Fast and Effective Approximations for Summarization and Categorization of Very Large Text Corpora

Andrew Godbehere

Electrical Engineering and Computer Sciences
University of California at Berkeley

Technical Report No. UCB/EECS-2015-251
http://www.eecs.berkeley.edu/Pubs/TechRpts/2015/EECS-2015-251.html

December 17, 2015

Acknowledgement

To my spouse Caitlin for all her love, support, teamwork, and patience.

Fast and Effective Approximations for Summarization and Categorization of Very Large Text Corpora

by

Andrew B Godbehere

B.S. Cornell University 2008

A dissertation submitted in partial satisfaction of the
requirements for the degree of
Doctor of Philosophy

in

Engineering - Electrical Engineering and Computer Sciences
and the Designated Emphasis
in
New Media

in the

GRADUATE DIVISION
of the
UNIVERSITY OF CALIFORNIA, BERKELEY

Committee in charge:
Laurent El Ghaoui, Chair
Lee Fleming, Trevor Darrell, Abigail De Kosnik

Fall 2015

Fast and Effective Approximations for Summarization and Categorization of Very Large Text Corpora

Abstract

Fast and Effective Approximations for Summarization and Categorization of Very Large Text Corpora

by

Andrew B Godbehere
Doctor of Philosophy in Engineering - Electrical Engineering and Computer Sciences
and the Designated Emphasis
in
New Media

University of California, Berkeley

Laurent El Ghaoui, Chair

Given the overwhelming quantities of data generated every day, there is a pressing need for tools that can extract valuable and timely information. Vast reams of text data are now published daily, containing information of interest to those in social science, marketing, finance, and public policy, to name a few. Consider the case of the micro-blogging website Twitter, which in May 2013 was estimated to contain 58 million messages per day [1]: in a single day, Twitter generates a greater volume of words than the Encyclopedia Brittanica. The magnitude of the data being analyzed, even over short time-spans, is out of reach of unassisted human comprehension.

This thesis explores scalable computational methodologies that can assist human analysts and researchers in understanding very large text corpora. Existing methods for sparse and interpretable text classification, regression, and topic modeling, such as the Lasso, Sparse PCA, and probabilistic Latent Semantic Indexing, provide the foundation for this work. While these methods are either linear algebraic or probabilistic in nature, this thesis contributes a hybrid approach wherein simple probability models provide dramatic dimensionality reduction to linear algebraic problems, resulting in computationally efficient solutions suitable for real-time human interaction.

Specifically, minimizing the probability of large deviations of a linear regression model while assuming a k-class probabilistic text model yields a k-dimensional optimization problem, where k can be much smaller than either the number of documents or features. Further, a simple non-negativity constraint on the problem yields a sparse result without the need of an ℓ_1 regularization. The problem is also considered and analyzed in the case of uncertainty in the model parameters. Towards the problem of estimating such probabilistic text models, a fast implementation of Sparse Principal Component Analysis is investigated and compared with Latent Dirichlet Allocation. Methods of fitting topic models to a dataset are discussed. Specific examples on a variety of text datasets are provided to demonstrate the efficacy of the proposed methods.

Laurent El Ghaoui
Dissertation Committee Chair

To my spouse Caitlin for all her love, support, teamwork, and patience.

Contents

Chapter 1

Introduction

In an age of Big Data, text content is omnipresent and voluminous, yet actionable information can be difficult to acquire. While every news article, patent application, financial transaction, and Tweet is archived, there are vastly more documents in these datasets than an individual could ever hope to read and understand. Comprehension of the content these corpora, while essential for decision-making and social-scientific research, is beyond the capability of the unassisted analyst or researcher.

For the methodologies and applications of this dissertation, the size of “Big Data” is relative to the abilities of a person to manage. It is not defined by data on the order of multiple terabytes, or by data distributed across many servers. Rather, “Big Data” can mean a text corpus on the order of a few hundred megabytes, which can contain on the order of 10,000 news articles, or 100,000 Twitter messages. An individual attempting to read a corpus of this size would have to resort to some sort of sampling approach: read a few articles here and there, possibly guided by a keyword search, and try to develop a general understanding of the content, major stories, and trends.

The contributions of this dissertation are directed towards the goal of developing a technology that can assist researchers, analysts, or lay-persons to navigate and comprehend text data.

There are some existing tools, such as Google n-grams [27], that provide a user with statistics and information about text on demand. These analyses count the occurrences of keywords over time, and the resulting time series can provide insight into topical trends or to changing word meanings. For researchers in the humanities and social sciences, however, it is important to interact directly with source content, which is inaccessible in Google n-grams; the difficulty is in navigation, categorization, comparing, and detecting relationships between words and concepts. Additionally, the most useful abstractions are flexible, permitting interaction and exploration, so the technical goals target a human-in-the-loop system. To facilitate rich real-time interaction, the methodologies described in this dissertation are designed to be as computationally efficient as possible in order to facilitate real-time interaction necessary for an assistive tool: the methods must be fast and they must be effective in delivering useful, relevant, and timely information.

1.1 Contributions

Theoretical contributions of this dissertation center around the development of a hybrid approach to text analysis, situated between existing approaches which either identify latent patterns in text with linear algebraic methods, or with generative probabilistic models. Linear algebraic methods, such as sparse regression and low-rank approximation [12, 13, 48] operate directly on observed data. Methods using probabilistic models, on the other hand, estimate statistics, the parameters of which convey information about the underlying patterns within text. This dissertation proposes methods at the intersection of the two approaches, in which estimated statistics are used to dramatically reduce the size of the linear regression and classification problems, and where Sparse Principal Component Analysis (SPCA) may be leveraged to estimate multiple probability models from observed data.

This dissertation also explores practical applications of the proposed tools. Given that the tools are intended to be interactive for a non-technical audience, the primary considerations in evaluation are computational efficiency and interpretability of results. The aim is not necessarily to develop accurate models of text but to develop useful abstractions.

First, an implementation of SPCA is compared to a fast estimation of the Latent Dirichlet Allocation (LDA) model [4, 18, 36, 42] in order to establish the suitability of SPCA to interactive systems. Subsequently, applications of the proposed methodology to real-world data such as news, Twitter messages, a work of fiction, and an archive of United States Patent applications. Examples establish the legibility of the results and describes how researchers can use and combine the proposed methods to gain insight.

Concretely, this dissertation contributes the following:

1. a robust approach to text classification and feature selection, where a sparse solution is recovered in the absence of a regularization term.

2. an analysis of the robust classification and feature selection problem in the presence of uncertainty in parameters. This approach is demonstrated to function well on unbalanced classification problems in Chapter 6.

3. an analysis of the performance of SPCA in contrast with LDA on an archive of over 400,000 news articles from BBC. SPCA is demonstrated to be suitable for real-time interaction and computes results at least an order of magnitude faster than LDA.

4. a method for computing similarity between a topic, represented as a categorical distribution, and an observed document, based on the Hellinger distance.

5. Example uses:

    (a) Discovering associated keywords to a query (query expansion) in an archive of news from Aljazeera English. Differences in results from using different probability models are described in terms of their relevance to specific semantic insights.

    (b) Discovering previously unknown conversations, opinions, and populations on Twitter.

    (c) Analyzing descriptions of and actions taken by a character in "Harry Potter and the Sorcerer's Stone", introducing new modes of reading.

    (d) Uncovering topics, sub-topics, and trends within a selection of United States Patent applications pertaining to Clean Technology.

1.2 Data Sources & Application Areas

Text data exists in a wide variety of contexts, outlined here to establish potential application areas.

- **Academic texts**: summarize research fields, find related documents, recommend articles.
- **News**: compare coverage of stories between news outlets or countries.
- **E-mail**: navigate content and conversations hierarchically based on content, useful when exact words used in desired email are unknown.
- **Social media**: uncover distinct populations and their opinions, identify timely topics, and track topic evolution over time.
- **Fan fiction**: explore different characterizations of characters among authors, discover how characters capture the imagination of fans.
- **Legal texts** (e.g. patents): discover legal precedent or prior art.
- **Congressional Proceedings Transcripts**: enable the voting public to read and understand the opinions and actions of representatives.
- **Earnings call transcripts**: tap into unstructured text sources that include business insights useful for investment.
- **Product reviews**: ascertain differences in opinion about a product, summarize recommendations or praise from customers.
- **File systems**: dynamic reorganization of file system relative to content and user input.

While text content is often dubbed “free” and “open”, in practice it may be impossible for a lay-person to read and comprehend, requiring domain specialists to interpret and convey to a broader audience. In the example of Congressional

Proceedings Transcripts, this has important sociopolitical consequences. Data that is touted to be public is, in practice, opaque. The voting public relies on domain specialists to interpret the proceedings of Congress and the Senate, which is typically filtered through a few major news outlets. Open and accessible democracy in this case requires tools usable and interpretable by a non-technical audience. Providing data is itself insufficient; this data must be delivered with the means to extract useful and relevant information. This application area has been analyzed previously to identify voting patterns among representatives. [33] The methods of this dissertation can expand upon this work to navigate the rich text content of the archives; what representatives say and how their statements evolve may provide much more feedback to voters than just their voting records.

1.3 Computational Abstractions

This section describes three major computational abstractions that serve as the foundation for this work: text vectorization, probabilistic models of text, and linear algebraic algorithms. Contributions of this dissertation, beginning with Chapter 2, involve a synthesis of these approaches.

1.3.1 Text Vectorization

The idea that useful information can be much simpler than the collection of data as a whole can be traced to the beginning of statistical methods of scientific inquiry. The work of Pierre Laplace [17], for instance, involved compiling meticulously recorded (but noisy and somewhat unreliable) data about the positions of celestial bodies over time, and leveraging new statistical methods to extract the useful information of the parameters of the simple Newtonian trajectories. Vast reams of data could be interpreted as a simple equation, and used to accurately predict the positions of celestial bodies, much to the amazement of Laplace's contemporaries.

In the mid 20th century, Claude Shannon famously defined a mathematical quantity termed "information" [38]. This measure professes to quantify how informative a stream of data may be, and offers a method by which to distill data into "information." Shannon's information, like Laplace's statistical estimation procedures, propose statistical frameworks within which to define signal (information) and noise, and thereby separate the two.

Text data gathered from news and social media, however, are very complicated signals, with no clear and simple probabilistic description. The useful perspectives, approaches, and models varies depending on the question or application or on the sort of action to be taken with the information.

With this caveat, consider the widely used "bag-of-words" representation of text [4, 12, 13, 19, 20, 39, 46]. This concept is expanded to a "bag-of-features" representation, where a "feature" is a generalization of a "word." Broadly, within the "bag-of-features" model, a text can be represented as a collection of symbols from some dictionary. These features may be more than just words, often called "unigrams." Unigrams may be augmented with tags, like parts-of-speech [28,39]. Or, features may be n-grams, representing short sequences of words. Information encoded in this model is contained within the features used and how often each is used within a document. Once a document transformed into features, order is ignored, and the collection of features is transformed into a numerical vector.

As an example: suppose a block of text reads "a b c b", and a dictionary maps each feature (in this case defined to be a letter) to a number, i.e. "1,2,3,2". If the dictionary is of size m, the vector $v \in \mathbb{R}^m$ is such that $v_1 = 1$, $v_2 = 2$, $v_3 = 1$, and $v_i = 0$ for all $i > 3$. This vector records simply which terms appeared in the block of text and how many times each appeared.

This model is not intended to be realistic; lacking order, much semantic information is lost. The benefit is simplicity and the ability to use computationally efficient tools to extract useful information. As an example, methods have been proposed to access the "latent semantic structure" of text [9, 19, 30], meaning identifying the groups of features commonly occurring together within the same document. It should be noted that the unit of analysis with this model is application-dependent. With very long documents, features from the beginning and end are considered to be just as related as words appearing together within the same sentence. As such, a document unit may be a sentence, paragraph, or chapter rather than a document in its entirety.

1.3.2 Probabilistic Models

Textual communication has been analyzed as a stochastic process since Claude Shannon introduced “A Mathematical Theory of Communication” [38]. Recent work in probabilistic modeling of text, such as Latent Dirichlet Allocation (LDA) [4], employ graphical models assuming a generative process by which documents and corpora are created. The statistics of this process, namely the parameters of the distributions in the model, are used as a succinct description of the content of the dataset. Within LDA, a random process selects a mixture of possible topics (represented as probability distributions on the set of possible features), and features in the document are generated by selecting a specific topic and subsequently generating a feature from that random process.

This work employs earlier probabilistic models of text, called the Binary Independence Model (BIM) [25] and probabilistic Latent Semantic Indexing (pLSI) [19]. These models use simpler generative probabilistic models and parameters of which are efficient to estimate. Further, their utility in generating meaningful results is demonstrated in Chapter 6. These models will be introduced in detail in Chapter 2.

Analysis of text data using probabilistic models can rely on well developed theories and algorithms of estimation.

1.3.3 Linear Algebraic Methods

Linear algebraic approaches do not explicitly posit a statistical model for the data. Rather, they constitute convex optimization problems solving regression, classification, and low-rank approximations. The solution of classification and regression problems report to a user a set of features most representative of one class with respect to another, or which features are most influential in an observed signal [12]. A list of all possible features would be overwhelming for an end-user, so algorithms are designed to generate sparse results, where the number of non-zero entries in the solution vector is small. This short list concentrates on the most important features, leading to more intuitive results.

First, the standard linear regression problem is as follows. For a data matrix $A \in \mathbb{R}^{n\times m}$, where n is the number of documents and m is the number of features, and for a target vector $b \in \mathbb{R}^n$ and regressor vector $x \in \mathbb{R}^m$:

$$\arg\min_x \|Ax - b\|_2 \tag{1.1}$$

This problem results in a vector x of weights for every feature in the data representing how predictive each is for the target vector. In order to make the result sparse and more interpretable, the Lasso [40, 48] is commonly used for text classification [13, 46], which introduces an ℓ_1 regularization term in the problem:

$$\text{Lasso: } \arg\min_x \|Ax - b\|_2 + \lambda\|x\|_1 \tag{1.2}$$

Changing the loss function from the 2-norm $\|\cdot\|$ yields two other methods, Sparse Support Vector Machine (SVM) and Logistic Regression [12]:

$$\text{Sparse SVM: } \arg\min_{x,\nu} \frac{1}{m}\sum_{i=1}^{m} h\left(y_i(A_i^T x + \nu)\right) + \lambda\|x\|_1 \tag{1.3}$$

$$\text{Logistic Regression: } \arg\min_{x,\nu} \frac{1}{m}\sum_{i=1}^{m} h\left(y_i(A_i^T x + \nu)\right) + \lambda\|x\|_1 \tag{1.4}$$

where

$$\text{Hinge Loss: } h(t) = \max(0, 1 - t) \tag{1.5}$$

$$\text{Smoothed Hinge Loss: } l(t) = \log(1 + e^{-t}) \tag{1.6}$$

The second linear algebraic problem employed is that of low-rank approximation of a matrix. This method finds a small representation of the text that extracts the most prevalent patterns, and is rooted in singular value decomposition (SVD). This type of approximation is considered by Deerwester [9] to extract the “latent semantic structure” of a body of text. The basic concept comes from the Eckart-Young-Mirsky

Theorem [11], wherein hard thresholding of singular values results in an optimal low-rank approximation in the Frobenius norm.

Represent the SVD of a rank-r matrix A as: $\sum_{i=1}^{r} \sigma_i u_i v_i^T$ where $\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_r$ are the singular values, and u_i, v_i are the left and right singular vectors respectively. The problem is to find an optimal rank $k < r$ approximation of A:

$$\arg\min_{u,v} \|A - uv^T\|_F \tag{1.7}$$

Then, the optimal rank $k < r$ approximation of A is: $\hat{A} = \sum_{i=1}^{k} \sigma_i u_i v_i^T$, where the largest k singular values are maintained and the remaining are discarded.

Sparse Principal Component Analysis (SPCA) is introduced to induce sparsity in the singular vectors, in order to make results interpretable by a person. An efficient algorithm to identify an approximate solution to this problem is described in detail in Chapter 5.

Chapter 2

Probabilistic Text Models

This chapter introduces two probabilistic models of text corpora, defining their structure, assumptions, and properties. Mentioned briefly in the Introduction, these are the Binary Independence Model (BIM) and the probabilistic Latent Semantic Indexing (pLSI) model. While these models have been around for several decades [19,25,30], their structure in expectation leads to very efficient solutions to regression problems, as will be discussed in Chapter 3. In addition, they represent effective abstractions of text for certain applications, as is explored in Chapter 6.

2.1 Notation and Core Assumptions

All probabilistic models discussed here share some fundamental assumptions. All build on top of the bag-of-features model of text, where documents are represented as vectors in some feature-space of words or short short sequences of words. The order of words or features within a document is ignored, and the models are designed to focus on relationships between words as they co-occur within documents. These relationships and associations among words can provide insight about topics discussed in a corpus, and these models are simpler to represent and analyze than more complete natural-language models.

Text corpora, as constructed in a bag-of-words model, are represented with a matrix $A \in \mathbb{R}^{n \times m}$, where n represents the number of documents in the corpora and m represents the size of the feature-space, e.g. the number of words that appear anywhere within the given text corpus. Rows of the matrix are vector representations of individual documents, and the values for each feature typically correspond to the number of occurrences of the word within the document.

Each document will be considered as a random vector generated from some probability distribution with parameter vector $\pi \in (0, 1]^m$. The parameter vector represents a condensed representation of the relationships among words for documents generated from the given distribution, and is referred to in this dissertation as a "topic". Multiple topics may exist for a corpus, but through Chapter 4 each document contains only a single topic. Further, the random vectors representing topics are pairwise independent.

2.2 Binary Independence Model

The BIM model [25] ignores the frequency of occurrence of individual words and instead represents the appearance of any given feature in a document with an indicator variable. Specifically, if feature k appears in a given document represented by vector A_i, then:

$$A_{ik} = \begin{cases} 1 & \text{if feature } k \text{ appears in document } i \\ 0 & \text{otherwise} \end{cases} \tag{2.1}$$

The model represents each feature as a Bernoulli random variable with an assigned probability of occurrence. Every feature of every document vector is pairwise independent.

Let A_i represent a random document vector under the BIM model. That is, the random vector takes on values in $\{0, 1\}^m$ and is parameterized by a vector $\pi \in (0, 1]^m$ representing the probabilities of occurrence of each feature. Assume the half-open interval for probabilities as features that never occur can be removed from the model.

We refer to the parameters of this model, the vector π, as the "topic model", with the understanding that clusters of frequently co-occurring words (such as "oil", "drill", "well") can be interpreted semantically as a topic.

2.2.1 Expectations

Developments in subsequent chapters will depend on expectations of a random text corpus. The random variable representing the corpus will be denoted in bold: $\mathbf{A}$, while the instance of the random variable will be denoted as: A.

Consider first a single row vector of $\mathbf{A}$, denoted $\mathbf{A}_i$. Each element of the vector is independent and distributed as a Bernoulli. The topic model, or vector of probabilities for occurrence of each feature, for this document is π_i. Therefore, $\mathbb{E}[\mathbf{A}_i] = \pi_i$. Given that each random row vector of $\mathbf{A}$ is i.i.d. under the BIM model, each row of the expectation will be π_i, where i corresponds to the row index.

Further developments also depend on the expectation of the random variable represented by $\mathbb{E}\left[\mathbf{A}^T\mathbf{A}\right]$, which yields a simple structure that will be exploited later.

Lemma 1. *Under BIM, assume a k-topic model where rows of $\mathbf{A}$ are distributed according to one of k possible parameter vectors π. Let n_i represent the number of rows that follow topic model π_i. Then, $\mathbb{E}\left[\mathbf{A}^T\mathbf{A}\right]$ is a diagonal matrix plus a rank-k matrix. Specifically: $\mathbb{E}\left[\mathbf{A}^T\mathbf{A}\right] = \Phi + \sum_{i=1}^{k} n_i\pi_i\pi_i^T$ where Φ is diagonal, and $\Phi_{ii} = \sum_{i=1}^{k} n_i\pi_i(1-\pi_i)$.*

Proof. Consider first the element in row i and column j:

$$\mathbb{E}\left[\mathbf{A}^T\mathbf{A}\right]_{ij} = \mathbb{E}\left[\left(\sum_{l=1}^{n}\mathbf{A}_{li}\mathbf{A}_{lj}\right)\right] \tag{2.2}$$

Let $\mathbb{J}_l \subset [1, n]$ be the set of indexes corresponding to documents generated from topic model π_l. Note that the cardinality of $\mathbb{J}_l$, or $|\mathbb{J}_l|$, is n_l. If $i \neq j$, each entry in the sum becomes $\pi_{li}\pi_{lj}$, due to independence, so $\mathbb{E}\left[\mathbf{A}^T\mathbf{A}\right]_{ij} = \sum_{l=1}^{k} n_l\pi_{li}\pi_{lj}$.

For $i = j$, we are concerned with the random variable $\mathbf{A}_{ii}^2$. Since $\mathbf{A}_{ii}$ is binary valued, $\mathbf{A}_{ii}^2 \equiv \mathbf{A}_{ii}$. Therefore, assuming $i \in \mathbb{J}_i$, $\mathbb{E}[\mathbf{A}_{ii}^2] = \pi_{li}$. Let $x \odot y$ represent the element-wise product of two vectors: $x \odot y = \sum_{i=1}^{m} x_i y_i$, and let $\texttt{diag}\,(x)$ represent a diagonal matrix with vector x on the diagonal.

Adding and subtracting $\sum_{l=1}^{k} n_l\texttt{diag}\,(\pi_l \odot \pi_l)$ from the resulting matrix yields:

$$\mathbb{E}\left[\mathbf{A}^T\mathbf{A}\right] = \left(\sum_{l=1}^{k}\texttt{diag}\,(n_l\pi_l \odot (1-\pi_l)) + \sum_{l=1}^{k} n_l\pi_l\pi_l^T\right) \tag{2.3}$$

For ease of notation, let $\Phi = \left(\sum_{l=1}^{k} \texttt{diag}\left(n_l \pi_l \odot (1 - \pi_l)\right)\right)$.

Note that $\pi_{li}(1 - \pi_{li})$ is the variance of feature i with respect to topic model π_l.

2.3 Probabilistic Latent Semantic Indexing

Probabilistic Latent Semantic Indexing (pLSI) [19,30] incorporates the number of feature occurrences in each document into the model. That is, if feature j appears 8 times within document represented by vector A_i, then $A_{ij} = 8$.

As with the BIM model, each document vector generated by the pLSI model is assumed to be pairwise independent. The topic models are again parameterized by a vectors $\pi \in (0, 1]^m$, but with the additional requirement that $\mathbf{1}^T \pi = 1$. Thus, the parameter vector is constrained to the $(m - 1)$-simplex. In other words, each feature generated by this model is distributed as a categorical random variable.

This dissertation makes a small modification to the pLSI model: each row of the text corpus matrix is re-scaled to sum to 1, by dividing each document vector A_i by the number of words in the document, r_i. In this section, refer to this row-wise scaled matrix as $\tilde{A}$, though in later sections the normalization will be inferred and the matrix denoted A.

Once scaled in this fashion, this model yields a similar structure to the BIM model.

2.3.1 Expectations

Each random document vector A_i is assumed to be IID and distributed as a multinomial distribution with parameters $\pi \in (0, 1]^m$ representing the parameters of the multinomial and $r_i \in \mathbb{N}$ representing the number of words in the document.

$$\mathbb{E}\left[\frac{\mathbf{A}_i}{r_i}\right] = \frac{r_i \pi}{r_i} = \pi \tag{2.4}$$

Consider next $\tilde{A}^T \tilde{A}$.

Lemma 2. *Under pLSI, assume a k-topic model where rows of $\mathbf{A}$ are distributed according to one of k possible parameter vectors π. Let n_i represent the number of* *rows that follow topic model π_i. Then, $\mathbb{E}\left[\mathbf{A}^T\mathbf{A}\right]$ is a diagonal matrix plus a rank-k matrix. Specifically: $\mathbb{E}\left[\mathbf{A}^T\mathbf{A}\right] = \Phi + \sum_{i=1}^{k}(n_i - \rho_i)\pi_i\pi_i^T$ where Φ is diagonal, $\Phi_{ii} = \sum_{i=1}^{k}\rho_i\pi_i$, and $\rho_i = \sum_{i=1}^{k}\frac{1}{r_i}$.*

Proof. Let the parameter of the multinomial distribution generating document k be represented as π_k.

The off-diagonal entry of $\mathbb{E}\left[\tilde{A}^T\tilde{A}\right]$ is:

$$\mathbb{E}\left[\tilde{A}^T\tilde{A}\right]_{ij} = \sum_{l=1}^{n}\pi_{li}\pi_{lj} - \sum_{l=1}^{n}\frac{1}{r_l}\pi_{li}\pi_{lj} \tag{2.5}$$

Let $\mathbb{J}_i$ represent the set of indices of documents corresponding to topic i, and $\bigcup_{i=1}^{k}\mathbb{J}_i = [1, n]$. Let n_i represent the number of documents corresponding to topic i, or the cardinality of the index set: $n_i = |J_i|$.

Then,

$$\mathbb{E}\left[\tilde{A}^T\tilde{A}\right]_{ij} = \sum_{v=1}^{t}\left[n_v\pi_{vi}\pi_{vj} - \pi_{vi}\pi_{vj}\sum_{k\in J_v}\frac{1}{r_k}\right] \tag{2.6}$$

On-diagonal entries are:

$$\sum_{v=1}^{t}\left[\pi_{vi}\sum_{k\in J_v}\frac{1}{r_k} + n_v\pi_{vi}^2 - \pi_{vi}^2\sum_{k\in J_v}\frac{1}{r_k}\right] \tag{2.7}$$

Let $\rho_v = \sum_{k\in J_v}\frac{1}{r_k}$. Combining into a single vector expression yields:

$$\mathbb{E}\left[\tilde{A}^T\tilde{A}\right] = \sum_{v=1}^{t}\left[\mathtt{diag}\left(\pi_v\right)\rho_v + (n_v - \rho_v)\pi_v\pi_v^T\right] \tag{2.8}$$

Note that the result is a diagonal plus rank-k matrix.

2.4 Unified Form of BIM and pLSI

The structures of the expectations A^TA under the BIM and (row-wise normalized) pLSI models are both low-rank plus a diagonal matrix. This simple representation requires only the parameter vectors π to be estimated and stored, a great dimensionality reduction from the raw $n \times m$ data matrix. In Chapter 3, this representation of the data will be leveraged to derive fast solutions to regression problems.

Note that the forms of $\mathbb{E}\left[\mathbf{A}^T\mathbf{A}\right]$ are similar enough to combine into a common formulation. Recall that we have used a row-wise normalized matrix in the pLSI model, previously denoted as $\tilde{\mathbf{A}}$. In this section, we simply refer to this matrix as $\mathbf{A}$.

Lemma 3. *Under a k-topic BIM or pLSI, $\mathbb{E}\left[\mathbf{A}^T\mathbf{A}\right] = \Phi + \sum_{i=1}^{k} c_i \pi_i \pi_i^T$, where $c_i = n_i$ under BIM or $(n_i - \sum_{k \in \mathbb{J}_i} \frac{1}{r_k})$ under pLSI. Further, $\Phi_{ii} = \sum_{l=1}^{k} n_l \pi_{li}(1 - \pi_{li})$ under BIM or $\sum_{l=1}^{k} \rho_l \pi_l$ under pLSI. This may also be represented as $\Phi + \Pi\Pi^T$ where $\Pi \in \mathbb{R}^{m \times k}$, and where column i is $\sqrt{c_i}\pi_i$.*

Proof. For both models:

$$\mathbb{E}\left[\mathbf{A}\right]_i = \pi_i^T \tag{2.9}$$

where π_i is the parameter vector for the topic that generated row i.

Further,

$$\mathbb{E}\left[\mathbf{A}^T\mathbf{A}\right] = \sum_{k=1}^{t} \check{\Phi}_k + c_k \pi_k \pi_k^T \tag{2.10}$$

where $\check{\Phi}_k$ is diagonal, and c_k is a constant.

An alternative form is:

$$\mathbb{E}\left[\mathbf{A}^T\mathbf{A}\right] = \Phi + \Pi\Pi^T \tag{2.11}$$

where $\Phi = \sum_{k=1}^{t} \check{\Phi}_k$ and $\Pi \in \mathbb{R}^{m \times t}$ is defined such that column j is $\sqrt{c_j}\pi_j$.

$$\Phi_{ii} = \begin{cases} \sum_{k=1}^{t} n_k \pi_{ki}(1 - \pi_{ki}) & \text{BIM} \\ \sum_{k=1}^{t} \rho_k \pi_{ki} & \text{pLSI} \end{cases} \tag{2.12}$$

$$c_k = \begin{cases} n_k & \text{BIM} \\ n_k - \rho_k & \text{pLSI} \end{cases} \tag{2.13}$$

It is important for the subsequent chapters to establish that $\Phi + \Pi\Pi^T$ is a positive-definite matrix.

Lemma 4. $\Phi + \sum_{k=1}^{T} c_k \pi_k \pi_k^T \succ 0$

Proof. Symmetry is apparent, as each matrix in the sum is symmetric. Next, consider a vector $v \in \mathbb{R}^m$ such that $v \neq 0$. Then:

$$v^T \Phi v + \sum_{k=1}^{T} c_k (\pi_k^T v)^2 = \sum_{i=1}^{m} \Phi_{ii} v_i^2 + \sum_{k=1}^{T} c_k (\pi_k^T v)^2 > 0 \tag{2.14}$$

This holds due to the following: In both BIM and pLSI, each diagonal entry of Φ is strictly > 0, so $v^T \Phi v > 0$. Note that $\rho_k = \sum_{j \in J_k} \frac{1}{r_j} \leq n_k$, so in both BIM and pLSI, $c_k \geq 0$.

Therefore, $\Phi + \sum_{k=1}^{T} c_k \pi_k \pi_k^T \succ 0$. □

2.4.1 Feature Scaling

Some applications require features to be scaled by importance relative to a user query, past results, or by a property of frequency within a dataset. A common method to scale features is by a TF-IDF [19, 26] transformation which is dependent on both the frequency of a feature within a document and the overall frequency of the feature in a corpus. Here, we consider a form of scaling where feature j is scaled by a constant γ_j within every document. This is achieved with multiplication by a diagonal matrix $\Gamma \in \mathbb{R}^{m \times m}$, where $\Gamma_{jj} = \gamma_j$:

$$A' = A\Gamma \tag{2.15}$$

The following demonstrates that the low-rank plus diagonal matrix structure is maintained and that the modification to $\mathbb{E}[A]$ and $\mathbb{E}\left[A^T A\right]$ are both trivial to compute. Further, it is trivial to scale features independently for distinct document classes. This opens up a lot of possibilities for enhancing text analysis results and for opening up opportunities for user interaction.

These results precipitate from linearity of expectation.

For instance, consider $\mathbb{E}[A']$:

$$\mathbb{E}[A\Gamma]_i = \mathbb{E}[A]_i \Gamma = \pi_i^T \Gamma \tag{2.16}$$

The result is a scaled version of the original expectation, where each parameter in π_i is scaled by γ_i.

Consider $\mathbb{E}\left[A'^T A\right]$, in a 2-class setting:

$$\begin{aligned}\mathbb{E}\left[\Gamma A^T A \Gamma\right] &= \Gamma \mathbb{E}\left[A^T A\right]\Gamma && (2.17)\\ &= \Gamma\Phi\Gamma + n_1\Gamma\pi_1\pi_1^T\Gamma + n_2\Gamma\pi_2\pi_2^T\Gamma && (2.18)\end{aligned}$$

The resulting form is exactly the same, assuming $\Gamma \succeq 0$. The result is computed by multiplying Φ_i by γ_i^2 for $i \in [1, m]$ and multiplying π_{1i} and π_{2i} by γ_i.

Consider now independent scaling of features from two classes. Let

$$A = \begin{bmatrix} A_1 \in \mathbb{R}^{n_1 \times m} \\ A_2 \in \mathbb{R}^{n_2 \times m} \end{bmatrix} \qquad (2.19)$$

where A_1 represents a sub-matrix of documents all in one class, and A_2 representing documents from a second distinct class. Let Γ_1 represent a diagonal feature scaling matrix for A_1 and Γ_2 for A_2. Let:

$$B = \begin{bmatrix} A_1\Gamma_1 \\ A_2\Gamma_2 \end{bmatrix} \qquad (2.20)$$

Note that:

$$\mathbb{E}\left[B^T B\right] = \Gamma_1\mathbb{E}\left[A_1^T A_1\right]\Gamma_1 + \Gamma_2\mathbb{E}\left[A_2^T A_2\right]\Gamma_2 \qquad (2.21)$$

Therefore, scaling of features within each class may be computed independently without any impact on the structure of the expected matrices.

Chapter 3

Robust Regression

The previously introduced probabilistic text corpus models can be applied to the problem of linear regression or classification, where the low-rank plus diagonal structure derived in the previous chapter yields computationally efficient and sparse solutions.

In this chapter, a robust regression methodology is employed to take advantage of an assumed probabilistic structure of a data matrix. Further, we show how these solutions may be made sparse without the need for regularizations as is required for the Lasso and similar problems [12, 40, 46].

3.1 Robust Regression

Consider the linear regression problem:

$$\arg\min_x \|Ax - b\|_2 \tag{3.1}$$

when $A \in \mathbb{R}^{n\times m}$ is a random matrix following either the Binary Independence Model (BIM) or the probabilistic Latent Semantic Indexing (pLSI) model.

Assume a two-topic model, where the first n_1 rows of A are drawn from one topic model parameterized by $\pi_1 \in (0,1]^m$ and the remaining n_2 rows are drawn from a second topic model with parameter $\pi_2 \in (0,1]^m$.

Rather than solving the regression problem directly for a given realization of the random corpus, we instead consider the problem of minimizing the probability of large deviation, using a bound closely related to the Chebyshev inequality and derived from Cramér [8]:

$$\mathbf{P}\left(\|Ax-b\|_2 \geq \epsilon\right) \leq \frac{\mathbb{E}\left[\|Ax-b\|_2^2\right]}{\epsilon^2} \tag{3.2}$$

Minimizing this upper bound yields a robust solution to the classification problem that does not rely on the raw observations and instead operates on the underlying structure and statistics of the data. So we arrive at the problem:

$$\arg\min_x \mathbb{E}\left[\|Ax-b\|_2^2\right] = \arg\min_x x^T\Phi x + x^T\Pi\Pi^T x - 2b^T\mathbb{E}\left[A\right]x + b^T b \tag{3.3}$$

The constant b^Tb term is irrelevant for the arg min optimization problem, so it is omitted in the following.

For now, we assume that b is a classification vector and oracle information is provided to correctly identify those documents following distribution π_1 or π_2. The goal is to identify the most important features of the corpus in discriminating between the two classes of documents.

Define b such that the first n_1 entries are some value $\frac{\alpha}{n_1}$ where $\alpha \in \mathbb{R}$, and the remaining n_2 entries are some other value $\frac{\beta}{n_2}$ where $\beta \in \mathbb{R}$.

Thus, the problem becomes:

$$\boxed{\arg\min_x x^T(\Phi + \Pi\Pi^T)x - 2\alpha\pi_1^T x - 2\beta\pi_2^T x} \tag{3.4}$$

3.2 Unconstrained Solution

Let $f(x) = x^T\Phi x + c_1(\pi_1^T x)^2 + c_2(\pi_2^T x)^2 - 2\alpha\pi_1^T x - 2\beta\pi_2^T x$. Note that this is quadratic in x as $(\Phi + c_1\pi_1\pi_1^T + c_2\pi_2\pi_2^T)$ is positive definite, as established in Lemma 4.

Calculating the gradient yields:

$$\nabla_x f(x) = 2\Phi x + 2c_1\pi_1\pi_1^T x + 2c_2\pi_2\pi_2^T x - 2\alpha\pi_1 - 2\beta\pi_2 \tag{3.5}$$

Because $(\Phi + \Pi\Pi^T)$ is positive definite, a solution can be achieved with a simple matrix inversion.

$$\overset{\star}{x} = (\Phi + \Pi\Pi^T)^{-1}(\alpha\pi_1 + \beta\pi_2) \tag{3.6}$$

The matrix inversion lemma [44] allows the inversion of $\Phi + \Pi\Pi^T$ to be calculated as the inverse of a 2×2 matrix, yielding a simple closed-form solution to the problem after parameter vectors π_1 and π_2 have been estimated.

By the Matrix Inversion Lemma:

$$(\Phi + \Pi\Pi^T)^{-1} = \Phi^{-1} - \Phi^{-1}\Pi\left(I + \Pi^T\Phi^{-1}\Pi\right)^{-1}\Pi^T\Phi^{-1} \tag{3.7}$$

As Φ is diagonal, its inverse is trivial to calculate. The remaining matrix requiring inversion is $I + \Pi^T\Phi^{-1}\Pi$, which is 2×2.

A recurring element in the calculation is the matrix product $\Phi^{-1}\Pi \in \mathbb{R}^{m\times 2}$, which is:

$$\Phi^{-1}\Pi = \left[\begin{array}{cc} \sqrt{c_1}\Phi^{-1}\pi_1 & \sqrt{c_2}\Phi^{-1}\pi_2 \end{array}\right] \tag{3.8}$$

where each entry above represents a column of the matrix.

The 2×2 matrix to invert, which we define as G is:

$$G \equiv I + \Pi^T\Phi^{-1}\Pi = \left[\begin{array}{cc} 1 + c_1\pi_1^T\Phi^{-1}\pi_1 & \sqrt{c_1c_2}\pi_2^T\Phi^{-1}\pi_1 \\ \sqrt{c_1c_2}\pi_2^T\Phi^{-1}\pi_1 & 1 + c_2\pi_2^T\Phi^{-1}\pi_2 \end{array}\right] \tag{3.9}$$

The determinant of G is:

$$\begin{aligned} \det G &= 1 + c_1\pi_1^T\Phi^{-1}\pi_1 + c_2\pi_2^T\Phi^{-1}\pi_2 + \\ &\quad c_1c_2\pi_1^T\Phi^{-1}\pi_1\pi_2^T\Phi^{-1}\pi_2 - c_1c_2(\pi_1^T\Phi^{-1}\pi_2)^2 \end{aligned} \tag{3.10}$$

Thus, G^{-1} is:

$$G^{-1} = \frac{1}{\det G}\left[\begin{array}{cc} 1 + c_2\pi_2^T\Phi^{-1}\pi_2 & -\sqrt{c_1c_2}\pi_2^T\Phi^{-1}\pi_1 \\ -\sqrt{c_1c_2}\pi_2^T\Phi^{-1}\pi_1 & 1 + c_1\pi_1^T\Phi^{-1}\pi_1 \end{array}\right] \tag{3.11}$$

Let $y \equiv \alpha\pi_1 + \beta\pi_2$.

$$G^{-1}\Pi^T\Phi^{-1}y = \frac{1}{\det G}\left[\begin{array}{c} (1 + c_2\pi_2^T\Phi^{-1}\pi_2)\sqrt{c_1}\pi_1^T\Phi^{-1}y - c_2\sqrt{c_1}\pi_2^T\Phi^{-1}\pi_1\pi_2^T\Phi^{-1}y \\ (1 + c_1\pi_1^T\Phi^{-1}\pi_1)\sqrt{c_2}\pi_2^T\Phi^{-1}y - c_1\sqrt{c_2}\pi_2^T\Phi^{-1}\pi_1\pi_1^T\Phi^{-1}y \end{array}\right] \tag{3.12}$$

This is a vector in $\mathbb{R}^2$. Let:

$$G^{-1}\Pi^T\Phi^{-1}y \equiv \left[\begin{array}{c} a \\ b \end{array}\right] \tag{3.13}$$

Therefore,

$$\begin{aligned} \overset{\star}{x} &= \Phi^{-1}y - (a\sqrt{c_1}\Phi^{-1}\pi_1 + b\sqrt{c_2}\Phi^{-1}\pi_2) && (3.14) \\ &= \Phi^{-1}\left((\alpha - a\sqrt{c_1})\pi_1 + (\beta - b\sqrt{c_2})\pi_2\right) && (3.15) \end{aligned}$$

In this case, a closed-form solution exists, once parameters π_1 and π_2 have been estimated.

3.3 Non-negative Constrained Solution

Some solutions seek to identify only features which *positively* identify a target class with respect to another class, so we introduce a non-negativity constraint. We discover that the solution to this problem tends to be sparse, without the need for regularization.

In this section, consider the problem:

$$\arg\min_{x \succeq 0} x^T\Phi x + c_1(\pi_1^T x)^2 + c_2(\pi_2^T x)^2 - 2\alpha\pi_1^T x - 2\beta\pi_2^T x \tag{3.16}$$

3.3.1 Introduction of helper variables

Let $z_1 = \pi_1^T x$ and $z_2 = \pi_2^T x$. Let

$$f(x, z) = x^T\Phi x + c_1 z_1^2 + c_2 z_2^2 - 2\alpha\pi_1^T x - 2\beta\pi_2^T x \tag{3.17}$$

We selectively replace the $\pi_*^T x$ expressions from the function to simplify the KKT conditions later.

With the introduced equality constraints, the Lagrangian is:

$$L(x, z, \lambda, \nu) = x^T \Phi x + c_1 z_1^2 + c_2 z_2^2 - 2\alpha\pi_1^T x - 2\beta\pi_2^T x - \lambda^T x - \nu_1(\pi_1^T x - z_1) - \nu_2(\pi_2^T x - z_2) \tag{3.18}$$

3.3.2 The Dual Problem

Taking the gradient with respect to x yields:

$$\nabla_x L = 2\Phi x - 2\alpha\pi_1 - 2\beta\pi_2 - \lambda - \nu_1\pi_1 - \nu_2\pi_2 \tag{3.19}$$

Therefore, we find a solution for λ:

$$\overset{\star}{\lambda} = 2\Phi x - (2\alpha + \nu_1)\pi_1 - (2\beta + \nu_2)\pi_2 \tag{3.20}$$

By the KKT conditions, $\lambda_i \geq 0 \ \forall i$. Therefore, we have the requirement that $2\Phi_i x_i - (2\alpha + \nu_1)\pi_{1i} - (2\beta + \nu_2)\pi_{2i} \geq 0$, or:

$$\boxed{x_i \geq \frac{(2\alpha + \nu_1)\pi_{1i} + (2\beta + \nu_2)\pi_{2i}}{2\Phi_i}} \tag{3.21}$$

Further, by the KKT conditions, $\lambda_i x_i = 0 \ \forall i$. Therefore, for all i:

$$2\Phi_i x_i^2 - (2\alpha + \nu_1)\pi_{1i} x_i - (2\beta + \nu_2)\pi_{2i} x_i = 0 \tag{3.22}$$

Suppose $x_i > 0$:

$$(2\alpha + \nu_1)\pi_{1i} + (2\beta + \nu_2)\pi_{2i} = 2\Phi_i x_i \tag{3.23}$$

and

$$\overset{\star}{x}_i = \frac{(2\alpha + \nu_1)\pi_{1i} + (2\beta + \nu_2)\pi_{2i}}{2\Phi_i} \tag{3.24}$$

If the RHS ≤ 0, then we have a contradiction, implying that $\overset{\star}{x}_i = 0$. Therefore,

$$\boxed{\overset{\star}{x}_i = \frac{\max\left(0, (2\alpha + \nu_1)\pi_{1i} + (2\beta + \nu_2)\pi_{2i}\right)}{2\Phi_i}} \tag{3.25}$$

We take a step back now to consider $\overset{\star}{z}_1$ and $\overset{\star}{z}_2$.

$$\frac{\partial L(x,z,\lambda,\nu)}{\partial z_1} = 2c_1 z_1 + \nu_1 \tag{3.26}$$

So, $\overset{\star}{z}_1 = -\frac{\nu_1}{2c_1}$. Similar reasoning yields $\overset{\star}{z}_2 = -\frac{\nu_2}{2c_2}$.

Due to the non-negativity constraint, $\pi_1^T x \geq 0$, so $z_1 \geq 0$. Therefore, $-\frac{\nu_1}{2c_1} \geq 0$, or $\nu_1 \leq 0$. The same reasoning applies for ν_2.

Now, we plug in our optimal values to derive $g(\nu)$ for our dual problem.

$$L(x,z,\overset{\star}{\lambda},\nu) = -x^T\Phi x + c_1 z_1^2 + c_2 z_2^2 + \nu_1 z_1 + \nu_2 z_2 \tag{3.27}$$

$$\begin{aligned} L(x,\overset{\star}{z},\overset{\star}{\lambda},\nu) &= -x^T\Phi x + \frac{\nu_1^2}{4c_1} - \frac{\nu_1^2}{2c_1} + \frac{\nu_2^2}{4c_2} - \frac{\nu_2^2}{2c_2} && (3.28) \\ &= -x^T\Phi x - \frac{\nu_1^2}{4c_1} - \frac{\nu_2^2}{4c_2} && (3.29) \end{aligned}$$

$$L(\overset{\star}{x},\overset{\star}{z},\overset{\star}{\lambda},\nu) = g(\nu) = -\sum_{i=1}^{m} \frac{\max\left(0,(2\alpha+\nu_1)\pi_{1i} + (2\beta+\nu_2)\pi_{2i}\right)\right)^2}{4\Phi_i} - \frac{\nu_1^2}{4c_1} - \frac{\nu_2^2}{4c_2} \tag{3.30}$$

Let $w_1 = 2\alpha + v_1$ and $w_2 = 2\beta + v_2$. With these variables, the dual problem becomes:

$$g(w) = -\sum_{i=1}^{m} \frac{\max(0, w_1\pi_{1i} + w_2\pi_{2i})^2}{4\Phi_i} - \frac{(w_1-2\alpha)^2}{4c_1} - \frac{(w_2-2\beta)^2}{4c_2} \tag{3.31}$$

Consider the solution for a few different cases. First, assume $w_1,\ w_2 < 0$. In this case, $\overset{\star}{x} \equiv 0$, a trivial solution.

In the next case, consider $w_1,\ w_2 > 0$. In this case, all elements of the sum are non-zero:

$$g(w) = -\sum_{i=1}^{m} \frac{(w_1\pi_{1i} + w_2\pi_{2i})^2}{4\Phi_i} - \frac{(w_1-2\alpha)^2}{4c_1} - \frac{(w_2-2\beta)^2}{4c_2} \tag{3.32}$$

For solutions in this first quadrant, let $w_2 = \eta w_1$ for $\eta > 0$.

$$g(w_1,\eta) = -w_1^2\sum_{i=1}^{m} \frac{(\pi_{1i} + \eta\pi_{2i})^2}{4\Phi_i} - \frac{(w_1-2\alpha)^2}{4c_1} - \frac{(\eta w_1-2\beta)^2}{4c_2} \tag{3.33}$$

Consider the partial derivative with respect to η:

$$\frac{\partial g(w_1, \eta)}{\partial \eta} = -w_1^2 \sum_{i=1}^{m} \left(\frac{\pi_{1i}\pi_{2i} + \eta\pi_{2i}^2}{2\Phi_i} \right) - \frac{\eta w_1^2}{2c_2} + \frac{\beta w_1}{2c_2} \tag{3.34}$$

Therefore,

$$\overset{\star}{\eta} w_1 = \frac{-w_1 \sum_{j=1}^{m} \frac{\pi_{1i}\pi_{2i}}{2\Phi_i} + \beta/c_2}{\sum_{j=1}^{m} \frac{\pi_{2i}^2}{2\Phi_i} + 1/(2c_2)} \tag{3.35}$$

To simplify notation, let:

$$\overline{\eta} = \frac{\sum_{j=1}^{m} \frac{\pi_{1i}\pi_{2i}}{2\Phi_i}}{\sum_{j=1}^{m} \frac{\pi_{2i}^2}{2\Phi_i} + \frac{1}{2c_2}} \tag{3.36}$$

Note that:

$$\frac{\partial \overset{\star}{\eta} w_1}{\partial w_1} = -w_1 \overline{\eta} \tag{3.37}$$

With this established, consider the partial derivative with respect to w_1:

$$\frac{\partial g(w_1, \overset{\star}{\eta})}{\partial w_1} = \sum_{i=1}^{m} \frac{(w_1\pi_{1i} + \overset{\star}{\eta} w_1 \pi_{2i})(\pi_{1i} - \overline{\eta})}{2\Phi_i} - \frac{w_1 - 2\alpha}{2c_1} + \overline{\eta}\frac{\overset{\star}{\eta} w_1 - 2\beta}{2c_2} \tag{3.38}$$

This derivative is linear in w_1, yielding a straight-forward solution by solving for w_1 after equating the derivative with 0. If the solution requires $w_1 \leq 0$, then a contradiction arises and the first quadrant can be ruled out for a solution. Further, consider $\overset{\star}{\eta} w_1$; if $\beta \leq 0$, then $\overset{\star}{\eta} w_1 \leq 0$, which also raises a contradiction, ruling out the first quadrant. If, for example, we are comparing a positive and negative class, and $\alpha = 1$ and $\beta = -1$, then the first quadrant is infeasible.

Next, consider the case where $w_1 > 0$ and $w_2 < 0$.

In this case, let $w_2 = -\eta w_1$ for $\eta > 0$. Consider one term in the summation:

$$\max(0, w_1\pi_{1i} - \eta w_1 \pi_{2i})^2 \tag{3.39}$$

Sparsity emerges with a likelihood ratio test, where η sets the threshold level:

$$\frac{\pi_{1i}}{\pi_{2i}} > \eta \;\; \Rightarrow \;\; \overset{\star}{x}_i > 0 \tag{3.40}$$

$$\frac{\pi_{1i}}{\pi_{2i}} \leq \eta \;\; \Rightarrow \;\; \overset{\star}{x}_i = 0 \tag{3.41}$$

If all that is desired is a list of the k features represented in a k-sparse solution vector, then all that is required is to sort features based on $\frac{\pi_{1i}}{\pi_{2i}}$, keeping the largest k features. The computation is carried out via estimation and sorting, which can be solved in time $O(nm\log(m))$.

Consider now the full solution. Let $J \subseteq i \in [1, m]$ be the set of indices where $\overset{\star}{x}_i > 0$, referred to as the activation set.

$$g(w_1, \eta) = -w_1^2 \sum_J \frac{(\pi_{1i} - \eta\pi_{2i})^2}{4\Phi_i} - \frac{(w_1 - 2\alpha)^2}{4c_1} - \frac{(\eta w_1 + 2\beta)^2}{4c_2} \tag{3.42}$$

$$\frac{\partial g(w_1, \eta)}{\partial \eta} = w_1 \sum_J \frac{(\pi_{1i} - \eta\pi_{2i})\pi_{2i}}{2\Phi_i} - \frac{\eta w_1 + 2\beta}{2c_2} \tag{3.43}$$

Solving for $\overset{\star}{\eta}$:

$$\overset{\star}{\eta} w_1 = \frac{w_1(\sum_{i\in J}\frac{\pi_{1i}\pi_{2i}}{2\Phi_i}) - \frac{\beta}{c_2}}{(\sum_{i\in J}\frac{\pi_{2i}^2}{2\Phi_i}) + \frac{1}{c_2}} \tag{3.44}$$

Note that the derivative of this with respect to w_1 takes the following form:

$$\frac{\partial(\overset{\star}{\eta} w_1)}{\partial w_1} = \frac{\sum_{i\in J}\frac{\pi_{1i}\pi_{2i}}{2\Phi_i}}{\sum_{i\in J}\frac{\pi_{2i}^2}{2\Phi_i} + \frac{1}{c_2}} \equiv h \tag{3.45}$$

Now consider

$$\frac{\partial g(w_1, \overset{\star}{\eta})}{\partial w_1} = \sum_{i\in J} \frac{(w_1\pi_{1i} - \overset{\star}{\eta} w_1\pi_{2i})(\pi_{1i} - h\pi_{2i})}{2\Phi_i} - \frac{w_1 - 2\alpha}{2c_1} - \frac{h(\overset{\star}{\eta} w_1 + 2\beta)}{2c_2} \tag{3.46}$$

Equating to 0 and solving for $\overset{\star}{w}_1$ yields a possible solution, given the activation set J. We can check if this is a valid solution by evaluating $\overset{\star}{\eta}$ with this value of $\overset{\star}{w}_1$. Given $\overset{\star}{\eta}$, we can calculate $\hat{J}$, the set of indices of $\overset{\star}{x}$ such that $x_i > 0$. If $\hat{J} \neq J$, then this possible solution is incorrect. At most, we must test m solutions: as η decreases, new features x_i are activated, and no currently active features will drop out of the solution.

Chapter 4

Dealing With Uncertainty

The preceding work assumed that the parameters of each probability distribution are known. In practice, these parameters must be estimated, with uncertainty surrounding the value of each parameter. This chapter describes methods to incorporate such uncertainty into the robust regression problem discussed in the previous chapter.

Generally, upon estimation of model parameters, we have a measure of uncertainty about the estimated parameters. This dissertation employs bounded confidence intervals to represent uncertainty around a parameter. Many methods are available for such an estimate for parameters of both BIM and pLSI [2,10,14,23,24]. This section focuses on methods by which any selected confidence interval may be integrated into the robust regression problem.

First, consider the feature selection portion of the regression problem. As discussed in Chapter 3, feature selection is solved with a likelihood ratio threshold. For use in a human-facing tool, it is desirable to limit false positives; spurious features appearing in the solution may lead to confusion or erroneous interpretations of the underlying data. The robust approach requires features to be selected only with strong and convincing evidence that they are indeed strongly associated with one set of documents relative to another. This need becomes apparent when the regression problem is unbalanced, meaning that there are many more observations in one class than in another. The statistics perspective shows that the confidence intervals for the parameters of the class with few observations will be much wider than those for the class with many observations. An estimate that doesn't take this uncertainty into account may encounter issues with spurious feature selection. This effect is demonstrated empirically in Chapter 6.

4.1 Robust Estimate of BIM Parameters

Unlike in the pLSI model, the parameters for each feature in the BIM model may be determined independently. When deciding on features for a solution to the non-negative constrained robust regression problem, the conservative estimate would be to ensure that the *minimum* likelihood ratio of the parameters within the confidence bounds exceeds the selected likelihood ratio threshold. Stated as an optimization problem:

$$\min_{\pi_{1i},\pi_{2i}} \frac{\pi_{1i}}{\pi_{2i}} \tag{4.1}$$

$$\text{s.t.} \quad \breve{\pi}_{1i} \leq \pi_{1i} \leq \hat{\pi}_{1i} \tag{4.2}$$

$$\breve{\pi}_{2i} \leq \pi_{2i} \leq \hat{\pi}_{2i} \tag{4.3}$$

The solution is immediate:

$$\frac{\overset{\star}{\pi}_{1i}}{\overset{\star}{\pi}_{2i}} = \frac{\breve{\pi}_{1i}}{\hat{\pi}_{2i}} \tag{4.4}$$

4.2 Robust Estimates of pLSI Parameters

The pLSI model requires more care. Note that exact confidence intervals are more challenging to estimate [15,41]. To proceed, confidence intervals are determined independently for each feature in the dataset in precisely the same fashion as for the BIM model. However, in the optimization problem, an additional constraint is added, so any estimated parameter vector must both satisfy the confidence interval box constraints and the simplex constraint: $\vec{1}^T\pi = 1$.

4.2.1 Maximum Entropy

Consider the maximum entropy estimate of a categorical distribution under box constraints. This approach is motivated by the principal of maximum entropy, or a mathematical interpretation of Occam's razor [39], an estimation approach previously applied to text modeling. Further, by computing maximum entropy distribution for both parameters π_1 and π_2, the estimates are both being pulled towards the same point in the parameter space, reducing their contrast.

Let $\log x$ be defined such that $(\log x)_i = \log x_i \ \ \forall i$. The maximum entropy problem is stated:

$$\begin{aligned} \arg\max_{\pi} \ & -\pi^T \log \pi && (4.5) \\ \text{s.t.} \ & \vec{1}^T \pi = 1 && (4.6) \\ & \breve{\pi} \preceq \pi \preceq \hat{\pi} && (4.7) \end{aligned}$$

The Lagrangian is:

$$L(\pi, \lambda_1, \lambda_2, \nu) = \pi^T \log \pi - \lambda_1^T (\pi - \breve{\pi}) - \lambda_2^T (\hat{\pi} - \pi) - \nu(\vec{1}^T \pi - 1) \tag{4.8}$$

The gradient with respect to π is:

$$\nabla_\pi L = -(\nu - 1)\vec{1} + \log \pi - \lambda_1 + \lambda_2 \tag{4.9}$$

λ_1 may be interpreted as a slack variable: $\overset{\star}{\lambda_1} = -(\nu - 1)\vec{1} + \log \pi + \lambda_2$.

Further, by the K.K.T. conditions [5], $\lambda_{1i}(\pi_i - \breve{\pi}_i) = 0$ and $\lambda_{2i}(\hat{\pi}_i - \pi_i) = 0$ for all i. There are three cases to consider in regards to the box constraints:

i $\lambda_{1i} = 0$, $\lambda_{2i} > 0$

ii $\lambda_{1i} > 0$, $\lambda_{2i} = 0$

iii $\lambda_{1i} = 0$, $\lambda_{2i} = 0$

Both λ_{1i} and λ_{2i} cannot be > 0 at the same time as the upper bound constraint and lower bound constraint cannot both be active.

Consider possible values for π_i in each of these cases:

i $\overset{\star}{\pi}_i = \hat{\pi}_i$, as the upper bound constraint is active. By plugging in for λ_{1i}:

$$-\nu + 1 + \log \hat{\pi}_i + \lambda_{2i} = 0 \tag{4.10}$$

Since $\lambda_{2i} > 0$, $-\nu + 1 + \log \hat{\pi}_i < 0$. Therefore, $\hat{\pi}_i < e^{\nu-1}$.

ii $\overset{\star}{\pi}_i = \breve{\pi}_i$, as the lower bound constraint is active. As $\lambda_{1i} > 0$, $-\nu + 1 + \log \pi_i > 0$, implying $\breve{\pi}_i > e^{\nu-1}$

iii Niether bound is active in this case, and the K.K.T. conditions yield the equality: $-\nu + 1 + \log \pi_i = 0$, so $\overset{\star}{\pi}_i = e^{\nu-1}$

These conditions imply a simple optimization scheme in one variable: ν. If $e^{\nu-1} < \breve{\pi}_i$, then $\overset{\star}{\pi}_i = \breve{\pi}_i$. If $e^{\nu-1} > \hat{\pi}_i$, then $\overset{\star}{\pi}_i = \hat{\pi}_i$. Otherwise, $\overset{\star}{\pi}_i = e^{\nu-1}$.

To solve the problem, we must find the value of ν such that $\vec{1}^T \overset{\star}{\pi}(\nu) = 1$ to satisfy the final constraint. Note that as a function of ν, $\vec{1}^T \overset{\star}{\pi}(\nu)$ is monotonically increasing. The solution may be uncovered by choosing an initial ν such that $e^{\nu-1} = \arg\min_i \breve{\pi}_i$. The initial value is the minimum lower bound. The resulting value of $\overset{\star}{\pi}(\nu) = \breve{\pi}$. If $\vec{1}^T \overset{\star}{\pi}(\nu) < 1$, ν is increased until the parameter vector meets the constraint. Note that it is assumed that $\vec{1}^T \breve{\pi} \leq 1$, otherwise the problem would be infeasible.

This problem can be solved efficiently using bisection in $O(\log m)$ time. as follows:

1. Combine the values of each upper and lower bound in a sorted list of length $2m$, called v. Set $i = m$.

2. Choose ν such that $e^{\nu-1} = v_i$

3. Set $l = 1$, $u = 2m$ representing upper and lower bounds on the solution.

4. Choose a candidate solution π such that each π_j is as close to v_i as bounds will permit.

5. Compute $\vec{1}^T \pi$. If < 1, set $l = i$. If > 1, set $u = i$. Update $i = \lfloor (u - l)/2 \rfloor$. If $u - l \leq 1$, then the solution lies in the continuous interval between v_u and v_l, and can be solved analytically. Otherwise, return to Step 2. In each iteration, half of the bounds in list v are eliminated.

In the following algorithm, let $\eta = e^{\nu-1}$.

Algorithm 1 Maximum Entropy Solution for π Under Box Uncertainty Constraints

```
procedure MaxEnt($\breve{\pi}, \hat{\pi}$)
    $v \in \mathbb{R}^{2m} \leftarrow$ sorted($\breve{\pi}, \hat{\pi}$)        ▷ Sort bounds in a single list
    $l \leftarrow 1$
    $u \leftarrow 2m$        ▷ Initialize upper and lower bounds on solution for $\nu$
    while $u - l \geq 1$ do
        $i \leftarrow \lfloor \frac{u-l}{2} \rfloor$
        $\eta \leftarrow v_i$
        for all $j \in [1, m]$ do $\pi_j \leftarrow \begin{cases} \hat{\pi}_j & \text{if } \eta \geq \hat{\pi}_j \\ \breve{\pi}_j & \text{if } \eta \leq \breve{\pi}_j \\ \eta & \text{otherwise} \end{cases}$
        if $\vec{1}^T \pi < 1$ then $u \leftarrow i$
        else $l \leftarrow i$
    $\pi \leftarrow$ solution in interval between $u$ and $l$
```

4.2.2 Adversarial Model

An adversarial perspective on the problem allows for the parameters π_1 and π_2 to always be the worst-case values for the value of x chosen in the original problem. Consider the robust regression problem with confidence intervals as box constraints and a second optimization over parameters π_1 and π_2:

$$\min_{x} \max_{\pi_1, \pi_2} x^T \Phi x + c_1(\pi_1^T x)^2 + c_2(\pi_2^T x)^2 - 2\alpha \pi_1^T x - 2\beta \pi_2^T x \tag{4.11}$$

$$\text{s.t. } \breve{\pi}_1 \preceq \pi_1 \preceq \hat{\pi}_1 \tag{4.12}$$

$$\breve{\pi}_2 \preceq \pi_2 \preceq \hat{\pi}_2 \tag{4.13}$$

The simple formulation of the diagonal matrix Φ within the pLSI model allows the problem to be analyzed relatively simply.

Consider first the sub-problem of maximization with respect to π, as solutions for π_1 and π_2 may be considered independently. As such, subscripts are omitted in the following for ease of notation. The optimization problem with respect to π is:

$$\begin{aligned} \max_{\pi} \quad & \rho x^T \texttt{diag}(\pi)\, x + c(\pi^T x)^2 - 2\alpha \pi^T x && (4.14)\\ \text{subject to} \quad & \breve{\pi} \preceq \pi \preceq \hat{\pi} && (4.15)\\ & \mathbf{1}^T \pi = 1 && (4.16) \end{aligned}$$

We approach the solution algorithmically. Begin with a candidate vector $\pi = \breve{\pi}$. It is assumed that $\mathbf{1}^T \breve{\pi} \leq 1$, otherwise the problem is infeasible. Values of individual components of π may be increased, though there is a fixed "budget", requiring $\mathbf{1}^T \pi = 1$.

Let $f(\pi) = \rho x^T \texttt{diag}(\pi)\, x + c(\pi^T x)^2 - 2\alpha \pi^T x$. The partial derivative with respect to an individual variable π_i is:

$$\frac{\partial f(\pi)}{\partial \pi_i} = \rho x_i^2 + 2x_i(c\pi^T x - \alpha) \equiv \tilde{f}_i \qquad (4.17)$$

Lemma 5. *Assume $x \succeq 0$ and that $c\breve{\pi}^T x - \alpha > 0$. Then, $x_i > x_j \Leftrightarrow \tilde{f}_i > \tilde{f}_j$.*

Proof. Assume $\tilde{f}_i > \tilde{f}_j$. For sake of contradiction, assume that $x_j > x_i$, or $x_j = x_i + \epsilon$ for $\epsilon > 0$. Then:

$$\begin{aligned} & \rho x_i^2 + 2x_i(c\pi^T x - \alpha) > \\ & \qquad \rho(x_i^2 + 2x_i\epsilon + \epsilon^2) + 2x_i(c\pi^T x - \alpha) + 2\epsilon(c\pi^T x - \alpha) \qquad (4.18) \end{aligned}$$

which implies that:

$$0 > 2x_i\epsilon + \epsilon^2 + 2\epsilon(c\pi^T x - \alpha) > 0 \qquad (4.19)$$

raising a contradiction, therefore $\tilde{f}_i > \tilde{f}_j \Rightarrow x_i > x_j$.

If $x_i > x_j$, then each term of $\tilde{f}_i$ is greater than the corresponding term of $\tilde{f}_j$, and the result follows. □

With Lemma 5 established, a solution for π is clear. Suppose $x_i > x_j$ and consider $\tilde{f}_i$ and $\tilde{f}_j$ when $\pi^T x$ increases slightly to $\pi^T x + \epsilon$, $\epsilon > 0$. The partial derivatives at this new value of $\pi^T x$ are denoted $\tilde{f}_i'$ and $\tilde{f}_j'$.

$$\tilde{f}'_i - \tilde{f}'_j = \rho x_i^2 + 2x_i(c\pi^T x + \epsilon - \alpha) - \rho x_j^2 - 2x_j(c\pi^T x + \epsilon - \alpha) \quad (4.20)$$

$$= \rho(x_i^2 - x_j^2) + 2(c\pi^T x - \alpha)(x_i - x_j) + 2\epsilon(x_i - x_j) \quad (4.21)$$

By our assumptions, $\tilde{f}'_i > \tilde{f}'_j$ for arbitrary values of $\epsilon > 0$.

Note that $\pi^T x$ increases on the solution path for the optimal value of π, so the ordering of partial derivatives remains constant. Further, the ordering matches the ordering of the individual components of x.

Thus, a solution may be described by the following algorithm:

Algorithm 2 Adversarial Solution for π in pLSI

```
procedure AdversarialParam($\pi$)
    $\pi \leftarrow \breve{\pi}$
    while $\vec{1}^T \pi < 1$ do
        $i \leftarrow \arg\max_{j \in [1,m]} x_j$
        $v \leftarrow \vec{1}^T \pi$
        $\pi_i \leftarrow \hat{\pi}_i$
        if $\vec{1}^T \pi > 1$ then
            $\pi_i \leftarrow \breve{\pi}_i + (1 - v)$
```

At the initial point, where $\pi = \breve{\pi}$, the most effective means to spend the “budget” is on the parameter π_i corresponding to the largest element of x. Once $\pi_i = \hat{\pi}_i$, the algorithm switches to the next largest element of x, repeating until the parameter vector π satisfies $\vec{1}^T \pi = 1$.

Chapter 5

Sparse Principal Component Analysis

The preceding chapters have focused on 2-class models of text and have assumed knowledge of document class membership. When document classes are unknown, we may use linear algebraic approximation methods such as Sparse Principal Component Analysis (SPCA) to identify underlying patterns in a text corpus. The results of SPCA are useful and legible in their own, providing accessible summaries of topics by identifying semantically consistent groups of features and documents. Further, the results of SPCA may be used in conjunction with probability models of text which enable principled methods of associating detected topics with text content.

SPCA computes a low-rank approximation to original raw data. The driving concept is that the low-rank approximation will capture interesting and useful semantic structures within the data, such as word usage patterns and associations. In 1990, Deerwester et. al. [9] described the use of Singular Value Decomposition (SVD), a low rank approximation, for exactly this purpose. They proposed that SVD would reduce noise and present a condensed representation of text retaining the most prevalent semantic structures. In the subsequent decades, sparsity was introduced in the interest of interpretability, efficient algorithms were developed, and the methods were applied to text content [12, 13, 21, 46, 48].

The implementation of SPCA explored in this dissertation identifies sparse principal components one at a time, approximately solving the problem:

$$\min_{u,v} \|A - uv^T\|_F + \lambda\|u\|_1 + \mu\|v\|_1 \tag{5.1}$$

where a rank-1 approximation problem is augmented with $\ell - 1$ regularizations to induce sparsity in the vectors u and v. Once the first problem is solved, the vectors u and v are referred to together as a "topic." To proceed, the original matrix A must be modified to reflect the fact that one prevalent pattern has been extracted, via a process called deflation. Typically, in the context of SVD, the updated matrix is $A' = A - \sigma uv^T$, which eliminates the rank-1 structure just uncovered. For the sake of computational efficiency, the implementation of SPCA described in this chapter is an approximation, and eliminates rows and columns of A corresponding to the sparse support of the vectors u and v. If, for instance, $v_i! = 0$, the ith column of the data matrix A is removed. The same process can be done for the rows of the matrix corresponding to non-zero elements of u.

The implementation considered is a modified power iteration [13,21], described in Algorithm 3, below.

5.1 Comparison with LDA

At present, popular perception is that the state-of-the art in document topic modeling is represented by Bayesian modeling approaches such as Latent Dirichlet Allocation (LDA) [4,18,31,34,36,42,43,47]. However, this section discovers empirically that the aforementioned SPCA implementation yields a competitive advantage over a collapsed Gibbs sampling implementation of LDA [34,45] in computation time, scalability, and quality of results. In summary, when computing a small number of topics over dataset sizes between 10K and 100K documents from the BBC News, the implementation of SPCA runs between 10 and 20 times faster than the point-of-comparison LDA implementation. A secondary experiment computing 1000 topics over 400K documents yields a 50-fold performance advantage for SPCA. Further, the results returned by SPCA are as good as or better than those returned by LDA, as

Algorithm 3 SPCA Approximation [13]

```
procedure SPCA($A$)
    $n_f$                    ▷ Number of features per topic, i.e. sparsity of $v$
    $n_d$                    ▷ Number of documents per topic, i.e. sparsity of $u$
    $n_t$                    ▷ Number of topics desired
    for all $i \in [1, n_t]$ do
        $u_i, v_i \leftarrow$ ITERATE($A, n_f, n_d$)
        $A \leftarrow$ DEFLATE($A, u_i, v_i$)
function ITERATE($A, n_f, n_d$)
    $u \leftarrow \vec{1}$
    $v \leftarrow \frac{1}{m} A^T \vec{1}$                    ▷ Initialization
    while $u, v$ not converged do
        $u \leftarrow \text{HardThresh}(Av, n_d)$             ▷ Enforce sparsity
        $u \leftarrow \frac{u}{\|u\|_2}$                      ▷ Normalize
        $v \leftarrow \text{HardThresh}(A^T u, n_f)$
        $v \leftarrow \frac{v}{\|v\|_2}$
    return $u, v$
function DEFLATE($A, u, v$)
    $J_i \leftarrow \{i \| \; u_i \neq 0\}$
    $J_j \leftarrow \{j \| \; v_j \neq 0\}$
    for all $i \in J_i$, $j \in J_j$ do
        $A_{ij} = 0$
    return $A$
```

the SPCA topics tend to be more focused on interesting stories in the news while LDA topics are more broad and general.

Given a body of text, the goal is to generate a list of k topics, each of which is described by a list of features (words) and a list of documents. The feature list contains words that are identified to be associated with one another within a subset of documents. The returned documents are relevant examples of documents using a mixture of words like the given word list. In application, this type of tool automatically organizes search results into distinct topics, and describes the dominant features of each topic as a word list.

LDA, on the other hand, represents topics as probability distributions over the set of words in the dataset. Computation involves estimation of the parameters of a graphical model, which is intractable in its complete form, as Blei et. al [4] state explicitly in their seminal paper. Efficient solutions utilize approximations to the full problem and sampling methods [16]; for example, the LDA implementation compared against in this section uses collapsed Gibbs sampling [18]. LDA continues by associating documents or portions of documents with topics via the estimated probability model.

LDA has been popular within machine learning and information retrieval communities due to its mathematical modeling of text and its formal analysis of its results. SPCA, on the other hand, is agnostic to the linguistic origin of the data it analyzes. Despite not modeling text sources explicitly, SPCA happens to extract salient features of a body of text numerically, and this chapter will show that the resulting topics are comparable to those returned by LDA. Low-rank data approximations such as SPCA have been used with much success in other domains, such as facial recognition [6] and genetic analysis [48], and the low-rank structure of large bodies of text are uncovered by SPCA.

5.1.1 Notable Differences in Methods

The two algorithms are not exactly interchangeable:

- SPCA is designed to be sparse, LDA is not. Sparsity is useful for:

  - user interpretation of results. Content needs to be concise enough for people to find it worthwhile to read.
  - achieving fast and memory-efficient computation

- SPCA clusters documents automatically, LDA requires an extra step of solving a maximum likelihood problem to identify relevant documents

- SPCA implementation returns topics one at a time. Once one topic is extracted, it can be immediately reported to a user, and subsequent computations are independent. LDA estimates all topics simultaneously, with no response to a user until computation is complete. If a user requests a new topic from SPCA, it can be computed quickly. If a user requests an additional topic from LDA, the entire computation must be redone.

5.1.2 Measurements

SPCA is compared with LDA in three areas:

- Computation time

- Quality of results (qualitative)

- Number of iterations until convergence is reached

Computation time is paramount for an interactive application. Researchers and analysts must run queries and have results reported with very little delay. Queries should be able to be modified painlessly, and words and topics a user isn't interested in should be able to be eliminated on-the-fly and results recomputed without long waits.

Equally important in an interactive system is the quality of the results. The utility of the results for the social sciences and humanities cannot be measured numerically, so the results of the algorithms are compared side-by-side and are described qualitatively.

Finally, while the implementation of LDA used did not include an explicit convergence criterion, one was added in order to investigate the optimization process. This is explained further in Section 5.1.6.

5.1.3 Test Implementation

The SPCA implementation described previously is compared to an implementation of LDA using collapsed Gibbs sampling [16,29,34,45] that is available in Python. This package, called `lda`[1], is freely available from the Python Package Index (`pip install lda`). This implementation of LDA is designed to be fast and efficient, representing a good target for comparison. An alternative LDA implementation, called Gensim [35](`https://radimrehurek.com/gensim/`), was evaluated for comparison as well, but computation times for each experiment were prohibitively long, and invariably much longer than either the SPCA implementation or the Python `lda` package.

The experiment was performed on an archive of articles from the BBC news:

- Contains $415,041$ documents
- Text in each document is contained in 3 fields. For this experiment, all 3 fields are combined together to represent a single document.
  - title
  - content
  - brief description of article
- Covers time range from April 21, 2010 to April 30, 2014

Computation time is measured across a range of various parameters:

- `n_topics`: number of topics: $(8, 12, 16)$
- `card_terms`: number of words in each topic: $(8, 12, 16)$

[1] Documentation can be found online: `http://pythonhosted.org/lda/`

- `dataset_size`: number of documents to analyze: Starting at 1000, going up to 409000[2] in increments of 8000.

Each algorithm is configured to run a maximum of 100 iterations before returning. The `lda` implementation has no explicit convergence criterion outside of the number of iterations. Each execution simply runs 100 iterations and terminates.

It should be noted that the SPCA algorithm is set to run a maximum of 100 iterations for every topic, while the `lda` implementation simply runs for 100 iterations total. LDA computes all topics at the same time, while SPCA computes one at a time. To explore this difference further, a separate experiment is reported in Section 5.1.6 where a convergence criterion is introduced into `lda` to determine the number of iterations required by each algorithm until convergence is reached. This comparison reveals a striking difference between `lda` and SPCA, where `lda` typically requires two orders of magnitude more iterations to converge than SPCA.

Computation time is measured by "wall-time". To mitigate the effects of variations in wall-time that are independent from the computation itself, each experiment is performed 3 times and the minimum time is reported.

For each tested dataset size, an appropriately sized random sub-sample of the entire corpus is generated. Documents are sampled uniformly and without replacement, and each sample is independent of the rest.

5.1.4 Computation Time

Figure 5.1 illustrates the computation time growth of the SPCA and `lda` implementations as the number of documents in the dataset increases. The computation growth is also shown for three different numbers of topics, 8, 12, and 16, to show the effect of adding topics to the overall computation time.

The SPCA results are illustrated in blue, and the `lda` results in red. Computation time of both algorithms increases steadily as the number of documents in the dataset gets larger. However, computation time for SPCA in the largest case remains well

[2] Experimental results are presented only up to roughly 113000 documents due to prohibitively long computation times on the part of `lda`.

below 50 seconds. On the other hand, over the dataset sizes explored here, `lda` requires a minimum computation time of between 110-120 seconds, and exceeds 400 seconds (over 6 minutes) for the larger datasets.

For each algorithm, note that computing additional topics adds some computation time cost. The slope of the `lda` curves seems to increase as well with the number of topics. A plot of the relative performance between the two algorithms, $\frac{\text{time(lda)}}{\text{time(spca)}}$, is presented in Figure 5.2.

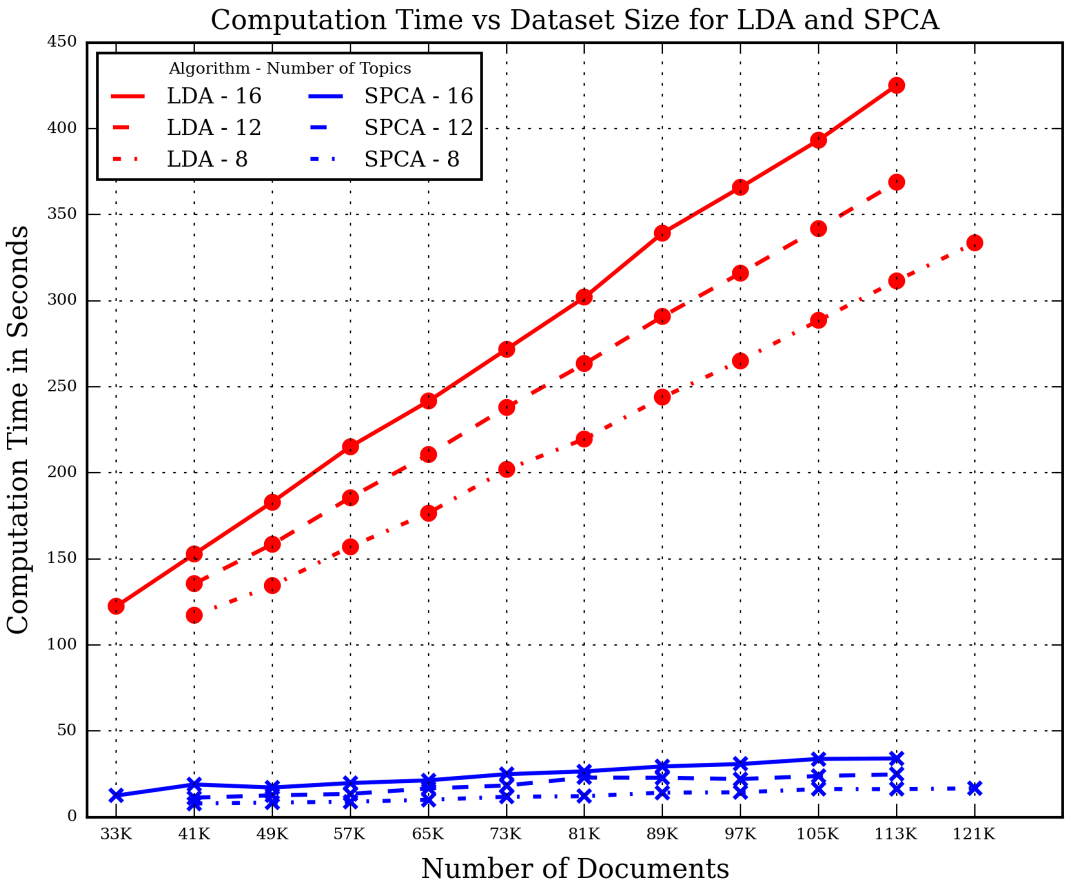

Figure 5.1: SPCA & LDA Computation Time Comparison

Each curve represents the relative computation times of `lda` and SPCA for different numbers of computed topics. Note that the relative performance of SPCA tends to increase in all three cases as the number of documents increases, implying an advantage for SPCA in terms of scalability.

Also, note that the performance gain of SPCA relative to LDA decreases as the number of computed topics increases. This is to be expected as the SPCA algorithm runs one optimization for every topic, so adding topics increases the total number of iterations required. The `lda` algorithm on the other hand always executes exactly 100 iterations of its optimization algorithm.

The performance of the two algorithms is also tested in a more demanding region: the computation time in measured for the task of returning 1000 topics over the entire BBC dataset, 415041 documents. Execution of `lda` was terminated after reaching 25% complete and full execution times are extrapolated. A very important difference is illuminated in the "time to first response." Here, as SPCA computes topics one at a time, and continues computing subsequent topics independently of the first, it may return topics immediately as they are uncovered, while `lda` must wait for the algorithm to run to completion.

	SPCA	**LDA**
Time to 25 % complete:	est. 8 minutes 21 seconds	7 hours 16 minutes 57 seconds
Total Time:	33 minutes 25 seconds	est. 29 hours 7 minutes 48 seconds
Time Per Topic:	2 seconds	105 seconds
Time to First Response:	2 seconds	est. 29 hours 7 minutes 48 seconds

In summary: for small numbers of topics and dataset sizes between 10K and 100K, SPCA runs between 10 and 20 times faster than `lda`. When large numbers of topics (1000) are required in a large dataset (400K documents), however, the performance gain is more dramatic. In total, LDA takes approximately 52 times longer. Since SPCA can return topics immediately once they are computed, SPCA can present to a user a new topic about every 2 seconds. This means that the user will begin to see results after 2 seconds. LDA, on the other hand, must run to completion before returning any results.

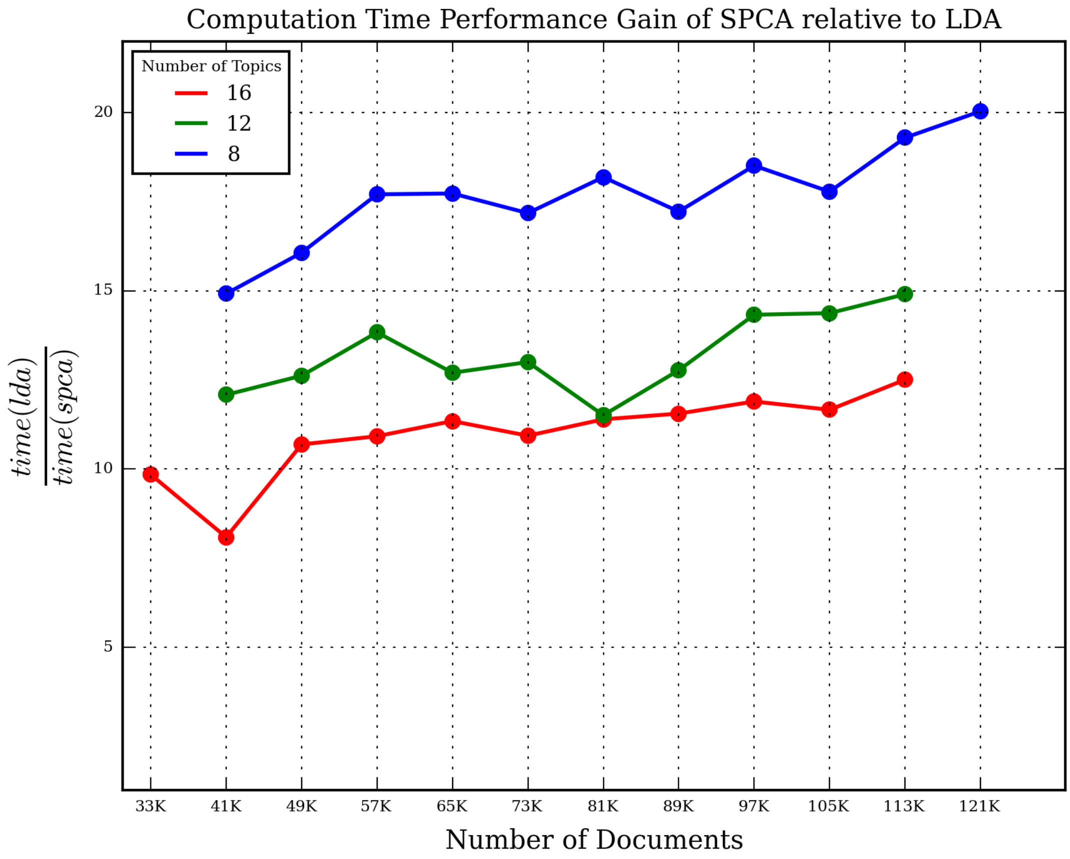

Figure 5.2: Performance Gain of SPCA relative to `lda`

5.1.5 Qualitative Comparison of Results

While speed is critical in an interactive system, it must be balanced with the quality of results. Presently, the textual human-readable results of SPCA and `lda` are compared and established to be similar, with some notable advantages for SPCA.

For this comparison, SPCA and `lda` are executed on two keyword search samples from the BBC dataset, constituting documents matching either "france" or "russia".

8 topics are generated, where each topic is thresholded to contain 32 features. For ease of presentation, the results are truncated to the top 12 features. For each topic, we also identify 16 documents that exemplify the given topic. To aid side-by-side comparison, topics generated by the two algorithms are matched with one another by solving the assignment problem with the Munkres algorithm[3]. This automatically pairs similar topics side-by-side. While some topics demonstrate similarity, it should be noted that others will be quite dissimilar as the two algorithms are quite distinct. The topics presented are manually assigned "names" which concisely describe their content. The results are presented below:

France This query matched 16,442 documents and 88,272 distinct words (features). Words marked in **bold** occur in multiple topics returned by `lda`. As the SPCA implementation is designed to be sparse, and employs an aggressive deflation scheme eliminating rows and columns from the data matrix, words may only occur in a single topic.

A few topics seem to correspond quite well: Nuclear Issues, the Eurozone Economy and the Middle East. The "Food, Culture, Gov-

[3] `https://pypi.python.org/pypi/munkres/`

Topic Name	**Features**
Food, Culture, Government	french - people - food - london - british - paris government - country - roma - hollande - president - english
Nuclear Issues	nuclear - power - energy - reactor - safety - reactors fukushima - epr - flamanville - plants - electricity - germany
Eurozone Economy	eurozone - greece - debt - euro - crisis - greek economic - banks - growth - austerity - bailout - euros
MidEast	syria - lebanon - syrian - israel - assad - minister hezbollah - israeli - lebanese - attack - beirut - security
Telecom	minitel - today - internet - telecom - service - services system - set - project - travel - offer - online
Turkey/Genocide	bill - turkey - genocide - turkish - law - ankara armenians - armenian - senate - erdogan - ottoman - armenia
Tax	tax - social - cgt - income - charge - rate taxes - capital - britons - contribution - residents - pay
Islam and War	malian - islamist - town - rebels - islamists - intervention west - northern - air - support - strikes - deployment

Table 5.1: SPCA Topics for “france”

ernment” topic returned by SPCA seems to broadly describe French issues including food, ties to England, and government. The other topics are more specific, describing discrete events such as the Greek debt crisis or recurring stories (like taxes) in the news. LDA offers some specificity in its topics as well, though “Family Words” and “General Words” don’t present tight ties to specific news stories. They appear to describe general relationships between words that appear throughout the corpus, yet the result is uninformative. Further, the LDA results include many words that are repeated across multiple topics. These words may be considered “stop-words,” words that are uninformative

Topic Name	**Features**
French Government	**french** - **president** - hollande - sarkozy - mali - party election - francois - **people** - **minister** - political - african
Nuclear Issues	nuclear - iran - china - power - **world** - russia britain - **president** - energy - **french** - countries - **government**
Eurozone Economy	european - eurozone - **europe** - germany - **government** - countries economic - **year** - growth - debt - crisis - economy
MidEast	**government** - forces - **people** - military - **president** - syria security - roma - libya - foreign - country - **minister**
General Economy	**people** - company - food - **year** - business - market industry - **years** - sales - firm - **europe** - number
General Legal	police - court - **french** - told - case - **government** paris - law - **year** - **years** - authorities - public
Family Words	**french** - **people** - **world** - **years** - **time** - children life - young - film - **year** - work - school
General Words	**people** - **year** - **time** - london - **day** - **years** british - team - tour - **world** - **french** - air

Table 5.2: LDA Topics for “france”

to the results (such as “year” and “years”.) SPCA may return such words as well, but they do not permeate all the returned topics. Of note, half of the words in the “Family Words” and “General Words” topics are repeated words, not specific to the particular topic. Over all the topics, a total of 40 out of the 96 words returned by LDA are repeated words, roughly 42%.

Now, compare the titles of documents associated with topics about two topics that seem to match well between LDA and SPCA, “Nuclear Issues” and the “Eurozone Economy”:

- **SPCA**

  - France expands nuclear power plans despite Fukushima
  - France struggles to cut down on nuclear power
  - Fessenheim: Splitting the atomic world
  - Japan disaster reopens nuclear debate in Europe and US
  - France nuclear: Marcoule site explosion kills one
  - Global fallout: Did Fukushima scupper nuclear power?
  - Cameron and Sarkozy hail UK-French relationship
  - Hundreds of problems at EU nuclear plants
  - Cameron and Sarkozy hail UK-France defence treaties
  - Nicolas Sarkozy and Manmohan Singh in nuclear deal Iran profile
  - Anti-nuclear protests in Germany and France
  - Parties clash over future of nuclear power in France
  - Greenpeace France nuclear action prompts security alert
  - UK nuclear plans put energy in French hands
  - Nuclear power gets little public support worldwide

- **LDA**
  - France expands nuclear power plans despite Fukushima
  - Iran nuclear talks resume in Geneva
  - Iran nuclear talks successful
  - Iran nuclear deal reached at Geneva talks

  - Nuclear deal: Iran couldnt take it
  - Iran FM Zarif: Geneva nuclear deal is first step
  - BAE-EADS merger: France and Germany must reduce stake
  - Iran nuclear talks to resume in Geneva amid optimism
  - Ministers urge nuclear safety tests after Japan crisis
  - UK and France agree to joint nuclear testing treaty
  - No deal at Iran nuclear talks
  - The South Atlantic question in French-British plan
  - Iran wants nuclear deal in months, says President Rouhani
  - Foreign powers disappointed at Iran nuclear talks
  - Iran nuclear: Israel PM warns against easing pressure
  - Khamenei: Iran will never give up its nuclear programme

The documents selected by the two algorithms are comparable, with each algorithm returning a mix of documents related to nuclear energy, concerns about Fukushima, and concerns about nuclear weaponry in Iran. Reflecting the mix of words in each topic, the SPCA results focus more on energy and Fukushima, while the LDA results focus more on nuclear weapons and Iran.

Next, consider documents related to the Eurozone Economy topics.

- **SPCA**
  - Viewpoints: Election impact on eurozone

  - Timeline: The unfolding eurozone crisis
  - Eurozone summits: Moments of truth or waste of time?
  - Germany v France: The eurozones next big battle?
  - Eurozone debt web: Who owes what to whom?
  - Eurozone crisis: European voices
  - George Osborne: Eurozone crisis threatens all Europe
  - France shrugs off loss of top triple-A credit rating
  - Why the eurozone downgrades matter
  - Q&A: Eurozone rescue proposals
  - France loses AAA rating as euro governments downgraded
  - How Dexia was caught out by the eurozone debt crisis
  - The domino effect in Europes debt crisis
  - Eurozone ministers approve 8bn euro Greek bailout aid
  - Who will dictate Europes future?
  - Moodys keeps French AAA credit rating

- **LDA**

  - Eurozone services sector growth slows again
  - Europe economy: Recession hits Italy and Netherlands
  - France to enter recession as eurozone growth slows
  - German exports set record of a trillion euros in 2011
  - ECB keeps eurozone interest rates unchanged at 1.5%

- French bank Credit Agricole to cut 2,350 jobs
- Stock markets down on Greek swap fears
- Eurozone business growth slows
- Eurozone economy grows 0.2% in third quarter
- Germanys economy grows by 0.3%
- Commerzbank sees profits increase
- French economic growth revised down
- French jobless rate climbs to highest level in 15 years
- Euro drops below $1.31 for first time since January
- Fitch revises outlook on France to negative
- European Central Bank keeps rate at record low

The documents returned by each algorithm in this case are quite similar, concentrating on the Greek debt crisis and other economic issues in the Eurozone.

Russia This query contains 10,456 documents and 65,698 distinct words (features).

In this case, 48 out of the 96 words returned by LDA are repeated words. 75% of the words in the "General News" topic are not specific to the topic, and include a lot of words about time such as "year/years", "time", and "day", which do not convey any information about some

Topic Name	Features
Crimean Crisis	crimea - ukraine - people - russian - country - ukrainian international - crimean - sevastopol - today - state - political
Georgia Conflict	georgia - georgian - south - saakashvili - tbilisi - ossetia abkhazia - troops - soviet - elections - opposition - parliament
China & Economy	china - trade - oil - energy - gas - economic foreign - resources - europe - economy - chinese - investment
Syria & Chemical Weapons	syria - weapons - syrian - chemical - assad - security arab - council - action - resolution - regime - middle (east)
N.Korea Nuclear	north - nuclear - korea - talks - programme - korean arctic - pyongyang - sea - fuel - test - officials
Gay Rights	gay - rights - propaganda - homosexuality - bbc - sexual public - news - hate - report - homophobic - live
Euro Relations	germany - serbia - austria - hungary - german - britain france - responsibility - conflict - leaders - vienna - berlin
Elections	election - vote - vladimir - medvedev - prime - communist duma - result - seats - leader - presidential - ruling

Table 5.3: SPCA Topics for “russia”

event or story. In comparing the topics returned by SPCA and LDA, SPCA again concentrates more on specific news events, while LDA tends to select topics that show broad corpus-wide relationships between words.

Now, compare the titles of documents selected by LDA and SPCA to be relevant to the topics of Crimea and Syria, topics that overlap well between LDA and SPCA.

For Crimea:

- **SPCA**
  - Crimea crisis: Russian President Putins speech annotated

Topic Name	**Features**
Crimean Crisis	ukraine - **russian** - **crimea** - ukrainian - **president** - kiev **moscow** - **putin** - pro - yanukovych - **government** - eastern
Energy	**russian** - **gas** - **president** - nuclear - georgia - nato soviet - union - **moscow** - energy - south - **military**
China & Economy	china - **russian** - **year** - **government** - economic - economy market - **country** - business - **world** - india - **oil**
Syria Chemical Weapons	syria - syrian - **government** - security - assad - **president** **military** - **people** - weapons - council - foreign - forces
Oil (Iran & US)	iran - **world** - countries - **oil** - **international** - **time** deal - arctic - israel - obama - programme - **gas**
General Words	**people** - **russian** - **world** - years - **year** - **time** city - **moscow** - bbc - **country** - day - children
General News	**people** - snowden - **russian** - **country** - **world** - **president** **political** - **crimea** - **state** - **international** - daily - media
Elections & Government	**russian** - **putin** - **moscow** - **president** - party - election vladimir - minister - court - **political** - opposition - **state**

Table 5.4: LDA Topics for "russia"

- Russia profile
- Vladimir Putin: The rebuilding of Soviet Russia
- Deadly clashes at Ukraine port base as leaders meet
- Voices from the conflict in Crimea
- Ukraine-Russia gas row: Red bills and red rags
- Analysis: Why Russias Crimea move fails legal test
- What is Russias vision of a federal Ukraine?
- Ukraine crisis: US warns Russia over destabilisation
- Ukraine crisis: US urges restraint and warns it is watching

  - Russia
  - Chechnya profile
  - Analysis: Russias carrot-and-stick battle for Ukraine
  - Ukraine crisis: Does Russia have a case?
  - Ukraine: Europes major test
  - Ukraine crisis: Whats driving Russias response?
  - Ukraine crisis: Deal to de-escalate agreed in Geneva

- **LDA**
  - Crimea result makes “a mockery” of democracy says Hague
  - UK will stand up for Ukraine, says David Cameron
  - Ukraine crisis: EU extends sanctions over Crimea
  - Ukraine crisis: Hague praises EU for action against Russia
  - Russia is more isolated, says EC chief Jose Manuel Barroso
  - Lithuanias Dalia Grybauskaite warns of prelude to new Cold War
  - The EU does not recognise outcome of Crimea referendum
  - Ukraine crisis: Sergei Lavrov news conference
  - Crimea referendum: Voters back Russia union
  - Crimea always part of Russia
  - Crimea MPs vote to join Russia and announce referendum
  - Crimea exit poll: More than 90% back Russia union

  - Russia will respect Crimea vote
  - Ukraine crisis: EU ponders Russia sanctions over Crimea vote
  - Muscovites on controversial Crimea referendum
  - Ukraine crisis: EU imposes sanctions over Crimea

For documents pertaining to Syria and Chemical Weapons: For Crimea:

- **SPCA**
  - Viewpoints: Can Russias chemical weapons plan for Syria work?
  - Analysis of Putins plea to Americans over Syria
  - Viewpoints: Is there legal basis for military intervention in Syria?
  - Chinese, Iranian press alone back UN Syria veto
  - Syria unrest: Russia pulled two ways
  - President Putins Middle East gambit
  - Why Russia sells Syria arms
  - Syria resolution: The diplomatic train-wreck
  - How to destroy Syrias chemical arsenal
  - Syria crisis: Assad confirms chemical weapons plan
  - Syria crisis: Why is Russia defending Bashar al-Assad?
  - Syria crisis: Tense US-Russia talks on chemicals deal

  - Syria profile
  - PJ Crowley: Syria crisis upends Mid-East positions
  - Syrias Assad will go, says US, as UN vote nears
  - All eyes on Russian ministers Syria trip

- **LDA**
  - Syria is implementing peace plan, says foreign minister
  - Ban Ki-moon calls for one voice on Syria
  - UN vote on Syrias chemical weapons stockpile
  - UN meets to discuss resolution to stop Syria violence
  - John Kerry: Syria needs political, not military solution
  - Can Syrias chemical arsenal be hunted down?
  - Syria: US backs Red Cross call for truce
  - Hillary Clinton: Syria violence unconscionable
  - Syria crisis: UN inspectors renew chemical attack probe
  - Russias Lavrov urges Syria to comply with Annan plan
  - Clashes in Syria leave 19 dead
  - Syria: New UN call over human rights abuses
  - Syria ceasefire: UN expected to vote on monitor team
  - Leaked report: Peace envoy suggests Assad should go
  - Arab League to call for UN backing on Syria plan
  - Russia cannot support UN Syria draft resolution

Once again, the results in the returned documents in these topics are quite comparable. The Syrian topic returned by SPCA focuses more on Russia-specific relationships, while the LDA topic returns a broader spread of international perspectives.

5.1.6 Convergence Issues

It was noted previously in this chapter that the `lda` algorithm does not have any criterion for convergence, and the algorithm is set to run for exactly 100 iterations. In this experiment, a convergence criterion is added to the software. The performance of `lda` is measured in terms of the log likelihood of the observed data with respect to the estimated probability model. This algorithm computes the log likelihood for the overall model along with the average log likelihood per word in the dataset. The `lda` algorithm is modified to identify convergence when the absolute change in the per-word log likelihood between consecutive iterations is $< 1^{-10}$. To compare this with SPCA: the convergence criterion is measured in the change in angle of the output vectors between iterations, which is also set to $< 1^{-10}$.

For different dataset sizes, the total number of iterations required by each algorithm is recorded over 8 executions on random subsets of data. The average and the standard deviation among the results are computed.

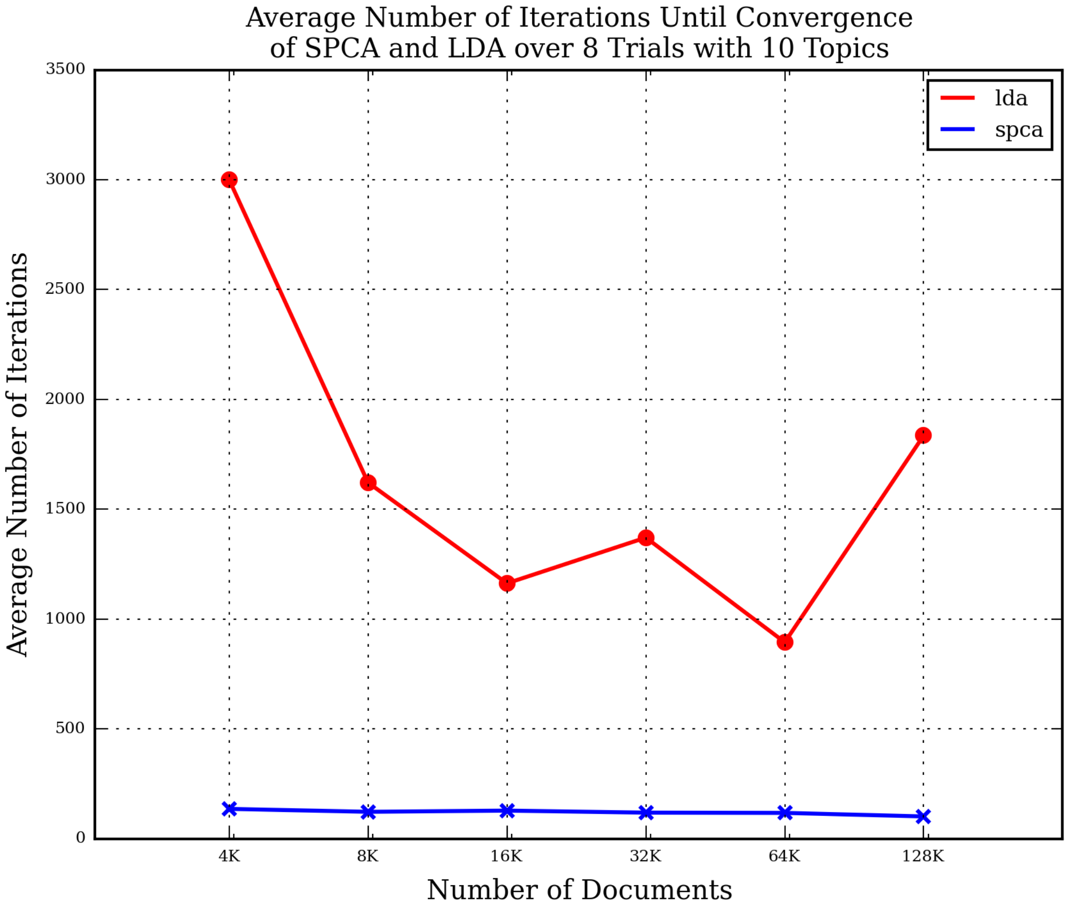

Figure 5.3: Average number of iterations until convergence

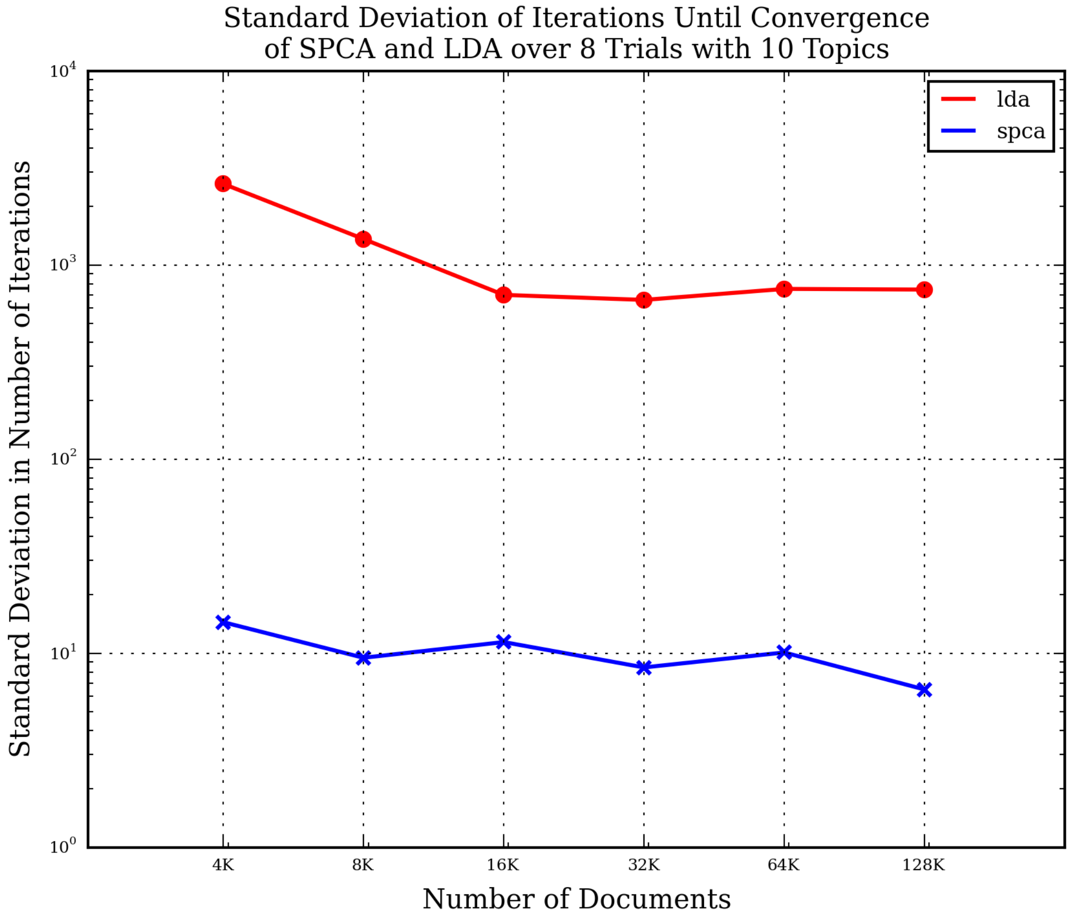

Figure 5.4: Standard deviation in the number of iterations required until convergence

Figure 5.3 indicates that the SPCA algorithm consistently requires just over 100 iterations to converge to a solution. LDA, on the other hand, is widely varied, typically requiring over 1000 iterations to converge, and sometimes many more. Figure 5.4 illustrates the standard deviation in the number of iterations required for convergence, highlighting a two-order of magnitude difference between the two algorithms. When run to convergence, therefore, the SPCA implementation is far more consistent in run-time, and requires many fewer iterations in general. Also, if run to convergence, the `lda` implementation typically takes at least 10 times longer to execute than in Section 5.1.4, where execution was truncated at exactly 100 iterations.

5.2 Topic Tagging

This section connects the results of SPCA to the pLSI model introduced in Chapter 2.3. Each topic returned by SPCA consists of two vectors, u and v, corresponding to left and right sparse singular vectors. Vector $v \in \mathbb{R}^m$ associates weights with features; intuitively it represents the relative rates with which features appear in documents associated with the given topic. This section describes how these feature vectors may be interpreted as parameters of categorical distributions and introduces similarity metrics based on the Hellinger distance to determine how well a topic is represented within a document.

First, some additional conditions are required for SPCA. The algorithm expects a centered data matrix, where each entry represents how much more or less frequently a feature appears in a document with respect to the corpus-wide average. Equivalently, the algorithm operates on centered matrix A' calculated as:

$$A' = A - \frac{1}{n}\vec{1}A^T \tag{5.2}$$

Additionally, the hard thresholding step in the algorithm is required to maintain only nonnegative components, so feature vector $v \succeq 0$. At the end of the algorithm, an additional step ℓ_1-normalizes the result.

With these conditions, each feature vector v is such that:

$$v \succeq 0 \tag{5.3}$$

$$\sum_{i=1}^{m} v_i = 1 \tag{5.4}$$

Thus, v may be interpreted as the parameter of a categorical distribution over the set of features.

Next, consider a vectorized document, $d \in \mathbb{R}^m$, within the same feature-space. With the interpretation of a topic's feature vector v as the parameter to a categorical distribution, assume that d is distributed as a categorical with some different parameter vector w, which is estimated from an observed document, perhaps including a model of uncertainty. The Hellinger distance is proposed to evaluate the similarity between v and w [3], and has precedent for use in analyzing text [22]. The Hellinger distance is defined as:

$$H(x, y) = \frac{1}{\sqrt{2}} \|\sqrt{x} - \sqrt{y}\|_2 \tag{5.5}$$

where $\sqrt{x}$ is defined as the element-wise square root of vector x.

Critically, for vectors representing parameters of categorical distributions, this is a bounded metric:

$$H : \mathbb{R}^m \times \mathbb{R}^m \rightarrow [0, 1] \tag{5.6}$$

$1 - H(x, y) = 0$ occurs in this case when $x^T y = 0$, or when there is no overlap in the support of the two probability distributions.

$1 - H(x, y) = 1$ occurs in the event that $x \equiv y$.

In the event of uncertainty in the estimate of w from an observed document, first note that a maximum-entropy estimate of w can be computed in time $O(\log m)$ as described in Section 4.2.1.

An alternative is to consider the following optimization problem for finding w maximizing the ℓ_2 distance to v subject to box constraints.

$$\arg\max_{w} \|w - v\|_2^2 \tag{5.7}$$

$$\text{s.t.} \quad \breve{w} \preceq w \preceq \hat{w} \tag{5.8}$$

$$\vec{1}^T w = 1 \tag{5.9}$$

While not directly maximizing the Hellinger distance, a simple method of estimating w emerges making it suitable in practice.

The Lagrangian of the problem is:

$$\begin{aligned} &L(w, \lambda_1, \lambda_2, \eta) = \\ &\qquad - w^T w + 2w^T v - \lambda_1^T (w - \breve{w}) - \lambda_2^T (\hat{w} - w) - \eta(\vec{1}^T w - 1) \end{aligned} \tag{5.10}$$

Calculating the gradient with respect to w, and interpreting λ_1 as a slack variable while equating to 0 yields:

$$\overset{\star}{\lambda}_1 = 2(v - w) + \lambda_2 - \eta\vec{1} \tag{5.11}$$

Consider three cases for the box constraints:

1. $w_i = \breve{w}_i$

2. $w_i = \hat{w}_i$

3. $\breve{w}_i < w_i < \hat{w}_i$

Case 1 implies $\lambda_{1i} \geq 0$ and $\lambda_{2i} = 0$. Therefore:

$$w_i = \breve{w}_i \Rightarrow 2(v - \breve{w}_i) - \eta \geq 0 \tag{5.12}$$

Case 2 implies $\lambda_{1i} = 0$ and $\lambda_{2i} \geq 0$. Therefore:

$$w_i = \hat{w}_i \Rightarrow 2(v - \hat{w}_i) - \eta \leq 0 \tag{5.13}$$

Finally, case 3 implies $\lambda_{1i} = \lambda_{2i} = 0$. Therefore:

$$w_i \in (\breve{w}_i, \hat{w}_i) \Rightarrow 2(v_i - w_i) = \eta \tag{5.14}$$

Combining these conditions yields a simple relationship between η and w_i, and the problem may be solved in a similar fashion to the maximum-entropy problem.

$$\overset{\star}{w}_i = \begin{cases} \breve{w}_i & \Leftarrow \frac{\eta}{2} \leq v_i - \hat{w}_i \\ \hat{w}_i & \Leftarrow \frac{\eta}{2} \geq v_i - \breve{w}_i \\ \frac{2v_i - \eta}{2} & \Leftarrow v_i - \hat{w}_i < \frac{\eta}{2} < v_i - \breve{w}_i \end{cases} \tag{5.15}$$

Note that the derivative of $\overset{\star}{w}_i$ with respect to η is always negative. A solution emerges by choosing an initial η_0 such that $\eta_0 = \max_i 2(v_i - \breve{w}_i)$. At this value, the candidate solution is $w = \hat{w}$. Assume that $\vec{1}^T \hat{w} \geq 1$, otherwise the problem is infeasible.

Repeatedly decrease η and recompute w according to Equation 5.15, checking the sign of $\vec{1}^T w - 1$ in order to satisfy the equality constraint. This problem can be discretized on the sorted list of confidence interval boundaries in exactly the same way as in the Maximum Entropy problem described in Section 4.2.1, and can exploit bisection to achieve a solution in time $O(\log m)$.

Chapter 6

Applications & Examples

This chapter discusses the application of the methods described in the preceding chapters to real-world datasets. One example regarding topical analysis of BBC news data, was presented in Chapter 5. This chapter discusses four more applications on: news from Aljazeera English, messages from Twitter, a work of fiction ("Harry Potter and the Sorcerer's Stone" [37]), and a collection of United States Patents pertaining to the general area of Clean Technology.

The examples demonstrate practical aspects of the methods of this dissertation and their applicability to a wide range of types of text content. In addition, each example describes how results may be read and interpreted by an individual, and what sorts of insights may be uncovered.

6.1 Keyword Expansion in Aljazeera English

Keyword expansion refers to the application wherein a user is uncertain about relevant keywords to use when searching for content. In this example, a user is interested in understanding how “Obama” is portrayed in this news source, and the tool recommends additional keywords that may be relevant. The keywords themselves hint at the type of content associated with the query (“obama”).

The Aljazeera English dataset used in this section comprises 13,289 articles spanning almost two years, from March 16, 2011 to February 26, 2013.

This example compares four different approaches: results from BIM, pLSI, and implementations of Logistic Regression and Lasso in the Python package “scikit-learn” [32].

First, consider the BIM model. Most of the computation involved is in the parameter estimation problem. The following table compares results for BIM using a maximum-likelihood estimate and a robust minimized ratio estimate, which is described in Equation 4.4. The top 15 keywords are extracted in each case: see Table 6.1

Note that “obama” appears as the top keyword in both cases. This is due to the keyword query design: the class of documents mentioning Obama is compared against the class of documents not mentioning Obama. It should be noted that this is an unbalanced classification problem, where 447 documents mention Obama and 12,842 do not. This can create issues for classification algorithms as noise can lead to

Maximum Likelihood	**Minimum Ratio**
obama	obama
barack	barack
hagel	hagel
mutually	boehner
unbreakable	apec
advertisement	unbreakable
apec	advertisement
obamacare	mutually
flickers	romney
qishan	mitt
abiding	chuck
reorienting	charlotte
payrolls	reorienting
pinching	fisher
prey	andrews

Table 6.1: Comparison of results of BIM for two different parameter estimates

poor performance and, in this case, misleading results.

Table 6.1 demonstrates that the robust estimate of the underlying parameters improves the clarity of the results. There is significant overlap between the two, including related individuals like Chuck Hagel and including agencies related to news stories such as APEC (Asia-Pacific Economic Cooperation). The maximum likelihood estimate results contain more action words such as “flickers”, “pinching”, and “reorienting”, which do not have an intuitive or informative association with Obama. Rather, these words emerge due to noise in the dataset and the unbalanced nature of the classification problem. The minimum ratio results, on the other hand, include more words describing major related actors: “Chuck Hagel”, “Mitt Romney”, and “John Boehner”. Each indicates an association with Obama that may provide further insight on an aspect of the news surrounding Obama.

Next, compare the robust BIM result to those of Logistic Regression and Lasso. Note that the optimization problem for Logistic Regression and Lasso both involve a regularization parameter which must be adjusted in order to achieve the desired sparsity level of 15 words. This necessitates solving the optimization problem multiple times.

BIM	**Logistic Regression**
0.28s	1.21s
obama	obama
barack	barack
hagel	administration
boehner	republican
apec	negotiations
unbreakable	trans
advertisement	ground
mutually	department
romney	prices
mitt	urges
chuck	bachmann
charlotte	halt
reorienting	attacks
fisher	crackdown
andrews	ally

Table 6.2: Comparison of results of BIM and Logistic Regression for keyword expansion on “Obama”.

Table 6.2 presents these results along with computation times recorded in the experiment. The computation time for BIM is mostly in computing Clopper-Pearson [7] exact confidence intervals.

Note that the logistic regression algorithm operates on the full count matrix, where the number of times a feature appears in a document is maintained. BIM, on the other hand, uses a binary matrix, recording only if a word appears in a document or not.

The differences in the results boils down to specificity. BIM concentrates on key actors and agencies. Logistic Regression includes some specific terms as well, such as “trans (Trans-Pacific Pipeline)” and “Michelle Bachmann”, though many words are much broader, like “administration”, “republican”, and “negotiations”. Each has a clear and intuitive association with Obama, yet seem too broad to be particularly insightful.

This yields an interesting insight about the simple Binary Independence Model. While the representation disregards a lot of information about precise sentences, it presents an abstraction that can be useful and insightful. The reason specific individuals appear more in the BIM results than others is the fact that the matrix is binary; the frequency of word usage within a document is ignored. Broad words, such as “administration”, may appear many times in a document about Obama (“the Obama administration”), yet the word itself may be used in different contexts in different parts of the news archive. Logistic Regression identifies that the word “administration” is indeed well associated with Obama when compared to all the other documents. However, it is not particularly *specific* to documents mentioning Obama. The binary representation of BIM avoids this problem as the model puts any document mentioning a given word on equal footing. Since words like “administration” are not specific to Obama, but certain words like “Romney” are, BIM tends to select the more specific keywords.

6.2 Topic Analysis in Twitter

In this section, we analyze a small collection of Tweets: 21,495 messages pertaining to womens’ health issues over the month of March 2014. A peculiarity of the dataset is that messages may be “re-tweeted”, leading to many duplicates of the same message. Naive application of the SPCA algorithm leads to document vectors (left principal components) that index identical messages, which is undesirable for an individual using such a tool. So, a pre-processing step is introduced to identify duplicates and combine them into a single message. If k duplicates are detected, the single stand-in message is given k-times the weight to maintain the “importance” of the message in the dataset.

The following show results of the topic analysis. Given the very short length of Twitter messages, only a small number of features are maintained in the feature vectors. Table 6.3 shows the identified keywords defining four topics.

Table 6.3 demonstrates an interesting and useful aspect of SPCA: automatic language clustering. Because SPCA is identifying the “latent semantic structure” of text, it can identify groups of words commonly associated with one another and isolate them. In this case, languages are distinct with few overlapping words.

Topic 1	**Topic 2**	**Topic 3**	**Topic 4**
kanker	study	life	abnormal
serviks	women	foolish	doctor
akibat	#cancer	extend	recommend
meninggal	caught	days	chances
setelah	#papsaveslives	#tdh	colposcopy

Table 6.3: Features defining four topics extracted automatically by SPCA from a collection of Twitter messages.

Table 6.4 contains two example messages exemplifying each topic, which were identified by the SPCA algorithm. Beyond segmenting by language, note that the algorithm segments more broadly by population or by opinion. For example, Topic 2 includes individuals promoting pap smears and touting their effectiveness at preventing cervical cancer. This is immediately counterposed with messages expressing skepticism about pap smears, and warning their audience about them. These two markedly different opinions are automatically detected and presented to the user, allowing for immediate analysis of the breadth of opinions expressed in a dataset. This ability would otherwise require manual reading, coding, sampling of data.

6.3 Topic Analysis of Fiction

A different use of topical analysis is described in application to the text of “Harry Potter and the Sorcerer’s Stone” [37]. The text is broken down into individual sentences, which are used as the unit of analysis. SPCA is used in this example to explore the characterization of the

Topic	Examples
2	Study: Paps save lives, even in women who get cervical cancer `http://t.co/SyxjiKKP` via usnews Regular Pap Smear Boosts Cervical Cancer Survival: Study: THURSDAY, March 1 (HealthDay News) – Women who have r... `http://t.co/Q3wH4gZX`
3	RT This Foolish Cancer "Prevention" May Only Extend Your Life by 2.8 Days `http://t.co/IOWrDI3Q` #TDH Dr.Mercola Health: This Foolish Cancer "Prevention" May Only Extend Your Life by 2.8 Days: By Dr. Mercola Women... `http://t.co/fDKICZIz`
4	Colposcopy After Abnormal Pap: If you have had an abnormal Pap smear, chances are your doctor has recommended th... `http://t.co/4MT4S9Zv` It can be scary when a Pap test comes back abnormal. The next step might be a colposcopy heres how that works: `http://t.co/wpscklWZ`

Table 6.4: Example messages pertaining to specific topics

character Hermoine Granger. Specifically, the algorithm analyzes the subset of all sentences that mention either “hermoine” or “granger”.

Table 6.5 shows features identified for 5 topics pertaining to Hermoine Granger. Names are associated with each topic which are assigned manually.

The first topic, “Harry & Ron”, identifies first the characters closest to Hermoine, and in addition their most common shared activities and locations. Those involve being in the library, studying charms, and working on homework.

The second topic, “Dialogue”, focuses on the qualities of dialogue

Topic Name	**Features**
Harry, Ron	ron - harry - checking - homework - charms - window - sat - boat
Dialogue	told - snape - gasped - clutching - chest - stitch - change - plan
Class	neville - fang - draco - flying - nervous - urged - suffering - looked
Hagrid	hagrid - path - voice - round - warm - flattering - puffing - running
Ominous Senses	thought - heard - sweets - rats - lurking - footsteps - sense - whisper

Table 6.5: Topics pertaining to Hermoine Granger in "Harry Potter and the Sorcerer's Stone."

Hermoine participates in. It touches on how conversation is delivered: with authority ("told") and with fear & anxiety ("gasped"). The anxiety is reinforced with "clutching" and "chest". The content of the conversations is included in discussion of "Snape" and in "plan"s and how they "change".

The first topics returned by SPCA are the broadest, with additional topics becoming more specific. This third topic, "Class", is much more specific, and captures major elements of one scene where students gather for Hagrid's class to learn to ride a Hippogriff. Neville and Draco both interact with the creature and express anxiety about the experience. The words of this topic capture both the major actors and the mood of this scene.

The "Hagrid" topic extracts key attributes of the character Hagrid, such as "round" and "warm" that describe the safe, comforting, parental qualities conveyed through his character. In addition, the topic captures major trends in interactions Hermoine has with Hagrid: she often runs down the path to his hut to seek him out.

Finally, the “Ominous Senses” topic focuses on the prevalent atmosphere of the novel, that something is lurking, hidden, and approaching.

6.4 Topic Analysis of United States Patents

This dissertation concludes with an analysis of the abstracts of 29,447 patents pertaining to clean technology and spanning nearly six decades from 1957 to 2013. This example leverages the topic tagging concept of Section 5.2 and demonstrates how quantification of topics can offer additional context, information, and insight.

First, SPCA was used to automatically extract 20 topics from the entire set of 29,447 patents. As in Section 6.3, topics are manually labeled with names representative of the content they summarize. The following list introduces these topics with names in bold and a set of defining features.

Solar Cells solar - cell - module - cells - diode - contact - layer - array

Wind Power power - wind - system - electrical - generator - output - voltage - converter

Hot Water Heater water - tank - heat - pump - hot - temperature - heating - flow

Fuel fuel - assembly - rods - nuclear - rod - composition - reactor - additive

Energy Conversion energy - device - storage - converting - conversion - wave - thermal - apparatus

Rotors rotor - blade - turbine - blades - hub - includes - axis - edge

Materials material - semiconductor - organic - photovoltaic - conductive - method - substrate - form

Soybeans soybean - plant - cultivar - parts - methods - plants - relates - produced

Surfaces & Photovoltaics surface - light - formed - electrode - silicon - side - transparent - front

DNA & Amino Acids acid - nucleic - fatty - sequence - producing - amino - encoding - comprising

Fluid Pressure fluid - working - pressure - transfer - collector - outlet - inlet - source

Chemical Processes process - production - ethanol - fermentation - high - preparation - acids - biomass

Air Pressure air - chamber - heated - compressed - combustion - panel - building - duct

Gas gas - liquid - stream - hydrogen - exhaust - oxygen - landfill - vessel

Controls control - signal - unit - drive - core - operation - signals - circuit

Thin Films film - thin - forming - oxide - amorphous - type - metal - deposited

Support Structure structure - support - element - member - wall - panels - frame - supporting

Physical Description portion - body - upper - lower - outer - extending - plate - portions

Assembly & Apparatus tube - tubes - guide - steam - absorber - cladding - glass - length

Shafts & Gears shaft - mounted - connected - rotation - gear - housing - vertical - bearing

Note that the topics correspond to distinct areas of technological innovation. Each area is quite broad, though a sub-topic analysis will be introduced later in this section demonstrating how additional detail and specificity can be uncovered with topic hierarchies.

First, using this set of 20 topics, determine the similarity between each document and topic pair, using the Hellinger distance methodology as defined in Section 5.2. Each document is assigned a value between 0 and 1 representing how well the document matches the given topic, where 1 is a perfect match.

Figure 6.1 illustrates the average strength of each topic over the entire dataset. This yields insight into the frequency with which patents matching the topic are generated, and allows for comparisons between topics. The chart provides a “fingerprint” of a dataset relative to a selection of topics, and can be recomputed on any subset of the data.

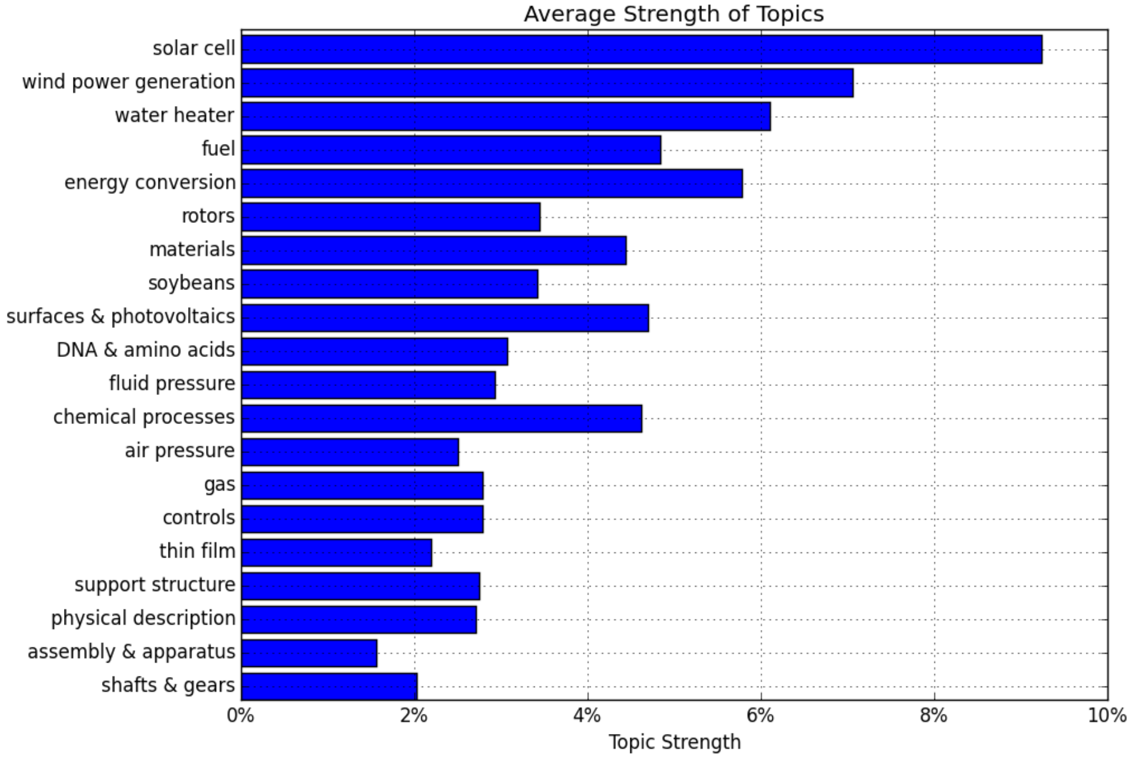

Figure 6.1: Average Topic Strengths of Top 20 Topics

This idea of computing “fingerprints” for subsets of data is employed to independently compute average topic strengths for sets of patents in each decade to generate a time-series. The time series can be used to identify trends and correlations between topics within the dataset. Figure 6.2 illustrates time series for the first 16 topics returned by SPCA.

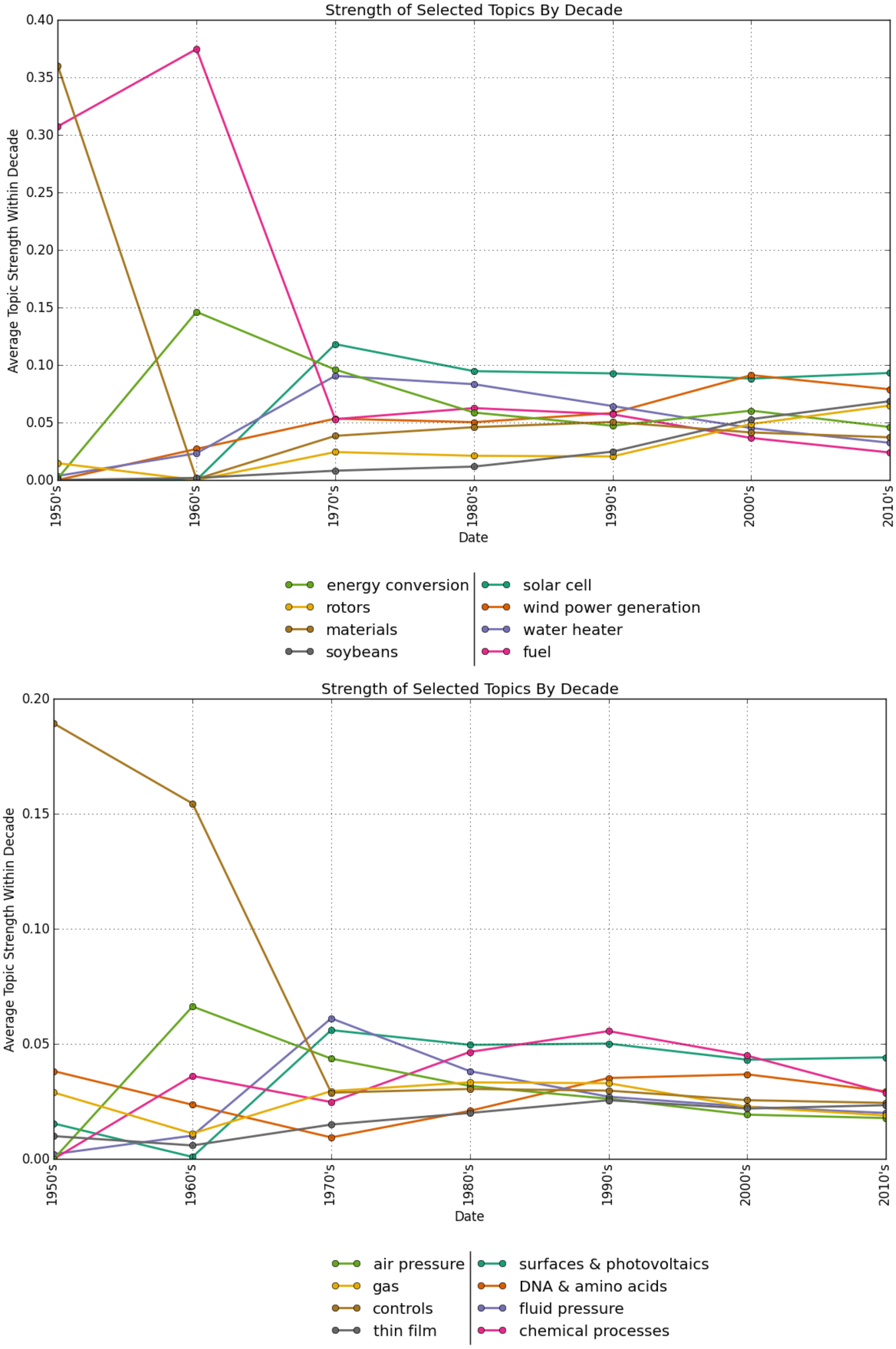

Figure 6.2: Average Strength of Topics by Decade

A few insights into the evolution of the interest in specific application areas are apparent from Figure 6.2. First, “Materials”, “Fuel”, and “Controls” are trendy in the 50’s and 60’s. There was a swell of interest into “Chemical Processes” in the 90’s. Also, interest in “soybeans” and “rotors” steadily increase over time to reach their largest strength in the 2010’s.

As these top-level topics are quite broad, it is useful to “drill down” into one topic to understand its subtleties and sub-patterns. In this example, consider the topic “Rotors”. While it is good to know that “Rotors” are discussed in patent applications and know how frequently they are discussed (from the topic strength), it would be useful to understand the context in which rotor technologies are discussed, what they are used for, what particular types of innovations are described in rotors, etc. This insight is achieved with a hierarchical sub-topic analysis, where SPCA is computed on the sub-set of documents in which the topic strength of “Rotors” is > 0.

The automatically generated and manually named subtopics for “Rotors” are as follows:

Wind Power wind - power - speed - method - installation - angle - energy - generator

Turbine Innovation, Control pitch - shaft - bearing - control - system - drive - connected - provided

Stator stator - plurality - outer - side - core - machine - windings - rotation

Air Flow, Pressure fluid - flow - direction - air - device - pressure - mounted - assembly

Tower tower - portion - tip - position - root - nacelle - rotational - support

Spar & Joints spar - cap - segment - joint - segments - surface - preform - attached

Electrical Load electrical - connection - conductor - plant - resistance - comprising - provide - disposed

Water Turbine water - body - extending - runner - wheel - comprises - driven - generating

Carbon Fiber fibers - material - carbon - flange - formed - including - reinforcing - embedded

Drive Train gear - stage - structure - transmission - ring - planet - carrier - forces

Airfoils airfoils - airfoil - lift - range - length - family - maximum - coefficient

Turbine Blade Deflection beam - deflection - determining - sensor - based - coupled - coupling - positioning

Wind Powered Engine engine - block - pistons - cooling - connecting - heating - reciprocate - causing

Rotor Innovations component - shell - access - window - configured - region - defined - generally

General main - medium - output - ambient - input - apparatus - low - velocity

Manufacturing Blades layer - cross - binding - fibre - fiber - element - central - front

Confinement & Seals tappets - housing - seal - number - adjustment - adjusted - adjustable - closer

Wind/Water Designs aerofoil - thereof - underwater - chord - symmetrical - unit - units - mid

Geothermal Vapor Generator series - geothermal - vapor - vapors - nickel - operating - high - life

Support Structure airframe - extend - supported - vertically - cables - poles - windmill - vertical

This subtopic breakdown reveals the different application areas for rotors, such as wind, water, and geothermal energy. Quite a few subtopics describe innovations related to wind power, such as "Carbon Fiber" material for the blades, "Airfoils", "Turbine Blade Deflection", and "Tower" as a support structure for rotors.

Computing sub-topic strengths as previously done with top-level topics reveals interesting patterns in the technological development interest into rotors and how it changes over time. Figure 6.3 illustrates the average topic strengths for each subtopic.

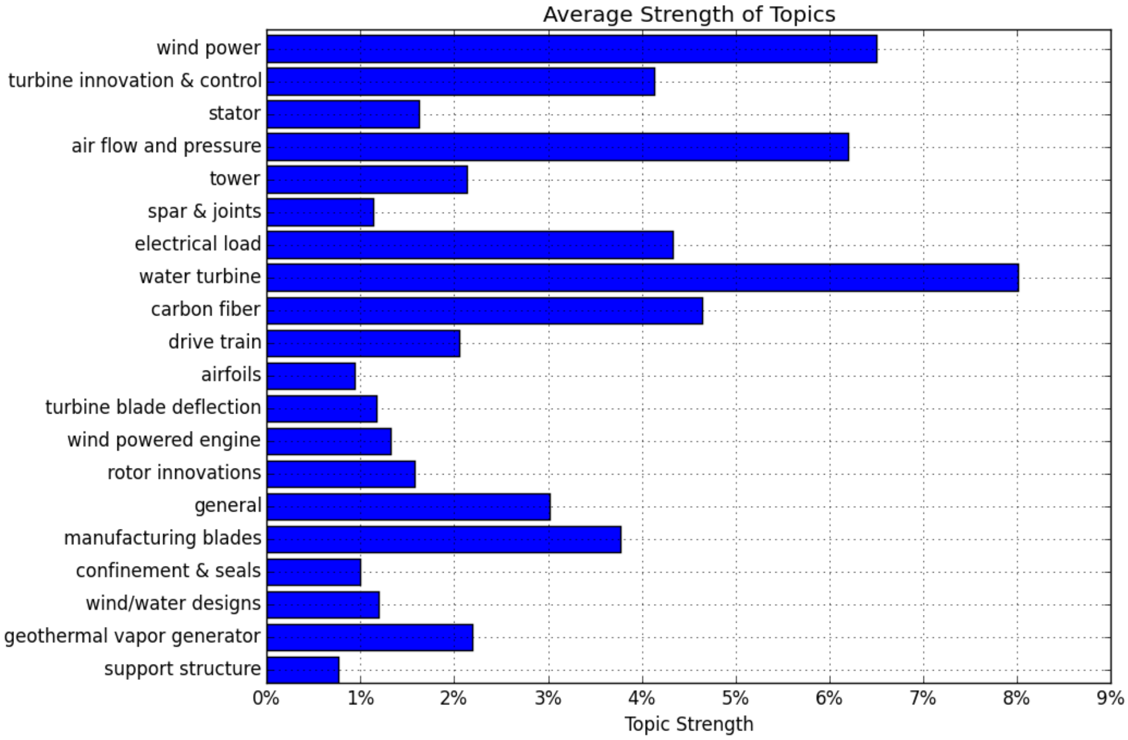

Figure 6.3: Average Topic Strengths of Top 20 Sub-topics of “Rotors”

“Wind Power”, “Water Turbines”, and “Air Flow and Pressure” are the strongest overall sub-topics, and the process of making turbines (“Carbon Fiber” and “Manufacturing Blades”) appears to be a strong interest.

Finally, consider Figure 6.4 which illustrates the fluctuations over each decade of the first 8 subtopics as returned by SPCA.

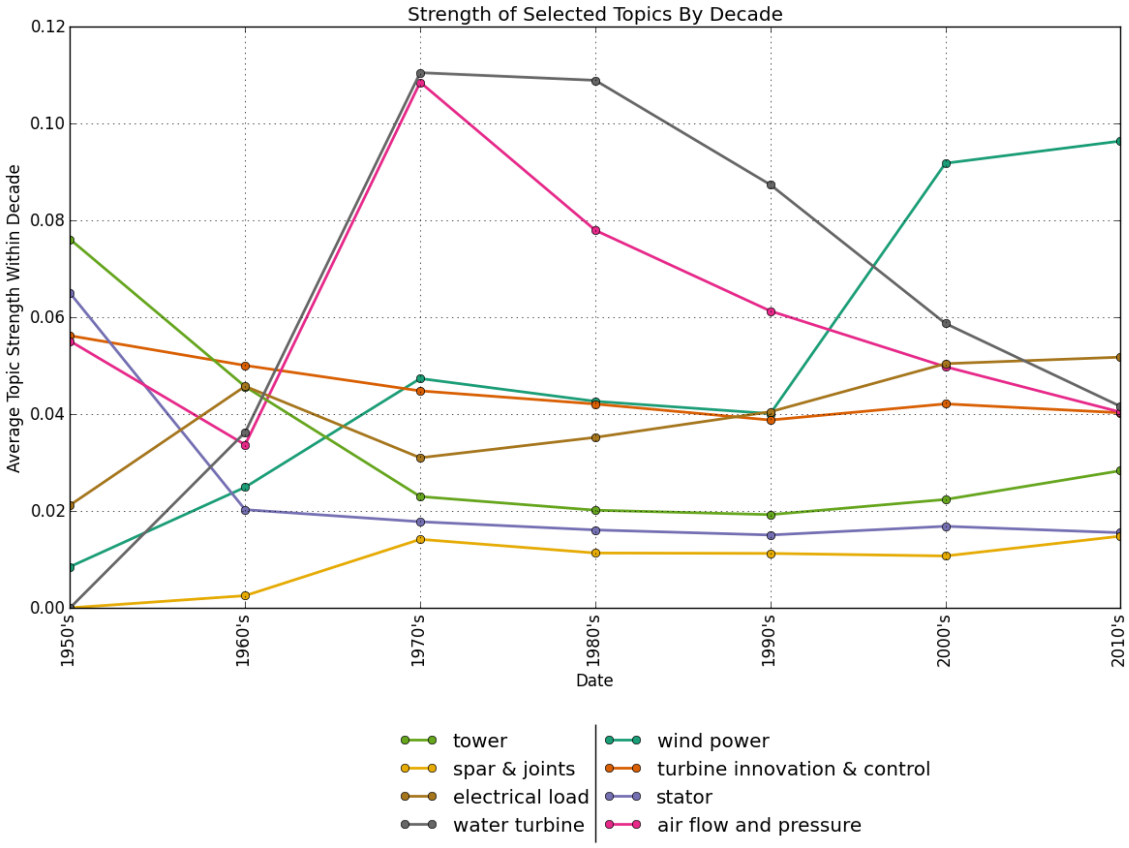

Figure 6.4: Average Strength of Sub-topics of “Rotors” by Decade

Observe that “Water Turbines” and “Air Flow and Pressure” are hot topics in the 70’s and 80’s, tapering off gradually thereafter. “Wind Power” gains strong interest in the 2000’s and 2010’s. “Towers” and “Stators” are strongest in the 50’s, tapering off since then.

The topic tagging methodology enabled a few graphs relating strengths of topics to one another and illustrating their evolution over time. Further, it allowed for a critical technological component for an interactive system: the ability to “zoom in” on a specific topic to investigate the context and find more specific details and relationships.

Chapter 7

Conclusion

The contributions introduced in this dissertation are directed towards the goal of a set of interactive tools for rich and insightful analysis of text content. Two generative probabilistic models of text, the Binary Independence Model and the Probabilistic Latent Semantic Indexing model, which are described in Chapter 2, are leveraged in Chapters 3 and 4 for efficient solutions to features selection and classification problems. The solutions offer robustness that proves important in unbalanced classification problems, and operate in time $O(nm \log(m))$, including model estimation. The abstractions represented by these probability models are lossy, requiring only $O(m)$ storage, where m is the number of features in the text corpus.

Chapter 5 describes the use of Sparse Principal Component Analysis for topic modeling in text, describing an existing fast, approximate algorithm, and contributing an analysis comparing the computational efficiency relative to Latent Dirichlet Allocation. With computational efficiency and suitability for an interactive system established, a method is introduced connecting the linear algebraic results to the Probabilistic Latent Semantic Indexing model. This approach leverages fast algorithms for robust estimation of categorical probability distributions to determine the strength with which a topic is expressed in a given document. The method handles documents of any length in a principled statistical domain, and employs the Hellinger distance to compare estimated distributions from documents to topic models extracted from Sparse Principal Component Analysis.

The utility of these methods is demonstrated in Chapter 6. While the models are significantly compressed representations of text, these abstractions enable very fast solutions to problems that are demonstrated to be effective in revealing practical, human-readable insights to queries across a wide variety of application domains.

Bibliography

[1] "Twitter Statistics," may 2013. [Online]. Available: http://www.statisticbrain.com/twitter-statistics/

[2] A. Agresti and B. A. Coull, "Approximate is better than "exact" for interval estimation of binomial proportions," *The American Statistician*, vol. 52, no. 2, pp. 119–126, 1998.

[3] R. Beran, "Minimum Hellinger Distance Estimates for Parametric Models," *The Annals of Statistics*, vol. 5, no. 3, pp. 445–463, 1977.

[4] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent dirichlet allocation," *the Journal of machine Learning research*, vol. 3, pp. 993–1022, 2003.

[5] S. Boyd and L. Vandenberghe, *Convex Optimization*, 2004, vol. 25, no. 3.

[6] E. J. Candes, X. Li, Y. Ma, and J. Wright, "Robust principal component analysis?" *Journal of the ACM*, vol. 58, pp. 1–37, 2011.

[7] C. J. Clopper and E. S. Pearson, "the use of confidence or fiducial limits illustrated in the case of the binomial," *Biometrika*, vol. 26, no. 4, pp. 404–413, 1934. [Online]. Available: http://www.jstor.org/stable/10.2307/2331986

[8] H. Cramér, *Mathematical methods of statistics.* Princeton university press, 1999, vol. 9.

[9] S. C. Deerwester, S. T. Dumais, G. W. Furnas, T. K. Landauer, and R. A. Harshman, "Indexing by latent semantic analysis," *JASIS*, vol. 41, no. 6, pp. 391–407, 1990.

[10] D. E. Duffy and T. J. Santner, "Confidence intervals for a binomial parameter based on multistage tests," *Biometrics*, pp. 81–93, 1987.

[11] C. Eckart and G. Young, "The approximation of one matrix by another of lower rank," *Psychometrika*, vol. I, no. 3, pp. 211–218, 1936.

[12] L. El Ghaoui, G.-C. Li, V.-A. Durog, V. Pham, A. Srivastava, and K. Bhaduri, "Sparse machine learning methods for understanding large text corpora," in *Proc. Conference on Intelligent Data Understanding*, 2011.

[13] L. El Ghaoui, V. Pham, G.-C. Li, V.-A. Duong, A. Srivastava, and K. Bhaduri, "Understanding Large Text Corpora via Sparse Machine Learning," *Statistical Analysis and Data Mining*, 2013.

[14] T. Fagan, "Exact 95% confidence intervals for differences in binomial proportions," *Computers in Biology and Medicine*, vol. 29, no. 1, pp. 83–87, 1999.

[15] L. A. Goodman, "On simultaneous confidence intervals for multinomial proportions," *Technometrics*, vol. 7, no. 2, pp. 247–254, 1965.

[16] T. L. Griffiths and M. Steyvers, "Finding scientific topics." *Proceedings of the National Academy of Sciences of the United States of America*, vol. 101 Suppl, pp. 5228–5235, 2004.

[17] I. Hacking, *The taming of chance.* Cambridge University Press, 1990, vol. 17.

[18] M. Hoffman, F. R. Bach, and D. M. Blei, "Online learning for latent dirichlet allocation," in *advances in neural information processing systems*, 2010, pp. 856–864.

[19] T. Hofmann, "Probabilistic latent semantic indexing," in *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval.* ACM, 1999, pp. 50–57.

[20] T. Joachims, "Text categorization with support vector machines: Learning with many relevant features," *Machine learning: ECML-98*, pp. 137–142, 1998.

[21] M. Journée, Y. Nesterov, P. Richtárik, and R. Sepulchre, "Generalized power method for sparse principal component analysis," *The Journal of Machine Learning Research*, vol. 11, pp. 517–553, 2010.

[22] R. Lebret and R. Collobert, "Word Embeddings through Hellinger PCA," in *EACL-2014*, 2014, pp. 482–490.

[23] J. J. Lee, D. M. Serachitopol, and B. W. Brown, "Likelihood-Weighted Confidence Intervals for the Difference of Two Binomial Proportions," *Biometrical journal*, vol. 39, no. 4, pp. 387–407, 1997.

[24] L. M. Leemis and K. S. Trivedi, "A comparison of approximate interval estimators for the Bernoulli parameter," *The American Statistician*, vol. 50, no. 1, pp. 63–68, 1996.

[25] D. D. Lewis, “Naive (Bayes) at forty: The independence assumption in information retrieval,” *Machine learning: ECML-98*, pp. 4–15, 1998.

[26] S. Li, Y. Ouyang, W. Wang, and B. Sun, “Multi-document summarization using support vector regression,” in *Proceedings of DUC*. Citeseer, 2007.

[27] J.-B. Michel, Y. K. Shen, A. P. Aiden, A. Veres, M. K. Gray, J. P. Pickett, D. Hoiberg, D. Clancy, P. Norvig, J. Orwant, S. Pinker, M. a. Nowak, and E. L. Aiden, “Quantitative analysis of culture using millions of digitized books.” *Science (New York, N.Y.)*, vol. 331, no. 6014, pp. 176–182, 2011.

[28] A. Moschitti and R. Basili, “Complex linguistic features for text classification: A comprehensive study,” in *Advances in Information Retrieval*. Springer, 2004, pp. 181–196.

[29] A. B. Murphy, “Rethinking multi-level governance in a changing European union: Why metageography and territoriality matter,” in *GeoJournal*, vol. 72, no. 1-2, 2008, pp. 7–18.

[30] C. H. Papadimitriou, H. Tamaki, P. Raghavan, and S. Vempala, “Latent semantic indexing: A probabilistic analysis,” in *Proceedings of the seventeenth ACM SIGACT-SIGMOD-SIGART symposium on Principles of database systems*. ACM, 1998, pp. 159–168.

[31] M. J. Paul and M. Dredze, “You are what you tweet: Analyzing Twitter for public health,” in *Fifth International AAAI Conference on Weblogs and Social Media (ICWSM 2011)*, 2011.

[32] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, “Scikit-learn: Machine Learning in {P}ython,” *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[33] K. T. Poole and H. Rosenthal, “Patterns of Congressional Voting,” *American Journal of Political Science*, vol. 35, no. 1, pp. 228–278, 1991.

[34] I. Porteous, D. Newman, A. Ihler, A. Asuncion, P. Smyth, and M. Welling, “Fast Collapsed Gibbs Sampling For Latent Dirichlet Allocation,” *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, p. 569, 2008.

[35] R. Rehurek and P. Sojka, “Software framework for topic modelling with large corpora,” in *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*. Valletta, Malta: ELRA, may 2010, pp. 45–50. [Online]. Available: http://www.muni.cz/research/publications/884893

[36] A. Ritter, C. Cherry, and B. Dolan, "Unsupervised modeling of twitter conversations," 2010.

[37] J. K. Rowling, "Harry Potter and the sorcerer's stone. New York: Arthur A," 1998.

[38] C. E. Shannon, "A mathematical theory of communication," *Bell System Technical Journal*, vol. 27, pp. 379–423, 1948.

[39] N. A. Smith, "Linguistic structure prediction," *Synthesis Lectures on Human Language Technologies*, vol. 4, no. 2, pp. 1–274, 2011.

[40] R. Tibshirani, "Regression shrinkage and selection via the lasso," *Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 58, no. 1, pp. 267–288, 1996. [Online]. Available: http://www.jstor.org/stable/10.2307/2346178

[41] H. Wang, "Exact confidence coefficients of simultaneous confidence intervals for multinomial proportions," *Journal of Multivariate Analysis*, vol. 99, no. 5, pp. 896–911, 2008.

[42] X. Wei and W. B. Croft, "LDA-based document models for ad-hoc retrieval," in *Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval.* ACM, 2006, pp. 178–185.

[43] J. Weng, E. P. Lim, J. Jiang, and Q. He, "Twitterrank: finding topic-sensitive influential twitterers," in *Proceedings of the third ACM international conference on Web search and data mining.* ACM, 2010, pp. 261–270.

[44] M. a. Woodbury, "Inverting modified matrices," Tech. Rep., 1950.

[45] H. Xiao and T. Stibor, "Efficient Collapsed Gibbs Sampling for Latent Dirichlet Allocation." *Acml*, pp. 63–78, 2010.

[46] Y. Zhang and L. El Ghaoui, "Large-scale sparse principal component analysis with application to text data," *Advances in Neural Information Processing Systems*, 2011.

[47] W. Zhao, J. Jiang, J. Weng, J. He, E. P. Lim, H. Yan, and X. Li, "Comparing twitter and traditional media using topic models," *Advances in Information Retrieval*, pp. 338–349, 2011.

[48] H. Zou, T. Hastie, and R. Tibshirani, "Sparse Principal Component Analysis," pp. 265–286, 2006.