# Discriminative Nonparametric Latent Feature Relational Models with Data Augmentation

**Bei Chen[†], Ning Chen[‡*], Jun Zhu[†*], Jiaming Song[†], Bo Zhang[†]**
Dept. of Comp. Sci. & Tech., State Key Lab of Intell. Tech. & Sys., [†]Center for Bio-Inspired Computing Research,
[‡]MOE Key lab of Bioinformatics, Bioinformatics Division and Center for Synthetic & Systems Biology,
TNList, Tsinghua University, Beijing, 100084, China
{chenbei12@mails., ningchen@, dcszj@, sjm12@mails., dcszb@}tsinghua.edu.cn

## Abstract

We present a discriminative nonparametric latent feature relational model (LFRM) for link prediction to automatically infer the dimensionality of latent features. Under the generic RegBayes (regularized Bayesian inference) framework, we handily incorporate the prediction loss with probabilistic inference of a Bayesian model; set distinct regularization parameters for different types of links to handle the imbalance issue in real networks; and unify the analysis of both the smooth logistic log-loss and the piecewise linear hinge loss. For the nonconjugate posterior inference, we present a simple Gibbs sampler via data augmentation, without making restricting assumptions as done in variational methods. We further develop an approximate sampler using stochastic gradient Langevin dynamics to handle large networks with hundreds of thousands of entities and millions of links, orders of magnitude larger than what existing LFRM models can process. Extensive studies on various real networks show promising performance.

## Introduction

Link prediction is a fundamental task in statistical network analysis. For static networks, it is defined as predicting the missing links from a partially observed network topology (and some attributes if exist). Existing approaches include: 1) Unsupervised methods that design good proximity / similarity measures between nodes based on network topology features (Liben-Nowell and Kleinberg 2003), e.g., common neighbors, Jaccard's coefficient (Salton and McGill 1983), Adamic/Adar (Adamic and Adar 2003), etc; 2) Supervised methods that learn classifiers on labeled data with a set of manually designed features (Lichtenwalter, Lussier, and Chawla 2010; Hasan et al. 2006; Shi et al. 2009); 3) others (Backstrom and Leskovec 2011) that use random walks to combine the network structure information with node and edge attributes. One possible limitation for such methods is that they rely on well-designed features or measures, which can be time demanding to get and/or application specific.

Latent variable models (Hoff, Raftery, and Handcock 2002; Hoff 2007; Chang and Blei 2009) have been widely applied to discover latent structures from complex network data, based on which prediction models are developed for link prediction. Although these models work well, one remaining problem is how to determine the unknown number of latent classes or features. A typical way using model selection, e.g., cross-validation or likelihood ratio test (Liu and Shao 2003), can be computationally prohibitive by comparing many candidate models. Bayesian nonparametrics has shown promise in bypassing model selection by imposing an appropriate stochastic process prior on a rich class of models (Antoniak 1974; Griffiths and Ghahramani 2005). For link prediction, the infinite relational model (IRM) (Kemp et al. 2006) is class-based and uses Bayesian nonparametrics to discover systems of related concepts. One extension is the mixed membership stochastic blockmodel (MMSB) (Airoldi et al. 2008), which allows entities to have mixed membership. (Miller, Griffiths, and Jordan 2009) and (Zhu 2012) developed nonparametric latent feature relational models (L-FRM) by incorporating Indian Buffet Process (IBP) prior to resolve the unknown dimension of a latent feature space. Though LFRM has achieved promising results, exact inference is intractable due to the non-conjugacy of the prior and link likelihood. One has to use Metropolis-Hastings (Miller, Griffiths, and Jordan 2009), which may have low accept rates if the proposal distribution is not well designed, or variational inference (Zhu 2012) with truncated mean-field assumptions, which may be too strict in practice.

In this paper, we develop discriminative nonparametric latent feature relational models (DLFRM) by exploiting the ideas of data augmentation with simpler Gibbs sampling (Polson and Scott 2011; Polson, Scott, and Windle 2013) under the regularized Bayesian inference (RegBayes) framework (Zhu, Chen, and Xing 2014). Our major contributions are: 1) We use the RegBayes framework for DLFRM to deal with the imbalance issue in real networks and naturally analyze both the logistic log-loss and the max-margin hinge loss under a unified setting; 2) We explore data augmentation techniques to develop a simple Gibbs sampling algorithm, which is free from unnecessary truncation and assumptions that typically exist in variational approximation methods; 3) We develop an approximate Gibbs sampler using stochastic gradient Langevin dynamics, which can handle large networks with hundreds of thousands of entities and millions of links (See Table 1), orders of magnitude larger than what the existing LFRM models (Miller, Griffiths, and Jordan 2009;

[*]Corresponding authors.

Zhu 2012) can process; and 4) Finally, we conduct experimental studies on a wide range of real networks and the results demonstrate promising results of our methods.

## Nonparametric LFRM Models

We consider static networks with $N$ entities. Let $Y$ be the $N \times N$ binary link indicator matrix, where $y_{ij} = 1$ denotes the existence of a link from entity $i$ to $j$, and $y_{ij} = -1$ denotes no link from $i$ to $j$. $Y$ is not fully observed. Our goal is to learn a model from the partially observed links and predict the values of the unobserved entries of $Y$. Fig. 1 illustrates a latent feature relational model (LFRM), where each entity is represented by $K$ latent features. Let $Z$ be the $N \times K$ feature matrix, each row is associated with an entity and each column corresponds to a feature. We consider the binary features[1]: If entity $i$ has feature $k$, then $z_{ik} = 1$, otherwise $z_{ik} = 0$. Let $Z_i$ be the feature vector of entity $i$, $U$ be a $K \times K$ real-valued weight matrix, $\eta = \mathrm{vec}(U)$ and $Z_{ij} = \mathrm{vec}(Z_i^\top Z_j)$, where $\mathrm{vec}(A)$ is a vector concatenating the row vectors of matrix $A$. Note that $\eta$ and $Z_{ij}$ are column vectors, while $Z_i$ is a row vector. Then the probability of the link from entity $i$ to $j$ is

$$p(y_{ij} = 1|Z_i, Z_j, U) = \sigma\left(Z_i U Z_j^\top\right) = \sigma\left(\eta^\top Z_{ij}\right), \quad (1)$$

where $\sigma(x) = \frac{1}{1+\exp(-x)}$ is the sigmoid function. We assume that links are conditionally independent given $Z$ and $U$, then the link likelihood is $p(Y|Z,U) = \prod_{(i,j)\in\mathcal{I}} p(y_{ij}|Z_i, Z_j, U)$, where $\mathcal{I}$ is the set of training links (observed links).

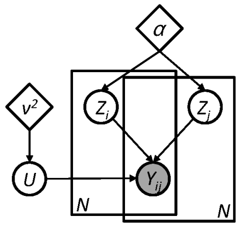


Figure 1: The graphical structure of LFRM.

In the above formulation, we assume that the dimensionality of the latent features $K$ is known a priori. However, this assumption is often unrealistic especially when dealing with large-scale applications. The conventional approaches that usually need a model selection procedure (e.g., cross validation) to choose an appropriate value by trying on a large set of candidates can be expensive and often require extensive human efforts on guiding the search. Recent progress on Bayesian optimization (Snoek, Larochelle, and Adams 2012) provides more effective solution to searching for good parameters, but still needs to learn many models under different configurations of the hyper-parameter $K$.

In this paper, we focus on the nonparametric Bayesian methods (Griffiths and Ghahramani 2005) for link prediction. The recently developed nonparametric latent feature relational models (LFRM) (Miller, Griffiths, and Jordan 2009) leverage the advancement of Bayesian nonparametric methods to automatically resolve the unknown dimensionality of the feature space by applying a flexible nonparametric prior. It assumes that each entity $i$ has an infinite number of binary features, that is $Z_i \in \{0,1\}^\infty$, and the Indian Buffet Process (IBP) (Griffiths and Ghahramani 2005) is used as a prior of $Z$ to produce a sparse latent feature vector for each entity.

We treat the weight matrix $U$ as random and put a prior on it for fully Bayesian inference. Then with Bayes' theorem, the posterior distribution is

$$q(Z, U|Y) \propto p_0(Z)p_0(U)p(Y|Z,U), \quad (2)$$

where the prior $p_0(Z)$ is an IBP and $p_0(U)$ is often assumed to be an isotropic Gaussian prior.

### Discriminative LFRM Models

The conventional Bayesian inference as above relies on Bayes' rule to infer the posterior distribution. In fact, this procedure can be equivalently formulated as solving an optimization problem. For example, the Bayes posterior in Eq. (2) is equivalent to the solution of the following problem:

$$\min_{q(Z,U)\in\mathcal{P}} \mathrm{KL}(q(Z,U)||p_0(Z,U)) - \mathbb{E}_q[\log p(Y|Z,U)], \quad (3)$$

where $\mathcal{P}$ is the space of well-defined distributions and $\mathrm{KL}(q||p)$ is the Kullback-Leibler (KL) divergence from $q$ to $p$. Such an optimization view has inspired the development of regularized Bayesian inference (RegBayes) which solves:

$$\min_{q(Z,U)\in\mathcal{P}} \mathrm{KL}(q(Z,U)||p_0(Z,U)) + c \cdot \mathcal{R}(q(Z,U)), \quad (4)$$

where $\mathcal{R}(q)$ is a posterior regularization defined on the target posterior distribution and $c$ is a non-negative regularization parameter that balances the prior part and the posterior regularization part. We refer the readers to (Zhu, Chen, and Xing 2014) for more details on a generic representation theorem of the solution and its application (Zhu et al. 2014; Mei, Zhu, and Zhu 2014) to learn latent feature models for classification. Below, we explore the ideas to develop effective latent feature relational models for link prediction.

Although we could define an averaging classifier and make predictions using the sign rule $\hat{y}_{ij} = \mathrm{sign}(\mathbb{E}_q[Z_i U Z_j^\top])$, the resulting problem needs to be approximately solved by truncated variational methods, which can be inaccurate in practice. Here, we propose to define a Gibbs classifier, which admits simple and efficient sampling algorithms that are guaranteed to be accurate. Our Gibbs sampler randomly draws the latent variables $(Z, U)$ from the unknown but preassumed to be given posterior distribution $q(Z, U)$. Once $Z$ and $U$ are given, we can make predictions using the sign rule $\hat{y}_{ij} = \mathrm{sign}(Z_i U Z_j^\top)$ and measure the training error $r(Z,U) = \sum_{(i,j)\in\mathcal{I}} \mathbb{I}(y_{ij} \neq \hat{y}_{ij})$, where $\mathbb{I}(\cdot)$ is an indicator function. Since the training error is non-smooth and non-convex, it is often relaxed by a well-behaved loss function. Let $\omega_{ij} = Z_i U Z_j^\top$, two well-studied examples are the logistic log-loss $r_1$ and the hinge loss $r_2$:

$$\begin{aligned} r_1(Z,U) &= -\sum_{(i,j)\in\mathcal{I}} \log p(\tilde{y}_{ij}|Z_i, Z_j, U), \\ r_2(Z,U) &= \sum_{(i,j)\in\mathcal{I}} (\ell - y_{ij}\omega_{ij})_+, \end{aligned}$$

where $p(\tilde{y}_{ij}|Z_i, Z_j, U) = \frac{e^{\omega_{ij}\tilde{y}_{ij}}}{1+e^{\omega_{ij}}}$, $(x)_+ := \max(0,x)$, $\ell$ is the pre-defined cost to penalize a wrong prediction, and

[1] Real-valued features can be learned by using composition, e.g., $R_i = Z_i \otimes H_i$, where $H_i$ is the real-valued vector representing the amplitudes of each feature while the binary vector $Z_i$ represents the presence of each feature.

$\tilde{y}_{ij} = (y_{ij} + 1)/2$ so that 0 refers to a negative link instead of $-1$. To account for the uncertainty of the latent variables, we define the posterior regularization as the expected loss:

$$\mathcal{R}_1(q(U,Z)) = \mathbb{E}_q[r_1(Z,U)], \mathcal{R}_2(q(U,Z)) = \mathbb{E}_q[r_2(Z,U)].$$

With these posterior regularization functions, we can do the RegBayes as in problem (4), where the parameter $c$ balances the influence between the prior distribution (i.e., KL divergence) and the observed link structure (i.e., the loss term). We define the un-normalized pseudo link likelihood

$$\varphi_1(\tilde{y}_{ij}|Z_i, Z_j, U) = \frac{(e^{\omega_{ij}})^{c\tilde{y}_{ij}}}{(1 + e^{\omega_{ij}})^c}, \quad (5)$$

$$\varphi_2(y_{ij}|Z_i, Z_j, U) = \exp(-2c(\ell - y_{ij}\omega_{ij})_+). \quad (6)$$

Then problem (4) can be written in the equivalent form:

$$\min_{q(Z,U)\in\mathcal{P}} \mathrm{KL}(q(Z,U)||p_0(Z,U)) - \mathbb{E}_q[\log \varphi(Y|Z,U)], \quad (7)$$

where $\varphi(Y|Z,U) = \prod_{i,j\in\mathcal{I}} \varphi(y_{ij}|Z_i, Z_j, U)$ and $\varphi$ can be $\varphi_1$ or $\varphi_2$. Then the optimal solution of (4) or (7) is the following posterior distribution with link likelihood

$$q(Z,U|Y) \propto p_0(Z)p_0(U)\varphi(Y|Z,U). \quad (8)$$

Notice that if adopting the logistic log-loss, we actually obtain a generalized pseudo-likelihood which is a powered form of likelihood in Eq. (1).

For real networks, positive links are often highly sparse as shown in Table 1. Such sparsity could lead to serious imbalance issues in supervised learning, where the negative examples are much more than positive examples. In order to deal with the imbalance issue in network data and make the model more flexible, we perform RegBayes by controlling the regularization parameter. For example, we can choose a larger $c$ value for the fewer positive links and a relatively smaller $c$ for the larger negative links. This strategy has shown effective in dealing with imbalanced data in (Chen et al. 2015; Zhu 2012). We will provide experiments to demonstrate the benefits of RegBayes on dealing with imbalanced networks when learning nonparametric LFRMs.

## Gibbs Sampling with Data Augmentation

As we do not have a conjugate prior on $U$, exact posterior inference is intractable. Previous inference methods for nonparametric LFRM use either Metropolis-Hastings (Miller, Griffiths, and Jordan 2009) or variational techniques (Zhu 2012) which can be either inefficient or too strict in practice. We explore the ideas of data augmentation to give the pseudo-likelihood a proper design, so that we can directly obtain posterior distributions and develop efficient Gibbs sampling algorithms. Specifically, our algorithm relies on the following unified representation lemma.

**Lemma 1.** *Both $\varphi_1$ and $\varphi_2$ can be represented as*

$$\varphi(y_{ij}|Z_i, Z_j, U) \propto \int_0^\infty \exp\left(\kappa_{ij}\omega_{ij} - \frac{\rho_{ij}\omega_{ij}^2}{2}\right)\phi(\lambda_{ij})\mathrm{d}\lambda_{ij},$$

*where for $\varphi_1$ we have*

$$\kappa_{ij} = c(\tilde{y}_{ij} - \frac{1}{2}),\ \rho_{ij} = \lambda_{ij},\ \phi(\lambda_{ij}) = \mathcal{PG}(\lambda_{ij}; c, 0);$$

*while for $\varphi_2$, let $\gamma_{ij} = \lambda_{ij}^{-1}$, we have*

$$\kappa_{ij} = cy_{ij}(1 + c\ell\gamma_{ij}),\ \rho_{ij} = c^2\gamma_{ij},\ \phi(\lambda_{ij}) = \mathcal{GIG}(\frac{1}{2}, 1, c^2\ell^2).$$

**Algorithm 1** Gibbs sampler for DLFRM

```
Init: draw $Z$ from IBP, $U$ from $\mathcal{N}(0, \nu^{-2})$; set $\lambda = 1$.
for $iter = 1, 2, \ldots, L$ do
    for $n = 1, 2, \ldots, N$ do
        draw $\{z_{nk}\}_{k=1}^K$ from Eq. (11);
        draw $k_n$ using Eq. (12).
        if $k_n > 0$ then
            update $K \leftarrow K + k_n$, update new weights;
        end if
    end for
    draw $U$ using Eq. (13) and draw $\lambda$ using Eq. (14).
end for
```

We have used $\mathcal{PG}$ to denote a Polya-Gamma distribution (Polson, Scott, and Windle 2013) and $\mathcal{GIG}$ to denote a generalized inverse Gaussian distribution. We defer the proof to Appendix A, which basically follows (Polson, Scott, and Windle 2013; Polson and Scott 2011) with some algebraic manipulation on re-organizing the terms.

## Sampling Algorithm

Lemma 1 suggests that the pseudo-likelihood $\varphi$ can be considered as the marginal of a higher dimensional distribution that includes the augmented variables $\lambda$:

$$\psi(\lambda, Y|Z, U) \propto \prod_{(i,j)\in\mathcal{I}} \exp\left(\kappa_{ij}\omega_{ij} - \frac{\rho_{ij}\omega_{ij}^2}{2}\right)\phi(\lambda_{ij}), \quad (9)$$

which is a mixture of Gaussian components of $U$ once $Z$ is given, suggesting that we can effectively perform Gibbs sampling if a conjugate Gaussian prior is imposed on $U$. We also construct the complete posterior distribution:

$$q(Z, U, \lambda|Y) \propto p_0(Z)p_0(U)\psi(\lambda, Y|Z, U), \quad (10)$$

such that our target posterior $q(Z, U|Y)$ is a marginal distribution of the complete posterior. Therefore, if we can draw a set of samples $\{(Z_t, U_t, \lambda_t)\}_{t=1}^L$ from the complete posterior, by dropping the augmented variables, the rest samples $\{(Z_t, U_t)\}_{t=1}^L$ are drawn from the target posterior $q(Z, U|Y)$. This technique allows us to sample the complete posterior via a Gibbs sampling algorithm, as outlined in Alg. 1 and detailed below.

**For** $Z$**:** We assume the Indian Buffet Process (IBP) prior on the latent feature $Z$. Although the total number of latent features is infinite, every time we only need to store $K$ active features that are not all zero in the columns of $Z$. When sampling the $n$-th row, we need to consider two cases, due to the nonparametric nature of IBP.

First, for the active features, we sample $z_{nk}(k = 1, ..., K)$ in succession from the following conditional distribution

$$q(z_{nk}|Z_{-nk}, \eta, \lambda) \propto p(z_{nk})\psi(\lambda, Y|Z_{-nk}, \eta, z_{nk}), \quad (11)$$

where $p(z_{nk} = 1) \propto m_{-n,k}$ and $m_{-n,k}$ is the number of entities containing feature $k$ except entity $n$.

Second, for the infinite number of remaining all-zero features, we sample $k_n$ number of new features and add them to the $n$th row. Then we get the new $N \times (K + k_n)$ matrix $Z^*$ which becomes old when sampling the $(n + 1)$-th row. Every time when the number of features changes, we also update $U$ and extend it to a $(K + k_n) \times (K + k_n)$ matrix $U^*$. Let $Z'$ and $U'$ be the parts of $Z^*$ and $U^*$ that correspond to the $k_n$ new features. Also, we define $\eta' = \text{vec}(U')$. During implementation, we can delete the all-zero columns after every resampling of $Z$, but here we ignore it. Let $\eta$ follow the isotropic Normal prior $\mathcal{N}(0, \nu^{-2})$. Now the conditional distribution for $k_n = 0$ is $p(k_n = 0|Z, \eta, \lambda) = p_0(k_n)$, and the probability of $k_n \neq 0$ is

$$p(k_n \neq 0|Z, \eta, \lambda) = p_0(k_n)|\Sigma|^{\frac{1}{2}} \nu^D \exp\left(\frac{1}{2}\mu^\top \Sigma^{-1} \mu\right), \quad (12)$$

where $p_0(k_n) = \text{Poisson}\left(k_n; \frac{\alpha}{N}\right)$ is from the IBP prior, $D = 2k_nK + k_n^2$ is the dimension of $\eta'$ and the mean $\mu = \Sigma(\sum_{(i,j)\in\mathcal{I}}(\kappa_{ij} - \rho_{ij}\omega_{ij})Z'_{ij})$ , covariance $\Sigma = (\sum_{(i,j)\in\mathcal{I}} \rho_{ij} Z'_{ij} {Z'_{ij}}^\top + \nu^2 I)^{-1}$.

We compute the probabilities for $k_n = 0, 1, ..., K_{max}$, do normalization and sample from the resulting multinomial. Here, $K_{max}$ is the maximum number of features to add. Once we have added $k_n(\neq 0)$ new features, we should also sample their weights $\eta'$, which follow a $D$ dimensional multivariate Gaussian, in order to resample the next row of $Z$.

**For $U$:** After the update of $Z$, we resample $U$ given the new $Z$. Let $\tilde{D} = K \times K$ and $\eta$ follow the isotropic Normal prior $p_0(\eta) = \prod_{d=1}^{\tilde{D}} \mathcal{N}(\eta_d; 0, \nu^{-2})$. Then the posterior is also a Gaussian distribution

$$q(\eta|\lambda, Z) \propto p_0(\eta)\psi(\lambda, Y|Z, \eta) = \mathcal{N}(\eta; \tilde{\mu}, \tilde{\Sigma}), \quad (13)$$

with the mean $\tilde{\mu} = \tilde{\Sigma}(\sum_{(i,j)\in\mathcal{I}} \kappa_{ij} Z_{ij})$ and the convariance $\tilde{\Sigma} = (\sum_{(i,j)\in\mathcal{I}} \rho_{ij} Z_{ij} Z_{ij}^\top + \nu^2 I)^{-1}$.

**For $\boldsymbol{\lambda}$:** Since the auxiliary variables are independent given the new $Z$ and $U$, we can draw each $\lambda_{ij}$ separately. From the unified representation, we have

$$q(\lambda_{ij}|Z, \eta) \propto \exp\left(\kappa_{ij}\omega_{ij} - \frac{\rho_{ij}\omega_{ij}^2}{2}\right)\phi(\lambda_{ij}). \quad (14)$$

By doing some algebra, we can get the following equations. For $\varphi_1$, $\lambda_{ij}$ still follows a Polya-Gamma distribution $q(\lambda_{ij}|Z, \eta) = \mathcal{PG}(\lambda_{ij}; c, \omega_{ij})$, from which a sample can be efficiently drawn. For $\varphi_2$, $\lambda_{ij}$ follows a generalized inverse Gaussian distribution $q(\lambda_{ij}|Z, U, Y) = \mathcal{GIG}(\frac{1}{2}, 1, c^2\zeta_{ij}^2)$, where $\zeta_{ij} = \ell - y_{ij}\omega_{ij}$. Then $\gamma_{ij} := \lambda_{ij}^{-1}$ follows an inverse Gaussian distribution $q(\gamma_{ij}|Z, U, Y) = \mathcal{IG}(\frac{1}{c|\zeta_{ij}|}, 1)$, from which a sample can be easily drawn in a constant time.

## Stochastic Gradient Langevin Dynamics

Alg. 1 needs to sample from a $K^2$-dim Gaussian distribution to get $U$, where $K$ is the latent feature dimension. This procedure is prohibitively expensive for large networks when $K$ is large (e.g., $K > 40$). To address this problem, we employ stochastic gradient Langevin dynamics (SGLD) (Welling and Teh 2011), an efficient gradient-based MCMC method that uses unbiased estimates of gradients with random mini-batches. Let $\theta$ denote the model parameters and $p(\theta)$ is a prior distribution. Given a set of i.i.d data points $\mathcal{D} = \{x_i\}_{i=1}^M$, the likelihood is $p(\mathcal{D}|\theta) = \prod_{i=1}^M p(x_i|\theta)$. At each iteration $t$, the update equation for $\theta$ is:

$$\Delta\theta_t = \frac{\epsilon_t}{2}\left(\nabla \log p(\theta_t) + \frac{M}{m}\sum_{x_i\in\mathcal{D}_t} \nabla \log p(x_i|\theta_t)\right) + \delta_t, \quad (15)$$

where $\epsilon_t$ is the step size, $\mathcal{D}_t$ is a subset of $\mathcal{D}$ with size $m$ and $\delta_t \sim \mathcal{N}(0, \epsilon_t)$ is the Gaussian noise. When the stepsize is annealed properly, the Markov chain will converge to the true posterior distribution.

Table 1: Statistics of datasets.

| Dataset | NIPS | Kinship | WebKB | AstroPh | Gowalla |
|---|---|---|---|---|---|
| Entities | 234 | 104 | 877 | 17,903 | 196,591 |
| Positive Links | 1,196 | 415 | 1,608 | 391,462 | 1,900,654 |
| Sparsity Rate | 2.2% | 4.1% | 0.21% | 0.12% | 0.0049% |

Let $\mathcal{I}_t$ be a subset of $\mathcal{I}$ with size $m$. We can apply SGLD to sample $\eta$ (i.e., $U$). Specifically, according to the true posterior of $\eta$ as in Eq. (13), the update rule is:

$$\Delta\eta_t = \frac{\epsilon_t}{2}\left(-\nu^2\eta_t + \frac{|\mathcal{I}|}{m}\sum_{(i,j)\in\mathcal{I}_t}(\kappa_{ij} - \rho_{ij}\omega_{ij})Z_{ij}\right) + \delta_t, \quad (16)$$

where $\delta_t$ is a $K^2$-dimensional vector and each entry is a Gaussian noise. After a few iterations, we will get the approximate sampler of $\eta$ (i.e., $U$) very efficiently.

## Experiments

We present experimental results to demonstrate the effectiveness of DLFRM on five real datasets as summarized in Table 1, where **NIPS** contains 234 authors who have the most coauthor-relationships with others from NIPS 1-17; **Kinship** includes 26 relationships of 104 people in the Alyawarra tribe in central Australia; **WebKB** contains 877 webpages from the CS departments of different universities, where the dictionary has $1,703$ unique words; **AstroPh** contains collaborations between $17,903$ authors of papers submitted to Arxiv Astro Physics in the period from Jan. 1993 to Apr. 2003 (Leskovec, Kleinberg, and Faloutsos 2007); and **Gowalla** contains $196,591$ people and their friendships on Gowalla social website (Cho, Myers, and Leskovec 2011). All these real networks have very sparse links.

We evaluate three variants of our model: (1) **DLFRM**: to overcome the imbalance issue, we set $c^+ = 10c^- = c$ as in (Zhu 2012), where $c^+$ is the regularization parameter for positive links and $c^-$ for negative links. We use a full asymmetric weight matrix $U$; (2) **stoDLFRM**: the DLFRM model that uses SGLD to sample weight matrix $U$, where the stepsizes are set by $\epsilon_t = a(b + t)^{-\gamma}$ for log-loss and AdaGrad (Duchi, Hazan, and Singer 2011) for hinge loss; (3) **diagDLFRM**: the DLFRM that uses a diagonal weight matrix $U$. Each variant can be implemented with the logistic log-loss or hinge loss, denoted by the superscript $l$ or $h$.

We randomly select a development set from training set with almost the same number of links as testing set and

Table 2: AUC on the NIPS coauthorship and Kinship dataset.

| Models | NIPS | Kinship |
|---|---|---|
| MMSB | 0.8705 ± − | 0.9005 ± 0.0022 |
| IRM | 0.8906 ± − | 0.9310 ± 0.0023 |
| LFRM rand | 0.9466 ± − | 0.9443 ± 0.0018 |
| LFRM w / IRM | 0.9509 ± − | 0.9346 ± 0.0013 |
| MedLFRM | 0.9642 ± 0.0026 | 0.9552 ± 0.0065 |
| BayesMedLFRM | 0.9636 ± 0.0036 | 0.9547 ± 0.0028 |
| DLFRM$^l$ | **0.9812** ± 0.0013 | **0.9650** ± 0.0032 |
| stoDLFRM$^l$ | **0.9804** ± 0.0007 | **0.9673** ± 0.0044 |
| diagDLFRM$^l$ | 0.9717 ± 0.0031 | 0.9426 ± 0.0028 |
| DLFRM$^h$ | **0.9806** ± 0.0027 | **0.9640** ± 0.0023 |
| stoDLFRM$^h$ | **0.9787** ± 0.0012 | **0.9657** ± 0.0031 |
| diagDLFRM$^h$ | 0.9722 ± 0.0021 | 0.9440 ± 0.0038 |

choose the proper hyper-parameters, which are insensitive in a wide range. All the results are averaged over 5 runs with random initializations and the same group of parameters.

## Results on Small Networks

We first report the prediction performance (AUC scores) on three relatively small networks. For fair comparison, we follow the previous settings to randomly choose 80% of the links for training and use the remaining 20% for testing. AUC score is the area under the Receiver Operating Characteristic (ROC) curve; higher is better.

**NIPS Coauthorship Prediction** Table 2 shows the AUC scores on NIPS dataset, where the results of baselines (i.e., LFRM, IRM, MMSB, MedLFRM and BayesMedLFRM) are cited from (Miller, Griffiths, and Jordan 2009; Zhu 2012). We can see that both DLFRM$^l$ and DLFRM$^h$ outperform all other models, which suggests that our exact Gibbs sampling with data augmentation can lead to more accurate models than MedLFRM / BayesMedLFRM that uses the variational approximation methods with truncated mean-field assumptions. The stoDLFRMs obtain comparable results to DLFRMs, which suggests that approximate sampler for $\eta$ using SGLD is very effective. With SGLD, we can improve efficiency without sacrificing performance which we will discuss later with Table 4. Furthermore, diagDLFRM$^l$ and diagDLFRM$^h$ also perform well, as they beat all other methods except (sto)DLFRMs. By using a lower dimensional $\eta$ derived from the diagonal weight matrix $U$, diagDLFRM has the advantage of being computationally efficient, as shown in Fig. 3(d). The good performance of stoDLFRMs and diagDLFRMs suggests that we can use SGLD with a full diagonal weight matrix or simply use a diagonal weight matrix on large-scale networks.

**Kinship Multi-relation Prediction** For multi-relational Kinship dataset, we consider the "single" setting (Miller, Griffiths, and Jordan 2009), where we infer an independent set of latent features for each relation. The overall AUC is obtained by averaging the results of all relations. As shown in Table 2, both (sto)DLFRM$^l$ and (sto)DLFRM$^h$ outperform all other methods, which again proves the effectiveness of our methods. Furthermore, the diagonal variants also obtain fairly good results, close to the best baselines. Finally, the better results by the discriminative methods in general demonstrate the effect of RegBayes on using various regularization parameters to deal with the imbalance issue; Fig. 3(b) provides a detailed sensitivity analysis.

Table 3: AUC scores on the WebKB dataset.

| Models | WebKB |
|---|---|
| Katz | 0.5625 ± − |
| Linear SVM | 0.6889 ± − |
| RBF SVM | 0.7132 ± − |
| MedLFRM | 0.7326 ± 0.0010 |
| DLFRM$^l$ | **0.8039** ± 0.0057 |
| stoDLFRM$^l$ | **0.8044** ± 0.0058 |
| diagDLFRM$^l$ | 0.7954 ± 0.0085 |
| DLFRM$^h$ | **0.8002** ± 0.0073 |
| stoDLFRM$^h$ | **0.7966** ± 0.0013 |
| diagDLFRM$^h$ | 0.7900 ± 0.0056 |

**WebKB Hyperlink Prediction** We also examine how DLFRMs perform on WebKB network, which has rich text attributes (Craven et al. 1998). Our baselines include: 1) **Katz**: a proximity measure between two entities—it directly sums over all collection of paths, exponentially damped by the path length to count short paths more heavily (Liben-Nowell and Kleinberg 2003); 2) **Linear SVM**: a supervised learning method using linear SVM, where the feature for each link is a vector concatenating the bag-of-words features of two entities; 3) **RBF-SVM**: SVM with the RBF kernel on the same features as the linear SVM. We use SVM-Light (Joachims 1998) to train these classifiers; and 4) **MedLFRM**: state-of-the-art methods on learning latent features for link prediction (Zhu 2012). Note that we don't compare with the relational topic models (Chang and Blei 2009; Chen et al. 2015), whose settings are quite different from ours. Table 3 shows the AUC scores of various methods. We can see that: 1) both MedLFRM and DLFRMs perform better than SVM classifiers on raw bag-of-words features, showing the promise of learning latent features for link prediction on document networks; 2) DLFRMs are much better than MedLFRM[2], suggesting the advantages of using data augmentation techniques for accurate inference over variational methods with truncated mean-field assumptions; and 3) both stoDLFRMs and diagDLFRMs achieve competitive results with faster speed.

## Results on Large Networks

We now present results on two much larger networks. As the networks are much sparser, we randomly select 90% of the positive links for training and the number of negative training links is 10 times the number of positive training links. The testing set contains the remaining 10% of the positive links and the same number of negative links, which we uniformly sample from the negative links outside the training set. This test setting is the same as that in (Kim et al. 2013).

**AstroPh Collaboration Prediction** Fig. 2 presents the test AUC scores, where the results of the state-of-the-art nonparametric models aMMSB (assortative MMSB) and

[2] MedLFRM results are only available when truncation level $<$ 20 due to its inefficiency.

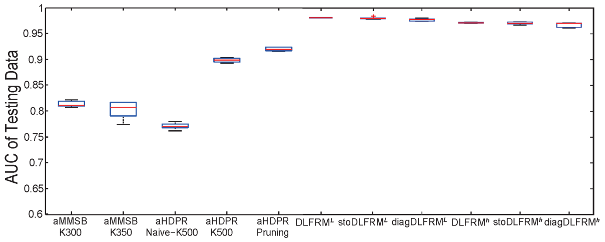


Figure 2: AUC scores on the AstroPh dataset.

Table 4: Split of training time (sec) on AstroPh dataset.

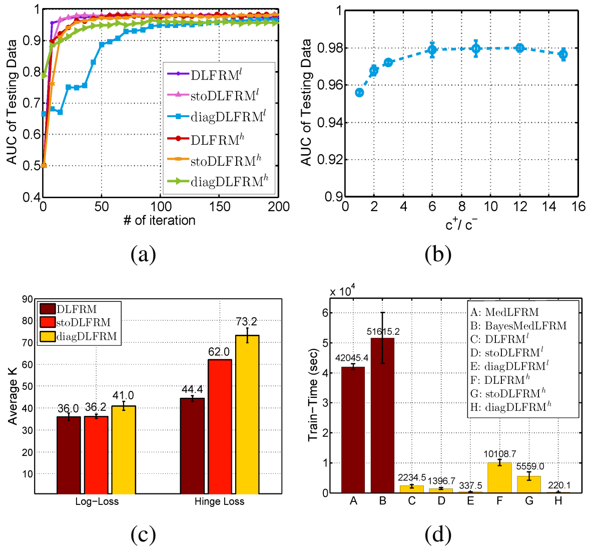
| Models | DLFRM$^l$ | stoDLFRM$^l$ |
|---|---|---|
| Sample $Z$ | 16312.0 (25.58%) | 32095.9 (95.18%) |
| Sample $U$ | 47389.9 (74.32%) | 1516.4 (4.50%) |
| Sample $\lambda$ | 65.7 (0.10%) | 109.0 (0.32%) |

aHDPR (assortative HDP relational model, a nonparametric generalization of aMMSB) are cited from (Kim et al. 2013). We can see that DLFRMs achieve significantly better AUCs than aMMSB and aHDPR, which again demonstrates that our models can not only automatically infer the latent dimension, but also learn the effective latent features for entities. Furthermore, stoDLRMs and diagDLFRMs show larger benefits on the larger networks due to the efficiency. As shown in Table 4, the time for sampling $U$ is greatly reduced with SGLD. It only accounts for $4.50\%$ of the whole time for stoDLFRM$^l$, while the number is $74.32\%$ for DLFRM$^l$.

Table 5: AUC scores on Gowalla dataset.

| Models | AUC | Time (sec) |
|---|---|---|
| CN | 0.8823 ± − | 12.3 ± 0.3 |
| Jaccard | 0.8636 ± − | 11.7 ± 0.5 |
| Katz | 0.9145 ± − | 8336.9 ± 306.9 |
| stoDLFRM$^l$ | **0.9722** ± 0.0013 | 220191.4 ± 4420.2 |
| diagDLFRM$^l$ | **0.9680** ± 0.0009 | 7344.5 ± 943.7 |

**Gowalla Friendship Prediction** Finally, we test on the largest Gowalla network, which is out of reach for many state-of-art methods, including LFRM, MedLFRM and our DLFRMs without SGLD. Some previous works combine the geographical information of Gowalla social network to analyze user movements or friendships (Cho, Myers, and Leskovec 2011; Scellato, Noulas, and Mascolo 2011), but we are not aware of any fairly comparable results for our setting of link prediction. Here, we present the results of some proximitiy-measure based methods, including common neighbors (**CN**), **Jaccard** coefficient, and **Katz**. As the network is too large to search for all the paths, we only concern the paths that shorter than 4 for Katz. As shown in previous results and Fig. 3(d), DLFRMs with logistic log-loss are more efficient and have comparable results of DLFRMs with hinge loss, so we only show the results of stoDLFRM$^l$ and diagDLFRM$^l$. The AUC scores and training time are shown in Table 5. We can see that stoDLFRM$^l$ outperforms all the other methods and diagDLFRM$^l$ obtain competitive results. Our diagDLFRM$^l$ gets much better performance than the best baseline with less time. It shows that our models can also deal with the large-scale networks.



(a)

(b)

(c)

(d)

Figure 3: (a) Sensitivity of burn in iterations; (b) Sensitivity of $c^+/c^-$ with DLFRM$^l$; (c) Average latent dimension $K$; (d) Training time of various models on NIPS dataset.

## Closer Analysis

We use NIPS network as an example to provide closer analysis. Similar observations can be obtained in larger networks[3], but taking longer time to run.

**Sensitivity to Burn-In** Fig. 3(a) shows the test AUC scores w.r.t. the number of burn-in steps. We can see that all our variant models converge quickly to stable results. The diagDLFRM$^l$ is a bit slower, but still within 150 steps. These results demonstrate the stability of our Gibbs sampler.

**Sensitivity to Parameter $c$** To study how the regularization parameter $c$ handles the imbalance in real networks, we change the value of $c^+/c^-$ for DLFRM$^l$ from 1 to 15 (with all other parameters selected by the development set); and report AUC scores in Fig. 3(b). The first point (i.e., $c^+ = c^- = 1$) corresponds to LFRM with our Gibbs sampler, whose lower AUC demonstrates the effectiveness of a larger $c^+/c^-$ to deal with the imbalance issue. We can see that the AUC score increases when $c^+/c^-$ becomes larger and the prediction performance is stable in a wide range (e.g., $6 < c^+/c^- < 12$). How large $c^+/c^-$ a network needs depends on its sparsity. A rule of thumb is that the sparser a network is, the larger $c^+/c^-$ it may prefer. The results also show that our setting ($c^+ = 10c^-$) is reasonable.

**Latent Dimensions** Fig. 3(c) shows the number of latent features automatically learnt by variant models. We can see that diagDLFRMs generally need more features than DLFRMs because the simplified weight matrix $U$ doesn't

[3] Closer Analysis on AstroPh dataset: http://bigml.cs.tsinghua.edu.cn/%7Ebeichen/pub/DLFRM2.pdf

consider pairwise interactions between features. Moreover, DLFRM$^h$ needs more features than DLFRM$^l$, possibly because of the non-smoothness nature of hinge loss. The small variance of each method suggests that the latent dimensions are stable in independent runs with random initializations.

**Running Time** Fig. 3(d) compares the training time. It demonstrates all our variant models are more efficient than MedLFRM and BayesMedLFRM (Zhu 2012) that use truncated mean-field approximation. Compared to DLFRM$^l$, DLFRM$^h$ takes more time to get the good AUC. The reason is that DLFRM$^h$ often converges slower (see Fig. 3(a)) with a larger latent dimension $K$ (see Fig. 3(c)). stoDLFRMs are more effective as we have discussed before. diagDLFRMs are much more efficient due to the linear increase of training time per iteration with respect to $K$. The testing time for all the methods are very little, omitted due to space limit.

Overall, DLFRMs improve prediction performance and are more efficient in training, compared with other state-of-the-art nonparametric LFRMs.

## Conclusions and Future Work

We present discriminative nonparametric LFRMs for link prediction, which can automatically resolve the unknown dimensionality of the latent feature space with a simple Gibbs sampler using data augmentation; unify the analysis for both logistic log-loss and hinge loss; and deal with the imbalance issue in real networks. Experimental results on a wide range of real networks demonstrate superior performance and scalability. For future work, we are interested in developing more efficient algorithms (e.g., using distributed computing) to solve the link prediction problem in web-scale networks.

## Acknowledgments

The work was supported by the National Basic Research Program (973 Program) of China (Nos. 2013CB329403, 2012CB316301), National NSF of China (Nos. 61305066, 61322308, 61332007), TNList Big Data Initiative, and Tsinghua Initiative Scientific Research Program (Nos. 20121088071, 20141080934).

## References

Adamic, L., and Adar, E. 2003. Friends and neighbors on the web. *Social Networks* 25(3):211–230.

Airoldi, E.; Blei, D.; Fienberg, S.; and Xing, E. 2008. Mixed membership stochastic blockmodels. *JMLR*.

Antoniak, C. 1974. Mixtures of Dirichlet processes with applications to Bayesian nonparametric problems. *Annals of Statistics* 1152–1174.

Backstrom, L., and Leskovec, J. 2011. Supervised random walks: Predicting and recommending links in social networks. In *WSDM*.

Chang, J., and Blei, D. 2009. Relational topic models for document networks. In *AISTATS*.

Chen, N.; Zhu, J.; Xia, F.; and Zhang, B. 2015. Discriminative relational topic models. *PAMI* 37(5):973–986.

Cho, E.; Myers, S.; and Leskovec, J. 2011. Friendship and mobility: User movement in location-based social networks. In *SIGKDD*.

Craven, M.; DiPasquo, D.; Freitag, D.; McCallum, A.; Mitchell, T.; Nigam, K.; and Slattery, S. 1998. Learning to extract symbolic knowledge from the world wide web. In *NCAI*.

Duchi, J.; Hazan, E.; and Singer, Y. 2011. Adaptive subgradient methods for online learning and stochastic optimization. *JMLR*.

Griffiths, T., and Ghahramani, Z. 2005. Infinite latent feature models and the indian buffet process. In *NIPS*.

Hasan, M. A.; Chaoji, V.; Salem, S.; and Zaki, M. 2006. Link prediction using supervised learning. In *SDM: Workshop on Link Analysis, Counter-terrorism and Security*.

Hoff, P.; Raftery, A.; and Handcock, M. 2002. Latent space approaches to social network analysis. *Journal of the American Statistical Association* 97(460):1090–1098.

Hoff, P. 2007. Modeling homophily and stochastic equivalence in symmetric relational data. In *NIPS*.

Joachims, T. 1998. Making large-scale svm learning practical. In *Universität Dortmund, LS VIII-Report*.

Kemp, C.; Tenenbaum, J.; Griffiths, T.; Yamada, T.; and Ueda, N. 2006. Learning systems of concepts with an infinite relational model. In *AAAI*.

Kim, D.; Gopalan, P.; Blei, D.; and Sudderth, E. 2013. Efficient online inference for bayesian nonparametric relational models. In *NIPS*.

Leskovec, J.; Kleinberg, J.; and Faloutsos, C. 2007. Graph evolution: Densification and shrinking diameters. *TKDD*.

Liben-Nowell, D., and Kleinberg, J. 2003. The link-prediction problem for social networks. In *CIKM*.

Lichtenwalter, R.; Lussier, J.; and Chawla, N. 2010. New perspectives and methods in link prediction. In *SIGKDD*.

Liu, X., and Shao, Y. 2003. Asymptotics for likelihood ratio tests under loss of identifiability. *Annals of Statistics* 31(3):807–832.

Mei, S.; Zhu, J.; and Zhu, X. 2014. Robust RegBayes: Selectively incorporating first-order logic domain knowledge into bayesian models. In *ICML*.

Miller, K.; Griffiths, T.; and Jordan, M. 2009. Nonparametric latent feature models for link prediction. In *NIPS*.

Polson, N., and Scott, S. 2011. Data augmentation for support vector machines. *Bayesian Analysis* 6(1):1–23.

Polson, N.; Scott, J.; and Windle, J. 2013. Bayesian inference for logistic models using polya-gamma latent variables. *Journal of the American Statistical Association*.

Salton, G., and McGill, M. 1983. *Introduction to Modern Information Retrieval*. McGraw-Hill.

Scellato, S.; Noulas, A.; and Mascolo, C. 2011. Exploiting place features in link prediction on location-based social networks. In *SIGKDD*.

Shi, X.; Zhu, J.; Cai, R.; and Zhang, L. 2009. User grouping behavior in online forums. In *SIGKDD*.

Snoek, J.; Larochelle, H.; and Adams, R. 2012. Practical bayesian optimization of machine learning algorithms. In *NIPS*.

Welling, M., and Teh, Y. 2011. Bayesian learning via stochastic gradient langevin dynamics. In *ICML*.

Zhu, J.; Chen, N.; Perkins, H.; and Zhang, B. 2014. Gibbs max-margin topic models with data augmentation. *JMLR* 1073–1110.

Zhu, J.; Chen, N.; and Xing, E. 2014. Bayesian inference with posterior regularization and applications to infinite latent svms. *JMLR* 15(1):1799–1847.

Zhu, J. 2012. Max-margin nonparametric latent feature models for link prediction. In *ICML*.

# Supplemental Material

## Appendix A: The Proof of Lemma 1

We prove the cases of logistic log-loss and hinge loss in Lemma 1 respectively.

*Proof.* For the case with logistic log-loss, we directly follow the data-augmentation strategy from (Polson, Scott, and Windle 2013). Let $X$ follow a Polya-Gamma distribution, denoted by $X \sim \mathcal{PG}(a, b)$, that is

$$X = \frac{1}{2\pi^2} \sum_{d=1}^{\infty} \frac{g_d}{(d - 1/2)^2 + b^2/(4\pi^2)}, \tag{17}$$

where $a > 0$ and $b \in \mathcal{R}$ are parameters and each $g_d \sim \mathcal{G}(a, 1)$ is an independent Gamma random variable. The main result of (Polson, Scott, and Windle 2013) provides an alternative expression for the form of $\varphi_1$ in Eq. (5) by incorporating an augmented variable $\lambda$:

$$\varphi_1(\tilde{y}_{ij}|Z_i, Z_j, U) = \frac{1}{2^c} \int_0^\infty \exp\Big(\kappa_{ij}\omega_{ij} - \frac{\lambda_{ij}\omega_{ij}^2}{2}\Big)\phi(\lambda_{ij})\mathrm{d}\lambda_{ij}, \tag{18}$$

where $\kappa_{ij} = c(\tilde{y}_{ij} - \frac{1}{2})$ and $\phi(\lambda_{ij}) = \mathcal{PG}(\lambda_{ij}; c, 0)$.

For the case with hinge loss, we take the advantage of data augmentation for support vector machines (Polson and Scott 2011) and $\varphi_2$ in Eq. (6) can be represented as a scale mixture of Gaussian distributions:

$$\varphi_2(y_{ij}|Z_i, Z_j, U) = \int_0^\infty \frac{1}{\sqrt{2\pi\lambda_{ij}}} \exp\Big(-\frac{(\lambda_{ij} + c\zeta_{ij})^2}{2\lambda_{ij}}\Big)\mathrm{d}\lambda_{ij}, \tag{19}$$

where $\zeta_{ij} = \ell - y_{ij}\omega_{ij}$ and $\lambda_{ij}$ is the augmented variable. By reformulating similar terms in Eq. (19), we have:

$$\begin{aligned} \varphi_2(y_{ij}|Z_i, Z_j, U) \propto & \int_0^\infty \frac{1}{\sqrt{2\pi\lambda_{ij}}} \exp\Big(-\frac{1}{2}\Big(\frac{c^2\ell^2}{\lambda_{ij}} + \lambda_{ij}\Big)\Big) \\ & \exp\Big(cy_{ij}\Big(1 + \frac{c\ell}{\lambda_{ij}}\Big)\omega_{ij} - \frac{c^2\omega_{ij}^2}{2\lambda_{ij}}\Big)\mathrm{d}\lambda_{ij} \\ \propto & \int_0^\infty \exp\Big(\kappa_{ij}\omega_{ij} - \frac{\rho_{ij}\omega_{ij}^2}{2}\Big)\phi(\lambda_{ij})\mathrm{d}\lambda_{ij} \end{aligned} \tag{20}$$

where $\kappa_{ij} = cy_{ij}(1 + c\ell\lambda_{ij}^{-1})$, $\rho_{ij} = c^2\lambda_{ij}^{-1}$ and $\phi(\lambda_{ij}) = \mathcal{GIG}(\frac{1}{2}, 1, c^2\ell^2)$. Given the results of Eq. (18) and Eq. (20), Lemma 1 holds true. □

## Appendix B: Closer Analysis on AstroPh dataset

Here, we provide more closer analysis on AstroPh dataset which is much larger than the NIPS dataset.

**Sensitivity to Burn-In** Fig. 4(a) shows the AUC scores on testing data with respect to the number of burn-in steps on AstroPh dataset. We can observe that all our variant models converge quickly to stable results, similar as on NIPS dataset. Our DLFRMs with full weight matrix (e.g., DLFRM$^l$, DLFRM$^h$, stoDLFRM$^l$ and stoDLFRM$^h$) converge quickly within 10 steps. The diagDLFRMs need more steps to converge, but still within 40 steps to converge to stable results. These results demonstrate the stability of our Gibbs sampling algorithm.

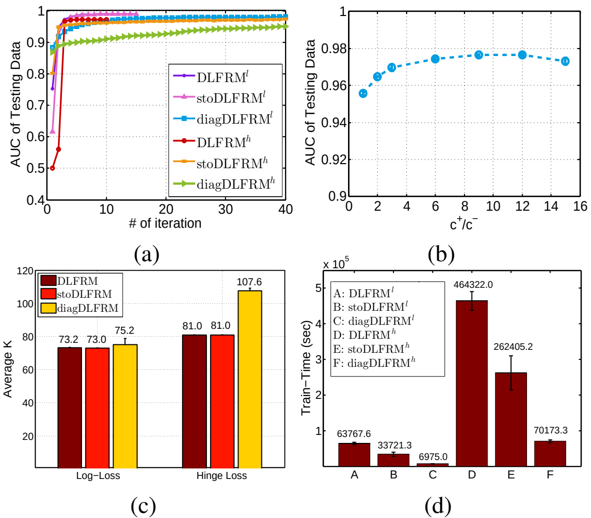


Figure 4: (a) Sensitivity of burn in iterations; (b) Sensitivity of $c^+/c^-$ with diagDLFRM$^l$; (c) Average latent dimension $K$; (d) Training time of various models on AstroPh dataset.

**Sensitivity to Parameter $c$** We analyze how the regularization parameter $c$ handles the imbalance in real networks using diagDLFRM$^l$, which is very efficient (see Fig. 4(d)). Following the settings on NIPS dataset, we change the ratio of $c^+/c^-$ for diagDLFRM$^l$ from 1 to 15 with all the parameters selected by the development set. As shown in Fig. 4(b), the AUC score increases when $c^+/c^-$ becomes larger and the prediction performance is stable in a wide range (e.g., $6 < c^+/c^- < 12$). These observations again demonstrate that using a larger $c^+$ than $c^-$ can effectively deal with the imbalance issue and our setting ($c^+ = 10c^-$) is reasonable.

**Latent Dimensions** Our variant models take the advantage of nonparametric technique to automatically learn the dimension of the latent features as shown in Fig. 4(c). We can see that diagDLFRMs generally need more features than DLFRMs because the simplified weight matrix $U$ does not consider pairwise interactions between features. Moreover, DLFRM$^h$ needs more features than DLFRM$^l$, possibly because of the non-smoothness nature of hinge loss. The small variance of each method suggests that the latent dimensions are stable in independent runs with random initializations.

**Running Time** The training time of our variant models on AstroPh dataset is shown in Fig. 4(d). We can see that for this relatively large network (with tens of thousands of entities and millions of links), the least time we need to obtain the good AUC score is only about $7 \times 10^3$ seconds. As on NIPS dataset, DLFRM$^h$ takes more time for training than DLFRM$^l$ and this phenomenon is more obvious here due to the scalability of the network. The reason is that DLFRM$^h$ often converges slower (see Fig 4(a)) with a larger latent dimension $K$ (see Fig. 4(c)). As discussed before, stoDLFRMs are more effective. When a full weight matrix $U$ is used, training time per iteration increases exponentially with respect to $K$. Therefore, diagDLFRMs are much more efficient due to the linear increase of training time per iteration with respect to $K$.

Overall, DLFRMs are stable and improve prediction performance efficiently .