UNIVERSITY OF CALIFORNIA,
IRVINE

Understanding Human Behavior Using Language and BOLD Variability

DISSERTATION

submitted in partial satisfaction of the requirements
for the degree of

DOCTOR OF PHILOSOPHY

in Psychology

by

Garren Gaut

Dissertation Committee:
Professor Mark Steyvers, Chair
Professor Michael Lee
Professor Padhraic Smyth

2018

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGMENTS

I would like to thank my advisor, Mark Steyvers for mentoring and supporting me throughout the program.

Thank you to my committee members for your support and feedback.

Thank you to my numerous collaborators from around the world: David Atkins; University of Washington ,Scott Brown; University of Newcastle,Pete Cassey; University of Newcastle,Eduardo Clark; National Digital Strategy Mexico,William Cunningham; Ohio State University,Jorge Díaz; National Digital Strategy Mexico,Rayid Ghani; University of Chicago,Zac Imel; University of Utah,Zhong-Lin Lu; Ohio State University,Xiangrui Lui; Ohio State University,Andrea Navarrete; University of Chicago,Padhraic Smyth; University of California Irvine,Brandon Turner; Ohio State University,Adolfo De Unánue; ITAM,Paul van der Boor; University of Chicago,Laila Wahedi; Georgetown University.

Thank you to Emily Grossman and Brandon Gaut for giving feedback on my work.

Thank you to my family for supporting me and keeping me happy; without your support I wouldn't have been able to even pursue a doctoral degree.

Thanks for funding: National Institutes of Health / National Institute on Alcohol Abuse and Alcoholism (NIAAA) under award number R01/AA018673 (David C. Atkins/Mark Steyvers, co-PIs) and National Institute on Drug Abuse under award number (R34/DA034860), Eric & Wendy Schmidt Data Science for Social Good Fellowship, and the National Sciences Foundation Integrative Strategies for Understanding Neural and Cognitive Systems Collaborative Research Grant (1533500 and 1533661).

CURRICULUM VITAE

Garren Gaut

EDUCATION

Doctor of Philosophy in Psychology — **2018**
University of California Irvine — *Irvine, CA*

Master's in Statistics — **2016**
University of California Irvine — *Irvine, CA*

Bachelor of Science in Applied Mathematics — **2011**
University of California Los Angeles — *Los Angeles, CA*

RESEARCH EXPERIENCE

Graduate Research Assistant — **2013–2018**
University of California, Irvine — *Irvine, California*

Data Science for Social Good — **Summer 2016**
University of Chicago, Chicago — *Chicago, Illinois*

Research Experience for Undergraduates — **Summers 2011, 2012**
University of California Los Angeles — *Los Angeles, CA*

Research Assistant — **2009-2012**
University of California Los Angeles — *Los Angeles, CA*

TEACHING EXPERIENCE

Teaching Assistant — **March 2017**
University of California Irvine — *Irvina, CA*

REFEREED JOURNAL PUBLICATIONS

Weusthoff, S, et al. "The language of interpersonal interaction: an interdisciplinary approach to assess and process acoustic and semantic data". **In Press**
European Journal of Counseling Psychology

Gaut, Garren, et al. "Content coding of psychotherapy transcripts using labeled topic models." **2017**
IEEE journal of biomedical and health informatics

Cassey, Peter J., et al. "A generative joint model for spike trains and saccades during perceptual decision-making." **2016**
Psychonomic bulletin

Barrow, D., Drayer, I., Elliott, P., Gaut, G. and Osting, B. Ranking rankings: an empirical comparison of the predictive power of sports ranking methods **2013**
Journal of Quantitative Analysis of Sports

Gaut, G., Goldring, K., Grogan, F., Haskell, C. & Sacker, R. Difference equations with the Allee effect and the periodic Sigmoid Beverton-Holt equation revisited. **2012**
Journal Biological Dynamics

REFEREED CONFERENCE PUBLICATIONS

Gaut, Garren, et al. "Improving Government Response to Citizen Requests Online". **In Press**
Proceedings of the First International Conference on Computing and Sustainable Societies, ACM

ABSTRACT OF THE DISSERTATION

Understanding Human Behavior Using Language and BOLD Variability

By

Garren Gaut

Doctor of Philosophy in Psychology

University of California, Irvine, 2018

Professor Mark Steyvers, Chair

This work consists of four projects that explore human behavior from two perspectives: language use and neural patterns.

In my first and second projects, I focused on language, which can be used to categorize human behavior. In the first project, I used topic models to categorize the subjects and symptoms a patient discussed during psychotherapy treatment. The model functions by identifying topics that are representative of each subject or symptom. The model can predict the subjects and symptoms discussed in new therapy sessions with higher accuracy than discriminative techniques. Furthermore, the model can identify specific passages of text representative of a given subject or symptom.

My second project developed an automated system for routing citizen requests to federal agencies within the Mexican government. The automated system functions by linking patterns in language and the appropriate federal agency. The automated system routes requests more efficiently than the current routing system.

The third and fourth projects focused on neuroimaging, which is used to understand the underlying neural processes associated with human behavior. My neuroimaging work related

blood-oxygen-level-dependent (BOLD) variability (BV) to experimental condition, behavior, and subject identity. The first phase of the neural work built on previous analyses showing that functional connectivity (FC) is predictive of the task a subject is performing and the identity of the subject performing a task. We extended these analyses to BV and compared its predictive accuracy with that of FC to assess whether some of the predictive power of FC is due to changes in BV. BV performed well compared to FC, suggesting that some of the predictive performance based on FC might be attributed to independent region-specific fluctuations.

Given the predictive relationship between BV and task/subject, the second phase of my neural work developed the Variance Design General Linear Model (VDGLM), a novel framework to facilitate the detection of BV effects. The framework models the mean and variance in the BOLD time course as functions of experimental design. This allows the VDGLM to i) simultaneously make inferences about a mean or variance effect while controlling for the other and ii) test for variance effects that could be associated with multiple conditions and/or noise regressors. We demonstrated the use of the VDGLM in a working memory application and showed that engagement in a working memory task is associated with whole-brain decreases in BOLD variance.

Chapter 1

Content Coding of Psychotherapy Transcripts Using Labeled Topic Models

1.1 Introduction

Across medical specialties, the basic medium of information gathering and intervention between the provider (i.e., MD, psychologist, nurse) and patient is a conversation. The patient describes problems and the provider listens, asks questions, and recommends solutions and specific treatment strategies. The content of this conversation can be useful across a broad variety of contexts, such as helping a primary care provider to detect and prevent suicide [33], promoting patient adherence to treatment recommendations [54], reducing cold severity and duration [112], and predicting a surgeon's history of malpractice lawsuits [2].

Psychotherapy (sometimes called counseling or behavioral treatment) represents a particular class of interventions that has a special focus on the provider-patient interaction. With psychotherapy, the interaction contains the treatment's active ingredients rather than simply being a means of developing rapport and forming a diagnosis. Psychotherapy ranges from brief, single session interventions [94] to multi-session interventions over weeks or months [17] and research suggests that psychotherapy is effective for a broad range of mental health disorders [100].

The typical method of summarizing the content of this conversation is based on the provider's recollection and self-report of what happened as they record it in the medical record. Many methods exist for obtaining summary measures from transcribed text–e.g., by separating a transcript into broad semantic topics [110, 29, 64, 118, 3, 68, 95], detailed behavioral features (such as requests for clarification [111]) or syntactic parts of speech [133], among others. These summary measures can be used as context to extract and evaluate treatment information, including patient diagnosis, analysis of client communication, and evaluation of suicide risk [105, 89, 119, 98, 14, 108].

At present the evaluation of psychotherapy sessions and other types of patient provider communication relies on human raters who summarize sessions by attaching codes (also called labeling or annotating) in order to quantify the information in treatment encounters [67]. The process of attaching these codes, called observational coding, provides theory-driven organizational systems through which complex linguistic data can be structured for further analysis. Codes can represent the subject of conversation (e.g., medications, spousal relationships), symptoms expressed (e.g., depression, anxiety, anger), or specific verbal behaviors in individual utterances or talk-turns for providers (e.g., open or closed questions by the therapist, degree of empathy) or patients (e.g., signaling intent to change or maintain behavior).

Observational coding has critical shortcomings, including intensive labor requirements, coder error, non-standardized coding systems (new codes require new training), and inability to scale up to larger coding projects [3]. Each hour of therapy takes roughly 10 hours to code and the number of alcohol and drug abuse sessions in the U.S. healthcare system alone runs into the hundreds of thousands per year. The burden of human coding leads typical psychotherapy research studies to be small, which contributes to the incredible heterogeneity across studies investigating the relationships between therapist behavior and patient outcome [141]. Accordingly, human-based coding is not a feasible method for evaluating the content of treatment encounters on a large scale. An objective, scalable method for summarizing the content of actual treatment encounters is needed.

We can describe the implementation of coding systems for text as multiple-label classification problems where multiple codes are attached to each document [44]. Machine learning approaches for automatic multiple-label document classification have been successfully used in various domains [122, 20, 103, 31, 8], including medical applications for disease diagnosis and medical error detection [82, 150, 86]. One such class of tools called topic models [83] has been used to assess the fidelity of therapist treatment [3] through prediction of behavioral codes, compare type of psychotherapy treatment [68], and predict therapy outcomes in schizophrenic patients [66, 65].

In this paper, we illustrate the ability of one specific type of topic model, Labeled Latent Dirichlet Allocation (L-LDA) [113, 118], to semi-automatically infer subject and symptom codes from a large heterogeneous psychotherapy corpus; i.e., what topics and symptoms were discussed during treatment. Every session in the corpus was manually annotated with general discussion content and patient symptom codes such that the observable outcomes of the manual annotation process are codes for the session as a whole. However, implicit in the coding process is a fine-grained, or local, evidence-accumulation process where each word,

utterance or talk-turn in a session affects the decision to attach a given code. Establishing a link between specific within-session passages of text and overall codes for the session (session-level codes) is fundamental to understanding the coding procedure. We implement a model that, in addition to learning session-level coding systems, can localize specific passages of text representative of a session code. In other words, the model is able to infer codes at a local (talk-turn) level from codes that were provided at the global (session) level.

Previous work on computational analysis of psychotherapy transcripts used topic models to summarize therapy corpora and extract features for use in predictive models for therapy type [68] or as a stand-alone model to predict behavioral codes [3]. Our current work expands upon past research by using topic models to predict session content, by providing a detailed quantitative evaluation of predictive performance that includes comparisons to baseline models, and by developing methodology for talk-turn annotation using session-level metadata.

The model is evaluated and compared against a baseline discriminative model (lasso logistic regression) using standard performance measure–the receiver operating curve (ROC) and area under the curve (AUC). Additionally, we provide R-precision [84] scores for talk-turn prediction. Session-level R-precision scores can be found in the supplementary files. For all performance evaluation, we use 10-fold cross-validation at the session level to emphasize the models' ability to predict novel data.

As we will discuss in the experimental results section, the accuracy of the proposed techniques, in terms of code prediction, are not yet at the level of human annotators. Thus, these approaches are not yet ready to be used for fully automated annotation of therapy transcripts in an off-the-shelf manner. Nonetheless, as we outline in the discussion section later in the paper, the current techniques could potentially be used as components within

a semi-automated approach, for example to assist in therapist training, using the model to rank and present to a supervising therapist specific talk-turns within a trainee session in terms of the talk-turn's likelihood of containing specific codes. There are multiple publicly-available L-LDA software packages [24, 101, 126] that could be used to support such efforts. More broadly, the work described in this paper represents the next step towards a long-term goal of fully automatic code prediction for psychotherapy transcripts.

1.2 Data

The primary source of data comes from a psychotherapy corpus maintained by Alexander Street Press and made available via library subscription. At the time of the present analyses, the corpus contained 1,181 therapy sessions with approximately 8 million words. Each session was conducted with a unique therapist and client. On average each session contains 250 talk-turns, which are defined as uninterrupted passages of time during which either the patient or therapist speaks. Talk turn length ranges from several words to several sentences. Sessions were conducted by prominent psychotherapists and serve as exemplars of different treatment approaches. Each session includes meta-data such as patient age, patient gender, type of psychotherapy, and two types of nominal content codes (i.e. labels) referring to subjects discussed in the session (161 possible codes) and patient symptoms discussed in the session (48 possible codes). We use subject and symptom codes because we are interested in the relationship between language and the codes' semantic meanings (as opposed to codes for type of therapy, client gender, etc). The list of symptom and subject codes was derived from the DSM-IV manual and other primary psychology/psychiatry texts. All codes annotated in the psychotherapy corpus are session-level codes, meaning that a single label is applied (as a binary present/absent label) to the entire session, and the original corpus did not

include any labels for specific subunits such as sentences, talk-turns or paragraphs. Each session is annotated with multiple codes (min = 1 code, max = 17 codes) and the average session is annotated with approximately 5 codes. Prior to analysis we applied a number of preprocessing steps, including stopword removal and n-gram extraction to convert the original corpus into a form suitable for text analysis. We chose stop words from standard lists used in natural language processing and augmented these lists with words from the corpus that were not on standard stop word lists, but that contain little semantic content (e.g., "mm-hmm") (see Models section for details on preprocessing and supplementary files for full stop word lists). Stop words were removed from both patient and therapist speech. In the case of a talk-turn comprised completely of stopwords, we removed the talk-turn from the data. The resulting representation of the text consisted of sparse vector counts of terms for each document, including unigrams (single words such as "medicine", "anger"), bigrams (e.g. "side effect"), and trigrams ("it sounds like").

In order to evaluate the ability of the model to find representative talk-turns we conducted additional coding to generate labels for talk-turns within selected sessions. The aim of the additional coding was to generate data for specific within-session sections of text (in this case talk turns) against which to test our model. These coded talk-turns were only used for model evaluation, not for model training. We focused on five symptom codes: anger, anxiety, depression, low self-esteem and suicide. These codes were chosen firstly for their therapeutic importance and secondly for their high frequency of annotation in order to provide a sufficient amount of additional data. We restricted the number of symptom codes to limit the amount of human coding required for talk-turn annotation. For each of these symptoms, we randomly selected 200 client talk-turns of at least 50 characters in length (before stop word removal) from sessions that had the symptom code attached. On average, the selected talk turns were approximately 277 characters in length before stop word removal. The process led to

a total of 993 talk-turns. Each talk-turn was rated in terms of the representativeness of the symptom on a scale of 1 (atypical) to 7 (very typical) by each of 6 graduate students or post-doctoral fellows with training in clinical/counseling psychology.

1.3 Models

We approach the problems of session coding and identifying representative talk-turns through the use of Labeled Latent Dirichlet Allocation (L-LDA) [113] [118], a semi-supervised extension of Latent Dirichlet Allocation (LDA). We first present the LDA model and then the L-LDA model. The model presentation is aimed at readers who have some experience with topic models. For readers new to topic modeling, we recommend reading a tutorial introduction [129]. Then, we show how these models can be used for document classification and how to apply the models to predicting codes and talk-turns in the general psychotherapy corpus. Finally, we present lasso logistic regression (LLR) as a baseline model against which to compare L-LDA.

1.3.1 Latent Dirichlet Allocation

LDA is an unsupervised modeling approach that learns a set of latent topics across a corpus of text. As opposed to L-LDA, there are no labels that are part of the data to learn from. The only data provided to LDA are a set of documents, where documents are treated as a "bag-of-words"; i.e., sparse vectors of word counts for each document. Thus, the order of words is not relevant for the model. We use both individual words and multi-word terms (n-grams) in the vocabulary for our model—but for simplicity will refer to both as "words."

Standard applications of topic models assume that the text corpus can be naturally divided into documents. For example, a corpus of scientific articles is naturally divided into documents according to article. In the case of spoken dialogue, choosing a rule for partitioning a corpus into documents is less straightforward. Documents can be defined as sentences, paragraphs, entire sessions or through any type of feasible partitioning. As in past research [68, 3], for the General Psychotherapy Corpus we define documents to be individual talk-turns (although other definitions are possible as well). Using talk-turns to define documents yields a larger set of documents with more localized word co-occurrences compared to defining documents at the session level. We have found in our experiments that these localized word co-occurrences tend to result in more specific topic-word distributions and improve classification performance.

LDA specifies a generative process for the creation of text documents. From this generative process we learn a predictive model by reverse-engineering the process–i.e., learning the parameters most likely to have generated the data. In LDA, each document (in this case talk-turn) is represented as a mixture of topics, where each topic is defined as a multinomial distributions over words. The creation of each document begins by sampling a document-specific distribution over topics. To generate each word in the document, a topic is sampled from the document specific-distribution over topics and a word is sampled from that topic. Formally, let T be the total number of topics in the model and V be the size of the vocabulary (number of unique words in the corpus). Then we can specify the marginal distribution over words for a document d as:

$$P(w) = \sum_{t=1}^{T} P(w|z_w = t)P(z_w = t|d).$$

where z_w indicates the topic from which word w was drawn, $P(w|z_w = t)$ is a V-dimensional distribution over words for topic t, and $P(z_w|d)$ is a T-dimensional distribution over topics for

document d. To simplify notation, we will let $\boldsymbol{\phi}^{(t)} = P(w|z_w = t)$ represent the distribution over words for topic t and $\boldsymbol{\theta}^{(d)} = P(z_w|d)$ represent the distribution over topics for each document d.

LDA incorporates *a priori* knowledge about topics likely to occur in a document by placing a Dirichlet prior on the distribution over topics, $\boldsymbol{\theta}^{(d)}$, for each document. The Dirichlet prior is the conjugate prior of the multinomial distribution and is used to express the prior probability of observing a topic in a given document before observing any data. The Dirichlet distribution is parameterized by the vector $(\alpha_1, ..., \alpha_T)$, where α_t can be interpreted as the prior observation count for the number of times topic t is sampled in a document before having observed any actual words from that document. Thus, we can view the distribution over topics for a document d as a sample from this group-level prior distribution over topics.

In a similar manner, LDA also incorporates prior information about which words are likely to occur in a given topic. LDA does this by placing another Dirichlet prior on the distribution over words, $\boldsymbol{\phi}^{(t)}$, for each topic t. This second Dirichlet distribution is parametrized by the vector $(\beta_1, ..., \beta_V)$ where β_w represents the prior observation counts of word w before observing any documents. Here we can interpret each topic as a sample from this group-level prior distribution over words. We follow the common practice of setting the Dirichlet parameters uniformly (i.e., $(\beta_1, ..., \beta_V) = (\beta, ..., \beta)$) which corresponds to the assumption that each word is equally likely a priori.

1.3.2 Labeled LDA Model

L-LDA is a semi-supervised variant of LDA in which some topics are placed in correspondence with labels that can be associated with a document. Documents in the training phase are assumed to have been pre-assigned to a subset of labels from a larger lexicon of possible

labels. In the context of the psychotherapy corpus, possible labels include symptom and content codes and L-LDA model infers a unique topic for each code. These topics are learned by restricting inference to only the word tokens in documents annotated with the topic's corresponding label. We use a separate unsupervised set of topics, called background topics, to account for words not associated with the known codes. These background topics allow the model to capture some of the linguistic variability in the data that is not directly related to subject and symptom codes. Without these background topics many words would have to be explained by the topics associated with the symptom and content codes, which would decrease the generalizability of those topics. During training of the L-LDA model, when sampling the topic for a word token in a document (as describe below), only topics that belong to labels associated with a document (including background labels) can be sampled. All other topics have zero probability of being expressed in the document.

Formally, let $T = T_c + T_b$ be the total number of topics. A subset of T_c topics are in one-to-one correspondence with the labels associated with documents. The remaining T_b topics capture background information. During the generative process, for each document d, we restrict the space of possible document mixtures by restricting the hyperparameters of the Dirichlet prior on $\boldsymbol{\theta}$ according to a binary topic assignment vector $\boldsymbol{\Lambda}^{(d)} = (\Lambda_1^{(d)}, ...\Lambda_T^{(d)})$. We define:

$$\boldsymbol{\Lambda}_t^{(d)} = \begin{cases} 1 & : (\text{code } t \text{ is attached to document } d) \text{ or } (t > T_c) \\ 0 & : \text{ otherwise} \end{cases}$$

We then define the hyperparameters for document d as $\boldsymbol{\alpha}_d = (\alpha_{d1}, ..., \alpha_{dT}) = \boldsymbol{\Lambda}^{(d)} \times \boldsymbol{\alpha}$. Note that the only topics that can be expressed for a particular document are topics corresponding

to a code associated with the document or background topics.

Letting D be the number of documents in the collection, the generative process of the L-LDA model can be described as follows:

1. For topic $t \in 1, ..., T$

a) Sample a multinomial distribution over words $\boldsymbol{\phi}^{(t)} \sim \text{Dirichlet}(\beta_1, ..., \beta_V)$

2. For document $d \in 1, ..., D$

a) Use the labels associated with document d to set the hyperparameters $\boldsymbol{\alpha}_d = \boldsymbol{\Lambda}^{(d)} \times \boldsymbol{\alpha}$.

b) Sample a multinomial distribution over topics $\boldsymbol{\theta}^{(d)} \sim \text{Dirichlet}(\boldsymbol{\alpha}_d = (\alpha_{d1}, ..., \alpha_{dT}))$.

c) For each term $i \in 1, ..., N_d$

(i) Sample a topic indicator $z_i \sim \text{Categorical}(\boldsymbol{\theta}^{(d)})$.

(ii) Sample a word token $w_i \sim \text{Categorical}(\boldsymbol{\phi}^{(t=z_i)})$.

where N_d is the number of word tokens in document d. Note that $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ are hyperparameters for the model. The graphical model for L-LDA is presented in Figure 1.

1.3.3 Training the L-LDA Model

The variables we would like to infer are the topic assignment variables z_w for each word w, the document mixtures $\boldsymbol{\theta}^{(d)}$, and the topic distributions $\boldsymbol{\phi}^{(t)}$. For sampling we use a collapsed Gibbs sampler [59] which integrates out $\boldsymbol{\phi}^{(t)}$ and $\boldsymbol{\theta}^{(d)}$ so that we only sample the topic assignments z_w. The topic assignments z_w are then used to generate point estimates of $\boldsymbol{\phi}^{(t)}$ and $\boldsymbol{\theta}^{(d)}$.

The Gibbs sampling procedure considers each word token in the text collection in turn, and estimates the probability of assigning the current word token to each topic, conditioned on

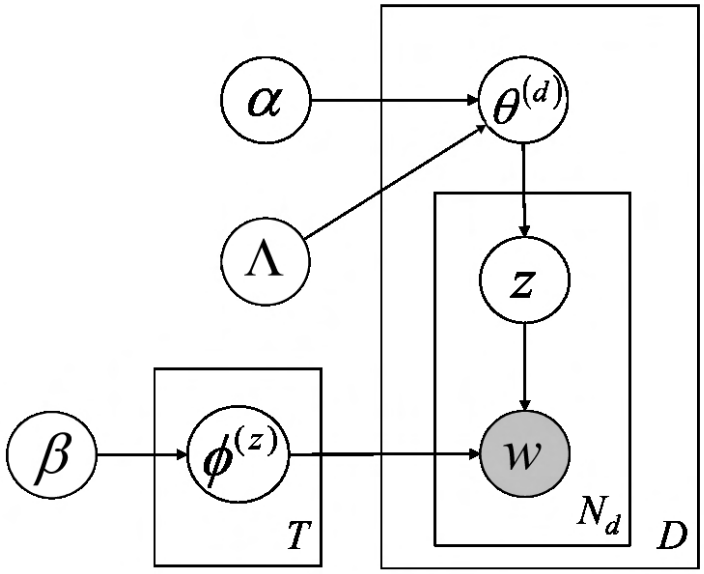

Figure 1.1: Graphical Model of L-LDA.

the current topic assignments to all other word tokens. From this conditional distribution we sample a topic assignment for the current word token. We write the conditional distribution as $P(z_i = t|\mathbf{z_{-i}}, w_i, d, \cdot)$ where $z_i = t$ represents the topic assignment of token i to topic t, $\mathbf{z_{-i}}$ refers to the topic assignments of all other word tokens, and "$\cdot$" refers to all other known or observed information such as all other word indices $\mathbf{w_{-i}}$, distributions over topics for all other documents, and hyperparameters $\boldsymbol{\alpha}$, and $\boldsymbol{\beta}$. The conditional distribution can be calculated as follows [59]:

$$P(z_i = t|\mathbf{z_{-i}}, w_i, d, \cdot) \propto \frac{C^{VT}_{w_i t} + \beta_{w_i}}{\sum_{w=1}^{V}(C^{VT}_{wt} + \beta_w)} \cdot \frac{C^{DT}_{dt} + \alpha_{dt}}{\sum_{j=1}^{T}(C^{DT}_{dj} + \alpha_{dj})} \tag{1.1}$$

where t is restricted to the set of topics defined by the union of (a) codes t attached to document d, and (b) background topics $t > T_c$. All other topics have probability 0 for

document d (as specified by the generative model) and are not eligible to be sampled. In the equation above C^{VT} and C^{DT} are matrices of counts with dimensions $V \times T$ and $D \times T$ respectively; $C^{VT}_{w_i t}$ contains the number of times word i occurred in a document with topic t and C^{DT}_{dt} contains the number of times a word token in document d was assigned to topic t. These matrices are incremented using the sampled topic assignment variables at each step of the Gibb's sampler for every word w.

The Gibbs sampling algorithm is initialized by assigning each word token in document d randomly to one of the set of eligible topics for document d (i.e., the codes t attached to document d or the background topics $t > T_c$). For each word token, the count matrices C^{VT} and C^{DT} are first decremented by one for the entries that correspond to the current topic assignment. Then, a new topic is sampled from the distribution in Equation 1 and the count matrices C^{VT} and C^{DT} are incremented with the new topic assignment. Each Gibbs sample consists of the set of topic assignments for all N word tokens in the corpus, achieved by a single pass through all documents.

The sampling algorithm gives us samples for the topic assignment variables z_w for each word w. However, we are interested in estimating the word-topic distributions $\boldsymbol{\phi}^{(t)}$ and topic-document distributions $\boldsymbol{\theta}^{(d)}$. We can approximate the probability of choosing the k-th word from the distribution over words for topic t, $\boldsymbol{\phi}^{(t)}$, using the word-topic count matrix (computed from the sampled topic assignment variables) as follows:

$$\hat{\phi}^{(t)}_k = \frac{C^{VT}_{w_k t} + \beta_{w_k}}{\sum_{w=1}^{V} (C^{VT}_{wt} + \beta_w)}.$$

Here $\hat{\phi}^{(t)}_k$ can be interpreted as the estimated probability of choosing word w_k from topic t. We can also estimate the probability of choosing the t-th topic from the distribution over topics for document d, $\boldsymbol{\theta}^{(d)}$, using the count matrix C^{DT} (also computed from the sampled

topic assignment variables) as follows:

$$\hat{\theta}_t^{(d)} = \frac{C_{dt}^{DT} + \alpha_{dt}}{\sum_{j=1}^{T} (C_{dj}^{DT} + \alpha_{dj})}.$$

Here $\hat{\theta}_t^{(d)}$ can be interpreted as the estimated probability of expressing topic t in document d. We later use $\hat{\boldsymbol{\phi}}^{(t)}$ to qualitatively examine topics corresponding to session codes and $\hat{\boldsymbol{\theta}}^{(d)}$ to estimate the topics (and therefore symptom and content codes) expressed in document d.

1.3.4 Prediction with the L-LDA Model

We evaluate the model by predicting labels for documents unseen by the model during training using the word-topic counts (C^{WT}) learned during training. The goal for prediction is to infer a document-topic count vector $C_{d't}^{DT}$ for each new document d' , where the inferred count vector contains information about the likely topics (and associated codes) for d'.

For a new document d', we set $\Lambda_t^{(d')} = 1 \; \forall \; t \in \{1, ..., T\}$ so that any topic can be part of the document mixture. We run a Gibbs sampling procedure where we compute the posterior distribution over topic assignments:

$$P(z_i = t|\mathbf{z}_{-\mathbf{i}}, w_i, d', \cdot) = \frac{C_{w_i t}^{WT} + \beta_{w_i}}{\sum_{w=1}^{W} (C_{wt}^{WT} + \beta_w)} \cdot \frac{C_{d't}^{DT} + \alpha_t}{\sum_{j=1}^{T} (C_{d'j}^{DT} + \alpha_j)}. \tag{1.2}$$

where $\alpha_t = \alpha$. The posterior for this sampling procedure is similar to the posterior used in the sampling procedure during training except that the word-topic count matrix C^{WT} is not

updated. Holding C^{WT} constant formalizes the assumption that the word-topic counts are learned and that prediction consists of learning just the document-topic counts. Another difference from the sampling procedure used during training is that we sample the posterior probabilities $P(z_i = t|\mathbf{z_{-i}}, w_i, d_i, \cdot)$ at each iteration (after burn-in) instead of the word-topic count assignments (that were sampled during training). While either word-topic counts or posterior probabilities can be used to compute prediction scores, we found that using posterior probabilities provided more accurate code predictions and required less samples for accurate prediction. We use the posterior samples to compute topic scores that represent the likelihood that a document should be annotated with the code corresponding to each topic. We compute a score $\eta_{t,d}$ for each topic t and test document d as follows:

$$\eta_{t,d} = \frac{1}{N_d} \sum_{i=1}^{N_d} \gamma_{t,d,i}$$

where the variable $\gamma_{t,d,i}$ estimates the probability in Equation 2 that the tth topic was assigned to the ith word token in document d. Thus $\eta_{t,d}$ can be interpreted as the average probability of assigning a word from document d to topic t. To calculate each word's posterior estimate of topic assignment ($\gamma_{t,d,i}$) we average over the posterior samples of the probability of assigning word i to topic t. We compute $\gamma_{t,d,i}$ as follows:

$$\gamma_{t,d,i} = \frac{1}{K} \sum_{k=1}^{K} p(z_{t,d,i})^k$$

where $p(z_{t,d,i})^k$ is the k-th sample of the posterior probability expressed in Equation 2 and K is the total number of samples.

1.4 Learning Topics from Labeled Sessions with L-LDA

1.4.1 Text Preprocessing

The original corpus contained 8 million word tokens and 40,000 unique words. Before fitting the L-LDA model, we applied a number of preprocessing steps on the corpus. We first removed any words that occur 5 or fewer times in the entire corpus on the assumption that these words are unlikely to be useful in general for categorization. This step reduced the number of unique words from 40,000 to 27,000 unique words. After removing infrequent words, we removed words that we thought contained little semantic content. We performed a preliminary filtering using a common stop word lists to remove words (see unigram stop word list in supplementary files) Next, we applied a part-of-speech tagger [134] that we used to remove determiners, adverbs, pronouns, interjections, particles, modal words, punctuation, and numbers. We used part-of-speech tags to create additional stop word lists for bigrams and trigrams, and performed a second stop word filtering using these lists. A final stop word filtering was done for interjections that are common in psychotherapy, but weren't identified by the part-of-speech tagger. See supplementary files for a full list of stop words. The final corpus contained 28,000 unique words (including generated bigrams and trigrams) and 1.4 million word tokens.

1.4.2 Model Parameters for Training L-LDA

The L-LDA model requires a number of decisions to be made and parameters to be selected before training the model, including the number of background topics T_b, the settings for the priors $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$, the number of iterations and the number of burn-in samples.

For the number of background topics T_b, we chose $T_b = 50$ for the results reported in this paper, and found that the model was not particularly sensitive to the number of background topics as long as T_b was at least 20. We used uniform $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ where each element was set to 1/50 and 1/100, respectively. These are typical values used in training LDA models and we found that that the method was reasonably robust to small perturbations in these values. Our results are from a model using 100 training iterations, and 20 iterations for prediction, where the last $S = 10$ iterations are used for generating prediction scores. We ran several models that varied the number of iterations and burn-in samples and found results similar to the model we report.

1.4.3 Inferred Topics

Prior to assessing predictive performance measures, we qualitatively examined the topics generated by the L-LDA model (Table 1.1). We examined three types of topics corresponding to subjects, symptoms, and background content. For the subject and symptom labels, we illustrate the topics learned by the model for the five most common labels. For the background topics, we picked an illustrative set of five topics. Qualitatively, subject and symptom topics are very interpretable–e.g., the medications topic consists of examples of medications, words used to describe administration of medication, and words used to describe the effects of medication. The background topics shown in Table 1.1 also have intuitive interpretations and contain words that are not covered by the content codes in the psychotherapy corpus. For example, there are background topics that explain word usage related to people, jobs, and sleeping (background topics 9, 36, and 39 respectively). Without these background topics, the high probability words associated with them would have to be redistributed over the content topics for subjects and symptoms, potentially decreasing their interpretability and predictive power.

Table 1.1: The most likely terms inferred for the topics associated with the five most common subject and symptoms and an illustrative set of background topics. For each topic, the 10 most likely n-grams are shown.

Subject	**Inferred Topic Distribution**
medications	medicine, mg, dose, wellbutrin, medicines, lamictal, prescription, effects, side_effects, ability
relationships	relationship, women, feels, friend, relationships, boyfriend, date, position, example, react
parent-child relations	mother, father, love, remember, relationship, parents, brother, emotional, loved, needed
depressive disorder	depression, medication, doctor, medicine, prozac, depressed, zoloft, generic, wellbutrin, add
spousal relationships	wife, marriage, married, husband, relationship, mhm, children, attitude, divorce, got_married
Symptom	**Inferred Topic Distribution**
anxiety	anxiety, anxious, panic, nervous, depression, worried, worst, fine, experience, helps
depression	depressed, depression, doctor, pain, die, needed, drugs, low, xanax, mg
anger	angry, feelings, anger, express, get_angry, be_angry, reaction, feels, pissed, 'm_feeling
low self-esteem	love, teaching, boyfriend, positive, stupid, attractive, fit, negative, sorta, criticism
irritability	annoyed, irritable, message, safe, dishes, cause, wife, skin, irritated, cats
Background	**Inferred Topic Distribution**
background 9	friends, family, mom, dad, close, sister, brother, daughter, men, lives
background 13	care, stop, took, weight, takes, ready, lose, take_care, amount, body
background 23	house, room, walk, bed, door, walking, rid, front, throw, clean
background 36	job, wants, work, business, works, office, busy, baby, buy, paper
background 39	morning, sleep, hours, friday, sleeping, monday, tomorrow, saturday, wake, bed

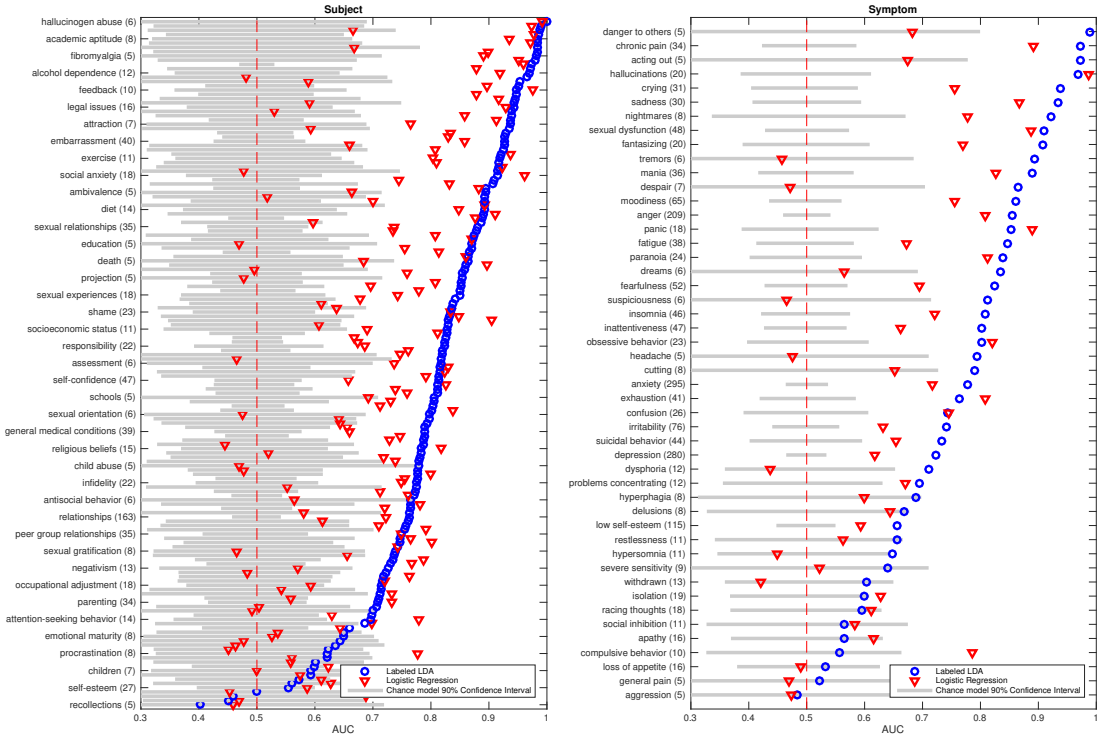

Figure 1.2: Session-level AUC scores for the labeled topic model, the lasso logistic regression model, and chance performance. Codes are reported along the y axis and are ordered by labeled topic model performance. For subject codes, one in every 4 codes names is shown.

1.5 Session-Level Prediction

1.5.1 Cross-Validation and Scoring

To test generalizability of the model to new data, we use 10-fold cross-validation where for each fold the sessions are partitioned into two disjoint sets: (a) a training set with 90% of the sessions used to train an L-LDA model and (b) a validation set with the other 10% of sessions used to evaluate the trained model. We compute an AUC score for each validation set (for each code) and report the average of the AUCs across the 10 validation sets.

To compute an AUC score for a topic corresponding to a particular code and a particular validation set we proceed as follows. For each session in a validation set, we predict scores at the talk-turn level (as described earlier) and then aggregate scores for all talk-turns in a session. We define the session likelihood score for session s and topic t as:

$$\eta_{t,s} = \frac{1}{D_s} \sum_{d \in d^{(s)}} \eta_{t,d}$$

where D_s is the number of talk-turns in session s and $d^{(s)}$ is the set of all talk-turns (documents) in session s. For each topic t, using these scores, we rank the sessions in the validation set and compute the area under the curve (AUC) using the known subject and symptom codes attached to each session.

1.5.2 Results for Session-Level Predictions

For each subject and symptom code we computed the AUC for each cross-validation fold and took the average across folds to measure classification performance. Values of the AUC range in theory from .5 (chance level performance, e.g., randomly generated rankings) to 1 (perfect predictive accuracy). In practice, performance that is above the level of chance can occur even from models where the scores are randomly distributed (and unrelated to the content of the sessions). This is especially the case with codes that occur infrequently. In order to assess the significance of the predictive accuracy of our model relative to chance performance, we calculated a set of AUC scores for each code using 1000 randomly generated rankings and computed the corresponding 90% confidence intervals.

Additionally, we compare the L-LDA model to a standard machine learning classifier, lasso logistic regression (LLR). LLR is often used in classification settings where the number of

features is larger than the number of observations because of it's ability to force feature weights to zero for uninformative features. For more details about LLR see Supplementary Files.

The results of the AUC analysis are shown in Figure 1.2. The widths of the 90% chance confidence intervals for each code correspond closely with the inverse of the code frequencies (lower frequencies are associated with larger chance confidence intervals). The L-LDA model showed higher predictive accuracy than the LLR model and both models performed significantly better than the chance model for a large number of codes. For the L-LDA model, the average AUC score over all codes is .789 (SD=.137) and average AUC for subject and symptom codes are .800 (SD=.131) and .753 (SD=.150), respectively.

All but 10 of 209 codes had AUC scores above .5. The 5 codes with lowest AUC scores are gender roles, withdrawn, recollections, general pain, and self-fulfilling prophecy. The language associated with each of these codes contains a broad spectrum of variation that may have contributed to poor model performance. The 5 codes with highest AUC are hallucinogen abuse, drug addiction, alcohol dependence, passiveness, and attraction.

For the LLR model, average AUC score over all codes is .702 (SD=.145) and average AUC for subject and symptom codes are .713 (SD=.146) and .667 (SD=.137), respectively. There were 29 codes with AUC scores below 0.5. Overall, the LLR model performed significantly worse on average than L-LDA ($p<.001$ in a Wilcoxon sign test).

A common goal of document classification is to identify the relationships between specific classifiers and characteristics of data that lead to high classification performance. Previous comparisons between L-LDA and discriminative models have shown that the L-LDA model can outperform discriminative models on low-frequency codes [118]. We analyzed this relationship on the general psychotherapy corpus and found only a weak correlation (R=0.22)

between the AUC difference for the two models and code frequency. This correlation showed that L-LDA model performs slightly better in comparison to LLR at predicting low frequency codes than high frequency codes. Post-hoc qualitative analysis suggests that highly predictable codes contain unique language that facilitate prediction. For example, sessions that discuss hallucinogen abuse and drug addiction contain a range of drug-specific terms that are highly specific. Conversely, we expect that hard to predict codes, such as gender roles, are attached to sessions containing a broad spectrum of language.

1.6 Talk-Turn Prediction

1.6.1 L-LDA Talk-Turn Prediction

As a second test of performance, we assessed the ability of the L-LDA model to find talk-turns that are representative of a session-level code. This comparison is novel in that the L-LDA model is trained using only session-level codes, but can then generalize the topics learned to identify representative talk-turns within each session. The evaluation procedure tests the models' abilities to distinguish the most representative talk-turns (as judged by human raters) from all other talk-turns.

We had 6 human coders generate ratings at the talk-turn level for 993 talk-turns using 5 symptom codes chosen from the set of general psychotherapy codes. The codes used were *anger, anxiety, depression, low self-esteem, suicidal behavior*. Each talk-turn was assigned a continuous rating from 1 (atypical) to 7 (very typical) by each of the 6 coders. To keep model performance measures on the same scale as the session-level performance measures, we converted the continuous human ratings to binary scores (thus allowing us to compute classification performance measures). To binarize ratings, we chose a rating threshold and

considered any ratings above the threshold to be representative of a symptom and any ratings below to be not representative. While there are many ways of choosing this threshold, we chose the threshold such that the top $c\%$ of ratings would be considered representative. We computed performance for $c = \{5, 10, \text{ and } 20\}\%$ to emphasize the model's ability to predict highly representative talk-turns.

Since raters did not rate talk-turns for the other symptom codes in the psychotherapy corpus, we created a mapping from the more detailed labels in the psychotherapy corpus to the five selected symptom codes. The motivation behind creating these code mappings is that a single symptom code (e.g., depression) might be aptly described by multiple codes in the psychotherapy corpus (e.g., depression, depressive disorder, hopelessness, ...). To create the mappings, we had a clinical psychologist mark which codes from the general psychotherapy corpus are related to each of the five symptom codes. See Appendix A.1 for more detail.

In addition to AUC scores we report the R-precision. R-precision is a measurement of precision at the threshold at which precision is equal to recall. To generate model predictions, AUC scores, and R -precisions at the talk-turn level we proceeded as follows:

- An L-LDA model was trained on each of the 10 training data sets used for session-level cross-validation. For each training set any session that contained any of the coded talk-turns was removed (making the prediction problem somewhat more difficult by not allowing the model access to the coded talk-turn nor any other talk-turns from the same session). We remove these talk-turns to avoid optimistic performance results since in application the model would be identifying codes for talk-turns from novel sessions.

- Each of the 10 trained models made predictions on the 993 labeled talk-turns. The $\eta_{t,d}$ scores were computed for each general psychotherapy code t and each talk-turn

(document) d as described earlier, using each model's word-topic count matrix. To compute a score for a symptom code, we averaged the model scores from each of the related general psychotherapy codes (as defined by the code mapping described above). The 10 scores for each code and each talk-turn were then averaged across the 10 trained models.

- For each code, AUC scores were generated as follows. The 993 talk-turns were ranked by their averaged model-based scores. These rankings were then compared to the ratings from each individual rater, where the ratings were binarized by using the highest 5%, 10%, top 20% of that individual's ratings, leading to 3 different AUC scores, one for each percentile cutoff. Overall AUC scores for the model, for each of the 3 cutoffs and each of the 5 codes, were then computed by averaging across the model's AUCs computed relative to each individual rater.

- For each code, R-precisions were generated as follows. The 993 talk-turns were ranked by their averaged model-based scores. These rankings were then compared to the ratings from each individual rater, where the ratings were binarized by using the highest 5%, 10%, and 20% of that individual's ratings. For each rater and rating cutoff $c \in \{5, 10, 20\}$, we compute the R-precision as the number of true positives in the top c% of ratings divided by the number of talk-turns in the top c% of ratings. The R-precision ranges from 0 to 1 and it can be shown that the R-precision is equal to recall for the top c % of ratings. We compute overall R-precision scores as the average of model R-precision scores computed relative to each individual rater.

1.6.2 Results for Talk-Turn Predictions

Table 1.2 shows example talk-turns for all 5 symptoms tested. Talk-turns are ordered by model representativeness score. We also report the human representativeness rating (1-7 scale) averaged across raters. Several talk-turns illustrate that the model learns words associated with a symptom and not just the symptom keyword itself. For example, the first example talk-turn for depression in Table 1.2 is rated by the model as most representative and is also judged by humans as highly representative. This talk-turn does not contain the word depression but only expressions related to depression (i.e. "I'm crying"). The first talk-turn for anxiety presents another interesting example. It is given the highest representativeness score by the model but only received a low human rating. The model may have learned to associate the word "roommate" with anxiety (through the other sessions in the training set), resulting in a high likelihood.

Table 1.2: Model predictions for most representative talk-turns for each symptom code. Talk-turns are ordered by model score Average human Likert rating (1-7) is reported to compare model scoring vs. human scoring.

Symptom	**Average Rating**	**Talk-Turn**
Anger	6.2	Nobody every got angry; they never got angry. I don't ever remember my parents screaming at each other, ever. I mean throughout all my childhood I can't remember them having a yelling fight. it was never that way. and I just never knew how to scream at anybody.

7 I have occasionally felt bad about the things I've told you about, I have. but it's interesting that this is the first time that a lot of anger has come out. you know, there's another side that I really affirm here, that there's a lot of anger that I have toward her that she's always been able to seem to get out and express at me.

6.8 I don't know. but I didn't get mad at Harold when he gave me genital warts. I felt mad, I mean, I felt betrayed and lied to and cheated on, but I didn't - I just dealt with it, I just deal with things, and I've always thought that that was a positive quality, I mean, I just-i don't think that anger is necessarily productive. but I guess in some ways it can be. I just-i work through things, I talk through things, I'm calm, I don't get mad or yell and scream. if i-you know, I can argue with people if I don't - you know, it's not like I won't express my opinions or, you know, talk about something that bothers me. but I don't yell and scream and I don't get angry.

7 At night, and then I take my zyprexa and I fall asleep in two hours. the one thing I'd say I notice about her is she will be talking like this and then all of a sudden I don't what happens, something happens and she just gets real angry, real fast, like that. we will be talking and all of sudden she will think of something that got her angry and it will be like boom.

3.6 Even just now, when you ask me that, I don't know, it just feels like, why are you asking me these questions? I don't understand them. I feel like ... it's just really uncomfortable.

Anxiety	2	The only-the only thing I can-like I thought back to this. when I was a senior in college I met a girl who was a roommate of my roo-well, my roommate's - my roommate and i-my roommate had his fiance and she had a roommate. and this, anyhow, to make it all work ....
	3.2	When I got there, he said that I needed to go to the hospital. so I went, he sent me to ... when I got to ..., they did another ekg. they told me I had a heart attack 2 weeks before that.
	7	And, um, had a little anxiety about it. I go 2 nights a week, monday and wednesday from 6 to 10, and yeah, had a little anxiety attack about it 'cause just the whole like possible failure, and like oh god, I'm like I really want, 'cause I really want it and I'm really, you know, I'm good at it, but it's like oh god, it's pressure, you know, that type of thing. well, I ended up calling ... remember I was seeing him?
	5	I don't really know. I've always been kind of just like - I'm always just really scared of - I don't really have like a lot of, in my family there's not really a lot of people that would help me if something like that was to happen. so I think that just kind of like fuels this like fear in me with like employment in general. it's just kind of like, ' well what if there is a cutback or what if somebody buys us out or ... ' just kind of like, I just want to be okay if that happened.

	2.2	Yeah. I think I just ... I never ... I don't know. since probably before I came to shimer was the last time I actually like really either showed interest in a guy. like even if I was interested, I haven't within the past three years, like done anything about it really. brendan, close, like we've actually kissed and ... but like ... that was.
Depression	6	As like two to three months ago, I was crying and this was more or less yknow I didn't want to be doing this but now I'm crying, I'm like, I don't care that I'm crying, yknow.
	5	Well I think one of the things I wanted to ask you about was what we talked about last week the matter of guilt when I touched on that briefly. I'm a little bit confused because it seems to me that a person has desires to kind of change their ways as it were that one of the motives of them wanting to do that is them some feeling of guilt or something approximating guilt about the way they're presently acting. and yet you said that you thought that I should feel that way, people in general too, but in this particular case me should not feel guilty for example about vanity because I've done that. so you think that they should feel that way but it seems to me that one of the motivating forces for me is a certain sense of guilt. or not exactly guilt but maybe something like ... well I suppose it is guilt the guilt of throwing away a good part of my life. I feel guilty about that even in a moral sense as well as a practical one. so what do you mean by that? how do you work around ... .

	6.6	Um, the pamelor 50, I take that at night, that seems to be doing okay. I mean I'm still a little depressed but, you know, basically that seems to be doing okay. nr : doctor, patients are leaving.
	3.6	It might be. I guess I feel the anger because it, like a given situation turned out unhappy or sad instead of happy.
	3.6	And yknow, be supportive for my nephews and my nieces and I found myself kind of leaning with my dad in just being sad, just being, yknow, it was just different.
Low Self-Esteem	5	Umm ... well certainly if I'm not being obsessive and worrying over things. because the truth is, it's obvious when I'm in that state, even if I don't tell people. you can see it all over my face. and he notices. and so if I'm not that way, and if I'm confident and mature.
	4.2	Which probably isn't a good thing. but I just don't ' do it. and they tell me," you should do it. you should do it. you should do it. ". and I say, notes don't work for me. I make up excuses. I tell them that, no, that's not me. leave me alone. let me do my own thing. and that's sort of me not taking criticism well.

3.4	It's funny, because when I talk about my relationship with my parents with cole it's always that I'll say something negative and I'll say, "but I know they love me, but-but-but-but," you know? like I'll always have an aside for "yeah, like, but it's okay because ... " right, I guess I feel like it's not okay to be. I know it is okay to be angry at times. and of course now I'm thinking, but I don't blame them, they're who they are, you know? yeah, I always have to have an excuse for them. but -.
2.6	I think I mentioned it a little bit earlier but like, I was talking to my friend about it. I mean he - we were sort of sitting in the lunchroom and he saw a girl that he thought was attractive walk by and he was like, "wow, she's a really attractive girl. ". and I was like, "eh. ". he's like "what, you don't think she is? ". and I was like, "yeah, but who cares? ". that's just sort of been my sort of feeling lately, that it's like yeah, she's an attractive girl. whatever.
2.6	No.. if someone is looking yeah because well first of all one of the answers that I think that I can't find is like this question of what is a man looking for?

Again, we compare the L-LDA model to lasso logistic regression. Table 1.3 shows a table of AUC scores for the L-LDA and LLR model with 5%, 10%, and 20% cutoffs used to create binary scores for the human representativeness ratings. For the L-LDA and LLR models, we computed an AUC score for the model relative to each individual rater and then averaged the AUCs across raters. To compare the models against human raters, we also calculated a human reliability score to serve as a measure of inter-annotator agreement.

Table 1.3: Talk-turn coding performance for the L-LDA model and Human Reliability scores. AUC and R-precision scores are shown for the top 5%, 10%, and 20% of talk-turns as rated by human coders. Human reliability is expressed in AUC and R-precision scores to enable direct comparison to model performance.

		AUC at 5%			**AUC at 10%**			**AUC at 20%**		
Code	**No. TTs**	**L-LDA**	**Human**	**L1 LR**	**L-LDA**	**Human**	**L1 LR**	**L-LDA**	**Human**	**L1 LR**
anger	197	0.89	0.94	0.91	0.84	0.94	0.85	0.75	0.87	0.73
anxiety	200	0.76	0.80	0.77	0.72	0.78	0.72	0.66	0.72	0.64
depression	198	0.73	0.87	0.74	0.67	0.84	0.73	0.66	0.84	0.70
low self-esteem	200	0.70	0.82	0.74	0.64	0.81	0.68	0.60	0.78	0.63
suicidal behavior	198	0.78	0.96	0.77	0.70	0.87	0.70	0.67	0.82	0.67
average		0.77	0.88	0.79	0.71	0.85	0.73	0.67	0.81	0.67
		R Precision at 5%			**R Precision at 10%**			**R Precision at 20%**		
Code	**No. TTs**	**L-LDA**	**Human**	**L1 LR**	**L-LDA**	**Human**	**L1 LR**	**L-LDA**	**Human**	**L1 LR**
anger	197	0.57	0.58	0.67	0.56	0.72	0.59	0.44	0.68	0.41
anxiety	200	0.22	0.33	0.26	0.23	0.43	0.19	0.31	0.46	0.28
depression	198	0.10	0.39	0.26	0.23	0.53	0.24	0.31	0.56	0.30
low self-esteem	200	0.08	0.36	0.14	0.11	0.44	0.14	0.25	0.45	0.25
suicidal behavior	198	0.42	0.78	0.12	0.34	0.64	0.13	0.38	0.60	0.178

These scores give us an upper bound on performance against which to compare our model (assuming that human raters are performing optimally). To calculate this reliability score, we compared each individual rater against each of the other raters by computing pairwise AUC scores. We express human-reliability as AUC scores so that L-LDA performance and human reliability are expressed in the same units and fair comparisons can be made. For an individual rater, we calculate AUCs using the ratings of the individual rater (analogous to the model scores) to predict the binarized ratings (at 5%,10%, and 20% cutoffs) of each of the other raters. We compute human reliability as the average of all AUCs calculated from each pair of raters and report the computed scores in Table 1.3. To compute human reliability in terms of R-precision, we perform an analogous computation using R-precision instead of AUC.

The table shows that both L-LDA and LLR perform well at identifying representative talk-turns relative to human reliability. On average, L-LDA AUC scores are between 10-18% lower than average inter-rater AUC scores. The L-LDA model performs distinctly better

at identifying talk-turns representative of anger than talk-turns representative of the other tested symptoms. The unique lexicon of words used to express anger may influence the model's performance. In addition, the other 4 symptoms may be expressed in a broader language that is more difficult to capture through uni-, bi-, and trigrams. In addition to variation in performance by symptom, the model performs better when identifying the top 5% of representative talk-turns as compared to the top 10%. Therefore, the model is able to identify the most relevant talk-turns in a session with reasonable precision. The comparison between L-LDA and the baseline model shows that the LLR model performs about the same or marginally better than the L-LDA model on each of the three cutoffs (p=0.28, p=0.11, p=.78, respectively in pairwise t-tests).

1.7 Discussion and Conclusion

In this article, we have presented the Labeled Latent Dirichlet Allocation Model as a method for the semi-automatic code annotation of psychotherapy sessions. L-LDA outperforms standard discriminative methods at identification of session-level codes, replicating results from prior psychotherapy process research and general applications in multi-document classification. In addition to session-level coding, machine-learning methods show promise for annotation of psychotherapy transcripts at fine-grained levels of detail, such as for talk-turn annotation. L-LDA and LLR can identify talk-turns representative of session-level codes with accuracy close to that of trained human coders.

Machine learning methods for document classification often focus either on topic-based classification involving large documents and many topics, or sentiment classification involving a small set of sentiment labels and often shorter documents [103]. Our work involves both topic-based classification (for session level prediction) and analysis more similar to sentiment

classification (talk-turn prediction for a small set of class labels). The generative nature of L-LDA provides a natural bridge between these two types of document classification problems by inferring labels for talk-turns based on session-level metadata. Topic-based classification is performed by integrating topic information over constituent parts of a document (in our case talk-turns), and sentiment classification is performed using a mapping between topic-based class labels to sentiment labels. In this way, L-LDA provides richer information than many sentiment classification methods and more flexibility than some topic-based classification models. Examining the relationships between the mapping from topic-based classes to sentiment classes is an interest for future work and we suspect that incorporating this information will lead to improved predictive performance.

Promising results in annotation of psychotherapy transcripts suggest potential for application to clinical settings in addition to reducing labor costs and improving the scalability of observational coding. For example, in the process of training junior therapists, supervising therapists review records of the junior therapist's sessions. Supervising therapists are often in charge of many junior therapists and are in need of tools that make the review process more efficient. One method for making this process more efficient would be to use text-based models that predict important topics discussed in the sessions (such as depression, suicide, etc.). The supervisor can get a quick summary of session content and can locate specific passages in the session by content labels. Additionally, the supervisor can provide feedback to the model on which passages were relevant to that topic and thus improve future code annotation.

L-LDA is a model for the semantics of language that, like all models, provides an approximation to the true underlying process of generating speech to convey meaning. L-LDA makes several simplifying assumptions about the process of text generation that could provide starting points for further model development. The "bag-of-words" assumption disregards

information about temporal characteristics of language and their relation to semantics. L-LDA also ignores syntactic dependencies. An important direction for semantic analysis of psychotherapy sessions would be to incorporate sequential information and context into our analysis. This would involve significant feature engineering, but could benefit from already existing text processing techniques such as word and sentence embedding.

The work presented above analyzes the relationship between semantic information contained in spoken language and subjects and symptoms that encompass not just semantics, but emotion, and behavior. To gain a deeper understanding of psychotherapy, semantic language models need to be extended to encompass behavior. Considerable information is contained in behavioral cues such as tone, laughter, or body language that encompass the semantic meaning of a statement. While these behavioral cues are most likely correlated with language, we think that jointly analyzing behavior and language will lead to deeper understanding of the psychotherapy process and its effect on patient outcome.

In conclusion, we used data from the patient provider interactions in psychotherapy to illustrate the potential of machine learning methods to automate coding of key aspects of clinical conversation and to understand the linguistic processes behind psychotherapy. L-LDA is a robust automated coding method that outperforms a baseline logistic regression discriminative method at predicting codes at the session level and that can be used to localize information using only session-level metadata.

Chapter 2

Improving Government Response to Citizen Requests Online

2.1 Introduction

In 1917, the Mexican government ratified Article 8 of the Constitution [91], which gives citizens the right to written request. Article 8 states:

> *Public officials and employees shall respect the exercise of the right of request, provided it is made in writing and in a peaceful and respectful manner; but this right may only be exercised in political matters by citizens of the Republic.*
>
> *Every request shall be replied to in writing by the official to whom it is addressed, and said official is bound to inform the requester of the decision taken within a brief period.*

The request system provides the citizens of Mexico with a crucial conduit for interfacing with their government. The majority of requests come from poor states (at least 46% of population below poverty line). Requests range from questions on how to enroll in social services, assistance in accessing a pension fund, and requests for medical coverage. In some requests, it is obvious that citizens turned to this system when they didn't know where else to go. One extreme request was a plea for help; the citizen had been assaulted but the charges were dropped because the lawyer and witnesses had been physically intimidated. Given the gravity of some of these requests, it is important that there exist a system to process and respond to as many requests as quickly as possible. To satisfy this need, the government of Mexico created the *Sistema Atención Ciudadana* (SAC), whose sole responsibility is to receive requests and direct them to the federal agencies that ultimately provide responses.

Traditionally, citizens submitted handwritten requests to a brick-and-mortar location. However, in 2015 the Mexican government launched an online request submission portal to respond more efficiently to citizens. The submission process consists of the following steps (see Figure 2.1):

1. **Step 1**: A citizen visits the online portal and lodges his or her request (www.gob.mx/atencion/). They receive a confirmation email and must verify their email address. Once the citizen has verified their email address, the request enters the system. Unverified requests are automatically rejected.

2. **Step 2**: After email verification, a human receptionist reads the request and decides whether to reject it, or accept it for further processing. Requests can be rejected for having inappropriate or disrespectful language or for being unintelligible.

3. **Step 3**: The receptionist decides to send the request to the SAC office or to a help desk. The help desk provides technical support for the website, and can help citizens find

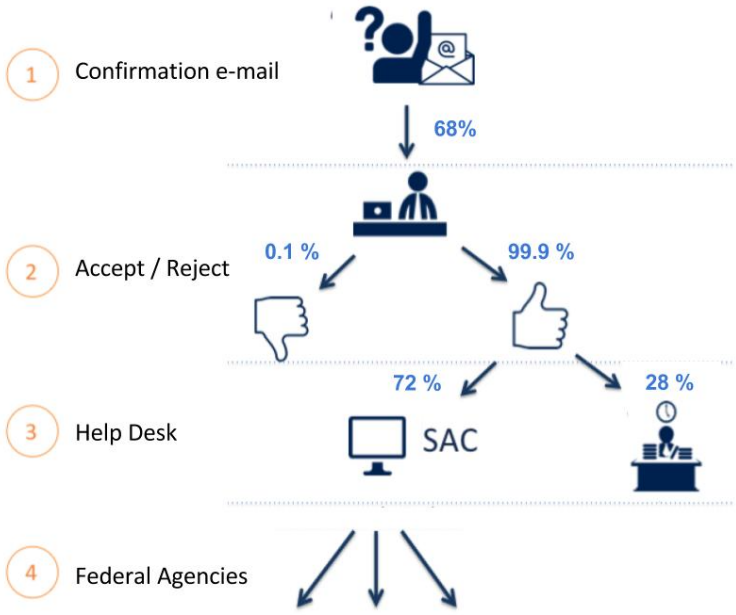

Figure 2.1: A step by step diagram of online request submission.

information or forms. All remaining requests are sent to the SAC office.

4. **Step 4**: SAC analysts send requests to the appropriate federal agency via the agency's contact person. Requests are routed to the different federal agencies according to the subject and the knowledge the analyst has of the federal agencies. If the agency can handle the request, they write a response. Analysts in the SAC coordinate with the federal agency to ensure that the request is resolved. If the federal agency cannot handle the request, they send it back to a SAC analyst to send to a different federal agency. Once the request is approved by the analysts, an answer is sent to the citizen.

A striking problem with the current system is that the supply of labor for processing requests is struggling to accommodate an increasing number of requests submitted. The number of online requests submitted daily doubled from ≈100 requests in 2015 to ≈ 200 requests in 2016, and the government anticipates further increases in request submissions as the website is publicized. The SAC processes requests manually, and cannot afford to hire additional

staff. To keep up with the increasing volume of requests, the SAC needs a scalable approach to processing requests.

The goal of this project is to improve the lives of the Mexican population by increasing the number of requests that the request system can efficiently process. We focus on 1) increasing the number of, and speed at which, citizen requests are addressed and 2) improving the scalability of the online request response system by developing automated methods for request processing. This automation will enable some percentage of requests to be processed and routed automatically while the rest will be handled manually (based on the system's confidence). Each step in this process has specific bottlenecks that we highlight in the following sections.

2.2 Step 1: Email Confirmation

2.2.1 Problem

During initial data exploration, we discovered that many requesters fail to verify their contact via the confirmation email: 32% of all incoming requests remain unconfirmed. The purpose of this step is to filter spam, but we found that many unconfirmed requests contain legitimate requests from citizens. Neither the email nor the web portal clearly expressed to the user that he or she needed to take action on the confirmation email in order to have his or her request read (See Figures 2.2 and 2.3).

The email contained a text hyperlink that was easily overlooked, an email header that translates to *'Your request is being Processed'* (when in reality the request will only be processed after email confirmation), an email subject that suggests that the request has been

submitted (rather than requesting confirmation from the citizen), and an informal sender address (*sac@presidencia.gob.mx*) that does not convey the official nature of the email.

Figure 2.2: On the left is the old confirmation email. Highlighted are misleading email titles and headers, informal sender address, and the use of a text hyperlink rather than a button. On the right the new confirmation email. We changed misleading email titles, sender address, and used a button rather than text hyperlink to link to confirmation page.

The website also failed to communicate that a citizen would be receiving an email that must be responded to. The site has a misleading title (*'Your request is Being Processed'*), and only at the bottom of the page informs the user that a confirmation email was sent and contains a link that needs to be clicked on. Furthermore, the hyperlink text is not highlighted or made visually distinct from other text on the page.

2.2.2 Solution

We provided a new version of the confirmation email that clearly expressed to the user that he or she needed to take action (see figures 2.3 and 2.2). We changed the body of the email to explain that email will be the primary form of contact between requester and respondent, and thus must be confirmed in order to proceed. We used strategies from behavioral science research and widely used in digital marketing to encourage citizens to confirm their email, including providing a large red button linking to the confirmation webpage (rather than a

hypertext link), changing the sender email account to reflect a formal sender, and changing the email subject to indicate that action is necessary.

In the new version of the website, we changed the title to indicate that the user has not yet finished submitting a request (*'You Have One More Step'*), updated content to inform the user that a confirmation email was sent, and added step by step instructions for verifying the user's confirmation email.

Figure 2.3: On the left the old website with Original Spanish version and English translation. On the right is the new website with Spanish version and translated version.

2.2.3 Evaluation

We used A/B testing to evaluate whether the new email and web portal would lead to an increase in confirmation rate. For a two-week period (July 27th to August 8th, 2016), we collected new requests (≈2500). For each new request, we randomly and uniformly gave the requester either A) the new versions of the website and email or B) the old versions of the website and email. We performed a two-way t-test to test the difference in confirmation rates between the two versions.

2.2.4 Results

We found that the new website and confirmation email resulted in a 39% ($p < 1 \times 10^{-33}$) increase in the rate of confirmed requests, compared to the old website and confirmation email. We believe that the large emphasis placed on advising the citizen that they had one more step to complete is the driving force behind this increase.

It's important to note here that simple behavioral science techniques and A/B testing can have a big impact on response rates in these types of problems.

Our partners at the Office of National Digital Strategy in México [92] have implemented the changes to the email and website changes, and the rate of unconfirmed requests has decreased from 32% to 19%. We also provided recommendations for further improvements, and advised them to continue running A/B tests to test the effects of minor changes to citizen communications.

2.3 Steps 2-4: request Automation

After a requester confirms his or her email, the SAC reads the corresponding request. Independent subdivisions of the SAC (2) accept or reject a request, (3) send a request to a technical help desk or to another analyst for agency routing, and (4) route requests to the appropriate federal agency for response. We implement an automatic routing system to reduce man hours spent reading requests and increase the scalability of the entire request system.

Automatic routing of requests can be viewed as a multi-label document classification problem [45, 123, 85] where each request can have up to two classes (i.e., accepted/rejected, sent to

helpdesk or sent to a given federal agency). Generally, there are three ways to handle a multi-label classification: convert into a multi-class classification problem using power-sets, converted into multiple binary problems (i.e., the Binary Relevance (BR) method), or adapt the learning algorithm, [45]. Since each stage of our problem is performed by independent agents, we use the BR method to mirror the independent nature of our problem domain and allow careful control of performance at each stage. Additionally, the BR method allows for problem scalability, reduces overfitting of infrequent labels, and allows for flexible inclusion and exclusion of labels in a potentially changing government environment [114].

Another important modeling choice in text classification is whether to use count or vector based embeddings [26, 93, 84]. We chose to use a count embedding approach rather than vector space encodings to easily interpret the relationship between features and model parameters and to reduce computational demand.

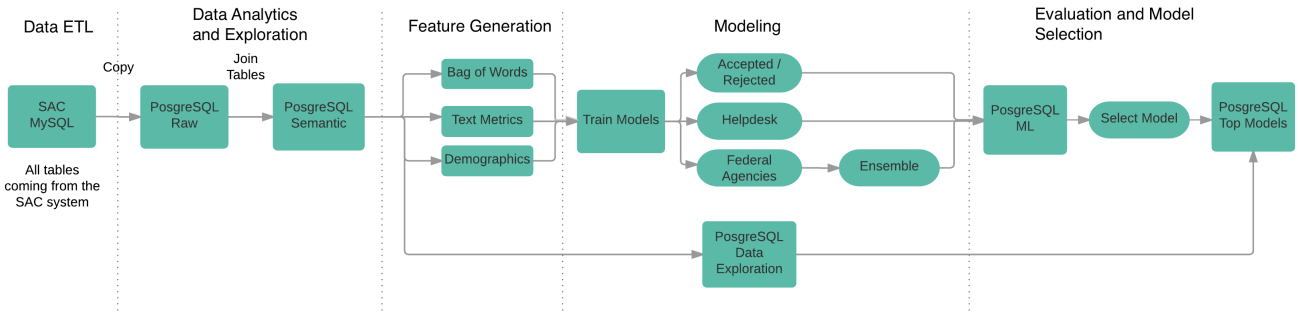

Figure 2.4: Computational pipeline.

We implemented machine learning models for each step in the request process (2-4) (see figure 2.1). We built a machine learning pipeline to run all sequential tasks: 1) Data ETL, 2) Data Analytics and Exploration, 3) Feature Generation, 4) Modeling, and 5) Evaluation and model selection (see figure 2.4).

2.3.1 Data

Our data comes from the SAC and contains information about each request, the citizen who submitted the request, and how the request was routed.

The data set consists of 69,402 online submissions from October 2014 to May 2016. Each online submission contains the text of the request, demographic information about the requester, and details from each processing step performed within the SAC, including who read the request, where the request was sent, and the final government agency that responded to the request.

The age of requesters ranges from 17 to 67 years, but half of all requesters are under 37 years old. The skew toward lower ages is most likely due to increased facility with computers among younger populations, increasing their ability to submit online requests. Men submitted more requests than women (56%, and 44% of all requests, respectively). Almost all requests (98%) were submitted in Mexico. The states that submitted the largest percent of all requests are also the states with the largest populations: Mexico City (19%, 8 million people), the State of Mexico (16%, 16mil), Jalisco (7%, 7mil), Veracruz (7%, 8mil) and Guanajuato (4%, 5mil). Out of the 47% of respondents that provided occupational status 28% were employed, 9% unemployed, 5% students, and 5% housewives.

We also computed descriptive statistics about the text of the requests to provide insight into feature generation for the machine learning models. Rejected requests and requests sent to Helpdesk are shorter than those sent to SAC or federal agencies (see Figure 2.5). Since, requests may be rejected if they contain curse words, we computed how many curse words each request contained. Less than 1% of requests contained curse words those that did contain an average of 1.47 curse words. A single request might mention one or more agencies, so we computed how often each federal agency was mentioned (either its full name

or abbreviation). From the entire corpus of requests, 44 agencies were mentioned at least once. The top mentioned agencies were IMSS (1724), ISSSTE (1157), INFONAVIT (795), SEP (703), and SAGARPA (380).

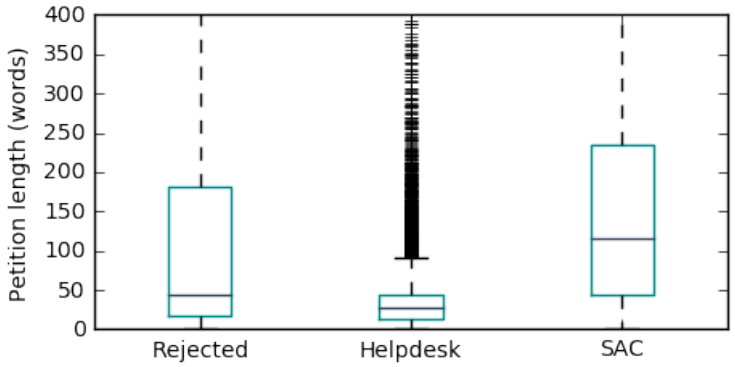

Figure 2.5: The length of requests for each step of the request process. Requests that were rejected or sent to the help desk are shorter than accepted requests.

2.3.2 Feature Generation

Table 2.1: Set of all features used in all models.

Set name	**Features**
BOW	8 BOWs
Document Statistics	request length (words/characters), request length after processing, number of words removed during preprocessing, number of curse words used, whether each agency was mentioned (one feature per agency), number of times each agency was mentioned (one feature per agency)
Demographics	gender, occupation, nationality, whether requesteris Mexican, time of day submitted

Our set of generated features consists of text statistics, document attributes, and demographics (see Table 2.1).

We created multiple Bags-of-words (BOWs) by varying 1) whether to apply a term frequency-inverse document frequency (TFIDF) transformation, 2) whether to tokenize documents into unigrams or unigrams+bigrams, and 3) whether to stem tokens. Request pre-processing resulted in 8 different BOWs: one for each combination of parameter options. We removed

a custom set of Spanish stopwords manually selected to have little semantic value. When fitting our models, we consider each BOW a separate corpus representation and test which representation leads to better performance.

We generated document attribute features by computing document level statistics such as the number of times curse words were used in a request, the length of the request, the number of words removed during pre-processing (as a marker of how different requesters expressed themselves), and the number of times each federal agency was mentioned in request text.

We used demographic information to generate categorical features such age, sex, location, and occupation. We also included the time of day when the request was sent. Since most of the demographic fields were not mandatory, we added additional binary features to flag whether the requester provided each demographic variable.

2.3.3 Modeling

We built machine learning models for automating steps 2-4 of the request submission process (accept/reject, help desk/SAC, routing to federal agencies). Each model takes features computed from one submission (request and requester) as input and returns a class label as output.

We fit our models on different subsets of the dataset. The accept/reject model was fit on all requests in the dataset. The help desk model was fit on all the accepted requests since its creation in January 2016. The federal agency model was fit on all requests sent to the SAC. By using all applicable data to train each model, we treat each step as independent. This gives the best possible estimates of how much time we could save at each step, but may not accurately reflect system performance across steps.

Table 2.2: Models and parameter values iterated over in the pipeline.

Model	**Parameters**	**Range**
Random Forest	estimators, max depth, min sample split penalty	[100-3000], [10-500], [5,10] [0.001,0.1,1,10]
Logistic Regression	penalty	[0.00001,0.001,0.1,10]
Naive Bayes	-	-
Linear Support Vector	penalty	[0.001,0.1,10]

For all steps, we performed a grid search over model types and parameters to optimize performance (See table 2.2 for a list of all model and parameter combinations). For step 2, we fit binary classifiers to predict whether a request would be accepted or rejected. For step 3, we fit binary classifiers to predict whether a request would be sent to the help desk.

For step 4, we fit independent binary classifiers for sending a request to each of the top five most solicited agencies (see Table 2.3). We focus on the top five most solicited agencies because they make up a plurality of request submissions. Other agencies had too few submissions to reliably classify. Focusing on the top five allowed us to create a classifier that could reduce the load on the SAC by handling the more common requests while minimizing error.

We compare two types of decision frameworks: one in which a request can be sent to one and only one agency (multi-class classification), and another in which a request may be sent to multiple agencies (multi-label classification). While the former approach fits more closely with current SAC practices and is therefore easier to implement, we find significant performance improvements in the latter, potentially allowing the SAC to process more petitions faster.

For the single-label case (a request can be sent to only one agency), we compare our threshold optimization routine and decision rule to a decision tree, and find that the threshold optimization routine performs better.

Table 2.3: The top five most solicited federal agencies and number of requests sent to each agency between October 2014 and May 2016.

Agency	**Abbr.**	**#Pet**
Secretaría de Educación Pública	SEP	4,112
Dirección General de Atención Ciudadana	DGAC	3,980
Secretaría de la Función Pública	SFP	2,197
Secretaría de Salud	SS	1,492
Secretaría de Economía	SE	1,315

Threshold Optimization Binary models traditionally produce classifications by defining a score threshold; examples with scores above the threshold are classified as 1 and examples with scores below the threshold are classified as 0. To produce a classification from a set of scores, we must find a set of thresholds (one per agency) and a decision rule (i.e., how to classify a request with multiple scores above the respective thresholds) that produce optimal classifications. Our threshold optimization routine searches over all possible combinations of score thresholds and selects the set of thresholds that gives the highest precision given a minimum recall of 0.15 (see Algorithm 1). In order to reduce computational demands, we run two iterations of the optimization procedure. On the first, we use larger-spaced thresholds. In the second, we manually select high-performing threshold ranges in consultation with the office on acceptable failure rates given the error rate of human classification.

Decision Tree We compared our threshold optimization routine to a decision tree that selects an agency from the binary classification scores. For a single request, the tree takes scores from the top models for each agency as input and produces a categorical label. The label indicates which agency to send the request to or, alternatively, if the request should be sent to another federal agency.

input : requests (P), class labels (C), set of classification models M, $n = |M|$, integer k
output: best threshold set $t_{\max}$

for *iteration = 1:2* **do**
  if *iteration == 1* **then**
    `// evenly-spaced threshold generating set;`
    $h_m = [0, \frac{1}{k}, \frac{2}{k}, ..., \frac{k-1}{k}] \ \forall m \in \{1, 2, \ldots, n\}$
  end
  `// generate threshold superset, all combinations of thresholds for each model;`
  T $= \{[h_1(j_2), h_2(j_2), \ldots, h_n(j_n)] \mid j_i \in \{1, \ldots, k\}$ and $\forall i = 1, \ldots, n \}$;
  for *threshold set* $t \in T$ **do**
    for *request* $p \in P$ **do**
      $s = [M_1(p), M_2(p), ..., M_n(p)]$ `// model scores;`
      if $sum(s > t) == 1$ **then**
        $C_{\text{out}}(p) = c : s_c(p) > t_c$ `// predict class`
      else
        $C_{\text{out}}(p) = -1$ `// send to SAC`
      end
    end
    $PREC(t) = \text{precision}(C_{\text{out}}, C)$;
    $REC(t) = \text{recall}(C_{\text{out}}, C)$;
    `// correct prediction if the correct class or we send to SAC and correct class is not among candidate models`
  end
  $t_{\max} = \text{argmax}_{t \in T: REC(t) > 0.15} PREC(t)$;
  `// set threshold generating sets for next iteration;`
  for $m \in \{1, ..., n\}$ **do**
    $h_m = [a_m : \frac{bm - a_m}{k} : b_m]$ `//` a_m, b_m `chosen by visually inspecting range of thresholds that gave highest precision for each model`
  end
end

Algorithm 1: Threshold Optimization Routine

Multi-label Agency Classifications We also explore a decision framework that allows a request to be sent to multiple agencies. The intuition behind this approach has two parts. First, if a request is sent to more than one agency at the same time, this in effect parallelizes the current process, allowing for faster processing time overall. Second, we could provide a

list of possible classifications to analysts to reduce classification errors, and speed up the time it takes to manually classify a request. We expect automation to allow the SAC to handle more requests. However, if too many redundant requests are sent to federal agencies, then any time-saving benefit will decrease as they divert energy toward handling misclassified or duplicate requests. Whether this approach is effective will depend on emerging bottlenecks as the system is deployed, and optimizing between the two approaches will require continued testing of the deployed system.

As in the first decision framework, we use a score threshold for classification; however, here we use a cost matrix to optimize the set of recommended classifications. The cost matrix allows us to weigh the consequences of false positives differently than false negatives. A false negative is very costly because eventually the request will have to be sent through the manual system. A false positive is less costly. As long as the correct federal agency is among the set of all agencies a request is sent to, that request will not have to re-enter the manual system. The set of possible classifications and corresponding costs for a request are as follows:

Cost = 1: A request is classified as not belonging to any of the top five agencies, and we send it through the manual system. This case does not require any additional work over the current manual system, so we assign it a low cost.

Cost = 0.5 $\times(\mathbf{N-1})$: N classifications to agencies are produced, of which one classification is correct. Since we sent the requests to $N-1$ incorrect agencies, we penalize by the number of incorrect agencies we send to. If we set the cost for sending a request to the incorrect agency as 1 or higher, then this analysis is unnecessary, since it would be less costly to send any request with more than a single classification through the manual system. This simplifies to the case in the single classification framework that we described above.

Cost = 2×N: N classifications to agencies are produced, and they are all incorrect. We penalize this case heavily because it wastes time, and we will have to send the request through the manual system after agencies deny the request.

We then use a threshold optimization routine for classification. Given a set of score thresholds and model scores for a request, the threshold routine classifies a request as follows:

1. If zero model scores are above their respective score thresholds, we classify the request as belonging to an agency outside the set of agencies we are considering.
2. If two or more agency scores are above their respective score thresholds, we send the request to all agencies with scores above threshold.

Similar to the threshold optimization routine for the single-classification framework, we iterate over all possible threshold sets. We compute a total cost for each threshold, and find the optimal threshold by choosing the threshold set with minimum cost that sends at least 15% of all requests to one or more federal agencies.

2.3.4 Evaluation and Model Selection

The general objective of our models (accept/reject, helpdesk, federal agency) is to automate as many requests as possible while maintaining a low error rate, where error rate is defined separately for each step in the request process. A key aspect of our problem scope is that our models do not have to automate *all* requests, since any requests we are not confident about can be sent through the existing manual system. Therefore, we only automate the requests that we are most confident about, and optimize the number of requests processed automatically at a given error rate. We worked with the National Digital Strategy office to

decide acceptable error rates and found that the maximum error rate allowed at any step is 6%. Thus in general, the metric we will optimize is the maximum number of requests we can classify at an error rate of 6%. We also report the number of additional man hours our models would save.

All model evaluation for single binary models is performed in a 10-fold cross validation routine. We use a stratified (by federal agency) 5-fold cross validation routine for training and testing multi-class federal agency classification.

Step 2 Accept/Reject

We evaluated our accept/reject model according to the maximum number of requests we can accept while guaranteeing that 94% of accepted requests are true positives. Specifically, we maximize the percent of requests that we can classify with a precision of 94%. Our modeling pipeline automates this process, and allows the SAC to choose any precision threshold. Importantly, we only automate requests that our model is highly confident will be accepted. This avoids the ethical problem of automatically rejecting requests, since rejecting a request that should have been accepted denies a citizen their constitutional right to request.

Step 3 Helpdesk/SAC

We evaluated our help desk model according to the maximum number of requests we can automate by funneling accepted requests to either the SAC or the help desk. For this model, the cost of misclassification is not as high, since misclassified requests remain inside the request system and are manually re-routed to the correct location. Specifically, we take the

model that maximizes

$$\max_m p_m^{0.94} + 1 - r_m^{0.975},$$

where $p_m^{0.94}$ is the percentage of requests classified when help desk precision for model m is 0.94 and $r_m^{0.975}$ is the percentage of requests classified as help desk when help desk recall is 0.975. Intuitively, maximizing $p_m^{0.94}$ gives the most requests we can send to the help desk with 6% error. Maximizing $1 - r_m^{0.975}$ gives the minimum percentile at which we attain a recall of 0.975, representing the a high separability between help desk and SAC scores.

Step 4 Federal Agency

For each federal agency, we searched for the binary classifier that allowed us to route the most requests with the minimum error rate. For each agency, we selected the model that optimizes

$$\max_m p_m^{0.94}$$

where $p_m^{0.94}$ is the percentage of requests classified as being sent to an agency when precision for model m is 0.94.

Using the best models for each federal agency, we compute a decision rule, either by decision tree or threshold optimization. For the framework where a request can be sent to just one agency, we evaluate models (decision tree vs. threshold optimization) by searching for the model with highest overall precision. For the best model, we report the precision, recall, and the number of requests we could automate at that precision. We evaluate the multi-label classification decision framework by comparing the total cost associated with our model to the total cost of the current manual system.

2.3.5 Results

The best accept/reject model is a Random Forest with 1000 trees, 500 maximum depth, 5 minimum sample split and log_2 maximum features using document statistics and bags of words as features (See figure 2.6 for a histogram of scores and true classes). The model can classify accepted requests with an error rate of 8.8%, but this is a small improvement upon the baseline error rate of 9.2% obtained by classifying all requests as accepted.

If we take only the requests that we are most confident in accepting, we can automate 85% of all requests with a 6% error rate, which would automate 170 requests and save 3.8 man hours per day (at a rate of 200 requests in an 8 hour work day).

We looked into the data for reasons for this low performance, and found that content was not greatly discriminative for accepted and rejected requests. We believe that incorporation of new features and manual pruning of the dataset may lead to improved performance, but we leave this to future work.

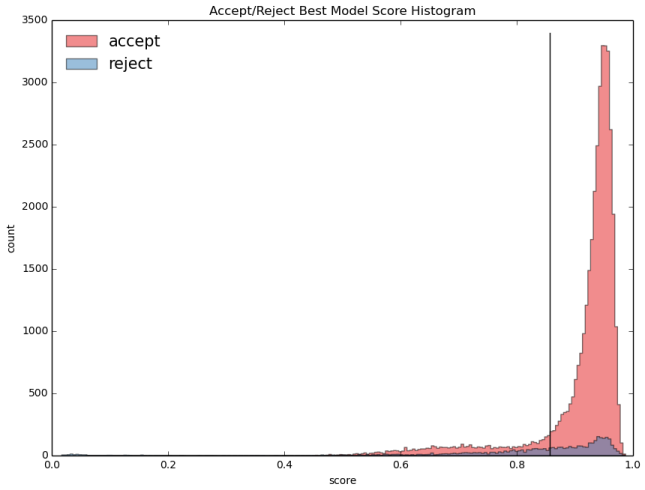

Figure 2.6: Density of random forest scores for accepting and rejecting requests. The threshold is drawn at a 6% error rate, which allows us to automate 85% of requests.

The best performing help desk/SAC model is a Random Forest with 1000 trees, 100 maximum depth, 5 minimum sample split and log2 maximum features using the demographic, bag of words, and text metric features (See figure 2.7 for a histogram of scores and true class). The vertical lines on the figure show score cutoffs that define score ranges that correspond to where to send requests: scores above the upper threshold are sent directly to the SAC, scores below the lower threshold are sent directly to the help desk, and scores in between are sent to a receptionist to process. The thresholds were chosen so that of the automated requests, there is only a 6% error rate. We found that using these thresholds we can automate 39% of requests at this step, saving approximately 3.12 man hours per day (at a processing rate of 200 requests in an 8 hour workday).

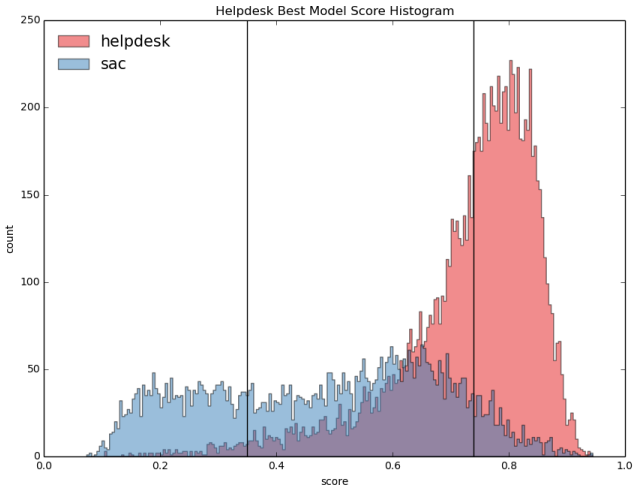

Figure 2.7: Density of random forest scores for sending requests to the help desk or the SAC. The thresholds are drawn at the positions where we can automate the most requests with an 6% error rate, allowing 39% of all requests to be automated.

For federal agency routing, we took the models that best automate requests by looking at the percent we could automate at an error rate of 6%, i.e., the percent of requests we could automate with precision of 0.94 (See Table 2.4). The model for routing requests to Secretaría de Educación Pública (SEP) provided the best individual level of automation and was able to

Table 2.4: Performance, parameters, and features used in our best performing models. LR=Logistic Regression, RF = Random Forest. Perc. at 0.94 is the percent of all requests we can classify with a precision of 0.94. This is the metric we maximize to choose the best model.

Agency	**Mod.**	**Parameters**	**Corpus**	**Perc. @ .94**
SEP	RF	max_depth=100 max_feat=log2 n_estim=3000 min_samp_split=10	tfidf=0 1-2gram stem=0	3.6
SS	RF	max_depth=500 max_feat=log2 n_estim=1000 min_samp_split=10	tfidf=0 1-2gram stem = 0	0.09
DGAC	LR	C=1 norm=L1	tfidf=0 1gram stem = 1	0.08
SFP	LR	C=10 norm=L1	tfidf=1 1-2gram stem=0	0.02
SE	LR	C=1 norm=L1	tfidf=0 1gram stem=0	0.01

automate 3.6% of requests at a 6% error rate. The next best models were for automating to Secretaría de Salud (SS,.09%), the Dirección General de Atención Ciudadana (DGAC,.06%), Secretaría de la Función Pública (SFP,.02%), and Secretaría de la Economía (SEP, .01%). Were we to incorporate a single model for agency routing, in this case the model for routing to SEP, we would be able to automatically route approximately 2-3 requests per day and save the SAC 1.14 man hours per day (3 analysts and one supervisor processing 200 total requests per 3.53 hours, given time spent on handwritten requests).

We evaluated our threshold optimization routine for producing single classifications from all independent models, and compared it to a decision tree in order to provide an option that could fit into the current SAC process. We also evaluated the performance of the threshold optimization routine in a multi-label agency classifications setting.

Threshold Optimization The threshold optimization routine performed better than the decision tree, resulting in a precision of 0.84, recall of 0.25. The threshold optimization routine would send 14% of all requests to one of the top five dependencies, which would save 2.03 man hours per day. However, the precision for the threshold optimization routine is still too low to satisfy our partners requirements. We believe that this poor performance is in part because in the available data there are relatively few requests sent to each agency. As the online request system matures and more requests are submitted to the top agencies, we hope that precision will increase enough for the system to be implemented. Because of this poor performance, we also presented the multi-label classification framework.

Decision Tree For the single classification federal agency decision framework, the decision tree resulted in a precision of 0.203, recall of 0.438, and accuracy of 0.55. The model had trouble distinguishing requests that should be sent to DGAC from requests that should be

sent through the manual system; 47% of requests sent to the DGAC were classified by the model as manual system, and 14% of requests sent through the manual system were classified by the model as DGAC. This result is not surprising since DGAC, the office that processes requests, is the final agency to send out a response, and most likely receives many requests inquiring about responses to other requests. The threshold optimization routine did a better job of distinguishing between agencies with the training data available so far.

Multi-label Classification The threshold optimization routine for multi-label classification performed better than an all-manual baseline, on average resulting in $\approx 66\%$ of the cost of running the manual system (18470.5 versus 27546). A cost of 1 corresponds to sending the request through the manual system, which is equivalent to 0.2748 man hours (three analysts and one supervisor processed on average 51.36 online requests in 3.528 hours, given handwritten request processing time). Thus, the multi-label classification system would have saved the SAC 4.7987 man hours a day over the 18 month period over which the data was collected, and to process an additional 17.46 petitions per day.

2.3.6 Technical Implementation

The pipeline is implemented using the pipeline manager tool Luigi [7] that allows the system to be deployed at scale. Luigi helps build complex pipelines of batch jobs by automatically handling dependency resolution. For example, for each task in our pipeline, we specify three functions:

1. **requires**: returns the output that must exist for the task to run
2. **run**: runs the task

3. **output**: returns the output of the task.

Luigi automatically checks whether the required input exists for each task, and only runs a task if its output does not exist or if its input has changed. We avoid redundant calculations by separating each calculation into modular tasks. This enables the SAC office to both automatically retrain the models as new data becomes available, and avoid time-intensive recalculations when adjusting score thresholds and other model specifications.

All computation was run using Amazon Cloud Computing, databases were hosted on Amazon Cloud Computing servers, and output was stored on Amazon S3. All code is available on github [99].

To have access to a larger suite of analytical tools and database resources, we migrated the MySQL database into a PostgreSQL database. We then created a non-normalized schema for feature generation and model building. Feature generation was done using PostgreSQL, sci-kit learn, and gensim [115]. Modeling was implemented in sci-kit learn [104] and we used luigi to perform a grid search over all models and parameters.

To easily deploy our system on the SAC's servers, we wrote Docker containers to initialize and run the pipeline. Docker containers are virtual environments that wrap software in a complete file system that contains everything needed to run: code, runtime, system tools, system libraries. Docker guarantees that software will run consistently, regardless of its environment. Our Docker files contained all the necessary code for setting up a Luigi server that runs all pipeline tasks.

2.3.7 System Integration

To seamlessly integrate our automated system, models will be initially tested side-by-side with the receptionist and analysts. For each model that accurately automates a step of the process, that model may be deployed. This testing period will build trust in the system as users (SAC employees) see how it works. It will also be an opportunity to get user feedback and determine acceptable error rates.

To ensure that our models remain robust to changing request topics, we recommend continually generating unbiased labeled data; a percentage of requests that were confidently classified should still be sent to the receptionist or the SAC. These labels should be used to re-train, re-test, and re-select top models at regular time intervals to ensure that models change over time with the data and continue to automate the maximum number of requests possible at acceptable error rates.

It is extremely important that the accept/reject model only automates high-confidence accepted requests. If a request is rejected that should not be rejected, we are denying a citizen access to his/her constitutional right and undermining the goals of the request system.

2.4 Future Work

A limitation of our work is that our models at each step were fit independently on perfect data from the proceeding step. For example, we did not test the case where our accept/reject model erroneously accepts a request and then sends that request to our help desk model. Since the accept/reject model automates the first step in the processing system, our results are accurate for this step and the model will still save 3.8 hours per day of manual processing

time. However, we expect the help/desk and federal agency routing error to be slightly larger in practice. Future work will involve integrating models into a single system, optimizing for system-wide performance, and analyzing system-wide error propagation.

Our evaluation framework can be improved to account for changing topics in the submitted requests. Since the data came from a 1.5 year time period, we don't think that this greatly impacted results. However, as the system matures, incorporating temporal changes in the data will become increasingly important. We plan to do this via temporal cross validation and periodically refitting our classifiers as additional data are generated.

Another method by which the evaluation framework should be improved is to further investigate the biases in our dataset. During data exploration, we looked at the demographics most likely to submit a request, but should expand these analyses to whether certain demographics are associated with accepting/rejecting requests or to which agencies requests are sent. In the case that systematic biases exist, further analysis should be conducted to determine whether biases are a result of request content or the decision making process. Together with the SAC, any decision-making biases should be addressed in incoming data streams. Importantly, the causal mechanisms of these biases need to be studied in order to understand how to correct them. We cannot correct artificial intelligence biases without correcting our own biases.

2.5 Summary

The presented work improves the request processing system of the government of Mexico by increasing the number of requests read by the government and developing automated methods for request processing. Using our work, the SAC will be able to process and respond

to more requests in a timely manner. The government of Mexico will be able to fulfill the guaranteed right to request and the lives of Mexican citizens will improve.

2.6 Acknowledgments

We thank the Eric & Wendy Schmidt Data Science for Social Good Fellowship for generously supporting this work, our partners at the Office of National Digital Strategy for sharing data, expertise, and feedback for this project, and all staff at Sistema Atención Ciudadana for their help and support.

Chapter 3

Predicting Task and Subject Differences with Functional Connectivity and BOLD Variability

3.1 Introduction

Functional connectivity (FC) and BOLD variability (BV) are two metrics that focus on changes of the BOLD signal around the mean. FC is the correlation or covariance in BOLD activation across regions [38, 10, 40, 136] and BV is the region-specific variance in BOLD activation [46, 47, 48]. Figure 3.1 illustrates the connections between FC and BV. FC (traditionally computed as the Pearson correlation) is based on a combination of the off-diagonal and diagonal entries of the covariance matrix, whereas BV is based on the diagonal entries of the covariance matrix. Unlike FC, BV can be computed independently for each region and does not require computationally expensive pairwise comparisons or the conceptualization

of an underlying network. Across different studies, FC and BV have both been linked to task and individual differences but they have not been compared directly in a single study. This paper presents an assessment of the relative diagnosticity of FC and BC in predicting task being performed and subject identity.

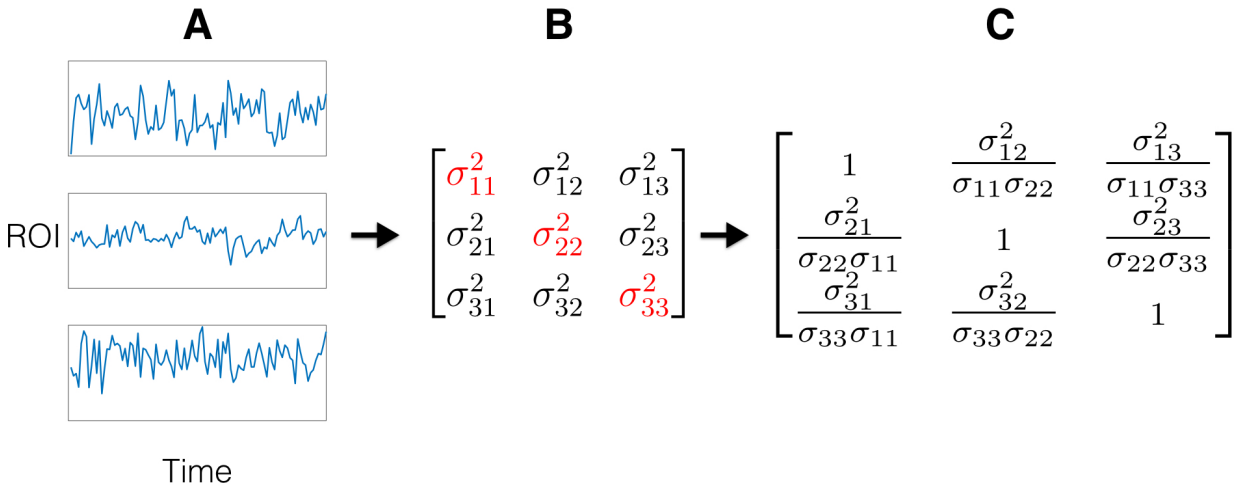

Figure 3.1: Example FC and BV computation. Time series for three ROIs (**A**) are used to compute the covariance matrix (**B**) where σ_{ij}^2 represents the covariation between ROIs i and j and the red diagonal entries represent BV. The covariance (**B**) can be used to compute the Pearson correlation matrix (**C**), where the ij-th entry of the matrix is $\sigma_{ij}^2/(\sigma_{ii}\sigma_{jj})$. FC can refer to either the covariance matrix, which explicitly includes BV, or the correlation, which indirectly includes information about the variance. FC is traditionally computed as the correlation and the diagonal of ones is discarded.

FC can be separated into two subcategories that have both been used to predict task: whole-brain (computed across the entire brain) and networks (computed between subsets of brain regions). Whole-brain FC has been used to accurately predict whether subjects are engaged in a task or at rest [117], to discriminate between subject-driven cognitive states [125], and to robustly track ongoing cognition. Furthermore, the ability to track states with FC has been

associated with measures of behavioral performance [55]. FC networks have similarly been shown to predict subject-driven cognitive states [130], have been associated with attention [78], and to accurately track task-evoked states [73].

Despite strong links between FC and task-evoked states, recent research suggests that the majority of the variance in FC is accounted for by "who you are and not what you are doing" [34, 35, 131]. Subjects exhibit individual resting state network architectures that are detectable in task-based fMRI [19]. These individual network architectures create a unique signature that can be used to accurately identify them within a group [34] – identification that is robust across both task and time. Individual resting state FC has also been used to predict changes in the BOLD signal across task conditions. For example, Tavor et. al. 2016 used resting state FC and gross brain morphology to accurately predict BOLD modulation across a range of cognitive paradigms, suggesting that individual differences in task-evoked activity are stable trait markers of underlying individual differences in resting state FC [131].

BV presents a different approach to study BOLD fluctuations that is also associated with task and subject. A series of neurocognitive aging experiments (for reviews see [51, 57]) showed age-related effects on task BOLD variability that are separate from and more predictive than the mean [46]. A follow-up study [47] identified regions that were associated with age, the speed of response, and consistency of behavioral performance. The difference in variability of high performance-associated regions versus low performance-associated regions was greater for younger, high-performing subjects. In a latent variable study, BV was linked to age, response time, and accuracy in a spatial working memory task. BV in neocortex was also associated with task-related disengagement of the default mode network [60]. BV has also been shown to be related to sub-optimal financial risk tasking among older adults [120]. In addition to age-related effects, individual differences in BV have been associated with lower visual discrimination thresholds [146]. BV has been also found to vary across task

conditions (fixation versus during task) [48] and to be associated with task-evoked activity [90]. BV has been linked to several physiological mechanisms. Increased BV has been linked to dopamine D1 receptor density in the caudate and DLPFC [60]. In a pharmalogical study, administration of a cholinergic enhancing drug was associated with improved performance during a matching task and lower FC and BV during the matching task, but not during a control task [116]. A study of older adults showed that greater BV was associated with better fluid abilities, better memory, and greater white matter integrity in all white matter tracts [12].

Taken together, these research results demonstrate that FC and BV reliably relate to task and subject differences. A key question is whether all of the changes in FC due to task and subject differences can be uniquely attributed to changes in the underlying network structure. Alternatively some of the effects might be related to changes in the BV (which we will refer to as functional variability, or BV) of individual regions that affect the measured FC. In this paper, we will apply two supervised machine learning approaches to assess the degree to which BV and FC are predictive of task and subject differences. If similar predictive performance can be achieved by BV relative to FC, it suggests that some of the statistical information contained in FC is also present in BV. By examining the difference in predictive performance between FC and BV, it is possible to assess the unique contribution of network-related information as opposed to changes in the variability of individual regions.

3.2 Materials and Methods

3.2.1 Data Acquisition

MRI recording was performed using a standard 12-channel head coil on a Siemens 3T Trio Magnetic Resonance Imaging System with TIM, housed in the Center for Cognitive and Behavioral Brain Imaging at the Ohio State University (OSU). BOLD functional activations for tasks were measured with a T2*-weighted EPI sequence (repetition time = 2000 msec, echo time = 28 msec, flip angle = 72 deg, field of view = 222 x 222 mm^2, in-plane resolution = 74 x 74 pixels or 3 x 3 mm^2, 38 slices with thickness of 3 mm). The resting state acquisition had higher resolution (repetition time = 2500 msec, echo time = 28 msec, flip angle = 75 deg, in-plane resolution = 2.5 x 2.5 mm^2, 44 slices with thickness of 2.5 mm). The T1-weighted brain volume (three-dimensional MPRAGE; 1 x 1 x 1 mm^3 resolution, inversion time = 950 msec, repetition time = 1950 msec, echo time = 4.44 msec, flip angle = 12 deg, matrix size = 256 x 224, 176 sagittal slices per slab; scan time 7.5 minutes) was acquired for each subject.

Stimuli were presented to subjects on a rear projection screen through a mirror on top of the head coil. Visual stimuli were generated on a Windows computer running Matlab programs based on Psychtoolbox extensions (http://psychtoolbox.org/). The subjects were recruited from the Ohio State University and the surrounding community, and gave informed consent. The experimental protocol was approved by the institutional review board at OSU. A total of 250 subjects participated in the study, but 174 of them (age 18 to 39, mean 21.6; 63 males and 111 females) were included in the data analysis. The subjects were excluded if, during any of tasks, part of the cerebral cortex was out of field of view due to head motion, or the mean frame-wise displacement of head motion was greater than 0.15 mm.

During the 1.5-hour MRI session, each subject performed eight behavioral tasks designed

to target basic cognitive function: emotional picture viewing [23], emotional face viewing [25], episodic memory encoding, episodic memory retrieval [88], Go/No-go [127], monetary incentive [77], working memory [148], and theory of mind stories/questions [28]. Only 5 of the 8 tasks had correct or incorrect responses, and thus be given a behavioral performance score. Resting state scans were also recorded for each subject. Each functional scan lasted about 6 minutes, ranging from 4.1 minutes for the episodic memory retrieval task to 8 minutes for the monetary incentive task. The task descriptions are presented in Table A.2 of the Appendix. For convenience of description, the resting state is treated as one of the 9 tasks.

Of the 174 subjects, 19 subjects returned and repeated the experiment on average 2.8 years (SD=0.4) later. We will refer to this group of subjects as the target group as all machine learning evaluations focus on this group.

3.2.2 Data Processing

All functional brain images were corrected for motion artifacts, spatially smoothed (2-mm FWHM Gaussian kernel), highpass temporal filtered (Gaussian-weighted least-squares straight line fitting, with sigma of 45 seconds), co-registered to T1-weighted image, and normalized to the standard brain and further refined using nonlinear registration in FSL (FMRIB software library, version 5.0.8, www.fmrib.ox.ac.uk/fsl). Linear regression was performed to remove task-related BOLD responses based on the design matrices of the tasks. Removal of task-related BOLD response ensures that patterns in the data reflect variation about the mean rather than mean BOLD activation. Due to inter-subject differences in hemodynamic response, as well as interregional differences in hemodynamic response timing, it is possible that some task-related BOLD response remains. While removing the mean BOLD trend may lower predictive performance, we emphasize using prediction to compare

BV and FC rather than as a goal in itself. Images were parceled into 299 regions of interest (ROIs) using a functional atlas derived by functional clustering of an outside dataset at the University of Western Ontario. The clustering uses a graph-theoretical approach [22]. In this group-level parcellation approach, resting state functional connectivity is represented as a graph where nodes are voxels and edges are functional similarity computed as the Pearson Correlation between voxels. The graph is locally spatially constrained so that only adjacent voxels (those sharing an edge or vertex) have nonzero similarity. The method 1) computes individual adjacency matrices using normalized cut spectral clustering (NCUT) [124], 2) averages the adjacency matrices (into a coincidence matrix), and 3) performs a group level parcellation on the group coincidence matrix using NCUT. A mask was used to remove edge voxels to prevent the machine learning classifiers from classifying subjects on the basis of edge-cortex misalignment artifacts created during brain co-registration. To create the mask, we removed any voxels that had low mean intensity in any scan. We removed all ROIs with any voxels that were removed (which is the most conservative approach for removing edge affects, e.g., as opposed to removing ROIs based on a threshold for percentage of voxels removed). After removal, 269 ROIs were left for analysis.

3.2.3 Feature Generation

We use the term FC to refer generally to any set of features that requires computing the covariance and BV to refer to any set of features that requires computing only the variance. For the time series from each task and subject, we compute FC using three different approaches that all depend on entries of the covariance matrix: 1) the Pearson correlation (FCP), 2) the off diagonal entries of the covariance matrix (FCC), and 3) the full covariance matrix (FCCV). FCP and FCC exclude direct information about the variance. However, FCP uses the variance as a normalizing term (see Figure 3.1). We compute BV using two different

approaches: the variance (BVV) and the standard deviation (BVSD). BOLD activation is de-meaned before computing features.

3.2.4 Machine Learning Approach

Our analysis consists of two prediction tasks, task prediction and subject identity prediction.

The goal of task prediction is to predict which task a test subject was performing during scanning given features computed from the scan (random performance in this task amounts to $1/9 = 11\%$ accuracy). The goal of subject identity prediction is to predict which subject generated a test scan given features computed from the scan (random performance in this task amounts to $1/174 = 0.57\%$ accuracy). Task prediction and subject identity prediction are evaluated in two settings: within-session and between-session. For within-session prediction, all training and test data are taken from session 1. For between-session prediction, training data are taken from session 1 and test data are taken from session 2. For task prediction, we exclude the session 1 scans from the target group from training so that the classifier learns from only task-related (i.e., not subject-related) information. Because fewer subjects participated in session 2, we restrict test sets to only the target group (i.e., 19 subjects that were scanned in both sessions 1 and 2), allowing us to directly compare within-session and between-session performance. However, note that for the subject identification task, the models are not informed of this restriction and have to discriminate between all 174 subjects who participated in the experiment.

We use multinomial logistic regression (LR) for task prediction and a nearest neighbor (1-NN) model for subject identity prediction to be consistent with previous analyses [34]. The models are evaluated differently as specified in the next section.

Multinomial Logistic Regression

Regularized multinomial logistic regression models [62, 132, 153] learn to discriminate between multiple class labels for a given data point. Feature weights are regularized using a choice of norm (L1, L2, elastic) and a parameter λ that controls the strength of regularization. We use an L2 penalty with regularization parameter $\lambda = 1$. Regularization usually results in improved generalization performance and is important in our analysis because it allows us to fit models using FC feature sets where the the number of features is larger than the number of data points. We used LIBLINEAR [32] to fit all logistic regression models. For each prediction task (task and subject identity prediction), we train independent models that take a set of features as input and output a class prediction (task or subject identity).

For all within-session prediction tasks, we use 5-fold stratified nested cross validation. The cross validation procedure is stratified in order to guarantee that a particular test subject always had some data used for training. For the within-session prediction tasks, the training set consists of a) all session 1 data not from the target group and b) a stratified 80% random partition of the session 1 data from the target group. The test set consists of the session 1 data from the target group that was not included in the training data. For the between-session task and subject identity prediction tasks, the training set consists of all session 1 data, and the test set consists of session 2 data (which only contains data from the target group). For each prediction task and setting, we use an inner cross validation loop over the training data to optimize the regularization parameter λ over the set $[0.001, 0.01, 0.1, 1, 10]$. For each training fold, we split our training data into validation-training and validation-test sets. For each of 2 validation folds, we train a model on the validation-training set, predict the labels of the validation-test set, and compute the validation fold accuracy. We compute the average validation accuracy (over 2 validation folds). To calculate an estimate of generalization accuracy, we average the average validation-test set accuracies over the 5

training folds. We use the parameter with the highest accuracy for testing. The inner cross validation loop is done using LIBLINEAR's built-in cross validation [32].

Nearest Neighbor Model

In contrast to LR models that learn from information across tasks, our 1-NN models are restricted to information from pairs of tasks where one is used for test and the other can be thought of as a training set. In principle, the 1-NN model could be set up analogously to the LR model, but we replicate analyses used in previous work [34] that were used to investigate whether functional signatures indicative of subject identity are preserved across pairs of tasks.

Each 1-NN model takes as input a test instance from task A and a set of labeled training instances from all subjects in task B, where each instance is comprised of features computed from a scan from a particular subject in a particular task. The predicted identity is the identity of the subject corresponding to nearest training instance, where we define similarity using the Pearson correlation. For between-session prediction, we iterate through all pairs of tasks A and B. For within-session prediction, we exclude pairs consisting of the same tasks (e.g., A-A) because each task was performed only once per session. To give a comparable set-up to LR task prediction, the test instances are always chosen from the target group and training instances are always chosen from session 1.

Credible Intervals on Classification Accuracy

For each model and prediction setting, we report a 95% credible interval on the classification accuracy. We model the classification outcome of the ith test instance as a Bernoulli random

variable x_i where the probability of a correct classification equals θ ($p(x_i = 1) = \theta$). Then the sum of classification outcomes $X = \sum_{i=1}^{N} x_i$ can be modeled as a Binomial random variable. Since we have no prior belief on the probability of correct classification, we place a uniform prior on θ ($\theta \sim \text{Beta}(1, 1)$). The posterior distribution of θ can be computed analytically as $p(\theta|\hat{\alpha}, \hat{\beta}) = \text{Beta}(\theta|\hat{\alpha}, \hat{\beta})$ where $\hat{\alpha} = 1 + X$ and $\hat{\beta} = 1 + N - X$. We report the 95% credible interval on probability of correct classification as the 2.5% and 97.5% percentiles of the posterior distribution over θ.

3.3 Results

First, we examine patterns in BV organized by subjects and tasks. Next, we show classification performance of the machine learning classifiers and contrast the relative diagnosticity of BV and FC in the three prediction tasks.

3.3.1 Visualizing BOLD Variability

Figure 3.2 shows BV for the target group subjects in session 1 (panel A) and session 2 (panel B). Rows are first grouped by subject and then by task. Columns are first grouped by brain lobe then by ROI. The results show subject-specific patterns in BV that are preserved between sessions. For example, subjects 2, 7, and 9 have relatively high BV in both sessions regardless of task and subject 18 seems to have relatively low BV in both session regardless of task. None of these subjects with outlying BV were outlying in demographic categories (weight, age, height, race). Subjects 2,7, and 9 have higher mean frame displacement than the other subjects (mean of 0.21 versus 0.09). Subject 5 has relatively low Frontal BV but average Occipital and Parietal BV. In addition, the results show lobe-specific effects that are

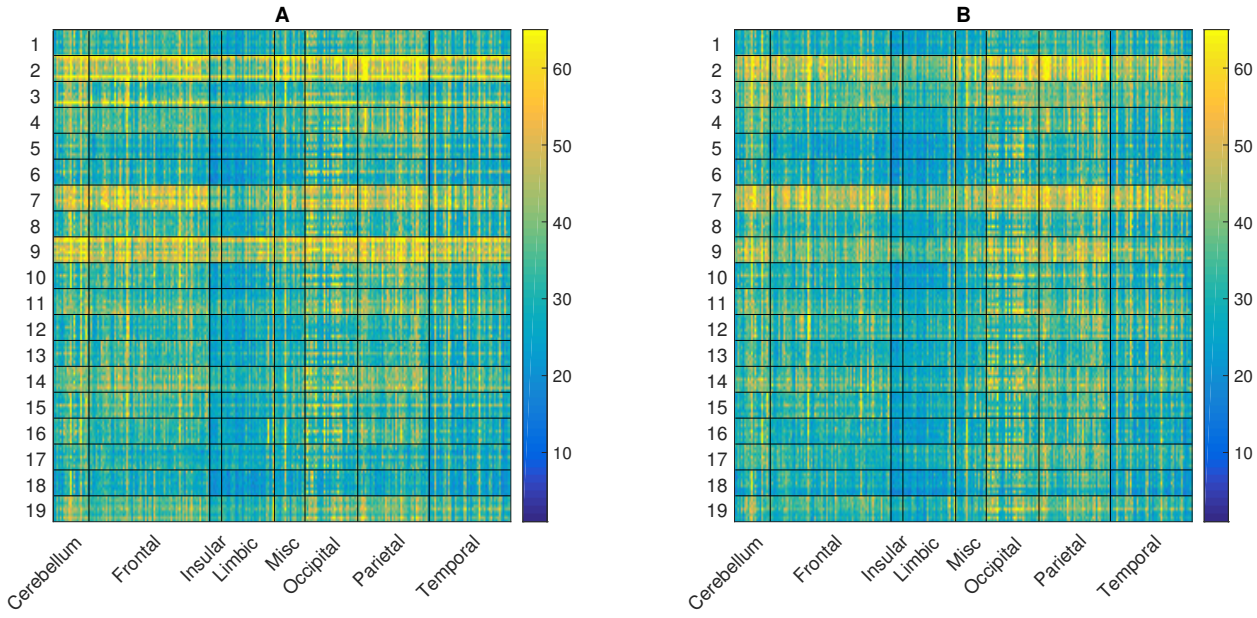

Figure 3.2: BV for 19 subjects from session 1 (**A**) and session 2 (**B**). The y-axis organizes scans first by subject and then by task. The x-axis organizes ROIs first by lobe and then ROI. Note that BV is computed by BVSD.

also preserved between sessions. For example, Limbic BV is on average lower than Parietal BV. The variance in BV between regions in the Occipital lobe is higher than in other lobes.

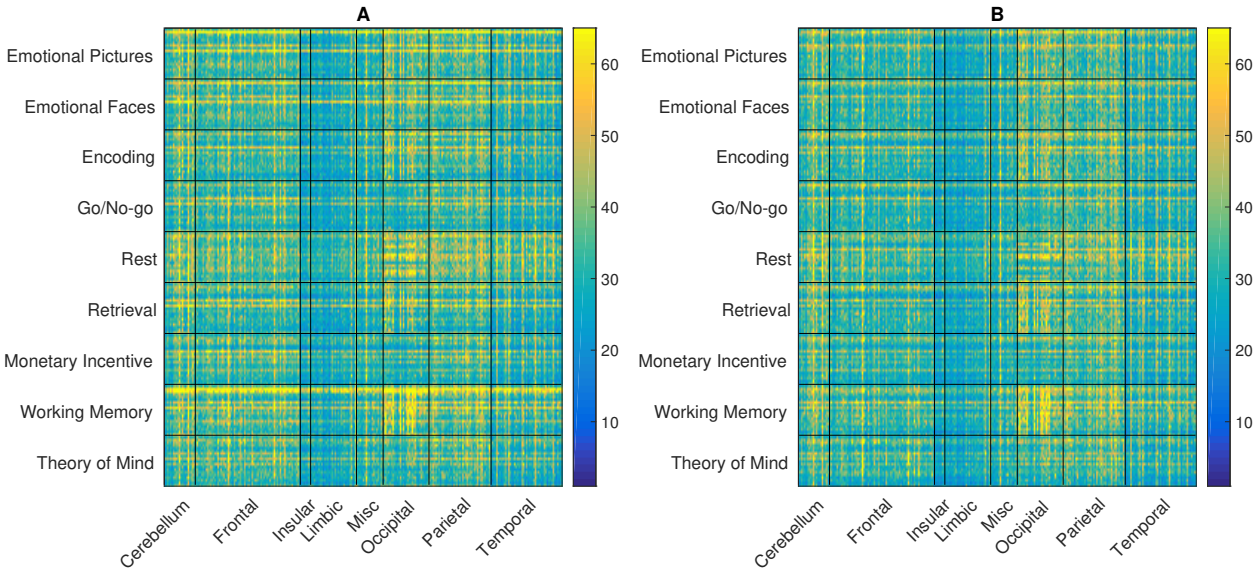

Figure 3.3: BV for 19 subjects from session 1 (**A**) and session 2 (**B**). On the y-axis are scans ordered by task. Within each task, scans are ordered by subject. On the x-axis are ROIs ordered by lobe of the brain.

Figure 3.3 shows BV (computed by BVSD) for the target group subjects in session 1 (panel A) and session 2 (panel B). Rows are first grouped by task and then by subject. Columns are first grouped by brain lobe then by ROI. When ordered by task, BV shows patterns that are preserved across session (e.g., lobe-specific or task-specific effects). For example, Occipital activation is higher for the Theory of Mind task, and temporal activation is higher during resting state. Aside from these two effects, based on visual inspection of Figure 3.3, the BV patterns do not seem to be task specific. However, the machine learning models (discussed in the next section) will demonstrate that the patterns contain diagnostic information to separate the tasks.

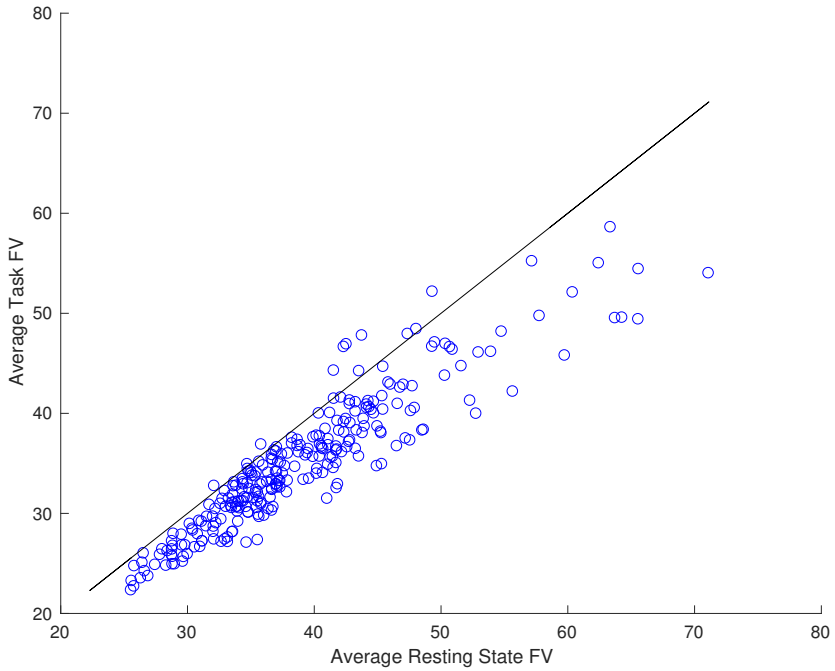

Figure 3.4: BOLD variability in a resting-state task versus non-resting cognitive tasks. Each point represents an individual ROI (averaged over subjects) and the reference line indicates equal BV in resting and non-resting state tasks.

Finally, we can re-examine known effects of FC through the lens of BV. For example, research has shown a reduction of covariance in the default mode network during task compared to rest [58]. We examine whether this result can be extended to BV. Figure 3.4 shows resting

Table 3.1: Predictive accuracy (percentage correct) of the Logistic Regression model for task classification for different methods of computing functional connectivity (FC) and BOLD variability (BV) and method for assessing generalization (within or between scanning sessions). The 95% credible interval is reported in parenthesis.

Type	**Feature**	**# Features**	**Within**	**Between**
BV	BVSD	269	79 (72, 84)	70 (62, 76)
BV	BVV	269	66 (59, 73)	60 (53, 67)
FC	FCP	$\frac{269*268}{2}$	95 (90, 97)	83 (77, 88)
FC	FCC	$\frac{269*268}{2}$	92 (87, 95)	82 (75, 87)
FC	FCCV	$\frac{269*269}{2}$	95 (91, 98)	84 (78, 89)

state BV versus non-resting state BV in each ROI averaged over all subjects and tasks. Analogously to the effect in FC, for almost all ROIs resting state BV is higher than task BV.

3.3.2 Task Prediction

Task out-of-sample LR prediction accuracy is reported in Table 3.1. Overall, BV and FC accurately predict task. All models show performance well above chance (1/9=11%) for all feature sets. However, there is a clear performance benefit when using FC versus BV (21% within, 18% between). Within-session performance is consistently better than between-session performance. The difference between within-session and between-session accuracy is lower on average for BV (7.5%) versus FC (14%), suggesting that FC contains more session-specific information than BV. Differences in accuracy do not simply reflect the number of features (independent of the information they contain) since redundant features would be regularized. The particular method of computing FC does not strongly affect predictive performance. BVSD leads to more accurate predictions than BVV, but this finding is not significant at 95% credibility.

In order to understand the relative performance differences between BV and FC, we compare

the confusion matrices in Figure 3.5. Some cognitive tasks are more difficult to discriminate on the basis of BV. For example, the Emotional Faces and Emotional Pictures tasks (both involving emotional processing despite different visual inputs) and Encoding and Retrieval tasks (both involving episodic memory) are occasionally confused on the basis of BV but less so for FC, suggesting that FC contains unique information that discriminates between these tasks. For a number of tasks (e.g., resting state or theory of mind) discrimination using BV is comparable to FC. For only a few cells in the confusion matrix does the FC model make more errors than the BV model (e.g., Rest-Emotional Faces, and Rest-Retrieval) and there is only one cell for which the FC model makes an error where the BV model does not (Rest-Retrieval). Additionally, the structure of errors is similar between BV and FC (i.e., the two models tend to make errors on similar pairs of tasks). The non-diagonal elements of the confusion matrices have a Pearson correlation of 0.75, suggesting that BV and FC make similar types of errors, but that BV makes those errors more often.

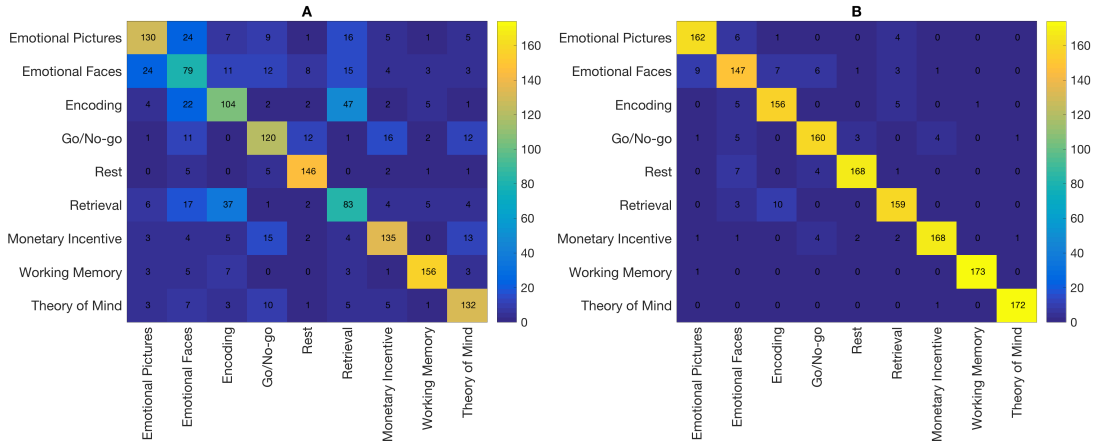

Figure 3.5: Confusion matrices for task prediction using BV (panel **A**) and FC (panel **B**). The y-axis corresponds to true task and the x-axis to predicted task. BV and FC were computed using the BVSD and FCP methods respectively

3.3.3 Subject Identity Prediction

Out-of-sample subject prediction accuracy and 95% credible intervals for the 1-NN models are reported in Table 3.2. Subject identity performance is high regardless of features used (chance performance is 1/174 = 0.57%). There is overlap in credible intervals for overall accuracy for all feature types except for BV computed as BVV, which performs significantly worse than the other methods examined. There is no significant overall performance advantage for any of the other methods used.

There are between-session performance differences based on whether the training and test images were recorded from the same task, from different tasks, or from rest. For all methods used to compute BV and FC, same task to same task accuracy is significantly higher than different task to different task accuracy (95% credible intervals do not overlap). For all methods but one, FC computed as the Pearson correlation (FCP), accuracy is significantly higher for same task to same task prediction compared to rest to rest prediction. Accuracy for different task to different task prediction is not significantly different than rest to rest accuracy. We suspect that given more rest to rest observations this difference would become significant; the relatively low number of rest to rest outcomes (one per subject) as compared to different task to different task outcomes (72 per subject) or same task to same task outcomes (8 per subject) leads to larger rest to rest credible intervals.

Pairwise Subject Identification Accuracy

The primary purpose of using nearest neighbor models is to build upon past results by Finn et al. 2015 that investigated how subject-specific FC is preserved across tasks. Figure 3.6 shows subject identity prediction accuracy as a function of training and test task. The x-axes show the training task and the y-axes show the test task. Prediction accuracy is averaged over

Table 3.2: Subject classification predictive accuracy (percentage correct) and 95% credible intervals for the Nearest Neighbor models using different methods of computing functional connectivity (FC) and BOLD variability (BV). For each model accuracy is testing within-session and between-session. For between-session accuracy we report whether the training and test image were selected from the same task, different task, or from rest.

Method	**# Feat**	**Within**	**Between**			
			All	**Same Task**	**Diff Task**	**Rest**
BVSD	269	83 (81, 85)	70 (67, 72)	93 (88, 96)	67 (65, 70)	58 (36, 77)
BVV	269	73 (70, 75)	54 (52, 57)	83 (76, 88)	51 (49, 54)	42 (23, 64)
FCP	$\frac{269*268}{2}$	83 (81, 85)	63 (61, 66)	80 (73, 86)	62 (59, 64)	53 (32, 73)
FCC	$\frac{269*268}{2}$	81 (79, 83)	63 (60, 65)	85 (78, 90)	60 (58, 63)	47 (27, 68)
FCCV	$\frac{269*269}{2}$	83 (81, 85)	67 (64, 69)	86 (79, 90)	65 (62, 67)	58 (36, 77)

subjects. The top panels and the lower panels correspond to FC and BV, respectively. The left panels and right panels correspond to between-session, and within-session, respectively. Within-session accuracy is higher than between-session accuracy for both FC and BV, which is consistent with classifier results in Table 3.2. For between-session prediction, performance is best when generalizing between the same tasks.

In the previous framework, Finn et. al. compared nearest neighbor subject identity prediction performance across sessions using FCP from resting state and another task and FCP two different tasks, as training and test sets. They found that rest-to-rest prediction is most accurate (92.9%) and that accuracy rates ranged from 54.0% to 87.3% for other database and target pairs, including rest-to-task and task-to-different-task comparisons. Our analysis includes the setting where the training and test sets from data recorded from the same task. We find that FCP prediction performance is highest for the task-to-same-task setting (80%), followed by task-to-different-task (62%), and rest-to-rest (53%). Overall, our average accuracy is lower (65% versus 82.1%), but the performance difference could be attributed to longer time between scanning session (2.8 years versus 2 days) and incorporating non-discriminative data (data from 155 non-target group subjects) into our framework.

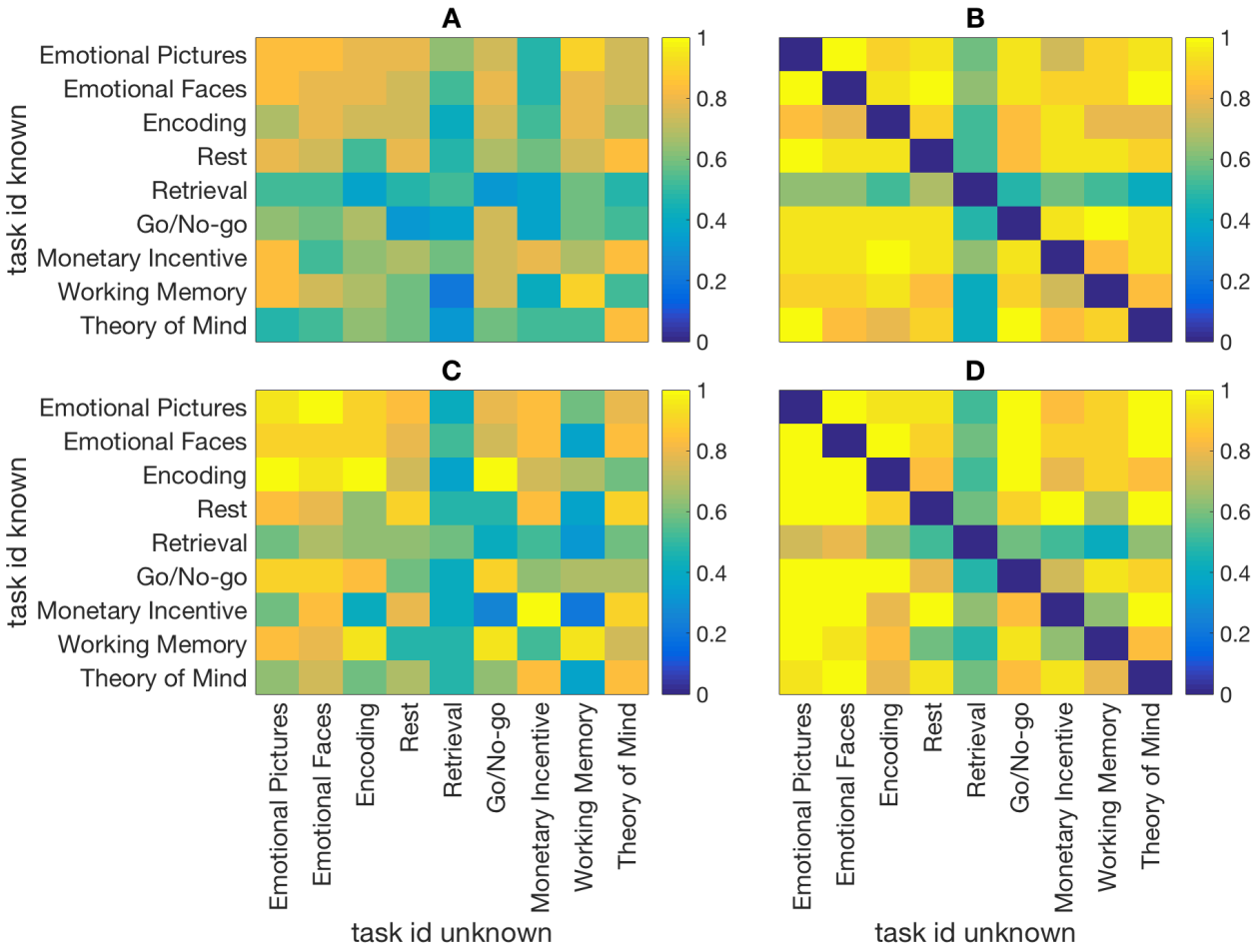

Figure 3.6: Heatmaps of between and within-session average subject identification accuracy ordered by task. The x-axis shows the task from which test scans were taken and the y-axis shows the task used to predict subject identity.

3.4 Discussion

Using two supervised machine learning approaches, we show that both BV and FC significantly predict task and subject differences above change levels. If we simply choose the statistic with highest predictive performance (FC), an important message gets lost: that the diagonal elements of the connectivity matrix (BV) contain a large amount of predictive information. Given that the off-diagonal elements consist of 36,046 data points in contrast to the 269 data points on the diagonal, one would expect models that use off-diagonal informa-

tion to greatly outperform models that use only diagonal information. However, this is not the case. The network structure of functional connectivity contributes to less improvement in performance than one might expect. Our results suggest that the theoretical connections between statistics related to the underlying network (FC) and statistics related to the variability of individual regions or voxels (BV) should be examined in order to disentangle the contributions of each. There are also several benefits from considering BV. BV offers a lower-dimensional functional signature than FC, making BV easy to visualize. In addition, the interpretation of BV results does not require network models that can complicate FC analysis [36, 139, 140, 106, 37].

An alternative explanation for the similarity in predictive performance of FC and BV is that our machine learning models could not adequately capture the valuable information present in FC given the relatively small lack of training data. While a larger training set and more advanced models would improve the performance of both FC and BV, we expect that these changes would better leverage the high dimensionality of FC and lead to better performance gains than BV. Since the goals of this work were to compare the information contained in BV and FC, we focused on simple ML models. Initially, both task and subject prediction was done using 1-NN models. However, subject effects dominate BV and FC similarity, which caused task prediction performance to suffer. Thus, we switched to using LR in order to allow our models to learn across task so that prediction was not based on BV or FC similarity alone, but could also incorporate differences between tasks.

Our nearest neighbor prediction framework mirrors past analyses [34] that investigated the persistence of subject-specific FC signatures across pairs of tasks, but extends the framework. Previous analyses did not include task-to-same-task prediction and found that rest-to-rest prediction performed best. Our analysis found that, for both FV and BV, task-to-same-task prediction performs best and that rest-to-rest prediction performs worst. These results

suggest that task engagement modulates subject fingerprints in a way that increases subject discriminability. The lack of this modulatory task effect in the Finn et. al. study could be due to the use of a different parcellation scheme; the parcellation method used by Finn et. al. focuses on preserving individual connectivity, whereas the graph-based method we used smooths more across individuals.

When we directly contrast performance in subject identity prediction and task prediction, we find that out-of-sample subject identity prediction is more accurate than task prediction, even though a priori the subject identity task is a more challenging task (identifying 1/174 versus 1/9). This provides further evidence that "the majority of the variance in [functional signature] is accounted for by who you are and not what you are doing" [35]. A recent study of task and subject FC expanded upon this idea by showing not only that FC individuality is a predominant factor in group-level FC variability, but that task sensitivity could be improved by removing subject connectivity [147].

There is debate in the field about whether subject-specific functional signatures are persistent across time. One long term study found that FC within a single individual changed over time and is paralleled by ongoing fluctuations in behavior, although many brain networks are largely stable [107]. Other studies found that parcellation of subject FC is stable over the span of a year [79], and that resting state FC in a single individual, and especially the executive resting state network, was stable over a three year period [16]. Our results show that FC and BV can be used to predict subject identity across time periods on the order of 3 years and suggest that subject-specific functional signatures are persistent across time.

One potential concern is that past research showed that vascular effects are present in motor tasks and to a much lesser extent cognitive tasks [71]. This research suggest a potential confound for our results: that vascular effects, rather than neural effects, lead to high predictive

performance. For two reasons, we believe that this is not the case. First, there were only moderate motor components to the cognitive tasks used in our experiment; the components involved reporting answers using button presses. We can expect the vascular effects due to motor control to be less for this task compared to the finger tapping task used in the Kannurpatti et al. study. Second, the similarity between motor components of each task (infrequent button pressing), suggests that even if large vascular effects were present, these effects alone would not be sufficient to discriminate between 9 separate tasks.

Another potential concern is that structural information separate from functional information contributes to the predictive performance. It is possible that differences in gross brain morphology create artifacts in functional signatures during the registration process [69] that affect both FC and BV measures. For the goal of predicting what cognitive task a subject is engaged in, only functional information can be used to distinguish between cognitive tasks. Therefore, the ability of the model to identify cognitive tasks demonstrates that both BV and FC contain diagnostic functional information and that these functional signatures persist over time. For the subject identity prediction task however, care has to be taken in interpreting the results. The identification of a person based on structural information is not an impressive outcome compared to the identification of a person based on functional connectivity or functional variability. For this reason, we did not use LR models for subject identification because they could easily overfit to a structural confound. The 1-NN classifier does not make use of any free parameters that can be tuned to particular ROIs and therefore the identification occurs on the basis of overall similarity between functional signatures and not any particular ROI. We also performed several during preprocessing to ensure that high subject identification performance was not due to structural confounds: we removed edge voxels (those most likely to be misaligned) from our analysis, used non-linear registration, and performed separate registration for each scanning session. Furthermore, brain parcella-

tion was performed using a dataset from a separate population, which reduces the probability that voxels were grouped into regions that a priori differentiate subject identity (i.e., that ROIs reflect subject specific rather than task specific functional differences). Therefore, even if structural information affects particular ROIs, it is unlikely that the classification results in the subject identification task are driven entirely by structural information. However, future research will have to investigate how structural information might contribute to classification performance.

3.5 Conclusions

Our results establish that BOLD variability, a much simpler approach than functional connectivity for investigating BOLD fluctuations is diagnostic of subject and task differences. We confirm the persistence of subject traits across task and session using BV and show that BV captures much of the information contained in FC. Overall, our work suggests that we should examine the theoretical connections between statistics related to the underlying network (FC) and statistics related to the variability of individual regions or voxels (BV). Since FC is indirectly connected to BV as a measurable BV is needed to have a measurable FC, a more in-depth investigation is needed to understand the meaning of our findings. Future work will examine how to disentangle effects of FC and BV using statistical modeling approaches.

3.6 Funding

This work was supported by a National Sciences Foundation Integrative Strategies for Understanding Neural and Cognitive Systems Collaborative Research Grant (1533500 and 1533661).

Chapter 4

Experimental Design Modulates Variance in BOLD Activation: The Variance Design General Linear Model

4.1 Introduction

At their core, neuroimaging analyses consist of relating a summary statistic of the blood-oxygen-level-dependent (BOLD) time course to experimental condition, behavior, or individual characteristics. The primary method for fMRI analysis, the General Linear Model (GLM) [42, 11], focuses on mean BOLD activation. Recently, researchers have moved beyond mean BOLD by studying functional connectivity (FC), which is calculated as the Pearson correlation between regions. However, it is possible that other statistics of the neuroimaging

data may contain important information. A natural candidate is BOLD variability (BV) defined as the variance in the BOLD time series. BV can be thought of as intermediate to mean BOLD and FC in terms of computational complexity; BV is based on a locally independent computations whereas FC incorporates between-regions dependencies. Importantly, as FC and mean BOLD have led to distinct avenues of research, BV could be a fundamentally different channel for studying brain function.

This article introduces the Variance Design General Linear Model (VDGLM), a novel framework that allows researchers to simultaneously test for effects on BV and on mean BOLD activation. The VDGLM can be conceptualized as a GLM that explicitly incorporates the experimental design into the model of the variance. Direct incorporation of the experimental design allows the VDGLM to be flexible enough to be used in any fMRI experiment. This new framework facilitates the analysis of BV effects and enables new discoveries that relate BV to disease, individual characteristics, and human behavior.

The development of the VDGLM was motivated in part by studies in EEG and fMRI that have demonstrated relationships between brain fluctuations and cognitive processes, behavior, and age. EEG studies have shown variance effects in the form of suppression of alpha and theta oscillatory waves (see [74] for a review). Alpha suppression is related to task engagement [142], opening the eyes [6], and sleep [27]. Alpha suppression is positively correlated with cognitive performance and memory performance, whereas the opposite holds true for the theta band [74]. Age studies have found that alpha, delta, and theta suppression increases with age during youth [128] and that alpha suppression decreases with age in older populations [30]. Event-related de-synchronization suggests that alpha suppression could be linked to inhibitory-control processes involved in attention [76, 75].

In fMRI, brain fluctuations measured by BV have been shown to vary with behavior and

age. BV varies between task and fixation, particularly in younger adults [48]. Differences in BV between task and fixation are associated with higher visual discrimination performance [146] and track task difficulty [50]. Age has also been related to BV; BV was shown to predict age with five times the explanatory power of mean BOLD [46] and was indicative of younger, faster, and more consistently performing subjects [47]. In both studies, the spatial distribution of BV effects was orthogonal to the distribution of mean effects. Despite these links to behavior and age, the study of signal variance has not been widely pursued in fMRI.

Whereas the aforementioned studies focused on BV, the studies did not use a general statistical framework for expressly studying these BV effects. Therefore, another key motivation for the VDGLM framework is to introduce a unified fMRI framework to allow the analysis of BV. The framework that we develop, the VDGLM, is a parametric approach that jointly models the mean and variance by explicitly incorporating the experimental design into the variance formulation. The inclusion of the design in the variance allows us to: *i)* jointly model mean and variance effects, *ii)* explicitly model the temporal dynamics between BV and experimental condition, and *iii)* include multiple experimental conditions in our variance analyses. The explicit structure of mean and variance design is supplied by the researcher, which allows for easy generalization to any experiment, and model fitting is computationally efficient enough to run whole-brain region of interest (ROI) analyses on large brain imaging studies. By developing this framework, we are providing an important tool to test for variance effects that has the potential to spur new research developments in various fields.

The plan for the rest of the paper is as follows. We first provide an overview of the GLM, and establish the theory behind the VDGLM. Next we provide an application of the VDGLM to Working Memory data from the Human Connectome Project Healthy Adult dataset. We finish with a discussion of the choices made when using the VDGLM and how variance

measured by the VDGLM compares to other measures of variance.

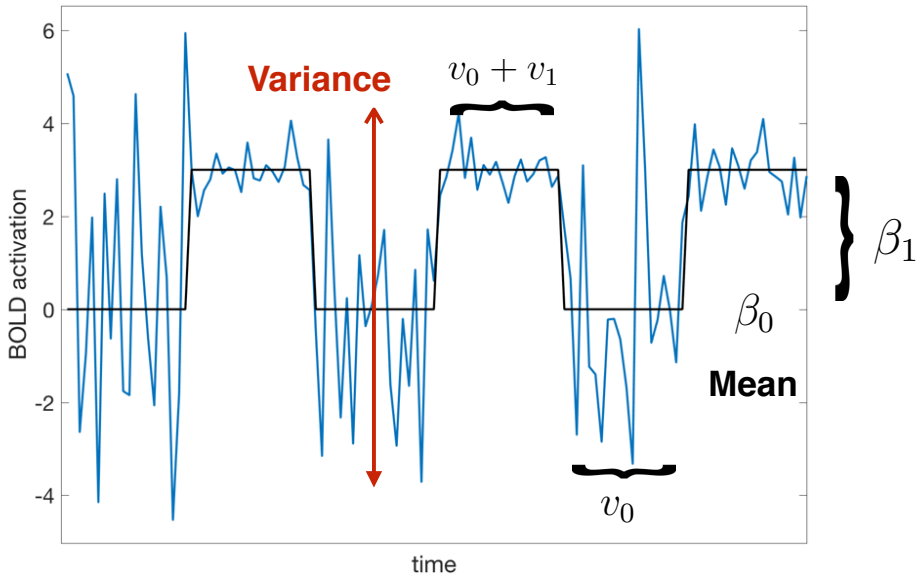

Figure 4.1: Illustration of artificial data where the presence of a single experimental condition (black) increases the mean but lowers the variance of the BOLD time course (blue). The values used to create this visualization are based on Equation 4.1 with $\beta_0 = 0, \beta_1 = 3, v_0 = 2$, and $v_1 = -1.5$

4.2 A Novel Framework for Studying BV

To motivate the VDGLM, consider a hypothetical BOLD activation time series where BV is affected by an experimental condition that indicates fixation versus task (see Figure 4.1). The single experimental condition is plotted in black and the BOLD time series from a single voxel is plotted in blue. For conceptual simplicity, the experimental condition time series is not convolved with a hemodynamic response function (HRF) model. The voxel time series varies as a function of task condition; the variance is higher during fixation compared to

task. We can describe these effects on the mean and variance using a simple VDGLM:

$$y \sim N(\beta_0 + x^T\beta_1, (v_0 + x^T v_1)I) \tag{4.1}$$

where y is the BOLD time series, x represents the condition indicator, I is the identity matrix, β_0 captures the mean activation, and v_0 captures the measurement variance, i.e., the out-of-task variation. Then β_1 and v_1 capture the degree of change in mean and variance due to task engagement, respectively. If the model is applied to the data from Figure 4.1, we expect $\beta_0 \approx 0$, $\beta_1 \approx 3$, v_0 to be a large positive value, and v_1 to be negative, but with the constraint that $|v_1| < |v_0|$. Here, the parameter $v_1 < 0$ reflects the fact that BOLD variation is lower within task compared to fixation.

For comparison, the GLM estimates a single variance parameter over the entire time series and ignores the change in variance due to the experimental manipulation:

$$y \sim N(\beta_0 + x^T\beta_1, v_0 I) \tag{4.2}$$

Note that Eq. 4.2 is equivalent to Eq. 4.1 when $v_1 = 0$, i.e., the GLM is a nested model of the VDGLM where there is no experimental modification of the variance. The GLM is a null model for no effect of the variance that can be compared to the VDGLM *i*) to explicitly test for variance inclusion and *ii)* test whether the mean effects found by the VDGLM are similar to the mean effects found by the GLM.

4.2.1 VDGLM Analysis Pipeline

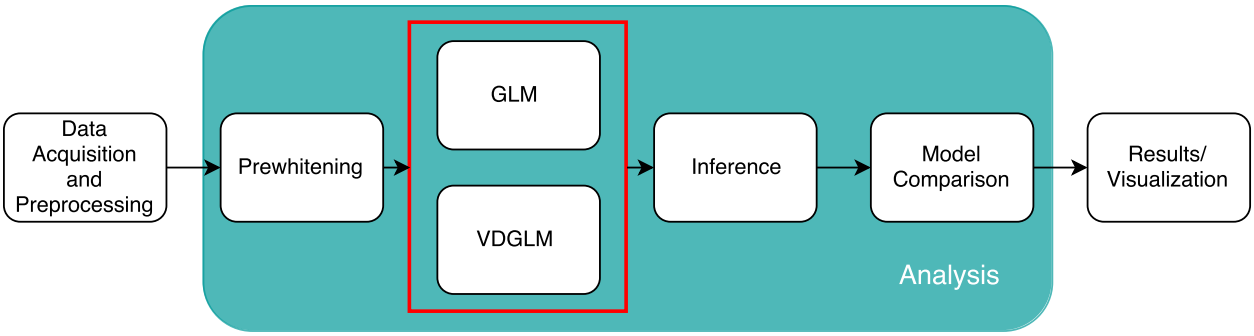

Figure 4.2: An illustration of a typical fMRI pipeline that uses either the GLM or the VDGLM for analysis. To use the VDGLM, the only steps from a traditional pipeline that must change are model formulation and estimation. Data acquisition, preprocessing, prewhitening, model comparison, and methods for result dissemination can remain the same. Some inference steps, such as effect size estimation, could remain the same. However, other inference procedures, such as significance testing, require statistics developed expressely for the VDGLM.

One goal of the VDGLM framework is to allow the VDGLM to be inserted into any standard fMRI analysis pipeline with minimal modifications (see Figure 4.2). The VDGLM does not change data acquisition, preprocessing, prewhitening, model comparison, or results dissemination. The main step that must change is model formulation and estimation. In some cases, the inference step is not affected (e.g., computing effect sizes using Cohen's d). However, more sophisticated inference such as parameter significance testing will require modification of the inference step to include statistics for testing variance parameters (see section 4.2.2). In a BV focused analysis, we also recommend additional preprocessing steps to remove variance confounds such as removing white matter and CSF signal and correcting for head motion, but these additional steps are not necessary to use the VDGLM.

4.2.2 Matrix Notation

We can write the GLM and the VDGLM in matrix notation to highlight the concept of inserting the design matrix into the variance. The GLM models the BOLD time series $y_{[T\times 1]}$ from a single voxel as a linear function of the experimental design [41, 11, 5, 143]. Formally, the GLM is defined:

$$\begin{aligned} y &= X\beta + \epsilon \\ \epsilon &\sim N(0, I\sigma^2) \end{aligned} \tag{4.3}$$

where X is a $T \times p$ design matrix, β is a $p \times 1$ vector of mean parameters, σ^2 is a variance parameter, and I is the identity matrix. The columns of the design matrix X include experimental events, experimental blocks, stimulus presentation, or mean activation. The VDGLM has the same formulation, but extends the variance model:

$$\begin{aligned} y &= X_m\beta + \eta \\ \eta &\sim N(0, \mathrm{diag}(X_v v)I) \\ \mathrm{diag}(X_v v)I &> 0 \end{aligned} \tag{4.4}$$

where $\mathrm{diag}(x)$ is the matrix with the entries of x along the diagonal. To emphasize that the mean and variance design matrices can be distinct, we use the notation X_m and X_v to denote the mean and variance designs, respectively. The parameters, β and v, capture mean

and variance effects, respectively. It is clear that the GLM (eq. 4.3) is a special case of the VDGLM for which the variance design matrix X_v is a single column of ones and $v = \sigma^2$.

Prewhitening and Noise Regressors

In GLM analyses, BOLD time serires are 'prewhitened' to account for BOLD autocorrelation [11, 144]. We also prewhiten before fitting the VDGLM to ensure that variance effects found by the VDGLM are not artifacts caused by autocorrelation. In VDGLM, as with GLM analyses, any standard autocorrelation estimator can be used [144, 39, 21]. In theory, one could use the residuals from either the VDGLM or the GLM to estimate the autocorrelation. We choose to use the GLM residuals for two reasons. First, the GLM is less computationally intensive than the VDGLM, and we expect that the VDGLM leads to similar residuals since in practice we've found the mean trend for the VDGLM is similar to the mean trend for the GLM when fit to unwhitened data. Second, by using the GLM, any variance signal that could be accounted for by either autocorrelation or the VDGLM is by default attributed to autocorrelation. Since we remove this autocorrelation using prewhitening, the whitened data will lead to more conservative estimation of variance effects than if we had used the VDGLM for prewhitening (i.e., it is less likely that artifactual autocorrelation will lead to detection of variance effects).

Other techniques for controlling noise (e.g., coloring or head motion correction) involve the addition of noise regressors. As with GLM analyses, the VDGLM can incorporate these techniques by including the appropriate regressors in the design matrices.

Estimation

Univariate GLM mean parameter estimation can proceed in one of two ways: *i*) ordinary or general least squares [41, 5] or *ii*) or *ii*) fully Bayesian inference [143]. Approximate Bayesian inference has also been used, but in a single group-level analysis that combines first and second-level models [43]. Variance estimation is usually done using by iteratively computing OLS estimates [144, 145], but can require more advanced methods depending on the structure of group-level models (see section 4.2.2).

There are potentially many approaches that could be used to estimate the VDGLM, including Bayesian and maximum likelihood approaches. For simplicity, we use a maximum likelihood approach. Estimation approaches must be computationally efficient enough to handle the high dimensionality of fMRI data. In practice, we found that sampling techniques were too slow to be practical for large data sets. Maximum likelihood (or maximum a posteriori) estimation using mode-finding algorithms is efficient enough to estimate parameters for an ROI analysis from a large fMRI dataset in about half a day using parallel computing techniques.

Group Level Analysis

Group-level GLM analyses typically incorporate two-stages, in which second stage analysis is based on summary statistics from the first [63, 5, 143]. The methodology for group-level significance testing depends on the experimental design. T-tests can be used provided that the experiment is balanced [63]. For unbalanced data, if the variance components of the data are known, then principled group-level inference can be done using univariate parameter estimates and their covariance estimates [5]. In most cases, these variance components are not known. Second-level variance parameter estimates have been found using the EM algorithm [145], approximate Bayesian inference [43] and fully Bayesian inference [143]. These

same ideas extend to the VDGLM. For the simple balanced-experiment setting, group-level inference can be computed using t-tests. In the unbalanced case, more work is needed due to the difficulty in computing the covariance of variance parameters. We initially tried to develop group-level inference procedures using asymptotic statistics (Wald test), but these tests were ill-behaved for several subjects due to high-condition matrix inversions. We leave development of alternative statistics and a fully Bayesian framework to future work.

Group effect sizes can be estimated using the set of parameter estimates from all subjects. In our application, we compute Cohen's d, which measures the standardized mean between two populations.

Model Comparison

Model comparison also proceeds as in a traditional fMRI pipeline. Model comparison can be done using AIC [1], BIC [121], or any other log-likelihood-based metric that is a function of a point estimate. Model comparisons can consist of traditional in-sample comparisons or can be generalized to new data using out-of-sample comparisons [97]. The outcomes of univariate comparisons can be aggregated into group level results that test whether a subject tends to prefer a certain model across the brain or whether a particular region tends to prefer a certain model across subjects.

4.3 Example Application: BV in Working Memory

In this example application, we used the VDGLM to find brain regions that are involved in working memory via changes in BOLD variance. We examined whether these regions differ from regions involved via changes in mean BOLD activation, and tested whether the

VDGLM better describes working memory data than the GLM. The goal is to illustrate how to use the VDGLM and to showcase its utility.

We used data from the Human Connectome Project (HCP) Working Memory Experiment. In the experiment, subjects alternated between fixation blocks and two different task blocks during which they were presented with sequences of visual stimuli. In a 2-back task block, subjects indicated whether the current stimulus was the same as one two presentations ago. In a 0-back task block, subjects indicated when a target stimulus was presented.

GLM analyses of the HCP data have found that engagement in the 2-back working memory task invokes regions thought to be involved in a cognitive control network, including bilateral dorsal and ventral prefrontal cortex, dorsal parietal cortex and dorsal anterior cingulate. Task engagement leads to a deactivation in the default mode network, namely in the medial prefrontal cortex, posterior cingulate, and the occipital parietal junction [4]. Similar activation patterns are found even when comparing 2-back versus 0-back. A 24 study meta-analysis of N-back studies found consistent activation in frontal and parietal areas, namely bilateral and medial posterior parietal cortex, bilateral premotor cortex, dorsal cingulate/medial premotor cortex, bilateral rostral prefrontal cortex or frontal poles, bilateral dorsolateral prefrontal cortex, and bilateral mid-ventrolateral prefrontal cortex or frontal operculum [102].

In our VDGLM analysis, the goal was to find both mean effects that overlap with known mean effects and also variance effects that are spatially orthogonal to known mean effects.

4.3.1 Methods

Data Acquisition and Preprocessing

The data was collected by the Washington University Minnesota Consortium Human Connectome Project (HCP, Van Essen et al., 2013). We used the Working Memory task data from the 1200 Subjects release using the minimal pre-processing pipeline [53]. Details of task fMRI processing can be found in [4]. Data from 875 subjects with one of the two Working Memory tasks were included in the current study. The downloaded data were in grayornidate system [53], and the time series for 333 surface regions of interest (ROIs) based on Gordon et al. were extracted for further analysis [56]. We perform additional preprocessing including scrubbing and motion correction.

Task Design

During the Working Memory experiment, subjects alternatively engaged in a 0-back and 2-back tasks that use faces, places, tools and body parts as the four categories of stimuli. Within each run, subjects were presented with blocks of stimuli, where all stimuli within a block were from the same category. For half of the blocks, subjects were given a "target" stimulus and were instructed to press a button whenever that stimulus was presented (0-back task). For the other half of blocks, subjects were instructed to respond when the stimulus was the same as the one presented two presentations ago (2-back task). Task blocks were interwoven with 15 second fixation blocks and instruction cues indicating the task type and 'target' stimulus if the task was the 0-back task. Each run contained 8 task blocks. We combined blocks from each stimuli type to create two task indicators (one for 0-back and one for 2-back). In total, the experimental design contained four conditions: the 0-back task,

the 2-back task, Fixation, and Instruction.

Modeling

We applied the VDGLM model (eq. 4.4) to the data. Building the VDGLM required specifying both the mean and variance design matrix. In the mean design, we included the 0-back, 2-back, Fixation, and Instruction conditions. In the variance design we included the the same regressors as in the mean design, but with an additional intercept regressor to reflect the assumption that there exists some measurement noise not captured by the other variance regressors.

We also fit a GLM (eq. 4.3) model to the data using a design matrix that is equivalent to the mean design matrix used in the VDGLM.

Prewhitening The first step in model estimation is to prewhiten the data. We fit a GLM model from which we computed the residuals and then used an AR(2) process to estimate residual autocorrelation. We chose the AR(2) process because it has been shown to outperform standard autocorrelation estimators on tests of autocorrelation present after prewhitening [80]. An AR(2) process models the GLM residuals r_t at time t as:

$$r_t = \phi_1 r_{t-1} + \phi_2 r_{t-2} + \epsilon \tag{4.5}$$

where ϕ_i measure the contribution of the i-th autoregressive component and ϵ $N(0, \sigma^2)$ is white noise. We estimated the autoregressive parameters using the Yule-Walker equation [152, 138], from which the estimated autocovariance was generated using a simple parametric form [80].

Estimation After prewhitening, we estimated GLM parameters using ordinary least squares. Since the VDGLM is analytically intractable, we estimated parameters using constrained trust-region optimization [96] (see Appendix A.4 for optimization details, and [151] for a review of trust-region optimization). We performed mass univariate estimation, i.e., we fit the VDGLM and the GLM for each ROI and subject. From parameter estimates, we created parameter contrasts for the 2-back minus Fixation, 0-back minus Fixation, and 2-back minus 0-back conditions.

Group Level Effect Sizes We estimated group-level effect sizes for each contrast using Cohen's d (the difference in standardized means) computed over subjects [18]. We visualized these effect sizes for each ROI using the HCP workbench software [87]. For a single region there exist 3 possible group-level effect patterns on BOLD: 1) both mean and variance effects are shown, 2) either mean or variance effects are shown, or 3) neither type of effect is shown. We plotted the whole-brain spatial distribution of each type of effect at small and medium effect sizes (Cohen's d of 0.2, 0.5, respectively). We compare the spatial distribution of VDGLM mean effect sizes versus the spatial distribution of voxel-wise GLM effect sizes from HCP analyses (see Figure 4.3) .

Model Comparison Because the VDGLM has more parameters than the GLM, it has the potential to explain more variability of the observed data, thus any model comparison metric should take complexity into account. We achieved this using out-of-sample log likelihood (OOSLL), which penalizes overfitting by testing how well a model generalizes to new unseen data. We used an out-of-sample metric, rather than traditional metrics of model fit (such as goodness of fit tests, or information criteria) to balance the goals of our analysis between prediction and explanation [149]; the model was compared to other models using predictive performance, but also evaluated on its explanatory power. We used 10-fold cross

validation to compute the out-of-sample log likelihood. For a single subject and region, we split the time series into 10 folds that each contain a test and training set. For each fold, we fit our model on the training set and computed the out-of-sample log likelihood of the test set given the parameters computed during training. To understand the level of general preference for the VDGLM, we compute the percent of subject/ROI time series with higher OOSLL for the VDGLM compared to the GLM. To understand subject VDGLM preference, we compute the percent of regions that prefer the VDGLM model for each subject.

To test that our model comparison results did not occur by chance, we compared the prevalence of VDGLM preference found in the real data to that found for a dataset simulated from the GLM (i.e., data without variance effects). In this comparison, we wanted to make explicit assurances that the VDGLM was preferred because of real effects in the data, i.e., that preference was not due to autocorrelation artifacts. We did this by adding autocorrelation to the dataset simulated from the GLM, where we estimated the autocorrelation from the real data. We generated a time series for each subject and ROI independently. The generation of a sample time series from a real time series y_k proceeded as follows:

1. Compute the GLM OLS solution $\hat{\beta}$ and variance solution $\hat{\sigma}^2$. We use GLM parameters estimated from the real data to better account for subject heterogeneity than if we simulated the underlying GLM parameters.

2. Estimate the autocorrelation of the residuals $y_k - X * \hat{\beta}$ using an AR(2) process and generate an estimated autocovariance matrix A.

3. Generate a sample time series $y_{\text{samp}} = X\hat{\beta} + \epsilon_{\text{samp}}$, where $\epsilon_{\text{samp}} \sim MVN(0, \hat{\sigma}^2 A)$.

For each subject and region, we generate a single sample time series (we only generate one sample to reduce the computational complexity of this test). Using the generated dataset,

we fit the VDGLM to each subject and region. Due to having just one sample from each subject and ROI, we cannot make statements about whether model comparisons for a single subject and ROI are due to chance. However, we can analyze the percent of ROIs for a given subject that prefer the VDGLM to assess whether the amount of subject-level preference is due to chance. We compute whether the subject-level VDGLM preference is greater in the actual data compared to the simulated data. This test shows whether VDGLM preference is caused by overfitting to autocorrelation or whether there is true variance-related signal in the data.

Model fitting and model comparison for 875 subjects took approximately 1/2 a day to fit on the UCI High Performance Cluster using in-house MATLAB code that can be found online [52].

4.3.2 Results

The analyses we ran on the VDGLM were designed with three goals in mind. First, we wanted to test for the existence of effects on BOLD variation during working memory engagement. We did this by computing effects size of parameter estimates. Second, we wanted to see whether these effects occur in regions that are spatially orthogonal to regions that exhibit mean effects. To do this, we visually examined whether mean and variance effect sizes are correlated and plotted whole-brain visualizations of where mean and variance effects occur. Finally, we wanted to verify that the VDGLM provides a better account of the data than the GLM using model comparison metrics based on out-of-sample log-likelihood.

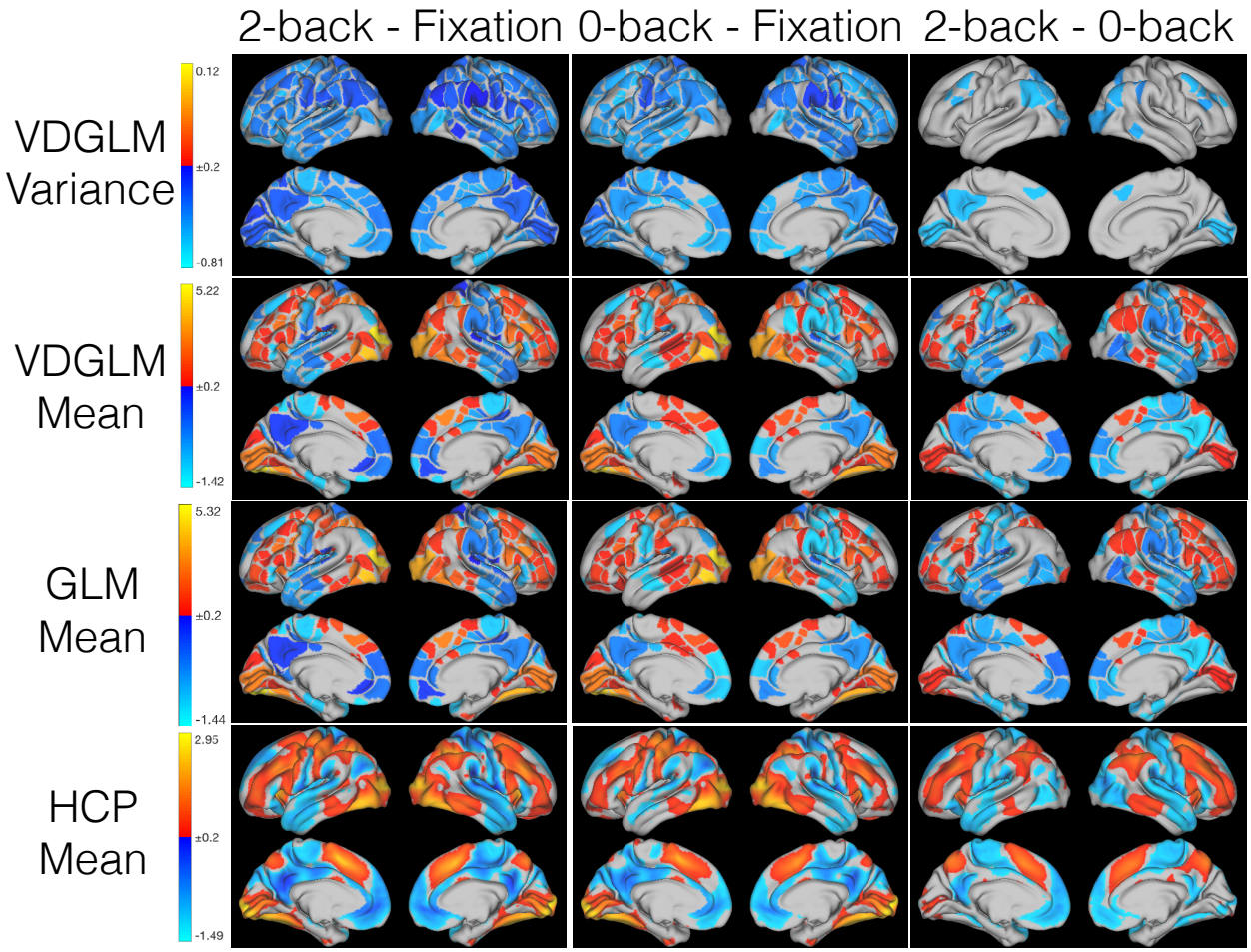

Figure 4.3: Group-wise Cohen's d for the 2-back minus Fixation, 0-back minus Fixation, and 2-back minus 0-back contrasts. The top row shows VDGLM variance effects, the middle row shows VDGLM mean effects, and the bottom row shows voxel-wise GLM results from previous analysis. Maps are thresholded at (-0.2, 0.2).

Group Level Effect Sizes

The first goal in our analysis was to test for existence of variance effects caused by working memory engagement. We measured effects by computing Cohen's d over subjects. For each parameter contrast (2-back minus Fixation, 0-back minus Fixation, and 2-back minus 0-back) we plotted whole-brain Cohen's d (see Figure 4.3, bottom row). We also plotted Cohen's d for mean parameter contrasts (top row) to verify that the VDGLM preserves known mean

effects.

We found small and medium sized variance effects during the 2-back and 0-back tasks compared to fixation across much of the entire brain. Both 2-back and 0-back engagement evoked less BOLD variation (i.e., negative Cohen's d) compared to Fixation across the whole brain. For the 2-back minus 0-back contrast, variance Cohen's d was low for some areas in the default mode network, some areas of the dorsal/ventral attention networks, some parts of visual cortex, and some parts of the fronto-parietal network. The 2-back task showed less variation than the 0-back task in occipital, temporal, parietal, and ventromedial prefrontal cortex.

The VDGLM found mean effects that overlap existing HCP GLM results [4] (see Figure 4.3). The 2-back minus Fixation and 0-back minus Fixation contrasts showed activation in bilateral dorsal and ventral prefrontal cortex, dorsal parietal cortex and dorsal anterior cingulate and deactivation in the default mode network, including medial prefrontal cortex, posterior cingulate, and the occipital parietal junction. These same regions were activated, but less intensely for the 2-back minus 0-back contrast.

In general, task engagement lead to both positive and negative mean effect sizes, but predominantly negative variance effect sizes.

Orthogonality of Mean and Variance Effects

The second goal in our analysis was to examine whether the VDGLM finds variance effects that are orthogonal to known mean effects. To analyze the degree of orthogonality between mean versus variance effects, we plotted regional Cohen's d for the mean contrasts versus variance contrasts (See figure 4.4). We grouped ROIs by effect size. ROIs with the same

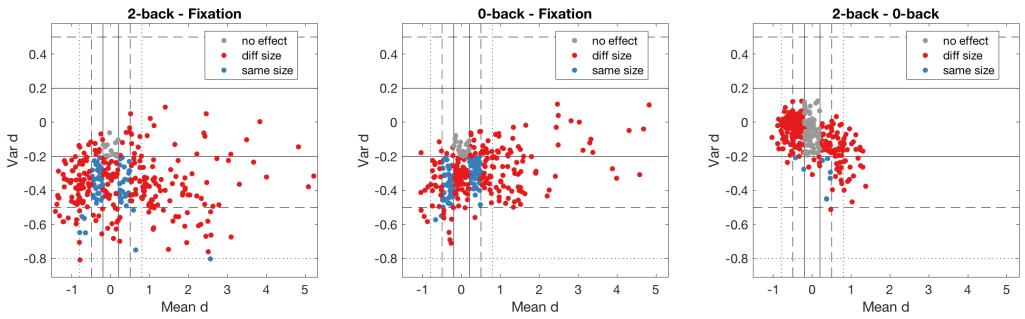

Figure 4.4: The magnitude of mean and variance effects sizes for the 2-back minus Fixation, 0-back minus Fixation, and 2-back minus 0-back contrasts. Each circle represents an ROI. ROIs are grouped by whether they exhibit the same size effect (blue) in the mean and variance or different sized effects (red). The black lines indicate the small (solid), medium (dashed), and large (dotted) effect sizes.

effect size in the mean and variance are plotted in red, those with difference effect sizes in the mean and variance are plotted in blue, ROIs with no mean nor variance effects are plotted in gray. The small, medium, and large, effect thresholds are plotted by the solid, dashed, and dotted black lines, respectively. ROIs exhibited mean and variance effects that span all possible combinations of effect sizes, although there were no large variance effects for the 0-back minus Fixation nor 2-back minus 0-back contrasts (see Figure 4.4). In general, mean effects were larger than variance effects. Effects were also much larger for the 2-back minus Fixation and 0-back minus Fixation contrasts compared to the 2-back minus 0-back contrast. While there was slight negative correlation between mean Cohen's d and variance Cohen's d for the 2-back minus 0-back contrast ($R^2 = 0.31$), the other two contrasts were uncorrelated ($R^2 = -0.00294$ and 0.132). Hence, mean and variance effects are orthogonal for the 2-back minus Fixation and 2-back minus 0-back contrasts.

Spatial Distribution of Effects

Given that mean and variance effects were orthogonal for the 2-back minus Fixation and 2-back minus 0-back tasks, we wanted to see where each type of effect occurs in the brain.

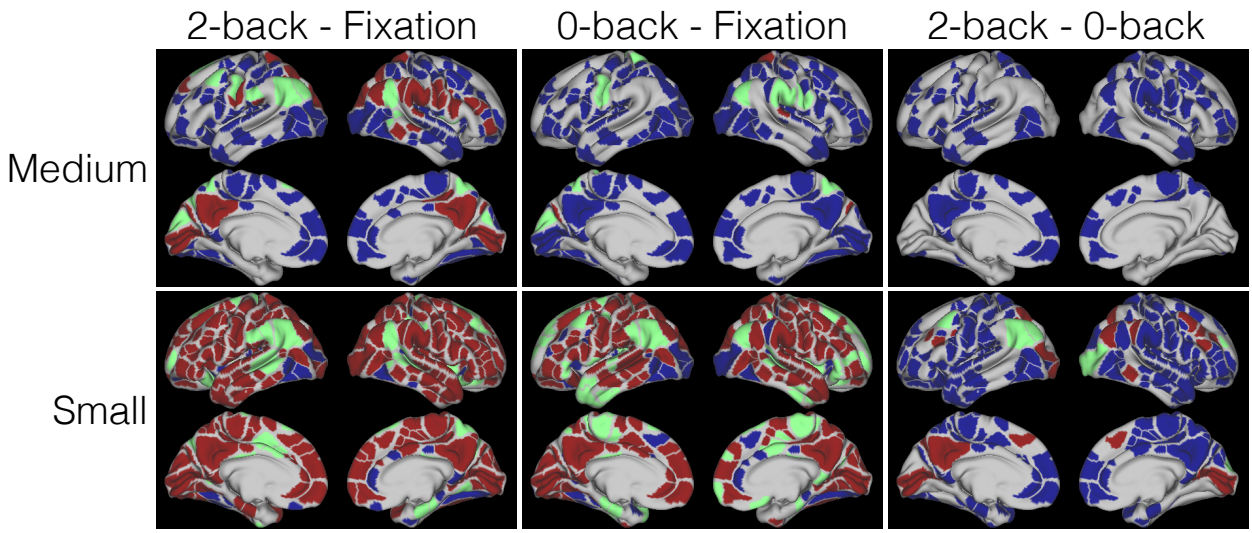

Figure 4.5: The figure shows which types of effects occur in which regions. Regions can have only a mean effect (blue), only a variance effect (green), or both effects (red). We plot effects for the 2-back minus Fixation, 0-back minus Fixation, and 2-back minus 0-back contrasts. We plot small (Cohen's d $\in [-0.2, 0.2]$) and medium (Cohen's d $\in [-0.5, 0.5]$) effects.

We plotted the type of small and medium effects that occur in each region (see Figure 4.5). A region with mean effect only is plotted in blue, variance effect only in green, and both effects in red. For the 2-back minus Fixation contrast, there are regions that exhibited all types of effects. Variance only effects occurred primarily in the default mode network, but also in sensorimotor mouth regions, and regions in the visual, dorsolateral attention, and cingulo-opercular networks. Mean only effects occurred in the frontoparietal, the auditory, cingulo-opercular, visual, and dorsolateral attention networks. Effects overlapped in some regions of the default mode, frontoparietal, visual, dorsal attention, and cingulo-opercular networks and in some sensorimotor regions. For the 0-back minus Fixation contrast, less regions exhibited medium variance effects than the 2-back minus Fixation contrast, suggesting that cognitive demand plays a role in the size of variance effects. There were, however, effects in sensorimotor areas used in control of the mouth and hand, a region in visual cortex, and some regions in the dorsolateral attention network. These effects were all present in the 2-back minus Fixation contrast as well. Mean 0-back minus Fixation effects also largely mirror

the mean effects in the 2-back minus Fixation tasks. There were no medium sized variance 2-back minus 0-back contrasts. This suggests that while there were different sized effects between 2-back minus Fixation and 0-back minus Fixation, these differences in effect sizes were small. The main differences in variance effects tended to be caused by task engagement rather than cognitive load.

Model Comparison

The VDGLM inferred that engagement in a working memory task leads to less BOLD variation. In this section, we pursue a different question. Does a model with these additional parameters give a significantly better account of the BOLD time series than a model without them? To test this, we perform model comparisons between the VDGLM and the GLM. Since the VDGLM has more parameters and will trivially better fit the data, we use out-of-sample log-likelihood to check the VDGLM's ability to describe new unseen data. We perform a model comparison for each subject and ROI time series in a mass univariate approach.

We found significant preference for the VDGLM model; 41% of subjects/ROIs had higher OOSLL for the VDGLM model over the GLM in the real data (7% in the simulated data). Model preference varies by subject and region (see Figure 4.6). The figure plots the percentage of ROIs that prefer each model in the real data (blue) and the simulated data (red). The figure is ordered by a subject's proportion of ROIs that prefer the VDGLM model in the real data. A subject's percentage of regions that favored the VDGLM ranged from 14% for GLM-leaning subjects to 73% for VDGLM-leaning subjects. For all 875 subjects, some number of regions (but not all) preferred the VDGLM model. Similarly, for all 333 ROIs, some number of subjects (but not all) preferred the VDGLM model.

To check that the VDGLM was not just fitting autocorrelation, we performed model com-

parisons for models fitted to data generated from the GLM plus autocorrelation. For the simulated data, only 7% of subjects/ROIs preferred the VDGLM model. For every subject, the percent of ROIs that preferred the VDGLM model was larger in the real data than in the simulated data, indicating a significant preference for the VDGLM that was not just due to fitting to autocorrelation.

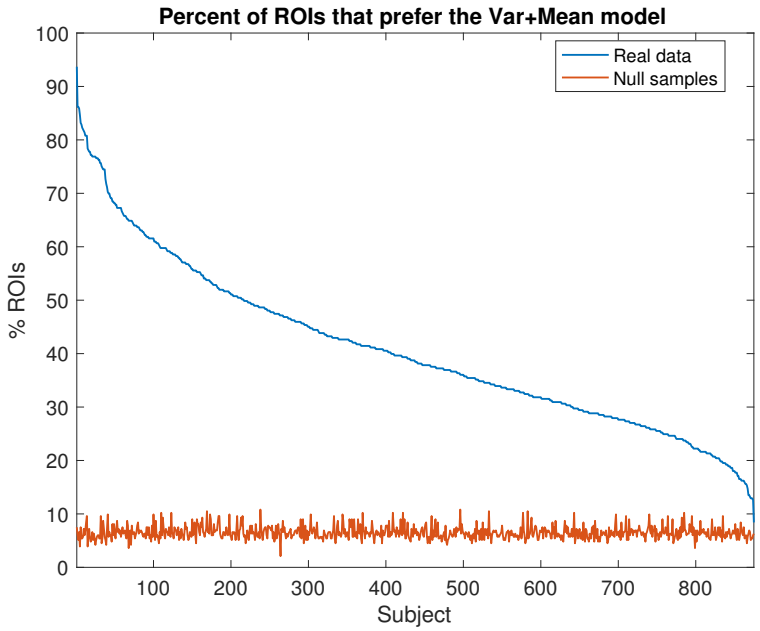

Figure 4.6: The percent of regions for each subject for which the VDGLM model better describes the HCP data (blue) and data simulated from the mean model (red). Subjects are ordered by percent of regions for which the VDGLM model has higher OOSLL.

4.3.3 Application Summary

The working memory application highlights the usage of the VDGLM. We used the model to show that working memory task engagement was related to a decrease in variance compared to fixation. Variance effects and mean effects were not spatially correlated, suggesting that the VDGLM reveals distinct brain patterns not captured by mean-based approaches. We found that the spatial distribution of mean effects was similar to the spatial distribution of

voxel-wise mean effects from previous analyses [4]. Variance effects tended to occur across the whole brain and reduce variance compared to baseline. We want to highlight that while many of the variance effect sizes in this application were fairly small, this will not necessarily be true in future applications. Importantly, if there exists some quantity of interest that consistently relates to small variance effect, then these small effects are worth studying. This is especially true in disease studies where discoveries have the potential to save human lives.

The application focused on testing for effects of BV, so we designed our application to reduce potential confounds. By using the HCP data we minimized the effects of noise from CSF, large veins, and white matter. By collecting data at a high resolution (2mm) and registering data to the cortical surface, the HCP data had less voxel-by-voxel overlap with these noise sources than compared to other data sources [53]. We corrected for head motion by including nuisance motion regressors and scrubbing particularly noisy volumes. We did not account for heart beat nor respiration, which are known to affect resting state BOLD variability [9, 70, 71, 72]. However, neither source of noise could account for the variance effects we demonstrated. Since neither heart rate nor respiration are correlated with the task design, presence of these sources of noise increases variance during task. Thus we suspect that correcting for physiological noise in future studies would lead to larger variance effect sizes. We performed several post-hoc analyses to check that VDGLM preference was not related to to mean frame displacement, nor grand mean intensity scaling factor [135] (adj$R^2 = -0.00114, -0.0011$, respectively).

In this application, we fit a single VDGLM model and a single corresponding GLM model. In practice, we could fit several VDGLM models to test hypotheses of the form: "should condition C be be included in our model and does it affect the mean or the variance in BOLD activation". In this set-up, each model takes the form of eq. 4.4 and the conditions to be

tested are defined by the entries of the mean and variance design matrices. For example, we could test a model with only intercept effects versus a model that allows each condition to affect the variance, but not the mean. Then model comparison indicates which experimental conditions are necessary in the model and whether those conditions are necessary as mean or variance regressors. This formulation allows us to define several nested models, which can be nested in the classical sense–i.e., that the set of mean regressors in the nested model is a subset of the regressors of the full model– or can have nested variance regressors.

To effectively develop and test the VDGLM, we chose an ROI approach to have more reliable BOLD signal and a lower computational load. Application of the VDGLM to voxel-wise analyses will be done in future work.

4.4 Discussion

Traditional fMRI analyses treat BOLD variation as a 'nuisance parameter' despite results linking BOLD variation to age, behavioral performance, and task engagement. The VDGLM fills this gap by providing a flexible framework for linking variance effects to experimental design. By directly incorporating the design matrix, the VDGLM can assess the independent contributions to BOLD variance from multiple experimental conditions while controlling for confounding factors. The VDGLM also controls for confounding between mean and variance effects; since both effects are modeled simultaneously, we can make inferences about one effect while controlling for the other. The VDGLM is fit in a mass univariate approach, which allows analysis at a more fine-grained resolution than previous empirical studies that analyzed latent structures of large spatial patterns in BV. Under the VDGLM framework hypothesis generation and comparison is easy; each hypothesis corresponds to an instantiation of the model and can be tested using model comparison.

In our application, we showed that the VDGLM can be used to find variance effects caused by working memory engagement (Figure 4.3). We showed that these effects are spatially orthogonal to mean effects (Figures 4.4, 4.5) and finally, we compared the GLM and VDGLM and showed that VDGLM provides a better description of the data even while accounting for model complexity (Figure 4.6).

An important feature the VDGLM is the facility for modeling mean and variance simultaneously while allowing for orthogonal spatial inferences. In the BV-age fMRI literature, variance effects were orthogonal to mean effects [46, 47]. This trend generalized to our working memory application, where task engagement resulted in predominantly negative BV effects across the brain, but a mix of positive and negative mean effects. These results constitute a growing body of evidence that BV is a novel dimension for studying brain function.

We want to highlight that there are alternatives to the methodological choices we made in our application. Alternative choices can be made regarding 1) the prewhitening model, 2) the inference statistic or effect size estimate, 3) the model comparison metric and 4) the method for assessing model comparison significance (see Figure 4.2). The prewhitening model (1) and comparison metric (3) can easily be substituted for another model and metric, and the additional model comparison significance test (4), while powerful, is not necessary in most standard analyses. Using an alternative choice of inference statistic (2) may require further work. Our choice to use effects size was motivated by the use of effect size in previous analyses [137] and the ease of using a statistic that depends solely on parameter estimates. In an effort to provide alternative inference statistics, we developed approximate t-tests for variance effects. However, we found that the estimates were sensitive to the condition number of the Hessian matrix specified by the VDGLM (a quantity required for computation of the approximate t-test). We tested the accuracy of the approximate t-tests for mean effects by comparing them to standard t-tests made by the GLM. While we found that while they

were close for most subjects and ROIs, for other subjects with poorly conditioned Hessian matrices the t-tests tended to be unrealistically large. While any individual data point could be excluded from group-level analysis using condition number threshold, we found this approach too cumbersome for a framework aimed at general public use. Development of well-behaved statistics for inference is ongoing work.

Many different measures of BV have been used in past fMRI studies: empirical variance [61], parametric variance [146], block-normalized standard deviation [46, 49], and mean squared successive difference (MSSD) [81, 120]. The goals of mean squared successive difference and block-normalized standard deviation are to measure the variance not accounted for by mean trends in the data. Since the VDGLM models the variance/standard deviation in BOLD activation after accounting for the mean trend, it's variance parameters can be conceptualized to measure a construct similar to mean squared successive difference or block-normalized variance (but where blocks are convolved with the canonical HRF). The parametrized model in Wutte et al. 2011 is similar to the VDGLM, but uses a mixing parameter to capture shared variance between task and fixation blocks rather than modeling the variance as a function of convolved experimental design. We expect that this approach leads to similar results, but with the caveat that it only incorporates a single experimental condition. Lastly, we consider the inter-quartile range, which is not used in the VDGLM and to which to our knowledge has not been used in fMRI analysis to date. The goal of the inter-quartile-range is to summarize the dispersion while limiting the effects of any highly outlying time points. In our application, we used scrubbing to a similar effect by manually removing any outlying time points and recommend this approach if large outliers are present.

The VDGLM could be improved by implementing it in a Bayesian framework. Bayesian frameworks would allow us to make more robust inferences, incorporate prior beliefs about regions likely (or unlikely) to exhibit BV effects, and to better quantity model comparisons.

The main disadvantage of Bayesian methods, and the reason we did not develop a Bayesian VDGLM, is the computational complexity of inference. Initially, we developed a Bayesian VDGLM in STAN, but inference on our toy dataset took days, so we opted to first develop a frequentist version. We leave the development of a Bayesian implementation for future work.

The VDGLM could also be improved by transforming the variance so that we did not need to enforce positivity. Log transformations have been widely used for covariance and variance estimation [109], however in the case of the VDGLM this would lead to a drastic conceptual change in the model. Since the VDGLM incorporates the design matrix into its variance formulation, the exponential transformation results in a variance parameters that are raised to the power of elements of the design matrix. We opted to keep the VDGLM as an additive variance model that requires constraints rather than as a multiplicative variance model so that parameters were more interpretable.

4.5 Conclusion

Studies have demonstrably shown that variance in BOLD activation is a functional construct orthogonal to mean BOLD that should be taken into account in future imaging analyses.

This work developed the VDGLM, a coherent statistical framework for incorporating BV into standard fMRI analyses. The VDGLM was motivated by strong evidence that variance in BOLD activation is linked to individuals and behavior. The VDGLM can be easily applied in any experimental setting and will allow for increased ease and flexibility in research on BOLD variability. We expect that it will lead to exciting new discoveries relating BOLD variability to human characteristics and behavior.

Bibliography

[1] H. Akaike. A new look at the statistical model identification. *IEEE transactions on automatic control*, 19(6):716–723, 1974.

[2] N. Ambady, D. LaPlante, and T. Nguyen. Surgeons' tone of voice: a clue to malpractice history. *Surgery*, 132(1):5–9, 2002.

[3] D. C. Atkins, M. Steyvers, and Z. E. Imel. Scaling up the evaluation of psychotherapy: evaluating motivational interviewing fidelity via statistical text classification. *Implementation Science*, 9(1):49, 2014.

[4] D. M. Barch, G. C. Burgess, M. P. Harms, S. E. Petersen, B. L. Schlaggar, M. Corbetta, M. F. Glasser, S. Curtiss, S. Dixit, C. Feldt, et al. Function in the human connectome: task-fmri and individual differences in behavior. *Neuroimage*, 80:169–189, 2013.

[5] C. F. Beckmann, M. Jenkinson, and S. M. Smith. General multilevel linear modeling for group analysis in fmri. *Neuroimage*, 20(2):1052–1063, 2003.

[6] H. Berger. Über das elektrenkephalogramm des menschen. *Archiv für psychiatrie und nervenkrankheiten*, 87(1):527–570, 1929.

[7] E. Bernhardsson and E. Freider. Luigi [computer software]. https://github.com/spotify/luigi, 2016.

[8] D. Billsus and M. J. Pazzani. A hybrid user model for news story classification. pages 99–108, 1999.

[9] B. Biswal, E. A. Deyoe, and J. S. Hyde. Reduction of physiological fluctuations in fmri using digital filters. *Magnetic Resonance in Medicine*, 35(1):107–113, 1996.

[10] B. Biswal, F. Zerrin Yetkin, V. M. Haughton, and J. S. Hyde. Functional connectivity in the motor cortex of resting human brain using echo-planar mri. *Magnetic resonance in medicine*, 34(4):537–541, 1995.

[11] E. Bullmore, M. Brammer, S. C. Williams, S. Rabe-Hesketh, N. Janot, A. David, J. Mellers, R. Howard, and P. Sham. Statistical methods of estimation and inference for functional mr image analysis. *Magnetic Resonance in Medicine*, 35(2):261–277, 1996.

[12] A. Z. Burzynska, C. N. Wong, M. W. Voss, G. E. Cooke, E. McAuley, and A. F. Kramer. White matter integrity supports bold signal variability and cognitive performance in the aging human brain. *PLoS One*, 10(4):e0120315, 2015.

[13] R. H. Byrd, R. B. Schnabel, and G. A. Shultz. Approximate solution of the trust region problem by minimization over two-dimensional subspaces. *Mathematical programming*, 40(1):247–263, 1988.

[14] W. W. Chapman, P. M. Nadkarni, and L. Hirschman. Overcoming barriers to nlp for clinical text: the role of shared tasks and the need for additional creative solutions. *Journal of the American Medical Informatics Association*, 18(5):540–543, 2011.

[15] T. Y. Chiu, T. Leonard, and K.-W. Tsui. The matrix-logarithmic covariance model. *Journal of the American Statistical Association*, 91(433):198–210, 1996.

[16] A. S. Choe, C. K. Jones, S. E. Joel, J. Muschelli, V. Belegu, B. S. Caffo, M. A. Lindquist, P. C. van Zijl, and J. J. Pekar. Reproducibility and temporal structure in weekly resting-state fmri over a period of 3.5 years. *PloS one*, 10(10):e0140134, 2015.

[17] A. Christensen, D. C. Atkins, and S. Berns. Traditional versus integrative behavioral couple therapy for significantly and chronically distressed married couples. *Journal of Consulting and Clinical Psychology*, 72(2):176, 2004.

[18] J. Cohen. Statistical power analysis for the behavioral sciences (revised ed.), 1977.

[19] M. W. Cole, D. S. Bassett, J. D. Power, T. S. Braver, and S. E. Petersen. Intrinsic and task-evoked network architectures of the human brain. *Neuron*, 83(1):238–251, 2014.

[20] G. V. Cormack. Email spam filtering: A systematic review. *Foundations and Trends in Information Retrieval*, 1(4):335–455, 2007.

[21] R. W. Cox. Afni: software for analysis and visualization of functional magnetic resonance neuroimages. *Computers and Biomedical research*, 29(3):162–173, 1996.

[22] R. C. Craddock, G. A. James, P. E. Holtzheimer, X. P. Hu, and H. S. Mayberg. A whole brain fmri atlas generated via spatially constrained spectral clustering. *Human brain mapping*, 33(8):1914–1928, 2012.

[23] W. A. Cunningham, N. L. Arbuckle, A. Jahn, S. M. Mowrer, and A. M. Abduljalil. Aspects of neuroticism and the amygdala: chronic tuning from motivational styles. *Neuropsychologia*, 48(12):3399–3404, 2010.

[24] R. D. and R. E. Stanford topic modeling toolbox. http://nlp.stanford.edu/software/tmt/tmt-0.4/, 2009.

[25] J. Decety, S. Echols, and J. Correll. The blame game: the effect of responsibility and social stigma on empathy for pain. *Journal of cognitive neuroscience*, 22(5):985–997, 2010.

[26] S. Deerwester, S. T. Dumais, G. W. Furnas, T. K. Landauer, and R. Harshman. Indexing by latent semantic analysis. *Journal of the American society for information science*, 41(6):391, 1990.

[27] W. Dement and N. Kleitman. Cyclic variations in eeg during sleep and their relation to eye movements, body motility, and dreaming. *Electroencephalography and clinical neurophysiology*, 9(4):673–690, 1957.

[28] D. Dodell-Feder, J. Koster-Hale, M. Bedny, and R. Saxe. fmri item analysis in a theory of mind task. *Neuroimage*, 55(2):705–712, 2011.

[29] M. Dowman, V. Savova, T. L. Griffiths, K. P. Kording, J. B. Tenenbaum, and M. Purver. A probabilistic model of meetings that combines words and discourse features. *IEEE Trans Audio Speech Lang Processing*, 16 no. 7:1238–1248, 2008.

[30] F. H. Duffy, M. S. Albert, G. McAnulty, and A. J. Garvey. Age-related differences in brain electrical activity of healthy subjects. *Annals of neurology*, 16(4):430–438, 1984.

[31] S. Dumais and H. Chen. Hierarchical classification of web content. *In Proceedings of the 23rd annual international ACM SIGIR conference on Research and development in information retrieval*, pages 256–263, 2000.

[32] R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, and C.-J. Lin. Liblinear: A library for large linear classification. *Journal of Machine Learning Research*, 9:1871–1874, 2008.

[33] M. D. Feldman, P. Franks, and P. R. Duberstein. Lets not talk about it: suicide inquiry in primary care. *The Annals of Family Medicine*, 5(5):412–418, 2007.

[34] E. Finn, X. Shen, D. Scheinost, M. Rosenberg, J. Huang, M. Chun, X. Papademetris, and R. Constable. Functional connectome fingerprinting: identifying individuals using patterns of brain connectivity. *Nature neuroscience*, 18:1664–1671, 2015.

[35] E. S. Finn and T. R. Constable. Individual variation in functional brain connectivity: implications for personalized approaches to psychiatric disease. *Dialogues in Clinical Neuroscience*, 18(3):277–287, 2016.

[36] J. Friedman, T. Hastie, and R. Tibshirani. Sparse inverse covariance estimation with the graphical lasso. *Biostatistics*, 9(3):432–441, 2008.

[37] K. Friston. Hierarchical models in the brain. *PLoS Comput Biol*, 4(11):e1000211, 2008.

[38] K. Friston, C. Frith, P. Liddle, and R. Frackowiak. Functional connectivity: the principal-component analysis of large (pet) data sets. *Journal of Cerebral Blood Flow & Metabolism*, 13(1):5–14, 1993.

[39] K. Friston, O. Josephs, E. Zarahn, A. Holmes, S. Rouquette, and J.-B. Poline. To smooth or not to smooth?: Bias and efficiency in fmri time-series analysis. *NeuroImage*, 12(2):196–208, 2000.

[40] K. J. Friston. Functional and effective connectivity: a review. *Brain connectivity*, 1(1):13–36, 2011.

[41] K. J. Friston, A. P. Holmes, K. J. Worsley, J.-P. Poline, C. D. Frith, and R. S. Frackowiak. Statistical parametric maps in functional imaging: a general linear approach. *Human brain mapping*, 2(4):189–210, 1994.

[42] K. J. Friston, P. Jezzard, and R. Turner. Analysis of functional mri time-series. *Human brain mapping*, 1(2):153–171, 1994.

[43] K. J. Friston, W. Penny, C. Phillips, S. Kiebel, G. Hinton, and J. Ashburner. Classical and bayesian inference in neuroimaging: theory. *NeuroImage*, 16(2):465–483, 2002.

[44] T. G. and K. I. Multi-label classification: An overview. *International Journal of Data Warehousing and Mining (IJDWM)*, 3.3:1–13, 2007.

[45] T. G. and K. I. Multi-label classification: An overview. *International Journal of Data Warehousing and Mining (IJDWM)*, 3.3:1–13, 2007.

[46] D. D. Garrett, N. Kovacevic, A. R. McIntosh, and C. L. Grady. Blood oxygen level-dependent signal variability is more than just noise. *Journal of Neuroscience*, 30(14):4914–4921, 2010.

[47] D. D. Garrett, N. Kovacevic, A. R. McIntosh, and C. L. Grady. The importance of being variable. *Journal of Neuroscience*, 31(12):4496–4503, 2011.

[48] D. D. Garrett, N. Kovacevic, A. R. McIntosh, and C. L. Grady. The modulation of bold variability between cognitive states varies by age and processing speed. *Cerebral Cortex*, page bhs055, 2012.

[49] D. D. Garrett, N. Kovacevic, A. R. McIntosh, and C. L. Grady. The modulation of bold variability between cognitive states varies by age and processing speed. *Cerebral Cortex*, 23(3):684–693, 2013.

[50] D. D. Garrett, A. R. McIntosh, and C. L. Grady. Brain signal variability is parametrically modifiable. *Cerebral Cortex*, 24(11):2931–2940, 2013.

[51] D. D. Garrett, G. R. Samanez-Larkin, S. W. MacDonald, U. Lindenberger, A. R. McIntosh, and C. L. Grady. Moment-to-moment brain signal variability: A next frontier in human brain mapping? *Neuroscience & Biobehavioral Reviews*, 37(4):610–624, 2013.

[52] G. Gaut. Vdglm [computer software]. https://github.com/https://github.com/geebioso/VDGLM, 2018.

[53] M. F. Glasser, S. N. Sotiropoulos, J. A. Wilson, T. S. Coalson, B. Fischl, J. L. Andersson, J. Xu, S. Jbabdi, M. Webster, J. R. Polimeni, et al. The minimal preprocessing pipelines for the human connectome project. *Neuroimage*, 80:105–124, 2013.

[54] C. E. Golin, H. Liu, and R. D. Hays. A prospective study of predictors of adherence to combination antiretroviral medication. *Journal of General Internal Medicine*, 17(10):756–765, 2002.

[55] J. Gonzalez-Castillo, C. W. Hoy, D. A. Handwerker, M. E. Robinson, L. C. Buchanan, Z. S. Saad, and P. A. Bandettini. Tracking ongoing cognition in individuals using brief, whole-brain functional connectivity patterns. In *Proceedings of the National Academy of Sciences*, pages 8762–8767. 112(28, 2015.

[56] E. M. Gordon, T. O. Laumann, B. Adeyemo, J. F. Huckins, W. M. Kelley, and S. E. Petersen. Generation and evaluation of a cortical area parcellation from resting-state correlations. *Cerebral cortex*, 26(1):288–303, 2014.

[57] C. L. Grady and D. D. Garrett. Understanding variability in the bold signal and why it matters for aging. *Brain imaging and behavior*, 8(2):274–283, 2014.

[58] M. D. Greicius, B. Krasnow, A. L. Reiss, and V. Menon. Functional connectivity in the resting brain: a network analysis of the default mode hypothesis. *Proceedings of the National Academy of Sciences*, 100(1):253–258, 2003.

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

[60] M. Guitart-Masip, A. Salami, D. Garrett, A. Rieckmann, U. Lindenberger, and L. Bckman. Bold variability is related to dopaminergic neurotransmission and cognitive aging. *Cerebral Cortex*, 26(5):2074–2083, 2016.

[61] B. J. He. Scale-free properties of the functional magnetic resonance imaging signal during rest and task. *The Journal of Neuroscience*, 31(39):13786–13795, 2011.

[62] A. E. Hoerl and R. W. Kennard. Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics*, 12(1):55–67, 1970.

[63] A. Holmes and K. Friston. Generalisability, random effects & population inference. *Neuroimage*, 7:S754, 1998.

[64] C. Howes, M. Purver, and R. McCabe. Investigating topic modeling for therapy dialogue analysis. In *Proceedings of IWCS*, 2013.

[65] C. Howes, M. Purver, and R. McCabe. Using conversation topics for predicting therapy outcomes in schizophrenia. *Biomed Inform Insights*, 2013.

[66] C. Howes, M. Purver, R. McCabe, P. G. Healey, and M. Lavelle. Predicting adherence to treatment for schizophrenia from dialogue transcripts. In *Proceedings of the 13th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 79–83. Association for Computational Linguistics, 2012.

[67] Z. E. Imel, S. A. Baldwin, J. S. Baer, B. Hartzler, C. Dunn, D. B. Rosengren, and D. C. Atkins. Evaluating therapist adherence in motivational interviewing by comparing performance with standardized and real patients. *Journal of consulting and clinical psychology*, 82(3):472, 2014.

[68] Z. E. Imel, S. M., , and D. C. Atkins. Computational psychotherapy research: Scaling up the evaluation of patient-provider interactions. *Psychotherapy, Chicago, Ill.*, 2014.

[69] M. Jenkinson and S. Smith. A global optimisation method for robust affine registration of brain images. *Medical image analysis*, 5(2):143–156, 2001.

[70] S. S. Kannurpatti and B. B. Biswal. Detection and scaling of task-induced fmri-bold response using resting state fluctuations. *NeuroImage*, 40:1567–1574, 2008.

[71] S. S. Kannurpatti, M. A. Motes, B. Rypma, and B. B. Biswal. Neural and vascular variability and the fmri-bold response in normal aging. *Magnetic resonance imaging*, 28(4):466–476, 2010.

[72] S. S. Kannurpatti, M. A. Motes, B. Rypma, and B. B. Biswal. Increasing measurement accuracy of age-related bold signal change: Minimizing vascular contributions by resting-state-fluctuation-of-amplitude scaling. *Human brain mapping*, 32(7):1125–1140, 2011.

[73] T. Kaufmann, D. Alnæs, C. L. Brandt, N. T. Doan, K. Kauppi, F. Bettella, T. V. Lagerberg, A. O. Berg, S. Djurovic, I. Agartz, et al. Task modulations and clinical manifestations in the brain functional connectome in 1615 fmri datasets. *NeuroImage*, 2016.

[74] W. Klimesch. Eeg alpha and theta oscillations reflect cognitive and memory performance: a review and analysis. *Brain research reviews*, 29(2-3):169–195, 1999.

[75] W. Klimesch. Alpha-band oscillations, attention, and controlled access to stored information. *Trends in cognitive sciences*, 16(12):606–617, 2012.

[76] W. Klimesch, B. Schack, and P. Sauseng. The functional significance of theta and upper alpha oscillations. *Experimental psychology*, 52(2):99–108, 2005.

[77] B. Knutson, C. M. Adams, G. W. Fong, and D. Hommer. Anticipation of increasing monetary reward selectively recruits nucleus accumbens. *J Neurosci*, 21(16):RC159, 2001.

[78] A. Kucyi, M. J. Hove, M. Esterman, R. M. Hutchison, and E. M. Valera. Dynamic brain network correlates of spontaneous fluctuations in attention. *Cerebral Cortex*, page bhw029, 2016.

[79] T. O. Laumann, E. M. Gordon, B. Adeyemo, A. Z. Snyder, S. J. Joo, M. Y. Chen, and B. L. Schlaggar. Functional system and areal organization of a highly sampled individual human brain. *Neuron*, 87(3):657–670, 2015.

[80] B. Lenoski, L. C. Baxter, L. J. Karam, J. Maisog, and J. Debbins. On the performance of autocorrelation estimation algorithms for fmri analysis. *IEEE Journal of Selected Topics in Signal Processing*, 2(6):828–838, 2008.

[81] A. Leo, G. Bernardi, G. Handjaras, D. Bonino, E. Ricciardi, and P. Pietrini. Increased bold variability in the parietal cortex and enhanced parieto-occipital connectivity during tactile perception in congenitally blind individuals. *Neural plasticity*, 2012, 2012.

[82] Q. Li, K. Melton, and T. Lingren. Phenotyping for patient safety: algorithm development for electronic health record based automated adverse event and medical error detection in neonatal intensive care. *Journal of the American Medical Informatics Association*, 2014.

[83] B. D. M., N. A.Y., and J. M.I. Latent dirichlet allocation. *The Journal of Machine Learning Research*, 3:993–1022, 2003.

[84] C. D. Manning, P. Raghavan, H. Schütze, et al. *Introduction to information retrieval*, volume 1. Cambridge university press Cambridge, 2008.

[85] C. D. Manning, H. Schütze, et al. *Foundations of statistical natural language processing*, volume 999. MIT Press, 1999.

[86] B. J. Marafino, J. M. Davies, and N. S. Bardach. N-gram support vector machines for scalable procedure and diagnosis classification, with applications to clinical free text data from the intensive care unit. *Journal of the American Medical Informatics Association*, 2014.

[87] D. S. Marcus, J. Harwell, T. Olsen, M. Hodge, M. F. Glasser, F. Prior, M. Jenkinson, T. Laumann, S. W. Curtiss, and D. C. Van Essen. Informatics and data mining tools and strategies for the human connectome project. *Frontiers in neuroinformatics*, 5, 2011.

[88] A. Maril, J. S. Simons, J. P. Mitchell, B. L. Schwartz, and D. L. Schacter. Feeling-of-knowing in episodic memory: an event-related fmri study. *Neuroimage*, 18(4):827–836, 2003.

[89] E. Mayfield, M. B. Laws, and I. B. Wilson. Automating annotation of information-giving for analysis of clinical conversation. *Journal of the American Medical Informatics Association*, 21(e1):e122–e128, 2014.

[90] M. Mennes, X.-N. Zuo, C. Kelly, A. Di Martino, Y.-F. Zang, B. Biswal, F. X. Castellanos, and M. P. Milham. Linking inter-individual differences in neural activation and behavior to intrinsic brain dynamics. *Neuroimage*, 54(4):2950–2959, 2011.

[91] Mexico. Constitución política' de los estados unidos mexicanos (artículo 8). http://www.diputados.gob.mx/LeyesBiblio/htm/1.htm, 1917.

[92] N. D. S. Mexico. Mèxico digital. https://www.gob.mx/mexicodigital/, 2015.

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

[94] W. R. Miller and S. Rollnick. *Motivational interviewing: Preparing people for change*. Guilford, New York, 2002.

[95] M. Mitchell, K. Hollingshead, and G. Coppersmith. Quantifying the language of schizophrenia in social media. In *Presented at the Proceedings of CLPsych*, 2015.

[96] J. J. Moré and D. C. Sorensen. Computing a trust region step. *SIAM Journal on Scientific and Statistical Computing*, 4(3):553–572, 1983.

[97] F. Mosteller and J. W. Tukey. Data analysis, including statistics. *Handbook of social psychology*, 2:80–203, 1968.

[98] L. Nadkarni, P. M.and Ohno-Machado and W. W. Chapman. Natural language processing: an introduction. *Journal of the American Medical Informatics Association*, 18(5):544–551, 2011.

[99] A. Navarrette, L. Wahedi, G. Gaut, A. D. Unanùe, E. Potash, and R. Ghani. Improving government response to citizen requests online. https://github.com/dssg/atencion, 2016.

[100] S. G. Office. Surgeon generals report. 1999.

[101] M. Ott. Jgibblabeledlda. https://github.com/myleott/JGibbLabeledLDA, 2013.

[102] A. M. Owen, K. M. McMillan, A. R. Laird, and E. Bullmore. N-back working memory paradigm: A meta-analysis of normative functional neuroimaging studies. *Human brain mapping*, 25(1):46–59, 2005.

[103] B. Pang and L. Lee. Opinion mining and sentiment analysis. *Foundations and trends in information retrieval*, 2(1-2):1–135, 2008.

[104] 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 Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[105] A. Perotte, R. Pivovarov, and K. Natarajan. Diagnosis code assignment: models and evaluation metrics. *Journal of the American Medical Informatics Association*, 21(2):231–237, 2014.

[106] C. Peterson, F. C. Stingo, and M. Vannucci. Bayesian inference of multiple gaussian graphical models. *Journal of the American Statistical Association*, 110(509):159–174, 2015.

[107] R. A. Poldrack, T. O. Laumann, O. Koyejo, B. Gregory, A. Hover, M.-Y. Chen, K. J. Gorgolewski, J. Luci, S. J. Joo, R. L. Boyd, et al. Long-term neural and physiological phenotyping of a single human. *Nature communications*, 6, 2015.

[108] C. Poulin, B. Shiner, and P. Thompson. Predicting the risk of suicide by analyzing the text of clinical notes. *PloS one*, 9(1):e85733, 2014.

[109] M. Pourahmadi. Covariance estimation: The glm and regularization perspectives. *Statistical Science*, pages 369–387, 2011.

[110] M. Purver. *Spoken language understanding: systems for extracting semantic information from speech*. Wiley, 2011.

[111] M. R. J. Purver. *The theory and use of clarification requests in dialogue*. PhD thesis, Dept. Comp. Sci., KCL, London, UK, 2004.

[112] D. Rakel, B. Barrett, Z. Zhang, T. Hoeft, B. Chewning, L. Marchand, and J. Scheder. Perception of empathy in the therapeutic encounter: Effects on the common cold. *Patient Education and Counseling*, 85(3):390–397, 2011.

[113] D. Ramage, D. Hall, and R. Nallapati. Labeled lda: A supervised topic model for credit attribution in multi-labeled corpora. *In Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, 1:248–256, 2009.

[114] J. Read, B. Pfahringer, G. Holmes, and E. Frank. Classifier chains for multi-label classification. *Machine learning*, 85(3):333, 2011.

[115] R. Řehůřek and P. Sojka. Software Framework for Topic Modelling with Large Corpora. In *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*, pages 45–50, Valletta, Malta, May 2010. ELRA. http://is.muni.cz/publication/884893/en.

[116] E. Ricciardi, G. Handjaras, G. Bernardi, P. Pietrini, and M. L. Furey. Cholinergic enhancement reduces functional connectivity and bold variability in visual extrastriate cortex during selective attention. *Neuropharmacology*, 64:305–313, 2013.

[117] J. Richiardi, H. Eryilmaz, S. Schwartz, P. Vuilleumier, and D. Van De Ville. Decoding brain states from fmri connectivity graphs. *Neuroimage*, 56(2):616–626, 2011.

[118] T. N. Rubin, A. Chambers, P. Smyth, and M. Steyvers. Statistical topic models for multi-label document classification. *Machine Learning*, 88(1-2):157–208, 2012.

[119] I. S., H. R., and L. P. Mining clinical text for signals of adverse drug-drug interactions. *Journal of the American Medical Informatics Association*, 21:3353–362, 2014.

[120] G. R. Samanez-Larkin, C. M. Kuhnen, D. J. Yoo, and B. Knutson. Variability in nucleus accumbens activity mediates age-related suboptimal financial risk taking. *Journal of Neuroscience*, 30(4):1426–1434, 2010.

[121] G. Schwarz et al. Estimating the dimension of a model. *The annals of statistics*, 6(2):461–464, 1978.

[122] F. Sebastiani. Machine learning in automated text categorization. *ACM Computing Surveys (CSUR)*, 34(1):1–47, 2002.

[123] F. Sebastiani. Machine learning in automated text categorization. *ACM computing surveys (CSUR)*, 34(1):1–47, 2002.

[124] J. Shi and J. Malik. Normalized cuts and image segmentation. *IEEE Transactions on pattern analysis and machine intelligence*, 22(8):888–905, 2000.

[125] W. R. Shirer, S. Ryali, E. Rykhlevskaia, V. Menon, and M. D. Greicius. Decoding subject-driven cognitive states with whole-brain connectivity patterns. *Cerebral cortex*, 22(1):158–165, 2012.

[126] N. Shuyo. Labeled latent dirichlet allocation. https://github.com/shuyo/iir/blob/master/lda/llda.py, 2010.

[127] D. J. Simmonds, J. J. Pekar, and S. H. Mostofsky. Meta-analysis of go/no-go tasks demonstrating that fmri activation associated with response inhibition is task-dependent. *Neuropsychologia*, 46(1):224–232, 2008.

[128] R. J. Somsen, B. J. van't Klooster, M. W. van der Molen, H. M. van Leeuwen, and R. Licht. Growth spurts in brain maturation during middle childhood as indexed by eeg power spectra. *Biological Psychology*, 44(3):187–209, 1997.

[129] M. Steyvers and T. Griffiths. Probabilistic topic models. *Handbook of latent semantic analysis*, 427(7):424–440, 2007.

[130] E. Tagliazucchi, F. von Wegner, A. Morzelewski, S. Borisov, K. Jahnke, and H. Laufs. Automatic sleep staging using fmri functional connectivity data. *Neuroimage*, 63(1):63–72, 2012.

[131] I. Tavor, O. P. Jones, R. B. Mars, S. M. Smith, T. E. Behrens, and S. Jbabdi. Task-free mri predicts individual differences in brain activity during task performance. *Science*, 352(6282):216–220, 2016.

[132] R. Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society; Series B (Methodological)*, pages 267–288, 1996.

[133] K. Toutanova, D. Klein, C. Manning, and Y. Singer. Feature-rich part-of-speech tagging with a cyclic dependency network. *In Proceedings of HLT-NAACL*, pages 252–259, 2003.

[134] K. Toutanova and C. D. Manning. Enriching the knowledge sources used in a maximum entropy part-of-speech tagger. In *Proceedings of the 2000 Joint SIGDAT conference on Empirical methods in natural language processing and very large corpora: held in conjunction with the 38th Annual Meeting of the Association for Computational Linguistics-Volume 13*, pages 63–70. Association for Computational Linguistics, 2000.

[135] B. O. Turner, B. Lopez, T. Santander, and M. B. Miller. One dataset, many conclusions: Bold variabilitys complicated relationships with age and motion artifacts. *Brain imaging and behavior*, 9(1):115–127, 2015.

[136] M. P. Van Den Heuvel and H. E. H. Pol. Exploring the brain network: a review on resting-state fmri functional connectivity. *European Neuropsychopharmacology*, 20(8):519–534, 2010.

[137] D. C. Van Essen, S. M. Smith, D. M. Barch, T. E. Behrens, E. Yacoub, K. Ugurbil, W.-M. H. Consortium, et al. The wu-minn human connectome project: an overview. *Neuroimage*, 80:62–79, 2013.

[138] G. Walker. On periodicity in series of related terms. *Proceedings of the royal society of London*, 131(818):518–532, 1931.

[139] H. Wang et al. Bayesian graphical lasso models and efficient posterior computation. *Bayesian Analysis*, 7(4):867–886, 2012.

[140] H. Wang, S. Z. Li, et al. Efficient gaussian graphical model determination under g-wishart prior distributions. *Electronic Journal of Statistics*, 6:168–198, 2012.

[141] C. A. Webb, R. J. DeRubeis, and J. P. Barber. Therapist adherence/competence and treatment outcome: A meta-analytic review., 2010.

[142] S. Williamson, L. Kaufman, Z.-L. Lu, J.-Z. Wang, and D. Karron. Study of human occipital alpha rhythm: the alphon hypothesis and alpha suppression. *International journal of psychophysiology*, 26(1-3):63–76, 1997.

[143] M. W. Woolrich, T. E. Behrens, C. F. Beckmann, M. Jenkinson, and S. M. Smith. Multilevel linear modelling for fmri group analysis using bayesian inference. *Neuroimage*, 21(4):1732–1747, 2004.

[144] M. W. Woolrich, B. D. Ripley, M. Brady, and S. M. Smith. Temporal autocorrelation in univariate linear modeling of fmri data. *Neuroimage*, 14(6):1370–1386, 2001.

[145] K. J. Worsley, C. Liao, J. Aston, V. Petre, G. Duncan, F. Morales, and A. Evans. A general statistical analysis for fmri data. *Neuroimage*, 15(1):1–15, 2002.

[146] M. G. Wutte, M. T. Smith, V. L. Flanagin, and T. Wolbers. Physiological signal variability in hmt+ reflects performance on a direction discrimination task. *Frontiers in psychology*, 2:185, 2011.

[147] H. Xie, V. Calhoun, J. Gonzalez-Castillo, E. Damaraju, R. Miller, P. Bandettini, and S. Mitra. Whole-brain connectivity dynamics reflect both task-specific and individual-specific modulation: a multitask study. *NeuroImage*, 2017.

[148] G. Xue, Q. Dong, Z. Jin, and C. Chen. Mapping of verbal working memory in nonfluent chinese–english bilinguals with functional mri. *Neuroimage*, 22(1):1–10, 2004.

[149] T. Yarkoni and J. Westfall. Choosing prediction over explanation in psychology: Lessons from machine learning. *Unpublished manuscript. Retrieved from http://jakewestfall. org/publications/Yarkoni_Westfall_choosing_prediction. pdf*, 2016.

[150] Y. Ye, F. R. Tsui, and M. Wagner. Influenza detection from emergency department reports using natural language processing and bayesian network classifiers. *Journal of the American Medical Informatics Association*, 2014.

[151] Y.-x. Yuan. A review of trust region algorithms for optimization. In *ICIAM*, volume 99, pages 271–282, 2000.

[152] G. U. Yule et al. Vii. on a method of investigating periodicities disturbed series, with special reference to wolfer's sunspot numbers. *Phil. Trans. R. Soc. Lond. A*, 226(636-646):267–298, 1927.

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

Appendix A

Appendix

A.1 Codes Used in Talk-Turn Prediction

Several of the symptoms for which we performed additional local coding are closely associated with more than a single code in the psychotherapy corpus (e.g., the anger symptom is closely associated with anger and frustration). The human raters who judged the representativeness of the 5 symptoms were unaware of the variety of content codes used in the psychotherapy corpus and therefore, the rater's concept of suicide might not map onto the (narrow) concept of suicide in the psychotherapy corpus. We therefore had a clinical psychologist create associated code sets by selecting from the list of psychotherapy symptom codes (See Table A.1) with the constraint that the code set contain the matching code term. These meta code sets were created prior to any evaluation of the model.

For each of the 5 symptoms in the ratings experiment, we take the set of codes from the psychotherapy corpus that are closely associated with the symptom (e.g. for the anger suicide, we take the set anger and frustration) and average the model predictions across the codes

Table A.1: Symptom codes and sets of associated codes

Symptom Code	**Code Set**
anger	anger, frustration
anxiety	anxiety, fear, nervousness, social anxiety, stress, death anxiety, fearfulness, panic, paranoia, restlessness
depression	depression, grief, guilt, hopelessness, loneliness, shame, crying, depressive disorder, despair, dysphoria, loss of appetite, problems concentrating, sadness, suicidal behavior, withdrawn
low self-esteem	low self-esteem, self-esteem
suicidal behavior	hospitalization, suicide, cutting, dysphoria, death

in the set. This creates a model representativeness score for each talk-turn in the ratings experiment that can be compared to the binarized human ratings (highly representative/not highly representative). This approach of combining predictions among closely-related labels can be viewed as a simple implementation of the idea that labels in multi-label document classification are often dependent and leveraging such dependencies is worthwhile and can improve predictive performance [44].

A.2 Supplementary Files: Semi-Automated Content Coding of Psychotherapy Transcripts Using Labeled Topic Models

A.2.1 Session-Level R-precision

Figure A.1 shows the R-precision values calculated for each label for both Labeled Latent Dirichlet Allocation (L-LDA) and lasso logistic regression (LLR).

The R-precision is the precision calculated at the threshold at which a classifier could score

a precision of 1. For example, if we are trying to classify 100 sessions of which 10 have the label, we choose the threshold to separate the top 10 and bottom 90 sessions by classifier probability of being annotated with that label. Then we calculate the precision. The R-precision avoids reporting precisions at each threshold and also has the nice property of equalling the recall for the chosen threshold. We found that R-precision and AUC were fairly correlated across labels for both models (Pearson's coefficient = 0.64 for L-LDA and 0.62 for LLR).

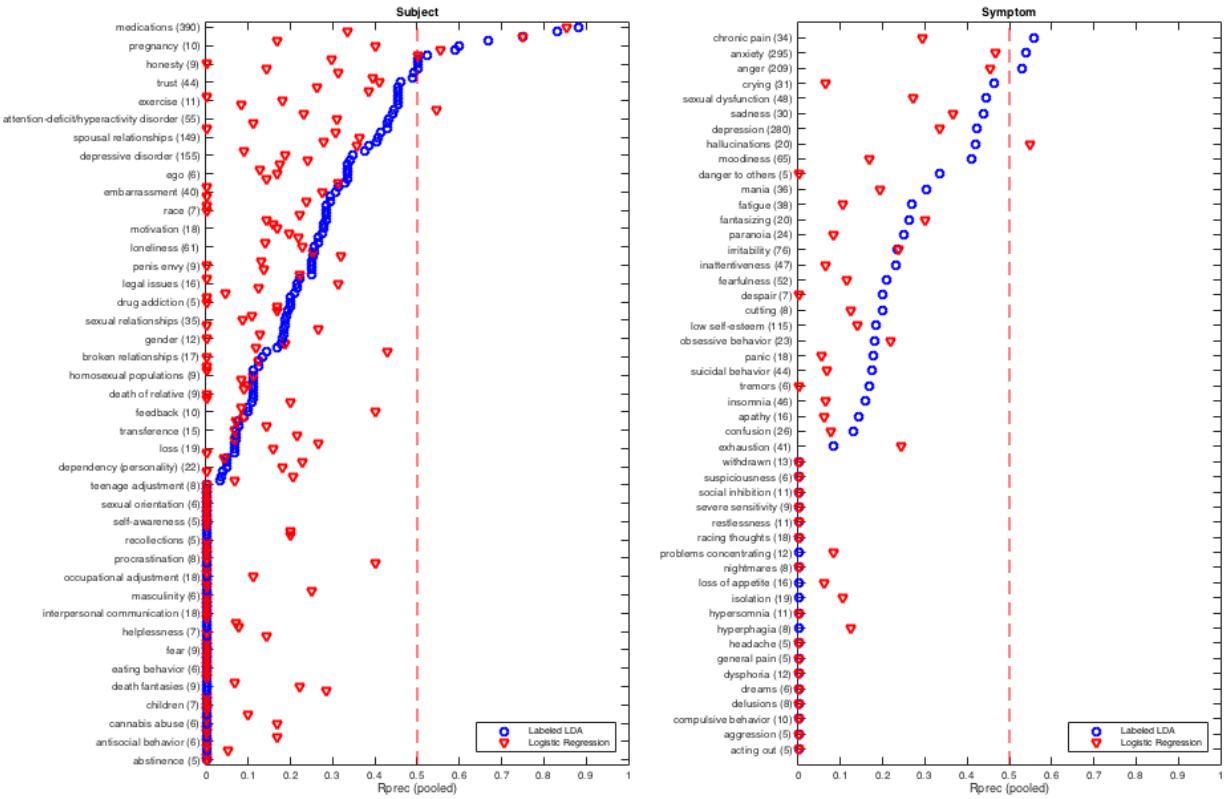

Figure A.1: Session-level R-precision scores for the labeled topic model, the lasso logistic regression model, and chance performance. Codes are reported along the y axis and are ordered by labeled topic model performance. For subject codes, one in every 4 codes names is shown. The number in parenthesis indicates the number of sessions to which a particular code was assigned out of the 1181 possible sessions.

A.2.2 Baseline Model: Lasso Logistic Regression Prediction

We compare L-LDA to a baseline discriminative model, lasso logistic regression [132]. Lasso logistic regression is a regularized logistic regression that penalizes feature weights using the L1 norm and a parameter λ that controls the strength of regularization. We use term counts as features to classify each document by label, using the same vocabulary of unigrams, bigrams, and trigrams used in the L-LDA experiments.

For each code y, let y_d be a binary indicator of code attachment for document d and p_d be the estimated probability that code y is attached to document d. Then the model is fit by minimizing the log loss plus the L1 penalty term:

$$\min_{\beta} -\frac{1}{D}\sum_{d=1}^{D}\left(y_d \log(p_d) + (1-y_d)\log(1-p_d)\right) + \lambda \sum_{i=0}^{V} |\beta_i|)$$

where β_0 is an intercept, β is a V-dimensional vector of feature weights. We calculate p_d as a logistic transformation of the linear combination of feature weights and word counts.

We trained the logistic regression model separately for the task of coding at the session level and the task of finding representative talk-turns.

For session level-prediction, we train a separate binary model for each code using session-wide counts as features. The models are trained and evaluated using the same 10-fold cross-validation partition as we used when evaluating L-LDA performance. The goal for each binary model is to predict the presence/absence of a particular code for the session using the counts observed at the session level as input. For each new session in the validation set, we compute the probability of a given code and compute AUC scores for the logistic model by combining the model predictions across validation sets.

For talk-turn level-prediction, we train a logistic regression model at the talk-turn level using the same 10-way train-test procedure as used for the L-LDA model with talk-turns. For each of the five symptom codes in the local tagging experiment, we train a separate logistic regression model. For each talk-turn in the training set, the logistic regression model is trained to predict the presence or absence of the session-level code associated with the talk-turn. For the 993 talk-turns in the local tagging experiment (that are always part of the validation set), the logistic regression model computes the probability of a code using as input the counts at talk-turn level. The AUC scores are then computed in a manner equivalent to the L-LDA model.

We ran 8 logistic regression models using 8 different regularization parameters λ drawn from the set $.0001, .001, .01, 1, 10, 100, 1000$. For each of the models, we computed the talk-turn level AUC. Average AUC varied smoothly across the set of regularization parameters and we report the model with highest average AUC.

A.3 Task Descriptions

A.4 VDGLM Optimization

We perform maximum likelihood estimation using Trust-region optimization (TRO). TRO is an iterative procedure for minimization. At each iteration, TRO locally approximates the negative log likelihood function using a Taylor expansion and finds a minimum within that step's trust-region, i.e. the region for which the local approximation accurately approximates the objective function. MATLAB restricts the local approximation to a 2-dimensional subspace to allow for faster convergence. The algorithm locally minimizes along the two-

Table A.2: Tasks and descriptions. Underlined tasks are those for which we can compute behavioral performance.

Task	**Description**
Emotional Pictures	Subjects see photographs of the screen, one at a time. These photographs appear to the left or right of the center of the screen. The task is to indicate whether the picture is shifted to the left or right relative to green dot in the center of the screen.
Emotional Faces	Subjects are presented with male and female faces, one at a time. The task is to determine whether the faces are male or female. There are task conditions for neutral, happy, sad, and fearful faces.
Episodic Memory Encoding	Subjects see name and face pairings on a screen. The task is to decide whether the name goes well with the face on a 1-4 (poor to well) scale. There are 4 face conditions: young and old faces that are novel or have been repeated during the experiment.
Episodic Memory Retrieval	Subjects are asked to remember which names were paired with which faces from the episodic memory encoding task. The task is to indicate whether the face name pairs are the same from the previous task, completely novel, or if the face is repeated, but was not paired with the given name.
Go/No-go	Subjects are presented with images of single letters. The task is to press a button when the letter is in the set $\{A, B, C, D, E\}$ and not to press the button when the letter is in the set $\{X, Y, Z\}$.
Monetary Incentive Delay	Subjects are asked to press a button as quickly as possible when a white square (cue) appears on the screen. Participants either win or lose money based on when and how fast they push the button.
Working Memory	Subjects are presented with a sequence of letters and switch between the control task and the 2-back memory task. In the control task, subjects are asked to indicate whether the current letter is underlined. In the memory task, subjects are asked to indicate whether the current letter is the same as the one that was presented two letters ago.
Theory of Mind	Subjects are presented with stories and true or false statements about the stories. The task is to indicate whether the statement was true or false.
Resting State	Subjects are asked to close eyes, relax but stay awake.

dimensional subspace spanned by the direction of steepest descent and one of either a) the approximate newton direction, if it exists, or b) the direction of negative curvature [13]. For a time series y from a single subject and region, we minimize the negative log likelihood $-\log p(y|\theta)$ where the likelihood is defined:

$$p(y, \theta) = \Big[\prod_{t=1}^{T} p(y_t|\beta, v, X_t^M, X_t^V)\Big]$$

where X_t^M is a $[1 \times p_M]$ vector: the single row of the mean design matrix at time t. X_t^V is a $[1 \times p_V]$ vector: the single row of the variance design matrix at time t. For single point in the time series, the likelihood is

$$p(y_t|\beta, v, X_t^M, X_t^V) = \frac{1}{\sqrt{2\pi X_t^V v}} \exp\Big(\frac{-1}{2X_t^V v}(y_t - X_t^M \beta)^2\Big).$$

and we can compute the joint log-likelihood as the product of the log-likelihoods for each point:

$$\log p(y|(\theta) = -\frac{T}{2}\log 2\pi + \Big[\sum_t^T -\frac{1}{2}\log(X_t^V v) - \frac{1}{2X_t^V v}(y_t - (X_t^M \beta))^2\Big]$$

subject to the inequality constraint that the variance is nonzero, i.e.:

$$X_t^V v > 0 \; \forall \; t \in \{1, ..., T\}$$

This constraint can be conceptualized in a Bayesian setting as a uniform prior over the constrained area. Each iteration of the trust-region algorithm uses a Newton-Raphson step to update. We supply the analytical gradients:

$$\frac{\partial \log p(\beta|\bullet)}{\partial \beta} = (X_t^M)^T \Big[\sum_{t=1}^{T} \frac{(y_t - X_t^M \beta)}{X_t^V v} \Big]$$

$$\frac{\partial \log p(v|\bullet)}{\partial v} = (X_t^V)^T \Big[\sum_{t=1}^{T} \frac{-1}{2X_t^V v} + \frac{(y_t - X_t^M \beta)^2}{2(X_t^V v)^2} \Big].$$

We used built-in MATLAB functions to perform optimization. We stop the optimization routine when the magnitude of the gradient is smaller than 1e-6, the change in objective value is smaller than 1e-6, the size of the trust region is below 1e-6, or the optimization routine reaches 1000 iterations.

We compute the Hessian matrix $H = \sum_t^T H_t$ for use in the Wald statistic, where

$$H_t = \begin{bmatrix} [3]\frac{-1}{Z_t}(X_t^M)^T X_t^M & \frac{-Q_t}{Z_t^2}(X_t^V)^T X_t^M \\ \frac{-Q_t}{Z_t^2} X_t^V (X_t^M)^T & \Big[\frac{1}{2Z_t^2} - \frac{Q_t^2}{Z_t^3}\Big](X_t^V)^T X_t^V \end{bmatrix},$$

$Q_t = y_t - X_t^M \beta$, and $Z_t = X_t^V v$.

A.4.1 VDGLM Exponential Transformation Optimization

We tried using an exponential transformation of the variance [15] to see if an unconstrained variance showed stronger results. To fit our model on the full dataset, we find the maximum likelihood estimate using trust region optimization. The objective function that we minimize

is $-\log p(y|\theta)$ where

$$p(y,\theta) = \Big[\prod_{t=1}^{T} p(y_t|\beta, v, X_t^M, X_t^V)\Big]$$

and

$$p(y_t|\beta, v, X_t^M, X_t^V) = \frac{1}{\sqrt{2\pi\exp(X_t^V v)}} \exp\Big(\frac{-1}{2\exp(X_t^V v)}(y_t - X_t^M\beta)^2\Big).$$

The joint log likelihood is:

$$\log p(y|(\theta) = -\frac{T}{2}\log 2\pi + \Big[\sum_{t}^{T} -\frac{1}{2}X_t^V v - \frac{1}{2\exp(X_t^V v)}(y_t - (X_t^M\beta))^2\Big].$$

The gradient is:

$$\frac{\partial \log p(\beta|\bullet)}{\partial\beta} = (X_t^M)^T\Big[\sum_{t=1}^{T}\frac{(y_t - X_t^M\beta)}{\exp(X_t^V v)}\Big]$$

$$\frac{\partial \log p(v|\bullet)}{\partial v} = (X_t^V)^T \Big[\sum_{t=1}^{T} \frac{-1}{2} + \frac{(y_t - X_t^M \beta)^2}{2 \exp(X_t^V v)} \Big]$$

,

and the Hessian matrix is $H = \sum_t^T H_t$, where

$$H_t = \begin{bmatrix} [3]\frac{-1}{Z_t}(X_t^M)^T X_t^M & \frac{-Q_t}{Z_t}(X_t^V)^T X_t^M \\ \frac{-Q_t}{Z_t} X_t^V (X_t^M)^T & \frac{Q_t^2}{2Z_t}(X_t^V)^T X_t^V \end{bmatrix}.$$

We define the variables $Q_t = y_t - X_t^M \beta$ and $Z_t = \exp(X_t^V v)$ to simplify computation.