Discovery

Justin Grimmer* Margaret E. Roberts † Brandon Stewart ‡

January 12, 2018

Abstract

1 Introduction

- How do we generate research ideas and ask questions?
- Usual assumption in the social science and the usual way we present our ideas: we arrive at our data with some hypothesis in mind
    - The prior literature gives us some way of thinking about the issue.
    - But there is some problem with that literature and we want to address it.
    - Or, there is some new way we want to attack a problem that leads us to new discoveries.
- But with new data, the prior literature might give us little to know.
    - Or, we might be interested in exploring different ways of thinking about organizing the data or uncovering new patterns in the data.
    - The big question, then, is where do our ideas for organizing data come from and how do these organizations then feed into our hypotheses
- This chapter is all about how to use statistical models for exploration. We show statistical models can lead to new ideas, organizations, and measurements that then feed into the rest of the research process.
- We focus on the use of methods that are intended to prompt new ideas

    a) Clustering

*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

b) Principal Components and Multidimensional Scaling

c) Methods for discriminating words.

d) Topic models?

- We also provide guidance on how to use these methods. There are common objections that we also address

1) Electric factor analysis machine

2) Overfitting

3) Role of concepts in the analysis.

- Examples in this chapter: Congress, China, and Digital Humanities

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. And empirical bargaining paper, examines how bargaining in front of an audience affects the probability of war. This requires organizing 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. 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).

In this chapter we provide a methodology for discovering and applying new ways of organizing our observations. In particular, we show how texts, quantitative methods for discovery, and careful reading can facilitate new concepts or reemphasize the importance of existing organizational schema. 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, 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 methos, 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 mentioned in Chapter **??** 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. While we conclude the chapter responding to these sorts of criticisms, we preview our response here. 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 meaures 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 attention to these concerns

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.

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" (**?**).

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 tra-

ditional 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 devlopment. 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 are intended to contribute to the sequence of the scientific process.

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.

When discovering conceptualizations, these concerns are misplaced. This is because there is no meaningful sense in which a conceptualization can be biased, because as we just discussed, there is no true value of a conceptualization. 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 on social media, or the thematic content of war negotiations.

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 beacuse 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. Below we describe several ways to make this split.

2 Conceptualizing the US Congress

To motivate how statistical models can facilitate new insights, inferences, and theories in text, we begin the chapter with 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.

2.1 Conceptualizing Conflict in the US Congress

Scholars of American politics develop theories to explain when and how Congress shapes public policy and legislators' diverse preferences are aggregated to reach decisions. 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 occassionally, 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 (CITATION). 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. 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. 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 dizzying array of other features of Congressional conflict. And 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 (CITATIONS). 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, donors, social media users, bureaucrats, and voters. (ADD CITATIONS HERE)

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. 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. 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.

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 na-

tional 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. This 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.

In this Chapter we describe methods that use text data to facilitate conceptualizations. We explore four distinct methods that suggest new ways to explore data and organize texts. We begin with *unsupervised cluster analysis*, a method that both discovers groups and places documents into those groups.

3 Unsupervised Clustering Analysis

The goal of unsupervised clustering analysis is to estimate a set of K categories and placing the documents into those K categories or *partitioning* 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 $\boldsymbol{C}_i = (C_{i1}, C_{i2}, \ldots, C_{iK})$, where each C_{ik} corresponds to the share of document i that is assigned to cluster k. If $C_{ik} \in \{0, 1\}$, then we will say that the partition is *hard* and if $C_{ik} \in [0, 1]$ 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: 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, 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 analysis of unsupervised 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 a description of 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 multinomial distributions.

3.1 K-Means Clustering

In this section we will analyze a collection of press releases from Grimmer (2013), which examines the ways members of Congress present their work to their constituents. We analyze a subset of NN press releases, from YEAR. Our first goal will be to use the k-means clustering algorithm to identify groups of press releases with similar content.

As we described in Chapter XX, we suppose that we have preprocessed our N texts, so that each text is a $J \times 1$ count vector, $\boldsymbol{X}_i$. 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 partition. 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 $\boldsymbol{C}$ 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:

$$d(\boldsymbol{X}_i, \boldsymbol{\mu}_k) \;=\; \sum_{j=1}^{J} (X_{ij} - \mu_{jk})^2 \tag{3.1}$$

Using this measure of dissimilarity, we can assess the quality of any partition $\boldsymbol{C}$ 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, $C_{ik} \in \{0, 1\}$. We will see below that the opposite, *soft* assignments, supposes that $C_{ik} \in [0, 1]$. 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 :

$$f(\boldsymbol{C}, \boldsymbol{\mu}, \boldsymbol{X}) = \sum_{i=1}^{N}\sum_{k=1}^{K}\sum_{j=1}^{J} \overbrace{C_{ik}}^{\text{Cluster indicator}} \underbrace{(X_{ij} - \mu_{jk})^2}_{\text{dissimilarity measure}} \tag{3.2}$$

In words, the objective function measures the dissimilarity of documents from their assigned cluster centers, using the definition of dissimilarity from Equation 3.1.

Given this objective function, there is a well defined best partition and corresponding set of cluster centers. Unfortunately, optimizing Equation 3.2 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 3.2. 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 $\boldsymbol{C}^1$. 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:

$$\mu_k^1 = \sum_{i=1}^{N} \frac{C_{ik}\boldsymbol{X}_i}{C_{ik}}$$

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 3.2, 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 $C_{ik} = 1$ if document i is closest to center k, set to 0 otherwise
    - Set $\mu_k^t = \sum_{i=1}^{N} \frac{C_{ik}\boldsymbol{X}_i}{C_{ik}}$
- Return $\boldsymbol{\mu}^*$ and $\boldsymbol{C}^*$.

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 solutions. This has two important implications for our analysis: the iterative algorithm we just described will often get stuck in local optima and there will be instability in the solutions across repeated runs of the K-means algorithm, where 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. CITATIONS HERE.

3.2 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 summary using some of the techniques we discussed in Chapter **??**. That is, for each of the summaries we discarded punctuation,

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

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

With this representation of the texts we first 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

$$\boldsymbol{x}_i^* = \frac{\boldsymbol{x}_i}{\sum_{j=1}^{J} x_{ij}}$$

We can then collect all 10,000 normalized summaries into the matrix $\boldsymbol{X}^*$. We then apply the K-Means algorithm in `R`, using the `kmeans` function. We use the default settings: we use the Hartigan-Wong algorithm to optimize the objective function.

In Table **??** 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.

3.3 A Mixture of Multinomial Distributions

K-Means provides a useful

1) What is the mixture 2) How is it applied to this particular problem 3) How do we think about the mixture of multinomials 4) Discuss estimation briefly.

3.4 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. Below, we apply the algorithms to the *ArXiv* data set and contrast the clustering to the K-Means clustering.

Table 3: Three Features of FAC Methods

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

3.4.1 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 XX. 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). 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 (CITATION), a mixture of Normal distributions measure dissimilarity as a function of squared-Euclidean distance (CITATION), 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're providing computers with a very different representation

of the information. But 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 later.

3.4.2 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 select-

ing the "best" partition given a specific notion of best. Adjudicating between the different objective functions will be an essential clustering task.

3.4.3 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 optimal partition, 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—running the same algorithm twice will sometimes result in different solutions. We can sta-

bilize 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.

3.5 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. While methods might vary in their assumptions, the methods we introduce here are useful when applied to almost any clustering, regardless of how the clustering was obtained. As **?** argue, unsupervised learning methods require little time investment upfront, but substantial work on interpretation. To interpret the output of clustering methods, we build on suggestions from **?** 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.

Method 1: Sampling Documents Assigned to Each Cluster The first 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]

Method 2: Identifying Distinctive Words The second approach that we recommend is to identify words that indicate a document will belong to a particular category. One way to identify these words will be to use parameters from a model. For example, with K-Means we can use the largest values of $\boldsymbol{\mu}_k$ and with a mixture of multinomial distributions we can use the words with the largest probability for each category $\boldsymbol{\theta}_k$. But some FAC methods, like Affinity Propagation, do not have an analogous parameter. 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.

In Section **??** we will introduce several methods for identifying distinguishing words that use sophisticated statistical techniques, building on a growing statistics and machine learning literature. But 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 **?**, while providing computation ease.

Suppose that we have document term matrix $\boldsymbol{X}$ and we have a matrix of cluster assignments $\boldsymbol{C}$ and recall that $C_{ik} = 1$ if document i is assigned to the k^{th} cluster and $C_{ik} = 0$ otherwise. For each feature j we regress $\boldsymbol{X}_j$ on C_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

[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.

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.

3.6 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.

3.6.1 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 include. This is because the K-Means objective function, like many other statistical models, improves as more cluster components 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 (CITATIONS HERE). The core intuition of the statistical approaches to determing the number of clusters is that statistical approaches can be used to

[2]We can prove that this will hold quickly. Suppose there is an optimal clustering with K clusters. Now suppose that we add one cluster and want to find a new solution with $K+1$ clusters. 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. While this may not be an optimal solution, it improves the objective function, so any optimal partition must make the objective function better.

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 Akake Information Criterion or the Bayesian Information Criterion.

PROVIDE THOSE STATISTICS HERE.

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

nese Restaurant Prior, is a prior over distributions, rather than parameters. 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 (CITE WALLACH STUFF HERE). Indeed, as the number of observations goes to infinity, the expected number of clusters from the DPP is $\xi \log(1 + \frac{N}{\xi})$ (CITE WALLACH PAPER). 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. And still other nonparametric priors, such as the Uniform process prior (WALLACH CITATION) 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. 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.

3.6.2 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 ane exclusivity (Roberts et al., 2014; **?**). 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 **?** 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 **?** 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 fo 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 nubmer 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,

$$\text{Cohesive} \quad = \quad \sum_{n=1}^{M} \sum_{l=1}^{l-1} \log \left(\frac{D(m_n, m_l)}{D(m_l)} \right) \tag{3.3}$$

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 topic $\boldsymbol{\mu}_k$ 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 calcuation is made:

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

Clusterings will 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 organiations. 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

[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 evluation is gone.

[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.

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.

3.7 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 (RAFTERY), others based on algorithmic approaches, and still others that appeal to graph-theory notions of community detection. Some methods search for a single partitioning of the data, other methods build a hierarchy or tree-based clustering. 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 breath taking. 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 (**?**), mixtures of von mises Fisher distributions (**?**), and mixtures of Dirichlet distributions to characterize mixtures of proportions (**?**). 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 (**?**).

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 (**?**). Similarly, affinity propagation uses message passing discover groups of high similarity documents (**?**). And even the K-Means algorithm supposes that the data are divisble 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" (**?**, 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" (**?**, 611). In contrast, algorithmic papers often tout their ability to solve difficult problems (CITATION) and prove theorems that demonstrate the conditions where the clustering algorithm will perform optimally (CITATION).

We view the model and algorithmic distinctions as overwrought (**?**). 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 **?** 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. A different unsupervised method, latent Dirichlet allocation, supposes that documents are comprised of a mixture of topics and offer a qualitatively different result.

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 hierar-

chy 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. While this does not provide a nesting of clusters, it provides a more general set of partitions of the documents. 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.

4 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 XXXX, and hierarchical clustering. We summarize the results in three ways. First, we use the distance metric between clusterings to create a visualization using methods we describe in the next section. Second, we use

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

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

a series of confusion matrices to compare the clusterings across methods. And third we summarize the key words for each category using a simple method that we describe below.

INSERT LOW-DIM Embedding here INSERT CONFUSION MATRICES HERE INSERT ClUSTER SUMMARY HERE.

4.1 Affinity Propagation

4.2 Mixture of von Mises Fisher distributions

4.3 Spectral Clustering

We use the normal kernel and spectral clustering

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

Table 8: Applying Spectral Clustering to *ArXiv* Summaries

Cluster Label	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

5 Computer-Assisted Clustering

The previous section demonstrated 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 converted (Spirling and Denny 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 (XXX) to create a distance matrix between the clusterings. Fourth, they project that distance matrix to two-dimensions using multi-dimensional 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*

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 (XXX)

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.

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. 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.

6 Latent Dirichlet Allocation and Vanilla Topic Models

Despite the diversity of clustering algorithms, at their core they share a common assumption that each documents belongs, is assigned to, only one cluster. 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) (**?**).[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

[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 CITATIONS HERE. 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 model.

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 x_{im}. We will suppose that each word has a corresponding topic indicator that is a draw from a Multinomial distribution $\tau_{im} \sim \text{Multinomial}(1, \boldsymbol{\pi_i})$. Then, conditional on $\tau_{imk} = 1$ we will assume that x_{im} is a draw from a multinomial distribution $x_{im}|\tau_{imk} = 1 \sim \text{Multinomial}(1, \boldsymbol{\theta}_k)$, where $\boldsymbol{\theta}_k = (\theta_{1k}, \theta_{2k}, \ldots, \theta_{Jk})$ is a $J \times 1$ vector where each θ_{jk} describes the probability of using word j when discussing topic k. We complete the data-generating process with priors, assuming that both $\boldsymbol{\pi}_i$ and $\boldsymbol{\theta}_k$ are drawn from Dirichlet distribution. (NOTE HERE ABOUT ASYMMETRIC PRIORs). The full posterior is described in Equation **??**

$$\begin{aligned} p(\boldsymbol{\pi}, \boldsymbol{\Theta}, \boldsymbol{T}|\boldsymbol{X}, \boldsymbol{\alpha}) &\propto p(\boldsymbol{\pi}|\boldsymbol{\alpha})p(\boldsymbol{T}|\boldsymbol{\pi})p(\boldsymbol{X}|\boldsymbol{\Theta}, \boldsymbol{T}) \\ &\propto \prod_{i=1}^{N} \left[p(\boldsymbol{\pi_i}|\boldsymbol{\alpha}) \prod_{m=1}^{M_i} p(\boldsymbol{\tau}_{im}|\boldsymbol{\pi}_i) p(x_{im}|\boldsymbol{\theta}_k, \tau_{imk} = 1) \right] \end{aligned} \tag{6.1}$$

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 (CITATION FROM THE TUTORIAL). To gain this intuition, we will focus on the representation of the prior in Equation 6.1. If we first focus on the component of the model for generating the text, $p(\boldsymbol{X}|\boldsymbol{\Theta}, \boldsymbol{T})$, we can note that we are going to try and find values of $\boldsymbol{\Theta}$ that make the observed document term matrix more likely. This will be true when the components of $\boldsymbol{\Theta}$ 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 CITATION HERE. 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) CITATION HERE. And an extension of this estimation strategy, online variational approximations, facilitates the application of topic models to extremely large collections of documents without requiring any additional computation power. (GENSIM CITATION

HERE).

6.1 Example: Catalanic Work on Japanese Documents

Describe the data and how Amy Fit the model.

6.2 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 becaus they are common. We explain in Section 8 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 (CITE STM HERE). 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 (2015) Describe here how Amy labeled the topics and how they were helpful. this is the link to the supplemental file where she has the content

7 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 introduce 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. We begin with perhaps the most widely

used method for generating low-dimensional approximations of high-dimensional observations: Principal Component Analysis (PCA).

7.1 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, $\boldsymbol{x}_i = (x_{i1}, x_{i2}, \ldots, x_{iJ})$, 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 $\boldsymbol{w}_k = (w_{k1}, w_{k2}, \ldots, w_{kJ})$ the k^{th} principal component and we will collect all the principal components into a $K \times J$ matrix $\boldsymbol{W}$. 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 $\boldsymbol{z}_i = (z_{i1}, z_{i2}, \ldots, z_{iK})$. Using the loadings and the principal components, we will write each observation $\boldsymbol{x}_i$ as,

$$\boldsymbol{x}_i = \underbrace{z_{i1}\boldsymbol{w}_1 + z_{i2}\boldsymbol{w}_2 + \ldots + z_{iK}\boldsymbol{w}_K}_{\tilde{\boldsymbol{x}}_i} + \overbrace{\boldsymbol{\epsilon}_i}^{\text{error}} \tag{7.1}$$

where the $J \times 1$ vector $\boldsymbol{\epsilon}_i = (\epsilon_{i1}, \epsilon_{i2}, \ldots, \epsilon_{iJ})$ is an error term and $\tilde{\boldsymbol{x}}_i$ is the approximation of $\boldsymbol{x}_i$. Notice that the loadings $\boldsymbol{z}$ 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 $\boldsymbol{z}_i$ and $\boldsymbol{W}$ to minimize the magnitude of the error. That is, we will choose $\boldsymbol{z}_i$ and $\boldsymbol{W}$ to minimize:

$$
\begin{aligned}
f(\boldsymbol{Z}, \boldsymbol{W}) &= \frac{1}{N}\sum_{i=1}^{N} \epsilon_i^{'}\epsilon_i \\
&= \frac{1}{N}\sum_{i=1}^{N} (\boldsymbol{x}_i - \sum_{k=1}^{K} z_{ik}\boldsymbol{w}_k)^{'}(\boldsymbol{x}_i - \sum_{k=1}^{K} z_{ik}\boldsymbol{w}_k) \qquad (7.2)
\end{aligned}
$$

The optimal solution to minimize the magnitude of the error in Equation 7.2 is to use the K eigenvectors associated with the largest K eigenvalues of $\boldsymbol{X}^{'}\boldsymbol{X}$, the variance-covariance matrix, as the K principal components. Further, the loading for the i^{th} observation on the k^{th} principal component is $z_{ik} = \boldsymbol{w}_k^{'}\boldsymbol{x}_i$. 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 associatd 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 distallation 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.

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 $\boldsymbol{w}_k$ that are especially positive or negative. Specifically, we can choose say the ten words with the highest values in $\boldsymbol{w}_k$ 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 z_{ik}. This is because $z_{ik} = \boldsymbol{x}_i^{'}\boldsymbol{w}_k$, so the entries in $\boldsymbol{w}_k$ 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 (WARSHAW TAUSANOVITCH).

Manual Methods for Labeling Principal Componennts 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.

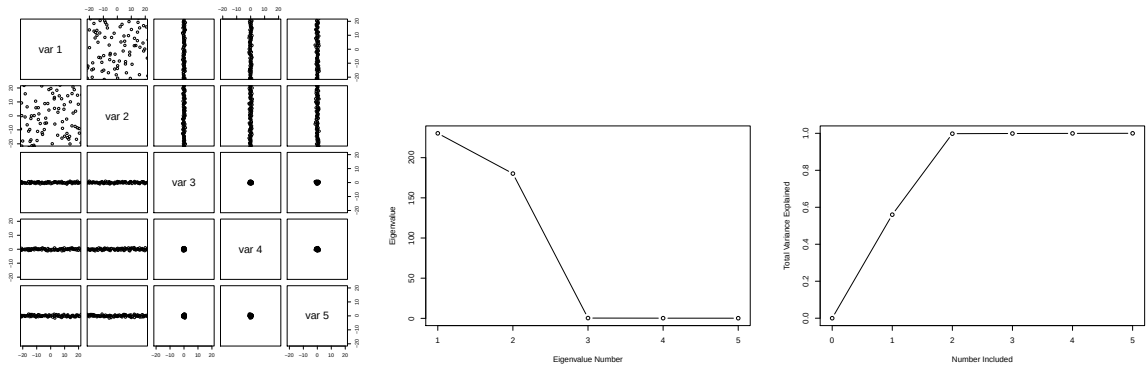
7.1.1 Principal Component Analysis of XX

7.1.2 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 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 (PROBABILISTIC MACHINE LEARNING). This implies that we can use the size of the eigenvalues to guide our model selection. To see why, consider Figure 1. 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

Figure 1: Example of Using PCA to Make Model Determination



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

Intuition from the right-hand plot in Figure 1 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 1. 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 1 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 dif-

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

ferent 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 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.

7.2 Multidimensional Scaling and close Relationship to PCA

7.3 WordFish, Factor Analysis, and Intro to IRT

USE CONTENT FROM OUR 2013. PAPER

Unsupervised Key word IR stuff.

Paper + Graphics.

7.4 Sparse Factor Analysis and the Indian Buffet Process

Example :

8 Discriminating Words

Fictitous prediction problem. 1) Simplest procedure a) Standardized difference in means

2) Fightin' words and Gentzkow, Shapiro, and Taddy and regularization (introduction).

3) Mutual Information

4) Jason Chuang. without the clutter of unimportant words.

8.1 Mutual Information

Here we describe an approach to label identifying words based on the *mutual information* between words and a cluster. The mutual information between words and a cluster is an information theoretic measure that describes how much knowing a particular word resolves our uncertainty about a document's category. To motivate the use of mutual information, suppose we were given the task of guessing a randomly selected document's cluster assignment. 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 p_k represents the proportion of documents that fall in category k and let $p_{-k} = 1 - p_k$ or the probability a document does not belong to category k.[7] p_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 $H(k) = -p_k \log_2 p_k - p_{-k} \log_2 p_{-k}$, 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 $p_{k,j}$ as the proportion of documents that are both in category k and have word j and $p_{k|j}$ as the proportion of documents in category k given that word j is present. We can then define conditional entropy as:

[7] In general, we will use a negative subscript to indicate that

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

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,

$$MI_{k,j} = H(j) - H(k|j)$$

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.

8.2 Example: What is a Republican word

9 Objections to text as data as discovery

1) Circularity of the argument 2) "Throwing spaghetti against the wall": data mining –"you can find whatever you want to" 3) Theory-less exploration 4) This isn't a substitute for going to the field. "I have an enormous data set for this country, i don't have to spend time in the field". But "spending time there gives you so much context". Does it tradeoff. There is this whole other world and information. Text can only tell what is in the text, but you have to go beyond the text to know what is missing. How was this sample created. 5) Nobody every

talks about ideas. Can we pose better questions and be better educators if we have methods that facilitate coherent and useful conceptualizations? But it doesn't matter, necessarily, where the questions come from.

Things that are legitimately discovered in this way: 1) Ideology in the US Congress 2) Industry examples. 3) Cheerleading.

Directed Reading is really the key.

Could have a much bigger impact. And if you're writing your prospectus. Starting a new project.

References

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.

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.

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

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

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.

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

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.

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.

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.

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

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.

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, 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.

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.

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.

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

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.