to suppose that each document is represented as $J \times 1$

count vector, $\boldsymbol{w}_i = (w_{i1}, w_{i2}, \ldots, w_{iJ})$, but now we will suppose that our document term matrix has been centered: we subtract the average number of times each word appears in each column. Given this centered document-term matrix, we will attempt to approximate each document with a set of K principal components ($k = 1, \ldots, K$). We will call the $J \times 1$ vector $\boldsymbol{\mu}_k = (\mu_{k1}, \mu_{k2}, \ldots, \mu_{kJ})$ the k^{th} principal component and we will collect all the principal components into a $K \times J$ matrix $\boldsymbol{W}$, with $\boldsymbol{\mu}_k \in \Re^J$. Further, for each observation i we will suppose that there are K *loadings* on the principal components. We will call the $K \times 1$ vector of loadings for the i^{th} observation $\boldsymbol{\pi}_i = (\pi_{i1}, \pi_{i2}, \ldots, \pi_{iK})$. Using the loadings and the principal components, we will write each observation $\boldsymbol{w}_i$ as,

$$\boldsymbol{w}_i = \underbrace{\pi_{i1}\boldsymbol{\mu}_1 + \pi_{i2}\boldsymbol{\mu}_2 + \ldots + \pi_{iK}\boldsymbol{\mu}_K}_{\tilde{\boldsymbol{w}}_i} + \overbrace{\boldsymbol{\epsilon}_i}^{\text{error}} \tag{4.1}$$

where the $J \times 1$ vector $\boldsymbol{\epsilon}_i = (\epsilon_{i1}, \epsilon_{i2}, \ldots, \epsilon_{iJ})$ is an error term and $\tilde{\boldsymbol{w}}_i$ is the approximation of $\boldsymbol{w}_i$. Notice that the loadings $\boldsymbol{\pi}$ stretch, shrink, or flip the principal components in order to approximate the particular observation. Because $K << J$, there will necessarily be some error in this approximation.

The goal in estimation is to choose $\boldsymbol{\pi}_i$ and $\boldsymbol{\mu}$ to minimize the magnitude of the error. That is, we will choose $\boldsymbol{\pi}_i$ and $\boldsymbol{\mu}$ to minimize:

$$\begin{aligned} f(\boldsymbol{\Pi}, \boldsymbol{\mu}) &= \frac{1}{N}\sum_{i=1}^{N} \epsilon_i^{'}\epsilon_i \\ &= \frac{1}{N}\sum_{i=1}^{N} (\boldsymbol{w}_i - \sum_{k=1}^{K} \pi_{ik}\boldsymbol{\mu}_k)^{'}(\boldsymbol{w}_i - \sum_{k=1}^{K} \pi_{ik}\boldsymbol{\mu}_k) \end{aligned} \tag{4.2}$$

The optimal solution to minimize the magnitude of the error in Equation 4.2 is to use the K eigenvectors associated with the largest K eigenvalues of $\boldsymbol{W}^{'}\boldsymbol{W}$ (the empirical variance

covariance matrix) as the K principal components. Further, the loading for the i^{th} observation on the k^{th} principal component is $\pi_{ik} = \boldsymbol{\mu}_k^{'}\boldsymbol{w}_i$. For intuition about why the eigenvectors associated with the largest eigenvalues are selected, consider the goal: to explain as much of the variation of the document-term matrix as possible using the small set of principal components. While we avoid appealing too much to linear algebra intuition, those who are familiar with diagonalization results might note that we can better approximate a matrix by first selecting the components of the approximation that are associated with the largest eigenvalues. Therefore, choosing the eigenvectors associated with the largest eigenvalues first enables us to approximate the variance-covariance matrix as well as possible. Alternatively, deriving principal components shows that the minimizing the magnitude of the error is equivalent to maximizing the variance of the loadings. This is done by choosing the eigenvectors with the largest eigenvalues.

Once we have estimated the principal components and how the documents load on each principal components, we have a distillation of the documents that minimizes the error in the approximation. For purposes of discovery, however, we still need to interpret the output: label the principal components, the loadings, and explain what underlying latent concepts the principal components capture. Just like cluster analysis methods we will use both automated and manual methods to understand the principal components and the loadings.

4.1.1 Automated Methods for Labeling Principal Components

The most direct way to interpret the low-dimensional representation from principal components is to examine the values in each $\boldsymbol{\mu}_k$ that are especially positive or negative. Specifically, we can choose say the ten words with the highest values in $\boldsymbol{\mu}_k$ and the ten words with the most negative values. These words are informative, because they tell us the words that, if present in a document, will lead it to have a particularly negative or positive loading on that principal component π_{ik}. This is because $\pi_{ik} = \boldsymbol{w}_i^{'}\boldsymbol{\mu}_k$, so the entries in $\boldsymbol{\mu}_k$ that are

particularly large will be particularly influential in determining where a document falls on the spectrum.

Beyond analyzing the principal components directly, we can attempt to predict where documents fall on a spectrum using other information. To do this, we can regress a document's loading against other meta-data for the document—including characteristics of the author or the document. This approach is particularly useful when using principal component analysis (and related methods) to measure the ideology of authors or their texts. For example, if we believe that a particular principal component measures ideology we might regress the loading of authors against a well-validated measure of ideology. In the US context this often involves regressing loadings from texts against DW-Nominate scores, the low-dimensional measures of ideological behavior in the US congress derived from roll call votes (Tausanovitch and Warshaw, 2017). We do encourage one point of caution: low correlations can emerge even if two latent concepts are related. This is because measurement error can be present, dampening the correlations.

4.1.2 Manual Methods for Labeling Principal Components

In addition to the quantitative approaches to labeling documents, we can use the output from principal component models to structure a close reading of the texts and then label the components. To do this, we recommend sampling documents at similar points in the spectrum. Specifically, we might read a sample of documents from the far ends of the spectrum in order to gain a sense of what those documents have in common. Alternatively, rather than deterministically select documents we can sample documents along the spectrum, weighting documents closer to the endpoints more to ensure that we can gain a sense of why documents are grouped together at particular locations.

Once documents have been read closely, we can then label the ends of the spectrum and get a sense of how the documents vary moving across the spectrum. Reading the documents

also provides us with important insights that can be used in refining the conceptualization that we learn from the data.

4.1.3 Principal Component Analysis of Senate Press Releases

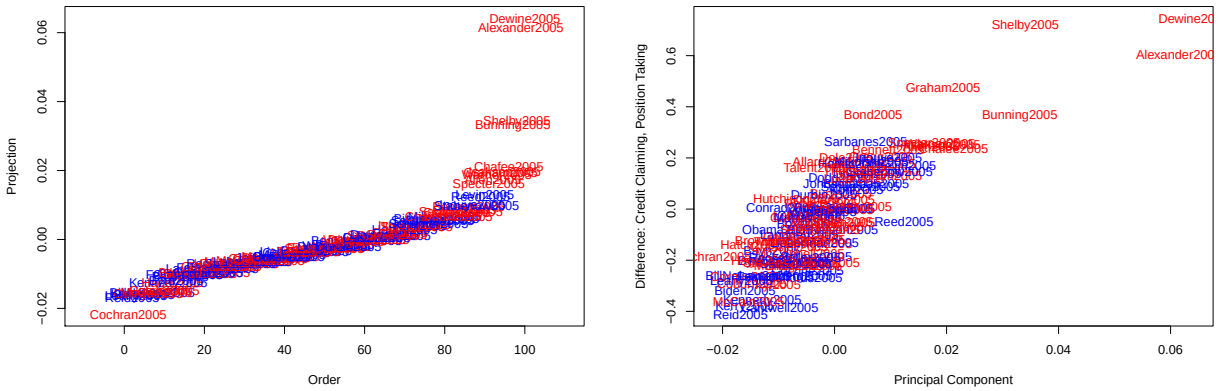
We apply PCA to a collection of Senate press releases from Grimmer (2010), covering all the Senate press releases issued in 2005. To analyze the collection of texts, we create a single count vector for all 100 senators who served in 2005. Call document $\boldsymbol{w}_{id}$ the 2796×1 count vector of terms for document d from senator i. We then normalize this count vector for each senator by the number of total terms uttered. We will call $\bar{w}_i = \sum_{d=1}^{D} \boldsymbol{w}_{id} \sum_{d=1}^{D} \sum_{j=1}^{J} w_{ijd}$ the normalized count vector.

We then use the `R` function `prcomp` to calculate the principal components and the loadings for each senator. Figure 3 provides two views of the principal components. The left-hand plot orders senators' loadings and colors the senators red if Republican and blue if Democratic. The left-hand plot shows that there does not appear to be a natural partisan ordering to the observations—in fact there is a great deal of overlap between the two parties. This does not, however, imply that the PCA was worthless. The right-hand plot shows that PCA captured a fundamental feature of legislators' communication strategy that Grimmer (2013) identifies: the trade off between position taking and credit claiming. So, the PCA captures systematic behavioral differences across legislators, but not in a way that might have been easy to anticipate before hand.

4.1.4 Choosing the Number of Principal Components

So far we have assumed that we know the number of principal components to include the model. But, of course, one of the most important modeling decisions is determining the number of principal components to include the model. And just like with clustering methods, we are unable to directly optimize the number of principal components using the model

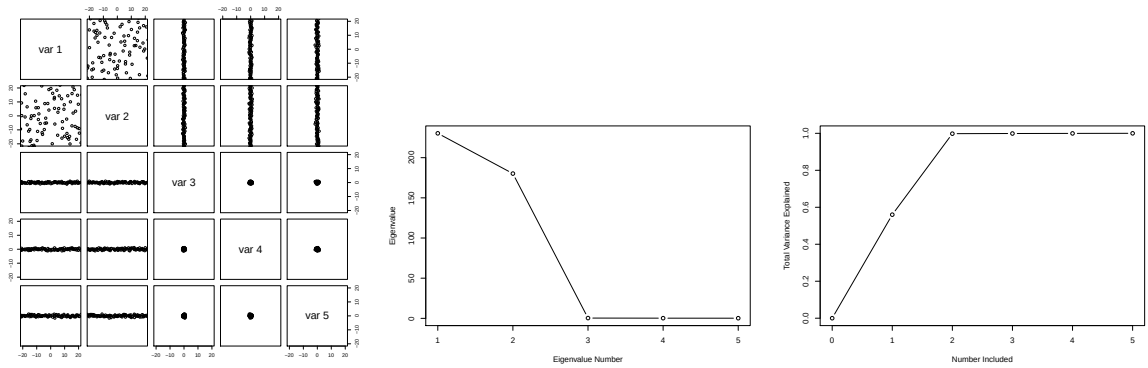
Figure 3: PCA Applied to the 2005 US Senate Press Releases



applied to the data. This is because principal component models are greedy: as more principal components are added to the approximation, the better the approximation becomes. Given this greedy behavior, evaluating the model using in-sample fit will yield a trivial solution—that we get the best in-sample approximation when we include as many principal components as features in the document-term matrix. This is not useful for discovery: we already know the number of features in our data.

Given this greedy property of PCA models means that we will have to use other approaches to select the number of components. We can use the properties of PCA models Numerically, the error that remains in the model is equal to the sum of the excluded eigenvalues. This implies that we can use the size of the eigenvalues to guide our model selection. To see why, consider Figure 4. In it, we present data that are ostensibly five-dimensional, but only two dimensions have variance, while the remaining three dimensions are merely small amounts of noise. We can see this in the left-hand plot with the dimensions where points are clumped together. The center-plot shows that applying PCA we get eigenvalues with two large initial values, while the remaining three are small. And finally, the right-hand shows the variance that each dimension explains. The first two included eigenvalues explain almost all of the variance, while the remaining three eigenvalues explain little, suggesting

Figure 4: Example of Using PCA to Make Model Determination



that our approximation is almost as good whether those dimensions are included or not.

Intuition from the right-hand plot in Figure 4 leads to the usual recommendation when selecting the number of components when performing a PCA: to look for the "elbow" in the right-hand plot of Figure 4. The elbow is the place where the percent explained variance bends, or where including more components does not lead to an increase in the explained variance.

Figure 4 is a stylized example and we caution that the strong conclusion that comes from the plot may not hold when deploying PCA to the example of interest. In actual applications the variance explained plot will almost never look so clear.[6] Therefore, the percent variance explained will be useful, but will not provide the clear guidance that this example suggests.

A different approach includes more dimensions so long as they provide new explanatory power. To label the dimensions we can examine a number of features, including how different documents load on different parts of the principal component. This more qualitative examination is also essential so we can develop substantive interpretations of the principal components.

The key when selecting the number of dimensions for a PCA is to remember the limitations of quantitative approaches to model selection. The quantitative measures of model

[6]The one noticeable exception is a PCA of roll call voting decisions in the US Congress

fit for PCA measure how much variance each additional dimension explains. It cannot tell you the "true" number of dimensions in the data, because the true number depends on your tolerance for error. In settings where simplicity is more important than extra error, we might be willing to use a smaller number of components. But, in other settings accuracy will be of paramount importance, so the number of components to include will need to be larger. And in still other settings we use PCA as an input to make predictions. In those cases we can use the clear objective function to determine the number of included components. Outside of the setting, though, the number of components to include will depend on the goals of our analysis.

4.2 Classic Multidimensional Scaling

Like PCA, the goal of Classic Multidimensional Scaling (MDS) is to find a low-dimensional approximation for a collection of documents. And we will see that with some specific assumptions, MDS will provide the same answer as PCA. To get this same representation, MDS focuses on preserving the distances between texts with a lower-dimensional representation.

To derive Classic MDS, suppose that we have a collection of N documents. Suppose further that we construct an $N \times N$ distance matrix $\boldsymbol{D}$ where entry d_{ij} represents the Euclidean distance between documents i and j (which we define in Chapter 3). The goal of MDS is to find a set of $N \times K$ vectors, $\boldsymbol{\pi}$ that approximate the distances as close as possible. That is, we look for a lower-dimensional representation of document i, $\boldsymbol{\pi}_i$, and document j, $\boldsymbol{\pi}_j$ such that we approximate the distances between i and j as closely as possible.

To make this approximation Classic MDS searches for $\boldsymbol{\pi}$ to minimize the following objective function, which compares the distance between the observations in the distance matrix d_{ij} to the distance between observations in the latent space:

$$f(\boldsymbol{D}, \boldsymbol{\pi}) \quad = \quad \sum_{i=1}^{N} \sum_{j<i} (d_{ij} - \sum_{k=1}^{K} (\pi_{ik} - \pi_{jk})^2)$$

Like PCA, optimization of MDS reduces to approximating the distance matrix with eigenvectors. To do this, we center the distance matrix by subtracting off the row means and the column means and then divide by -2. This provides the $N \times N$ matrix $\boldsymbol{W}\boldsymbol{W}'$. We then approximate this matrix with the first K eigenvectors. The embedding for each observation, $\pi_i = \left(\sqrt{\lambda_1}\pi_{i1}, \sqrt{\lambda_2}\pi_{i2}, \ldots, \sqrt{\lambda_k}\pi_{ik}\right)$ where λ_k refers to the eigenvalue of the k^{th} eigenvector. See Hastie, Tibshirani and Friedman (2001) for a full derivation.

The low-dimensional embeddings from Classic MDS are useful for discovering underlying structure in a document collection. When applying MDS, however, it useful to remember that the low-dimensional structure is strongly dependent upon the distance metric used. This is not surprising—after all, MDS is attempting to approximate the distances with fewer variables. But it is important to remember that altering the definition of close using different distance metrics will cause different structure to be found. Further, methods like MDS and PCA are only going to preserve the relative position of documents. The derivation of PCA makes this clear, because it preserves the correlation structure in the data. And MDS only preserves the distance between objects. This implies that methods like PCA and MDS are only able to provide relative information about locations in low-dimensional space.

4.3 Extensions of Classic MDS

Classic MDS is a powerful method that is useful for identifying the low-dimensional structure for a data set. Classic MDS' objective function assesses the quality of any low-dimensional representation of the text by comparing the distances between observations in the low-dimensional space to the distances between documents in the original distance matrix. This

comparison is reasonable, but it prioritizes embeddings that do a better job of approximating larger distances between documents. This could be an important goal, however, it is often the case that researchers are most concerned with accurately representing the documents in a neighborhood around a particular objection. And more error might be tolerated as documents are further away from each other.

There are several modifications to the classic MDS objective function that ensure that a scaling of the documents prioritizes the closest distances. The most straightforward is Sammon MDS. Sammon MDS normalizes the classic MDS distance function by normalizing by the distance between two documents. That is, the Sammon MDS objective function is:

$$f(\boldsymbol{D}, \boldsymbol{\pi}) \quad = \quad \sum_{i=1}^{N} \sum_{j<i} \frac{(d_{ij} - \sum_{k=1}^{K} (\pi_{ik} - \pi_{jk})^2)}{d_{ij}}.$$

When two documents are close, or d_{ij} is small, differences between the distance matrix distance and the embedded distance will be magnified. But for documents at a larger distance discrepancies between d_{ij} and the embedded distance will matter less. Altering the objective function does increase the difficulty of optimization. Rather than straightforward application of linear algebra, obtaining the embedding for a Sammon MDS requires a gradient descent algorithm.

There are several other modifications we could make to scale our observations. For example, we could use landmark MDS to embed a set of representative points and then embed the remaining documents around those points (Platt, 2005). Other methods seek to a manifold—or more complicated geometric structure that is locally euclidean in order to uncover a potentially complicated latent structure (Roweis and Saul, 2000).

4.4 Applying Classic MDS to Senate Press Releases

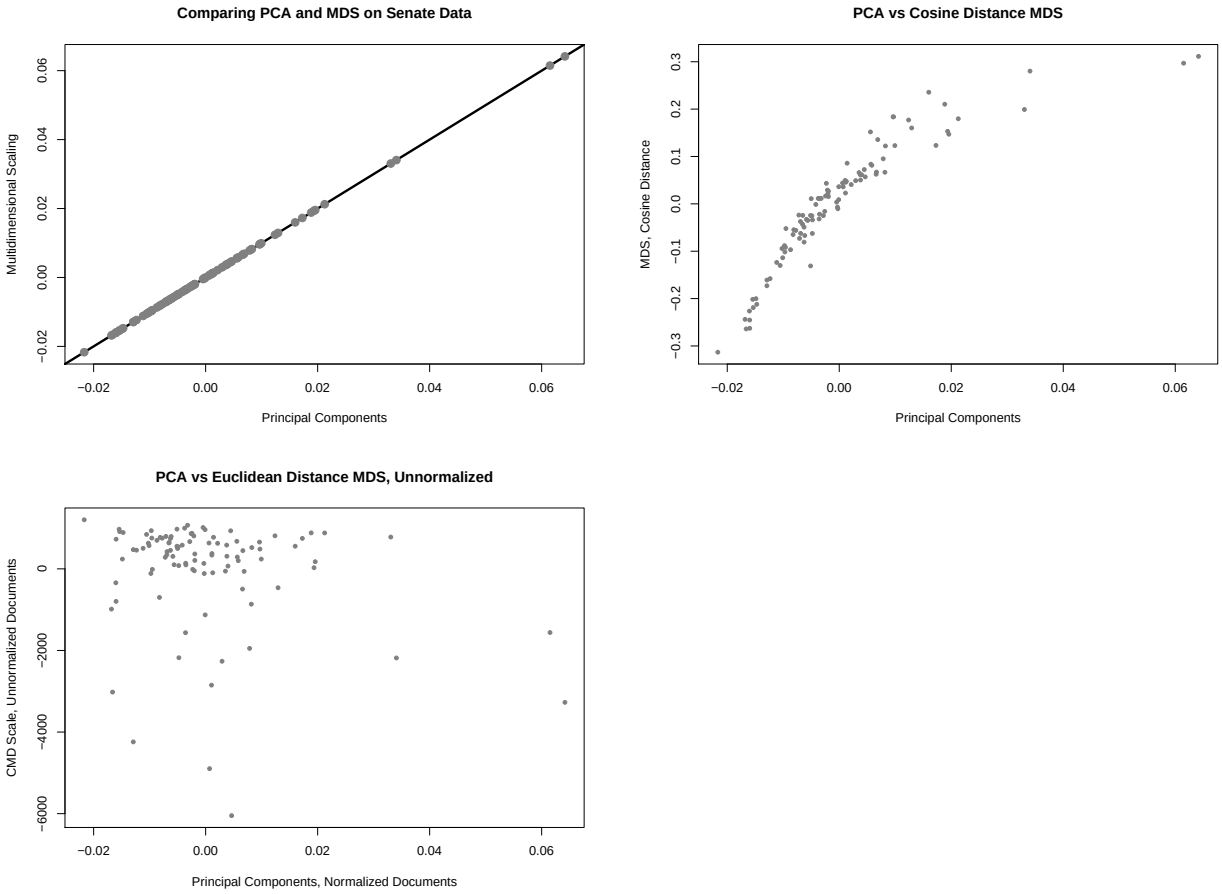
For the purposes of discovery MDS and PCA have the same goal: finding a latent representation of the texts that provides insights into the documents. In Section 4.1.4 we saw how applying PCA to the collection of Senate press releases reveals how senators trade off between claiming credit for money in their district and articulating positions on salient issues.

The left-hand plot in Figure 5 compares the embeddings from MDS, where the distance between senators' aggregated press releases is calculated using Euclidean Distance and PCA. This plot shows that the embeddings are exactly equal: measuring distance between documents using Euclidean distance and then applying Classic MDS is equivalent to applying PCA to the document collection. The next two plots show how different distance metrics and representations of the texts lead to different embeddings. In the middle plot where we measure distance as 1 - cosine similarity between the documents. This shows that Classic MDS provides a different, though still similar, embedding of the observations. And finally the right hand plot shows that if we do not normalize the aggregated count vectors we end up with a vastly different embedding.

4.5 WordFish, Factor Analysis, and Intro to IRT

PCA and MDS place documents into a low-dimensional space. These low-dimensional representations are useful on their own as a way to learn about the primary variation in texts. But the low-dimensional representations are also useful because they can correspond to political economy models of conflict in elections and political institutions (Poole and Rosenthal, 1991). As we mention in Section 1 above, a large literature assumes that actors and policy proposals are situated in policy space. And a large literature has sought to use votes in political institutions to measure where actors fall in the space and to test theories of how political conflict occurs in institutions (Poole and Rosenthal, 1991, 1997; Clinton, Jackman

Figure 5: Comparing MDS and PCA



and Rivers, 2004). There is perhaps no more successful measure in political science than DW-Nominate scores, providing a basic measure that has been used across thousands of papers. But other measures use responses to surveys (Bafumi and Herron, 2010), donations (Bonica, 2013), and even social media behavior (Barberá, 2014; Bond and Messing, 2015).

Each of the measures of ideological location are useful, but have important limitations. Roll call votes are regularly used to scale legislators (Poole and Rosenthal, 1997; Clinton, Jackman and Rivers, 2004), but outside of the US Congress roll call votes are less reliable (Spirling and McLean, 2007). And other political actors–presidents, bureaucrats, and political candidates– do not cast votes. Other methods for scaling political actors have been developed (for example Gerber and Lewis 2004; Bonica 2013), but they rely on particular

disclosure institutions that are often absent in other democracies.

But nearly all political actors speak. A method that could use this text to place actors in a political space would facilitate testing some of the most important theories of politics. We describe two methods for scaling political actors using texts. One method, based on Laver, Benoit and Garry (2003), situates actors in a political space based on their language and a set of exemplar texts that define the space. A second method extends a model commonly used to estimate location from roll call voting data to text, using the assumption that an individual's ideological location affects the rate they use particular words (Monroe and Maeda, 2004; Slapin and Proksch, 2008).

In this section we describe a set of methods designed to locate actors into an underlying space. The methods are useful because they can provide some intuition about where individuals fall in an ideological space and can drive insights into how conflict is structured in American politics. The methods have two broad types. One method uses a set of actors to define an ideological space and then scales the remaining actors in that space. A second method assumes that the rate actors use words is associated with their position in the ideological space. Both models make an explicit reference to ideological space. But, as we will see, there is no guarantee that the space discovered corresponds to an ideological space where policy conflict occurs.

The scaling literature holds great promise for testing spatial theories of politics. Recognizing this, several recent papers have offered important technical contributions that improve the methods used to perform the scalings (Lowe, 2008; Martin and Vanberg, 2007; Lowe et al., 2011). These papers are important, but we think that the scaling literature would benefit from a clearer articulation of its goals. Recent papers have implicitly equated the goal of scaling methods as replicating expert opinion (Benoit, Laver and Mikhaylov, 2009; Mikhaylov, Laver and Benoit, 2010) or well validated scalings made using non-text data (Beauchamp, 2011). Certainly plausibility of measures is important, but if the goal is to

replicate expert opinion, or already existent scalings, then text methods are unnecessary. Simple extrapolation from the experts or existing scaling would suffice.

Improving the validation of scales will help improve current models, which rely on the strong assumption of *ideological dominance* in speech. Both supervised and unsupervised scaling methods rely on the strong assumption that actors' ideological leanings determine what is discussed in texts. This assumption is often useful. For example, Beauchamp (2011) shows that this works well in Senate floor speeches and we replicate an example from Slapin and Proksch (2008) that shows that the model works well with German political platforms. But in other political speech, this may not be true–we show below that the ideological dominance assumption appears to not hold in Senate press releases, where senators regularly engage in non-ideological credit claiming.

Scaling methods will have more even performance across texts if they are accompanied with methods that separate ideological and non-ideological statements. Some of this separation is now done manually. For example, it is recommended in Slapin and Proksch (2008). But more nuanced methods are essential for the scaling methods to effectively capture the underlying ideological spectrum.

4.5.1 WordScores and Ideological Space Via Examples

Laver, Benoit and Garry (2003) represents a true breakthrough in the use of text as data in political science. One of the first articles to use large scale text analysis in the social sciences, Laver, Benoit and Garry (2003) introduce a method for placing actors in an ideological space, using their written words. Rather than rely on difficult to replicate and hard to validate manual coding or dictionary methods, Laver, Benoit and Garry (2003) introduced a fully automated method for scaling political actors, *wordscores*.

Wordscores uses a set of exemplar texts to define an ideological space and then scores documents based on their distinguishing features. The first step is the selection of *reference*

texts that define the political positions in the space. In the simplest example, we may select two texts to define the liberal and conservative ends of the spectrum. If we wanted to scale US Senators based on their speeches, for example, we may define as a reference text all the speeches from a very liberal senator, like Ron Wyden (D-OR) or Barabara Boxer (D-CA), and a very conservative senator, like Tom Coburn (R-OK) or Jim DeMint (R-OK). The reference (training) texts are then used to generate a *score* for each word. The score measures the relative rate each word is used in the reference texts. This creates a measure of how well the word separates liberal and conservative members–one measure of whether a word is *liberal* or *conservative*. The word scores are then used to scale the remaining texts. Laver, Benoit and Garry (2003) calls these the *virgin* texts, but in supervised learning we would call these texts the *test set*. To scale the documents using the word scores, first Laver, Benoit and Garry (2003) calculate the relative rate words are used in each of the test documents. The position of the texts is then determined by taking the weighted average of the word scores of the words in a text, where the weights are given by the rate the words are used.

Wordscores is rich and generalizable to multiple dimensions and to include several reference texts. But facets of wordscores constrain the method and make it difficult to recommend for general use (see Lowe (2008) for an extended critique). By defining "liberal" and "conservative" using *only* reference texts, Laver, Benoit and Garry (2003) conflate ideological language with stylistic differences across authors and impose the ideological dominance assumption on the texts. The result is that every use of wordscores will depend strongly on the reference texts that are used, in part because of stylistic differences across authors and in part because the reference texts will discuss non-ideological content. Careful preprocessing of texts, to remove words that are likely only stylistic, can mitigate part of this problem. Beauchamp (2011) for instance shows that results are significantly improved by removing technical language which coincides more with party power than with ideology. But

no amount of preprocessing can completely eliminate it. The explicit adoption of a supervised learning approach might limit the influence of style substantially. Unfortunately, this also requires a substantial increase in effort and time, which makes it application unwieldy.

4.6 Scaling Via Differential Word Rates

Rather than rely on reference texts for scaling documents, unsupervised scaling methods estimate words that distinguish locations on a political spectrum. First Monroe and Maeda (2004), then later Slapin and Proksch (2008), introduce statistical models based on *item response theory* (IRT) to automatically estimate the spatial location of the parties. Politicians are assumed to reside in a low-dimensional political space, which is represented by the parameter θ_i for politician i. A politician's (or party's) position in this space is assumed to affect the rate words are used in texts. Using this assumption and text data, unsupervised scaling methods estimate the underlying positions of political actors.

Slapin and Proksch (2008) develop their method, *wordfish*, as a Poisson-IRT model. Specifically, Slapin and Proksch (2008) assume that each word j from individual i, W_{ij} is drawn from a Poisson distribution with rate λ_{ij}, $W_{ij} \sim \text{Poisson}(\lambda_{ij})$. λ_{ij} is modeled as a function of individual i's loquaciousness (α_i), the frequency word j is used (ψ_j), the extent that a word discriminates the underlying ideological space (μ_j) and the politician's underlying position (π_i),

$$\lambda_{ij} \quad = \quad \exp(\alpha_i + \psi_j + \mu_j \times \pi_i).$$

The next section applies this model to political texts from two different contexts–demonstrating conditions when the model is able to reliably retrieve underlying policy positions.

4.6.1 Party Positions in Political Speech

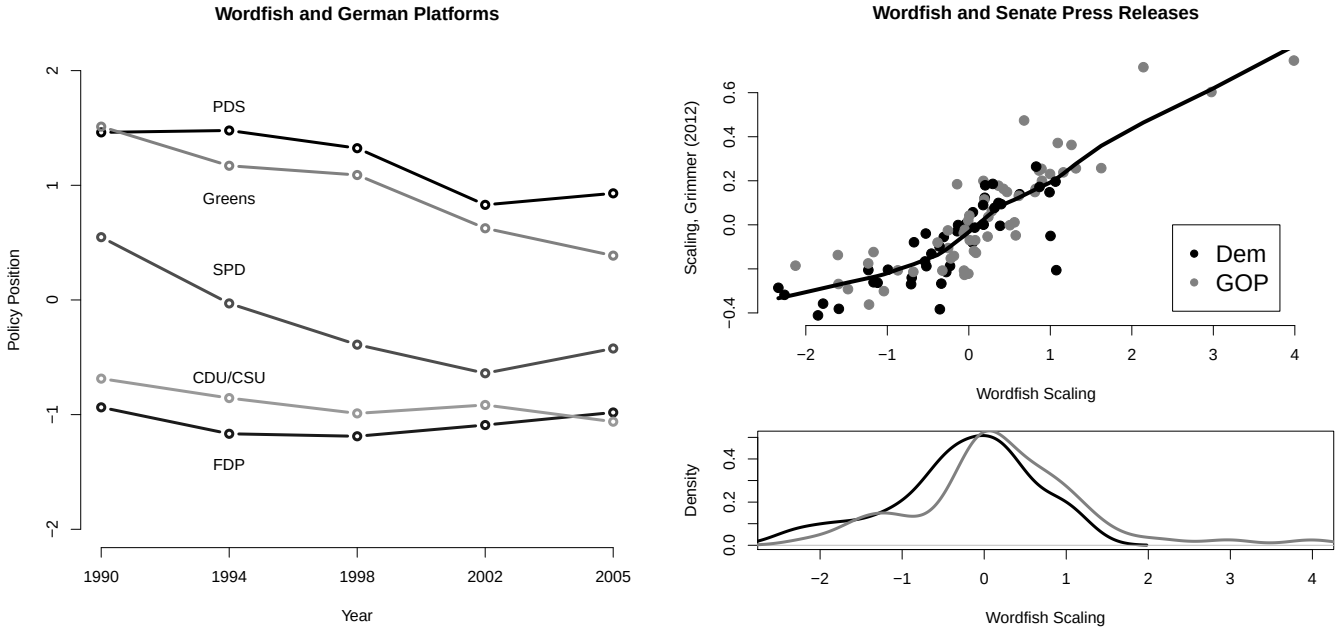
The ideological dominance assumption in language is essential for applying text analytic methods to measure the spatial locatio of political actors. When the assumption holds the model will reliably capture ideological differences in the use of language. But the models are not constructed to distinguish between differences that are ideological or other differences that merely describe the primary variation in the text. The non-ideological differences across actors may be about their focus on policy or pork (Grimmer, 2013), the style the essays were written, or the tone of the statements. Because the model does not include supervision explicitly, it is difficult to guarantee that the output of the model will reliably identify the revealed ideological locations of political actors. It is worth emphasizing that this *is not* a shortcoming of wordfish. In fact, we will show below that non-ideological locations that wordfish identifies are quite useful. But this does suggest that one should not assume that wordfish output measures an ideological location without careful validation.

When the ideological dominance assumption fits the data the model can reliably retrieve valid ideological dimensions form political texts. Take, for example, the left-hand plot of Figure 6. This replicates a plot in Figure 1 of Slapin and Proksch (2008), who apply the wordfish algorithm to German party platforms. As Slapin and Proksch (2008) show, the estimates in the left-hand plot separate the German parties and replicate expert assessments of German party movement over time.

But when ideological dominance assumption fails to fit the data, wordfish fails to retrieve underlying policy dimensions. The right-hand plot applies the wordfish algorithm to Senate press release data introduced in Grimmer (2010). The bottom, right-hand plot in Figure 6 is a density of the Democrats (black-line) and Republican (gray-lines) positions from the wordfish algorithm. The model clearly fails to separate Democrat and Republican senators–a necessity for any valid scaling of political ideology in the now polarized Senate.

The wordfish scaling is meaningful substantively, but it does not correspond to standard

Figure 6: Wordfish Algorithm: Performance Varies Across Context



policy space. The top plot shows that the wordfish algorithm reproduces the spectrum Grimmer (2013) identified using the expressed agenda model–how senators balance position taking and credit claiming in press releases. This plot presents the scaling from Grimmer (2013) against the scaling from wordfish on the horizontal axis and the black line is a lowess curve (Cleveland, 1979). The relationship between the two measures is extremely strong–correlating at 0.86. Clear evidence that wordfish has identified this interesting–though non-ideological–spectrum in the Senate press releases.

This exemplifies when scaling methods are likely to recover an ideological position. When political actors are engaging in heavily ideological speech–as in German party platforms–unsupervised methods appear able to retrieve reliable position estimates. But when political actors can avoid ideological speech–as in Senate press releases–scaling methods retrieve some other, non-ideological scaling. Therefore when applying scaling algorithms, careful validation is needed to confirm that the intended space has been identified. And an essential future area of future research will simultaneously isolate ideological statements and then employ

those ideological statements to scale political actors.

4.7 Sparse Factor Analysis

There are a multitude of methods for embedding observations beyond PCA, MDS, and methods intended to measure an actor's ideological location. One large class of methods, that we will discuss later in Chapter 6, are sparse factor analysis methods. Sparse factor analysis (sFA) methods are similar to other methods, with the key difference that observations do not "load" on each dimension. Like other embedding dimensions, these methods approximate higher-dimensional data using a lower-dimensional representation. Unlike those other methods, however, each observation is represented using only a subset of those underlying dimensions. It is also useful to introduce sFA because of its close relationship to the Indian Buffet Process (IBP), which is a workhorse machine learning tool for learning underlying structure in a data set (Ghahramani and Griffiths, 2006).

We will describe the sFA using a statistical data generating process, where we first suppose that we have normalized our documents—subtracting the column means and dividing by the column standard deviations. We will suppose that there are a set of K underlying dimensions with $\boldsymbol{\mu}_k \sim$ Multivariate Normal$(\mathbf{0}, \sigma_k^2 \boldsymbol{I}_J)$ and we collect the vectors into the $K \times J$ matrix $\boldsymbol{\mu}$. We suppose that the proportion of documents that load on a dimension are given by $\pi_k \sim \beta(\alpha_1, \alpha_2)$. We will call π_{ik} an indicator variable that is 1 when a document contains feature k and is 0 if feature k is not present. We assume that $\pi_{ik} \sim$ Bernoulli(π_k). We collect the indicators into the K-component long vector $\boldsymbol{\pi}_i$. Using the underlying features and the latent indicators, we suppose that the contents of a document arise from a combination of the underlying latent features $\boldsymbol{W}_i \sim$ Multivariate Normal$(\boldsymbol{\pi}_i \boldsymbol{\mu}, \sigma_n^2 \boldsymbol{I}_j)$. The $\boldsymbol{\pi}_i$ model enforce the sparse component of the model—it determines the features to contribute to the model. Estimation usually is done using EM, Gibbs sampling, or a variational approximation (Ghahramani and Griffiths, 2006).

The model can be extended in a variety of ways. For example, a poisson-gamma framework can be used to describe document generation, an Indian Buffet Process (IBF) can be used to describe the emergence of new factors, and in Chapter 6 we extend the Indian Buffet Process to include supervision for discovery and causal inference (Fong and Grimmer, 2016). Regardless of the model used, the key insight is that sparse models vary the embedding assumptions slightly, enabling the discovery of new underlying structure that can facilitate inferences.

5 Discriminating Words and Fictitious Prediction Problems

The discovery methods that we have considered so far partitions the documents into categories in order to highlight the salient differences across the documents or embed documents (and actors) into a low-dimensional space to draw attention to the facets of the documents that best distinguish across documents. The final discovery method that we discuss in this chapter is the discovery of discriminating words, or words that are associated with a particular category. The goal of this method is to identify a set of words that convey the distinct content of a particular category—such as an author's gender, partisanship, or ideology. As a necessary input, this discovery method supposes that there are a set of categories already in place. For example, a regular application of the methods is to understand Congressional rhetoric, where the label is whether the member of Congress is a Republican or Democrat (Monroe, Colaresi and Quinn, 2008; Gentzkow, Shapiro and Taddy, 2016). Then, the methods measure how well words distinguish between the two categories. Words that are measured to be particularly distinctive are then usually associated with that category and subsequently used as either an input to measurement (Gentzkow and Shapiro, 2010) or as a tool on their own to understand what sort of language makes the categories distinct (Monroe,

Colaresi and Quinn, 2008). This analysis has been applied across a variety of categories and texts to understand differences in ideology (Diermeier et al., 2011), gender (Cunha et al., 2014), and race.

Discriminating word methods are widely used, in part, because they have a strong intuitive appeal. Language that one group disproportionately uses is often thought to be indicative of the kind of arguments the group wants to make or characterizes the language that is distinct for that group. And the applications often validate this intuition, demonstrating that methods that uncover distinctive words do often provide valuable insight into the particular language of a group. Further, we will show in Chapter 5 that the methods we describe here are useful ways to create dictionaries for classification.

Even though there is a strong intuitive appeal, there are limitations to keep in mind when uncovering discriminating words. The challenges emerge from how the distinctive words are measured. To identify the discriminating words most methods setup a *fictitious prediction problem*. A word's distinctiveness is measured in the context of a prediction problem that is never actually of interest. For example, Gentzkow, Shapiro and Taddy (2016) identifies the words that best predict whether a member of Congress is a Republican or a Democrat. Obviously, predicting a member of Congress' political party is never of interest. Members of Congress clearly identify with a party. And while they might change sides or identify with another party, we never need to make a prediction about any serious candidate or legislator's partisanship.

Rather than building the model to perform some future prediction or classification, the fictitious prediction problem is used for analytic convenience. Entertaining the fiction that there is a prediction problem is useful, because it enables researchers to repurpose prediction methods to obtain a word-level measure of discrimination (see Chapters 5 and 7). That is, the model is constructed as if prediction is of interest, but then the discriminating capacity of the words is the output from the model that is used.

Recognizing that prediction is only fiction is important, because it shows that demonstrating that a model performs well at prediction is insufficient to show that the words provide an accurate distillation of the rhetoric of a particular group. In the next chapter we will show that procedures like cross validation are essential for assessing the performance of classification methods, because they provide an estimate of how the method performs on the key task. But cross validation is less useful here, because we might prefer models that are less predictive but do a "better" job of identifying words that are indicative of particular categories. Rather, the validations we use will necessarily be less direct and require subject matter expertise to assess the extent to which the measures of words help us to understand the key distinguishing features.

Fictitious prediction problems also highlight a potential risk when identifying discriminating words. We set up the problem to identify words that are associated with a particular category, but there might be other characteristics of the speaker correlated with the category. For example, Taddy (2013) analyzes Congressional speech to find words associated with Democrats and Republicans. But many Republicans represent states where there are large allocations of space to public lands and, therefore, models tend to identify words associated with national parks as Republican words, even though discussions about these issues are not particularly partisan.

The conflation of public land discussion and partisanship highlights a key conceptual issue. Words that distinguish categories might be distinct from the words that are indicative of a category. That is, words that are particularly partisan—indicative of how Republicans and Democrats talk when they convey their party's stance—might not overlap completely with the set of words that distinguish Republican and Democratic rhetoric. Taddy (2013) recommends including covariates, but this begs the question of what is an appropriate word and what is inappropriate.

Rather than include covariates, we recommend using this class of methods for discovery:

to learn about words associated with category and to use that information as inputs to a second round of analysis where a researcher explicitly defines the categories. Or, at least, has clear evidence about what the categories mean.

In addition to conceptual concerns, there is the statistical concern that we will overfit when discovering distinctive words. For intuition about how this overfitting occurs, suppose that we are using the fictitious prediction problem to identify words that predict whether the speaker is a Democrat or Republican. The overfitting occurs because in any data set any word that is spoken only once will perfectly classify the speaker as either a Republican or Democrat. Note, that this is true regardless of the partisan content of the word—this will happen merely because the word is rare. And this problem persists even if we eliminate words that are spoken only once. Whenever there are rarely used words, there is the possibility those words are used by just one group, solely by chance. If this happens, then we are no longer identifying words that are used systematically more often by one group, but rather whatever words just happen to align with a particular set of categories.

To avoid overfitting, the methods that we describe below usually regularize the estimates. Intuitively, we will shrink coefficients towards some common value. When words are used regularly they will be able to move away from the common mean if they truly are distinctive. But rare words used only occasionally will be shrunk to the average. As we will see, regularization is essential to identifying the words that are indicative of how a particular group uses language.

As a simple first example, we describe a simple procedure to identify distinctive words—test statistics that are commonly used in statistics. We then move to two statistical models for estimating word discrimination: Fightin' words and Mulitinomial Iverse Regression (MNIR) (Monroe, Colaresi and Quinn, 2008; Taddy, 2013). And finally, we show how to use a key concept from information theory, mutual information, to examine the distinctiveness of words (Shannon, 1949).

5.1 Standardized Test Statistics as Measures of Separation

We first set up our fictitious prediction problem and provide simple methods. We discussed this method above, but we present it again here to set up our notation and to provide an intuitive introduction to the statistical problem. Suppose that we have attribute Y_i, such as a politician's partisanship, or a measure of an author's ideology, gender, or race. And suppose that we our documents are in a document term matrix $\boldsymbol{W}$. For the simplest case, we will suppose that $Y_i \in \{0, 1\}$. For the simplest measure of how well words separate categories, we first estimate the following regression,

$$Y_i = \mu_0 + \mu_{1j} W_{ij} + \epsilon_i \tag{5.1}$$

Using the output from the regression, we compute the test statistic that corresponds to the null hypothesis that $\mu_{1j} = 0$ and then use this test statistic as our measure of separation:

$$\text{Standard}_j = \frac{\widehat{\mu_{1j}}}{\widehat{\text{SE}(\mu_{1j})}}$$

This measure prioritizes words that discriminate between the categories better, where the discrimination is scaled by the estimated standard error of the coefficient.

If Y_i is a continuous value then this same procedure can be used. If Y_i is categorical we will then estimate several measures of word separation. For each of the K potential categories, define $Y_{ik} = 1$ if $Y_i = k$ and $Y_{ik} = 0$ otherwise. We then update Equation 5.1, running a separate regression for each of the categories.

The more positive the test statistic the more discriminating a coefficient is, scaled by its variance. Cast in terms of the test statistic, under the null of a coefficient of zero, a larger test statistic corresponds to a lower p-value.

χ^2 test-statistics Gentzkow and Shapiro (2010) suggest a related procedure. Gentzkow and Shapiro (2010) are interested in identifying words that political parties in Congress use disproportionately. To construct this measure they use a χ^2 statistic from a χ^2 test of equal distribution across categories in a table. We will focus on the two category case, but the idea generalizes naturally to a χ^2 test. To construct the statistic, call $w_{j1} = \sum_{i=1}^{N} W_{ij} I(Y_i = 1)$ or the total number of times observations in group 1 uttered word j and $w_{-j1} = \sum_{m \neq j}^{J} \sum_{i=1}^{N} W_{im} I(Y_i = 1)$ or the total number of words not j uttered by individuals in group 1. We can then define w_{j0} and w_{-j0} analogously. The χ^2_j statistic is then defined as,

$$\chi^2_j = \frac{w_{j1}w_{-j1} - w_{j0}w_{-j0}}{(w_{j1} + w_{j0})(w_{j0} + w_{-j0})(w_{j1} + w_{-j1})(w_{-j1} + w_{-j0})}$$

Using this measure, we are able to identify words that are disproportionately used by individuals in group 1 or 2, consistent with a lower p-value under a null hypothesis of equal use across the categories.

With either standardized regression coefficients or χ^2 statistics, we might be concerned that the method could confuse a rarely used word that happens to separate groups with a more regularly used word that captures a systematic difference in language usage across the categories. We could address this issue manually, removing words that are used that fall below a threshold of use. This threshold, however, can be difficult to determine—after all, before hand it is difficult to judge the point where words are no longer randomly coinciding with a division. Further, manually removing words creates a substantial burden on the researcher, making it difficult to apply generally.

Rather than manually remove words we will focus on models that regularize the parameter estimates: shrink the estimates to a common value. As we will see, the amount of shrinkage

will depend on the information available in the data to form the estimate. Rare words that just happen to coincide with a particular category will be smoothed towards the common value. But words that are used regularly and distinguish the categories will only have a small amount of shrinkage. The general idea of regularization will appear regularly throughout the book. As a first introduction we turn to the Fightin' words model from Monroe, Colaresi and Quinn (2008).

5.2 Fightin' Words

Monroe, Colaresi and Quinn (2008) provide one of the first, and most widely used, solutions identify words that systematically separate categories, called Fightin' words. To address the problem of randomly occurring words, Monroe, Colaresi and Quinn (2008) use regularization—additional modeling assumptions to shrink coefficients to a common value and avoid overfitting. To develop the model, Monroe, Colaresi and Quinn (2008) measure the rate a particular group uses a word. Suppose, for example, that we are attempting to predict $Y_i \in \{0, 1, 2, \ldots, K\}$, such as whether the speaker belongs to one of several political parties. For each level k we can compute the probability a speaker who belongs to category k utters a particular word when speaking. Define the total number of words individuals from category k utter as $n_k = \sum_{j=1}^{J} \sum_{i=1}^{N} I(Y_i = k) W_{ij}$ and call $\boldsymbol{W}_k = \sum_{i=1}^{N} I(Y_i = k) \boldsymbol{W}_i$. We can then construct a regularized estimate of the probability an individual from category k (ie, has $Y_i = k$) utters word j with

$$\widehat{\mu}_{jk} = \frac{w_{jk} + \alpha_j}{n_k + \sum_{j=1}^{J} \alpha_j}$$

where α_j represents a small value used to *smooth* the estimates. We will call $\widehat{\boldsymbol{\mu}}_k = (\mu_{1k}, \mu_{2k}, \ldots, \mu_{Jk})$. While not necessary, we can motivate this estimate of the proportion using a conjugate Bayesian model. If we suppose that $\boldsymbol{\mu} \sim \text{Dirichlet}(\alpha)$ and that $\boldsymbol{w}_k \sim$

Multinomial$(n_k, \boldsymbol{\mu}_k)$. The estimate then corresponds to the expected value of the posterior distribution after observing the words.

Using this estimate of word-rate usage, Monroe, Colaresi and Quinn (2008) constructed a standardized log odds ratio to measure word separation. Specifically, they first calculate the log odds a particular word is used,

$$\text{log odds ratio}_{kj} \quad = \quad \log\left(\frac{\mu_{kj}}{1-\mu_{kj}}\right) - \log\left(\frac{\mu_{-k}}{1-\mu_{-kj}}\right)$$

Monroe, Colaresi and Quinn (2008) then standardize the log odds ratio. To standardize the ratio, they suggest using an approximation of the variance of the odds ratio, $\text{Var}(\text{log odds ratio})_{kj} \approx \frac{1}{w_{kj}+\alpha_j} + \frac{1}{w_{-kj}+\alpha_j}$. Using this variance, they calculate the standardized log odds ratio, which is the suggested measure of word separation

$$\text{Std. Log Odds}_j \quad = \quad \frac{\text{log odds ratio}_{kj}}{\sqrt{\text{Var}(\text{log odds ratio})_{kj}}}$$

As Monroe, Colaresi and Quinn (2008) show, this addresses several issues that occur without regularization. Yet, there are some challenges remaining. For example, Monroe, Colaresi and Quinn (2008) model is difficult to apply with a continuous category, such as ideology. To address this, we turn to a different model: multinomial inverse regression.

5.3 Multinomial Inverse Regression

In Section 5.1 we regressed the label, Y_i on the words $\boldsymbol{W}$. Of course, regressing on all J features simultaneously is problematic. One problem is that if $J > N$ then the regression coefficients are not identified. And even if we are able to compute the regression coefficients, including all J features will likely lead to noisy estimates that produce unhelpful comparisons

across categories. In Section 5.1 we avoided this by including each word one at a time in the regression. This ad hoc solution ensures we can estimate the regression coefficients, but it sets aside information about words that coocur in documents.

Taddy (2013) introduces multinomial inverse regression (MNIR) to perform a similar regularization task as Fightin' words, while ensuring that we are able to include continuous Y_i. As the name suggests, MNIR inverts the regression problem: rather than regressing Y_i on the features, it regresses the features on Y_i. Using this framework, Taddy (2013) introduces a multinomial regression model. That is,

$$\begin{aligned} \boldsymbol{W}_i &\sim \text{Multinomial}(n_i, \boldsymbol{\pi}_i) \\ \pi_{ij} &= \frac{\exp(\mu_{0j} + \mu_{1j} Y_i)}{\sum_{l=1}^{J} exp(\mu_{0l} + \mu_{1l} Y_i)} \end{aligned} \tag{5.2}$$

To address the problem of coincidental alignment, Taddy (2013) uses a Laplace prior– which provides regularization analogous to the regularization in LASSO regression (Hastie, Tibshirani and Friedman, 2001), which we discuss in Chapter 5. Using the MNIR, we can use the estimated $\widehat{\mu}_{1j}$ coefficients as measures of how well particular words separate the categories.

5.4 Mutual Information

As a final measure of word separation, we use a measure from information theory which assesses the *mutual information* between words and the category of a document Y_i. For this discussion we will assume that $Y_i \in \{0, 1, \ldots, K\}$. The mutual information between words and a value of Y_i is an information theoretic measure that describes how much knowing a particular word resolves our uncertainty about whether a document belongs to a particular category. To motivate the use of mutual information, suppose we were given the task of guessing a randomly selected document's value of Y_i. Mutual information measures how

much the presence or absence of a word informs our guess.

As a baseline level of uncertainty, we describe the entropy for a category. Suppose that π_k represents the proportion of documents that fall in category k—with $Y_i = k$—and let $\pi_{-k} = 1 - \pi_k$ or the probability a document does not belong to category k. π_k also represents the proportion of time we would be correct in guessing category k for a randomly selected document. Define the entropy for category k as $H(k) = -\pi_k \log_2 \pi_k - \pi_{-k} \log_2 \pi_{-k}$, which is a measure of our uncertainty about our guess. We might be able to make a better guess if we knew a word j was present in the document or not. Define $\pi_{k,j}$ as the proportion of documents that are both in category k and have word j and $\pi_{k|j}$ as the proportion of documents in category k given that word j is present. We can then define conditional entropy as:

$$H(k|j) \quad = \quad \sum_{k,-k} \sum_{j,-j} \pi_{k,j} \log \pi_{k|j}$$

where $\sum_{k,-k}$ indicates that we sum over documents in k and not in k. The conditional entropy describes how much our uncertainty about the label of a document decreases after we condition on a word. If a word is a perfect predictor of a cluster label, then the uncertainty will go to zero and if a word is orthogonal to a category than the measure will merely return the entropy. This property motivates the Mutual Information between category k and word j,

$$\mu_{k,j} \quad = \quad H(j) - H(k|j)$$

When a word is very predictive, the mutual information will be at a maximum and when

a word has no predictive power, it will be zero.

5.5 Example: What is a Republican word?

Word separating algorithms have been widely applied to measure the extent to which one American political party disproportionately uses a word. For example Monroe, Colaresi and Quinn (2008) examine differences in party rhetoric across topics of political debate, Gentzkow and Shapiro (2010) examine differences in rhetoric to measure partisan slant in newspapers, and Gentzkow, Shapiro and Taddy (2016) measure how words separate Republicans and Democrats in Congress to measure rhetorical polarization.

To motivate this application, we use a collection of House and Senate floor speeches from 2009-2016, constituting over 294,000 speeches. [Summary of material to be added: Example here using floor speeches.]

6 Texts and Discovery

In this chapter we have introduced four broad classes of methods for discovering organizations of texts: clustering, topic models, document embeddings, and measuring distinctive words. While the methods differ in the technical content, they share several properties. Each of the methods seek to simplify the content of documents according to some function g in order to highlight some salient difference. The goal of reducing the complexity of the documents is to facilitate the discovery of new ideas, new organizations, and ultimately new theories of the social world.

The methods that we present in this chapter form a useful foundation for any research project. At the start of any empirical paper, dissertation prospectus, or research agenda, researchers proceed with a conceptualization of the world. They decide which observations are similar, which are different, and which are not comparable. These comparison implicitly

highlight features of the observations and suppress other features, rendering some dimensions less salient.

When determining how to conceptualize the world, we often use organizations first defined on other data sets, the product of prior thinkers, or based on assumptions about how the world works. Discovery methods enable us to revise those conceptualizations and to update the building blocks of our theories to be in line with our new data or to highlight salient divisions that prior researchers might have missed.

The conceptualizations that we discover are useful to researchers. New organizations and patterns can suggest new theoretical approaches to studying an area—such as KPR with Chinese censorship—or it can affirm and enable empirical assessments in line with long standing theoretical models—such as DW-NOMINATE (Poole and Rosenthal, 1997). And these organizations form the basis of measurements and, ultimately, how we test our social science theories. What is key when revising our organizations is that prior scholarship is not wrong—it is even hard to know what it means to be wrong about a particular organization. Rather, different conceptualizations are more or less useful for different purposes. Discovering conceptualizations and then comparing them to existing organizations helps researchers to assess whether some new contribution is also useful.

The sequential approach of this book and the procedures that we describe in this chapter avoid some of the classic critiques of using discovery for social science research (Armstrong, 1967). For example, it is common to worry that discovery methods will be tautological: we will discover something because the model is guaranteed to uncover some latent feature or output a partition of the documents. Or, we might worry that discovering conceptualizations through data mining is atheoretical and fails to carefully consider other empirical evidence.

We view these objections as misplaced, outdated, or mitigated when research is sequential. The sequential approach that we take to data analysis in this book avoids the circularity concerns. If we are merely overfitting noisy data, then subsequent analyses will reveal little

underlying structure. Further, we are able to assess the content of our conceptualizations and assess whether they are meaningful or not.

Concerns about the atheoretical nature of "data mining" are also the product of now outdated views of research. "Mere" data mining was particularly maligned when data were sparse and costly to collect. When data were expensive careful theories were necessary to avoid specification hunts that would quickly exhaust data and undermine the inferences presented. But now in many settings there is an onslaught of data—in fact in many instances the problem is that data are too abundant for inferences. This enables the use of data mining for theory development.

This does not imply that we can analyze data theory free. A key signifier of social science is that researchers group together empirical discoveries to articulate a generalizable insight into social phenomena. Our existing theories will guide the data that we collect and the approaches that we think are relevant for those data. And those existing theories provide us with context to interpret the output from our models. Once we examine this output, we can then revise and refine our theories. Far from eliminating theory, then, the sequential approach to discovery places it at the center of the empirical exercise.

And once you have a conceptualization that is useful it is yours. It does not matter where it came from, how you obtained it, or why you found it. We rarely dismiss key insights because scholars have them while exercising or driving a car. Similarly, so long as a discovery method leads to new insights about the world, we should evaluate the conceptualization based on its usefulness. The key, then, is to apply the concepts to a new data set and to examine how well it helps clarify the world.

To that end, the next chapter conditions on this organization in order to *measure* key quantities of interest in text.

References

Airoldi, Edoardo M and Jonathan M Bischof. 2016. "Improving and evaluating topic models and other models of text." *Journal of the American Statistical Association* 111(516):1381–1403.

Anderson, Chris. 2008. "The End of Theory: The Data Deluge Makes the Scientific Method Obsolete." *Wired* .
URL: *https://www.wired.com/2008/06/pb-theory/*

Armstrong, J.S. 1967. "Derivation of theory by means of factor analysis or Tom Swift and his electric factor analysis machine." *American Statistician* pp. 17–21.

Bafumi, Joseph and Michael C Herron. 2010. "Leapfrog representation and extremism: A study of American voters and their members in Congress." *American Political Science Review* 104(3):519–542.

Banerjee, Arindam, Inderjit Dhillon, Joydeep Ghosh and Suvrit Sra. 2005. "Clustering on the Unit Hypersphere Using von Mises-Fisher Distributions." *Journal of Machine Learning* 6:1345–1382.

Barberá, Pablo. 2014. "Birds of the same feather tweet together: Bayesian ideal point estimation using Twitter data." *Political Analysis* 23(1):76–91.

Beauchamp, Nick. 2011. "Using Text to Scale Legislatures with Uninformative Voting." New York University Mimeo.

Benoit, K., M. Laver and S. Mikhaylov. 2009. "Treating words as data with error: Uncertainty in text statements of policy positions." *American Journal of Political Science* 53(2):495–513.

Berry, Christopher R and Anthony Fowler. 2015. "Cardinals or Clerics? Congressional Committees and the Distribution of Pork." *American Journal of Political Science* .

Bischof, Jonathan and Edoardo M Airoldi. 2012. Summarizing topical content with word frequency and exclusivity. In *Proceedings of the 29th International Conference on Machine Learning (ICML-12)*. pp. 201–208.

Blackwell, David and James B MacQueen. 1973. "Ferguson distributions via Pólya urn schemes." *The annals of statistics* pp. 353–355.

Blei, David, Andrew Ng and Michael Jordan. 2003. "Latent Dirichlet Allocation." *Journal of Machine Learning and Research* 3:993–1022.

Blei, David M and John D Lafferty. 2006. Dynamic Topic Models. In *Proceedings of the 23rd international conference on Machine learning*. pp. 113–120.

Blei, David M. and John D. Lafferty. 2007. "A Correlated Topic Model of Science." *The Annals of Applied Statistics* 1(1):17–35.

Bond, Robert and Solomon Messing. 2015. "Quantifying social medias political space: Estimating ideology from publicly revealed preferences on Facebook." *American Political Science Review* 109(1):62–78.

Bonica, Adam. 2013. "Ideology and interests in the political marketplace." *American Journal of Political Science* 57(2):294–311.

Breiman, Leo. 2001. "Statistical Modeling: The Two Cultures." *Statistical Science* 16(3):199–215.

Cameron, Charles M. 2000. *Veto bargaining: Presidents and the politics of negative power*. Cambridge University Press.

Canes-Wrone, Brandice, David W Brady and John F Cogan. 2002. "Out of step, out of office: Electoral accountability and House members' voting." *American Political Science Review* 96(1):127–140.

Catalinac, Amy. 2016. "From Pork to Policy: The Rise of Programmatic Campaigning in Japanese Elections." *The Journal of Politics* 78(1):1–18.

Celebi, M Emre, Hassan A Kingravi and Patricio A Vela. 2013. "A comparative study of efficient initialization methods for the k-means clustering algorithm." *Expert systems with applications* 40(1):200–210.

Chang, Jonathan, Sean Gerrish, Chong Wang, Jordan L Boyd-Graber and David M Blei. 2009. Reading Tea Leaves: How Humans Interpret Topic Models. In *Advances in Neural Information Processing Systems*. pp. 288–296.

Cleveland, William S. 1979. "Robust Locally Weighted Regression and Scatterplots." *Journal of the American Statistical Association* 74(368):829–836.

Clinton, Joshua, Simon Jackman and Douglas Rivers. 2004. "The Statistical Analysis of Roll Call Data." *American Political Science Review* 98(02):355–370.

Corbin, Juliet and Anselm Strauss. 1990. "Grounded Theory Research: Procedures, Canons and Evaluative Criteria." *Zeitschrift für Soziologie* 19(6):418–427.

Cunha, Evandro, Gabriel Magno, Marcos André Gonçalves, César Cambraia and Virgilio Almeida. 2014. "He votes or she votes? Female and male discursive strategies in Twitter political hashtags." *PloS one* 9(1):e87041.

Deerwester, Scott, Susan T Dumais, George W Furnas, Thomas K Landauer and Richard Harshman. 1990. "Indexing by latent semantic analysis." *Journal of the American society for information science* 41(6):391.

Dempster, Arthur P., N.M. Laird and D.B. Rubin. 1977. "Maximum Likelihood Estimation from Incomplete Data via the EM Algorithm." *Journal of the Royal Statistical Association* 39:1–38.

Denny, Matthew James and Arthur Spirling. 2018. "Text Preprocessing for Unsupervised Learning: Why It Matters, When It Matters, and What To Do About It." *Political Analysis* .

Diermeier, Daniel, Jean-Francois Godbout, Bei Yu and Stefan Kaufmann. 2011. "Language and Ideology in Congress." *British Journal of Political Science* . Forthcoming.

Dueck, Delbert and Brendan J Frey. 2007. Non-Metric Affinity Propagation for Unsupervised Image Categorization. In *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*. IEEE pp. 1–8.

Fenno, Richard F. 1978. *Home style: House members in their districts*. HarperCollins.

Fong, Christian and Justin Grimmer. 2016. Discovery of treatments from text corpora. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Vol. 1 pp. 1600–1609.

Fraley, C. and A.E. Raftery. 2002. "Model-based clustering, discriminant analysis, and density estimation." *Journal of the American Statistical Association* 97(458):611–631.

Frey, BJ and D Dueck. 2007. "Clustering by Passing Messages Between Data Points." *Science* 315(5814):972.

Gentzkow, Matthew and Jesse M Shapiro. 2010. "What drives media slant? Evidence from US daily newspapers." *Econometrica* 78(1):35–71.

Gentzkow, Matthew, Jesse M Shapiro and Matt Taddy. 2016. Measuring polarization in high-dimensional data: Method and application to congressional speech. Technical report National Bureau of Economic Research.

Gerber, Elisabeth and Jeff Lewis. 2004. "Beyond the Median: Voter Preferences, District Heterogeneity, and Political Representation." *Journal of Political Economy* 112(6):1364–1383.

Ghahramani, Zoubin and Thomas L Griffiths. 2006. Infinite latent feature models and the Indian buffet process. In *Advances in neural information processing systems*. pp. 475–482.

Grimmer, Justin. 2010. "A Bayesian Hierarchical Topic Model for Political Texts: Measuring Expressed Agendas in Senate Press Releases." *Political Analysis* .

Grimmer, Justin. 2013. *Representational Style in Congress: What Legislators Say and Why It Matters*. Cambridge University Press.

Grimmer, Justin and Gary King. 2011. "General Purpose Computer-Assisted Clustering and Conceptualization." *Proceedings of the National Academy of Sciences* 108(7):2643–2650.

Hastie, Trevor, Robert Tibshirani and Jerome Friedman. 2001. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. New York: Springer.

Hofmann, Thomas. 1999. Probabilistic latent semantic analysis. In *Proceedings of the Fifteenth conference on Uncertainty in artificial intelligence*. Morgan Kaufmann Publishers Inc. pp. 289–296.

Humphreys, Macartan, Raul Sanchez de la Sierra and Peter Van der Windt. 2013. "Fishing, commitment, and communication: A proposal for comprehensive nonbinding research registration." *Political Analysis* 21(1):1–20.

Johnson, Stephen C. 1967. "Hierarchical clustering schemes." *Psychometrika* 32(3):241–254.

Krehbiel, Keith. 1990. "Are congressional committees composed of preference outliers?" *American Political Science Review* 84(1):149–163.

Krehbiel, Keith. 1998. *Pivotal politics: A theory of US lawmaking*. University of Chicago Press.

Laver, Michael, Kenneth Benoit and John Garry. 2003. "Extracting Policy Positions from Political Texts Using Words as Data." *American Political Science Review* 97(02):311–331.

Li, Wei and Andrew McCallum. 2006. Pachinko allocation: DAG-structured mixture models of topic correlations. In *Proceedings of the 23rd international conference on Machine learning*. ACM pp. 577–584.

Lowe, Will. 2008. "Understanding Wordscores." *Political Analysis* 16(4):356–371.

Lowe, Will, Ken Benoit, Slava Mihaylov and M Laver. 2011. "Scaling Policy Preferences from Coded Political Texts." *Legislative Studies Quarterly* 36(1):123–155.

MacRae, Duncan. 1965. "A Method for Identifying Issues and Factions from Legislative Votes." *American Political Science Review* 59(4):909–926.

Martin, Lanny and Georg Vanberg. 2007. "A Robust Transformation Procedure for Interpreting Political Text." *Political Analysis* 16(1):93–100.

McCallum, Andrew Kachites. 2002. "Mallet: A machine learning for language toolkit.".

McGhee, Eric, Seth Masket, Boris Shor, Steven Rogers and Nolan McCarty. 2014. "A primary cause of partisanship? Nomination systems and legislator ideology." *American Journal of Political Science* 58(2):337–351.

McLachlan, Geoffrey and David Peel. 2004. *Finite Mixture Models*. John Wiley & Sons.

Meilă, Marina. 2007. "Comparing Clusteringsan Information Based Distance." *Journal of Multivariate Analysis* 98(5):873–895.

Mikhaylov, S., M. Laver and K. Benoit. 2010. Coder reliability and misclassification in the human coding of party manifestos. In *66th MPSA Annual National Conference, Palmer House Hilton Hotel and Towers*.

Mimno, David, Hanna M Wallach, Edmund Talley, Miriam Leenders and Andrew McCallum. 2011. Optimizing Semantic Coherence in Topic Models. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*. pp. 262–272.

Monroe, Burt and Ko Maeda. 2004. "Talk's Cheap: Text-Based Estimation of Rhetorical Ideal Points." 21st Annual Summer Meeting of the Society of Political Methodology.

Monroe, Burt, Michael Colaresi and Kevin Quinn. 2008. "Fightin' Words: Lexical Feature Selection and Evaluation for Identifying the Content of Political Conflict." *Political Analysis* 16(4):372–403.

Mosteller, Frederick and David L Wallace. 1963. "Inference in an authorship problem: A comparative study of discrimination methods applied to the authorship of the disputed Federalist Papers." *Journal of the American Statistical Association* 58(302):275–309.

Nelson, Laura K. 2017. "Computational Grounded Theory: A Methodological Framework." *Sociological Methods & Research* .

Ng, Andrew, Michael Jordan and Yair Weiss. 2002. "On Spectral Clustering: Analysis and an Algorithm." *Advances in Neural Information Processing Systems 14: Proceedings of the 2002 Conference* .

Pena, José M, Jose Antonio Lozano and Pedro Larranaga. 1999. "An empirical comparison of four initialization methods for the k-means algorithm." *Pattern recognition letters* 20(10):1027–1040.

Pitman, Jim and Marc Yor. 1981. Bessel processes and infinitely divisible laws. In *Stochastic integrals*. Springer pp. 285–370.

Platt, J.C. 2005. "Fastmap, MetricMap, and Landmark MDS are all Nystr
"om algorithms." *Proceedings of the 10th International Workshop on Artificial Intelligence and Statistics* pp. 261–268.

Poole, Keith and Howard Rosenthal. 1991. "Patterns of Congressional Voting." *American Journal of Political Science* 35:228–278.

Poole, Keith T and Howard Rosenthal. 1985. "A spatial model for legislative roll call analysis." *American Journal of Political Science* pp. 357–384.

Poole, Keith T and Howard Rosenthal. 1997. *Congress: A political-economic history of roll call voting*. Oxford University Press on Demand.

Powell, Eleanor Neff. N.d. *Where Money Matters in Congress: A Window into How Parties Evolve*. Cambridge University Press.

Quinn, Kevin et al. 2010. "How to Analyze Political Attention with Minimal Assumptions and Costs." *American Journal of Political Science* 54(1).

Reed, Colorado. N.d. "Latent Dirichlet Allocation: Towards a Deeper Understanding." . Forthcoming. UC Berkeley, Mimeo.
URL: *http://obphio.us/pdfs/lda_tutorial.pdf*

Roberts, Margaret E., Brandon M. Stewart and Dustin Tingley. 2017. *stm: R Package for Structural Topic Models*. R package version 1.2.3.

Roberts, Margaret E, Brandon M Stewart, Dustin Tingley, Christopher Lucas, Jetson Leder-Luis, Shana Kushner Gadarian, Bethany Albertson and David G Rand. 2014. "Structural Topic Models for Open-Ended Survey Responses." *American Journal of Political Science* 58(4):1064–1082.

Roberts, Margaret E, Brandon M Stewart and Edoardo M Airoldi. 2016*a*. "A model of text for experimentation in the social sciences." *Journal of the American Statistical Association* pp. 1–49.

Roberts, Margaret E, Brandon M Stewart and Edoardo M Airoldi. 2016*b*. "A model of text for experimentation in the social sciences." *Journal of the American Statistical Association* 111(515):988–1003.

Roweis, Sam T and Lawrence K Saul. 2000. "Nonlinear dimensionality reduction by locally linear embedding." *Science* 290(5500):2323–2326.

Schofield, Alexandra and David Mimno. 2016. "Comparing Apples to Apple: The Effects of Stemmers on Topic Models." *Transactions of the Association for Computational Linguistics* 4:287–300.

Schofield, Alexandra, Måns Magnusson and David Mimno. 2017. Pulling out the stops: Rethinking stopword removal for topic models. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*. Vol. 2 pp. 432–436.

Schofield, Alexandra, Måns Magnusson, Laure Thompson and David Mimno. N.d. "Understanding Text Pre-Processing for Latent Dirichlet Allocation." . Forthcoming.

Schultz, Kenneth A. 1998. "Domestic opposition and signaling in international crises." *American Political Science Review* 92(4):829–844.

Shannon, Claude E. 1949. *The Mathematical Theory of Communication*. Urbana-Champaign: University of Illinois Press.

Shor, Boris and Nolan McCarty. 2011. "The ideological mapping of American legislatures." *American Political Science Review* 105(3):530–551.

Slapin, Jonathan and Sven-Oliver Proksch. 2008. "A Scaling Model for Estimating Time-Series Party Positions from Texts." *American Journal of Political Science* 52(3):705–722.

Snyder Jr, James M. 1992. "Committee power, structure-induced equilibria, and roll call votes." *American Journal of Political Science* pp. 1–30.

Spirling, Arthur. 2012*a*. "US treaty making with American Indians: Institutional change and relative power, 1784–1911." *American Journal of Political Science* 56(1):84–97.

Spirling, Arthur. 2012*b*. "US treaty making with American Indians: Institutional change and relative power, 1784–1911." *American Journal of Political Science* 56(1):84–97.

Spirling, Arthur and Iain McLean. 2007. "UK OC OK? Interpreting Optimal Classification Scores for the UK House of Commons." *Political Analysis* 15(1).

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

Tausanovitch, Chris and Christopher Warshaw. 2017. "Estimating candidates political orientation in a polarized congress." *Political Analysis* 25(2):167–187.

Vrehuuvrek, Radim and Petr Sojka. 2011. "GensimStatistical Semantics in Python.".

Wallach, Hanna M, Iain Murray, Ruslan Salakhutdinov and David Mimno. 2009. Evaluation Methods for Topic Models. In *International Conference on Machine Learning*. pp. 1105–1112.

Wallach, Hanna, Shane Jensen, Lee Dicker and Katherine Heller. 2010. An Alternative Prior Process for nonparametric Bayesian clustering. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*. pp. 892–899.