The Dissertation Committee for Shalmali Dilip Joshi
certifies that this is the approved version of the following dissertation:

Constraint Based Approaches to Interpretable and Semi–supervised Machine Learning

Committee:

Joydeep Ghosh, Supervisor

Oluwasanmi Koyejo

Sujay Sanghavi

Haris Vikalo

Raymond Mooney

Constraint Based Approaches to Interpretable and Semi–supervised Machine Learning

by

Shalmali Dilip Joshi

DISSERTATION

Presented to the Faculty of the Graduate School of

The University of Texas at Austin

in Partial Fulfillment

of the Requirements

for the Degree of

DOCTOR OF PHILOSOPHY

THE UNIVERSITY OF TEXAS AT AUSTIN

December 2018

Dedicated to my parents Dilip and Aruna Joshi
and my little sister Chaitali Joshi

Acknowledgments

This dissertation has come to fruition due of the support of my advisor, collaborators, peers, mentors, family, and friends. I have been immensely grateful to have Dr. Joydeep Ghosh as my academic adviser. He always encouraged me to pursue independent research ideas, explore diverse subjects, while at the same time, guiding me to ensure constant progress. His visionary style of thinking, and formulating problems, has influenced my development as a researcher while pursuing my PhD. He has been ever so approachable, not only in terms advising on the dissertation but also in making the right career decisions. I am thankful for his guidance and constant support.

I have had the privilege of collaborating with some of the best researchers in the field during my PhD and I am grateful for their investment in this endeavor. I would first and foremost like to thank Dr. Oluwasanmi Koyejo for being a supportive and an ever enthusiastic research collaborator and mentor. His drive in pushing boundaries of research has motivated me explore ideas that have culminated in parts of this dissertation. His prolific research contributions, and mentorship is irreplaceable and I am grateful for his inclination to invest his time and effort in mentoring me. I have also had the privilege of working with David Sontag on one of the most fulfilling and rewarding contributions I made to this dissertation. His contributions to the field in general continues to be a source of inspiration for me. I have recently had the privilege of collaborating with Been Kim on an impactful project and her breadth of knowledge in the field made the experience an intellectually

satisfying journey. Finally, I would like to thank Dr. Liangjie Hong, my mentor during an internship at Yahoo! Labs for being extremely supportive, approachable, during this time and beyond. I thank Kristine Resurreccion, Dr. Saul Blecker, and Dr. Stephanie Kreml for validating clinical results of the algorithms proposed in part in this dissertation. I would like to thank Melanie Gulick, Karen, Apipol, Jaymie, and Melody Singleton for helping me navigate the graduate school logistics seamlessly.

I thank all IDEA lab mates, for making this journey even more exciting and gratifying. Avro, Suriya, and Jette have been immensely supportive throughout my time here and I have formed a lasting friendship with them. Joyce and Rajiv have been my go-to labmates to ask for guidance. Sreangsu, Yubin, and Ayan have always been generous with advice. I have had the most fun times with Michael, Taewan, Woody, Diego, Dany, Alan, Farzan. I am thankful to Preeti and Megha for keeping company for friendly banter.

I have had the privilege to form some of the closest friendships during my PhD. I would like to thank Vatsal, Shreya, Tejas, Stavana and Jenny for sharing the ups and downs of graduate life. Aditya, Deepti, and Pradeep made Austin feel like home all these years. Madhura, Harshit, Prasanna, and Bharath never forgot to check-in with me. I cannot thank Caitlin and Murat enough for making the last leg of my PhD extremely exciting and joyous.

I have had constant support from my extended family, cousins, and grandparents. Finally, I cannot possibly thank Aai, Baba, and my sister Chaitali enough for their unconditional support and love. It is their encouragement and confidence that allowed me to set into and complete this journey.

Constraint Based Approaches to Interpretable and Semi–supervised Machine Learning

Publication No. _____

Shalmali Dilip Joshi, Ph.D.
The University of Texas at Austin, 2018

Supervisor: Joydeep Ghosh

Interpretability and *Explainability* of machine learning algorithms are becoming increasingly important as Machine Learning (ML) systems get widely applied to domains like clinical healthcare, social media and governance. A related major challenge in deploying ML systems pertains to reliable learning when expert annotation is severely limited. This dissertation prescribes a common framework to address these challenges, based on the use of constraints that can make an ML model more interpretable, lead to novel methods for explaining ML models, or help to learn reliably with limited supervision.

In particular, we focus on the class of latent variable models and develop a general learning framework by constraining *realizations* of latent variables and/or model parameters. We propose specific constraints that can be used to develop *identifiable* latent variable models, that in turn learn interpretable outcomes. The proposed framework is first used in Non–negative Matrix Factorization and Probabilistic Graphical Models. For both models, algorithms

are proposed to incorporate such constraints with seamless and tractable augmentation of the associated learning and inference procedures. The utility of the proposed methods is demonstrated for our working application domain – *identifiable* phenotyping using Electronic Health Records (EHRs). Evaluation by domain experts reveals that the proposed models are indeed more clinically relevant (and hence more interpretable) than existing counterparts. The work also demonstrates that while there may be inherent trade–offs between constraining models to encourage interpretability, the quantitative performance of downstream tasks remains competitive.

We then focus on constraint based mechanisms to explain decisions or outcomes of supervised black-box models. We propose an explanation model based on generating examples where the nature of the examples is constrained i.e. they have to be sampled from the underlying data domain. To do so, we train a generative model to characterize the data manifold in a high dimensional ambient space. Constrained sampling then allows us to generate naturalistic examples that lie along the data manifold. We propose ways to summarize model behavior using such constrained examples.

In the last part of the contributions, we argue that heterogeneity of data sources is useful in situations where very little to no supervision is available. This thesis leverages such heterogeneity (via constraints) for two critical but widely different machine learning algorithms. In each case, a novel algorithm in the sub-class of *co–regularization* is developed to combine information from heterogeneous sources. Co–regularization is a framework of constraining latent variables and/or latent distributions in order to leverage heterogeneity. The proposed algorithms are utilized for clustering, where the intent is to generate a partition or grouping of observed samples, and for Learning to Rank algorithms

– used to rank a set of observed samples in order of preference with respect to a specific search query. The proposed methods are evaluated on clustering web documents, social network users, and information retrieval applications for ranking search queries.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

With wider applications of machine learning in e–commerce, web search, clinical healthcare, criminal justice platforms and systems, a few critical practical challenges have come to the fore. These challenges remain a primary reason for a lack of trust as well as a hindrance to wider acceptance of machine learning based algorithms in practice. We discuss three of these challenges in the following and discuss how mechanisms formulated in this dissertation can be used to address them.

1.1 Interpretable machine learning

For application domains like clinical healthcare and criminal justice systems, *interpretability* of machine learning algorithms is critical. Interpretability refers to designing learning and inference mechanisms to generate outcomes that are understandable to human or domain experts and are at acceptable levels of abstractions.

1.2 Explainable machine learning

Certain state of the art machine learning models, like deep learning methods, involve learning millions of parameters. Understanding such complex models requires a sophisticated understanding of model behavior and is

therefore inaccessible to a consumer of the model. Such models have become black-box models. It is therefore necessary to develop equally sophisticated mechanisms that probe such models to understand and summarize their behavior using abstractions that are accessible to non–experts.

1.3 Semisupervised machine learning

Typically, supervised learning algorithms learn a functional mapping from attributes over samples to a known target score or categorical label. Collecting the target score/labels is called annotation. This requires expensive human as well as engineering resources for large datasets. Semisupervised learning refers to learning reliably in the absence of reliable and/or limited expert annotation. For instance, it is desirable to rank patients at a caregiving facility in order of their risk of outcome like mortality. It is increasingly difficult to get such subjective scores as it requires expensive clinical expertise. However, heterogeneous sources of information are sometimes available to describe a single data point. For instance, patients in a hospital can be described by their prescription information, claim information, as well as lab test information.

1.4 A Constraint Based Framework

This dissertation focuses on developing a common framework to build interpretable, explainable models as well as to learn reliably when little to no annotation is available. We rely on a framework that assumes such observations can be represented in an unobserved low–dimensional space. This space is called the latent space and the corresponding class of models are known

as latent variable models. Characterizing such a latent space, the distributions over the latent space, as well as the realizations corresponding to the observed samples are the main tasks of an associated learning algorithm. We demonstrate that constraining different aspects of the latent space allows one to (i) encourage models to satisfy specific interpretability criterion (ii) probe complex black-box models to summarize model behavior, and (iii) leverage heterogeneous data sources in lieu of expert annotation to learn reliably at scale.

The first part of the dissertation demonstrates a framework that constrains realizations of latent variables. The corresponding learning algorithms are augmented to impose these constraints during training. Tractable approximations are used when exact imposition of constraints is infeasible. We demonstrate the utility of our mechanism on our working application – automated phenotyping of chronic conditions from Electronic Health Records (EHRs). We rely on two existing latent variable frameworks, namely, probabilistic graphical models (Wainwright and Jordan, 2008) and non–negative matrix factorization (Lee and Seung, 1999) to demonstrate our constraining formulation. Next, we demonstrate how a constrained generative model can be used to probe complex supervised black-box models to generate explanations or summaries of model behavior. Finally, we demonstrate how distributions over latent variables can be constrained as a means to leverage heterogeneity of data sources. This mechanism allows us to leverage multi–modal data sources to learn without expensive supervision. This is applied using *co–regularization* for clustering as well as for learning to rank (LeTOR) (Trotman, 2005) algorithms.

This dissertation is organized as follows. Chapter 2 defines necessary

mathematical constructs used to propose the constraining mechanisms. In particular, the latent variable models used to demonstrate the explainability mechanisms are variational auto–encoders (VAEs) (Kingma and Welling, 2013), probabilistic graphical models (Koller and Friedman, 2009), and latent factor models, specifically non–negative matrix factorization (NMF) (Lee and Seung, 2001). These modeling paradigms are introduced in detail. We introduce a latent variable paradigm for clustering, called mixture models. Finally, a learning to rank (LeTOR) framework based on Generalized Linear Models (McCullagh, 1984) is introduced. All models and associated training and inference mechanisms are substantially generalized in the following chapters to incorporate appropriate constraints.

Chapter 3 details the paradigm of learning interpretable latent variable models using constraints. To do so, we first review existing literature toward developing interpretable and explainable machine learning. We contextualize the proposed formulation's relevance to existing literature on interpretability and explainability models. The chapter focuses on introducing conditions that could be imposed on a machine learning model to encourage it to be inherently *interpertable* given the application domain and an abstraction level. Next, the general mechanism of constraining individual realizations of latent variables is described for latent space models (to specifically encourage interpretability), including augmenting the learning and inference mechanisms. Finally, we propose an instance of such constraints (called *grounding*) that are relevant to our application for phenotyping chronic conditions using EHR data. We discuss some inherent trade-offs of these constraints in terms of model performance and/or interpretability.

Chapter 4 defines the application task of generating interpretable phe-

notypes for chronic conditions of an ICU population using Electronic Health Records. This is our working example to demonstrate the utility of our constraining formulation for enhancing interpretability of ML models. We review existing phenotyping algorithms, and discuss how the proposed formulation addresses existing issues of identifiability and interpretability for phenotyping. We further discuss how EHRs, specifically clinical notes are pre–processed to generate observations as well as weak diagnoses required to impose *grounding* constraints discussed in Chapter 3.

Chapter 5 demonstrates how the *grounding* framework is applied to – (i) a NMF framework, and (ii) admixture of PMRFs (Inouye et al., 2014a) . A new algorithm for learning as well as inference is proposed in each case drawing on *Maximum-a-Posteriori* estimator, and an Alternating-Minimization framework (Koren et al., 2009). The proposed models are evaluated qualitatively and quantitatively. Qualitatively, domain experts (clinicians) were asked to evaluate the quality of the learned phenotype representations based on their relevance of the target conditions as well as their discriminative ability relative to well known baselines. Finally, the phenotype representations are evaluated for their predictive power in determining patient outcomes (30-day mortality outcomes) as a quantitative evaluation of their utility on a down-stream evaluation. We conclude this chapter by discussing some limitations of learning phenotypes without accounting for interventional information.

Chapter 6 proposes xGEMs or *manifold constrained exemplars*, a framework to understand black–box classifier behavior by exploring the landscape of the underlying data manifold as data points cross decision boundaries. To do so, we train an unsupervised generative model – treated as a proxy to the data manifold. We summarize black-box model behavior quantitatively by

generating perturbations of existing samples constrained along the data manifold. Constraining these perturbations requires restricting the latent variables by transforming them using the generative function mapping. We demonstrate xGEMs' ability to detect and quantify observed attribute confounding in model learning and also for understanding the changes in model behavior as training progresses.

Chapter 7 is devoted to leveraging heterogeneous data sources using constraints, in order to effectively learn in the absence of annotation. We do so by effectively constraining distributions over latent spaces and/or latent variables themselves. The first part of the chapter develops a latent variable framework specifically for clustering. In particular, we demonstrate that an effective choice of divergence function to constrain the distributions over the latent variables across heterogeneous data sources can help to learn a partitioning even when the individual data sources may be slightly biased w.r.t. the true clustering distribution. Two algorithms are proposed utilizing this choice of divergence based on variational inference (Wainwright and Jordan, 2008) for estimation. The proposed algorithms have been extensively evaluated on clustering document and social network data. The latter half of Chapter 7 proposes to use heterogeneous sources to learn reliable ranking models when rank ordering is only available for a very few queries. This requires us to substantially generalize an existing listwise ranking framework known as Monotone Retargeting (Acharyya et al., 2012; Acharyya and Ghosh, 2014). We develop novel constraints to enforce agreement across rank–orderings generated by heterogeneous data sources, specifically those of unlabeled queries. Consequently, this ranking framework is evaluated on semi–supervised LeTOR tasks for information retrieval applications.

We conclude in Chapter 8 with some directions for future work focusing on incorporating structural and domain constraints that inform causal influences for generating interpretable and explainable models.

Chapter 2

Background

We first introduce necessary notation that will be used throughout this dissertation.

2.1 Notation

A vector of dimension d is denoted by $\mathbf{x} \in \mathbb{R}^d$. A matrix of dimensions $d \times k$ is denoted by a bold caps letter, e.g. $\mathbf{X} \in \mathbb{R}^{d \times k}$. The space of non-negative reals is denoted by $\mathbb{R}_+^{d \times k}$. $\mathbf{x}_j$ is the j^{th} column of matrix $\mathbf{X}$ while $\mathbf{x}^{(i)}$ denotes the i^{th} row of matrix $\mathbf{X}$. x_{ij} is the entry in the i^{th} row and the j^{th} column of $\mathbf{X}$.

The set of indices $\{1, 2, \cdots, m\}$ is denoted by $[m]$. Δ^{d-1} is the Simplex in dimension d, $\Delta^{d-1} = \{x \in \mathbb{R}_+^d : \sum x_i = 1\}$. Similarly, a λ–Δ^{d-1} (called the lambda-Simplex) in dimension d is the set λ–$\Delta^{d-1} = \{x \in \mathbb{R}_+^d : \sum x_i = \lambda\}$. $\text{supp}(\mathbf{x})$ is the support of vector $\mathbf{x}$. That is, $\text{supp}(\mathbf{x}) = \{i : x_i \neq 0\}$. A generic set is denoted by sans–serif letter C. The set of all vectors isotonic to a vector $\mathbf{y}$ is denoted by $\mathcal{R}_{\downarrow_{\mathbf{y}}}$, i.e. it denotes the set of all vectors that result in the same rank order as $\mathbf{y}$.

We focus on the class of latent variable and latent factor models in order to demonstrate the utility of our constrained based algorithms for interpretability, explainability, and semi–supervised learning. The following builds

the necessary background to formulate our models.

2.2 Non-Negative Matrix Factorization

Non–negative matrix factorization (NMF) is a latent factor model. NMF approximates observational data (represented in non–negative matrix form) as a factorization of two low–rank non–negative matrices. The quality of the approximation is measured using a divergence function defined below.

2.2.1 Bregman divergences

Bregman Divergences is a class of divergence functions closely related to the exponential family distributions, that is, there is a one–to–one mapping between regular exponential family distributions and the class of regular Bregman Divergences (Banerjee et al., 2005b).

Definition 2.2.1. Let $f : dom(f) \rightarrow \mathbf{R}$ be a continuously differentiable strictly convex function defined on the closed convex set $dom(f)$. The *Bregman Divergence* between $x, y \in dom(f)$ is defined as:

$$B_f(x, y) = f(x) - f(y) - \langle \nabla f(y), x - y \rangle \tag{2.1}$$

For two matrices $\mathbf{X}, \mathbf{Y} \in \mathbf{R}_+^{N \times d}$, the divergence $\mathcal{D}(\mathbf{X}, \mathbf{Y})$ is given by:

$$\mathcal{D}(\mathbf{X}, \mathbf{Y}) = \sum_{ij} B(x_{ij}, y_{ij}) \tag{2.2}$$

We restrict the class of divergence functions to belong to Bregman divergences (Definition 2.2.1) for formulating non–negative matrix factorization for interpretability given their attractive properties like convexity and associations with the exponential family distributions.

2.2.2 Non–negative matrix factorization

Let $\mathbf{X} \in \mathbf{R}_+^{N \times d}$ be a matrix with non–negative entries. Non–negative matrix factorization approximates the observation matrix as a factorization of two low–rank non–negative matrices. Let $\mathbf{A} \in \mathbf{R}_+^{N \times K}$ and $\mathbf{W} \in \mathbf{R}_+^{d \times K}$ be two rank K matrices. Then generalized non–negative matrix factorization aims to find estimates $\tilde{\mathbf{A}}$ and $\tilde{\mathbf{W}}$ via constrained minimization of the following cost function:

$$\tilde{\mathbf{A}}, \tilde{\mathbf{W}} = \underset{\mathbf{A} \in \mathbf{R}_+^{N \times K}, \mathbf{W} \in \mathbf{R}_+^{d \times K}}{\arg\min} \mathcal{D}(\mathbf{X}, \mathbf{A}\mathbf{W}^T) \tag{2.3}$$

where C_A and C_W denote appropriate constraints on the non–negative factors, either obtained via supervision or appropriate domain constraints. The choice of the divergence function $\mathcal{D}(\mathbf{X}, \mathbf{Y})$ is determined by the type of data comprising the observation matrix $\mathbf{X}$ and the probabilistic assumptions made for data generation. The choice of K is determined by empirical evaluation on a validation set or can be determined from the application domain.

2.2.3 Alternating–Minimization algorithm

In order to estimate the low–rank factor matrices $\mathbf{A}$ and $\mathbf{W}$, an effective and scaleable algorithm is called Alternating Minimization (Koren et al., 2009). Without any constraints on the estimates of $\mathbf{A}$ and $\mathbf{W}$ other than non–negativity, Algorithm 1 can be used to obtain the estimates.

2.3 Admixtures of Markov Random Fields

We demonstrate the utility of our domain specific interpretable models by applying it to a probabilistic latent variable model. In particular, we restrict to the class of admixture of Markov Random Fields, detailed in the following.

Algorithm 1 Alt-Min for NMF
Input: $\mathbf{X}$. Initialization: $\mathbf{A}_{(0)}$

while Not converged **do**
  $\mathbf{W}_{(t)} \leftarrow \arg\min_{\mathbf{W} \in \mathbf{R}_+^{d \times K}} \mathcal{D}(\mathbf{X}, \mathbf{A}_{(t-1)}\mathbf{W})$
  $\mathbf{A}_{(t)} \leftarrow \arg\min_{\mathbf{A} \in \mathbf{R}_+^{N \times K}} \mathcal{D}(\mathbf{X}, \mathbf{A}\mathbf{W}_{(t)})$
end while

2.3.1 Admixture models

Admixture models were primarily introduced to model heterogeneity in genetic linkage analysis data. The probabilistic assumptions underlying admixture models is as follows. Let K be the number of populations (or generally mixture components). Let $0 \leq w_k \leq 1$ be the proportion with which mixture k contributes to the observed population and let Θ_k parametrize the probability of observing a sample from the k^{th} component of the mixture (denoted by $p(x; \Theta_k)$). Let $\mathbf{x} \in \mathbb{R}_+^d$ be the random variable representing observation. Then an admixture model is represented by the following generative process:

$$\mathbf{x}_m \sim p(\mathbf{x}; \sum_k w_{mk}\Theta_k) \, \forall m \in [N] \tag{2.4}$$

where N is the total number of samples in the observation.

2.3.2 Poisson Markov Random Fields (PMRFs)

Poisson Markov Random Fields (PMRFs) (Yang et al., 2013), are markov random fields defined in order to incorporate correlation between multivariate Poisson random variables. Let $\mathbf{x} \in \mathcal{R}^V$ be a V dimensional count vector drawn from a PMRF. The distribution of $\mathbf{x}$ can be parametrized by $\theta \in \mathcal{R}^V$

and $\mathbf{\Theta} \in \mathcal{R}^{V \times V}$ and is given by[1]:

$$p(\mathbf{x}|\theta, \mathbf{\Theta}) \propto \exp\left\{\theta^T \mathbf{x} + \mathbf{x}^T \mathbf{\Theta} \mathbf{x} - \sum_{v=1}^{V} \ln(x_v!)\right\} \tag{2.5}$$

As can be seen from Equation (2.5), a PMRF explicitly accounts for potential correlation between the $\mathbf{x}_v$ vectors. $\mathbf{\Theta}$ plays a similar role as the precision matrix as in a multivariate Gaussian distribution i.e. encoding conditional independence structure. Note that (Inouye et al., 2014a) use a slightly modified distribution to account for positive correlations based on (Yang et al., 2013). An important distinction here is that in comparison to the multinomial distribution, used in LDA (Blei et al., 2003), PMRFs allow to model positive as well as negative correlations between words in the vocabulary. A multinomial distribution accounts for weak negative correlations by fixing the total count of trials and does not model correlations explicitly (Inouye et al., 2014a).

2.3.3 Admixtures of PMRFs (APM)

APM may be considered to be an undirected graphical model based analogue of LDA. Both model a document as a bag-of-words. Each document is represented as a vector, so each dimension counts the number of times a given word appears in the document. APMs are based on Poisson Markov Random Fields (PMRFs) (Inouye et al., 2014a), to incorporate correlation between multivariate Poisson random variables.

Consider K PMRFs - one for each topic, with parameters $\{\theta_k, \mathbf{\Theta}_k\}$. Topic models assume that a document is composed of words from multiple

[1]Note that proportionality signs imply appropriate normalization so that the distribution sums to 1.

topics. Therefore, to generate each document n in a corpus of N documents, each consisting of one or more of K topics, one can follow the following generative procedure using PMRFs:
For each $n \in N$,

- Sample $w_n \in \Delta^K$ according to a Dirichlet distribution $p(\mathbf{w}|\alpha)$, where $\alpha \in \mathcal{R}^K$, $\alpha > 0$ and Δ^K indicates the $K-1$ dimensional simplex (see 2.1). These are known as the admixing weights.
- Let $\theta_n = \sum_{k=1}^{K} w_{nk}\theta_k$ and $\mathbf{\Theta}_n = \sum_{k=1}^{K} w_{nk}\mathbf{\Theta}_k$. Since the 'weight' vector w_n lies on the simplex, θ_n and $\mathbf{\Theta}_n$ are convex combinations of the topic parameters.
- The document $\mathbf{x}_n$ is generated by sampling from a new PMRF with parameters $\{\theta_n, \mathbf{\Theta}_n\}$.

The complete distribution of the corpus $\mathbf{X}$ consisting of independent document samples $\mathbf{x}_n$, each drawn from a PMRF, is given by,

$$p(\mathbf{X}|\theta_k, \mathbf{\Theta}_k) \propto \prod_{n=1}^{N} p(\mathbf{x}_n|\theta_n = \sum_{k=1}^{K} w_{nk}\theta_k, \mathbf{\Theta}_n = \sum_{k=1}^{K} w_{nk}\mathbf{\Theta}_k) \tag{2.6}$$

where each entity in the product on the right hand side can be modeled according to Equation (1). In addition, prior probabilities $p(\theta_k, \mathbf{\Theta}_k|\beta)$ may be imposed on the parameters θ_k and $\mathbf{\Theta}_k$, $\forall\, k \in \{1, 2, ..., K\}$ (Inouye et al., 2014a). β can thus be considered as a tuneable hyperparameter.

2.3.4 *Maximum-a-Posteriori* algorithm for PMRFs

Inouye et al. (2014a) propose to obtain a *Maximum-a-Posteriori* (MAP) estimate of the parameters θ_k and $\mathbf{\Theta}_k$, $\forall\, k = \{1, 2, ..., K\}$, and an improved

scaleable approach for the MAP estimation procedure is proposed in Inouye et al. (2014b). We build upon this procedure for incorporating topic–level supervision. The unsupervised MAP estimation procedure involves alternating co–ordinate descent type optimization. One equation updates parameters of the topics i.e. the PMRF parameters with constant admixing weights and the other equations updates the admixing weights with constant topic parameters.

Let $\mathbf{z}_i = [1,\ \mathbf{x}_i^T]^T$, $\phi_{kv} = [\theta_{kv},\ \mathbf{\Theta}_{kv}]$ where v indexes the v^{th} row of θ_k and $\mathbf{\Theta}_k \,\forall\, k \in \{1, 2, ..., K\}$. In addition let $\mathbf{\Phi}_v = [\phi_{1v}, \phi_{2v}, ...\phi_{Kv}]$. The optimization problem is given by the following two equations optimized in an alternating manner.

$$\begin{aligned} &\arg\min_{\mathbf{\Phi}_v} -\frac{1}{n}\sum_{v=1}^{V}[tr(\mathbf{\Psi}_v\mathbf{\Phi_v}) - \sum_{i=1}^{N}\exp\left(\mathbf{z}_i^T\mathbf{\Phi}_v\mathbf{w}_i\right)] + \\ &\lambda\sum_{v=1}^{V}\|vec(\mathbf{\Phi}_v)_{-i}\|_1 \end{aligned} \tag{2.7}$$

$$\arg\min_{w_1,...w_n\in\Delta^K} -\frac{1}{n}\sum_{i=1}^{N}[\,\Psi_i^T w_i - \sum_{v=1}^{V}\exp\mathbf{z}_i^T\mathbf{\Phi}_v w_i] \tag{2.8}$$

where $\mathbf{\Psi}_v$ and Ψ_v can be calculated from observations $\mathbf{X}$. The subscript $-i$ indexes the i^{th} subvector of the vectorized form of Φ_v. The above equations are iteratively minimized to obtain a local optimum over the PMRF parameters $\mathbf{\Phi}_v$ and the admixing weights $w_1, ..., w_n$. Equation (2.7) updates the PMRF parameters when the admixing weights are fixed and the admixing weights are updated in Equation (2.8) with PMRF parameters fixed from latest estimates of (2.7).

2.4 Deep Generative Models

Generative Models can be described as stochastic procedures that generate samples (denoted by the random variable $\mathbf{x} \in \mathbb{X}^d$) from the data distribution $p(\mathbf{x})$ without explicitly parameterizing $p(\mathbf{x})$. The two most significant types are the Variational Auto–Encoders (VAEs) (Kingma and Welling, 2013) and Generative Adversarial Networks (GANs) (Goodfellow et al., 2014a). Implicit generative models generally assume an underlying latent dimension $\mathbf{z} \in \mathbb{R}^k$ that is mapped to the ambient data domain $\mathbf{x} \in \mathbb{R}^d$ using a deterministic function $\mathcal{G}_\theta$ parametrized by θ, usually as a deep neural network. The primary difference between GANs and VAEs is the training mechanism employed to learn function $\mathcal{G}_\theta$. GANs employ an adversarial framework by employing a discriminator that tries to classify generated samples from the deterministic function versus original samples and VAEs maximize an approximation to the data likelihood. The approximation thus obtained has an encoder-decoder structure of conventional autoencoders. We use VAEs for our explainability experiments. One can obtain a latent representation of any data sample within the latent embedding using the trained encoder network. While GANs do not train an associated encoder, recent advances in adversarially learned inference like BiGANs (Dumoulin et al., 2016; Donahue et al., 2016) can be utilized to obtain the latent embedding. In this work, we assume access to an implicit generative model that allows us to obtain the latent embedding of a data point.

2.5 Learning to Rank (LeTOR)

Learning to Rank or LeTOR models consider the problem of estimating a preference order over a set of items (Liu, 2009). For instance, ranking a fixed

set of web documents in order of relevance to a search query. Listwise ranking requires to rank order a list of objects in order of preference or relevance. We briefly discuss the listwise LeTOR algorithm used in this work below.

2.5.1 LeTOR using Monotone Retargeting (MR)

MR is a supervised listwise ranking technique that learns a Generalized Linear Model (GLM) on scores/ranks over a set of objects. MR leverages the idea that only the ordering induced by the scores over items are of consequence in a LeTOR framework. MR thus searches for parameter estimates over all monotonic transformations of the scores. That is, Acharyya et al. (2012) observe that listwise ranking only aims to learn an appropriate permutation over items in a query which can be interpreted as learning a scoring function on any monotonic transformation of the original scores to preserve order over items. This allows for parameter estimation of the GLM based cost function by fitting over target score vectors in addition to all scores isotonic to the original – called *retargeting*. Allowing such a retargeting has significant advantages under model–misspecification by providing more flexibility to under–specified models. For instance, linear GLMs can be used to fit integer scores (most commonly used in practice for annotating) by searching for an appropriately *retargeted* set of scores. Further, MR exploits the fact that the set of all vectors *isotonic* with the scores associated with a group of items in a query is a convex cone. Thus, the listwise ranking can be formulated as a biconvex optimization problem that alternately estimates the scoring function parameter and *retargets* the scores within the appropriate convex cone.

Specifically (notation is consistent with that of Acharyya et al. (2012)), let $\mathcal{Q} = \{q_1, q_2, \cdots, q_t\}$ be a set of queries each consisting of items $\mathcal{V}_{q_i} \subset \mathcal{V}, i \in$

Algorithm 2 Monotone Retargeting (MR)
Input: $\mathbf{X}_q \in \mathbb{R}^{|\mathcal{V}_q| \times d}, \mathbf{y}_q, q \in \mathcal{Q}; \phi$

Initialize $\mathbf{w}$, $\mathbf{r}_q, q \in \mathcal{Q}$:
while Not converged **do**
Solve using parameter estimation for GLMs:
$\mathbf{w} = \arg\min_{\mathbf{w} \in \mathbb{R}^d} \sum_{q \in \mathcal{Q}} D_\phi(\mathbf{r}_q \| \nabla \phi^{-1}(\mathbf{X}_q \mathbf{w}))$
Retargeting step:
$\mathbf{r}_q = \arg\min_{\mathbf{r}_q \in \mathcal{R}_{\downarrow_{\mathbf{y}_q}}} D_\phi(\mathbf{r}_q \| \nabla \phi^{-1}(\mathbf{X}_q \mathbf{w})) \, \forall q \in \mathcal{Q}$ in parallel
end while

$[t]$ to be ranked. Let $\mathbf{X}_q \in \mathbb{R}^{|\mathcal{V}_q| \times d}, q \in \mathcal{Q}$ be the feature matrix associated with these items and let $\mathbf{y}_q$ be scores representing the ranking permutation. Let $D_\phi(\mathbf{x}, \mathbf{y})$ be an appropriate[2] distance like function between two vectors $\mathbf{x}, \mathbf{y} \in \mathbb{R}^d$. Finally let $\mathcal{R}_{\downarrow_{\mathbf{y}_q}}$ represent the convex cone of all vectors that are isotonic to $\mathbf{y}_q$, i.e. all vectors that result in the same rank order as $\mathbf{y}_q$. Then MR for listwise ranking can be formulated to estimate a function parametrized by $\mathbf{w} \in \mathbb{R}^d$ that fits any monotonic transformation of the score vector. That is,

$$\tilde{\mathbf{w}} = \underset{\mathbf{w} \in \mathbb{R}^d, \mathbf{r}_q \in \mathcal{R}_{\downarrow_{\mathbf{y}_q}}, q \in \mathcal{Q}}{\arg\min} \sum_{q \in \mathcal{Q}} D_\phi(\mathbf{r}_q, \nabla \phi^{-1}(\mathbf{X}_q \mathbf{w})) \tag{2.9}$$

MR uses Bregman Divergences (see Definition 2.2.1) in order to measure the quality of the fit to the rank scores. The estimation algorithm is an alternating minimization procedure comprising of a standard parameter estimation step as that of Generalized Linear Models (GLMs) and a 'Retargeting Step' solved using the Pool-Adjacent Violators (PAV) algorithm (Best and Chakravarti, 1990). The retargeting step allows to fit the GLM over any vector isotonic to the target scores and hence have to be recomputed every time

[2]Bregman Divergence

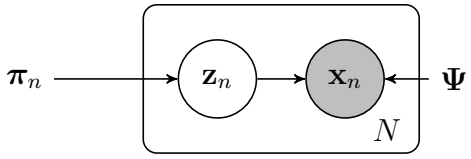


Figure 2.1: Latent Variable Model for Clustering

the GLM parameter estimate updates. The complete algorithm for LeTOR using MR is summarized in Algorithm 2. The formulation easily allows to account for partial ordering by augmenting the algorithm with a simple permutation step (Acharyya et al., 2012).

Extensions of MR, called Margin Equipped MR (MEMR) mitigate issues like degeneracy of solutions by allowing to add margins within the ranking scores and augmenting the cost function using ℓ_2–regularization to ensure joint convexity.

2.6 Clustering

Clustering is the task of estimating a partition of the data given finite samples from the data distribution. Without further assumptions, this problem is ill–posed. We formulate our clustering problem using a probabilistic latent variable framework. Figure 2.1 show the corresponding graphical model that induces appropriate probabilistic dependencies to describe the generative process of the clustering formulation. Specifically, $\mathbf{z}$ is the latent (unobserved) random variable representing cluster membership for any sample. π_n is the prior probability of a sample n belonging to one of K clusters. $\mathbf{x}_n$ is the observed sample that is generated as follows: 1. $\mathbf{z}_n \sim p(\mathbf{z}; \pi_n)$ 2. $\mathbf{x}_n \sim p(\mathbf{x}; \Psi_{\mathbf{z}_n})$

where Ψ_k parametrizes the probability corresponding to the k^{th} cluster.

A *categorical distribution* is a discrete distribution over outcomes $\omega \in [K]$ parameterized by $\theta \in \Delta^K$ so that $Pr(\omega = k) = \theta_k$. It is a member of the exponential family of distributions. The natural parameters of categorical distribution are $\log \theta = (\log \theta_k)_{k \in [K]}$ and sufficient statistics are given by the vector of indicator functions for each outcome $\omega \in [K]$, denoted by $\mathbf{z}(\omega) \in \{0, 1\}^K$ with:

$$z_k(\omega) = \begin{cases} 1, & \text{if } \omega = k, \\ 0, & \text{otherwise.} \end{cases}$$

In the proposed generative model, $\mathbf{z}$ is modeled as a categorical variable.

Given two categorical distributions $p(\omega)$ and $q(\omega)$, describing the distribution over the categorical random variable ω, the *divergence* of $p(\omega)$ from $q(\omega)$, denoted $\mathcal{D}(p(\omega) \| q(\omega))$, is a non-symmetric measure of the difference between the two probability distributions. The *Kullback-Leibler* or KL-divergence is a specific divergence denoted by $\mathrm{KL}(p(\omega) \| q(\omega))$ and is defined as follows.

KL-divergence of $p(\omega)$ from $q(\omega)$ is given by:

$$\mathrm{KL}(p(\omega) \| q(\omega)) = \mathrm{E}_{p(\omega)} \left[\log p(\omega) - \log q(\omega) \right] \tag{2.10}$$

This is also known as the relative entropy between $p(\omega)$ and $q(\omega)$. The relative entropy is non-negative and jointly convex with respect to both arguments. Further, we have that $\mathrm{KL}(p(\omega) \| q(\omega)) = 0$ iff $p(\omega) = q(\omega)$, for all ω. Note that the KL–divergence is a special case of the Bregman Divergence 2.2.1.

The Rényi divergences (Rényi, 1960) are a parametric family of divergences with many similar properties to the KL-divergence. Since our focus is on using these divergences to measure distances of distributions over cluster labels, we will focus on Rényi divergences for distributions over discrete random variables.

Definition 2.6.1. (van Erven and Harremoës, 2012) Let p, q be two distributions for a random variable $\omega \in [K]$. The Rényi divergence of order $\gamma \in (0,1) \cup (1,\infty)$ of $p(\omega)$ from $q(\omega)$ is,

$$\mathcal{D}_\gamma(p(\omega)\|q(\omega)) = \frac{1}{\gamma - 1} \log \Big(\sum_{\omega=1}^{K} p(\omega)^\gamma q(\omega)^{(1-\gamma)} \Big) \tag{2.11}$$

The definition may been extended for divergences of other orders like $\gamma = 0,\ \gamma \to 1,$ and $\gamma \to \infty$ (van Erven and Harremoës, 2012). Rényi divergences are non-negative $\forall \gamma \in [0,\infty]$. In addition, they are jointly convex in $(p,\ q)$ $\forall \gamma \in [0,1]$ and convex in the second argument q $\forall \gamma \in [0,\infty]$. As discussed in the comprehensive survey of Rényi divergences by van Erven and Harremoës (2012), many special cases of other commonly used divergences are recovered for specific choices of γ. For example, $\gamma = \frac{1}{2}$ and $\gamma = 2$ give Rényi divergences which are closely related to the Hellinger and χ^2 divergences, respectively, and the KL-divergence is recovered as a limiting case when $\gamma \to 1$. For the rest of the manuscript, we will abuse notation slightly and use $p(\omega)$ and $p(\mathbf{z})$ interchangeably to denote the same categorical distribution over outcomes in $[K]$.

2.7 Discussion

The following chapters unify all models under the latent variable framework and discusses how different aspects of these models can be constrained for improved interpretability, explainability and semisupervised learning.

Chapter 3

Interpretable Latent Variable Models

Interpretability and Explainability of machine learning models are becoming increasingly imperative as they become widely applied to domains like the criminal justice system (Angwin et al., 2016), clinical healthcare (Callahan and Shah, 2017), etc. The COMPAS (Angwin et al., 2016) system learns recidivism scores to determine pre–trial bail and detention. Clinical interventions determined using machine learning algorithms can affect patient lives, thus making it important for caregivers to provide explanations for such interventions. Such applications that substantially impact human lives have motivated regulatory agencies like the EU Parliament[1] to codify a right to data protection and "obtain an explanation of the decision reached using such automated systems[2]".

Challenges in this domain are compounded by a lack of characterization of what constitutes a sufficient explanation (Lipton, 2016). Additionally, different levels of abstractions are necessary depending on the stakeholders (Miller, 2017). For instance, explanations of interesting behaviors that may assist a data science practitioner are vastly different from those that help caregivers and/or patients make better interventional choices. Doshi-Velez (2017); Miller (2017) have recently attempted to characterize such abstractions from the

[1] in collaboration with the EU Commission and the Council of the Eurpean Union
[2] https://www.privacy-regulation.eu/en/r71.htm

perspective of the desired outcome as well as drawing from extensive social scientific literature on how humans process explanations. Generally, there is ground to believe that such a suite of methods can be useful not only help improve understanding of opaque models[3] (Higgins et al., 2016; Karpathy et al., 2015) but can also uncover biases (inherent in the data) that models pick up on e.g. learned gender and racial biases (Bolukbasi et al., 2016).

3.1 Related Work

While generally referred to interchangeably, we distinguish interpretable machine learning models as those that learn easily understandable outcomes to a target user. On the other hand, explainability tools refer to models that can be used to provide post–hoc explanations of pre–trained complex models. Some models have been exclusively developed in order to serve as *diagnostic tools* to 'explain' existing or pre–trained models. Notable ones are described in the following. We describe existing work relevant to developing interpretable models, as well as explainable models in the following. The rest of the chapter is thereafter devoted to exposing the utility of interpretable machine learning using constraints in latent variable models. The explainability exposition is relegated to Chapter 6.

3.1.1 Interpretable machine learning

Interpretable ML methods focus on developing machine learning models whose outcomes inherently satisfy a specific interpretability criterion. Usually, such criterion tend to be domain as well as application specific. Notable

[3] https://distill.pub/2018/building-blocks/

among these are methods that use tools based on model distillation (Hinton et al., 2015) and attention based mechanisms (Ba et al., 2014). As a working example, we focus on interpretable models that have been developed for clinical healthcare. The main goal of interpretability of ML models in clinical decision making is to expect the model to learn clinically relevant, physiologically plausible, and represented in a form or abstraction that is understandable to clinical experts. For instance, Choi et al. (2016) use attention based mechanism for time series data for training explainable models for outcome prediction, while Che et al. (2016) use model compression and distillation, similarly for outcome prediction for an ICU patient population. This dissertation focuses on phenotyping (Pathak et al., 2013) of co-occurring chronic conditions for ICU patients as the working application for developing inherently interpretable models. EHR driven phenotypes are concise representations of observable clinical traits that can facilitate reliable querying of individuals from the EHRs (NIH Health Care Systems Research Collaboratory, 2014). While most interpretability mechanisms described above focus on supervised models, EHR driven phenotype has to be posed as an unsupervised learning problem with availability of weak or noisy supervision.

3.1.2 Explainable machine learning

Stock and Cisse (2017); Kim et al. (2016); Gupta et al. (2016); Lundberg and Lee (2017) develop models specifically to explain classifier decision and behavior. Different methodologies are used in order to provide such *post-hoc* explanations. For instance Elenberg et al. (2017); Kim et al. (2016) select prototypes/examples and/or groups or semantically relevant features from the training dataset as a means to detect failure cases of supervised models.

It may happen that points that explain a model according to the predetermined criterion may not exist in the dataset. In order to solve this problem we propose a method to generate samples by approximating the data manifold using a generative model like a GAN (Goodfellow et al., 2014a) or a VAE (Kingma and Welling, 2013). Stock and Cisse (2017) use the adversarial attack paradigm (Goodfellow et al., 2014b) to generate prototypes and/or examples where the classifier shows interesting failure cases (called *criticisms*). Another class of methods locally approximate complex classifiers with a simpler model class (e.g. linear) in order to generate explanations (Lundberg and Lee, 2016; Ribeiro et al., 2016; Shrikumar et al., 2016; Bach et al., 2015). These methods inherently assume a trade–off between model complexity and explainability. Empirically, it is observed that simpler model classes also tend to be empirically sub–par. Thus such models inherently assume a trade-off between model performance and explainability. Li et al. (2015); Selvaraju et al. (2016) focus on understanding the workings of different layers of a deep network and studying saliency maps for feature attribution (Simonyan et al., 2013; Smilkov et al., 2017; Sundararajan et al., 2017). Saliency methods, while powerful, can be demonstrated to be unreliable without stronger conditions over the saliency model (Kindermans et al., 2017; Adebayo et al., 2018). Koh and Liang (2017) use influence functions, motivated by robust statistics (Cook and Weisberg, 1980) to determine importance of each training sample for model predictions.

3.2 Latent Variable Models for Interpretability

This dissertation focuses on the class of latent variable models to propose the interpretability and explainability framework. We posit that constraining latent variable models appropriately can allow to learn models that

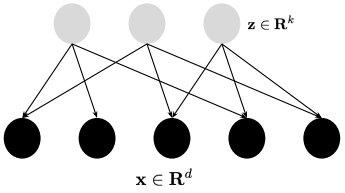
generate interpretable outcomes and to explain existing ML models in a post–hoc manner. Probabilistic graphical models, latent factor models like matrix factorization, and implicit generative models are a few well known examples within this class. In particular, this framework offers the following advantages in terms of its amenability to formulating explainable and interpertable machine learning models:

- Latent variable models induce an associated probabilistic generative procedure for the observed data. Constraining the generative process allows to easily encode constraints that make the model (say physiologically) plausible and therefore more interpretable.
- In particular, constraints on the model class, parameters of the model class, as well as the generative procedure itself can be imposed for individual observational samples, lending the model to be more amenable to generating *individualized/personalized* explanations whenever necessary.
- Scaleable learning and inference procedures can be non–trivially extended for this class of models that can be augmented seamlessly to incorporate any relevant constraints.
- Specifically for the working example of phenotyping chronic conditions, this framework allows to learn phenotypes for all chronic conditions simultaneously (a modeling requirement since such chronic conditions tend to co-occur or are *comorbidities* (Elixhauser et al., 1998)).

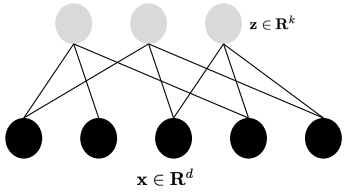
The following describes a general framework to formulate interpretable latent variable models by constraining latent variable models. A detailed motivation for constrained based models to (post-hoc) explain black-box models is deferred to Chapter 6.

3.3 Constraint Based Framework for Interpretability

Let $\mathbf{x} \in \mathbb{R}^d$ be the random variable representing observations. Let $\mathbf{z} \in \mathbb{R}^k$, $k << d$ represent the set latent (unobserved) variables that can well approximate the observation $\mathbf{x}$ via function f_θ. That is, let $f_\theta : \mathbb{R}^k \to \mathbb{R}^d$ define an approximation to the observations as a function of unobserved variables $\mathbf{z}$. Let $\mathcal{L} : \mathbb{R}^{d \times d} \to \mathbb{R}_+$ determine the quality of such an approximation. A few examples making this framework concrete in different settings are given below:



(a) Directed Graphical Model



(b) Undirected Graphical Model

1. **Probabilistic Graphical Models:** A probabilistic graphical model is a framework to encode dependencies between a set of random variables and an associated realizable probabilistic distribution. Figure 3.1(a) shows a graphical model that demonstrates the dependency between the latent variables $\mathbf{z}$ and the observational data $\mathbf{x}$, while Figure 3.1(b) shows an undirected graphical model analogue encoding the dependency structure between latents $\mathbf{z}$ and observed variables $\mathbf{x}$.

2. **Latent Factor Models:** Latent factor models is a class of models that expresses observations a linear combination of shared 'factors' or variables, where both, the shared factors as well as the strength of the linear combinations are unknown (latent). We restrict to the class of non–negative matrix factorization in this study. Refer to background in Sec 2.2 for details.

In particular, the probabilistic assumptions induced in NMF can be represented as the following. Let $\mathbf{w} \in \mathbb{R}_+^k$ be the latent variable representing the unknown linear combinations or loadings while let the columns of the matrix $\mathbf{A} \in \mathbb{R}_+^{d \times K}$ (denoted by $\mathbf{a}^{(k)}$) represent the common or shared factors across observed samples. Then,

$$\mathbb{E}[\mathbf{x}|\mathbf{w}] = \sum_{k \in [K]} \mathbf{a}^{(k)} w_k \tag{3.1}$$

3. **Implicit Generative Models:** Implicit Generative Models are generative models that map latent variable $\mathbf{z}$ to observed data $\mathbf{x}$ via a deterministic function $\mathcal{G}_\theta$ without parametrizing the underlying stochastic process. Examples of such a deterministic function can be a deep neural network. Such models are usually trained either via a maximum likelihood procedure or an adversarial training procedure (see Sec 2.4) for more details.

To formulate interpretable models, we propose to impose model constraints on 1. latent variables 2. model parameters 3. generative procedure or a combination thereof. We represent constraints on the latent variables as $C_{\mathbf{z}}$. Constraints on model parameters are represented as C_θ and that on the generative process can be described as part of model assumptions. In general, the optimization process can be formulated as the following:

$$\begin{aligned} \tilde{\mathbf{z}}, \tilde{\boldsymbol{\theta}} &= \arg\min_{\mathbf{z}, \theta} \mathbb{E}[L(\mathbf{x}, f_\theta(\mathbf{z}))] \\ &\textbf{s. t. } \mathbf{z} \in C_{\mathbf{z}}, \boldsymbol{\theta} \in C_{\boldsymbol{\theta}} \end{aligned} \tag{3.2}$$

3.3.1 Augmented training

Without interpretability requirements, the loss function determining the quality of the approximation can be optimized with respect to model pa-

rameters without additional constraints. Specifically, in the absence of constraints on latent variables, out-of-the-box training algorithms can be utilized for learning model parameters, which are of primary interest in general. However, interpretability constraints on latent variables can be imposed by algorithms that allow to interleave model parameter estimation with imposing required constraints. Thus, we suggest that the class of algorithms that rely on model parameter estimation without marginalizing latent variables are more amenable to developing interpretable models via constraints. Typically such a class of algorithms follow the prescription of majorization–maximization (Hunter and Lange, 2000) construct of algorithms and leverage the latent variable inference framework to achieve tractability of an otherwise complex optimization algorithm. Examples of algorithms used in this dissertation for imposing interpretability constraints are 1. Expectation-Maximization 2. Variational Inference (Wainwright and Jordan, 2008) for the class of probabilistic graphical models, 1. Alternating–Minimization for latent factor analysis.

3.3.2 *Grounding* mechanism

One way to impose constraints, that is specifically useful for the phenotyping application is described in the following. The mechanism, called *grounding*, is extensively evaluated for different models in the following chapters. The mechanism involves enforcing constraints on the latent variables and/or on the model parameters.

1. *Support Constraints:* This set of constraints is imposed on the support of individual samples of the latent variable $\mathbf{z}$. Let $j \in [N]$ index individual sample observations. We assume a set C_j can be determined from side information such that an *interpretable* model would only allow for estimates satisfying:

$\text{supp}(\mathbf{z}^{(j)}) = C_j$. We motivate this using the non-negative matrix factorization setting in the context of phenotyping. The non-negative rank–K factorization of $\mathbf{X}$ is said to be 'grounded' to K target comorbidities by constraining the support of loadings $w^{(j)}$ corresponding to patient j using weak diagnosis C_j that can be easily computed from administrative patient data. This amounts to restricting the set of allowable linear combinations that can describe an observed phenotype for any patient sample. As we shall see in Chapter 5, if C_j are accurate, then this constraint follows from the definition of phenotypes.

2. *Sparsity Constraints:* In many applications, model parameter estimates are eventually consumed by domain experts for final decision making. Thus, it is desirable that the phenotype representations be easily interpretable for human experts. Sparsity of the model parameter $\boldsymbol{\theta}$ is used as a measure of domain specific interpretability. For the case of phenotyping using non–negative matrix factorization, sparsity is induced using the scaled simplex constraints on the columns of $\mathbf{A}$. Associated with the constraint is a tuneable parameter $\lambda > 0$ to encourage sparsity of phenotypes.

3.4 Discussion

Advantages of such constraints are 1. they follow easily from generative assumptions made on the data, 2. convexity and tractability – allowing to impose exact constraints during training. This dissertation further demonstrates the empirical advantages of such a *grounding* mechanism for phenotyping (over competitive models) as well as for downstream applications like mortality or risk prediction.

Chapter 4

Applications to Interpretable Phenotyping

Raw EHR data has demonstrated great potential in determining patient outcomes as well the possibility of providing individualized or precision medical care (Callahan and Shah, 2017). While predicting outcome and modeling disease progression have been identified as important tasks that can benefit from Machine Learning techniques, a few requirements remain fundamental across all clinical applications. Specifically, it is important that such models satisfy certain interpretability requirements. An instance of an interpretable model is one that provides physiologically plausible outcomes. Typically, interpretability of models in clinical healthcare refers to the availability of abstractions to non-experts in a manner suitable to make reliable decisions. Machine Learning models generally do not satisfy these criteria without additional constraints. This chapter motivates the need for interpretable and automated phenotyping using Electronic Health Records (EHRs). We also describe the data pre–processing that served as a precursor to evaluating our *grounding* procedure for unsupervised phenotyping of chronic conditions for an ICU population. The preprocessing procedures described here are detailed further in Joshi et al. (2015, 2016b).

This chapter is based on content pubished in Joshi et al. (2015, 2016b). The author of this dissertation contributed to problem formulation, and the data preprocessing detailed in the chapter.

4.1 Automated EHR based Phenotyping

Reliably querying for patients with specific medical conditions across multiple organizations facilitates many large scale healthcare applications such as cohort selection, multi-site clinical trials, epidemiology studies etc. (Richesson et al., 2013; Hripcsak and Albers, 2013; Pathak et al., 2013). However, raw EHR data collected across diverse populations and multiple caregivers can be extremely high dimensional, unstructured, heterogeneous, and noisy. Manually querying such data is a formidable challenge for healthcare professionals.

EHR driven phenotypes are concise representations of medical concepts composed of clinical features, conditions, and other observable traits facilitating accurate querying of individuals from EHRs. Efforts like eMerge Network[1] and PheKB[2] are well known examples of EHR driven phenotyping. Traditionally used rule–based composing methods for phenotyping require substantial time and expert knowledge and have little scope for exploratory analyses. This motivates automated EHR driven phenotyping using machine learning with limited expert intervention.

4.1.1 Prognosis of Comorbidities

Our working example focuses on phenotyping 30 co–occurring conditions (comorbidities) observed in intensive care unit (ICU) patients. *Comorbidities* are a set of co-occurring conditions in a patient at the time of admission that are not directly related to the primary diagnosis for hospitalization (Elixhauser et al., 1998). Phenotypes for the 30 comorbidities listed in Table 4.1 are

[1] http://emerge.mc.vanderbilt.edu/
[2] http://phekb.org/

derived using text-based features from clinical notes in a publicly accessible MIMIC-III EHR database (Saeed et al., 2011).

The following aspects of our model distinguish our work from prior efforts in phenotyping:

1. **Identifiability:** A key shortcoming of standard unsupervised latent factor models such as NMF (Lee and Seung, 2001) and Latent Dirichlet Allocation (LDA) (Blei et al., 2003) for phenotyping is that, the estimated latent factors learnt are interchangeable and *unidentifiable* as phenotypes for specific conditions of interest. We tackle identifiability by incorporating weak (noisy) but inexpensive supervision as constraints our framework. Specifically, we obtain weak supervision for the target conditions in Table 4.1 using the Elixhauser Comorbidity Index (ECI) (Elixhauser et al., 1998) computed solely from patient administrative data (without human intervention). We then ground the latent factors to have a one-to-one mapping with conditions of interest by incorporating the comorbidities predicted by ECI as *support constraints* on the patient loadings along the latent factors.

2. **Simultaneous modeling of comorbidities:** ICU patients studied in this work are frequently afflicted with multiple co–occurring conditions besides the primary cause for admission. In the proposed NMF model, phenotypes for such co–occurring conditions jointly modeled to capture the resulting correlations.

3. **Interpretability:** For wider applicability of EHR driven phenotyping for advance clinical decision making, it is desirable that these phenotype definitions be clinically interpretable and represented as a concise set of rules. We consider the sparsity in the representations as a proxy for interpretability

and explicitly encourage conciseness of phenotypes using tuneable sparsity–inducing soft constraints.

4.1.2 Data Pre-Processing

We describe data pre–processing for our phenotyping application as the processing can be significantly inter-leaved with the mechanisms used to impose *grounding* to develop interpretable models. In the following, we describe the pre–processing for each of the models that will be described in chapter 5.

4.1.2.1 Phenotyping using grounded NMF

MIMIC-III (Saeed et al., 2011) was used for phenotyping using constrained non–negative matrix factorization (see 5.1). The MIMIC-III dataset consists of de-identified EHRs for $\sim 38,000$ adult ICU patients at the Beth Isreal Deaconess Medical Center, Boston, Massachusetts from 2001–2012. For all ICU stays within each admission, clinical notes including nursing progress reports, physician notes, discharge summaries, ECG, etc. are available. We analyze patients who have stayed in the ICU for at least 48 hours ($\sim$ 17000 patients). We derive phenotypes using clinical notes collected within the first 48 hours of patients' ICU stay to evaluate the quality of phenotypes when limited patient data is available. Further, we evaluate the phenotypes on a 30 day mortality prediction problem. To avoid obvious indicators of mortality and comorbidities, apart from restricted to first 48 hour data, we exclude discharge summaries as they explicitly mention patient outcomes (including mortality).

1. **Clinically relevant bag-of-words features:** Aggregated clinical notes from all sources are represented as a single *bag-of-words* features. To

Table 4.1: Target comorbidities

Congestive Heart Failure	Cardiac Arrhythmias	Valvular Disease	Pulmonary Circulation Disorder	Peripheral Vascular Disorder
Hypertension	Paralysis	Other Neurological Disorders	Chronic Pulmonary Diseases	Diabetes Uncomplicated
Diabetes Complicated	Hypothyroidism	Renal Failure	Liver Disease (excluding bleeding)	Peptic Ulcer
AIDS	Lymphoma	Metastatic Cancer	Solid Tumor (without metastasis)	Rheumatoid Arthritis
Coagulopathy	Obesity	Weight loss	Fluid Electrolyte Disorder	Blood Loss Anemia
Deficiency Anemia	Alcohol abuse	Drug abuse	Psychoses	Depression

enhance clinical relevance, we create a custom vocabulary containing clinical terms from two sources (a) the Systematized Nomenclature of Medicine-Clinical Terms (SNOMED CT), and (b) the level-0 terms provided by the Unified Medical Language System (UMLS), consolidated into a standard vocabulary format using MetamorphoSys — an application provided by UMLS for custom vocabulary creation.[3] To extract clinical terms from the raw text, the notes were tagged for chunking using a conditional random field tagger[4]. The tags are looked up against the custom vocabulary (generated from MetamorphoSys) to obtain the *bag-of-words* representation. Our final vocabulary has ∼3600 clinical terms.

2. **Computable weak diagnosis:** We incorporate domain constraints from weak supervision to ground the latent factors to have a one-to-one mapping with the conditions of interest. This is enforced by constraining the non-zero entries on patient loading along the latent factors using a weak diagnosis for comorbidities. The weak diagnoses of target comorbidities in Table 4.1 are obtained using ECI[5], computed solely from patient administrative data without human annotation. We refer to this index as *weak diagnoses* as it is not a physician's exact diagnosis and is subject to noise and misspecification. Note that ECI ignores diagnoses code related to the primary diagnoses of ad-

[3]See https://www.nlm.nih.gov/healthit/snomedct/ and https://www.nlm.nih.gov/research/umls/

[4]https://taku910.github.io/crfpp/

[5]https://git.io/v6e7q

mission. Thus, ECI models presence and absence of conditions other than the primary reason for admission (comorbidities). The phenotype candidates from the proposed model can be considered as concise representations of such comorbidities.

4.1.2.2 Phenotyping using grounded APM

The data pre–processing is similar for the algorithm described in 4.1.2.1 except that the clinical bag-of-words are generated from a standard English vocabulary and instead of restricting to first 48 hours of patient data, we use all available clinical notes (except discharge data) to generate our observation matrix.

4.2 Discussion

The bag-of-words clinical features extracted above are represented in a matrix form and determine the phenotype representation in conjunction with the associated learning algorithm and constraints. For grounded NMF, the phenotype representation is a collection of (a few) relevant terms (from the clinical vocabulary) associated with the target conditions. For the admixture model based on APM, a graph representation of a phenotype is learned, where the nodes of the graph are relevant terms associated with a chronic condition of interest and the edges between the terms represent co-occurrence structure. The target conditions are determined by the derived computational weak diagnoses that inform the grounding constraints. In the following chapters, we describe both models in detail.

Chapter 5

Identifiable Phenotyping of Chronic Conditions

This chapter explains how the aforementioned *Grounding* mechanism (see Chapter 3) is applied to phenotyping in two distinct machine learning models. Each section in the following describes the model in conjunction with its learning and inference procedure as well as results that compare the quality of the learned phenotypes. Additionally, we evaluate the predictive ability of the phenotype representations on patient outcomes like mortality and disease classification. The learning algorithms and evaluations detailed here appeared in Joshi et al. (2015, 2016b).

5.1 *Grounded* Non–Negative Matrix Factorization

We present a constrained non–negative matrix factorization (CNMF) for the EHR matrix that uses *grounding* to align the factors with target comorbidities yielding sparse, interpretable, and identifiable phenotypes. The method is evaluated for its efficacy toward learning interpretable, clinical relevant, and predictive phenotypes on EHR data from MIMIC-III. Although we focus on ICU patients using clinical notes, the proposed model and algo-

This chapter is based on content published in Joshi et al. (2015, 2016b). The author of the dissertation contributed to model formulation, implementation, and empirical evaluation described in this chapter.

Notation	Description
Observations	
N, d	Number of patients ($\sim$ 17000) and features ($\sim$ 3600), respectively.
$\mathbf{X} \in \mathbb{R}_+^{d \times N}$	EHR matrix from MIMIC III: Clinically relevant bag-of-words features from notes in first 48 hours of ICU stay for N patients.
$k = 1, 2, \ldots, K$	Indices for $K = 30$ comorbidities in Table 4.1.
$C_j \subseteq [K]$ for $j \in [N]$	Set of comorbidities patient j is diagnosed with using ECI .
Factor matrices	
$\tilde{\mathbf{W}} \in [0,1]^{K \times N}$	Estimate of *patients' risk* for the K conditions.
$\tilde{\mathbf{A}} \in \mathbb{R}_+^{d \times K}$, $\tilde{b} \in \mathbb{R}_+^d$	Estimate of *phenotype factor matrix* and *feature bias vector*.

Table 5.1: Additional notation used in this chapter

rithm are general and can be applied on any non-negative EHR data from any population group.

5.1.1 Identifiable high–throughput phenotyping

Additional notation used in this work are enumerated in Table 5.1 and in the following.

For each patient $j \in [N]$, (a) the bag-of-words features from clinical notes is represented as column $x^{(j)}$ of EHR matrix $\mathbf{X} \in \mathbb{R}_+^{d \times N}$, and (b) the list of comorbidities diagnosed using ECI is denoted as $C_j \subseteq [K]$.

Let an unknown $\mathbf{W}^* \in [0,1]^{K \times N}$ represent the risk of N patients for K comorbidities of interest; each entry w_{kj}^* lies in the interval $[0,1]$, with 0 and 1 indicating no-risk and maximum-risk, respectively, of patient j being afflicted with condition k. If $C_j^* \subseteq [K]$ denotes an accurate diagnosis for patient j, then $w^{*(j)}$ satisfies $\text{supp}(w^{*(j)}) \subseteq C_j^*$.

Definition 5.1.1 (EHR driven phenotype)**.** *EHR driven phenotypes* for K co–occurring conditions are a set of vectors $\{a^{*(k)} \in \mathbb{R}_+^d : k \in [K]\}$, such that

for a patient j afflicted with conditions $C_j^* \subseteq [K]$,

$$\mathbb{E}[x^{(j)}|w^{*(j)}] = \sum_{k \in C_j^*} w_{kj}^* a^{*(k)} + b^*, \tag{5.1}$$

where b^* is a bias representing the feature component observed independent of the K target conditions. $\mathbf{A}^* \in \mathbb{R}^{d \times K}$ with $a^{*(k)}$ as columns is referred as the *phenotype factor matrix*.

Note that we explicitly model a feature bias b^* to capture frequently occurring terms that are not discriminative of the target conditions, e.g., temperature, pain, etc. The choice of K is known and determined by the target latent topics we wish to model (can be determined from noisy supervision).

5.1.2 Incorporating grounding using convex constraints

Cost Function The bag-of-words features are represented as counts in the EHR matrix $\mathbf{X}$. We consider a factorized approximation of $\mathbf{X}$ parametrized by matrices $\mathbf{A} \in \mathbb{R}_+^{d \times K}$, $\mathbf{W} \in \mathbb{R}_+^{K \times N}$ and $b \in \mathbb{R}_+^d$ as $\mathbf{Y} = \mathbf{AW} + b\mathbb{1}^\top$, where $\mathbb{1}$ denotes a vector of all ones of appropriate dimension. The approximation error of the estimate is measured using the I–divergence defined as follows:

$$\mathcal{D}(\mathbf{X}, \mathbf{Y}) = \sum_{ij} y_{ij} - x_{ij} - x_{ij} \log \frac{y_{ij}}{x_{ij}}. \tag{5.2}$$

Minimizing the I–divergence is equivalent to maximum likelihood estimation under a Poisson distributional assumption on individual entries of the EHR matrix parameterized by $\mathbf{Y} = \mathbf{AW} + b\mathbb{1}^\top$ (Banerjee et al., 2005a).

Phenotypes For the K comorbidities, columns of $\mathbf{A}$, $\{a^{(k)}\}_{k \in [K]}$ are proposed as candidate phenotypes derived from the EHR $\mathbf{X}$, i.e. approximations to $\{a^{*(k)}\}_{k \in [K]}$.

The following *grounding* constraints are incorporated in learning $\mathbf{A}$ and $\mathbf{W}$.

1. *Support Constraints:* The non-negative rank–K factorization of $\mathbf{X}$ is 'grounded' to K target comorbidities by constraining the support of risk $w^{(j)}$ corresponding to patient j using weak diagnosis C_j from ECI as an approximation of the conditions in Definition 5.1.1.

2. *Sparsity Constraints:* Scaled simplex constraints are imposed on the columns of $\mathbf{A}$ with a tuneable parameter $\lambda > 0$ to encourage sparsity of phenotypes. Restricting the patient loadings matrix as $\mathbf{W} \in [0,1]^{K\times N}$ not only allows to interpret the loadings as the patients' risk, but also makes simplex constraints effective in a bilinear optimization.

5.1.3 λ-CNMF

Simultaneous phenotyping of comorbidities using constrained NMF is posed as follows:

$$\begin{aligned} \tilde{\mathbf{A}}, \tilde{\mathbf{W}}, \tilde{b} = &\operatorname{argmin}_{\mathbf{A}\geq 0, \mathbf{W}\geq 0, b\geq 0} && \mathcal{D}(\mathbf{X}, \mathbf{A}\mathbf{W} + b\mathbb{1}^\top) \\ &\text{s.t.} && \operatorname{supp}(w^{(j)}) = C_j \ \forall j \in [N],\ \mathbf{W} \in [0,1]^{K\times N}, \\ & && a^{(k)} \in \lambda\Delta^{d-1} \ \forall j \in [K], \end{aligned} \tag{5.3}$$

The optimization in (5.3) is convex in either factor with the other factor fixed. It is solved using alternating minimization with projected gradient descent (Parikh and Boyd, 2014; Lin, 2007). See complete algorithm in Algorithm 3. The proposed model in general can incorporate any weak diagnosis of medical conditions. In this work, we note that, since we use ECI, the results are not representative of the primary diagnoses at admission.

Algorithm 3 Phenotyping using constrained NMF.
Input: $\mathbf{X}$, $\{C_j : j \in [N]\}$ and paramter λ. Initialization: $\mathbf{A}_{(0)}, b_{(0)}$.

while Not converged **do**

$$\begin{aligned} \mathbf{W}_{(t)} \leftarrow \arg\min_{\mathbf{W}} \mathcal{D}(\mathbf{X}, \mathbf{A}_{(t-1)}\mathbf{W} + b_{(t-1)}\mathbb{1}^\top) \\ \text{s.t. } \mathbf{W} \in [0,1]^{K\times N}, \text{supp}(w^j) = C_j, \ \forall j \end{aligned} \tag{5.4}$$

$$\begin{aligned} \mathbf{A}_{(t)}, b_{(t)} \leftarrow \arg\min_{\mathbf{A}, b\geq 0} \mathcal{D}(\mathbf{X}, \mathbf{A}\mathbf{W}_{(t)} + b\mathbb{1}^\top) \\ \text{s.t. } a_j^{(k)} \in \lambda\Delta^{d-1}, \forall k \end{aligned} \tag{5.5}$$

end while
Return $\mathbf{A}_{(t)}, \mathbf{W}_{(t)}$

5.1.4 Learned phenotypes and predictive analyses

The estimated phenotypes are evaluated on various metrics. We denote the model learned using Algorithm 3 with a given parameter $\lambda > 0$ as λ–CNMF. The following baselines are used for comparison:

1. **Labeled LDA (LLDA):** LLDA (Ramage et al., 2009) is the supervised counterpart of LDA, a probabilistic model to estimate topic distribution of a corpus. It assumes that word counts of documents arise from multinomial distributions. It incorporates supervision on topics contained in a document and can be naturally adapted for phenotyping from bag-of-words clinical features, where the topic–word distributions form candidate phenotypes. While LLDA assumes that the topic loadings of a document lie on the probability simplex Δ^{K-1}, λ–CNMF allows each patient–condition w_{kj} loading to lie in $[0, 1]$. In interpreting the patient loading as a disease risk, the latter allows patients to have varying levels of disease prevalence. Also, LLDA can induce sparsity only indirectly via a hyperparameter β of the informative prior on the

topic–word distributions. While this does not guarantee sparse estimates, we obtain reasonable sparsity on LLDA estimates. We use the Gibbs sampling code from MALLET (McCallum, 2002) for inference. For a fair comparison to CNMF which uses an extra bias factor, we allow LLDA to model an extra topic shared by all documents in the corpus.

2. **NMF with support constraints (NMF+support)**: This NMF model incorporates non–negativity and support constraints from weak supervision but not the sparsity inducing constraints on the phenotype matrix. This allows to study the effect of sparsity inducing constraints for interpretability. On the other hand, imposing sparsity without our grounding technique does not yield identifiable topics and hence is not studied as a baseline.

3. **Multi-label Classification (MLC)**: This baseline treats weak supervision (from ECI) as accurate labels in a fully supervised model. A sparsity inducing ℓ_1 regularized logistic regression classifier is learned for each condition independently. The learned weight vector for each condition k determines importance of clinical terms towards discriminating patients with condition k and are treated as candidate phenotypes for condition k.

The weak supervision does not account for the primary diagnosis for admission in the ICU population as the ECI ignores primary diagnoses at admission (Elixhauser et al., 1998). However, the learning algorithm can be easily modified to account for the primary diagnoses, if required by using a modified form of supervision or absorbing the effects in an additional additive term appended to the model. Nevertheless, the proposed model generates highly interpretable phenotypes for comorbidities. Finally, to mitigate the effect of local minima, whenever applicable, for each model, the corresponding algorithm was run with 5 random initializations and results providing the

lowest divergence were chosen for comparison.

5.1.4.1 Interpretability–accuracy trade–off

Sparsity of the latent factors is used as a proxy for interpretability of phenotypes. Sparsity is measured as the median of the number of non–zero entries in columns of the phenotype matrix $\mathbf{A}$ (lower is better). The λ parameter in λ–CNMF controls the sparsity by imposing scaled simplex constraints on $\mathbf{A}$. CNMF was trained on multiple λ in the range of 0.1 to 1. Stronger sparsity-inducing constraints results in worse fit to the cost function. This trade–off is indeed observed in all models (see A.1 for details). For all models, we pick estimates with lowest median sparsity while ensuring that the phenotype candidate for every condition is represented by at least 5 non-zero clinical terms.

5.1.4.2 Clinical relevance of phenotypes

We requested two clinicians to evaluate the candidate phenotypes based on the top 15 terms learned by each model. The ratings were requested on a scale of 1 (poor) to 4 (excellent). The experts were asked to rate based on whether *the terms are relevant towards the corresponding condition and whether the terms are jointly discriminative of the condition*. Figure 5.1 shows the summary of qualitative ratings obtained for all models. For each model, we show two columns (corresponding to two experts). The stacked bars show the histogram of the ratings for the models. Nearly 50% of the phenotypes learned from our model were rated 'good' or better by both annotators. In contrast, NMF with support constraints but *without* sparsity inducing constraints hardly learns clinically relevant phenotypes. The proposed model 0.4–CNMF

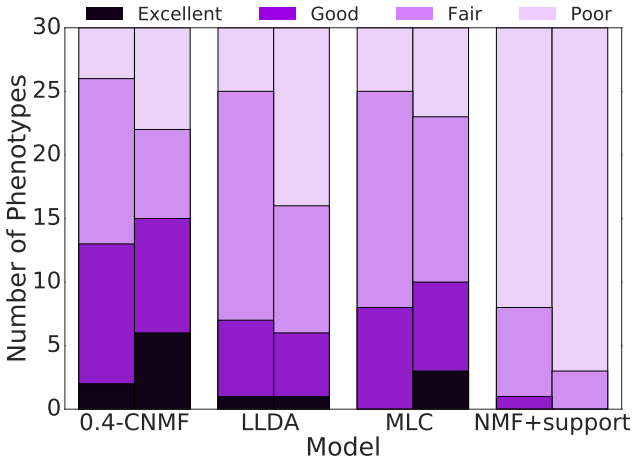


Figure 5.1: Qualitative Ratings from Annotation: The two bars represent the ratings provided by the two annotators. Each bar is a histogram of the scores for the 30 comorbidities sorted by scores.

	0.4–CNMF	LLDA	MLC	NMF
0.4–CNMF	**0**	**28**	**20**	**44**
LLDA	**7**	0	12	35
MLC	**6**	21	0	42
NMF+support	**1**	0	1	0

Table 5.2: Relative Rankings Matrix: Each row of the table is the number of times the model along the row was rated *strictly* better than the model along the column by clinical experts, e.g., column 3 in row 2 implies that LLDA was rated better than MLC 12 times over all conditions by all experts.

also received significantly higher number of 'excellent' and 'good' ratings from both experts. Although LLDA and MLC estimate sparse phenotypes, they are not at par with λ–CNMF. Table 5.2 shows a summary of relative rankings for all models. Each cell entry shows the number of times the model along the corresponding row was rated *strictly better* than that along the column. 0.4–CNMF is better than all three baselines. The supervised baseline MLC outperforms LLDA even though LLDA learns comorbidities jointly suggesting

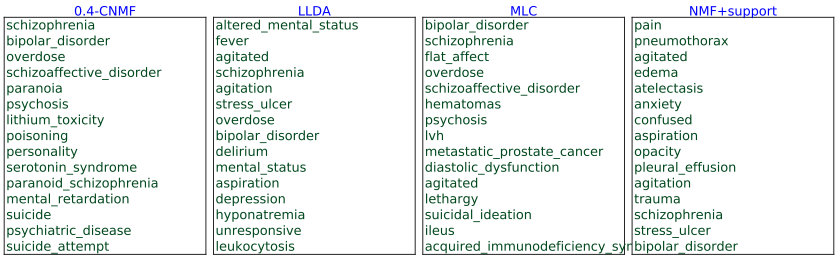


Figure 5.2: Phenotypes learned for 'Psychoses' (words are listed in order of importance)

that the simplex constraint imposed by LLDA may be restrictive.

Figure 5.2 is an example of a phenotype (top 15 terms) learned by all models for psychoses. For this condition, the proposed model was rated "excellent" and strictly better than both LLDA and MLC by both annotators while LLDA and MLC ratings were tied. However, the phenotype for Hypertension (in Figure 5.3) learned by 0.4–CNMF has more terms related to 'Renal Failure' or 'End Stage Renal Disease' rather than hypertension. One of our annotators pointed out that "Candidate 1 is a fairly good description of renal disease, which is an end organ complication of hypertension", where the anonymized Candidate 1 refers to 0.4–CNMF. Exploratory analysis suggests that hypertension and renal failure are the most commonly co-occurring set of conditions. Over 93% of patients that have hypertension (according to ECI) also suffer from Renal Failure. Thus, our model is unable to distinguish between highly co-occurring conditions. Other baselines were also rated poorly for hypertension, while LLDA was rated only slightly better. More examples of phenotypes are provided in A.2.

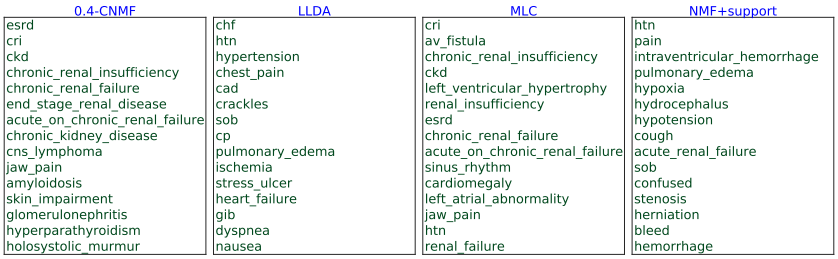


Figure 5.3: Phenotypes learned for 'Hypertension'

5.1.4.3 Mortality prediction

To quantitatively evaluate the utility of the learned phenotypes, we consider the 30 day mortality prediction task. We divide the EHR into 5 cross-validation folds of 80% training and 20% test patients. As this is an imbalanced class problem, the training–test splits are stratified by mortality labels. For each split, all models were applied on the training data to obtain phenotype candidates $\tilde{\mathbf{A}}$ and feature biases $\tilde{b}$. For each model, the patient loadings $\tilde{\mathbf{W}}$ along the respective phenotype space $(\tilde{\mathbf{A}}, \tilde{b})$ are used as features to train a logistic regression classifier for mortality prediction. For CNMF and NMF+support, these are obtained as $\mathbf{W}_{\text{train/test}} = \text{argmin}_{\mathbf{W}\in[0,1]^{K\times N}} \mathcal{D}(\tilde{\mathbf{A}}\mathbf{W} + \tilde{b}\mathbb{1}^\top, \mathbf{X}_{\text{train/test}})$ for fixed $(\tilde{\mathbf{A}}, \tilde{b})$. For LLDA, these are obtained using Gibbs sampling with fixed topic–word distributions. For MLC, the predicted class probabilities of the comorbidities are used as features. Additionally, we train a logistic regression classifier using the full EHR matrix as features.

We clarify the following points on the methodology: (1) $\tilde{\mathbf{A}}$ is learned on the patients in the training dataset only, hence there is no information leak from test patients into training. (2) Test patients' comorbidities from ECI are *not* used as support constraints on their loadings. (3) Regularized logistic regression classifiers are used to learn models for mortality prediction. The

	Model	AUROC	Sensitivity	Specificity
1.	0.4–CNMF	0.63(0.02)	0.59(0.04)	0.62(0.03)
2.	NMF+support	0.52(0.02)	0.56(0.13)	0.51(0.14)
3.	LLDA	0.64(0.02)	0.62(0.03)	0.61(0.05)
4.	MLC	0.66(0.01)	0.62(0.06)	0.62(0.05)
5.	**Full EHR**	**0.72(0.02)**	**0.69(0.02)**	**0.63(0.04)**
6.	**CNMF+Full EHR** (ℓ_1, $C = 0.1$)	**0.72(0.02)**	**0.61(0.09)**	**0.71(0.07)**

Table 5.3: 30 day mortality prediction: 5–fold cross-validation performance of logistic regression classifiers. Classifiers for 0.4–CNMF and competing baselines (NMF+support, LLDA, MLC) were trained on the 30 dimensional phenotype loadings as features. Full EHR denotes the baseline classifier (ℓ_1-regularized logistic regression) using full $\sim$ 3500 dimensional EHR as features. CNMF+Full EHR denotes the performance of the ℓ_1-regularized classifier learned on Full EHR augmented with CNMF features (hyperparameter was manually tuned to match performance of the Full EHR model).

regularization parameters are chosen via grid-search.

The performance of the above baselines trained on ℓ_2 regularized logistic regression over a 5-fold cross-validation is reported in Table 5.3: rows 1–5. The classifier trained on the full EHR unsurprisingly outperforms all baselines as it uses richer high dimensional information. All phenotyping baselines, except NMF+support, show comparable performance on mortality prediction which in spite of learning on a small number of 30 features, is only slightly worse than predictive performance of full EHR with $\sim$ 3500 features.

Augmented features for mortality prediction (CNMF+Full EHR) Unsurprisingly, Table 5.3 suggests that the high dimensional EHR data has additional information towards mortality prediction which are lacking in the 30 dimensional features generated via phenotyping. To evaluate whether this additional information can be captured by CNMF if augmented with a small number of raw EHR features, we train a mortality prediction classifier using

ℓ_1 regularized logistic regression on CNMF features/loadings combined with raw bag–of–words features, with parameters tuned to match the performance of the full EHR model. The results are reported in the final row of Table 5.3.

In exploring the weights learned by the classifier for all features, we observe that only 8.3% of the features corresponding to raw EHR based *bag-of-words* features have non–zero weights. This suggests that comorbidities capture significant amount of predictive information on mortality and achieve comparable performance to full EHR model with a small number of additional terms. See Figure A.31 in Appendix showing the weights learned by the classifier for all features. Figure 5.4 shows comorbidities and EHR terms with top magnitude weights learned by the CNMF+full EHR classifier. For example, it is interesting to note that the top weighted EHR term – dnr or 'Do Not Resuscitate' is not indicative of any comorbidity but is predictive of patient mortality.

5.2 *Grounded* Admixtures of PMRFs

This work introduces Labeled Admixtures of Poisson Markov Random Fields (*Labeled APM*), which is inspired by APM but is able to incorporate topic-level supervision via *grounding*. To this end, we impose a one-to-one mapping between chronic conditions and topics to encourage the model to estimate parameters in the context of comorbidities. The model is therefore able to jointly predict presence or absence of the set of chronic conditions in a patient given the estimated parameters. We compare the proposed method to current state-of-the-art methods, namely *Labeled LDA* and multilabel SVMs to demonstrate that modeling correlations between terms given each chronic conditions allows better representation of documents leading to better predic-

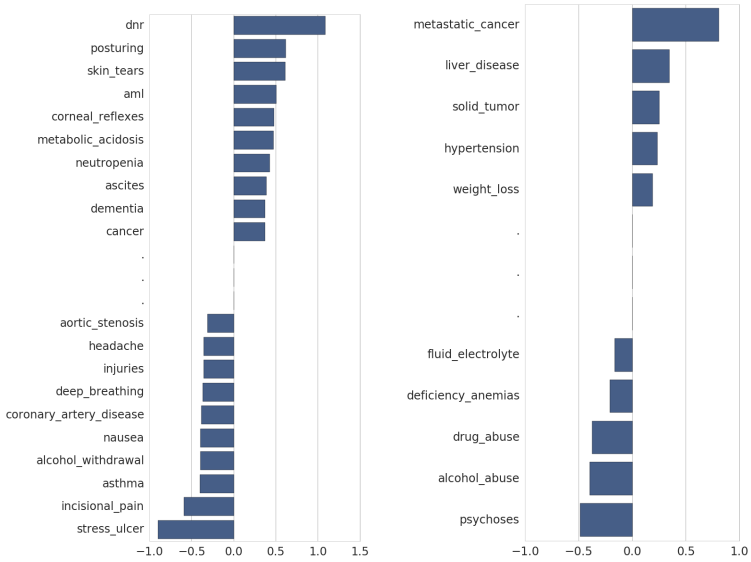


Figure 5.4: Top magnitude weights on (a) EHR and (b) CNMF features in CNMF+Full EHR classifier

tion of chronic conditions. We also discuss trade-offs between computational costs, interpretability and predictive abilities of all methods in context of usefulness to healthcare providers. Our main contributions can be summarized as follows:

- We propose a new method (called *Labeled* APM) that incorporates topic-level supervision via *grounding* into the Admixture of Poisson Markov Random Field (MRF) model.
- Labeled APM is used to model multiple chronic conditions among patients using clinical notes available in EHRs (see data-processing in 4.1.2.2).
- We analyze quantitative performance of *Labeled* APM compared to existing state-of-the-art baselines, namely, *Labeled* LDA and multilabel SVM for diagnosing chronic illnesses.

- We interpret results from *Labeled* APM to explore whether *Labeled* APM has particular advantages over *Labeled* LDA for analyzing clinical notes in the context of comorbidities. In particular, each chronic condition can be visualized as a graph over terms via APMs, while LDA will only provide a ranked list of the most probable terms or the most discriminative terms. Thus, the interpretation and visualization capabilities of *Labeled* LDA are limited. This feature yields a novel and powerful exploratory tool that can be used by clinicians to further understand the interaction of different terms associated with each comorbidity.

Recall that the topics represent chronic conditions and the training data includes information about whether a patient is diagnosed with a given chronic condition or not. Thus for each patient's note or document n, our MIMIC-II database includes a vector $C_n \in \{0,1\}^K$, such that its k^{th} element is 1, if comorbidity k is diagnosed in the patient and 0 otherwise. Our model assumes that a patient's clinical notes are composed only from the topics corresponding to the conditions he/she is diagnosed with. However, the degree of severity of disease is unknown. In other words, the contribution of the topic towards generating the clinical note is unknown.

In order to add topic-level supervision to APM, we define a support set $\mathcal{S}_n \, \forall \, n \in \{1, 2, ..., N\}$. Let K be the total number of topics for which supervision is available. This is equal to the total number of comorbidity conditions that have been diagnosed for all patients whose clinical notes are to be analyzed. Let $\mathcal{S}_n$ be the set of all comorbidities that patient n is diagnosed with i.e. the set of all indices of vector C_n that are 1.

If for a given sample n, condition/disease $k \notin \mathcal{S}_n$, we fix the corresponding weight to 0. Let $w_{\mathcal{S}_n}$ be the subvector of w_n of all indices in $\mathcal{S}_n$. Thus

$w_{\mathcal{S}_n,k} >= 0 \, \forall \, k \in \mathcal{S}_n$. This subvector lies on a simplex of dimension $|\mathcal{S}_n| - 1$. Thus, the dual coordinate descent of Inouye et al. (2014b) can be modified to update only the subvector $w_{\mathcal{S}_n}$ subject to simplex constraints. Equation (2.8) remains convex in the admixing weights for fixed estimates of parameters, as before, guaranteeing convergence to local minima of the complete MAP estimation problem even after incorporating supervision. It is important to note the Equations (2.7) and (2.8) are separately convex in $\mathbf{\Phi}_v$ and $w_1, ..., w_n$ respectively. However, the overall MAP estimation problem is not jointly convex in $\mathbf{\Phi}_v$ and the admixing weights. Thus only convergence to local minima can be guaranteed.

5.2.1 Inference in PMRFs for comorbidity prognosis

Let θ_k^* and $\mathbf{\Theta}_k^*$, for all $k \in \{1, 2, ..., K\}$ topics, be the set of learned parameters once the above learning procedure converges. Then for any new test document x_{test}, the existence of comorbidities can be predicted by solving Equation (2.8), with the rest of the parameters fixed to θ_k^* and $\mathbf{\Theta}_k^*$. Note that for prediction, no supervision is available, hence (2.8) is computed as in standard APM. The resulting weight vector can be thresholded by a parameter δ^* s.t. $w_{test,k} = 1$ if $w_{test,k} > \delta^*$ else $w_{test,k} = 0$. The threshold δ^* can be learned via cross-validation.

5.2.2 Empirical evaluation

The empirical evaluation is designed to determine how well the latent representations captured using *grounding* can classify whether or not a patient has any of the target chronic conditions on a set of held-out ICU patient population. The following presents these results in comparison to important

Table 5.4: Average F1-scores for Chronic Disease Prediction on MIMIC-II

Model	Micro-F1	Micro-Precision	Micro-Recall	Instance-Averaged F1	Instance-Avg. Precision	Instance-Avg. Recall
Labeled APM	**0.2972 (0.0060)**	**0.4361 (0.0059)**	0.3763 (0.0062)	**0.2792** (0.0057)	**0.3292 (0.0034)**	0.2993 (0.0093)
Labeled LDA	0.2008 (0.0033)	0.3317 (0.0020)	**1.0 (0.0080)**	0.1545 (0.0025)	**0.3292 (0.0034)**	0.2818 (0.0118)
ML-SVM	0.2792 (0.0057)	0.1843 (0.0040)	0.3025 (0.0087)	0.2063 (0.0047)	0.1819 (0.0041)	**0.3067 (0.0085)**

Table 5.5: Average F1-scores on low risk patients from MIMIC-II

Model	Micro-F1	Micro-Precision	Micro-Recall	Instance-Averaged F1	Instance-Avg Precision	Instance-Avg Recall
Labeled APM	**0.2925 (0.0135)**	**0.3129 (0.0157)**	0.2748 (0.0133)	**0.2765 (0.0109)**	**0.3288 (0.0168)**	0.2927 (0.0126)
Labeled LDA	0.2078 (0.0056)	0.1647 (0.0049)	0.2815 (0.0071)	0.1903 (0.0050)	0.1734 (0.0038)	0.2835 (0.0126)
ML-SVM	0.2341 (0.0106)	0.1895 (0.0099)	**0.3067 (0.0142)**	0.2067 (0.0087)	0.1835 (0.0089)	**0.3096 (0.0159)**

Table 5.6: Average F1-scores on high risk patients from MIMIC-II

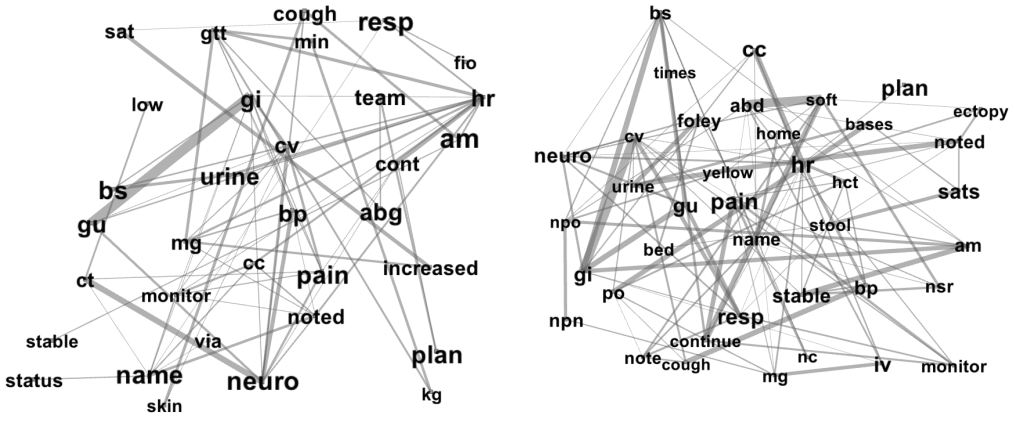
Model	Micro-F1	Micro-Precision	Micro-Recall	Instance-Averaged F1	Instance-Avg Precision	Instance-Avg Recall
Labeled APM	**0.3295 (0.0158)**	**0.3349 (0.0116)**	**0.3244 (0.0204)**	**0.3088 (0.0212)**	**0.3489 (0.0230)**	**0.3309 (0.0246)**
Labeled LDA	0.1966 (0.0193)	0.1554 (0.0162)	0.2685 (0.0282)	0.1832 (0.0196)	0.1596 (0.0166)	0.2903 (0.0398)
ML-SVM	0.2255 (0.0073)	0.1767 (0.0053)	0.3121 (0.0161)	0.1972 (0.0152)	0.1718 (0.0092)	0.2986 (0.0333)

baselines at different levels of patient risk.

We evaluated model performance using 5-fold cross-validation to test for stability of the results across variations in the data. The data was split the data into training (80%) and test (20 %) sets in each fold. Primarily, two parameters (λ, δ^*) need to be set. The parameter λ regulates the sparsity of θ_k and $\mathbf{\Theta}_k \forall k \in [K]$ using the ℓ_1 regularization of the vectorized form of Φ_v (see Equation (2.7)) and the threshold δ^* is used to obtain a hard cluster membership. We predict that the patient n suffers from condition k if $w_{n,k} > \delta^*$. The regularization for the correlation like matrix in each topic PMRF is fixed to $\lambda = 0.0001$ based on preliminary experiments. The weight vectors are initialized uniformly over the support set provided by the supervision. The best threshold parameter δ^* for *Labeled* LDA $\in [0, 1]$ is obtained as the δ that provides the best Micro-F1 measure on the test set, and we fixed $\delta^* = 0$ for *Labeled* APM. The objective of multiple chronic disease prognosis is to obtain best predictive performance for a given patient. In particular, a model with better instance level predictions demonstrates better predictive abilities for

a new patient. Thus for chronic disease prognosis, instance based decisions and the corresponding metrics (Instance-Averaged F1, Precision and Recall) are the most clinically relevant. While less clinically relevant, we also provide the Micro-F1 score (Lewis et al., 2004) for completeness and comparison with the baselines. Micro-F1 scores measures the overall performance of the model averaged across patients and diseases. Note that precision and recall have also been provided in each case for completeness but should not interpreted independently. This is because there is a fundamental trade-off between precision and recall, and each can be separately tuned at the cost of another.

The resulting performance for each method is demonstrated in Table 5.4. The performance shows that the proposed model outperforms the baselines approaches in terms of both Micro-F1 scores and Instance-Averaged F1. The Instance-Averaged F1 performance suggests that the per patient disease prediction performance is better than the baselines. Thus the model is the best predictor of potential chronic diseases for a new patient among all compared. In addition, the trends are consistent for low risk and high risk patients as shown in Table 5.5 and Table 5.6 respectively. The Micro-F1 score and Instance-Averaged F1 are significantly better using *Labeled* APM suggesting the model is overall better in terms of per-patient performance especially if the patient is significantly at risk. The proposed model performs better on Precision and Recall for high risk patients compared to both baselines suggesting the proposed model's efficacy in identifying a high risk patients. If any condition is currently undiagnosed by physicians, necessary steps can be undertaken to verify the prognosis obtained using *Labeled* APM. It is also imperative that unnecessary health-care costs are prevented for the healthcare facility and the patient. Thus a better precision per patient, as obtained using



(a) Drug Abuse (b) Solid (Brain) Tumor

Figure 5.5: Graph visualization of chronic conditions learned by the *Labeled* APM model

Labeled APM is useful for providing guidance towards avoiding unnecessary expenditure.

An additional advantage of the proposed model is the ability to explore most relevant symptoms, treatments, medications associated with each disease. Although *Labeled* LDA can also provide the most relevant term per disease, *Labeled* APM has an added advantage of visualizing the relationship among the most relevant terms. This provides a previously unavailable exploratory tool to clinical experts for discovering new associations or the most relevant day-to-day behavior of patients likely for each chronic condition. We explore a sample of these visualizations for a few diseases in the following.

The graphs estimated by *Labeled* APM were plotted using gephi[1], then analyzed by a medical expert to investigate the correlations learned by the model between the few terms with highest marginal probability. In each of

[1]http://gephi.github.io/

the graphs, the relative size of each node/term reflects the importance of the term for the corresponding topic. This is based on the estimates of θ_k. A higher value of the i^{th} element of θ_k compared to any other word implies that the i^{th} term in the vocabulary is more relevant to topic k than other terms in the vocabulary. In addition, the strength of any off-diagonal elements Θ_k, say the $\{i, j\}^{th}$ element models the correlation between the i^{th} and j^{th} terms from the vocabulary in topic k. The edges in the graphs between the words denotes this correlation. In particular, the thickness of the connections between nodes reflects the strength of the positive correlations learned between the terms. Some preliminary interpretation based on this investigation is presented.

- *Drug Abuse*: According to (Kowalchuk and Reed, 2011; Volkow, 2014; MM, 2012; RD, 2012)[2], substance use disorder occurs when a person needs alcohol or another substance (drug) to function normally. Abruptly stopping the substance leads to withdrawal symptoms. Certain drugs cause increase or decrease in blood pressure and affect neurological status i.e. alters mental state so patient is confused, belligerent or may even be comatose. A decrease in respiratory drive may be observed affecting the pulmonary system. Blood gases (abg's) are often checked as part of diagnosis. Decreased respiratory drive may cause respiratory failure thus requiring intubation and medication (often propofol). Certain drugs also affect skin integrity. Fig. 5.5(a) shows that the recovered graph captures many of the relationships between these terms. In addition, the graph recovers correlations like 'skin' and 'dry' which may be associated with drug abuse and do not occur in other disease graphs. Other disease spe-

[2]http://www.nlm.nih.gov/medlineplus/ency/article/001522.htm

cific words such as 'rehab', 'propofol', 'pulm' are among the few terms observed in the graph.

- *Solid Brain Tumor*: According to (bra, 2014), a brain tumor is a growth of abnormal cells in the tissues of the brain. A tumor in the brain can affect (neurological) status and, depending on the location of the tumor, (respiratory and cardiovascular) status may decline. Signs and symptoms of respiratory decline include decreased oxygen saturation in the blood (sat), thus affecting perfusion and distribution of hemoglobin (hgb) and hematocrit (hct). Brain stem tumors may also alter the functions of the autonomic centers of the brain stem, e.g. auto regulation of the cardiovascular system - resulting in abnormal heart rhythms (ectopy) and effects on blood pressure (BP). Cardiac monitors are utilized to monitor these symptoms. Fig. 5.5(b) shows that the recovered graph captures many of the relationships between these terms.

- *Renal failure*: According to (Ren, 2015), kidney failure, also known as renal failure, is a term used to describe a situation in which the kidneys are no longer able to function effectively, to maintain proper fluid balance in the body, remove waste and eliminate toxins from the blood. Kidney dysfunction can affect the neurological (neuro), pulmonary (resp), cardiovascular (cv) and gastrointestinal (gi) systems of a patient. Due to the kidney's diminished ability to remove toxins, renal failure may predispose patients to infectious diseases (id), pain, respiratory failure and electrolyte (lyte) imbalances. Standard practice is for a nurse to routinely monitor electrolytes and (due to fluid imbalances) lasix is given to the patient to assist with urination and decrease hypertension (htn)

or blood pressure (bp). These terms are among the most frequently occurring as can be seen from Fig. A.11.

5.3 Conclusion

This chapter presents two constrained based latent variable algorithms to learn interpretable models. We focus on phenotyping chronic conditions for an ICU patient population. Specifically, we study two phenotype representations, each of which can be learned using specific latent variable models. In both cases, the grounding framework, introduced in Chapter 3 is demonstrated to be effective in generating clinically relevant (and therefore interpretable) phenotypes. Specifically, all methods were evaluated by clinicians and compared to existing phenotyping baselines. Further, the effectiveness of the phenotypes is used for personalized outcome prediction of the patient population. In each case, the proposed method has been found to be comparable or better at predicting patient mortality compared to the baselines. The results demonstrate that weak supervision can be effectively leveraged as constraints to develop models that generate interpretable outcomes. Our proposed framework as well as the training and inference mechanisms can be generalized to other applications where weak supervision can be leveraged to constrain latent variable models for interpretability.

Chapter 6

Explainability using Manifold Constrained Examples

This chapter proposes a tool designed to explain outcomes of supervised black-box models. We generate explanations using examples and their summary statistics. Demonstrating model behavior via examples is known to be beneficial for improving and understanding the decision making process (Aamodt and Plaza, 1994). The examples are generated in a constrained manner to allow for insightful explanations. Specifically, we assume that the data lies within a lower dimensional manifold in a high dimensional ambient space. We design a mechanism to perturb existing data points by constraining the perturbations along the data manifold as well as in a manner that is most likely to change the decision of the black–box. Parts of the algorithm described here have appeared in Joshi et al. (2018b).

To do so, we learn an approximate manifold of the data distribution using recent advances in implicit generative models (Kingma and Welling, 2013; Goodfellow et al., 2014a). Thus we can explore model behavior in the range of this generator function. We note that this generative model is learned in an unsupervised way assuming access to the training samples (without label in-

This chapter is based on content published in Joshi et al. (2018b). The author of the dissertation contributed to model formulation, implementation and experimental evaluation described in this chapter.

formation) used to train the target black-box classifier. The proposed method can be utilized as a diagnostic tool to analyze training progression, compare classifier performance, and/or uncover inherent biases the classifier may have learned.

6.1 Related Work

Most closely related works to our approach are those that provide explanations by sub-selecting meaningful samples and/or semantically relevant features (like super-pixels) that highlight undesirable model behavior (Elenberg et al., 2017; Kim et al., 2016). Most of these methods require the selected samples to be part of training/test dataset. This means that if the training/test set did not include the instance that best explains a specific decision, we would have to settle for a suboptimal choice. Our method aims to relax this constraint by generating new examples that are better suited for this purpose. In terms of generating examples, adversarial criticisms (Stock and Cisse, 2017) and the class of generative networks like GANs are relevant approaches. Specifically, (Stock and Cisse, 2017) use the adversarial attack paradigm as a means to select examples from existing training data to explain model behavior, similar to Kim et al. (2016). However note that the goal of generating adversarial examples and our explanations are fundamentally different. The primary goal of adversarial examples is to focus on exploiting the worst case confounding scenario given a decision boundary, while our work focuses on generating an example that lies on the data manifold as it crosses a decision boundary. See Figure 6.1 for a more intuitive explanation. We posit that it is important to uncover classifier behavior when data points are constrained to the data manifold. Such data instances are more 'realistic' and

likely to be created by the underlying phenomenon that led to the training data. They provide an alternative method to probe a black-box, specially in non-adversarial settings. They also characterize the residual vulnerabilities of a model that defends itself against adversarial attacks by detecting directed "noise" that is orthogonal to the manifold of the data or of an associated latent space.

We position our work as a diagnostic framework for understanding model behavior at an abstraction that may be most useful to a data science practitioner and/or a machine learning expert. However, as suggested before, explainable models focus on different notions of explainability. For example, Koh and Liang (2017) use influence functions, motivated by robust statistics Cook and Weisberg (1980) to determine importance of each training sample for model predictions. Li et al. (2015); Selvaraju et al. (2016) focus on understanding the workings of different layers of a deep network and studying saliency maps for feature attribution (Simonyan et al., 2013; Smilkov et al., 2017; Sundararajan et al., 2017). Saliency methods, while powerful, can be demonstrated to be unreliable without stronger conditions over the saliency model (Kindermans et al., 2017; Adebayo et al., 2018). Other paradigms of explainable models focus on locally approximating complex models using a simpler functional form to approximate the (local) decision boundary. For instance, LIME based approaches (Ribeiro et al., 2016; Shrikumar et al., 2016; Bach et al., 2015) locally approximate complex models with linear fits. Decision Trees are also considered more explainable if they are not too large. These approaches inherently assume a trade off between model performance and explanability, as less complex model classes tend to be empirically subpar in performance relative to the success of the target black-box models they

endeavor to explain. The proposed framework, however, does not rely on local approximations to provide explanations or assume such a trade-off.

We summarize our key contributions as follows: 1. We introduce xGEMs, a framework for explaining supervised black-box models via examples constrained along the underlying data manifold. 2. We demonstrate the utility of xGEMS in (a) detecting confounding bias in learned models, (b) characterizing the probabilistic decision manifold w.r.t. examples, and (c) facilitating model comparison beyond standard performance metrics.

6.2 Additional Notation

Implicit Generative Models can be described as stochastic procedures that generate samples (denoted by the random variable $\mathbf{x} \in \mathbb{X}^d$) from the data distribution $p(\mathbf{x})$ without explicitly parameterizing $p(\mathbf{x})$. Assume an underlying latent space $\mathbf{z} \in \mathbb{R}^k$ that is mapped to the ambient data domain $\mathbf{x} \in \mathbb{R}^d$ using a deterministic function $\mathcal{G}_\theta$ parametrized by θ, usually as a deep neural network. The primary difference between GANs and VAEs is the training mechanism employed to learn function $\mathcal{G}_\theta$. GANs employ an adversarial framework by employing a discriminator that tries to classify generated samples from the deterministic function versus original samples and VAEs maximize an approximation to the data likelihood. The approximation thus obtained has an encoder-decoder structure of conventional autoencoders (Doersch, 2016). One can obtain a latent representation of any data sample within the latent embedding using the trained encoder network. While GANs do not train an associated encoder, recent advances in adversarially learned inference like BiGANs (Dumoulin et al., 2016; Donahue et al., 2016) can be utilized to obtain the latent embedding. In this work, we assume access to a generative

model such as a GAN or a VAE that allows us to obtain the latent embedding of a data point.

Let $\mathcal{F}_\psi : \mathbb{R}^d \to \mathbb{R}^k$ (parametrized by ψ) be the inverse mapping function that provides the latent representation for a given data sample. Let $\mathcal{L} : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}_+$ be the analogous loss function such that for a given data sample $\tilde{\mathbf{x}}$:

$$\tilde{\mathbf{z}} = \arg\min_{\mathbf{z}} \mathcal{L}(\tilde{\mathbf{x}}, \mathcal{G}_\theta(\mathbf{z})) \triangleq \mathcal{F}_\psi(\tilde{\mathbf{x}}) \tag{6.1}$$

Examples of $\mathcal{F}_\psi$ are the encoder in a VAE, or an inference network in a BiGAN. An appropriate distance function in the data domain can be used as the loss $\mathcal{L}$.

Without loss of generality, we assume that we would like to provide explanations for a binary classifier. Let $y \in \{-1, 1\}$ be the target label. Let $f_\phi : \mathbb{R}^d \to \{-1, 1\}$ be the target black-box classifier to be 'explained' and $\ell(f_\phi(\mathbf{x}), y)$ be the loss function used to train the black-box classifier.

Adversarial criticisms Adversarial criticisms to explain black-box classifiers look for perturbations $\delta_{\mathbf{x}}$ to data samples $\mathbf{x}$ such that the perturbations maximize the loss $\ell(f_\phi(\mathbf{x}+\delta_{\mathbf{x}}), y)$ or change the predicted label. These perturbations are invisible to the human eye. That is, if $\tilde{\mathbf{x}}$ is the target adversarial sample, an adversarial attack solves a Taylor approximation to the following:

$$\tilde{\mathbf{x}} = \arg\max_{\tilde{\mathbf{x}}:\|\tilde{\mathbf{x}}-\mathbf{x}\|_p<\epsilon} \ell(f_\phi(\tilde{\mathbf{x}}), y) \tag{6.2}$$

6.3 Generating xGEMs

To provide explanations via examples over more *naturalistic* perturbations, we introduce a new set of examples, called *manifold constrained examples*

Algorithm 4 **Find $(\mathbf{x}^*, y^*)$–xGEM**
Input: $(\mathbf{x}^*, y^*) \in \mathbb{R}^d \times \{-1, 1\}, y_{tar}, \mathcal{G}_\theta, \mathcal{F}_\psi, f_\phi, \lambda, \eta > 0$

Initialize $\mathbf{z} = \mathcal{F}_\psi(\mathbf{x}^*)$
while Not converged **do**
  $\tilde{\mathbf{z}} \leftarrow \tilde{\mathbf{z}} + \eta \nabla_{\tilde{\mathbf{z}}} (\mathcal{L}(\mathbf{x}^*, \mathcal{G}_\theta(\mathbf{z})) + \lambda \ell(f_\phi(\mathcal{G}_\theta(\mathbf{z})), y_{tar}))$
end while
$\tilde{\mathbf{x}} = \mathcal{G}_\theta(\tilde{\mathbf{z}})$
Return $\tilde{\mathbf{x}}$

or xGEMs. First, we train an implicit generative model $\mathcal{G}_\theta$ and an encoder network $\mathcal{F}_\psi$.

$$\tilde{\mathbf{x}} = \mathcal{G}_\theta(\arg\min_{\mathbf{z} \in \mathbb{R}^k} \mathcal{L}(\mathbf{x}^*, \mathcal{G}_\theta(\mathbf{z})) + \lambda \ell(f_\phi(\mathcal{G}_\theta(\mathbf{z})), y_{tar})) \tag{6.3}$$

A manifold constrained example is defined w.r.t. a given data sample $\mathbf{x}^*$.

Definition 6.3.1 ($\mathbf{x}^*, y^*$**-xGEM**)**.** An xGEM corresponding to a data point $(\mathbf{x}^*, y^*)$ and a target label $y_{tar} \neq y^*$, refers to the solution of Equation (6.3) for a fixed and known $\lambda > 0$. The xGEM is denoted by $\tilde{\mathbf{x}}$.

We propose Algorithm 4 to estimate a manifold constrained exampleor xGEM for any data point $\mathbf{x}^*$. Intuitively, for a point $\mathbf{x}^*$, we first determine its latent representation using $\mathcal{F}_\psi$. This allows us to explain model behavior from a common latent representation across all black-boxes. Then we look for the closest point to $\mathbf{x}^*$ along the data manifold that changes the outcome of the classifier. To do so, we take gradient steps along the latent space of the generator $\mathcal{G}_\theta$ (our proxy for the data manifold) until the label switches to the desired target label y_{tar}. That is we take the shortest path along the data manifold to change the decision outcome of a given data point and analyze the corresponding perturbed sample to provide explanations. The desired *manifold constrained example* or xGEM is the sample generated at the switch point

in the latent embedding. We empirically highlight the benefits of the discovering manifold constrained examples in different contexts and abstractions that provide insights into model behavior.

6.4 Explanations using xGEMs

We first use a simple setting with simulated data to highlight the differences between the proposed explanation tool compared to criticisms and prototypes derived from adversarial attacks (Stock and Cisse, 2017).

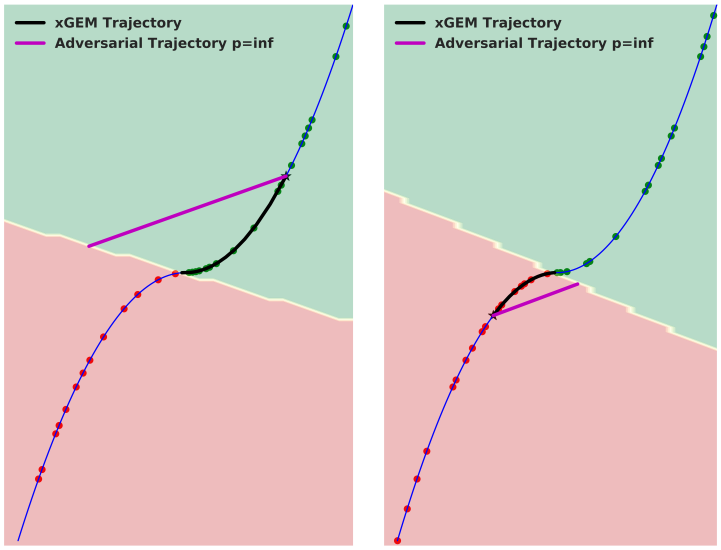


Figure 6.1: xGEMs versus *Adversarial* criticisms (Stock and Cisse, 2017), for a parabolic manifold (shown in blue). Green points belong to class 1 and red points to class -1. The black trajectories in all figures are gradient steps taken by Algorithm 4 while the magenta trajectories correspond to adversarial trajectories determined by Equation 6.2 with $p = \infty$. Note that all decision boundaries in Figures (a) and (b) separate the data. The decision boundary is trained by optimizing a softmax regression using the cross-entropy loss function.

6.4.1 An alternative view to adversarial criticisms

Figure 6.1 demonstrates a linear decision boundary trained on data with ambient dimension equal to 2. The one-dimensional data manifold is parabolic as shown by the blue curve. The green points are in class 1 and red points are samples belonging to class label -1. The figure illustrates manifold constrained examples as well as the trajectory taken by the gradient steps of Algorithm 4. The trajectory to generate an adversarial criticism stems from Equation (6.2). A generative model maps from a 1d latent dimension to the data manifold shown by the blue curve. A single layer (softmax) neural network with output dimension=2 is trained on points sampled from this manifold (the yellow decision boundary separates the two classes – regions marked by the pink and green regions). As demonstrated by the figure, navigating along the latent dimension of the generator encourages the xGEM trajectory to be constrained along the data manifold, while adversarial criticisms may lie well outside the manifold. Thus *manifold constrained examples* offer alternative view of classifier behavior via examples. We defer examples of xGEM evaluated for the MNIST dataset to the Appendix in the interest of space.

6.4.2 Towards attribute confounding detection

We demonstrate the utility of generating manifold constrained examples to detect if a target classifier is confounded w.r.t. a given attribute of interest. In particular, we wish to determine whether a black-box is differentiating among the target labels using spurious correlations in the data. For instance, a classifier trained to determine a gender neutral label like hair color may be inadvertently relying on an attribute like gender to predict the label. It is desirable to have an automated mechanism to detect such behavior. We use

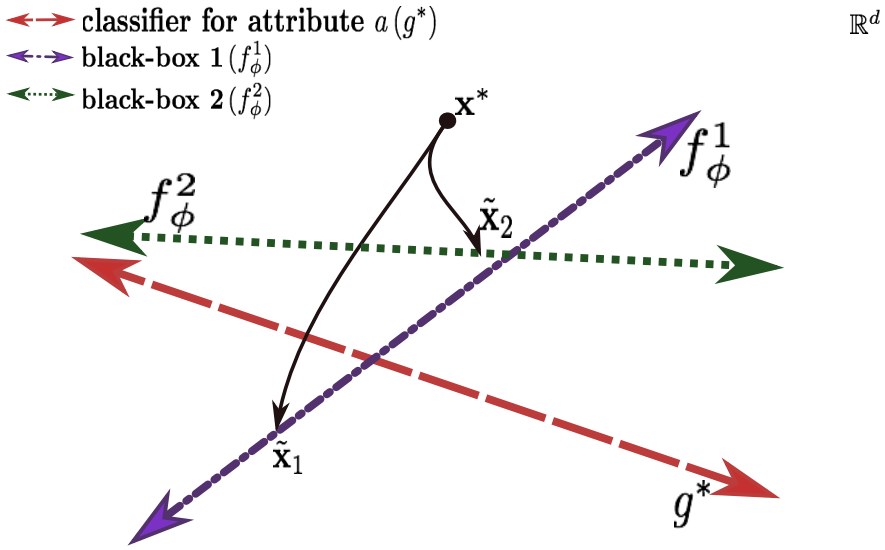


Figure 6.2: Example of bias detection. Target black-boxes:f_ϕ^1 and f_ϕ^2. g^* classifies points w.r.t. a. $\tilde{\mathbf{x}}_1$ and $\tilde{\mathbf{x}}_2$ are xGEMs corresponding to $\mathbf{x}^*$ for f_ϕ^1 and f_ϕ^2 resp. $\tilde{\mathbf{x}}_2$'s attribute prediction (w.r.t g^*) is the same as that of $\mathbf{x}^*$ while that of $\tilde{\mathbf{x}}_2$ is different. Thus we say that f_ϕ^1 is biased w.r.t. attribute a for sample $\mathbf{x}^*$.

the proposed framework to detect if a black-box is confounding an attribute of interest a with the target decision task. We demonstrate how to achieve this in a concrete manner within our framework below.

	Target black-box label	
Attribute (a) Classifier	Black Hair	Blond Hair
$\hat{g}$ (orig)	FP:0.003 FN:0.002 Acc: 0.997	FP:0.000 FN:0.018 Acc:0.999
$\hat{g}$ (recalibrated)	FP:0.003 FN:0.018 Acc:0.989	FP:0.003 FN:0.018 Acc:0.996

Table 6.1: Recalibrated Gender Classifier.

Without loss of generality let $a \in \{-1, 1\}$ be the (potentially protected) binary attribute of interest. We wish to examine whether the target classifier estimate $\hat{f}_\phi$ is biased/confounded by a. Intuitively, we hope that attribute a of an xGEM should be the same as that of the original point. In order to detect this, we assume there exists an oracle $g^* : \mathbb{R}^d \to \{-1, 1\}$ that perfectly classifies the confounding attribute a when considered as the dependent variable, based on the other (d) independent variables. Additionally, we assume that g^* is not confounded by the target label of the black-box y and is not used by g^* to predict a. Let $\mathbb{R}^d \times \{-1, 1\} \times \{-1, 1\} \supset \mathcal{D} \triangleq \{(\mathbf{x}_i, y_i, a_i), i \in [N]\}$ be the training data where i indexes a given point. Let $\tilde{\mathbf{x}}_i$ be the xGEM of $\mathbf{x}_i$ w.r.t. $\hat{f}_\phi$ as returned by Algorithm 4. We argue that classifier $\hat{f}_\phi$ is confounded by the attribute a if equation (6.4) holds for a given $\delta > 0$.

$$E_{\mathcal{D}}[\mathbb{1}(g^*(\tilde{\mathbf{x}}) \neq a)] > \delta \tag{6.4}$$

Black-box Classifier	Accuracy	Confounding metric
$\hat{f}_{\phi}^{1}$	0.9933	0.1704
$\hat{f}_{\phi}^{2}$	0.9155	0.4323

Table 6.2: Confounding metric

In practice, access to a perfect oracle g^* is infeasible or prohibitively expensive. In some cases, such a classifier may be provided by regulatory bodies, thereby adhering to predetermined criterion as to what accounts for a *reliable* proxy oracle. For this case study, we assume it is sufficient that the proxy oracle has the same false positive and false negative error rates w.r.t. the target label, which is a fairness condition known as the Equalized Odds Criterion (Hardt et al., 2016). To demonstrate our algorithm, we assume access to a proxy oracle $\hat{g} : \mathbb{R}^d \to \{-1, 1\}$ that satisfies the following conditions, given a $0.5 \ll \tau < 1$:
(i) $E_{\mathcal{D}}[(\mathbb{1}(\hat{g}) = a)] > \tau$ (ii) $\hat{g}$ satisfies the Equalized Odds (Hardt et al., 2016) criterion w.r.t. the target label y.

	Target label	
Black-box	Black Hair	Blond Hair
$\hat{f}_{\phi}^{1}$	Male:0.4550 Female:0.0159 Overall:0.2430	Male:0.1432 Female:0.0484 Overall:0.0539
$\hat{f}_{\phi}^{2}$	Male:0.7716 Female:0.0045 Overall:0.4012	Male:0.1475 Female:0.5024 Overall:0.4821

Table 6.3: Confounding metric by gender

Note that while we consider $\hat{g}$ as an inexpensive proxy for g^*, we prescribe that the experiment be carried out with g^*. Figure 6.2 demonstrates how such confounding could be detected, as well as used for model comparison w.r.t. their biases. As shown in the figure, $\hat{f}_\phi^1$ and $\hat{f}_\phi^2$ are the classification boundaries of two black-box models classifying a target label of interest. g^* is a classifier that classifies the data according to attribute a. Consider the sample $\mathbf{x}^*$ and let $\tilde{\mathbf{x}}_1$ and $\tilde{\mathbf{x}}_2$ be the manifold constrained examples of $\mathbf{x}^*$ corresponding to classifiers $\hat{f}_\phi^1$ and $\hat{f}_\phi^2$ respectively. As shown in the figure, the attribute a of the xGEM $\tilde{\mathbf{x}}_1$ is different from that of $\mathbf{x}^*$ while that of $\tilde{\mathbf{x}_2}$ is not. We conclude that a black-box $\hat{f}_\phi^1$ is confounded if the fraction of points whose manifold constrained examples or xGEMs that change attribute a is greater than δ. Thus an empirical estimate of Equation (6.4) gives a metric that can quantify the amount of confounding in a given black-box, while also allowing to compare different black-boxes w.r.t. the target attribute a.

We evaluate our framework for confounding detection in facial images using the CelebA (Liu et al., 2015) dataset. The target black-box classifier predicts the binary facial attribute – hair color (black or blond). We determine whether or not the black-box is confounded with gender. We restrict to two genders, male and female, based on annotations available in CelebA. In particular, $\hat{g}$ is a ResNet model (He et al., 2016)[1] that classifies celebA faces by gender . $\hat{g}$ is recalibrated to satisfy the two conditions mentioned earlier. Details of $\hat{g}$'s performance and recalibration are provided in Table 6.1.

Two ResNet models $\hat{f}_\phi^1$ and $\hat{f}_\phi^2$ are trained to detect the hair color attribute (black hair vs blond hair) using two different datasets. $\hat{f}_\phi^1$ is trained

[1] https://github.com/tensorflow/models/tree/master/tutorials/image/cifar10_estimator

on all face samples with either black or blond hair whereas $\hat{f}^2_\phi$ is trained such that all black hair samples are male while blond haired samples are all female. Table 6.2 gives the overall validation accuracy of both classifiers. Note that the validation set used for $\hat{f}^1_\phi$ and $\hat{f}^2_\phi$ are the same.

Table 6.2 also shows the fraction of samples whose manifold constrained examples' predicted attribute a (in this case gender) is different from the original training sample w.r.t. $\hat{g}$. The fraction of confounded samples is clearly much larger for the classifier trained on a biased dataset as determined by the proxy oracle $\hat{g}$. Additionally, Table 6.3 suggests a 10–fold increase in the fraction of confounding for blond haired females with the biased classifier $\hat{f}^2_\phi$. Notice the decrease in the amount of confounding for black haired females while a general increase in confounding for all black haired faces. As an aside, the biased model $\hat{f}^2_\phi$ also changes the background more than hair color in order to change the hair color label (see Figure 6.3). This suggests that quantifying such confounding using manifold constrained examplesallows us to characterize biases w.r.t. any attribute of interest.

Figure 6.3 shows a few examples of such confounded images for the two black-boxes. In particular, we show examples where the black-box trained on biased data for hair color classification changes gender of the sample as it crosses the decision boundary whereas the black-box trained on unbiased data does not[2].

Figure 6.3: We test whether ResNet models $\hat{f}_\phi^1$ and $\hat{f}_\phi^2$, both trained to detect hair color but on different data distributions are confounded with gender. Two samples for classifiers $\hat{f}_\phi^1$ (first sub row) and $\hat{f}_\phi^2$ (second sub row) are shown. The leftmost image is the original figure, followed by its reconstruction from the encoder F_ψ. Reconstructions are plotted as Algorithm 4 (with $\lambda = 0.01$) progresses toward crossing the decision boundary. The red bar indicates change in hair color label indicated at the top of each image along with the confidence of prediction. The label at the bottom indicates gender as predicted by $\hat{g}$. For both samples, classifier $\hat{f}_\phi^1$, trained on biased data changes the gender (1^{st} and 3^{rd} rows) while crossing the decision boundary whereas the other black-box does not.

6.4.3 Case Study: Model assessment

An important aspect of black-box analyses is to study the progression of training complex models. Specifically, observing manifold constrained examples allows us to consider model behavior in the following aspects: 1) Discerning shifts in features relied on by the black-box to differentiate between

[2]All qualitative figures were chosen based on the confidence of the prediction from the black-box and confidence of the reconstructed image

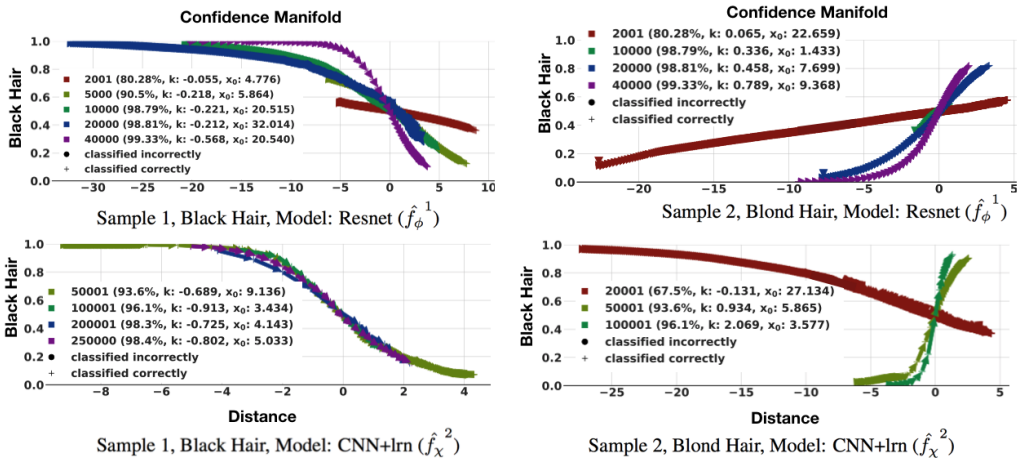


Figure 6.4: Confidence manifolds for a few data samples for black-box models 1 and 2. In each inset, this confidence manifold is traced during different stages of training the black-box. In each inset, the legends denote: **global training step (accuracy, parameter k, x_0)** denoting the global step at which the confidence manifolds are plotted, and their corresponding logistic curve estimates and the overall black-box accuracy at that stage of training. Additionally, the curve shows whether the sample is misclassified at that training step. The top left and bottom left inset denote curves for a single sample – Sample 1 for the first and the second black-box respectively at different training stages. The true label for Sample 1 is 'Black Hair'). The top right and bottom right curves show similar curves for black-box 1 and 2 respectively for Sample 2. The true label for Sample 2 is 'Blond Hair'.

classes during training. 2) Characterizing the probabilistic manifolds of manifold constrained examples as training progresses and its relation to calibration of complex networks (DeGroot and Fienberg, 1983). 3) Qualitative trade-offs and/or mistakes made by the classifier for prototypical examples.

Reliability Diagrams have been used as a summary statistic to evaluate model calibration (DeGroot and Fienberg, 1983) that aims to study whether the confidence of a prediction matches the ground truth likelihood of the prediction. It has been observed that while model performance has improved sub-

stantially in recent years because of deep networks, such models are typically more prone to mis-calibration (Guo et al., 2017). We provide a complementary statistic to Reliability Diagrams to assist model assessment/comparison.

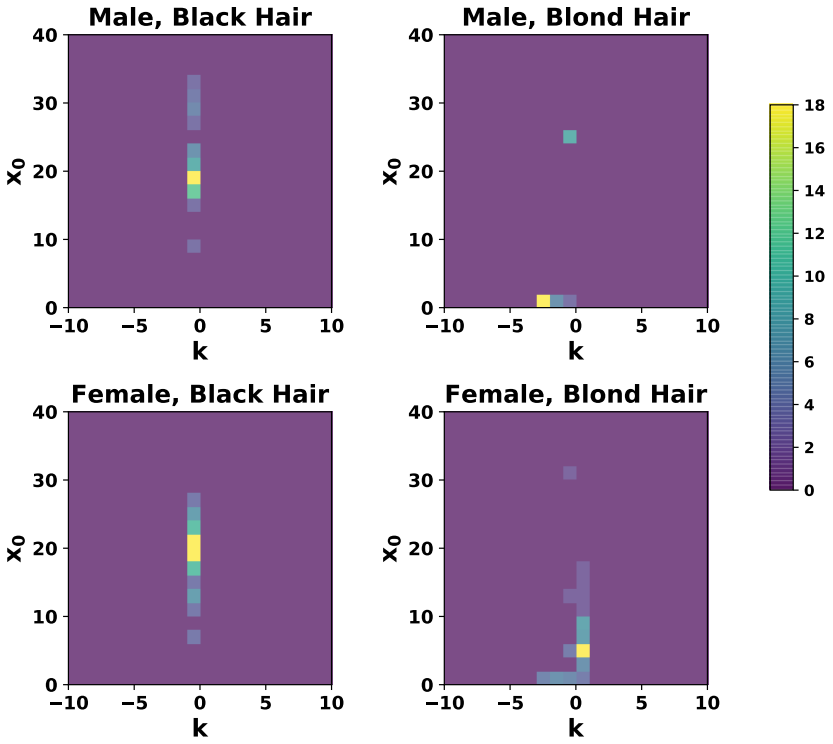
For this study we train two deep networks $\hat{f_\phi}^1$ (a ResNet model) and $\hat{f_\chi}^2$ (a four layer CNN with local response normalization (lrn) [3]) with CelebA face images for the hair color (black/blond) binary classification task. For a given face, we evaluate the corresponding xGEM at multiple incremental training steps. We plot the confidence of labeling a point to have black hair with respect to the distance of the original reconstruction and its xGEM including all intermediate points from the decision boundary (called 'confidence manifold'). Thus, all samples originally labeled black should have high confidence of being labeled and the confidence decreases as the sample crosses the decision boundary (vice-versa for blond haired faces). Figure 6.4 shows the confidence manifolds for two samples (one in each column).

The top and bottom rows represent the manifolds obtained during training for model 1 ($\hat{f_\phi}^1$) and model 2 ($\hat{f_\chi}^2$) respectively. Sample 1(column 1) is a face with black hair while Sample 2 (column 2) has blond hair. Legends show the distance of reconstructions from the original sample along the gradient steps, followed by overall classifier performance. Additionally, we fit a logistic function $f(x) = \frac{1}{1+\exp^{-k(x-x_0)}}$ to each curve. All curves have been aligned such that the decision boundary lies at 0 along the x-axis (denoting x_0) in Figure 6.4. Specifically, for a single sample, as the manifold is traversed to generate its corresponding xGEM, we estimate the classifier's confidence for the label 'Black Hair' and plot the entire probability curve (called confidence

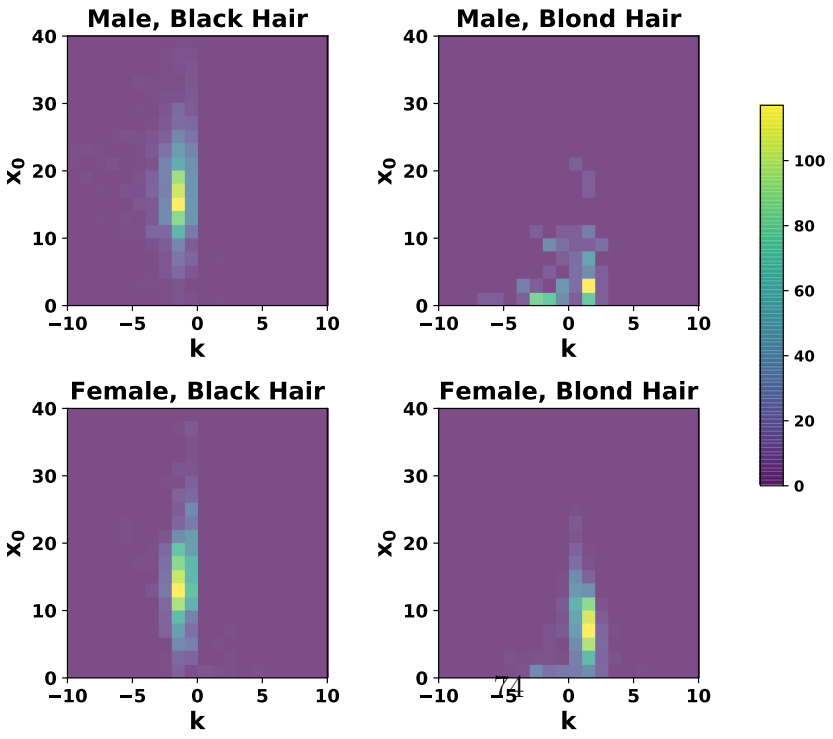
[3] https://github.com/tensorflow/models/tree/master/tutorials/image/cifar10

manifold). For each curve, a logistic curve is fit to estimate two parameters k and x_0 that specify the curve. These curves are shown in each subplot. In each inset, this confidence manifold is traced during different stages of training the black-box. In each inset of Figure 6.4, the legends denote: **global training step (accuracy, parameter k, x_0)** denoting the global step at which the confidence manifolds are plotted, and their corresponding logistic curve estimates and the overall black-box accuracy at that stage of training. Additionally, the curve shows whether the sample is misclassified at that training step. The top left and bottom left inset denote curves for a single sample – Sample 1 for the first and the second black-box respectively at different training stages. The true label for Sample 1 is 'Black Hair'). The top right and bottom right curves show similar curves for black-box 1 and 2 respectively for Sample 2. The true label for Sample 2 is 'Blond Hair'. The confidence manifold for the same instance is fairly different across each model. As expected, the overall steepness increases as model trains to better discriminate samples. Intuitively, higher x_0 suggests that the classifier can easily discriminate the label with high confidence. For instance, for comparable overall accuracy, the manifolds suggest that model 2 has trained a decision boundary such that a manifold constrained example is relatively close in image distance (compared to that of model 1). In the case of Sample 2, it is clear that model 2 mis-labels the data point with high confidence initially while learning to predict the correct label eventually. However, a decrease in x_0 as training progresses for both models suggests a significant shift of the decision boundary to be closer to Sample 2. Qualitative images corresponding to these manifolds are shown in the Appendix.

Figures 6.5(a) and 6.5(b) show the 2d histogram of the logistic function



(a) Black-box 1, ResNet ($\hat{f}^1_\phi$)



(b) Black-box 2, CNN+lrn ($\hat{f}^2_\chi$)

Figure 6.5: (a) and (b): 2d-Histograms of the parameters of the logistic function fits to the confidence manifolds for a ~ 4000 samples.

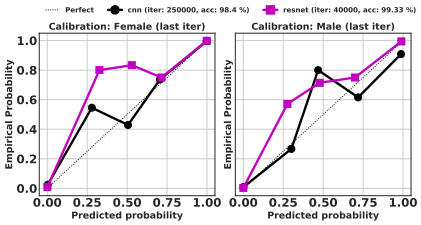


Figure 6.6: Reliability Diagram for Calibration stratified by (potentially protected) attributes of interest (gender): A perfectly calibrated classifier should manifest an identity function. Deviation from the identity function suggests mis-calibration and can be used for model comparison when accuracy and other metrics are comparable.

parameter estimates stratified by the target label and the attribute of interest (gender). This allows to summarize the confidence manifolds across groups of interest for overall model comparison. For reference, Figure 6.6 shows the Reliability Diagram for both black-boxes. The ResNet model generally demonstrates more uniform steepness across samples at different distances from the decision boundary compared to the CNN+lrn model. Both models have a relatively small x_0 for blond haired males suggesting lower confidence in their predictions. Thus, summarizing confidence manifolds provides additional insight that may not be characterized by Reliability Diagrams for model comparison.

6.5 Discussion

This chapter presents a novel approach to characterizing and explaining black-box supervised models via examples. An unsupervised implicit generative model is used as to approximate the data manifold, and subsequently used to guide the generation of increasingly confounding examples given a starting point. These examples are used to probe the target black-box in several ways. In particular, we demonstrate the utility of manifold constrained

examples in automatically detecting bias in black-box learning w.r.t. a (potentially protected) attribute as well as for model comparison. The proposed method also allows one to visualize training progression and provides insights complementary to notions of calibration of the black-box model. Limitations of the proposed method include reliance on the implicit generator as a proxy of the data manifold. However, we note that we do not rely on specific architectures and/or training mechanisms for the generative model. We used images as they are easy to visualize even in high-dimensions. However extending our studies to complex datasets beyond images is a compelling future extension.

Thus constraining generative models allows one to probe complex black-boxes for providing explanations of model outcomes, evaluating training progression, and providing complementary methods of model comparison when conventional metrics may be similar.

Chapter 7

Leveraging Heterogeneity via Constraints

This chapter focuses on leveraging constraints for semi-supervised learning. We demonstrate the use on constraints in two different machine learning paradigms in which heterogeneity in data sources can be leveraged *in lieu* of lack of expert annotation. To objective is to learn effectively with limited annotation with respect to the specified criterion in each framework. The first section applies this framework to a listwise LeTOR framework while the latter focuses on a clustering task.

7.1 Heterogeneous Sources as *Views*

Lack of expert annotation is a significant practical problem in machine learning. In order to learn effectively in this regime, we propose to leverage heterogeneity in data sources (called *views*). Complementary information available in these views can be better harnessed to improve over conventional machine learning algorithms. In particular, this chapter demonstrates how to leverage multiview information using constraints.

This chapter is based on content published in (Joshi et al., 2016a, 2018a). The author of this dissertation contributed to model formulation, implementation, and empirical evaluation described in this chapter.

7.2 Constrained Semi-Supervised LeTOR

Most LeTOR methods can be considered to be developed for three practical scenarios where estimating a preference order is desirable: pointwise, pairwise and listwise. These paradigms are primarily determined by the type of supervision available to the learning algorithm. Specifically, pointwise methods like McRank (Li et al., 2007) require a score (indicating some notion of preference) associated with the entities to be ranked and the learning algorithm in turn learns a mapping from entities to associated scores. The scores can then be interpreted to provide a preference order over the entities. Pairwise methods, as the name suggests, use preferences provided over pairs of objects, for example, rank-SVM (Elisseeff et al., 2001), rankBoost (Freund et al., 2003) etc. and learn a preference order over the pairs of target entities. This remains the most popular form of supervision due to the ease of collecting pairwise preferences over ordinal scores or permutations over a large number of entities. Finally, listwise LeTOR methods (Xia et al., 2008; Acharyya et al., 2012; Acharyya and Ghosh, 2014) are the most general and require a preference order provided for entities and is then able to provide a ranking over a new set of objects.

Training listwise LeTOR models in a supervised manner poses practical problems like the prohibitive cost associated with collecting reliable preferences for entities from human annotators. Additionally, preference learning is a combinatorial problem as the target variables are jointly learned as permutations over the data samples. Costly human annotations naturally motivate us to consider whether data samples with unknown preferences via annotations (unlabeled samples) can be harnessed intelligibly to augment supervised LeTOR counterparts to improve their performance as measured by well stud-

ied metrics like the Normalized Discounted Cumulative Gain (Järvelin and Kekäläinen, 2000), Kendall's Tau correlation (Kendall, 1938) etc.

In this work, we focus on listwise ranking methods. We posit that there are several advantages of augmenting LeTOR models with entities for whom no preference order is provided, specifically in the listwise setting. First, given that supervision in a listwise form is the most difficult to obtain, in practice, it is only natural to leverage any unlabeled data available. When very few entities have a preference graph available for learning, such semi-supervision can help improve generalization performance over unseen or new entities. For example, in a healthcare institution, clinicians may be extremely busy to score patients according to disease risk, but it is required to rank a set of newly admitted patients by disease severity. The algorithm detailed here have appeared in Joshi et al. (2018a).

Most existing methods for semi–supervised ranking operate under transductive settings (Joachims, 1999; Amini et al., 2008). Transductive methods assume that test samples are known at training time and can be used as unlabeled samples. While this assumption is reasonable, it may not always be practical, especially in predictive settings. For instance, patients to be ranked by disease risk may be admitted at a future time and are unavailable during training. Inductive semi–supervised ranking methods that can estimate rank order on entities not used for training have been explored for pairwise and listwise ranking settings (Szummer and Yilmaz, 2011; Gao and Yang, 2014). Thus inductive methods, such as that proposed here, are significantly more applicable in novel clinical applications. Among such methods, Szummer and Yilmaz (2011) require pairwise preferences over the labeled samples. Converting listwise preferences to pairwise ones scales the set of constraints

quadratically in the number of entities to be ranked. The constraints thus arising are pruned for computational reasons using heuristics such as considering only K-nearest neighbors (Szummer and Yilmaz, 2011). On the other hand, our proposed model only requires a listwise preference order for the labeled entities. While probabilistic models like Gao and Yang (2014) based on the listwise Plackett-Luce model (Marden, 1996) mitigate this issue, they rely on co–training (Blum and Mitchell, 1998b) to train an inductive ranking model. Co–training models, conventionally developed for classification or regression use multiple *views* of data samples to train the models. A *view* is a partition of features assumed to be sufficient to learn a classifier independently given enough samples. Co–training iteratively enhances the training set by labeling a subset of the unlabeled samples where they are chosen to minimize disagreement between labels learned by each *view* (Li et al., 2009). In the ranking setting, however, such a 'disagreement' has to be measured over a complete *Directed Acyclic Graph* structure or permutations over the unlabeled samples, a combinatorial problem in itself. Gao and Yang (2014), therefore approximate the level of disagreement across views using a probabilistic surrogate.

The proposed method is an inductive ranking model that uses co-regularization in order to encourage different *views* to agree over the preference order of unlabeled samples. In particular, we substantially generalize an existing listwise LeTOR method, called Monotone Retargeting (MR) (Acharyya et al., 2012; Acharyya and Ghosh, 2014) to a multiview setting (Blum and Mitchell, 1998b) and then propose to augment it with unlabeled data using co–regularization. MR is an efficient listwise LeTOR algorithm that leverages the simplicity of training conventional Generalized Linear Models (GLMs) for ranking. In particular, observing that the rank scores need not be regressed to

exactly so long as the ordering is preserved, MR searches over all monotonic transformations of the rank scores that may be easier for GLM estimates to fit to. MR also develops an efficient technique to search over such transformations. Co–regularization has been previously explored for regression and classification for semi-supervised learning. Our co–regularization technique is novel, in that it exploits the geometric structure of a preference order. This allows to explicitly impose/encourage agreement across views over the rank ordering estimated by each *view* of the unlabeled input items. Our key contributions can be summarized as follows:

1. We propose a novel inductive semi–supervised listwise LeTOR algorithm.

2. A novel co–regularization method is developed in order to leverage unlabeled data.

3. We exhaustively evaluate our algorithm in three settings (including comparisons to inductive and transductive models) commonly observed in practice and demonstrate the effectiveness of the proposed algorithm.

7.2.1 Co–regularization in LeTOR

We first introduce the paradigm of multi–view classification where co–regularization was first introduced. Multiview learning assumes that there exist two or more *views* or input spaces of samples such that they also agree on the class labels of the samples and are independent conditioned on the class labels. Each view can learn a classifier independently given enough samples. This 'multi–view' assumption allows one to augment supervised methods with unlabeled data. Specifically, co–regularized multi–view algorithms allow to explicitly minimize disagreement between views over the label assigned to

unlabeled samples – generally imposed via some form of regularization. In practice, an example of such *views* could be **(a)** Physiological measurements of patients that need to be ranked on disease severity or mortality risk and **(b)** Prescription data corresponding to the same set of patients .

In order to extend this notion to the listwise LeTOR setting, we make the following assumptions:

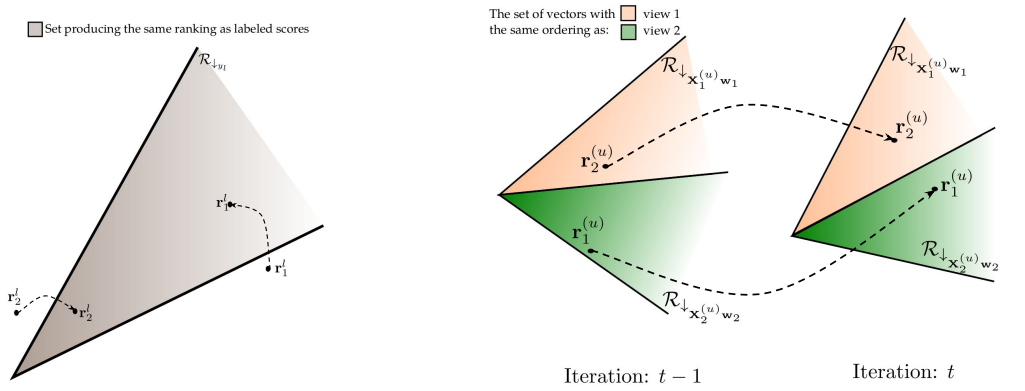
1. Each view agrees on the rank order assigned to a set of items within the query.

2. Each view is conditionally independent given the rank ordering assigned to items in the query.

Without loss of generality, we now present a co–regularized multiview LeTOR method assuming our data consists of two views. The MR model is first generalized to a multiview supervised setting. To incorporate unlabeled data, we leverage the fact that the set of all isotonic vectors to a given vector is a convex cone and formulate a novel co–regularization over the unlabeled samples. The multiview MR cost function can be easily augmented using such a co–regularization for which a coordinate descent based algorithm is proposed and evaluated.

7.2.2 MR-CORE: Algorithm for semi-supervised LeTOR

$\mathbf{X}^{(l)} \in \mathbb{R}^{n \times d}$ denotes the subset of (n) labeled samples in the dataset while $\mathbf{X}^{(u)} \in \mathbb{R}^{m \times d}$ denotes (m) unlabeled samples. Let V be the number of views available that are required to be ranked according to target preferences. The subscript v is hereafter used to denote a view $\mathbf{X}_v$. Let $\mathbf{X}_v^{(l)} \in \mathbb{R}^{n \times d_v}, v \in [V]$ denote the v^{th} view of the labeled samples and $\mathbf{X}_v^{(u)} \in \mathbb{R}^{m \times d_v}, v \in [V]$ the



(a) Retargeting step on scores of a labeled query

(b) Co–regularization on unlabeled queries across time

Figure 7.1: Visual representation of the proposed MR-CORE algorithm

v^{th} view of the unlabeled samples. Let $\mathbf{y}^{(l)}$ be the scores denoting the rank or a permutation of the labeled samples. Let $\mathbf{w}_v \in \mathbb{R}^{d_v}$ parametrize the GLM to be learned for view v. In the absence of any unlabeled data, a multiview LeTOR task entails learning view–specific models such that all views rank the labeled documents similarly, i.e. the learned rank estimates are isotonic to $\mathbf{y}^{(l)}$ in all views. Let $\mathbf{r}_v^{(l)}, v \in [V]$ be the *retargeted* scores learned for view v. That is, the GLM in view v will estimate a model with $\mathbf{r}_v^{(l)}$ as targets where $\mathbf{r}_v^{(l)}$ are isotonic to $\mathbf{y}^{(l)}$. Let $\mathcal{Q}^{(l)}$ be the set of labeled queries and $\mathcal{Q}^{(u)}$, the set of unlabeled queries. As before, let q be the subscript used for each query. Supervised LeTOR using Multiview Monotone Retargeting can formulated as the following estimation:

$$\underset{\substack{\mathbf{w}_v \in \mathsf{C}_v, v \in [V] \\ \mathbf{r}_{v,q}^{(l)} \in \mathcal{R}_{\downarrow_{\mathbf{y}_q^{(l)}}}}}{\arg\min} \sum_{\substack{v \in [V] \\ q \in \mathcal{Q}^{(l)}}} \mathcal{D}_\phi(\mathbf{r}_{v,q}^{(l)} \| \nabla \phi^{-1}(\mathbf{X}_{v,q}^{(l)} \mathbf{w}_v)) \tag{7.1}$$

where C_v are (convex) constraints imposed on the parameters to avoid degeneracy (like the s–Simplex). Note that the parameters are view–specific

but shared across all queries. The estimation procedure to solve Equation (7.1) is an extension to Algorithm 2 where the GLM parameters for each view are updated in parallel. The *retargeting* step involves re-estimating the target scores to lie in the set of all vectors isotonic to $\mathbf{y}^{(l)}$. Figure 7.1 visually demonstrates the proposed algorithm. Specifically, Figure 7.1(a) demonstrates the first retargeting step. Consequent retargeting steps in each view will search for a target score closest in distance to $\nabla\phi^{-1}(\mathbf{X}_v^{(l)}\mathbf{w}_v)$ within the same convex cone $\mathcal{R}_{\mathbf{y}^{(l)}}$. Incorporating unlabeled data in a ranking framework poses multiple challenges. For example, co–regularization requires to explicitly minimize the disagreement between the preference order of items in a query across views – an inherently combinatorial problem. For instance (Gao and Yang, 2014) rely on surrogate measures of rank–based disagreements across views. Further, relying on regularization methods based solely on similarities of the documents (Szummer and Yilmaz, 2011) may inherently produce a ranking without harnessing the discriminative powers of the rank scores themselves.

The fact that the set of all vectors isotonic to a given score vector is a convex cone (Acharyya et al., 2012) allows us to augment the Multiview MR framework to incorporate unlabeled samples. Specifically, we maintain view–specifc rank scores $\mathbf{r}_{v,q}^{(u)}$ for unlabeled items in each query q and each view v. Explicitly measuring disagreement between rank scores is, as suggested before, a combinatorial problem. Instead, we constrain the rank scores estimated by a given view to lie in the convex cone determined by the parametric estimates of the second view. That is, consider the case of two views and let $\mathbf{r}_{1,q}^{(u)}$ and $\mathbf{r}_{2,q}^{(u)}$ be the rank scores assigned to the unlabeled items in the q^{th} query. We impose the following constraints on each of the target scores: (a) $\mathbf{r}_{1,q}^{(u)} \in \mathcal{R}_{\downarrow_{\mathbf{X}_{2,q}^{(u)}\mathbf{w}_2}}$ and (b) $\mathbf{r}_{2,q}^{(u)} \in \mathcal{R}_{\downarrow_{\mathbf{r}_{1,q}^{(u)}\mathbf{w}_1}}$

$$\underset{\substack{\mathbf{w}_1\in\mathsf{C}_1,\mathbf{w}_2\in\mathsf{C}_2\\ \mathbf{r}_{v,q}^{(l)}\in\mathcal{R}_{\downarrow_{\mathbf{y}_q^{(l)}}},v\in[2]\\ \mathbf{r}_{1,q}^{(u)}\in\mathcal{R}_{\downarrow_{\mathbf{X}_{2,q}^{(u)}\mathbf{w}_2}}\\ \mathbf{r}_{2,q}^{(u)}\in\mathcal{R}_{\downarrow_{\mathbf{X}_{1,q}^{(u)}\mathbf{w}_1}}}}{\arg\min}\left[\sum_{\substack{v\in[2]\\ q\in\mathcal{Q}^{(l)}}}\mathcal{D}_\phi(\mathbf{r}_{v,q}^{(l)}\|\nabla\phi^{-1}(\mathbf{X}_{v,q}^{(l)}\mathbf{w}_v))+\lambda\sum_{\substack{v\in[2]\\ q\in\mathcal{Q}^{(u)}}}\mathcal{D}_\phi(\mathbf{r}_{v,q}^{(u)}\|\nabla\phi^{-1}(\mathbf{X}_{v,q}^{(u)}\mathbf{w}_v))\right] \tag{7.2}$$

Thus the LeTOR MR estimator that imposes such co–regularization is given by the cost function in Equation 7.2, where $\lambda > 0$ determines the relative importance of the unlabeled queries. Note that the convex cones defined by the scores on unlabeled queries, i.e. $\mathcal{R}_{\downarrow_{\mathbf{X}_{v,q}^{(u)}\mathbf{w}_v}}, v \in [2], q \in \mathcal{Q}^{(u)}$ shift every time the GLM parameter estimates are updated. Hence estimating all parameters jointly is not feasible. Further, this shifting property is desirable because this allows to iteratively update the region where both views agree on the rank scores of unlabeled samples. In order to ease the optimization procedure, a coordinate descent method (with line search) is developed for solving Equation (7.2). The proposed algorithm iteratively estimates $\mathbf{r}_v^{(l)}, \mathbf{r}_v^{(u)}$ and $\mathbf{w}_v, v \in [2]$ in each descent step (we have dropped the subscript for queries for brevity). Note that when $\mathbf{r}_v^{(l)}, \mathbf{r}_v^{(u)}, \forall v \in [2]$ are held constant, the updates for $\mathbf{w}_v, v \in [2]$ can occur in parallel. For experimentation, we interleave update to all parameters associated with views 1 and 2. Figure 7.1(b) demonstrates the shifting cone behavior across iterations and the dynamic co–regularization for each view. The complete coordinate descent algorithm is described in Algorithm 5.

Algorithm 5 MRCORE
Input: $\mathbf{X}_{v,q}^{(l)}, \mathbf{X}_{v,q}^{(u)} \in \mathbb{R}^{|\mathcal{V}|\times d_v}, v, q \in [2] \times \mathcal{Q}; \mathbf{y}_q^{(l)}, q \in \mathcal{Q}^{(l)}; \lambda > 0; \phi$

Initialize $w_v, v \in [2]$; $\mathbf{r}_{v,q}^{(l)}, v, q \in [2] \times \mathcal{Q}^{(l)}$ and $r_{v,q}^{(u)}, v, q \in [2] \times \mathcal{Q}^{(u)}$:
while Not converged **do**
Update GLM parameters of view 1:

$$\mathbf{w}_1 = \arg\min_{\mathbf{w}\in\mathbb{R}^d} \sum_{q\in\mathcal{Q}^{(l)}} D_\phi(\mathbf{r}_{1,q}^{(l)} \| \nabla\phi^{-1}(\mathbf{X}_{1,q}^{(l)}\mathbf{w})) + \lambda \sum_{q\in\mathcal{Q}^{(u)}} D_\phi(\mathbf{r}_{1,q}^{(u)} \| \nabla\phi^{-1}(\mathbf{X}_{1,q}^{(u)}\mathbf{w}))$$

Retargeting step for view 1, i.e. $\mathbf{r}_{1,q}^{(l)}, q \in \mathcal{Q}^{(l)}$:

$$\mathbf{r}_{1,q}^{(l)} = \arg\min_{\mathbf{r}_q \in \mathcal{R}_{\downarrow_{\mathbf{y}_q^{(l)}}}} D_\phi(\mathbf{r}_q \| \nabla\phi^{-1}(\mathbf{X}_{1,q}^{(l)}\mathbf{w}_1)) \; \forall q \in \mathcal{Q}^{(l)} \text{ in parallel}$$

Co-regularization step for view 2:

$$\mathbf{r}_{2,q}^{(u)} = \arg\min_{\mathbf{r}_q \in \mathcal{R}_{\downarrow_{\mathbf{X}_q^{(u)}\mathbf{w}_1}}} D_\phi(\mathbf{r}_q \| \nabla\phi^{-1}(\mathbf{X}_{2,q}^{(u)}\mathbf{w}_2)) \forall q \in \mathcal{Q}^{(u)} \text{in parallel}$$

Update GLM parameters of view 2:

$$\mathbf{w}_2 = \arg\min_{\mathbf{w}\in\mathbb{R}^d} \sum_{q\in\mathcal{Q}^{(l)}} D_\phi(\mathbf{r}_{2,q}^{(l)} \| \nabla\phi^{-1}(\mathbf{X}_{2,q}^{(l)}\mathbf{w})) + \lambda \sum_{q\in\mathcal{Q}^{(u)}} D_\phi(\mathbf{r}_{2,q}^{(u)} \| \nabla\phi^{-1}(\mathbf{X}_{2,q}^{(u)}\mathbf{w}))$$

Retargeting step for view 2, i.e. $\mathbf{r}_{2,q}^{(l)}, q \in \mathcal{Q}^{(l)}$:

$$\mathbf{r}_{2,q}^{(l)} = \arg\min_{\mathbf{r}_q \in \mathcal{R}_{\downarrow_{\mathbf{y}_q^{(l)}}}} D_\phi(\mathbf{r}_q \| \nabla\phi^{-1}(\mathbf{X}_{2,q}^{(l)}\mathbf{w}_2)) \forall q \in \mathcal{Q}^{(l)} \text{ in parallel}$$

Co-regularization step for view 1

$$\mathbf{r}_{1,q}^{(u)} = \arg\min_{\mathbf{r}_q \in \mathcal{R}_{\downarrow_{\mathbf{X}_q^{(u)}\mathbf{w}_2}}} D_\phi(\mathbf{r}_q \| \nabla\phi^{-1}(\mathbf{X}_{1,q}^{(u)}\mathbf{w}_1)) \forall q \in \mathcal{Q}^{(u)} \text{ in parallel}$$

end while

As demonstrated by Algorithm 5 and Figure 7.1(b), the co–regularization always *retargets* each view such that it enforces the ranked lists to lie in the convex cone defined by the other view on unlabeled data, thus enforcing an iterative agreement on the rank order of unlabeled data. Over iterations, these convex cones shift as new estimates for $\mathbf{w}_v, v \in [V]$ are obtained. We call this phenomenon of dynamic constraints as shifting cones for co–regularization henceforth.

Adding Margins: Note that for the retargeting step on labeled queries, adding fixed margins (for total order) (Acharyya and Ghosh, 2014) can lead to better empirical performance and can be easily added by using a modified version of the PAV algorithm. Our experiments use this version of PAV for empirical evaluation. Note that margins cannot be added to ranked scores over unlabeled data as no information about the ranked scores is known.

Partial Order: In order to extend the above algorithm to handle partial order, a simple sorting step can be added after each view's retargeting step for labeled queries and results in minor book–keeping over the indices.

7.2.3 Consensus ranking & ranking novel queries:

Once the model is learned, for any new query and appropriate views, we would like to rank documents in the query. Our algorithm does not directly provide a single rank order across all views for a given query. We therefore learn a weighted combination of the view–specific scores by holding out a few queries (as a validation set) during training. Let $\tilde{\mathbf{w}}_1$ and $\tilde{\mathbf{w}}_2$ be the estimates learned using Algorithm 5. Let α and $(1-\alpha)$ be the weights associated with

view 1 and 2 respectively whose weighted combination is used to obtain a consensus rank ordering. Then the ranking on a query in the validation set $\mathcal{Q}^{(val)}$ is given by:

$$\mathbf{r} = \sum_{q \in \mathcal{Q}^{(val)}} \alpha \nabla \phi^{-1}(\mathbf{X}_{1,q} \tilde{\mathbf{w}}_1) + (1 - \alpha) \nabla \phi^{-1}(\mathbf{X}_{2,q} \tilde{\mathbf{w}}_2)$$

where $\tilde{\mathbf{w}}_1$ and $\tilde{\mathbf{w}}_2$ are the weights learned using Algorithm 5. We grid search over α on the validation queries to estimate the weighted combination (denoted by α^*) that obtains the best consensus ranking (as measured by any standard metric of interest like nDCG, Kendall's tau distance or Spearman's rank order correlation) on the validation set. Given a new query $\mathbf{X}^{(new)}$, the rank ordering of items in this query is given by:

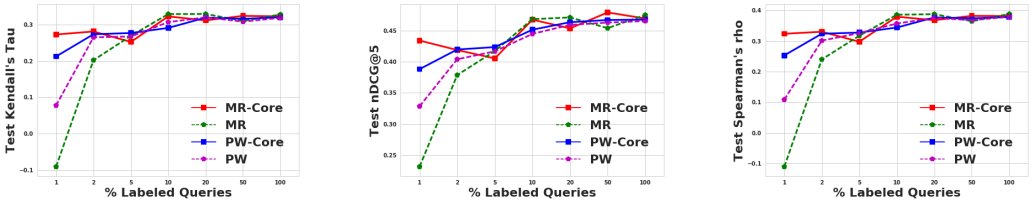
$$\mathbf{r}^{(new)} = \alpha^* \nabla \phi^{-1}(\mathbf{X}_1^{(new)} \tilde{\mathbf{w}}_1) + (1 - \alpha^*) \nabla \phi^{-1}(\mathbf{X}_2^{(new)} \tilde{\mathbf{w}}_2)$$

The rank order determined by $\mathbf{r}^{(new)}$ is thus the desired permutation of items in the test query.

7.2.4 Incorporating multiple views:

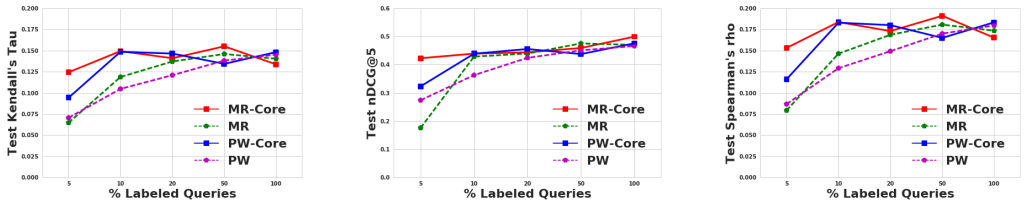
In practice, more than two views may be available. In order to incorporate them, we propose to use sequential co–regularization, wherein a single view is retargeted into the isotonic set defined by all other views. Intuitively, this may be understood as a single step of an alternating projection algorithm that projects the ranked scores to an intersection of the convex cones defined by all other views. As before, only one view is updated at a time with parameters estimated in all other views held fixed.

7.2.5 Empirical evaluation



(a) Kendall's Tau (b) nDCG@5 (c) Spearman's rho

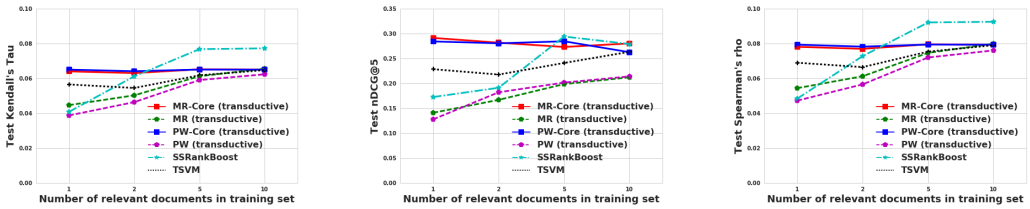
Figure 7.2: Ranking performance on held-out set of MR-Core when augmented using unlabeled data on MQ-2008. The x-axis sweeps over the percentage of queries used as labeled data from the training set. **MR-Core:** proposed model, **PW-Core:** pointwise model augmented with unlabeled data, **MR:** Supervised MR, **PW:** Supervised pointwise model.



(a) Kendall's Tau (b) nDCG@5 (c) Spearman's rho

Figure 7.3: Ranking performance on held-out data when rank scores are only available as relevance/ pairwise scores on OHSUMED data.

We evaluate the proposed model for its effectiveness at improving ranking performance when augmented by unlabeled data. Specifically, we evaluate the model performance in a conventional listwise ranking setting. That is, the model is trained to rank a fixed set of documents in order of preference or relevance in the context of a query. At test time, a new query is provided and the documents associated to the query are ranked according to preference/relevance. The performance of these models is measured using



(a) Kendall's Tau (b) nDCG@5 (c) Spearman's rho

Figure 7.4: Comparison to popular transductive ranking algorithms. The x-axis sweeps over the number of relevant documents in the labeled set. **TSVM:** Transductive SVM, **ssRankBoost:** Transductive Boosting for LeTOR.

three well known ranking metrics, Normalized Discounted Cumulative Gain or nDGC (Järvelin and Kekäläinen, 2000), Spearman's Rank Ordered Correlation (Kendall, 1948) and Kendall's Tau distance (Kendall, 1938). We evaluate the model in the following contexts:

1. First, we evaluate the advantage of using unlabeled data in a co–regularized ranking compared to a completely supervised list-wise setting, especially when very few queries are labeled or the average number of documents associated with each query is very small.

2. We evaluate whether the proposed ranking algorithm generalizes better than point-wise methods like supervised and inductive regression to evaluate the advantages of the *multiview retargeting* mechanism.

3. We further investigate the utility of our algorithm to learn a preference structure when supervision is provided as pairwise preferences. Note that this is an inductive setting, i.e. test queries are not observed at training time. Therefore this setting cannot be compared with transductive semi–supervised algorithms.

4. Finally, we also evaluate whether our algorithm performs at par in a transductive setting. Most existing models in this regime are available for bipartite ranking (Joachims, 1999; Amini et al., 2008). Note again that transductive models use test data during training (as unlabeled samples).

Table 7.1: LETOR Datasets Description

Name	**#queries** $\lvert\Omega\rvert$	**#document features** d
OHSUMED	106	46
MQ2008	1692	46
TREC2004	75	44

We consider two standard list-wise datasets OHSUMED and MQ2008 from the LeTOR 4.0 (Qin and Liu, 2013) dataset to evaluate our first two objectives. OHSUMED is a collection of online articles from the medical information database MEDLINE.[1] MQ2008 contains query sets called the Million Query Track[2] from the TREC2008 dataset. For comparison to transductive settings, we use the TD2004 dataset from LeTOR 1.0 (Liu et al., 2007) containing pairwise relevances. Note that all our baseline transductive algorithms have been primarily designed for bipartite ranking. The details of the datasets are provided in Table 7.1. For all datasets, two views are generated by splitting in half, a random permutation of the features indices.

We compare to the following supervised baselines to evaluate the improvement in ranking performance when augmented using unlabeled data.

[1] https://www.nlm.nih.gov/bsd/pmresources.html
[2] http://ir.cis.udel.edu/million/index.html

1. **Margin-Equipped Monotone Retargeting (MR) (Acharyya and Ghosh, 2014)**: This is the supervised list-wise ranking method which the proposed model MR-Core builds upon. Note that we compare to a multi-view version of MR without co–regularization over unlabeled data. The final preference order for any new sample is determined in the same manner as that of MR-Core, i.e. by learning an appropriate weighting. Note that while MR can use any of the Bregman divergences, we demonstrate results using the Euclidean distance ($\ell_{\mathbf{2}}$-norm).

2. **Ridge Regression (a pointwise method – PW)**: This is a supervised multiview linear regression that estimates exact scores provided by supervision. The preference order for a new list of documents can be determined by first estimating scores and sorting them in order of the scores.

3. **Co-regularized Linear Regression (PW-Core)**: This is a multiview linear regression that incorporates co–regularization by measuring disagreement between the target estimates in each view. The model primarily differs from co–regularized MR in that the co–regularization can be considered to be *pointwise*, i.e. fitting the ranking scores exactly in a semisupervised manner. Specifically, this is formulated as follows:

$$\begin{aligned} \text{argmin}_{\mathbf{w}_1,\mathbf{w}_2} &\|\mathbf{y}_l - \mathbf{X}_{l1}\mathbf{w}_1\|^2 + \|\mathbf{y}_l - \mathbf{X}_{l2}\mathbf{w}_2\|^2 \\ &+ \lambda\|\mathbf{X}_{u1}\mathbf{w}_1 - \mathbf{X}_{u2}\mathbf{w}_2\|_2^2 \\ &\text{s.t. } \mathbf{w}_1, \mathbf{w}_2 \in \mathsf{C} \end{aligned} \tag{7.3}$$

where $\mathbf{w}_1$, $\mathbf{w}_2$ are the regression weights similar to a linear regression model and C is a convex set that may be used to avoid any degenerate solutions. We also add ℓ_2–regularization to all methods to avoid over–fitting.

The results evaluating our first and second objectives are provided in Figure 7.2 on the MQ-2008 dataset. Specifically, we sweep over the percent-

age of queries used as labeled data. The rest are only incorporated via co–regularization. Additionally, we also ensure that labeled queries are added in order of their sizes, i.e. we train on as few total number of documents as possible. The evaluation suggests that when augmented with unlabeled data, the *retargeting* mechanism of MR combined with the proposed co–regularization improves ranking performance over pointwise algorithms (including those augmented with unlabeled data), especially when very few queries are labeled. Specifically, in the regime where very few labeled queries are available, supervised MR performs worse than the supervised pointwise method, but outperforms all baselines when augmented with unlabeled data using our co–regularization algorithm.

To evaluate performance of MR-Core when supervision is provided as pairwise preferences, we use the OHSUMED dataset but split the queries to generate queries with pairwise preferences. That is, consider a partially ranked list where the documents in a query are rated from 1 to K. Then each query can be divided into $\binom{K}{2}$ queries generating pairwise preferences. We would like to evaluate if the deterioration in performance when only pairwise preference is observed is prohibitive for practical use. Note that the test queries require a listwise outcome/ranking of all documents opposed to just a relevance score. This evaluation (shown in Figure 7.3) demonstrates that re–targeting methods that simply focus on the ordering still perform comparably to pointwise methods. This demonstrates their utility even if listwise preferences are unavailable.

Finally, we evaluate the model in the context of existing transductive semi–supervised ranking settings. Note that most transductive algorithms are primarily designed for bipartite ranking. Specifically we compare the perfor-

mance to two popular baselines detailed below:

1. **ssRankBoost (Amini et al., 2008)**: This model proposes a boosting algorithm to learn a bipartite ranking on data samples in a transductive setting. Note that the model has to be learned for each query independently.

2. **TSVM (Joachims, 1999)**: This is a bipartite ranking algorithm based on Support Vector Machines (Cortes and Vapnik, 1995) augmented to incorporate unlabeled data.

Note that transductive methods are evaluated on a single query and tested to evaluate whether they score unranked documents in the query desirably. To this end, we compare the models in a *Relevance Feedback Task* (Szummer and Yilmaz, 2011), more suitable for transductive settings. The results (in Figure 7.4) suggest that the proposed method performs comparably or better than other transductive models, again in the regime where the number of labeled documents is very small.

7.3 Constraints Based Clustering

We propose a new co–regularization framework in order to leverage heterogeneity of data to improve a latent variable clustering framework. This is known as multiview clustering and specifically focuses on unsupervised learning of cluster membership of entities from all views. Multiview clustering operates under the assumption that the underlying latent or unknown cluster membership is the same across different views. Two main principles, namely *co-training* (Blum and Mitchell, 1998a) and *co-regularization* (Sindhwani et al., 2005), form the basis of algorithms used to impose the multiview assumption for inference and/or learning. Co-training methods iteratively bootstrap

estimates of a view using learned hypotheses from other views to converge to a coherent model (Blum and Mitchell, 1998a). Co-regularization methods minimize disagreement between view-specific hypotheses explicitly during training (Sindhwani et al., 2005). The two learning and inference procedures described here along with the following empirical evaluation appeared in Joshi et al. (2016a)

Multiview methods offer significant theoretical and practical advantages compared to concatenating views. However, model mis-specification, noisy measurements and/or unknown biases views do not necessarily cluster data simultaneously in practice. To account for such mis-specification, this work proposes to explicitly characterize biases across different views. We maintain view-specific cluster membership distributions but impose coherence by encouraging the learned or posterior distributions to be 'close' where closeness is measured via appropriate divergences. In particular, a weighted sum of Rényi Divergences between view-specific posterior distributions is minimized. Storkey et al. (2014) have shown that when aggregating distributions from biased agents, Rényi Divergence based aggregation provides a target distribution with the maximum entropy. Our model therefore does not assume a bias free condition and explores various aggregation strategies most suited for the data. We recover an existing multiview clustering algorithm, Co-EM (Bickel and Scheffer, 2005), as a special case of our framework. Co-EM uses linear aggregation of views which corresponds to the maximum entropy aggregation when views completely disagree on cluster memberships. We propose an algorithm that augments the conventional Expectation-Maximization framework using Rényi Divergence based co-regularization for learning multiview mixture models. We also treat a special case of this algorithm separately

as Co-EM can be recovered as a special case of this algorithm for a particular choice of the Rényi Divergence.

The proposed algorithms have been extensively evaluated and compared not only with existing multiview clustering techniques, but also with related methods that deal with modeling multiple views jointly. A few instances of such relevant techniques are consensus clustering models, collective matrix factorization, joint latent variable models where the latent cluster memberships are exactly the same across views. The following section provides details on the proposed algorithms to learn multiview mixture models using Rényi Divergence based co-regularization.

Generative Model

Assume the generative model of the data as noted in Figure 2.1. Let N denote the number of samples and V denote the number of views. Let $n \in [N]$ index samples and $v \in [V]$ index views. Let $\mathbf{x}_n^v$ represent the observed features for sample n in view v. Let the total number of clusters be K. Let $\mathbf{z}_n^v \in \{0,1\}^K$ be the latent cluster membership of sample n in view v such that only one element of the vector is 1. Thus if sample n belongs to cluster k in view v, then the k^{th} element of view v, i.e. $\mathbf{z}_{n,k}^v$ is 1 and the rest are 0. Let $\boldsymbol{\pi}_n \in \Delta^K$ be the prior probability of cluster membership $\mathbf{z}_n^v \forall v \in [V]$ for sample n. Let $\Psi^v = \{\Psi_k^v\}$ denote the set of all parameters of the mixture model for view v. The generative procedure is briefly described here:

- For each sample n
  - For each view v

* Sample cluster membership indicator $\mathbf{z}_n^v \sim p(\mathbf{z}; \boldsymbol{\pi}_n)$ from a categorical distribution independently.
* Sample feature $\mathbf{x}_n^v \sim p(\mathbf{x}; \mathbf{z}_n^v, \boldsymbol{\Psi}^v)$ independently.

7.3.1 Alternating co-regularization and aggregation

Without co-regularization, the parameters $\boldsymbol{\pi}_n$ and $\boldsymbol{\Psi}^v$ can be estimated using Expectation-Maximization (Dempster et al., 1977) or EM that maximizes the expected log-likelihood of the data ($\sum_n \sum_v \log p(\mathbf{x}_n^v, \mathbf{z}_n^v; \boldsymbol{\pi}, \boldsymbol{\Psi}^v)$). Our proposed method augments the EM procedure to incorporate co-regularization using Rényi divergences and is briefly described below.

7.3.2 GRECO and LYRIC: Algorithms for multiview clustering

Our first strategy, called global co-regularization estimates a global posterior distribution $g(\mathbf{z}_n)$ from per-view posteriors $p(\mathbf{z}_n^v|\mathbf{x}_n^v, \boldsymbol{\Psi}^v)$ as given by minimizing (7.4)

$$g_t^*(\mathbf{z}_n) = \arg\min_{g(\mathbf{z}_n)} \sum_{i \in [V]} \frac{w_i}{\gamma} \mathcal{D}_\gamma(p(\mathbf{z}_n^i|\mathbf{x}_n^i, \boldsymbol{\Psi}_t^i)||g(\mathbf{z}_n)) \tag{7.4}$$

where $w \triangleq \{w_i\} \in \Delta^V$ weighs each view appropriately and D_γ is an appropriate Rényi divergence parametrized by view v. Indexing by t suggests estimates of parameters at iteration t. Further, for any given view, we expect its posterior estimates to be a trade-off between the current global estimate as well as the view-specific cluster membership distribution. Thus, we estimate a new posterior $q(\mathbf{z}_n^v)$ using the global estimate $g_t^*(\mathbf{z}_n)$ and the view-specific posterior estimate $p(\mathbf{z}_n^v|\mathbf{x}_n^v, \boldsymbol{\Psi}_t^v)$ by minimizing (7.5)

$$q_t(\mathbf{z}_n^v) = \arg\min_{q(\mathbf{z}_n^v)} \frac{w_g}{\gamma} \mathcal{D}_\gamma(g^*{}_t(\mathbf{z}_n)||q(\mathbf{z}_n^v)) + \frac{(1-w_g)}{\gamma} \mathcal{D}_\gamma(p(\mathbf{z}_n^v|\mathbf{x}_n^v, \boldsymbol{\Psi}_t^v)||q(\mathbf{z}_n^v)) \tag{7.5}$$

Algorithm 6 GRECO

Given data $\{\mathbf{x}_n^v\}$, γ, $\mathbf{w}$, Initialize $\boldsymbol{\pi}_n$, $\boldsymbol{\Psi}^v$ $\forall v \in [V]$, $t = 0$
repeat
  for all $v \in [V]$ **do**
    //View specific E-step with the latest estimates of $\boldsymbol{\Psi}^i$s
    for all $i \in [V]$ **do**
      $p(\mathbf{z}_n^i|\mathbf{x}_n^i, \boldsymbol{\Psi}_t^i) \propto p(\mathbf{x}_n^i|\mathbf{z}_n^i, \boldsymbol{\Psi}_t^i)p(\mathbf{z}_n^i; \boldsymbol{\pi}_n)$ $\forall n \in [N]$ in parallel
    end for
    //Coherence enforcing steps for current view v:
    Estimate $g_t^*(\mathbf{z}_n)$ by solving equation (7.4) using Algorithm 7 $\forall n \in [N]$ in parallel
    Estimate $q_t(\mathbf{z}_n^v)$ by solving equation (7.5) using Algorithm 8 $\forall n \in [N]$ in parallel
    //M-step for current view v:
    Using fixed responsibilities $q_t(\mathbf{z}_n^v)$,
    $\boldsymbol{\Psi}_{t+1}^v \leftarrow \arg\max_{\boldsymbol{\Psi}^v} \sum_{n\in[N]} \sum_{k\in[K]} q_t(\mathbf{z}_{n,k}^v) \log p(\mathbf{x}_n^v, z_{n,k}^v = 1; \boldsymbol{\Psi}^v)$
  end for
  $t \leftarrow t + 1$
until converged

The M-step is now executed for each view independently following standard estimation EM procedure albeit with the co-regularized estimates of posteriors. Note that while inter-leaving expectation, co-regularization and maximization steps, we update only a single view in any iteration to avoid co-regularization against old and potentially disparaging posterior estimates. This, we believe helps the algorithm avoid convergence to local minima providing improved empirical performance on convergence. The complete algorithm is called GRECO (Global REnyi divergence based CO-regularization) and is provided as Algorithm 6:

Specific variational procedure to solve (7.4) and (7.5) are provided in Algorithms 7 and 8.

Algorithm 7 Variational Update to solve (7.4)

Given $\mathbf{w} \in \Delta^V$, γ and $\mathbf{\Psi}_t^i \forall i \in [V]$
repeat
$\kappa^i(\mathbf{z}_n) \propto p(\mathbf{z}_n^i|\mathbf{x}_n^i, \mathbf{\Psi}_t^i)^\gamma g(\mathbf{z}_n)^{(1-\gamma)} \forall i \in [V]$
$g(\mathbf{z}_n) \propto \sum_{i\in[V]} w_i \kappa^i(\mathbf{z}_n)$
until converged

Algorithm 8 Variational Update to solve (7.5)

Given w_g, w_v, γ, $g^*(\mathbf{z}_n)$ and current parameter estimates, $\mathbf{\Psi}_t^v$
repeat
$\kappa^*(\mathbf{z}_n) \propto g^*(\mathbf{z}_n)^\gamma q(\mathbf{z}_n^v)^{(1-\gamma)}$
$\kappa^v(\mathbf{z}_n^v) \propto p(\mathbf{z}_n^v|\mathbf{x}_n^v, \mathbf{\Psi}_t^v)^\gamma q(\mathbf{z}_n^v)^{(1-\gamma)}$
$q(\mathbf{z}_n^v) \propto w_g \kappa^*(\mathbf{z}_n) + w_v \kappa^v(\mathbf{z}_n^v)$
until converged

A special case that we study separately is when $w_g = 1$. Thus, we do not have the additional trade-off between the view-specific posterior and the global co-regularized posterior given by (7.5). In the absence of such regularization, every iteration designates $q(\mathbf{z}_n^v)$ to be equal to $g(\mathbf{z}_n^v)$. This procedure, is called LYRIC (LocallY weighted Rényi divergence Co-regularization) and is provided as Algorithm 9. Note that this produces different co-regularized estimates per iteration and hence may converge to different local minima.

A few other special cases that are noteworthy are briefly included here.

Special case I: $\gamma \to 1$

Rényi divergence corresponding to the $\gamma \to 1$ reduces the cost to a weighted sum of KL-divergences with the target distribution on the right hand side of KL-divergence (Storkey et al., 2014). Let the per-view posterior, $p(\mathbf{z}^i|\mathbf{x}^i, \mathbf{\Psi}^i)$ be parametrized by $\boldsymbol{\theta}^i \in \Delta^K$. Let the target distribution

Algorithm 9 LYRIC

Given data $\{\mathbf{x}_n^v\}$, γ, $\mathbf{w}$, Initialize $\boldsymbol{\pi}_n$, $\boldsymbol{\Psi}^v$ $\forall v \in [V]$, $t = 0$
repeat
  for all $v \in [V]$ **do**
    //View specific E-steps with the latest parameter estimates of $\boldsymbol{\Psi}^i$s
    for all $i \in [V]$ **do**
      $p(\mathbf{z}_n^i|\mathbf{x}_n^i, \boldsymbol{\Psi}_t^i) \propto p(\mathbf{x}_n^i|\mathbf{z}_n^i, \boldsymbol{\Psi}_t^i)p(\mathbf{z}_n^i; \boldsymbol{\pi}_n)$ $\forall n \in [N]$ in parallel
    end for
    //Coherence enforcing step for current view v:
    Estimate $q_t(\mathbf{z}_n^v)$ with equation 7.6 using Algorithm (8) $\forall n \in [N]$ in parallel
    //M-step for current view v:
    Using fixed responsibilities $q_t(\mathbf{z}_n^v)$,
    $\boldsymbol{\Psi}_{t+1}^v \leftarrow \arg\max_{\boldsymbol{\Psi}^v} \sum_{n\in[N]} \sum_{k\in[K]} q_t(\mathbf{z}_{n,k}^v) \log p(\mathbf{x}_n^v, z_{n,k}^v = 1; \boldsymbol{\Psi}^v)$
  end for
  $t \leftarrow t + 1$
until converged

$q(\mathbf{z}_n^v)$, be parametrized by $\boldsymbol{\phi}^v \in \Delta^K$. The cost function given by (7.6)

$$q_t(\mathbf{z}_n^v) = \arg\min_{q(\mathbf{z}_n^v)} \sum_{i\in[V]} \frac{w_i}{\gamma} \mathcal{D}_\gamma(p(\mathbf{z}_n^i|\mathbf{x}_n^i, \boldsymbol{\Psi}_t^i)||q(\mathbf{z}_n^v)) \tag{7.6}$$

can be simplified to (7.7).

$$q(\mathbf{z}^v) = \arg\min_{q(\mathbf{z}^v)} \sum_{i\in[V]} w_i \mathrm{KL}\big(p(\mathbf{z}^i|\mathbf{x}^i, \boldsymbol{\Psi}^i)\|q(\mathbf{z}^v)\big) \tag{7.7}$$

For categorical distributions, the closed form solution of (7.7) is given by (7.8) as was derived by Garg et al. (2004).

$$\boldsymbol{\phi}^v = \sum_{i\in[V]} w_i \boldsymbol{\theta}^i \tag{7.8}$$

The linear aggregation closed form solution is not specific to LYRIC and can be generalized to GRECO for the choice of $\gamma \to 1$ as well. Further, if $w_v = (1-\alpha)$

for the view v currently being updated, and $w_i = \frac{\alpha}{V-1}$, where $0 \leq \alpha \leq 1$ for $i \neq v, i \in [V]$, the LYRIC algorithm recovers Co-EM when $\gamma \to 1$. Thus we recover that Co-EM is a special case of LYRIC.

Special case II: $\gamma \to 0$

When $\gamma \to 0$, (7.6) has been shown by Storkey et al. (2014) to be equivalent to a minimization over a weighted sum of the KL-divergences with the target distribution as the argument on the left-hand side of KL-terms. The closed form solution in this case is an averaging of the parameters $\boldsymbol{\theta}^i \; \forall i \in [V]$ in the *log*-space weighted by $w_i \; \forall i \in [V]$ (Garg et al., 2004) as shown in (7.9). The proof is detailed in Appendix C.

$$\log \boldsymbol{\phi}^v = \sum_{i \in [V]} w_i \log \boldsymbol{\theta}^i \tag{7.9}$$

This result is also general and applicable to (7.4) and (7.5) with appropriate weighting. Conventionally, a product of experts (or equivalently log-aggregation) model (Hinton, 2002; Storkey et al., 2014) uses such a product to combine beliefs from independently trained models, for example in an ensemble setting.

Computational Complexity

Co-regularization in each GRECO and LYRIC adds an additional complexity of $\mathcal{O}(NKV^2)$ per iteration where N is the sample size, K is the number of clusters and V is the number of views, compared to the unregularized method. These operations, however, can be trivially parallelized over data samples as well as for calculations required to estimate unnormalized variational parameters for each cluster. For the case where all views are Gaus-

sian mixtures, the complexity per outer iteration is $\mathcal{O}(NKV^2T_{inner} + NKV + \sum_{v\in[V]} d_v^2 K)$ where T_{inner} is the number of inner iterations for variational estimation of co-regularized posteriors, d_v is the dimension of view v. In each of the special cases described earlier, i.e. when $\gamma \to 0$ and $\gamma \to 1$, the complexity reduces to $\mathcal{O}(NKV + \sum_{v\in[V]} d_v^2 K)$ per iteration, same as that of Co-EM, due to closed form solutions available for co-regularization.

7.3.3 Choice of weights and Rényi Divergences

The choice of γ and weights for both LYRIC and GRECO can be determined using cross-validation w.r.t. a desired metric. For empirical studies, we parametrize the weights for easy cross-validation. Let $0 \leq \alpha \leq 1$ be a scalar. For every view $v \in [V]$ being updated, $w_v = 1 - \alpha$. For all other views, $w_i = \frac{\alpha}{V-1} \, \forall \, i \in [V], \; i \neq v$. At every stage in the outer loop of either GRECO or LYRIC, the current view being updated is weighted by $1 - \alpha$ and the rest are weighted equally $\frac{\alpha}{V-1}$. This also ensures fair comparison with Co-EM by maintaining the same parametrization of weights. All experiments therefore demonstrate that the choice of Rényi Divergences has a significant boost in clustering performance.

7.3.4 Prediction on hold-out samples

For out-of sample cluster prediction, the conventional E-step with the learned parameters is used to obtain per-view posteriors for a test sample for all views independently. It is now desirable to obtain a single aggregate posterior for each sample.

$$q(\mathbf{z}) = \arg\min_{q(\mathbf{z})} \sum_{v\in[V]} w^* \mathcal{D}_{\gamma^*}(p(\mathbf{z}|\mathbf{x}^v, \mathbf{\Psi}^{v*})||q(\mathbf{z})) \tag{7.10}$$

For LYRIC, a global posterior can then be obtained using (7.10), where γ^* and w^* are chosen during training via cross-validation as described in Sec. (7.3.3) and $\mathbf{\Psi}^{v*}$ is the set of learned parameters from LYRIC. Similarly for GRECO, the E-step is run for all views independently followed by executing (7.10) to obtain a global posterior. A hard clustering is simply the MAP assignment of $q(\mathbf{z})$.

7.3.5 Empirical evaluation

The proposed methods have been extensively compared with existing multiview clustering models to show that the choice of divergence obtained by tuning γ is of significance, as well as to demonstrate that Rényi divergence is a reasonable choice for co-regularization. All datasets were trained using both LYRIC and GRECO algorithms for different values of $\gamma \in [0, 1]$ discretized in the corresponding log-space. Very high values of Rényi divergences did not matter significantly affect the performance. For all datasets, ground-truth cluster labels are known and utilized for objective evaluation and comparison to baselines. All models and baselines were trained on the same training and hold-out data for five trials with best performing models chosen based on average clustering accuracy for comparison purposes. The mapping between cluster labels to ground truth labels is solved using Hungarian matching (Kuhn, 1955). For comparison to baselines, we only report the best performance obtained across different choices of $\mathbf{w}$ and γ. Hold-out assignment results have only been compared to baselines that explicitly mention a mechanism to obtain hold-out cluster assignment and empirically test the same. We report Clustering Accuracy, Precision, Recall, F-measure, NMI (Strehl and Ghosh, 2003) and Entropy (Bickel and Scheffer, 2005) for

our evaluation. Lower entropy is better while higher values of other metrics show a better performing algorithm. All metrics are defined in Appendix E. Note that the empirical evaluation here maintains prior cluster distribution $\boldsymbol{\pi}_n$ to be equal for all samples n for all probabilistic models, including GRECO and LYRIC without loss of generality. Empirical convergence for a sample fold with multiple initializations (in negative log-likelihood) of GRECO and LYRIC have been included in Appendix F[3]. To the best of our knowledge, our empirical evaluation is the most extensive evaluation of multiview clustering methods compared to prior work in terms of the number of datasets, number of views and comparison to existing baselines.

7.3.5.1 Baselines

The proposed methods are compared to an extensive set of baselines. The baselines are briefly described here.

- **Shared Latent Variable Model (Joint)**: An alternative way of modeling multiple views is to have one latent variable that denotes the cluster membership across all views. This is called the 'Joint' model. This model is equivalent to concatenating views especially in the most commonly assumed scenario i.e. all views are Gaussian mixtures with diagonal covariances.

- **Ensemble Clustering Model (Ensemble) (Strehl and Ghosh, 2003)**: This model trains each view independently followed by a consensus evaluation. To predict the hard clustering assignment, the label cor-

[3]For the CUB dataset, we only have results with a single initialization for a single train-test split. However, average over different splits shows the same trend.

respondence among views is obtained using Hungarian matching (Kuhn, 1955). A single posterior is obtained using the same equation as (7.10) with KL-divergence (log-aggregation), followed by a MAP assignment. This method is compared to only when at most two views are available.

- **Co-EM (Bickel and Scheffer, 2005)**: Co-EM estimates a mixture model per view subject to cross-entropy constraints. The weights for each view are parametrized by $\eta \in [0, 1]$ and the results corresponding to the best performing η are reported.

- **Co-regularized Spectral Clustering (Co-reg(Sp)) (Kumar et al., 2011):** This is the state-of-the-art spectral multiview clustering. The results corresponding to the best performing λ parameter (between 0.01 to 0.1 as suggested by authors) are reported. The implementation provided by the authors is used.[4].

- **Minimizing Disagreement (Min-dis(Sp)) (Sa, 2005):** This is another spectral clustering technique proposed by (Sa, 2005) for 2 views only. The implementation used was implemented and compared to by Kumar et al. (2011).

- **CCA for Mixture Models (CCA-mvc) (Chaudhuri et al., 2009):** This method uses Canonical Correlation Analysis to project views on a lower dimensional space. This model can be used for 2 views only.

- **NMF based Multiview Clustering (NMF-mvc)(Liu et al., 2013):** This method uses non-negative matrix factorization for multiview clus-

[4] http://www.umiacs.umd.edu/~abhishek/papers.html

tering. The original implementation provided by the authors was used for empirical evaluation[5].

A k-means clustering algorithm is used independently for each view to initialize distribution parameters for all probabilistic models. An approximate Hungarian matching problem is solved using the k-means cluster assignments for initialization.

7.3.5.2 Datasets

The datasets are chosen referencing prior work in multiview clustering. Details of the datasets are provided in the following.

- **Twitter multiview[6](Greene and Cunningham, 2013):** This is a collection of twitter datasets in five topical areas (politics-UK, politics-Ireland, Football etc.). Each user has views corresponding to users they follow, their followers, mentions, tweet content etc. We use the politics-uk dataset with three views (mentions, re-tweets and follows). The labels correspond to one of five party memberships of each user. Each view is a *bag-of-words* vector and modeled as a mixture of multinomials for the probabilistic models.

- **WebKB[7]:** This dataset consists of web page information from four university websites: Cornell, Texas, Washington and Wisconsin. We show results for the Cornell dataset. Each sample is a web page with two views, one view of which is the text content (*bag-of-words*) format and

[5] http://jialu.cs.illinois.edu/
[6] http://mlg.ucd.ie/aggregation/
[7] http://lig-membres.imag.fr/grimal/data.html

web-links into and out of the web page (binary *bag-of-words* vector). Each web page can be clustered into one of five topics. Each view is modeled as a mixture of multinomials.

- **NUS Wide Object**[8]**(Chua et al., 2009):** This dataset consists of 31 object classes. Of these, we sub-sample in a balanced manner for 10 classes, with 50 samples belonging to each class. We use 6 views, namely edge histograms (mixture of Gaussians), *bag-of-visual words* of SIFT features (mixture of multinomial distributions) and normalized correlogram (mixture of Gaussians), color histogram (mixture of multinomials), wavelet texture (mixture of Gaussians) and block-wise color moments (mixture of Gaussians).

- **CUB-200-2011**[9]**(Wah et al., 2011):** This dataset consists of 200 classes and 11,800 data samples. We use the binary attributes and Fisher Vector representations of images as our views. The binary attributes are modeled as mixture-of-multinomials and the Fisher vectors as Gaussian mixtures. We assume diagonal covariances for all views modeled as a mixture of Gaussians in all datasets.

7.3.5.3 Results

Tables 7.2, 7.3, 7.4 and 7.5 show clustering and out-of-sample cluster assignment results for the datasets mentioned in Section 7.3.5.2 in that order. Note that results are marked NA if any of the baseline methods were not extendable to more than two views or could not be compared due to limit-

[8] http://lms.comp.nus.edu.sg/research/NUS-WIDE.htm
[9] http://www.vision.caltech.edu/visipedia/CUB-200-2011.html

ing model assumptions e.g. non-negativity required by NMF-mvc (Liu et al., 2013). The tables only consist of results corresponding to the γ parameter that provided the best results across different choices of γ for both GRECO and LYRIC on a hold-out dataset. Additionally, Figure 7.5 shows performance of GRECO and LYRIC using different Rényi divergences parametrized by $\log(\gamma)$ in comparison with Co-EM, that uses linear aggregation, corresponding to $\gamma \to 1$. The performance across different γ provides further insights into performance of the proposed co-regularization method.

The proposed methods outperform almost all the baselines consistently across different datasets. In addition, hold-out cluster assignment performance is better for both models across most datasets. Improved performance over ensemble methods suggests co-regularization improves on the view-wise clustering approaches. In addition, results also suggest that sharing a single latent variable (see Joint Model) across views is restrictive. In the low bias regime, GRECO has particular advantages over LYRIC because of the additional trade-off in regularization. When the bias across views is low, the additional regularization potentially accelerates convergence by restricting the deviation from view-specific unregularized posteriors, especially when initial model parameters may be noisy. In the high bias case, LYRIC shows some advantage (see Table 7.3-WebKB data). It is important to note that overall, the general trend of performance of both GRECO and LYRIC is consistent for each dataset (see Figure 7.5). In particular, the performance peaks for the most appropriate choice of γ that best captures inherent biases across views for both algorithms for all datasets and this choice of divergence is the same for GRECO as well as LYRIC.

For Twitter data, the γ parameter of 0.01 resulted in the best clustering

accuracy as measured on hold-out set (see Table 7.2). This provides further insight that the views do have some bias in the latent clustering distribution. In the absence of such a bias, the best clustering parameter should have corresponded to $\gamma \to 0$. Thus the value of the divergence parameter γ provides an intuitive understanding of inherent incoherence in clustering beliefs in the data. It is notable that characterizing this bias has resulted in almost an order of magnitude increase in clustering accuracy compared to baselines like multiview NMF and spectral clustering methods. To the best of our knowledge, there is little work in terms of designing robust learning models when underlying model assumptions may be violated. The results on Twitter data strongly highlight the significance of such an approach.

Similar observations on the WebKB data suggests a high degree of incoherence across views on the clustering distributions, suggested by the fact that linear aggregation ($\gamma \to 1$) provides the best results on the hold-out dataset. Note that in such a scenario, i.e. when views completely disagree (in terms of the MAP estimate of the clustering) across views, learning each view independently is equally useful, as demonstrated by competitive performance of Ensemble methods relative to GRECO and LYRIC. Again, this further reinforces the advantage of our model in terms of robustness to violations of model assumptions. Figure 7.5 also suggests that as the underlying bias is assumed increase, the model performance in both LYRIC and GRECO consistently improves. In addition, the improvement over Co-EM at $\gamma \to 1$ suggests that the method proposed to estimate a hold-out clustering assignment using (7.10) is better or comparable to that of Co-EM. Note that although GRECO and LYRIC do not perform the best on training data in terms of NMI and Entropy, the results on hold-out set are competitive - suggesting that the models do not

overfit the training data.

From the results of NUS Wide Object dataset, where 6 views are modeled jointly, the improvement in performance is significant when an appropriate divergence parameter γ is used, as compared to Co-EM, which enforces linear aggregation and the joint model that estimates a single clustering posterior across all views. This further suggests advantages of GRECO and LYRIC when the number of views available is large. The best performing divergence parameter is relatively high ($\gamma = 0.1$). This also suggests that as the number of views being modeled increases, the views are likely to be more incoherent and an assumption of a high bias (higher γ) is a better modeling assumption. This is also apparent from the deteriorated performance of the joint model. Both GRECO and LYRIC perform the best at the limiting case $w_g = 1$ as expected in a slightly high bias case, when additional regularization of GRECO is not necessarily advantageous. Figure 7.5 also suggests that at lower values of γ both LYRIC and GRECO may be getting stuck in local minima (suggested by the high observed variance at $\gamma = 0.01$) potentially reflecting sensitivity to choice of γ for this data.

For a large dataset like CUB-200-2011 with 200 clusters and ∼11,000 samples and high dimensionality (∼8000), the improvement in unsupervised learning performance of GRECO and LYRIC is more pronounced compared to Co-EM even though the best performance is obtained at $\gamma \to 1$. This suggests that our inference on hold-out set works better than Co-EM. Further, the best performance divergence parameter $\gamma \to 1$ suggests the attribute view and the Fisher vector views, used from the CUB_200_2011 data, are potentially incoherent in terms of the latent clustering distribution. Comparison to other probabilistic methods, i.e. Joint model and Ensemble model, suggest

Table 7.2: Twitter data (politics-uk, 3 views), best results obtained for $\gamma = 0.01$ for GRECO and LYRIC. Ensemble model, CCA-mvc, Min-dis(Sp) can cluster at most two views and marked NA otherwise. Co-reg(Sp), Min-dis(Sp) and NMF-mvc do not explicitly compare hold-out cluster assignment results and have not been compared to for hold-out assignment performance. Top two methods w.r.t. each metric are highlighted.

Clustering Results							
Method	Accuracy	Precision	Recall	F-measure	NMI	Entropy	Time (sec.)
GRECO	**0.9075(0.0201)**	**0.9403(0.0316)**	**0.8713(0.0366)**	**0.9039(0.0217)**	**0.7887(0.0478)**	**0.2971(0.1001)**	7.1241(1.0122)
LYRIC	**0.886(0.0284)**	**0.9601(0.01)**	0.8441(0.0596)	**0.8976(0.0372)**	**0.8045(0.0403)**	**0.2434(0.0431)**	7.0888(0.9755)
Co-EM	0.8346(0.0488)	0.8973(0.0346)	0.7559(0.0757)	0.8197(0.0566)	0.7058(0.0406)	0.3876(0.0536)	2.3714(0.9746)
Joint	0.7893(0.0491)	0.7737(0.0792)	0.7167(0.0679)	0.7413(0.0535)	0.5806(0.053)	0.6497(0.11)	**0.3623(0.1047)**
Ensemble	NA	NA	NA	NA	NA	NA	NA
Co-reg(Sp)	0.557(0.0221)	0.7122(0.0215)	0.4326(0.0197)	0.5382(0.0213)	0.5079(0.018)	0.6329(0.0293)	**1.6324(0.1538)**
CCA-mvc	NA	NA	NA	NA	NA	NA	NA
Min-dis(Sp)	NA	NA	NA	NA	NA	NA	NA
NMF-mvc	0.4418(0)	0.3802(0)	**0.972(0)**	0.5466(0)	0.0161(0)	1.5769(0)	6.0709(0.0895)
Hold-out Cluster Assignment Results							
GRECO	**0.9238(0.0136)**	**0.9047(0.0384)**	**0.9021(0.0559)**	**0.9022(0.0307)**	**0.7784(0.0418)**	**0.3417(0.0703)**	NA
LYRIC	**0.8452(0.0854)**	**0.8537(0.0283)**	**0.8123(0.1353)**	**0.8291(0.0888)**	**0.6803(0.0635)**	**0.4438(0.0697)**	NA
Co-EM	0.781(0.0287)	0.8282(0.0406)	0.6735(0.0425)	0.7425(0.0379)	0.5916(0.0566)	0.4988(0.1454)	NA
Joint	0.769(0.0644)	0.6629(0.064)	0.7093(0.1399)	0.6797(0.0828)	0.4916(0.0829)	0.7895(0.1495)	NA
Ensemble	NA	NA	NA	NA	NA	NA	NA
CCA-mvc	NA	NA	NA	NA	NA	NA	NA

restrictive model assumptions may fail and general methods like GRECO and LYRIC may be more reliable in large scale settings. Ensemble model also relies on Hungarian matching to solve the correspondence problem between cluster indices (200 clusters) across views. Improved performance in GRECO and LYRIC is obtained at a significant computational cost compared to CCA-mvc which provides comparable performance very fast. This corroborates the model assumptions made by CCA-mvc, namely that views of a sample are uncorrelated conditioned on cluster identity of sample (weaker assumptions than those made by the Joint model) can provide improvement in unsupervised learning performance. Faster inference for GRECO/LYRIC in such settings can be obtained by parallelization and/or any improvements to the variational inference procedure used to impose co-regularization.

Overall, the best Rényi divergence suitable for a particular dataset

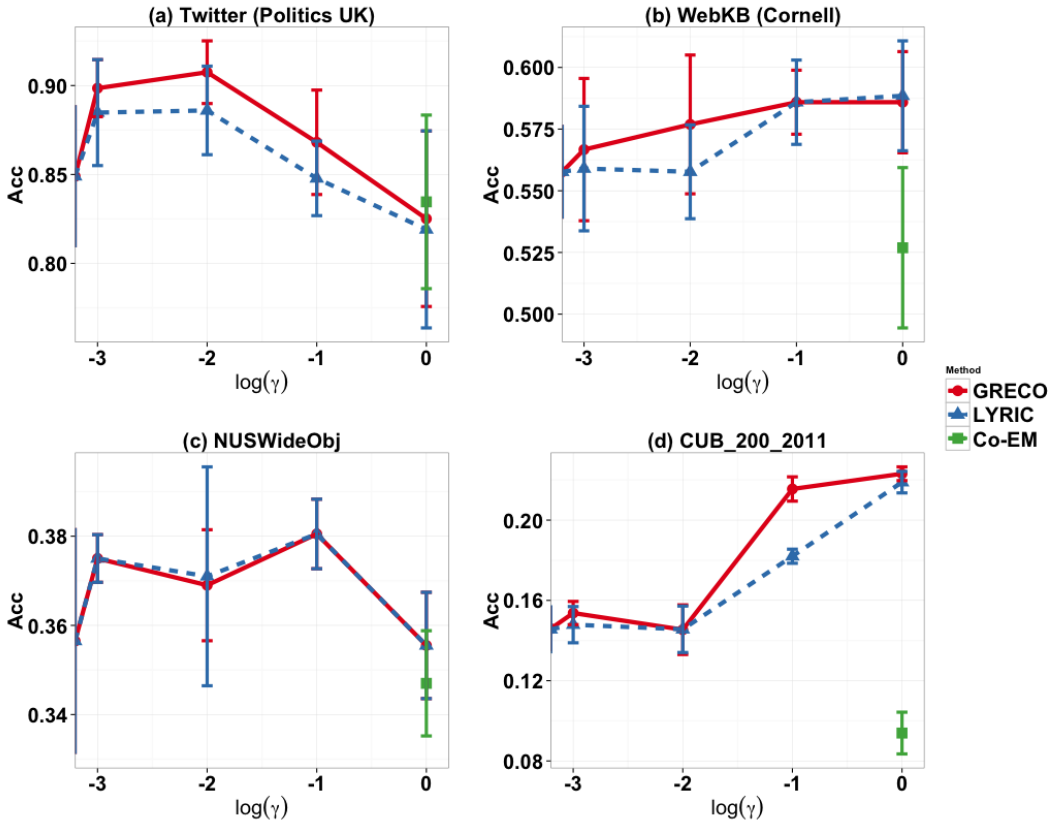


Figure 7.5: Clustering Accuracy of GRECO and LYRIC w.r.t. $\log \gamma$ on (a) Twitter data (b) WebKB data (c) NUSWideObj data and (d) CUB_200_2011 data

differs, indicating that GRECO and LYRIC capture potential differences in coherence between views with respect to cluster memberships significantly better than comparable methods. The biases between views demonstrably affect clustering performance. This also suggests that the multiview assumption of a single underlying cluster membership distribution is not always satisfied in real data. Thus flexible models such as GRECO and LYRIC are preferable. All results further show that the choice of the class of Rényi divergences is beneficial for improving multiview clustering performance and both methods

Table 7.3: Cornell (WebKB 2-views), best results obtained for $\gamma = 0.1$ for GRECO and $\gamma \to 1$ for LYRIC

Clustering Results							
Method	Accuracy	Precision	Recall	F-measure	NMI	Entropy	Time(sec.)
GRECO	**0.5859(0.0148)**	**0.431(0.0385)**	0.617(0.0451)	**0.5066(0.0327)**	0.2747(0.0145)	1.5578(0.0379)	0.3404(0.0323)
LYRIC	**0.5885(0.0254)**	0.4135(0.0351)	**0.6591(0.0292)**	**0.5075(0.0295)**	**0.2771(0.024)**	1.5697(0.0489)	0.3174(0.03)
Co-EM	0.5269(0.0325)	0.3753(0.0324)	0.5485(0.0777)	0.4432(0.0374)	0.1908(0.0187)	1.7216(0.0366)	0.8036(0.1299)
Joint	0.4179(0.025)	0.3232(0.0184)	0.4805(0.0849)	0.3846(0.0334)	0.1405(0.0084)	1.8257(0.0051)	0.1855(0.0178)
Ensemble	0.5064(0.0304)	0.3535(0.0199)	0.6008(0.1335)	0.4376(0.0326)	0.2099(0.0352)	1.7026(0.0592)	**0.0341(0.0011)**
Co-reg(Sp)	0.5551(0.0494)	**0.5083(0.0157)**	0.4596(0.0354)	0.4824(0.0252)	**0.3929(0.0167)**	**1.2719(0.0386)**	2.065(0.0201)
CCA-mvc	0.4526(0.014)	0.3118(7e-04)	0.4751(0.0304)	0.3762(0.0095)	0.1665(0.0019)	**1.2664(0.0903)**	**0.0786(0.0188)**
Min-dis(Sp)	0.3756(0.0154)	0.32(0.0023)	0.3116(0.0524)	0.3139(0.0251)	0.1614(0.0048)	1.7744(0.0207)	0.0904(0.0366)
NMF-mvc	0.4103(0)	0.2606(0)	**0.9605(0)**	0.41(0)	0.0569(0)	2.0497(0)	5.6911(0)
Hold-out Cluster Assignment Results							
GRECO	0.4513(0.0739)	0.2995(0.051)	0.5782(0.1985)	0.3872(0.0795)	0.1777(0.0573)	1.7211(0.147)	NA
LYRIC	**0.5026(0.0693)**	**0.3493(0.0683)**	0.5541(0.2054)	**0.4238(0.1034)**	**0.2223(0.1096)**	**1.63(0.2153)**	NA
Co-EM	0.4205(0.0862)	0.2788(0.0606)	0.538(0.1851)	0.3626(0.0908)	0.1762(0.035)	1.7269(0.0966)	NA
Joint	0.4564(0.0585)	0.2861(0.0344)	**0.6214(0.0806)**	0.39(0.0391)	0.1934(0.0583)	1.7096(0.0844)	NA
Ensemble	**0.5487(0.1082)**	**0.4123(0.1742)**	**0.7356(0.107)**	**0.5016(0.1051)**	**0.2981(0.1633)**	**1.5027(0.407)**	NA
CCA-mvc	0.4103(0)	0.3103(0.007)	0.4(0.0123)	0.3494(0.0074)	0.1192(0.0191)	1.7107(0.0361)	NA

Table 7.4: NUSWideObj Dataset (6 views), best results obtained for γ= 0.1 for GRECO and LYRIC. Since this data has three views that take negative values, we do not compare against NMF-mvc. CCA-mvc and Min-dis(Sp) cannot be extended for more than two views.

Clustering Results							
Method	Accuracy	Precision	Recall	F-measure	NMI	Entropy	Time (sec.)
GRECO	**0.3805(0.0089)**	**0.245(0.0058)**	**0.3362(0.0347)**	**0.2829(0.0146)**	**0.3276(0.0199)**	**2.2687(0.0574)**	8.0385 (1.2579)
LYRIC	**0.3805(0.0089)**	**0.245(0.0058)**	**0.3362(0.0347)**	**0.2829(0.0146)**	**0.3276(0.0199)**	**2.2687(0.0574)**	8.0099(1.2586)
Co-EM	0.347(0.0118)	0.2171(0.011)	0.3006(0.0184)	0.2518(0.0092)	0.2903(0.0089)	2.3918(0.0319)	4.3041(0.7188)
Joint	0.3115(0.0151)	0.1882(0.016)	0.346(0.0303)	0.2437(0.0202)	0.2454(0.0157)	2.5884(0.0481)	**2.8231(0.7605)**
Ensemble	NA	NA	NA	NA	NA	NA	NA
Co-reg(Sp)	0.3785(0.0202)	0.2629(0.0128)	0.2816(0.0196)	0.2718(0.0153)	0.318(0.0162)	2.273(0.0531)	**2.5275(0.0541)**
CCA-mvc	NA	NA	NA	NA	NA	NA	NA
Min-dis(Sp)	NA	NA	NA	NA	NA	NA	NA
NMF-mvc	NA	NA	NA	NA	NA	NA	NA
Hold-out Cluster Assignment Results							
GRECO	**0.412(0.0409)**	**0.225(0.0228)**	**0.3369(0.0177)**	**0.2691(0.017)**	**0.4178(0.0246)**	**1.9934(0.0893)**	NA
LYRIC	**0.412(0.0409)**	**0.225(0.0228)**	**0.3369(0.0177)**	**0.2691(0.017)**	**0.4178(0.0246)**	**1.9934(0.0893)**	NA
Co-EM	0.372(0.0217)	0.2074(0.0232)	0.2964(0.0405)	0.2437(0.0289)	0.3975(0.026)	2.052(0.0856)	NA
Joint	0.334(0.0241)	0.1806(0.019)	0.352(0.0374)	0.2387(0.0248)	0.329(0.0294)	2.3533(0.092)	NA
Ensemble	NA	NA	NA	NA	NA	NA	NA
CCA-mvc	NA	NA	NA	NA	NA	NA	NA

generalize better to unseen data compared to baselines.

A comparison of training time suggests that the increased accuracy of GRECO and LYRIC is obtained at the cost of increased training time. However, the variational update required for co-regularization is the major con-

Table 7.5: CUB-200-2011 (2 views), best results obtained for $\gamma \to 1$ for GRECO and LYRIC. Since this data has a view that takes negative values, we do not compare against NMF-mvc.

Clustering Results							
Method	Accuracy	Precision	Recall	F-measure	NMI	Entropy	Time (sec.)
GRECO	**0.2231(0.0039)**	**0.1052(0.0034)**	**0.1757(0.005)**	**0.1316(0.0038)**	**0.5109(0.006)**	3.8498(0.0508)	2255.5(169.34)
LYRIC	0.2189(0.0061)	0.099(0.004)	**0.1748(0.005)**	**0.1264(0.0036)**	**0.5071(0.0051)**	**3.8867(0.045)**	2069.6(143.74)
Co-EM	0.0939(0.0104)	0.0111(0.0014)	0.0891(0.0135)	0.0197(0.0019)	0.301(0.0146)	5.5905(0.1318)	3355.9(2382.4)
Joint	0.0715(0.0035)	0.0109(2e-04)	0.0582(0.0035)	0.0183(2e-04)	0.2473(0.0063)	5.9822(0.0511)	2004.1(124.85)
Ensemble	0.0432(9e-04)	0.0084(3e-04)	0.0809(0.0119)	0.0151(3e-04)	0.1756(0.0067)	6.5442(0.0589)	767.78(56.32)
Co-reg(Sp)	0.2118(0.0081)	**0.1031(0.0042)**	0.118(0.0053)	0.11(0.0046)	0.4896(0.0059)	3.9224(0.0431)	901.21(11.716)
CCA-mvc	**0.2213(0.007)**	0.0759(0.0066)	0.1527(0.0069)	0.1012(0.006)	0.5003(0.0038)	**3.4551(0.0454)**	**4.8814(0.1651)**
Min-dis(Sp)	0.1994(0.0093)	0.0795(0.0043)	0.1214(0.0077)	0.0961(0.0054)	0.4691(0.0055)	4.1377(0.0408)	**594.78(20.514)**
NMF-mvc	NA	NA	NA	NA	NA	NA	NA
Hold-out Cluster Assignment Results							
GRECO	**0.2133(0.0078)**	**0.0601(0.0046)**	**0.1304(0.008)**	**0.0822(0.0057)**	**0.57(0.0048)**	**3.4714(0.0417)**	NA
LYRIC	0.2066(0.0085)	0.0531(0.0027)	0.1284(0.0045)	0.0751(0.0025)	0.5644(0.0043)	3.5276(0.0417)	NA
Co-EM	0.0712(0.0712)	0.0086(0.0086)	0.1208(0.1208)	0.0159(0.0159)	0.3347(0.02)	5.5129(0.1822)	NA
Joint	0.0603(0.0603)	0.0093(0.0093)	0.0671(0.0671)	0.0163(0.0163)	0.3259(0.0116)	5.5296(0.0935)	NA
Ensemble	0.0508(0.0508)	0.0088(0.0088)	0.09(0.09)	0.016(0.016)	0.2808(0.0182)	5.8884(0.1399)	NA
CCA-mvc	**0.2512(0.0064)**	**0.0727(0.0048)**	**0.1444(0.0081)**	**0.0965(0.004)**	**0.6043(0.0023)**	**3.158(0.0248)**	NA

tributing factor to training time. Since these updates can be trivially executed in a distributed setting across samples as well as for estimating unnormalized cluster membership distributions, the training time can be easily improved. Further, any alternative inference procedure to solve the co-regularization constraint will directly improve training times for the proposed method. Also note that training times are comparable to Co-EM and other baselines for special cases (see Tables 7.3 and 7.5).

Additional advantages of GRECO and LYRIC compared to other methods are noteworthy. Both Twitter and WebKB datasets consist of at least one view with relational data. The twitter data is sparse (as is the case with social network data), i.e., a lot of the entries are 0. In these cases probabilistic methods outperform other methods suggesting the importance of probabilistic models in general. The NUS Wide Object dataset and CUB datasets have mixed views, i.e. *bag-of-words* as well as numeric features (e.g. Fisher vector

representations). Empirical evaluation also demonstrates that our methods handle mixed data well.

Some limitations of the proposed methods arise in selecting an appropriate choice of weights and the best suited Rényi divergence parameter for a given dataset. Storkey et al. (2014) have proposed a method for automatic selection of weights which can be easily incorporated in GRECO or LYRIC via minor changes to the variational procedures described in Appendix A. However, we chose to use manual selection of weights in order to highlight significance of the choice of Rényi divergences as opposed to a finer choice of weights, especially to highlight the generalization over Co-EM. Note that automatic selection or learning the best divergence parameter in an unsupervised setting suitable for a given data is a challenging and novel problem that we expose. Particularly, conventional model selection methods that trade-off model complexity and likelihood are not applicable in this scenario as model complexity does not change w.r.t. different γ. Automatic selection of such a model parameter is deferred to future work. However, we point out that both GRECO and LYRIC provide better performance compared to all existing baselines for all choices of γ that we tested. A more appropriate choice of γ further boosts performance.

7.4 Conclusion

This chapter proposes a constraint based method for semi-supervised learning. We propose algorithms for LeTOR as well as clustering domain. We assume the existence of heterogeneous data sources or views, as is quite common in practice. To leverage these views in lieu of expert annotation, we constrain the view specific model parameters to agree on the target rank

ordering for the ranking task or the distributions over the latent cluster memberships for the clustering task. We propose a co–regularization technique to constrain the views to encourage such agreement across views. The proposed method demonstrated competitive and/or state of the art performance with respective baselines for both the ranking as well as the clustering task on multiple datasets. Thus, constraint based methods like co-regularization are useful in leveraging additional information to learn reliably in the absence of expert annotation.

Chapter 8

Conclusions and Future Work

8.1 Conclusions

This dissertation addresses the task of developing interpretable, explainable, and semi-supervised machine learning models. We develop our framework using latent variable models. We argue that constraining different aspects of latent variable models in novel ways allows to address the aforementioned practical machine learning challenges. The constraints can be applied to realizations of sample specific latent representations, model parameters and/or as probabilistic generative assumptions on the data.

In particular, we motivate interpretable models as a class of models that generate outcomes that satisfy specific interpretability criterion. For this class of models, interpretability is encouraged by constraining individual realizations of latent variables. In order to constrain the latent variables, we show that algorithms under the class of Majorization–Maximization, i.e. those that do not marginalize our the latent variables as the most amenable class of algorithms. Chapter 3 discusses a general abstraction of such an interpretable machine learning framework, including potential advantages and/or disadvantages.

The proposed framework is evaluated on clinical healthcare, our primary application domain. We focus on phenotyping for chronic conditions of an ICU patient population using clinical notes. A *grounding* framework is

proposed in Chapter 3 as a means to ensure the proposed phenotyping models are physiologically relevant to the target chronic conditions. Chapter 4 discusses how the *grounding* framework can be incorporated in our model using computable weak diagnoses that are easily available in administrative clinical data. This suggests that expensive human annotations is not a requirement to make these models inherently interpretable. Chapter 4 also discuss in detail how such weak diagnoses can be extracted from clinical data and the procedure for extracting clinical notes to generate observational data. We discuss this in the context each of the proposed models, namely, 1. *Grounded* NMF (see 5.1) and 2. *Grounded* admixtures of PMRFs (see 5.2) .

Chapter 5 discusses each of the proposed models in detail. In particular, for each model, the interpretability constraints are represented as constraints on latent realizations as well as model parameters within each framework. An algorithm is proposed in each case to tractably learn model parameters. An updated inference procedure is proposed, if necessary. Each model is learned to recover phenotypes for chronic conditions for ICU patients as represented by the data processing detailed in Chapter 4. Each model is extensively evaluated in terms of two main criteria: (a) The quality of learned phenotypes in terms of their clinical relevance in comparison to competitive baselines, thereby evaluating the potential of the proposed *grounding* framework, and (b) The predictive power of the learned phenotype representations in determining patient outcome, thereby quantitatively evaluating how model performance is affected by the proposed interpretability criteria . The evaluation demonstrates that the learned phenotypic representations are qualitatively better as well as competitive in terms of predicting patient outcome and/or diagnoses.

Explainable Machine Learning refers to models that explain decisions of

existing complex models at a desirable abstraction. We demonstrate a method to provide such explanations using examples. The proposed method relies on constraining the generation of examples so that the manifold around the decision boundary can be explored in novel ways. Chapter 6 details the algorithm used to constrain sample generation by characterizing the data manifold. We demonstrate how this procedure can be used to provide explanations for complex black-box models, like deep learning models, when other performance metrics may be comparable.

In Chapter 7, we discuss how constraints can be used to learn ranking models as well as clustering models in the presence of little to no supervision. In particular, we leverage the available heterogeneous data sources for semi-supervised learning. We encourage models to agree on solutions across all data sources over samples where annotation is not available via constraints. These constraints, when imposed during training, is known as co–regularization. We design novel clustering and a listwise LeTOR models using this general framework. The proposed methods are extensively compared to existing baselines, in each case to demonstrate reliable performance on web-document clustering, social media, as well as information retrieval applications.

8.2 Future Work

One can also generalize the interpretability framework to not only apply the proposed framework to other relevant class of models, but also to applications beyond clinical healthcare (like fMRI, climate science data, etc.). In particular, this requires generalizing the *grounding* framework in order to incorporate other application specific domains. However, the general abstraction as well as the general insight remains the same. An interesting and compelling

direction of theoretical research is to analyze the identifiability of *grounding* framework in all the proposed latent variable models.

One can also formulate a representation of *explanations* for black-box models in multiple ways. Particularly, we could extend the explanations framework beyond relying on individual examples. Additionally, drawing from Miller (2017), we can develop *explanations* using *counterfactuals* using the causal inference framework (Pearl, 2009). In general, we posit that these framework are amenable to personalized explanations, especially relevant for clinical healthcare.

Finally, theoretically analyzing the class of *multiview* models, particularly clustering as well as LeTOR models (proposed in Chapter 7) remains a challenging problem. In particular, as observed in 7.3, the choice of Bregman Divergences crucially affects clustering performance. We conjecture that this may be associated in the amount of bias *views* may have with respect to the underlying clustering distribution. Similarly, while supervised listwise LeTOR using Monotone Retargeting has been theoretically analyzed for convergence by Acharyya et al. (2012); Acharyya and Ghosh (2014), analyzing the semi–supervised version as proposed in Chapter 7 has not, in part due to what we call the *dynamic shifting* problem of the conic set of isotonic vectors as model training progresses. Further, generalizing MR-CORE to other divergence functions as well as scaling the method to large scale web dataset is a compelling future direction.

Appendices

Appendix A

Phenotyping using Grounded NMF

Additional Results for Phenotyping using *Grounded* NMF (Chapter 5) are provided in the following.

A.1 Phenotype sparsity

As suggested in Section 5.1.4.1, there is an inherent trade off between fit to the cost function and desired sparsity. The trade-off is made explicit for λ–CNMF in Figure A.1. The sparsity of LLDA is controlled by tuning the hyperparameter (β) of the word-topic multinomial parameters (Blei et al., 2003) and for MLC via the ℓ_1 regularization parameter η. A smaller value of β ensures that the word-topic probabilities are sparse. As the value of β is increased, sparsity decreases (i.e. number of non-zero elements increases). For logistic regression (used by MLC), as the ℓ_1 regularization parameter increases, sparsity increases. Figure A.2(a) demonstrates the sparsity of the estimated phenotypes for LLDA and Figure A.2(b) shows that of logistic regression. We choose phenotypes obtained at $\beta = 1 \times 10^{-8}$ and $\eta = 100$ for qualitative annotation. The parameters were chosen to achieve the lowest median sparsity while ensuring that for each chronic condition, the corresponding phenotype candidate is represented by at least 5 non-zero clinical terms. Our fourth baseline (NMF + support) did not estimate sparse phenotypes and does not have a tuneable sparsity parameter (but were nevertheless annotated for qual-

itative evaluation). The proposed model provides the best sparsity among all baselines.

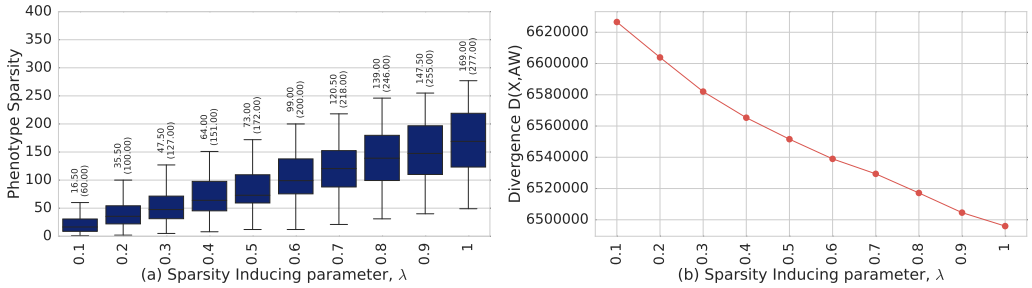
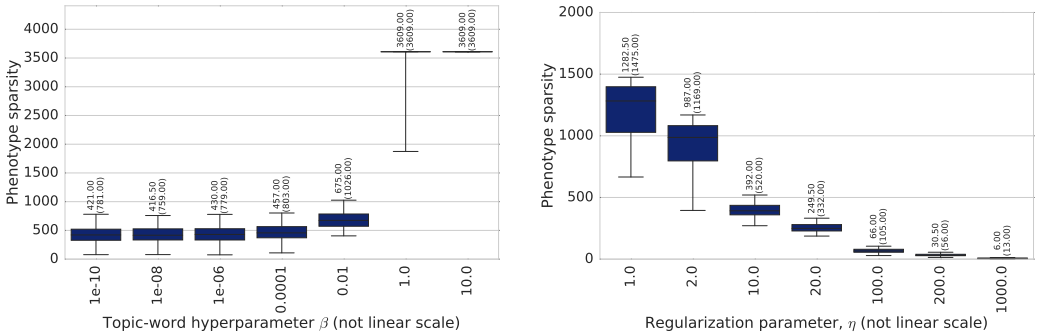


Figure A.1: Sparsity–Accuracy Trade–off. Sparsity of the model is measured as the median of the number of non-zero entries in columns of the phenotype matrix $\mathbf{A}$. (a) shows a box plots of the median sparsity across the 30 chronic conditions for varying λ values. The median and third–quartile values are explicitly noted on the plots. (b) divergence function value of the estimate from Algorithm 3 plotted against λ parameter.

A.2 Sample phenotypes for baseline models

Figures A.3–A.29 show the top 15 terms learned for all target chronic conditions for the proposed model and baselines. The sparsity level chosen is based on the criterion described in Section 5.1.4.1. For all conditions, the terms are ordered in decreasing order of importance as learned by the models.



(a) LLDA (b) MLC

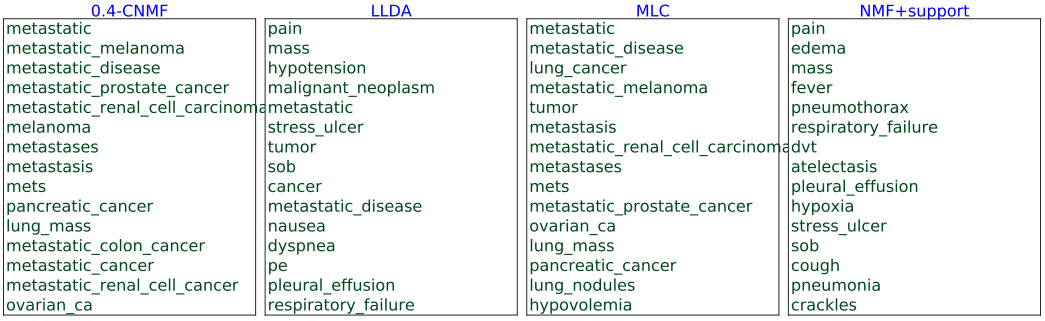
Figure A.2: Phenotype sparsity for baseline models

0.4-CNMF	LLDA	MLC	NMF+support
cirrhosis	cirrhosis	cirrhosis	pain
varices	ascites	hepatitis_c	pneumothorax
portal_hypertension	bleeding	ascites	atelectasis
hepatitis_c	gi_bleed	liver_failure	pleural_effusion
esophageal_varices	gib	hep_c	ascites
cirrhosis_of_liver	hepatic_encephalopathy	cryptogenic_cirrhosis	bleeding
gastric_varices	varices	fatty_liver	edema
alcoholic_cirrhosis	encephalopathy	etoh_abuse	pneumonia
hep_c	gastrointestinal_bleed	autoimmune_hepatitis	liver_failure
hepatocellular_carcinoma	altered_mental_status	colitis	cough
spontaneous_bacterial_peritonit	abdominal_pain	dyspnea	afebrile
portal_hypertensive_gastropathy	liver_failure	withdrawal	hypotension
ascites	hypotension	volume_overload	chf
end_stage_liver_disease	renal_failure	endocarditis	bm
primary_biliary_cirrhosis	esophageal_varices	end_stage_liver_disease	free_fluid

Figure A.3: Learned Phenotypes for Liver Disease

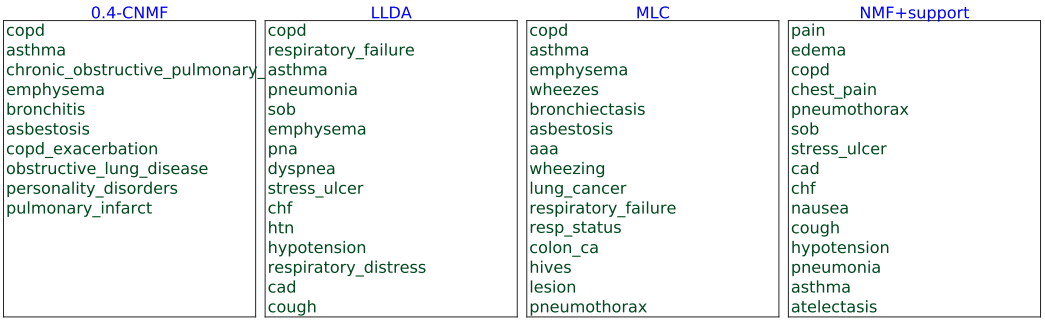
0.4-CNMF	LLDA	MLC	NMF+support
pancreatic_cancer	bleeding	hepatocellular_carcinoma	bleeding
ovarian_cancer	pain	thyroid_ca	pain
metastatic_ovarian_cancer	pericardial_effusion	brain_tumor	nausea
pelvic_mass	mass	glioblastoma	dvt
glioblastoma	hypotension	end_stage_liver_disease	chest_pain
brain_tumor	stress_ulcer	calcifications	edema
nsclc	dvt	prostate_cancer	gi_bleed
neoplasm	abdominal_pain	cancer	cad
abdominal_mass	edema	bladder_ca	gib
hepatocellular_carcinoma	hemoptysis	ovarian_cancer	vomiting
bladder_ca	malignant_neoplasm	pancreatic_cancer	diverticulosis
chemoradiation	cancer	incisional_pain	hypotension
mets	sob	colon_cancer	stress_ulcer
bronchopleural_fistula	chest_pain	lung_cancer	abdominal_pain
partial_obstruction	pe	tumor	bleed

Figure A.4: Learned Phenotypes for Solid Tumor



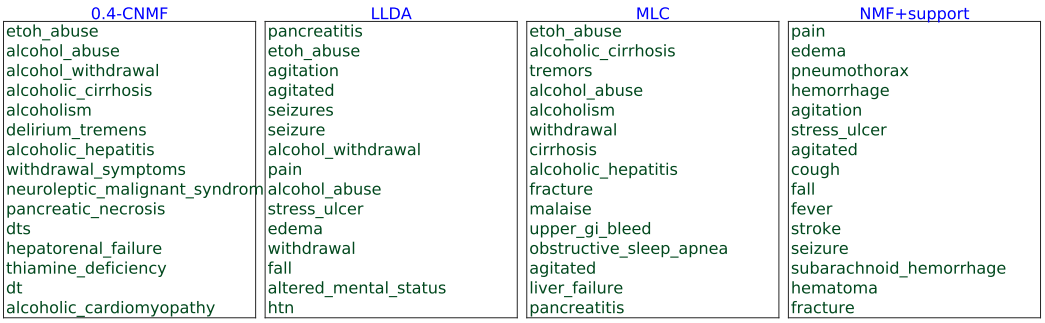
0.4-CNMF	LLDA	MLC	NMF+support
metastatic	pain	metastatic	pain
metastatic_melanoma	mass	metastatic_disease	edema
metastatic_disease	hypotension	lung_cancer	mass
metastatic_prostate_cancer	malignant_neoplasm	metastatic_melanoma	fever
metastatic_renal_cell_carcinoma	metastatic	tumor	pneumothorax
melanoma	stress_ulcer	metastasis	respiratory_failure
metastases	tumor	metastatic_renal_cell_carcinoma	dvt
metastasis	sob	metastases	atelectasis
mets	cancer	mets	pleural_effusion
pancreatic_cancer	metastatic_disease	metastatic_prostate_cancer	hypoxia
lung_mass	nausea	ovarian_ca	stress_ulcer
metastatic_colon_cancer	dyspnea	lung_mass	sob
metastatic_cancer	pe	pancreatic_cancer	cough
metastatic_renal_cell_cancer	pleural_effusion	lung_nodules	pneumonia
ovarian_ca	respiratory_failure	hypovolemia	crackles

Figure A.5: Learned Phenotypes for Metastatic Cancer



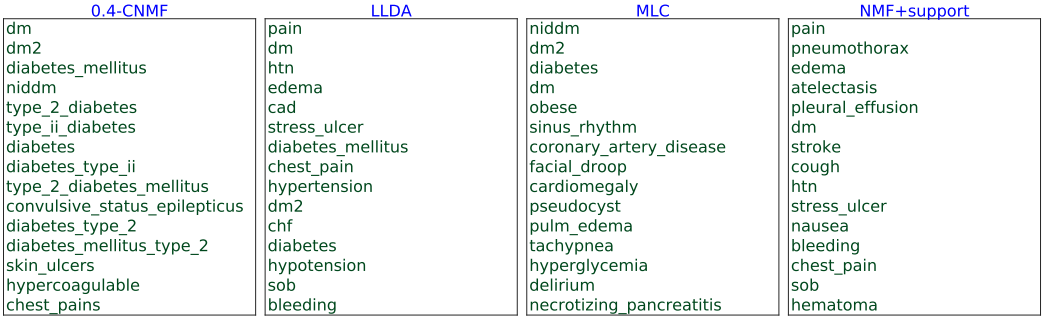
0.4-CNMF	LLDA	MLC	NMF+support
copd	copd	copd	pain
asthma	respiratory_failure	asthma	edema
chronic_obstructive_pulmonary_	asthma	emphysema	copd
emphysema	pneumonia	wheezes	chest_pain
bronchitis	sob	bronchiectasis	pneumothorax
asbestosis	emphysema	asbestosis	sob
copd_exacerbation	pna	aaa	stress_ulcer
obstructive_lung_disease	dyspnea	wheezing	cad
personality_disorders	stress_ulcer	lung_cancer	chf
pulmonary_infarct	chf	respiratory_failure	nausea
	htn	resp_status	cough
	hypotension	colon_ca	hypotension
	respiratory_distress	hives	pneumonia
	cad	lesion	asthma
	cough	pneumothorax	atelectasis

Figure A.6: Learned Phenotypes for Chronic Pulmonary Disorder



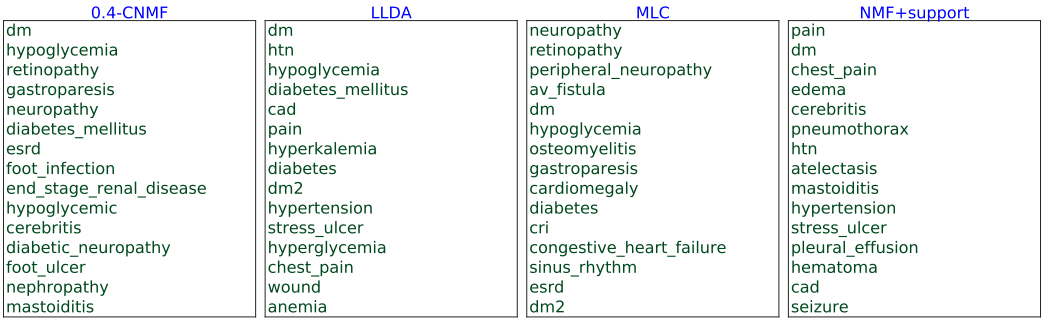
0.4-CNMF	LLDA	MLC	NMF+support
etoh_abuse	pancreatitis	etoh_abuse	pain
alcohol_abuse	etoh_abuse	alcoholic_cirrhosis	edema
alcohol_withdrawal	agitation	tremors	pneumothorax
alcoholic_cirrhosis	agitated	alcohol_abuse	hemorrhage
alcoholism	seizures	alcoholism	agitation
delirium_tremens	seizure	withdrawal	stress_ulcer
alcoholic_hepatitis	alcohol_withdrawal	cirrhosis	agitated
withdrawal_symptoms	pain	alcoholic_hepatitis	cough
neuroleptic_malignant_syndrom	alcohol_abuse	fracture	fall
pancreatic_necrosis	stress_ulcer	malaise	fever
dts	edema	upper_gi_bleed	stroke
hepatorenal_failure	withdrawal	obstructive_sleep_apnea	seizure
thiamine_deficiency	fall	agitated	subarachnoid_hemorrhage
dt	altered_mental_status	liver_failure	hematoma
alcoholic_cardiomyopathy	htn	pancreatitis	fracture

Figure A.7: Learned Phenotypes for Alcohol Abuse



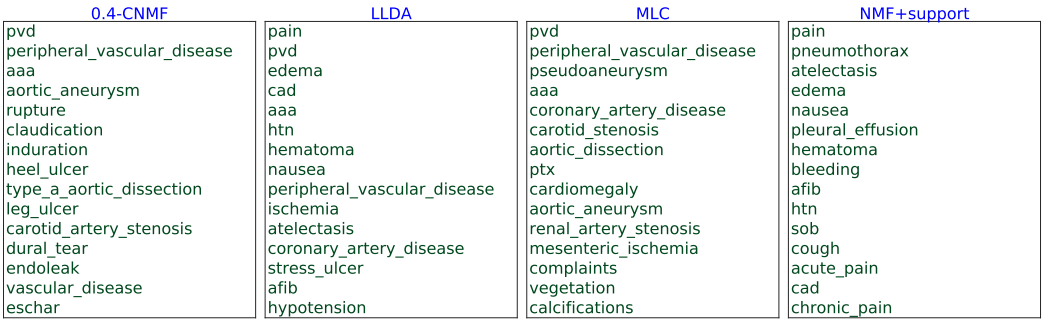
0.4-CNMF	LLDA	MLC	NMF+support
dm	pain	niddm	pain
dm2	dm	dm2	pneumothorax
diabetes_mellitus	htn	diabetes	edema
niddm	edema	dm	atelectasis
type_2_diabetes	cad	obese	pleural_effusion
type_ii_diabetes	stress_ulcer	sinus_rhythm	dm
diabetes	diabetes_mellitus	coronary_artery_disease	stroke
diabetes_type_ii	chest_pain	facial_droop	cough
type_2_diabetes_mellitus	hypertension	cardiomegaly	htn
convulsive_status_epilepticus	dm2	pseudocyst	stress_ulcer
diabetes_type_2	chf	pulm_edema	nausea
diabetes_mellitus_type_2	diabetes	tachypnea	bleeding
skin_ulcers	hypotension	hyperglycemia	chest_pain
hypercoagulable	sob	delirium	sob
chest_pains	bleeding	necrotizing_pancreatitis	hematoma

Figure A.8: Learned Phenotypes for Diabetes Uncomplicated



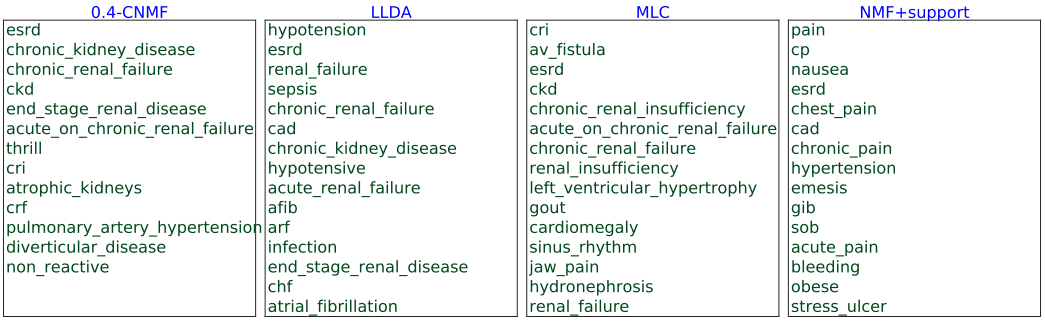
0.4-CNMF	LLDA	MLC	NMF+support
dm	dm	neuropathy	pain
hypoglycemia	htn	retinopathy	dm
retinopathy	hypoglycemia	peripheral_neuropathy	chest_pain
gastroparesis	diabetes_mellitus	av_fistula	edema
neuropathy	cad	dm	cerebritis
diabetes_mellitus	pain	hypoglycemia	pneumothorax
esrd	hyperkalemia	osteomyelitis	htn
foot_infection	diabetes	gastroparesis	atelectasis
end_stage_renal_disease	dm2	cardiomegaly	mastoiditis
hypoglycemic	hypertension	diabetes	hypertension
cerebritis	stress_ulcer	cri	stress_ulcer
diabetic_neuropathy	hyperglycemia	congestive_heart_failure	pleural_effusion
foot_ulcer	chest_pain	sinus_rhythm	hematoma
nephropathy	wound	esrd	cad
mastoiditis	anemia	dm2	seizure

Figure A.9: Learned Phenotypes for Diabetes Complicated



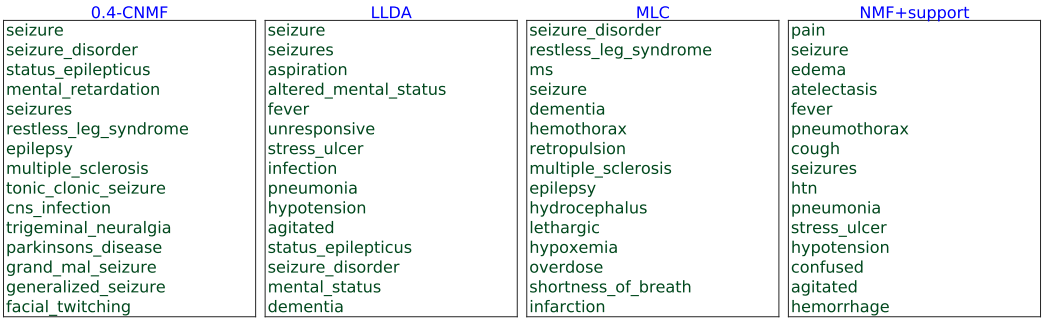
0.4-CNMF	LLDA	MLC	NMF+support
pvd	pain	pvd	pain
peripheral_vascular_disease	pvd	peripheral_vascular_disease	pneumothorax
aaa	edema	pseudoaneurysm	atelectasis
aortic_aneurysm	cad	aaa	edema
rupture	aaa	coronary_artery_disease	nausea
claudication	htn	carotid_stenosis	pleural_effusion
induration	hematoma	aortic_dissection	hematoma
heel_ulcer	nausea	ptx	bleeding
type_a_aortic_dissection	peripheral_vascular_disease	cardiomegaly	afib
leg_ulcer	ischemia	aortic_aneurysm	htn
carotid_artery_stenosis	atelectasis	renal_artery_stenosis	sob
dural_tear	coronary_artery_disease	mesenteric_ischemia	cough
endoleak	stress_ulcer	complaints	acute_pain
vascular_disease	afib	vegetation	cad
eschar	hypotension	calcifications	chronic_pain

Figure A.10: Learned Phenotypes for Peripheral Vascular Disorder



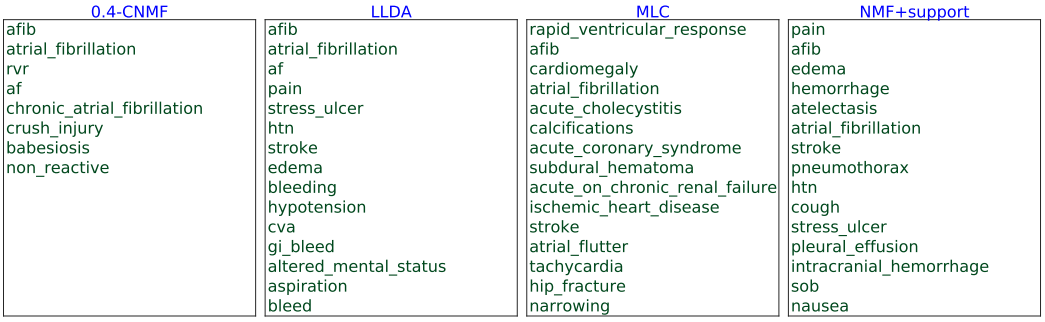
0.4-CNMF	LLDA	MLC	NMF+support
esrd	hypotension	cri	pain
chronic_kidney_disease	esrd	av_fistula	cp
chronic_renal_failure	renal_failure	esrd	nausea
ckd	sepsis	ckd	esrd
end_stage_renal_disease	chronic_renal_failure	chronic_renal_insufficiency	chest_pain
acute_on_chronic_renal_failure	cad	acute_on_chronic_renal_failure	cad
thrill	chronic_kidney_disease	chronic_renal_failure	chronic_pain
cri	hypotensive	renal_insufficiency	hypertension
atrophic_kidneys	acute_renal_failure	left_ventricular_hypertrophy	emesis
crf	afib	gout	gib
pulmonary_artery_hypertension	arf	cardiomegaly	sob
diverticular_disease	infection	sinus_rhythm	acute_pain
non_reactive	end_stage_renal_disease	jaw_pain	bleeding
	chf	hydronephrosis	obese
	atrial_fibrillation	renal_failure	stress_ulcer

Figure A.11: Learned Phenotypes for Renal Failure



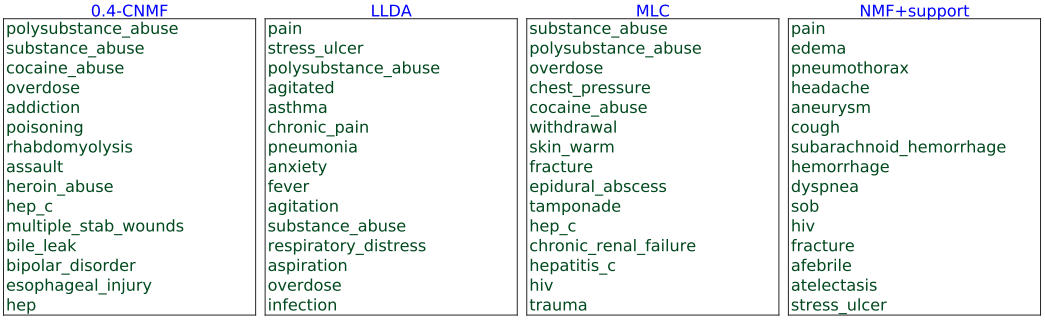
0.4-CNMF	LLDA	MLC	NMF+support
seizure	seizure	seizure_disorder	pain
seizure_disorder	seizures	restless_leg_syndrome	seizure
status_epilepticus	aspiration	ms	edema
mental_retardation	altered_mental_status	seizure	atelectasis
seizures	fever	dementia	fever
restless_leg_syndrome	unresponsive	hemothorax	pneumothorax
epilepsy	stress_ulcer	retropulsion	cough
multiple_sclerosis	infection	multiple_sclerosis	seizures
tonic_clonic_seizure	pneumonia	epilepsy	htn
cns_infection	hypotension	hydrocephalus	pneumonia
trigeminal_neuralgia	agitated	lethargic	stress_ulcer
parkinsons_disease	status_epilepticus	hypoxemia	hypotension
grand_mal_seizure	seizure_disorder	overdose	confused
generalized_seizure	mental_status	shortness_of_breath	agitated
facial_twitching	dementia	infarction	hemorrhage

Figure A.12: Learned Phenotypes for Other Neurological Disorders



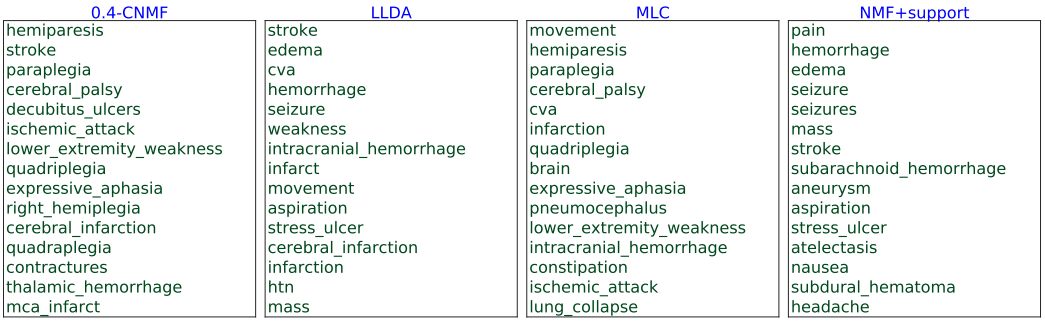
0.4-CNMF	LLDA	MLC	NMF+support
afib	afib	rapid_ventricular_response	pain
atrial_fibrillation	atrial_fibrillation	afib	afib
rvr	af	cardiomegaly	edema
af	pain	atrial_fibrillation	hemorrhage
chronic_atrial_fibrillation	stress_ulcer	acute_cholecystitis	atelectasis
crush_injury	htn	calcifications	atrial_fibrillation
babesiosis	stroke	acute_coronary_syndrome	stroke
non_reactive	edema	subdural_hematoma	pneumothorax
	bleeding	acute_on_chronic_renal_failure	htn
	hypotension	ischemic_heart_disease	cough
	cva	stroke	stress_ulcer
	gi_bleed	atrial_flutter	pleural_effusion
	altered_mental_status	tachycardia	intracranial_hemorrhage
	aspiration	hip_fracture	sob
	bleed	narrowing	nausea

Figure A.13: Learned Phenotypes for Cardiac Arrhythmias



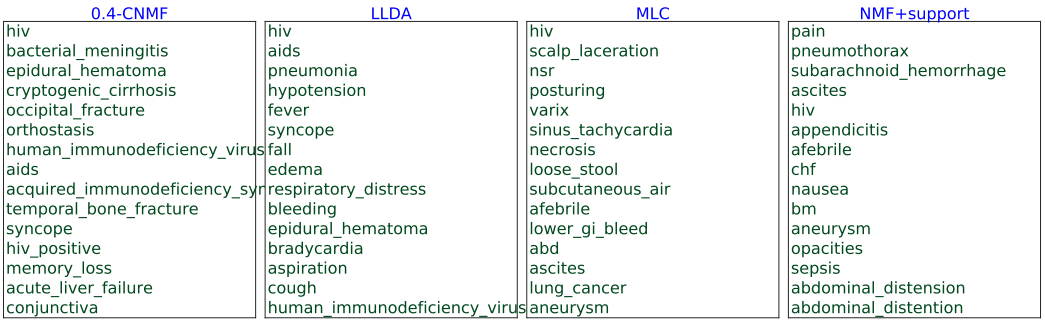
0.4-CNMF	LLDA	MLC	NMF+support
polysubstance_abuse	pain	substance_abuse	pain
substance_abuse	stress_ulcer	polysubstance_abuse	edema
cocaine_abuse	polysubstance_abuse	overdose	pneumothorax
overdose	agitated	chest_pressure	headache
addiction	asthma	cocaine_abuse	aneurysm
poisoning	chronic_pain	withdrawal	cough
rhabdomyolysis	pneumonia	skin_warm	subarachnoid_hemorrhage
assault	anxiety	fracture	hemorrhage
heroin_abuse	fever	epidural_abscess	dyspnea
hep_c	agitation	tamponade	sob
multiple_stab_wounds	substance_abuse	hep_c	hiv
bile_leak	respiratory_distress	chronic_renal_failure	fracture
bipolar_disorder	aspiration	hepatitis_c	afebrile
esophageal_injury	overdose	hiv	atelectasis
hep	infection	trauma	stress_ulcer

Figure A.14: Learned Phenotypes for Drug Abuse



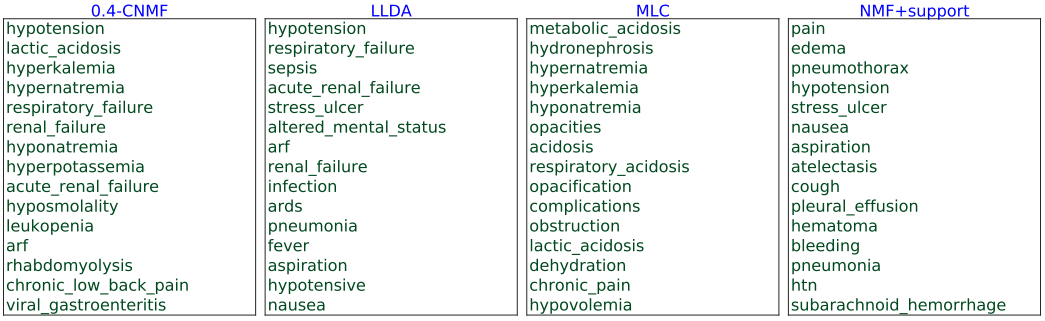
0.4-CNMF	LLDA	MLC	NMF+support
hemiparesis	stroke	movement	pain
stroke	edema	hemiparesis	hemorrhage
paraplegia	cva	paraplegia	edema
cerebral_palsy	hemorrhage	cerebral_palsy	seizure
decubitus_ulcers	seizure	cva	seizures
ischemic_attack	weakness	infarction	mass
lower_extremity_weakness	intracranial_hemorrhage	quadriplegia	stroke
quadriplegia	infarct	brain	subarachnoid_hemorrhage
expressive_aphasia	movement	expressive_aphasia	aneurysm
right_hemiplegia	aspiration	pneumocephalus	aspiration
cerebral_infarction	stress_ulcer	lower_extremity_weakness	stress_ulcer
quadraplegia	cerebral_infarction	intracranial_hemorrhage	atelectasis
contractures	infarction	constipation	nausea
thalamic_hemorrhage	htn	ischemic_attack	subdural_hematoma
mca_infarct	mass	lung_collapse	headache

Figure A.15: Learned Phenotypes for Paralysis



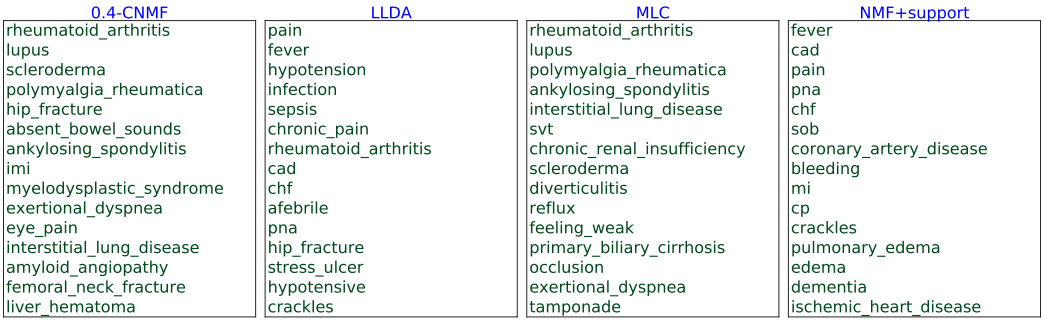
0.4-CNMF	LLDA	MLC	NMF+support
hiv	hiv	hiv	pain
bacterial_meningitis	aids	scalp_laceration	pneumothorax
epidural_hematoma	pneumonia	nsr	subarachnoid_hemorrhage
cryptogenic_cirrhosis	hypotension	posturing	ascites
occipital_fracture	fever	varix	hiv
orthostasis	syncope	sinus_tachycardia	appendicitis
human_immunodeficiency_virus	fall	necrosis	afebrile
aids	edema	loose_stool	chf
acquired_immunodeficiency_syr	respiratory_distress	subcutaneous_air	nausea
temporal_bone_fracture	bleeding	afebrile	bm
syncope	epidural_hematoma	lower_gi_bleed	aneurysm
hiv_positive	bradycardia	abd	opacities
memory_loss	aspiration	ascites	sepsis
acute_liver_failure	cough	lung_cancer	abdominal_distension
conjunctiva	human_immunodeficiency_virus	aneurysm	abdominal_distention

Figure A.16: Learned Phenotypes for AIDS



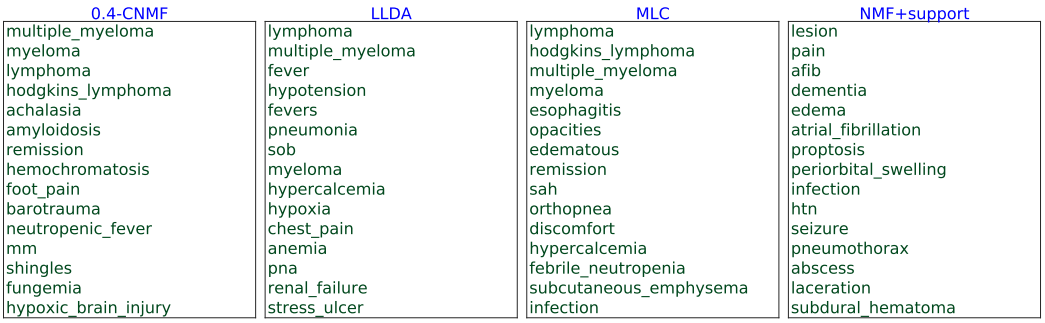
0.4-CNMF	LLDA	MLC	NMF+support
hypotension	hypotension	metabolic_acidosis	pain
lactic_acidosis	respiratory_failure	hydronephrosis	edema
hyperkalemia	sepsis	hypernatremia	pneumothorax
hypernatremia	acute_renal_failure	hyperkalemia	hypotension
respiratory_failure	stress_ulcer	hyponatremia	stress_ulcer
renal_failure	altered_mental_status	opacities	nausea
hyponatremia	arf	acidosis	aspiration
hyperpotassemia	renal_failure	respiratory_acidosis	atelectasis
acute_renal_failure	infection	opacification	cough
hyposmolality	ards	complications	pleural_effusion
leukopenia	pneumonia	obstruction	hematoma
arf	fever	lactic_acidosis	bleeding
rhabdomyolysis	aspiration	dehydration	pneumonia
chronic_low_back_pain	hypotensive	chronic_pain	htn
viral_gastroenteritis	nausea	hypovolemia	subarachnoid_hemorrhage

Figure A.17: Learned Phenotypes for Fluid Electrolyte Disorders



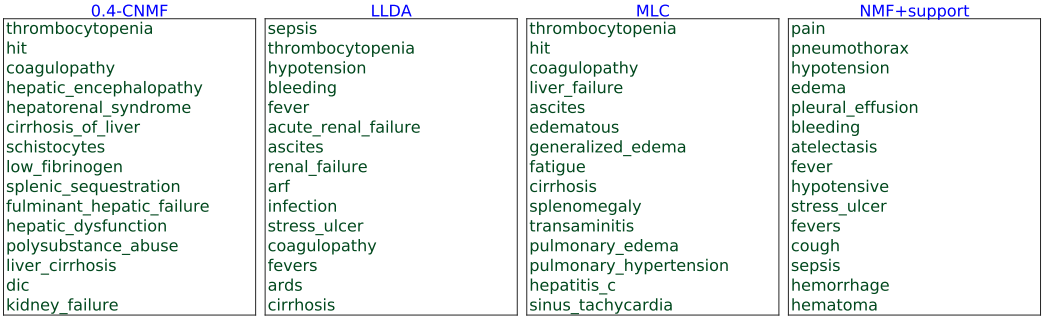
0.4-CNMF	LLDA	MLC	NMF+support
rheumatoid_arthritis	pain	rheumatoid_arthritis	fever
lupus	fever	lupus	cad
scleroderma	hypotension	polymyalgia_rheumatica	pain
polymyalgia_rheumatica	infection	ankylosing_spondylitis	pna
hip_fracture	sepsis	interstitial_lung_disease	chf
absent_bowel_sounds	chronic_pain	svt	sob
ankylosing_spondylitis	rheumatoid_arthritis	chronic_renal_insufficiency	coronary_artery_disease
imi	cad	scleroderma	bleeding
myelodysplastic_syndrome	chf	diverticulitis	mi
exertional_dyspnea	afebrile	reflux	cp
eye_pain	pna	feeling_weak	crackles
interstitial_lung_disease	hip_fracture	primary_biliary_cirrhosis	pulmonary_edema
amyloid_angiopathy	stress_ulcer	occlusion	edema
femoral_neck_fracture	hypotensive	exertional_dyspnea	dementia
liver_hematoma	crackles	tamponade	ischemic_heart_disease

Figure A.18: Learned Phenotypes for Rheumatoid Arthritis



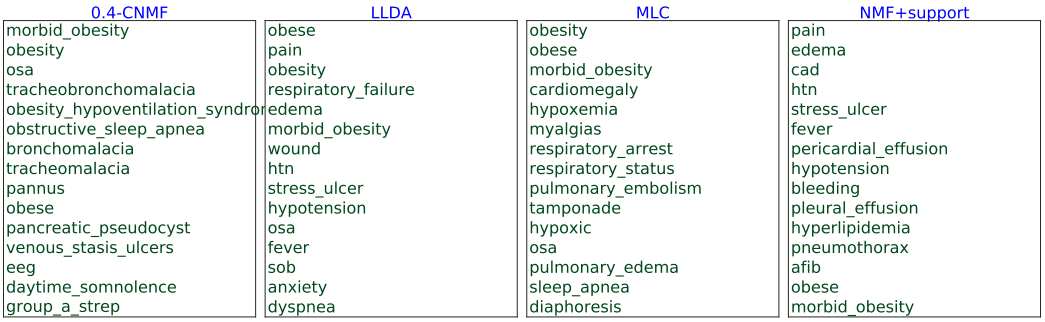
0.4-CNMF	LLDA	MLC	NMF+support
multiple_myeloma	lymphoma	lymphoma	lesion
myeloma	multiple_myeloma	hodgkins_lymphoma	pain
lymphoma	fever	multiple_myeloma	afib
hodgkins_lymphoma	hypotension	myeloma	dementia
achalasia	fevers	esophagitis	edema
amyloidosis	pneumonia	opacities	atrial_fibrillation
remission	sob	edematous	proptosis
hemochromatosis	myeloma	remission	periorbital_swelling
foot_pain	hypercalcemia	sah	infection
barotrauma	hypoxia	orthopnea	htn
neutropenic_fever	chest_pain	discomfort	seizure
mm	anemia	hypercalcemia	pneumothorax
shingles	pna	febrile_neutropenia	abscess
fungemia	renal_failure	subcutaneous_emphysema	laceration
hypoxic_brain_injury	stress_ulcer	infection	subdural_hematoma

Figure A.19: Learned Phenotypes for Lymphoma



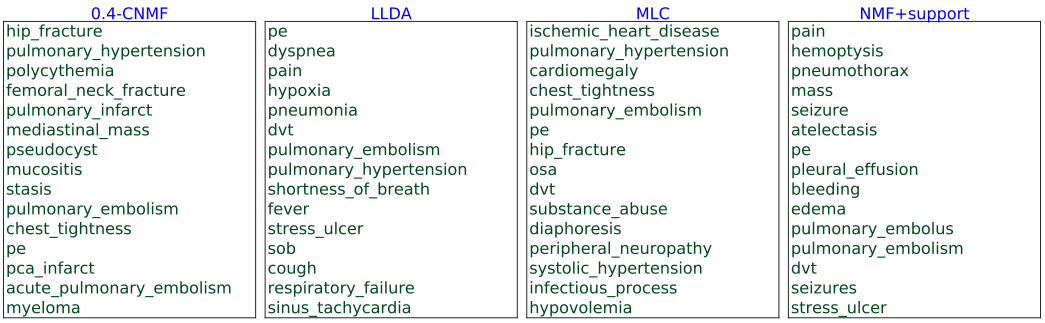
0.4-CNMF	LLDA	MLC	NMF+support
thrombocytopenia	sepsis	thrombocytopenia	pain
hit	thrombocytopenia	hit	pneumothorax
coagulopathy	hypotension	coagulopathy	hypotension
hepatic_encephalopathy	bleeding	liver_failure	edema
hepatorenal_syndrome	fever	ascites	pleural_effusion
cirrhosis_of_liver	acute_renal_failure	edematous	bleeding
schistocytes	ascites	generalized_edema	atelectasis
low_fibrinogen	renal_failure	fatigue	fever
splenic_sequestration	arf	cirrhosis	hypotensive
fulminant_hepatic_failure	infection	splenomegaly	stress_ulcer
hepatic_dysfunction	stress_ulcer	transaminitis	fevers
polysubstance_abuse	coagulopathy	pulmonary_edema	cough
liver_cirrhosis	fevers	pulmonary_hypertension	sepsis
dic	ards	hepatitis_c	hemorrhage
kidney_failure	cirrhosis	sinus_tachycardia	hematoma

Figure A.20: Learned Phenotypes for Coagulopathy



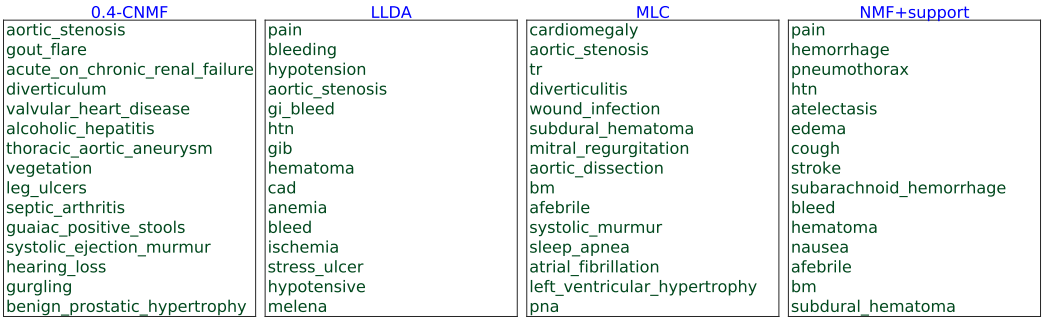
0.4-CNMF	LLDA	MLC	NMF+support
morbid_obesity	obese	obesity	pain
obesity	pain	obese	edema
osa	obesity	morbid_obesity	cad
tracheobronchomalacia	respiratory_failure	cardiomegaly	htn
obesity_hypoventilation_syndro	edema	hypoxemia	stress_ulcer
obstructive_sleep_apnea	morbid_obesity	myalgias	fever
bronchomalacia	wound	respiratory_arrest	pericardial_effusion
tracheomalacia	htn	respiratory_status	hypotension
pannus	stress_ulcer	pulmonary_embolism	bleeding
obese	hypotension	tamponade	pleural_effusion
pancreatic_pseudocyst	osa	hypoxic	hyperlipidemia
venous_stasis_ulcers	fever	osa	pneumothorax
eeg	sob	pulmonary_edema	afib
daytime_somnolence	anxiety	sleep_apnea	obese
group_a_strep	dyspnea	diaphoresis	morbid_obesity

Figure A.21: Learned Phenotypes for Obesity



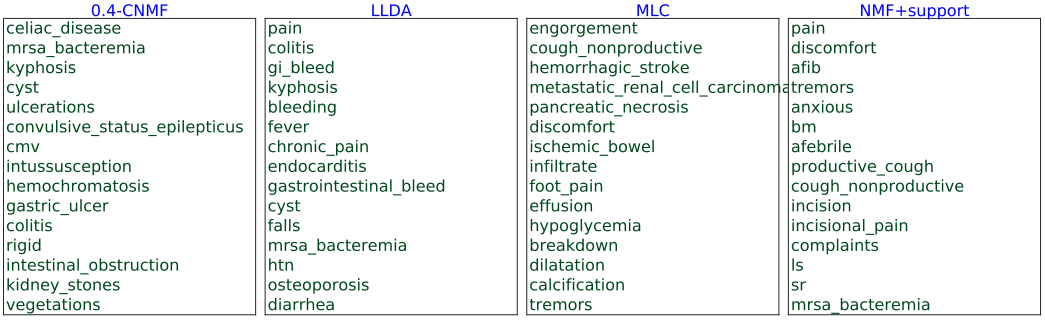
0.4-CNMF	LLDA	MLC	NMF+support
hip_fracture	pe	ischemic_heart_disease	pain
pulmonary_hypertension	dyspnea	pulmonary_hypertension	hemoptysis
polycythemia	pain	cardiomegaly	pneumothorax
femoral_neck_fracture	hypoxia	chest_tightness	mass
pulmonary_infarct	pneumonia	pulmonary_embolism	seizure
mediastinal_mass	dvt	pe	atelectasis
pseudocyst	pulmonary_embolism	hip_fracture	pe
mucositis	pulmonary_hypertension	osa	pleural_effusion
stasis	shortness_of_breath	dvt	bleeding
pulmonary_embolism	fever	substance_abuse	edema
chest_tightness	stress_ulcer	diaphoresis	pulmonary_embolus
pe	sob	peripheral_neuropathy	pulmonary_embolism
pca_infarct	cough	systolic_hypertension	dvt
acute_pulmonary_embolism	respiratory_failure	infectious_process	seizures
myeloma	sinus_tachycardia	hypovolemia	stress_ulcer

Figure A.22: Learned Phenotypes for Pulmonary Circulation Disorder



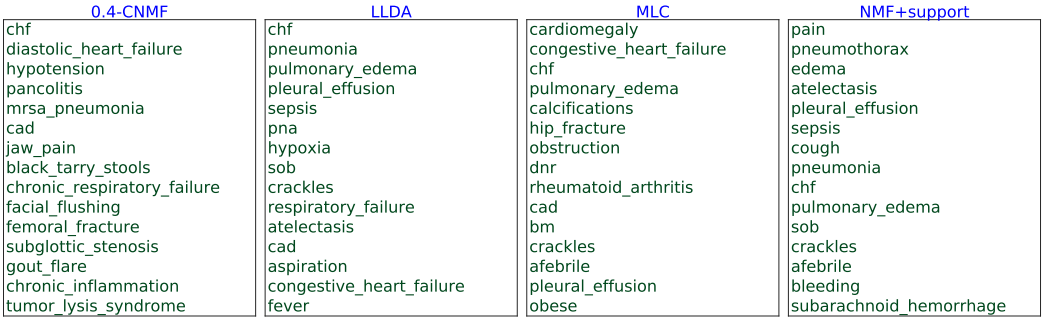
0.4-CNMF	LLDA	MLC	NMF+support
aortic_stenosis	pain	cardiomegaly	pain
gout_flare	bleeding	aortic_stenosis	hemorrhage
acute_on_chronic_renal_failure	hypotension	tr	pneumothorax
diverticulum	aortic_stenosis	diverticulitis	htn
valvular_heart_disease	gi_bleed	wound_infection	atelectasis
alcoholic_hepatitis	htn	subdural_hematoma	edema
thoracic_aortic_aneurysm	gib	mitral_regurgitation	cough
vegetation	hematoma	aortic_dissection	stroke
leg_ulcers	cad	bm	subarachnoid_hemorrhage
septic_arthritis	anemia	afebrile	bleed
guaiac_positive_stools	bleed	systolic_murmur	hematoma
systolic_ejection_murmur	ischemia	sleep_apnea	nausea
hearing_loss	stress_ulcer	atrial_fibrillation	afebrile
gurgling	hypotensive	left_ventricular_hypertrophy	bm
benign_prostatic_hypertrophy	melena	pna	subdural_hematoma

Figure A.23: Learned Phenotypes for Valvular Disease



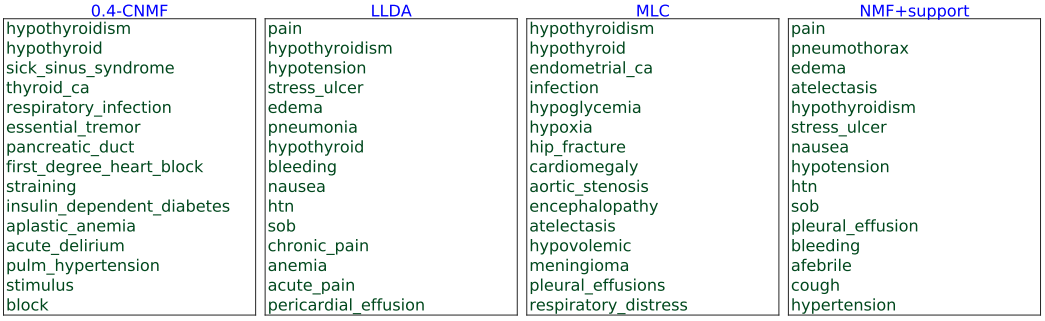
0.4-CNMF	LLDA	MLC	NMF+support
celiac_disease	pain	engorgement	pain
mrsa_bacteremia	colitis	cough_nonproductive	discomfort
kyphosis	gi_bleed	hemorrhagic_stroke	afib
cyst	kyphosis	metastatic_renal_cell_carcinoma	tremors
ulcerations	bleeding	pancreatic_necrosis	anxious
convulsive_status_epilepticus	fever	discomfort	bm
cmv	chronic_pain	ischemic_bowel	afebrile
intussusception	endocarditis	infiltrate	productive_cough
hemochromatosis	gastrointestinal_bleed	foot_pain	cough_nonproductive
gastric_ulcer	cyst	effusion	incision
colitis	falls	hypoglycemia	incisional_pain
rigid	mrsa_bacteremia	breakdown	complaints
intestinal_obstruction	htn	dilatation	ls
kidney_stones	osteoporosis	calcification	sr
vegetations	diarrhea	tremors	mrsa_bacteremia

Figure A.24: Learned Phenotypes for Peptic Ulcer



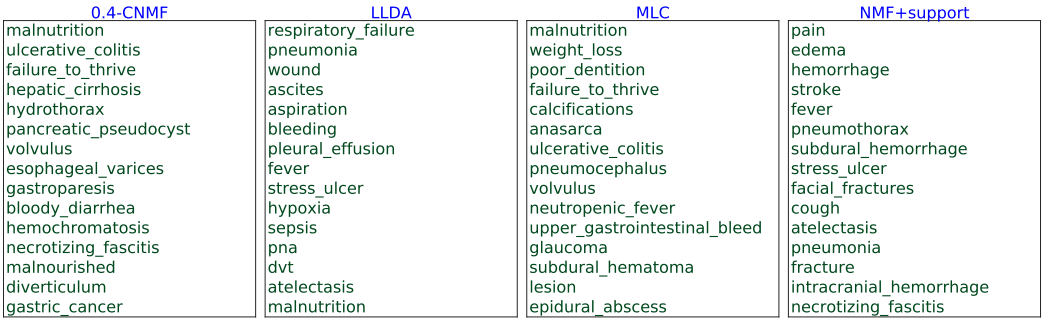
0.4-CNMF	LLDA	MLC	NMF+support
chf	chf	cardiomegaly	pain
diastolic_heart_failure	pneumonia	congestive_heart_failure	pneumothorax
hypotension	pulmonary_edema	chf	edema
pancolitis	pleural_effusion	pulmonary_edema	atelectasis
mrsa_pneumonia	sepsis	calcifications	pleural_effusion
cad	pna	hip_fracture	sepsis
jaw_pain	hypoxia	obstruction	cough
black_tarry_stools	sob	dnr	pneumonia
chronic_respiratory_failure	crackles	rheumatoid_arthritis	chf
facial_flushing	respiratory_failure	cad	pulmonary_edema
femoral_fracture	atelectasis	bm	sob
subglottic_stenosis	cad	crackles	crackles
gout_flare	aspiration	afebrile	afebrile
chronic_inflammation	congestive_heart_failure	pleural_effusion	bleeding
tumor_lysis_syndrome	fever	obese	subarachnoid_hemorrhage

Figure A.25: Learned Phenotypes for Congestive Heart Failure



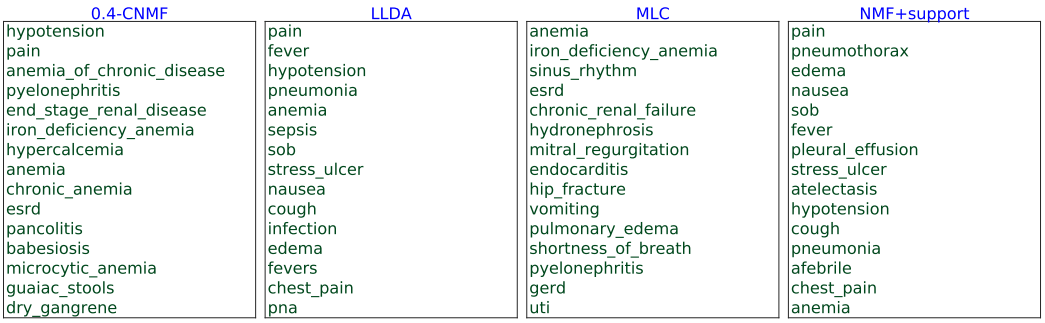
0.4-CNMF	LLDA	MLC	NMF+support
hypothyroidism	pain	hypothyroidism	pain
hypothyroid	hypothyroidism	hypothyroid	pneumothorax
sick_sinus_syndrome	hypotension	endometrial_ca	edema
thyroid_ca	stress_ulcer	infection	atelectasis
respiratory_infection	edema	hypoglycemia	hypothyroidism
essential_tremor	pneumonia	hypoxia	stress_ulcer
pancreatic_duct	hypothyroid	hip_fracture	nausea
first_degree_heart_block	bleeding	cardiomegaly	hypotension
straining	nausea	aortic_stenosis	htn
insulin_dependent_diabetes	htn	encephalopathy	sob
aplastic_anemia	sob	atelectasis	pleural_effusion
acute_delirium	chronic_pain	hypovolemic	bleeding
pulm_hypertension	anemia	meningioma	afebrile
stimulus	acute_pain	pleural_effusions	cough
block	pericardial_effusion	respiratory_distress	hypertension

Figure A.26: Learned Phenotypes for Hypothyroidism



0.4-CNMF	LLDA	MLC	NMF+support
malnutrition	respiratory_failure	malnutrition	pain
ulcerative_colitis	pneumonia	weight_loss	edema
failure_to_thrive	wound	poor_dentition	hemorrhage
hepatic_cirrhosis	ascites	failure_to_thrive	stroke
hydrothorax	aspiration	calcifications	fever
pancreatic_pseudocyst	bleeding	anasarca	pneumothorax
volvulus	pleural_effusion	ulcerative_colitis	subdural_hemorrhage
esophageal_varices	fever	pneumocephalus	stress_ulcer
gastroparesis	stress_ulcer	volvulus	facial_fractures
bloody_diarrhea	hypoxia	neutropenic_fever	cough
hemochromatosis	sepsis	upper_gastrointestinal_bleed	atelectasis
necrotizing_fascitis	pna	glaucoma	pneumonia
malnourished	dvt	subdural_hematoma	fracture
diverticulum	atelectasis	lesion	intracranial_hemorrhage
gastric_cancer	malnutrition	epidural_abscess	necrotizing_fascitis

Figure A.27: Learned Phenotypes for Weight loss



0.4-CNMF	LLDA	MLC	NMF+support
hypotension	pain	anemia	pain
pain	fever	iron_deficiency_anemia	pneumothorax
anemia_of_chronic_disease	hypotension	sinus_rhythm	edema
pyelonephritis	pneumonia	esrd	nausea
end_stage_renal_disease	anemia	chronic_renal_failure	sob
iron_deficiency_anemia	sepsis	hydronephrosis	fever
hypercalcemia	sob	mitral_regurgitation	pleural_effusion
anemia	stress_ulcer	endocarditis	stress_ulcer
chronic_anemia	nausea	hip_fracture	atelectasis
esrd	cough	vomiting	hypotension
pancolitis	infection	pulmonary_edema	cough
babesiosis	edema	shortness_of_breath	pneumonia
microcytic_anemia	fevers	pyelonephritis	afebrile
guaiac_stools	chest_pain	gerd	chest_pain
dry_gangrene	pna	uti	anemia

Figure A.28: Learned Phenotypes for Deficiency Anemias

0.4-CNMF	LLDA	MLC	NMF+support
cryptogenic_cirrhosis	pain	fulminant_hepatic_failure	pain
squamous_cell_carcinoma	bleeding	tired	bleeding
heel_ulcer	gi_bleed	hocm	chf
diverticular_disease	gib	hit	pneumothorax
lactate_levels	anemia	restless	atelectasis
anastomotic_leak	stress_ulcer	lower_gi_bleed	pleural_effusion
dark_stools	hives	effusions	edema
gangrenous_cholecystitis	hypotension	calcifications	hematoma
gastropathy	gastrointestinal_bleed	peripheral_neuropathy	pna
bowel_perforation	abdominal_pain	blood_loss	afebrile
portal_hypertensive_gastropathy	chest_pain	unresponsiveness	sob
syncopal_episodes	melena	sinus_tachycardia	pulmonary_edema
angioedema	wound	bacteremia	pneumonia
neutropenic_fever	chf	upper_gi_bleed	cough
irritable_bowel_syndrome	diarrhea	duodenal_perforation	confused

Figure A.29: Learned Phenotypes for Blood Loss Anemia

0.4-CNMF	LLDA	MLC	NMF+support
depression	pain	depression	pain
overdose	depression	systolic_dysfunction	hypotension
serotonin_syndrome	stress_ulcer	overdose	bleeding
od	anxiety	chronic_pain	sob
fibromyalgia	nausea	osteoarthritis	edema
clonus	chest_pain	ha	depression
blurred_vision	hypotension	blurred_vision	stress_ulcer
elevated_ammonia	aspiration	chest_pressure	nausea
type_1_diabetes	fever	cerebral_edema	bleed
crohns_disease	sob	back_pain	pneumothorax
fulminant_hepatic_failure	chronic_pain	lightheaded	atelectasis
liver_injury	bleeding	pulmonary_edema	aspiration
toxic_ingestion	abdominal_pain	obesity	hematoma
vp_shunt	vomiting	osa	pleural_effusion
bronchopleural_fistula	htn	hypothyroidism	anxiety

Figure A.30: Learned Phenotypes for Depression

A.3 Augmented mortality prediction

Figure A.31 shows weights learned by the classifier for all features. The weights shaded red correspond to phenotypes and are relatively high compared to raw notes based features (shaded blue), indicating that comorbidities capture significant amount of predictive information on mortality and achieve comparable performance to full EHR model when augmented with additional raw clinical terms.

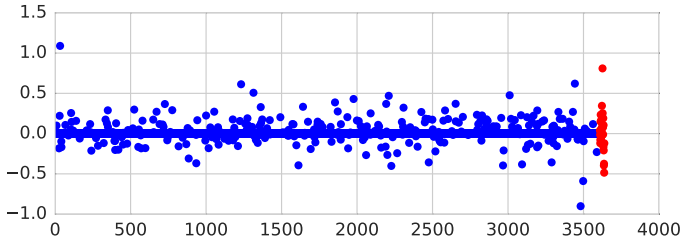


Figure A.31: Weights learned by the CNMF+Full EHR classifier for all features. The weights shaded red correspond to phenotypes.

Appendix B

Explainability Using Manifold Constrained Examples

Additional explainability results for Chapter 6 are provided in the following.

B.1 xGEMs for MNIST

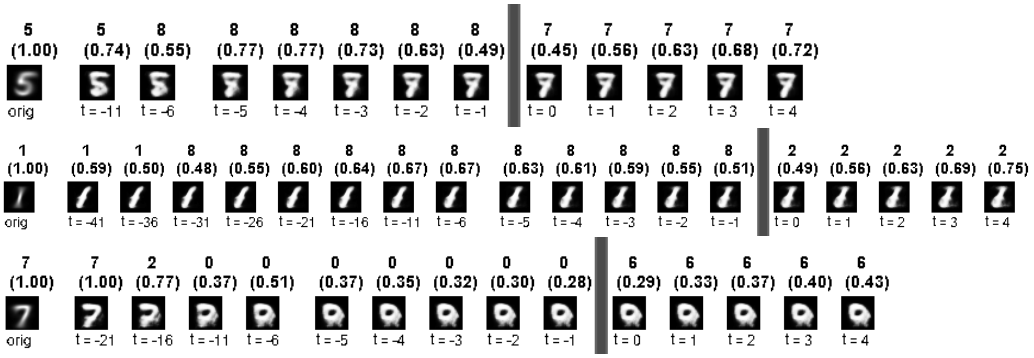


Figure B.1: xGEMs for MNIST data. $\mathcal{G}_\theta : \mathbb{R}^{100} \rightarrow \mathbb{R}^{28\times28}$ is a VAE while the target black-box is a softmax classifier. Each row shows a manifold constrained exampletransition for a single digit (labeled 'orig'). The gray vertical bars indicate transition to the target label $\mathbf{y}_{tar}$. Reconstructions in each row are intermediate reconstructions obtained using Algorithm 4. The confidence of the clas prediction is shown in parentheses for each reconstruction.

Figure B.1 shows manifold constrained examples generated for a (multi-

Figure B.2: Training progression for celebA face image for the CNN+lrn model.

class) softmax classifier for MNIST[1] digit data. The first row in Figure B.1 shows manifold constrained examplefor digit 5 if $\mathbf{y}_{tar} = 7$, while second and third row show manifold constrained examplesfor digits 1 and 7 with $\mathbf{y}_{tar} = 2$ and $\mathbf{y}_{tar} = 6$ respectively. Notice how while traversing the manifold, the classifier switches decision from 5 to 8 and then to the target label 7 (row 1). While the intermediate samples look like 7 to human eye, the classifier is biased toward predicting 8. Row 2 suggests a bias toward predicting 1 as 8 for a minor smudging (visible to human eye). Finally, the third row demonstrates how the manifold constrained examplefor 7 suggests that the classifier considers a 0 to be labeled as 6. Thus manifold constrained examples can provide insight into the decision boundary of the classifier for each pair of digits.

[1]http://yann.lecun.com/exdb/mnist/

B.2 Case Study: Evaluating Model Training Progression

Figure B.3: Training progression for celebA face image for the ResNet model.

Figures B.2 and B.3 show **xGEMs** for the face corresponding to Sample 1 in Figure 6.4 for models CNN+lrn and ResNet respectively. Notice significant differences in the **xGEMs** and their trajectories even at comparable overall performance.

Appendix C

Constraints based Clustering

This chapter contains supplementary material for the mode proposed in Section 7.3.

C.1 Derivation of Variational Inference for Weighted Sum of Divergence Minimization

We wish to minimize the weighted sum of divergence between M distributions $p^m(\mathbf{z})$. Let $q^*(\mathbf{z})$ be the corresponding minimizing distribution. Let $w \in \Delta^M$ be the weight vector determining how important a given distribution is. The specific cost function is provided in (C.1). We only consider the case when each of the distributions are categorical distributions over clusters $[K]$ and $\mathbf{z} \in \{0,1\}^K$ such that only one of the elements is 1.

$$q^*(\mathbf{z}) = \arg\min_{q(\mathbf{z})} \sum_{i \in [M]} w_i \mathcal{D}_\gamma(p^m(\mathbf{z}) \| q(\mathbf{z})) \tag{C.1}$$

Let $\kappa^i(\mathbf{z})$ be a variational distribution corresponding to $p^i(\mathbf{z})$. Using the log-sum inequality, we have a lower on (C.1) given by (C.2).

$$\begin{aligned} &\sum_{i \in [M]} \frac{w_i}{\gamma} \mathcal{D}_\gamma(p^i(\mathbf{z}) || q(\mathbf{z})) \geq \\ &\sum_{m \in [M]} \frac{w_i}{\gamma(\gamma-1)} \sum_{k \in K} \kappa^i(z_k = 1)(\log\left[p^i(\mathbf{z})^\gamma q(\mathbf{z})^{(1-\gamma)}\right] - \log \kappa^i(z_k = 1)) \end{aligned} \tag{C.2}$$

Algorithm 10 Variational Update to solve (C.2)

Given $\mathbf{w}$, γ and initial estimates of $q(\mathbf{z})$,
repeat
  $\kappa^i(\mathbf{z}) \propto p^i(\mathbf{z})^\gamma q(\mathbf{z})^{(1-\gamma)} \, \forall \, i \in [M]$
  $q(\mathbf{z}) \propto \sum_{i \in [M]} w_i \kappa^i(\mathbf{z})$
until Converged

We optimize the lower bound by iteratively optimizing $\kappa^i(\mathbf{z})$'s and $q(\mathbf{z})$. To update $\kappa^i(\mathbf{z})$, $\kappa^j(\mathbf{z}) \, \forall \, j \in [M], \; j \neq i$ and $q(\mathbf{z})$ are held fixed. Then update for $\kappa^i(\mathbf{z}) \propto p^i(\mathbf{z})^\gamma q(\mathbf{z})^{(1-\gamma)}$. When all $\kappa^i(\mathbf{z})$ are held fixed, $q(\mathbf{z})$ is again obtained by setting the gradient of the bound w.r.t. $q(\mathbf{z})$ to 0. The iterative update is described by algorithm (10).

$$q(\mathbf{z}) \propto \sum_{i \in [M]} w_i \kappa^i(\mathbf{z}_n) \tag{C.3}$$

C.2 Proof that aggregation in E-step can be solved in parallel over samples

Let $\mathbf{z}^i = \{\mathbf{z}^i_n : n \in [N]\}$ and $\mathbf{x}^i = \{\mathbf{x}^i_n : n \in [N]\}$. Let $\mathbf{z} = \{\mathbf{z}^i : i \in [V]\}$, $\mathbf{x} = \{\mathbf{x}^i : i \in [V]\}$ and $\mathbf{\Psi} = \{\mathbf{\Psi}^i : i \in [V]\}$. Let $g(\mathbf{z})$ be the target posterior for GRECO is obtained by solving (C.4).

$$\begin{aligned} g(\mathbf{z}) &= \arg\min_{q(\mathbf{z})} \sum_{i \in [V]} w_i \mathcal{D}_\gamma(p(\mathbf{z}^i|\mathbf{x}^i, \mathbf{\Psi}^i) \| g(\mathbf{z})) \\ &= \arg\min_{q(\mathbf{z})} \sum_{i \in [V]} \frac{w_i}{\gamma(\gamma-1)} \log E_{p(\mathbf{z}|\mathbf{x}, \mathbf{\Psi})} \left[\left(\frac{g(\mathbf{z})}{p(\mathbf{z}^i|\mathbf{x}^i, \mathbf{\Psi})} \right)^{(1-\gamma)} \right] \end{aligned} \tag{C.4}$$

We wish to estimate the complete posterior $g(\mathbf{z})$ such that it is independent across all samples, i.e. $g(\mathbf{z}) = \prod_{n \in [N]} g(\mathbf{z}_n)$. By the IID assumption on the log-likelihood, the posterior $p(\mathbf{z}|\mathbf{x}, \mathbf{\Psi})$ can be factored into per-view, per-sample

posteriors as in (C.5)

$$p(\mathbf{z}|\mathbf{x}, \mathbf{\Psi}) = \prod_{n\in[N]} \prod_{i\in[V]} p(\mathbf{z}_n^i|\mathbf{x}_n^i, \mathbf{\Psi}^i) \tag{C.5}$$

Equation (C.4) can be simplified as in (C.6)

$$\begin{aligned}
g(\mathbf{z}) &= \arg\min_{g(\mathbf{z})} \sum_{i\in[V]} w_i \log E_{\prod_{n\in[N]}\prod_{i\in[V]} p(\mathbf{z}_n^i|\mathbf{x}_n^i,\mathbf{\Psi}^i)} \left[\prod_{n\in[N]} \left(\frac{g(\mathbf{z}_n)}{p(\mathbf{z}_n^i|\mathbf{x}_n^i, \mathbf{\Psi}^i)} \right)^{(1-\gamma)} \right] \\
&= \arg\min_{g(\mathbf{z})} \sum_{i\in[V]} w_i \log \prod_{n\in[N]} E_{p(\mathbf{z}_n^i|\mathbf{x}_n^i,\mathbf{\Psi}^i)} \left[\left(\frac{g(\mathbf{z}_n)}{p(\mathbf{z}_n^i|\mathbf{x}_n^i, \mathbf{\Psi}^i)} \right)^{1-\gamma} \right] \\
&= \arg\min_{g(\mathbf{z})} \sum_{i\in[V]} w_i \sum_{n\in[N]} \log E_{p(\mathbf{z}_n^i|\mathbf{x}_n^i,\mathbf{\Psi}^i)} \left[\left(\frac{g(\mathbf{z}_n)}{p(\mathbf{z}_n^i|\mathbf{x}_n^i, \mathbf{\Psi}^i)} \right)^{1-\gamma} \right] \\
&= \arg\min\nolimits_{\prod_{n\in[N]} g(\mathbf{z}_n)} \sum_{n\in[N]} \sum_{i\in[V]} w_i D_\gamma(p(\mathbf{z}_n^i|\mathbf{x}_n^i, \mathbf{\Psi}^i) \| g(\mathbf{z}_n)) \\
\therefore g(\mathbf{z}_n) &= \arg\min\nolimits_{g(\mathbf{z}_n)} \sum_{i\in[V]} w_i D_\gamma(p(\mathbf{z}_n^i|\mathbf{x}_n^i, \mathbf{\Psi}^i) \| g(\mathbf{z}_n))
\end{aligned} \tag{C.6}$$

Equation (C.6) can be solved in parallel for each sample n to obtain $g(\mathbf{z}) = \prod_{n\in[N]} g(\mathbf{z}_n)$. This completes the proof and can be analogously proved for LYRIC and view-specific updates.

C.3 M-step for Standard Mixture Models

Let N be the total number of samples in a mixture model with K classes. Let at any iteration t, $q(\mathbf{z}_n)$ be the posterior responsibilities calculated using current model parameters of the mixture model. Let $x_n \in \mathcal{R}^D$ represent the observed features e.g. numeric data modeled as a gaussian mixture or count data that can be modeled as a mixture of multinomials.

- **Gaussian Mixture Models:** If the mixture model is a gaussian mixture with parameters $\boldsymbol{\mu}_k$ and $\boldsymbol{\Sigma}_k \forall\, k \in [K]$, the mean $\boldsymbol{\mu}_k$ and Covariance $\boldsymbol{\Sigma}_k$ are updated using (C.7) and (C.8) respectively.

$$\boldsymbol{\mu}_{t+1,k} = \frac{\sum_{n\in[N]} q(\mathbf{z}_{n,k}) x_n}{\sum_{n\in[N]} q(\mathbf{z}_{n,k})} \tag{C.7}$$

$$\boldsymbol{\Sigma}_{t+1,k} = \frac{\sum_{n\in[N]} q(\mathbf{z}_{n,k})(x_n - \boldsymbol{\mu}_{t+1,k})(x_n - \boldsymbol{\mu}_{t+1,k})^T}{\sum_{n\in[N]} q(\mathbf{z}_{n,k})} \tag{C.8}$$

- **Multinomial Mixture Models:** Let Multinomial distribution parameters for each cluster $\theta_k \,\forall\, k \in [K]$ can be updated using (C.9)

$$\boldsymbol{\theta}_{t+1,k} = \frac{\sum_{n\in[N]} q(\mathbf{z}_{n,k}) x_n}{\sum_{n\in[N]} q(\mathbf{z}_{n,k}) \sum_{d\in[D]} x_{n,d}} \tag{C.9}$$

C.4 Formulae of Evaluation Metrics:

All evaluation metrics assume that ground-truth cluster memberships are known. We assume that correspondence between clustering labels and ground-truth labels is already estimated using Hungarian matching Kuhn (1955) and the number of learned clusters is the same as number of ground-truth clusters, specifically for the metrics clustering accuracy, precision, recall and F-measure.

Definition C.4.1. If C_n represents the cluster label determined by the learning algorithm and ω_n represents the ground-truth clustering, the **clustering accuracy** for a dataset with N samples and K clusters is given by,

$$\textbf{Accuracy} = \frac{\sum_{n\in[K]} \sum_{k\in[K]} \mathbf{1}(C_n == \omega_n)}{N} \tag{C.10}$$

where,

$$\mathbf{1}(C_n == \omega_n) = \begin{cases} 1, & \text{if } C_n = \omega_n, \\ 0, & \text{otherwise.} \end{cases}$$

Following terms are defined per cluster $k \in [K]$

- *True Positives* (TP_k): This is the number of samples that were clustered correctly by the learning model.
- *False Positives* (FP_k): It is the number of samples assigned to a cluster they do not belong to.
- *True Negatives* (TN_k): This is defined per cluster label i.e. total number of samples not belonging to a given cluster and is clustered correctly i.e. clustered into a different cluster than for which true negatives are measured.
- *False Negatives* (FN_k): This is also defined per cluster label i.e. total number of samples belonging to a given cluster that were not actually assigned to the cluster by the learning algorithm.

Definition C.4.2.

$$\textbf{Precision} = \frac{\sum_{k\in[K]} TP_k}{\sum_{k\in[K]} TP_k + FP_k} \tag{C.11}$$

Definition C.4.3.

$$\textbf{Recall} = \frac{\sum_{k\in[K]} TP_k}{\sum_{k\in[K]} TP_k + FN_k} \tag{C.12}$$

Definition C.4.4.

$$\textbf{F-measure} = \frac{2 \times Precision \times Recall}{Precision + Recall} \tag{C.13}$$

The following metrics do not assume a correspondence between ground-truth labels and learned cluster labels. These metrics are based on measures of information, namely Mutual Information and Entropy.

Definition C.4.5. Let C be the categorical random variable over K clusters with a distribution obtained from clustering i.e. $Pr(C = k)$ is the fraction of samples clustered into k by the learning algorithm. Let ω represent the categorical variable with a distribution obtained from true clustering. The joint distribution $p(C, \omega)$ is the fraction of samples clustered as C and lie in ground-truth cluster ω. The mutual information $I(C, \omega)$ is given by,

$$I(C, \omega) = \sum_{k \in [K]} \sum_{j \in [K]} p(C = k, \omega = j) \log \frac{p(C = k, \omega = j)}{p(C = k)p(\omega = j)} \tag{C.14}$$

The Entropy of $H(C) = \sum_{k \in [K]} p(C = k) \log p(C = k)$ and analogously for $H(\omega)$. **Normalized Mutual Information (NMI) Strehl and Ghosh (2003)** is the symmetrized and normalized mutual information between C and ω.

$$NMI(C, \omega) = \frac{I(C, \omega)}{\frac{H(C)+H(\omega)}{2}} \tag{C.15}$$

Definition C.4.6.

$$\textbf{Average Entropy} = -\sum_{j \in [K]} p(C = j) \sum_{k \in [K]} p(C = j, \omega = k) \log p(C = j, \omega = k) \tag{C.16}$$

Bibliography

Brain tumors, 2014. URL http://www.nlm.nih.gov/medlineplus/braintumors.html.

Radiology info for patients, 2015. URL http://www.radiologyinfo.org/en/info.cfm?pg=kidneyfailure.

Agnar Aamodt and Enric Plaza. Case-based reasoning: Foundational issues, methodological variations, and system approaches. *AI communications*, 7 (1):39–59, 1994.

Sreangsu Acharyya and Joydeep Ghosh. Memr: A margin equipped monotone retargeting framework for ranking. In *UAI*, pages 2–11, 2014.

Sreangsu Acharyya, Oluwasanmi Koyejo, and Joydeep Ghosh. Learning to rank with bregman divergences and monotone retargeting. *arXiv preprint arXiv:1210.4851*, 2012.

Julius Adebayo, Justin Gilmer, Ian Goodfellow, and Been Kim. Local explanation methods for deep neural networks lack sensitivity to parameter values. *arXiv preprint*, 2018.

Massih Reza Amini, Tuong Vinh Truong, and Cyril Goutte. A boosting algorithm for learning bipartite ranking functions with partially labeled data. In *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*, pages 99–106. ACM, 2008.

J Angwin, J Larson, S Mattu, and L Kirchner. Machine bias risk assessments in criminal sentencing. *ProPublica https://www. propublica. org*, 2016.

Jimmy Ba, Volodymyr Mnih, and Koray Kavukcuoglu. Multiple object recognition with visual attention. *arXiv preprint arXiv:1412.7755*, 2014.

Sebastian Bach, Alexander Binder, Grégoire Montavon, Frederick Klauschen, Klaus-Robert Müller, and Wojciech Samek. On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. *PloS one*, 10(7):e0130140, 2015.

A. Banerjee, S. Merugu, I. S. Dhillon, and J. Ghosh. Clustering with bregman divergences. *Journal of Machine Learning Research*, 2005a.

Arindam Banerjee, Chase Krumpelman, Joydeep Ghosh, Sugato Basu, and Raymond J. Mooney. Model-based Overlapping Clustering. In *Proceedings of the Eleventh ACM SIGKDD International Conference on Knowledge Discovery in Data Mining*, KDD '05, pages 532–537, New York, NY, USA, 2005b. ACM. ISBN 1-59593-135-X. doi: 10.1145/1081870.1081932.

Michael J Best and Nilotpal Chakravarti. Active set algorithms for isotonic regression; a unifying framework. *Mathematical Programming*, 47(1-3):425–439, 1990.

Steffen Bickel and Tobias Scheffer. Estimation of Mixture Models Using Co-EM. In *Machine Learning: ECML 2005, 16th European Conference on Machine Learning, Porto, Portugal*, 2005.

D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. *Journal of Machine Learning Research*, 2003.

Avrim Blum and Tom Mitchell. Combining Labeled and Unlabeled Data with Co-training. In *Proceedings of the Eleventh Annual Conference on Computational Learning Theory*. ACM, 1998a.

Avrim Blum and Tom Mitchell. Combining labeled and unlabeled data with co-training. In *Proceedings of the eleventh annual conference on Computational learning theory*, pages 92–100. ACM, 1998b.

Tolga Bolukbasi, Kai-Wei Chang, James Y Zou, Venkatesh Saligrama, and Adam T Kalai. Man is to computer programmer as woman is to homemaker? debiasing word embeddings. In *Advances in Neural Information Processing Systems*, pages 4349–4357, 2016.

Alison Callahan and Nigam H Shah. Chapter 19 - machine learning in healthcare. In Aziz Sheikh, Kathrin M. Cresswell, Adam Wright, and David W. Bates, editors, *Key Advances in Clinical Informatics*, pages 279 – 291. Academic Press, 2017.

Kamalika Chaudhuri, Sham M. Kakade, Karen Livescu, and Karthik Sridharan. Multi-view Clustering via Canonical Correlation Analysis. In *Proceedings of the 26th Annual International Conference on Machine Learning*, 2009.

Zhengping Che, Sanjay Purushotham, Robinder Khemani, and Yan Liu. Interpretable deep models for icu outcome prediction. In *AMIA Annual Symposium Proceedings*, volume 2016, page 371. American Medical Informatics Association, 2016.

Edward Choi, Mohammad Taha Bahadori, Jimeng Sun, Joshua Kulas, Andy Schuetz, and Walter Stewart. Retain: An interpretable predictive model for

healthcare using reverse time attention mechanism. In *Advances in Neural Information Processing Systems*, pages 3504–3512, 2016.

Tat-Seng Chua, Jinhui Tang, Richang Hong, Haojie Li, Zhiping Luo, and Yan-Tao. Zheng. NUS-WIDE: A Real-World Web Image Database from National University of Singapore. In *Proc. of ACM Conf. on Image and Video Retrieval*, 2009.

R Dennis Cook and Sanford Weisberg. Characterizations of an empirical influence function for detecting influential cases in regression. *Technometrics*, 1980.

Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Machine learning*, 20, 1995.

Morris H DeGroot and Stephen E Fienberg. The comparison and evaluation of forecasters. *The statistician*, pages 12–22, 1983.

A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the em algorithm. *JOURNAL OF THE ROYAL STATISTICAL SOCIETY, SERIES B*, 39, 1977.

Carl Doersch. Tutorial on variational autoencoders. *arXiv preprint arXiv:1606.05908*, 2016.

Jeff Donahue, Philipp Krähenbühl, and Trevor Darrell. Adversarial feature learning. *arXiv preprint arXiv:1605.09782*, 2016.

Been Doshi-Velez, Finale; Kim. Towards a rigorous science of interpretable machine learning. In *eprint arXiv:1702.08608*, 2017.

Vincent Dumoulin, Ishmael Belghazi, Ben Poole, Olivier Mastropietro, Alex Lamb, Martin Arjovsky, and Aaron Courville. Adversarially learned inference. *arXiv preprint arXiv:1606.00704*, 2016.

Ethan Elenberg, Alexandros G Dimakis, Moran Feldman, and Amin Karbasi. Streaming weak submodularity: Interpreting neural networks on the fly. In *Advances in Neural Information Processing Systems 30*. Curran Associates, Inc., 2017.

André Elisseeff, Jason Weston, et al. A kernel method for multi-labelled classification. In *NIPS*, volume 14, pages 681–687, 2001.

A. Elixhauser, C. Steiner, D. R. Harris, and R. M. Coffey. Comorbidity measures for use with administrative data. *Medical Care*, 1998.

Yoav Freund, Raj Iyer, Robert E Schapire, and Yoram Singer. An efficient boosting algorithm for combining preferences. *Journal of machine learning research*, 4(Nov):933–969, 2003.

Wei Gao and Pei Yang. Democracy is good for ranking: Towards multi-view rank learning and adaptation in web search. In *Proceedings of the 7th ACM international conference on Web search and data mining*, pages 63–72. ACM, 2014.

Ashutosh Garg, TS Jayram, Shivakumar Vaithyanathan, and Huaiyu Zhu. Generalized opinion pooling. In *AMAI*, 2004.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in Neural Information Processing Systems*, pages 2672–2680, 2014a.

Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*, 2014b.

Derek Greene and Pádraig Cunningham. Producing a Unified Graph Representation from Multiple Social Network Views. In *Proceedings of the 5th Annual ACM Web Science Conference*, 2013.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q Weinberger. On calibration of modern neural networks. *arXiv preprint arXiv:1706.04599*, 2017.

Maya Gupta, Andrew Cotter, Jan Pfeifer, Konstantin Voevodski, Kevin Canini, Alexander Mangylov, Wojciech Moczydlowski, and Alexander Van Esbroeck. Monotonic calibrated interpolated look-up tables. *Journal of Machine Learning Research*, 2016.

Moritz Hardt, Eric Price, Nati Srebro, et al. Equality of opportunity in supervised learning. In *Advances in neural information processing systems*, pages 3315–3323, 2016.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

Irina Higgins, Loic Matthey, Xavier Glorot, Arka Pal, Benigno Uria, Charles Blundell, Shakir Mohamed, and Alexander Lerchner. Early visual concept learning with unsupervised deep learning. *arXiv preprint arXiv:1606.05579*, 2016.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

Geoffrey E Hinton. Training products of experts by minimizing contrastive divergence. *Neural computation*, 14(8):1771–1800, 2002.

G. Hripcsak and D. J. Albers. Next-generation phenotyping of electronic health records. *Journal of the American Medical Informatics Association*, 2013.

David R Hunter and Kenneth Lange. Quantile regression via an mm algorithm. *Journal of Computational and Graphical Statistics*, 9(1):60–77, 2000.

David Inouye, Pradeep D. Ravikumar, and Inderjit S. Dhillon. Admixture of poisson mrfs: A topic model with word dependencies. In *Proceedings of the 31th International Conference on Machine Learning, ICML 2014, Beijing, China, 21-26 June 2014*, volume 32 of *JMLR Proceedings*. JMLR.org, 2014a.

David Inouye, Pradeep K Ravikumar, and Inderjit S Dhillon. Capturing semantically meaningful word dependencies with an admixture of poisson mrfs. In Z. Ghahramani, M. Welling, C. Cortes, N.D. Lawrence, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems 27*. Curran Associates, Inc., 2014b.

Kalervo Järvelin and Jaana Kekäläinen. Ir evaluation methods for retrieving highly relevant documents. In *Proceedings of the 23rd annual international ACM SIGIR conference on Research and development in information retrieval*, pages 41–48. ACM, 2000.

Thorsten Joachims. Transductive inference for text classification using support vector machines. In *ICML*, volume 99, pages 200–209, 1999.

Shalmali Joshi, Oluwasanmi Koyejo, Kristine Resurreccion, and Joydeep Ghosh. Simultaneous prognosis and exploratory analysis of multiple chronic

conditions using clinical notes. In *2015 International Conference on Healthcare Informatics*, pages 243–252. ieeexplore.ieee.org, October 2015. URL http://ieeexplore.ieee.org/abstract/document/7349697/.

Shalmali Joshi, Joydeep Ghosh, Mark Reid, and Oluwasanmi Koyejo. Rényi divergence minimization based co-regularized multiview clustering. *Mach. Learn.*, 104(2-3):411–439, 1 September 2016a. URL https://link.springer.com/article/10.1007/s10994-016-5543-2.

Shalmali Joshi, Suriya Gunasekar, David Sontag, and Joydeep Ghosh. Identifiable phenotyping using constrained Non-Negative matrix factorization. In *Machine Learning for Healthcare Conference*, pages 17–41. jmlr.org, 10 December 2016b. URL https://arxiv.org/abs/1608.00704.

Shalmali Joshi, Rajiv Khanna, and Joydeep Ghosh. Co-regularized monotone regtargeting for semi-supervised LeTOR. In *Siam International Conference on Data Mining (SDM)*, 2018a.

Shalmali Joshi, Oluwasanmi Koyejo, Been Kim, and Joydeep Ghosh. **xGEMS:** Generating Exemplars to Explain Black-Box Models. 2018b.

Andrej Karpathy, Justin Johnson, and Li Fei-Fei. Visualizing and understanding recurrent networks. *arXiv preprint arXiv:1506.02078*, 2015.

Maurice George Kendall. Rank correlation methods. 1948.

MG Kendall. A new measure of rank correlation. *Biometrika*, page 93, 1938.

Been Kim, Rajiv Khanna, and Oluwasanmi O Koyejo. Examples are not enough, learn to criticize! criticism for interpretability. In *Advances in Neural Information Processing Systems*, pages 2280–2288, 2016.

P.-J. Kindermans, S. Hooker, J. Adebayo, M. Alber, K. T. Schütt, S. Dähne, D. Erhan, and B. Kim. The (un)reliability of saliency methods. *NIPS workshop on Explaining and Visualizing Deep Learning*, 2017.

Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

Pang Wei Koh and Percy Liang. Understanding black-box predictions via influence functions. *arXiv preprint arXiv:1703.04730*, 2017.

Daphne Koller and Nir Friedman. *Probabilistic graphical models: principles and techniques*. MIT press, 2009.

Yehuda Koren, Robert Bell, and Chris Volinsky. Matrix factorization techniques for recommender systems. *Computer*, 42(8), 2009.

A Kowalchuk and BC Reed. *Textbook of Family Medicine*. Philadelphia, Pa: Elsevier Saunders, 2011.

H.W. Kuhn. The Hungarian method for the assignment problem. *Naval research logistics quarterly*, 1955.

Abhishek Kumar, Piyush Rai, and Hal Daume. Co-regularized Multi-view Spectral Clustering. In *Advances in Neural Information Processing Systems 24*. Curran Associates, Inc., 2011.

D. Lee and H. S. Seung. Learning the parts of objects by non-negative matrix factorization. *Nature*, 1999.

Daniel D Lee and H Sebastian Seung. Algorithms for non-negative matrix factorization. In *Advances in neural information processing systems*, pages 556–562, 2001.

David D. Lewis, Yiming Yang, Tony G. Rose, and Fan Li. Rcv1: A new benchmark collection for text categorization research. *J. Mach. Learn. Res.*, 5, 2004.

Jiwei Li, Xinlei Chen, Eduard Hovy, and Dan Jurafsky. Visualizing and understanding neural models in nlp. *arXiv preprint arXiv:1506.01066*, 2015.

Ming Li, Hang Li, and Zhi-Hua Zhou. Semi-supervised document retrieval. *Information Processing & Management*, 45(3):341–355, 2009.

Ping Li, Christopher JC Burges, Qiang Wu, JC Platt, D Koller, Y Singer, and S Roweis. Mcrank: Learning to rank using multiple classification and gradient boosting. In *NIPS*, volume 7, pages 845–852, 2007.

C. J. Lin. Projected gradient methods for nonnegative matrix factorization. *Neural computation*, 2007.

Zachary C Lipton. The mythos of model interpretability. *arXiv preprint arXiv:1606.03490*, 2016.

Jialu Liu, Chi Wang, Jing Gao, and Jiawei Han. Multi-View Clustering via Joint Nonnegative Matrix Factorization. In *Proc. of 2013 SIAM Data Mining Conf.*, 2013.

Tie-Yan Liu. Learning to rank for information retrieval. *Found. Trends Inf. Retr.*, 2009.

Tie-Yan Liu, Jun Xu, Tao Qin, Wenying Xiong, and Hang Li. Letor: Benchmark dataset for research on learning to rank for information retrieval. In *Proceedings of SIGIR 2007 workshop on learning to rank for information retrieval*, volume 310, 2007.

Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *Proceedings of International Conference on Computer Vision (ICCV)*, December 2015.

Scott Lundberg and Su-In Lee. An unexpected unity among methods for interpreting model predictions. *arXiv preprint arXiv:1611.07478*, 2016.

Scott Lundberg and Su-In Lee. A unified approach to interpreting model predictions. *Computing Research Repository*, abs/1705.07874, 2017.

John I Marden. *Analyzing and modeling rank data*. CRC Press, 1996.

A. K. McCallum. Mallet: A machine learning for language toolkit, 2002. URL http://mallet.cs.umass.edu.

Peter McCullagh. Generalized linear models. *European Journal of Operational Research*, 16(3):285–292, 1984.

Tim Miller. Explanation in artificial intelligence: Insights from the social sciences. *arXiv preprint arXiv:1706.07269*, 2017.

Stager MM. *Kliegman RM, Stanton BF, St. Geme JW III, et al., eds. Nelson textbook of pediatrics*. Saunders Elsevier, 2012.

NIH Health Care Systems Research Collaboratory. Rethinking Clinical Trials: A Living Textbook of Pragmatic Clinical Trials. July 2014.

N. Parikh and S. P. Boyd. Proximal algorithms. *Foundations and Trends in optimization*, 2014.

J. Pathak, A. N. Kho, and J. C. Denny. Electronic health records-driven phenotyping: challenges, recent advances, and perspectives. *Journal of the American Medical Informatics Association*, 2013.

Judea Pearl. *Causality*. Cambridge university press, 2009.

Tao Qin and Tie-Yan Liu. Introducing letor 4.0 datasets. *arXiv preprint arXiv:1306.2597*, 2013.

D. Ramage, D. Hall, R. Nallapati, and C. D. Manning. Labeled lda: A supervised topic model for credit attribution in multi-labeled corpora. In *Empirical Methods in Natural Language Processing*, 2009.

Weiss RD. *Goldman L, Shafer AI, eds. Goldman's Cecil Medicine.*, volume 2. Elsevier Health Sciences, 2012.

Alfréd Rényi. On Measures Of Entropy And Information. In *Proceedings of the 4th Berkeley Symposium on Mathematics, Statistics and Probability*, 1960.

Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. Why should i trust you?: Explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1135–1144. ACM, 2016.

R. L. Richesson, W. E. Hammond, M. Nahm, D. Wixted, G. E. Simon, J. G. Robinson, A. E. Bauck, D. Cifelli, M. M. Smerek, J. Dickerson, et al. Electronic health records based phenotyping in next-generation clinical trials: a perspective from the nih health care systems collaboratory. *Journal of the American Medical Informatics Association*, 2013.

Virginia R De Sa. Spectral Clustering with Two Views. In *Proceedings of the Workshop on Learning with Multiple Views, International Conference on Machine Learning*, 2005.

M. Saeed, M. Villarroel, A. T. Reisner, G. Clifford, L. W. Lehman, G. Moody, T. Heldt, T. H. Kyaw, B. Moody, and R. G. Mark. Multiparameter intelligent monitoring in intensive care II (MIMIC-II): a public-access intensive care unit database. *Critical Care Medicine*, 2011.

Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. *See https://arxiv. org/abs/1610.02391 v3*, 7(8), 2016.

Avanti Shrikumar, Peyton Greenside, Anna Shcherbina, and Anshul Kundaje. Not just a black box: Learning important features through propagating activation differences. *arXiv preprint arXiv:1605.01713*, 2016.

Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Deep inside convolutional networks: Visualising image classification models and saliency maps. *arXiv preprint arXiv:1312.6034*, 2013.

Vikas Sindhwani, Partha Niyogi, and Mikhail Belkin. A Co-Regularization Approach to Semi-supervised Learning with Multiple Views. In *Proceedings of the Workshop on Learning with Multiple Views, 22nd ICML*, 2005.

Daniel Smilkov, Nikhil Thorat, Been Kim, Fernanda Viégas, and Martin Wattenberg. Smoothgrad: removing noise by adding noise. *arXiv preprint arXiv:1706.03825*, 2017.

Pierre Stock and Moustapha Cisse. Convnets and imagenet beyond accuracy: Explanations, bias detection, adversarial examples and model criticism. *arXiv preprint arXiv:1711.11443*, 2017.

Amos Storkey, Zhanxing Zhu, and Jinli Hu. A Continuum from Mixtures to Products: Aggregation under Bias. In *ICML Workshop on Divergence Methods for Probabilistic Inference*, 2014.

Alexander Strehl and Joydeep Ghosh. Cluster ensembles—a knowledge reuse framework for combining multiple partitions. *The Journal of Machine Learning Research*, 3, 2003.

Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks. *arXiv preprint arXiv:1703.01365*, 2017.

Martin Szummer and Emine Yilmaz. Semi-supervised learning to rank with preference regularization. In *Proceedings of the 20th ACM international conference on Information and knowledge management*, pages 269–278. ACM, 2011.

Andrew Trotman. Learning to rank. *Information Retrieval*, 8(3):359–381, 2005.

T. van Erven and P. Harremoës. Rényi Divergence and Kullback-Leibler Divergence. *ArXiv e-prints*, 2012.

N Volkow. Drugs, brains, and behavior: The science of addiction, 2014. URL http://www.drugabuse.gov/publications/drugs-brains-behavior-science-addiction/preface.

C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie. The Caltech-UCSD Birds-200-2011 Dataset. Technical Report CNS-TR-2011-001, California Institute of Technology, 2011.

M.J. Wainwright and M.I. Jordan. Graphical models, exponential families, and variational inference. *Foundations and Trends in Machine Learning*, 2008.

Fen Xia, Tie-Yan Liu, Jue Wang, Wensheng Zhang, and Hang Li. Listwise approach to learning to rank: theory and algorithm. In *Proceedings of the 25th international conference on Machine learning*, pages 1192–1199. ACM, 2008.

Eunho Yang, Pradeep K Ravikumar, Genevera I Allen, and Zhandong Liu. On poisson graphical models. In *Advances in Neural Information Processing Systems*, pages 1718–1726, 2013.