Unsupervised Nonlinear Feature Selection from High-Dimensional Signed Networks

Qiang Huang,[1,3] **Tingyu Xia,**[2] **Huiyan Sun,**[2*] **Makoto Yamada,**[4,5*] **Yi Chang**[2,3]
[1]College of Computer Science and Technology, Jilin University, China
[2]School of Artificial Intelligence, Jilin University, China
[3]Key Laboratory of Symbolic Computation and Knowledge Engineering of Ministry of Education, China
[4]Kyoto University, Kyoto, Japan, [5]RIKEN AIP, Kyoto, Japan
huangqiang18@mails.jlu.edu.cn, xiatingyu1012@163.com,
sunhuiyan111@foxmail.com, myamada@i.kyoto-u.ac.jp, yichang@jlu.edu.cn

Abstract

With the rapid development of social media services in recent years, relational data are explosively growing. The signed network, which consists of a mixture of positive and negative links, is an effective way to represent the friendly and hostile relations among nodes, which can represent users or items. Because the features associated with a node of a signed network are usually incomplete, noisy, unlabeled, and high-dimensional, feature selection is an important procedure to eliminate irrelevant features. However, existing network-based feature selection methods are *linear* methods, which means they can only select features that having the linear dependency on the output values. Moreover, in many social data, most nodes are unlabeled; therefore, selecting features in an unsupervised manner is generally preferred. To this end, in this paper, we propose a *nonlinear* unsupervised feature selection method for signed networks, called SignedLasso. This method can select a small number of important features with nonlinear associations between inputs and output from a high-dimensional data. More specifically, we formulate unsupervised feature selection as a nonlinear feature selection problem with the Hilbert-Schmidt Independence Criterion Lasso (HSIC Lasso), which can find a small number of features in a nonlinear manner. Then, we propose the use of a deep learning-based node embedding to represent node similarity without label information and incorporate the node embedding into the HSIC Lasso. Through experiments on two real world datasets, we show that the proposed algorithm is superior to existing *linear* unsupervised feature selection methods.

1 Introduction

With the development of the increasingly prosperous online world, people are becoming involved in online social networks in a variety of ways everyday, which means social media data is growing explosively (Tang et al. 2016). Therefore, extracting knowledge from social media data has become an important and urgent part of data mining in recent years. For example, Twitter[1] generates 6000 twitters per second. High dimensional data brings great difficulty to the subsequent data mining tasks (Duda, Hart, and Stork 2012).

Users typically demonstrate their preferences on social media by posting comments, following someone on the Internet, and leaving positive/negative ratings, which creates positive or negative relationships between users and generates a social network. Moreover, each user has a set of attributes (e.g., age, sex, etc.). This data composes a signed network in which each node has an attribute vector. To understand the relationships among users, we must determine which user attributes (i.e., features) best represent the these relationships. Moreover, in many social data, most nodes are unlabeled, which means selecting features in unsupervised manner is generally preferred. Thus, the development of an unsupervised feature selection algorithm for signed network would have a significant impact on the field of data mining.

Feature selection is a widely studied type of machine learning algorithm and there exist many feature selection algorithms, including the Least Absolute Shrinkage and Selection Operator (Lasso) (Tibshirani 1996) and minimum redundancy maximum relevance (mRMR) (Peng, Long, and Ding 2005). These feature selection algorithms employs the independent and identically distributed (i.i.d) assumption; therefore, the performances of these methods can be poor for network data.

To handle feature selection from network data, network-based feature selection algorithms are useful. One of the most widely used approach incorporates the network information into a regularizer. For example, LinkedFS (Tang and Liu 2012) extracts various link information and evaluates the effects of user–user and user–post relationships in the linked data to find a relevant feature subset. Recently, a novel framework called SignedFS (Cheng, Li, and Liu 2017) was proposed to find the most relevant features in *signed* social networks by modeling user proximity and link information of both positive and negative links. However, existing approaches are all based on *linear* models.

To this end, in this paper, we propose a *nonlinear* unsupervised feature selection method for signed networks called SignedLasso, which can select a small number of important features with nonlinear association between inputs and output, from a high-dimensional data. More specifically, we for-

*Joint corresponding authors.

[1]https://twitter.com/

mulate unsupervised feature selection as a nonlinear feature selection problem with the Hilbert-Schmidt Independence Criterion Lasso (HSIC Lasso) (Yamada et al. 2014; 2018; Climente-González et al. 2019), which can find a small number of features in a nonlinear manner. Then, we propose the use of a deep learning-based node embedding to represent the node similarity without label information, and we incorporate the node embedding into the HSIC Lasso. Through experiments on two real world datasets, we show that the proposed algorithm outperforms existing *linear* unsupervised feature selection methods.

2 Related Work

In this section, we review feature selection algorithms.

2.1 Feature selection without network information

Feature selection is an effective approach to handle large-scale and high-dimensional data and can improve the efficiency of subsequent machine learning tasks, alleviate the curse of dimensionality, speed up the learning process, and improve the generalization capability of a model. Traditional feature selection algorithms can be divided into supervised algorithms (Ding and Peng 2005; Nie et al. 2010; Tibshirani 1996) and unsupervised algorithms (He, Cai, and Niyogi 2006; Li et al. 2012). Supervised algorithms include filter, wrapper, and embedded methods. Specifically, filter methods score each feature by evaluating its relevance with class labels, which is independent of the subsequent learning tasks. Wrapper methods directly take the performance of the final learning task as the criterion for evaluating the feature subset, but the cost of computation is much higher than that of filter methods. Finally, embedded methods accomplish feature selection in the training of model.

Recently, nonlinear feature selection algorithms have demonstrated their specific advantages by capturing the nonlinear dependency between the inputs and output, especially for high-dimensional data. *Minimum redundancy and maximum relevance* (mRMR) (Peng, Long, and Ding 2005) is the most widely used nonlinear feature selection algorithm; it can select features that are most related to the output and are mutually independent of each other. *Hibert-Schmidt feature selection* (HSFS) (Masaeli, Dy, and Fung 2010) is a HSIC-based greedy feature selection method that can select representative features but has very high computational cost for large and high-dimensional data because of its non-convexity. *Quadratic programming feature selection* (QPFS) (Rodriguez-Lujan et al. 2010) is the relaxed version of mRMR; it is convex and has a globally optional solution. *Sparse additive models* (SpAM) are convex and can be well optimized by back-fitting algorithms (Liu, Wasserman, and Lafferty 2009; Raskutti, Wainwright, and Yu 2012). The HSIC Lasso, which can also be considered as a convex variant of the mRMR algorithm, is a state-of-the-art nonlinear supervised feature selection algorithm.

Unsupervised methods have gained more attention recently due to the unavailability of labeled data. Most existing unsupervised algorithms apply some alternative methods, such as data similarity, local discriminative information, and reconstruction error, for feature selection. There exist a large number of unsupervised feature selection algorithms. *Laplacian Score* (LS) (He, Cai, and Niyogi 2006) is a widely used unsupervised feature selection algorithm, which is based on Laplacian Eigenmaps (Belkin and Niyogi 2002) and Locality Preserving Projection (He and Niyogi 2004). *Spectral Feature Selection* (SPEC) is a general framework of spectral feature selection for both supervised and unsupervised learning, which facilitates the joint study of supervised and unsupervised feature selection. *Multi-Cluster Feature Selection* (MCFS) (Cai, Zhang, and He 2010) uses multiple eigenvectors of the graph Laplacian, which is defined on the affinity matrix of data points, to capture the multi-cluster structure of the data and then find the important feature subset. *Nonnegative Discriminative Feature Selection* (NDFS) (Li et al. 2012) exploits the discriminative information and feature correlation simultaneously to select a better feature subset. In contrast, there are a small number of nonlinear unsupervised feature selection algorithms (Li et al. 2018) because they present an extremely difficult problem due to vast number of potential features.

2.2 Feature selection with network information

Traditional feature selection methods assume that the data are i.i.d. However, the data obtained from a social network tend to be non i.i.d, because social network data consist of links between users. *Unsupervised Streaming Feature Selection* (USFS) (Li et al. 2015) is a novel unsupervised streaming feature selection framework that exploits link information to conduct streaming feature selection. Guided by social theories (Tang et al. 2016), *Linked Unsupervised Feature Selection* (LUFS) (Tang and Liu 2014) is an unsupervised feature selection framework that builds a mathematical model based on the correlations among instances, and it uses pseudo-class labels to perform feature selection. *Linked Data Feature Selection* (LinkedFS) (Tang and Liu 2012) extracts various link information and evaluates the effects of user–user and user–post relationships in the linked data to find a relevant feature subset. By modeling link information, *Robust Unsupervised Feature Selection for Networked Data* (NetFS) (Li et al. 2016) embeds the latent representation learning of samples into feature selection for unsigned networks. Recent works have shown that negative links are also informative in many learning tasks. A novel feature selection framework for signed social networks (SignedFS) (Cheng, Li, and Liu 2017) was proposed to find most relevant features by modeling user proximity and link information of both positive and negative links. It has been reported that using network information helps significantly in selecting important features. However, these algorithms are all *linear*. In this paper, we propose a *nonlinear* feature selection algorithm with network information.

3 Problem Formulation

In this section, we describe the notations used in this paper and formally define the problem of nonlinear unsupervised feature selection from a signed network.

We use bold uppercase characters for matrices (e.g., $\boldsymbol{M}$), bold lowercase characters for vectors (e.g., $\boldsymbol{m}$), and normal characters for scalars (e.g., a). $\boldsymbol{m}_i$ and $\boldsymbol{m}^j$ represent the i-th row and j-th column of matrix $\boldsymbol{M}$, respectively, $\boldsymbol{M}_{ij}$ represents the i,j-th entry of $\boldsymbol{M}$, and $\boldsymbol{m}^{(i)}$ represents the i-th element of vector $\boldsymbol{m}$. We represent the transpose of $\boldsymbol{M}$ as $\boldsymbol{M}^\top$, and the Frobenius norm is defined as $\|\boldsymbol{M}\|_F = \sqrt{\sum_{i=1}^n \sum_{j=1}^d \boldsymbol{M}_{ij}^2}$. For a vector $\boldsymbol{m} \in \mathbb{R}^d$, the ℓ_1-norm is $\|\boldsymbol{m}\|_1 = \sum_{i=1}^d |\boldsymbol{m}^{(i)}|$, amd the ℓ_2-norm is $\|\boldsymbol{m}\|_2 = \sqrt{\sum_{i=1}^d \boldsymbol{m}^{(i)^2}}$.

Let $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ be a signed network, where $\mathcal{V} = \{v_1, v_2 \ldots, v_n\}$ is the set of n users and $\mathcal{E}$ is the set of positive and negative links. In particular, the value of any link e_{ij} could be 1, -1, or 0, which indicate a positive, negative, or no relation between v_i and v_j, respectively.

Let $\mathcal{X} \subset \mathbb{R}^d$ be the domain of vector $\boldsymbol{x}$. In each node i, we have an associated feature vector $\boldsymbol{x}_i$. $\boldsymbol{X} = (\boldsymbol{x}_1, \ldots \boldsymbol{x}_n)^\top = (\boldsymbol{f}_1, \ldots \boldsymbol{f}_d) \in \mathbb{R}^{n \times d}$ is the feature information matrix of n instances, where $\boldsymbol{f} \in \mathbb{R}^n$ is a feature vector.

The goal of unsupervised feature selection from a signed network is to select the m ($m \ll d$) most relevant features of $\boldsymbol{X}$ by exploiting the signed network $\mathcal{G} = (\mathcal{V}, \mathcal{E})$.

4 Proposed Method

In this section, we introduce a novel unsupervised feature selection method that utilizes positive and negative link information and captures nonlinear dependencies between features and latent representations for a high-dimensional signed social network.

4.1 Feature Selection by SignedLasso

We consider a feature-wise kernelized nonlinear method called the Hilbert-Schmidt Independence Criterion Lasso (HSIC Lasso) (Yamada et al. 2014) and extend it to an unsupervised scenario for a signed network, which we call SignedLasso. The HSIC Lasso is a supervised nonlinear feature selection method. Given supervised paired data $\{(\boldsymbol{x}_i, \boldsymbol{y}_i)\}_{i=1}^n$, the optimization problem of HSIC Lasso is given as

$$\min_{\boldsymbol{\beta} \in \mathbb{R}^d} \frac{1}{2}\|\bar{\boldsymbol{L}} - \sum_{k=1}^d \beta_k \bar{\boldsymbol{K}}^{(k)}\|_F^2 + \rho\|\boldsymbol{\beta}\|_1, \text{s.t. } \beta_1, \ldots, \beta_d \geq 0,$$

where $\bar{\boldsymbol{L}} = \boldsymbol{HLH}$ is the centered Gram matrix, $\boldsymbol{L}_{ij} = L(\boldsymbol{y}_i, \boldsymbol{y}_j)$ is the kernel for output, $\boldsymbol{H} = \boldsymbol{I}_n - \frac{1}{n}\boldsymbol{1}_n\boldsymbol{1}_n^\top$ is the centering matrix, $\boldsymbol{I}_n$ is the n-dimension identity matrix, $\boldsymbol{1}_n$ is the n-dimension vector whose elements are all 1, $\bar{\boldsymbol{K}}^{(k)} = \boldsymbol{HK}^{(k)}\boldsymbol{H}$ is the centered Gram matrix for the k-th feature, $\boldsymbol{K}_{ij}^{(k)} = K(\boldsymbol{x}_i^{(k)}, \boldsymbol{x}_j^{(k)})$ is the kernel for the k-th dimensional input, $\boldsymbol{\beta} = (\beta_1, \beta_2, \ldots \beta_d)^\top \in \mathbb{R}^d$ is a vector of regression coefficients, and ρ is the regularization parameter to control the sparsity. However, for unsupervised learning with a signed network setup, we cannot directly use the HSIC Lasso, since we do not have $\boldsymbol{y}$ information under unsupervised setups.

Thus, we propose an unsupervised variant of the HSIC Lasso, which utilizes the user latent representation matrix $\boldsymbol{U} = (\boldsymbol{u}_1, \boldsymbol{u}_2, \ldots, \boldsymbol{u}_n)^\top \in \mathbb{R}^{n \times c}$ computed from the signed network $\mathcal{G}$, where $\boldsymbol{u}_i \in \mathbb{R}^c$ and $\boldsymbol{u}_j \in \mathbb{R}^c$ are the corresponding low-dimensional vector representations of users i and j, respectively by an embedding method. Supposing we know the latent representation matrix $\boldsymbol{U}$, then the objective function of the SignedLasso is given as

$$\begin{aligned} \min_{\boldsymbol{\beta} \in \mathbb{R}^d} \quad & \frac{1}{2}\|\bar{\boldsymbol{M}} - \sum_{k=1}^d \beta_k \bar{\boldsymbol{K}}^{(k)}\|_F^2 + \rho\|\boldsymbol{\beta}\|_1 \\ \text{s.t.} \quad & \beta_1, \ldots, \beta_d \geq 0, \end{aligned} \tag{1}$$

where $\bar{\boldsymbol{M}} = \boldsymbol{HMH}$ is the centered kernel Gram matrix and $\boldsymbol{M}_{ij} = L(\boldsymbol{u}_i, \boldsymbol{u}_j)$.

In this paper, we use the Gaussian kernel for both input and the user latent representation vector:

$$K(\boldsymbol{x}_i^{(k)}, \boldsymbol{x}_j^{(k)}) = \exp\left(-\frac{(\boldsymbol{x}_i^{(k)} - \boldsymbol{x}_j^{(k)})^2}{2\sigma_x^2}\right),$$

$$L(\boldsymbol{u}_i, \boldsymbol{u}_j) = \exp\left(-\frac{\|\boldsymbol{u}_i - \boldsymbol{u}_j\|_2^2}{2\sigma_u^2}\right),$$

where σ_x and σ_u are the Gaussian kernel widths.

SignedLasso interpretation: As similar to the HSIC Lasso (Yamada et al. 2014), the SignedLasso contains the main concepts of mRMR (Peng, Long, and Ding 2005) which is a widely used classical supervised feature selection algorithm. We can rewrite the first term of Eq. (1) as

$$\begin{aligned} &\frac{1}{2}\|\bar{\boldsymbol{M}} - \sum_{k=1}^d \beta_k \bar{\boldsymbol{K}}^{(k)}\|_F^2 \\ &= \frac{1}{2}\sum_{k,m=1}^d \beta_k\beta_m \text{HSIC}(\boldsymbol{f}_k, \boldsymbol{f}_m) - \sum_{k=1}^d \beta_k \text{HSIC}(\boldsymbol{f}_k, \boldsymbol{U}) \\ &\quad + \frac{1}{2}\text{HSIC}(\boldsymbol{U}, \boldsymbol{U}), \end{aligned} \tag{2}$$

where $\text{HSIC}(\boldsymbol{f}_k, \boldsymbol{U}) = \text{tr}(\bar{\boldsymbol{K}}^{(k)}\bar{\boldsymbol{M}})$ is the empirical estimate of the Hibert-Schmidt independence criterion (HSIC) (Gretton et al. 2005), and $\text{tr}(\cdot)$ is the trace operator. HSIC based on a universal reproducing kernel, such as the Gaussian kernel, is a non-negative function that estimates the independence between two random variables. A larger HSIC value indicates more dependency between the two variables, and it is zero if and only if the two random variables are statistically independent.

In Eq. (2), we ignore the constant HSIC value and consider $\text{HSIC}(\boldsymbol{f}_k, \boldsymbol{U})$ and $\text{HSIC}(\boldsymbol{f}_k, \boldsymbol{f}_m)$. If there is strong dependency between the k-th feature vector $\boldsymbol{f}_k$ and user representation vector $\boldsymbol{U}$, the value of $\text{HSIC}(\boldsymbol{f}_k, \boldsymbol{U})$ should be large and the corresponding coefficient β_k should also take a large value in order to minimize Eq. (1). Meanwhile, if $\boldsymbol{f}_k$ is independent of $\boldsymbol{U}$, the value of $\text{HSIC}(\boldsymbol{f}_k, \boldsymbol{U})$ should be small so that β_k tends to be eliminated by ℓ_1-regularizer. This property can help select the most relevant features from the user latent representation.

For HSIC($\boldsymbol{f}_k, \boldsymbol{f}_m$), if $\boldsymbol{f}_k$ and $\boldsymbol{f}_m$ are strongly dependent (i.e., redundant features), the value of HSIC($\boldsymbol{f}_k, \boldsymbol{f}_m$) should be large and either of the two coefficients β_k and β_m tends to be zero to minimize the Eq. (1). This property means that redundant features will not be selected by SignedLasso.

Reducing the computational complexity of SignedLasso: The computational complexity of SingedLasso depends on the dimensions of input data n and d. If n is large, the cost of computing the input Gram matrix $\boldsymbol{M}$ will be very high. If d is large, the computation of Gram matrices $\boldsymbol{K}^{(k)}$ ($k = 1, 2 \ldots, d$) is also expensive.

Thus, we employ the block estimator for HSIC (Zhang et al. 2018). More specifically, we divide the n samples into n/B blocks, where B is the size of each block. The value of B is usually set to have a relatively small value i.e., 20 to 50 ($B \ll n$), and then the independence measurement HSIC can be rewritten as

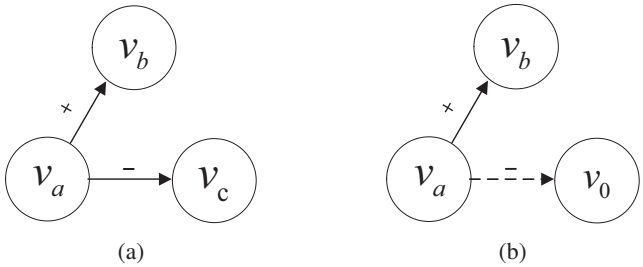
$$\text{HSIC}_b(\boldsymbol{f}_k, \boldsymbol{u}) = \frac{B}{n} \sum_{l=1}^{n/B} \text{HSIC}(\boldsymbol{f}_k^{(l)}, \boldsymbol{u}^{(l)}),$$

where $\boldsymbol{f}_k^{(l)} \in \mathbb{R}^B$ denotes the k-th feature vector of the l-th partition. The memory required to compute HSIC($\boldsymbol{f}_k^{(l)}, \boldsymbol{u}^{(l)}$) is $O(B^2)$, so the total memory required for n/B blocks is $O(nB)$. The computation of the original HSIC($\boldsymbol{f}_k, \boldsymbol{u}$) requires $O(n^2)$ memory, and $O(nB) \ll O(n^2)$ because $B \ll n$. Similarly, the computation of original HSIC($\boldsymbol{f}_k, \boldsymbol{f}_m$) can be reduced by block SignedLasso. Note that the block SignedLasso is an unsupervised variant of the block HSIC Lasso (Climente-González et al. 2019).

The (block) SignedLasso is a simple approach based on the HSIC Lasso. However, its performance heavily depends on the user latent representation matrix $\boldsymbol{U}$. In the next section, we introduce a deep learning-based embedding approach for estimating $\boldsymbol{U}$.

4.2 User Embedding by Deep Learning

In this section, we propose to use a deep learning-based embedding method to obtain the user latent representation matrix $\boldsymbol{U} = (\boldsymbol{u}_1, \boldsymbol{u}_2, \ldots, \boldsymbol{u}_n)^\top \in \mathbb{R}^{n \times c}$.



(a) (b)

Figure 1: Two types of training triplets in signed network.

We develop a set $\mathcal{P}$ of triplets (v_a, v_b, v_c), as shown in Figure 1(a), where user a has a positive link with user b, while user a has a negative link with user c from signed network $\mathcal{G}$. Considering the fact that there are many users whose 2-hop networks have only positive or negative links, we cannot contain such users in $\mathcal{P}$. Therefore, we add a virtual node v_0 to the signed network $\mathcal{G}$ and create a negative link between v_0 and each node whose 2-hop network has only positive links (as shown in Figure 1(b)) and develop a set $\mathcal{P}_0$ of triplets (v_a, v_b, v_0), where user a has a positive link with user b and a negative link with virtual node v_0. We do not consider the nodes whose 2-hop network has only negative links, there are two main reasons: Firstly, a node tends to be a spam user (a.k.a., internet trolls) if there exist many negative links from the node. Secondly, there are few such nodes because the cost of forming a negative link is much larger than positive one (Tang, Hu, and Liu 2014). Formally, $\mathcal{P}$ and $\mathcal{P}_0$ can be defined as:

$$\mathcal{P} = \{(v_a, v_b, v_c) | e_{ab} = 1, e_{ac} = -1, v_a, v_b, v_c \in \mathcal{V}\},$$
$$\mathcal{P}_0 = \{(v_a, v_b, v_0) | e_{ab} = 1, e_{a0} = -1, v_a, v_b, v_0 \in \mathcal{V}\}.$$

Then, we can model the similarity measurement of triplets in signed network $\mathcal{G}$:

$$f(\boldsymbol{u}_a, \boldsymbol{u}_b) \geq f(\boldsymbol{u}_a, \boldsymbol{u}_c) + \eta, \tag{3}$$
$$f(\boldsymbol{u}_a, \boldsymbol{u}_b) \geq f(\boldsymbol{u}_a, \boldsymbol{u}_0) + \eta_0, \tag{4}$$

where $\boldsymbol{u}_a$, $\boldsymbol{u}_b$, $\boldsymbol{u}_c$, $\boldsymbol{u}_0 \in \mathbb{R}^c$ are the low dimensional latent representation vectors of v_a, v_b, v_c, and v_0. $f(\boldsymbol{u}_a, \boldsymbol{u}_b)$ is a function to evaluate the similarity between v_a and v_b, and η and η_o are to control the difference between two similarities. For example, the larger the parameter η, the closer v_a and v_b are, and the farther away v_a and v_c are at the same time. The Eq. (3) and Eq. (4) are based on the extended balance theory (Cartwright and Harary 1956), The key idea of the theory is to assume "friends" are more important than "foes", and this is quite an intuitive setup in practice. That is, a vector should be embedded closer to their "friends" (or users with positive links) than their "foes" (or users with negative links). In terms of what is said above, the objective function (Wang et al. 2017) based on Eq. (3) and Eq. (4) for user latent representations of signed network is as follows:

$$\begin{aligned} \min_{\boldsymbol{U}, \boldsymbol{u}_0, \theta} \quad & \frac{1}{C}[\sum_{(\boldsymbol{u}_a, \boldsymbol{u}_b, \boldsymbol{u}_c) \in \mathcal{P}} \max(f(\boldsymbol{u}_a, \boldsymbol{u}_c) - f(\boldsymbol{u}_a, \boldsymbol{u}_b) + \eta, 0) \\ & + \sum_{(\boldsymbol{u}_a, \boldsymbol{u}_b, \boldsymbol{u}_0) \in \mathcal{P}_0} \max(f(\boldsymbol{u}_a, \boldsymbol{u}_0) - f(\boldsymbol{u}_a, \boldsymbol{u}_b) + \eta_0, 0)] \\ & + \gamma(\mathfrak{R}(\theta) + \|\boldsymbol{U}\|_F^2 + \|\boldsymbol{u}_0\|_2^2), \end{aligned} \tag{5}$$

where C is the size of training triplets in $\mathcal{P}$ and $\mathcal{P}_0$, c is the number of user latent factors, θ is a set of parameters in similarity measurement function f, $\mathfrak{R}(\theta)$ is a regularizer to reduce overfitting, and γ is the weight value to control the regularizer.

Now we discuss how to obtain a good similarity measurement function f. Recently, nonlinear methods have demonstrated their superiority to linear functions, such as matrix factorization, for representation learning. Among the nonlinear methods, deep learning is an overwhelmingly powerful way to obtain effective nonlinear latent representations. We use a deep framework called SiNE (Wang et al. 2017) to obtain user representations. The structure of the deep model with two networks and N hidden layers is as shown in Figure 2. The deep model shares the same parameters, where

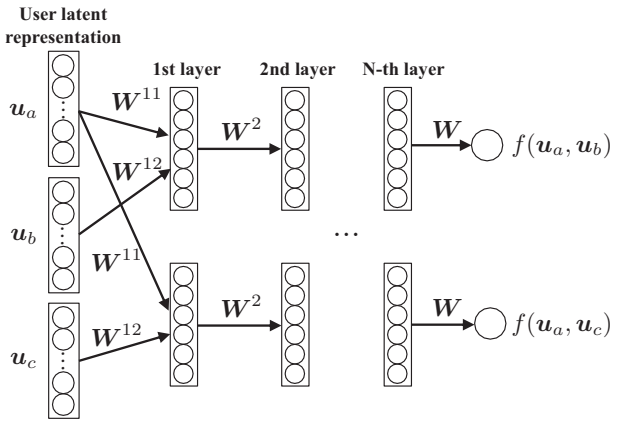


Figure 2: Structure of two deep networks sharing same parameters.

$\theta = \{\boldsymbol{W}^{11}, \boldsymbol{W}^{12}, \boldsymbol{W}^2, \ldots, \boldsymbol{W}^N, \boldsymbol{b}^1, \ldots, \boldsymbol{W}, b\}$ is the parameters set of the two deep networks. The input of deep networks is the set of triplets from $\mathcal{P}$ and $\mathcal{P}_0$. Let $\boldsymbol{o}^{ij}$ (j =1 or 2) be the output of the i-th hidden layer. The output of the first layer are

$$\boldsymbol{o}^{11} = \tanh(\boldsymbol{W}^{11}\boldsymbol{u}_a + \boldsymbol{W}^{12}\boldsymbol{u}_b + \boldsymbol{b}^1),$$
$$\boldsymbol{o}^{12} = \tanh(\boldsymbol{W}^{11}\boldsymbol{u}_a + \boldsymbol{W}^{12}\boldsymbol{u}_c + \boldsymbol{b}^1),$$

and the outputs of the deep network are two similarity measurements $f(\boldsymbol{u}_a, \boldsymbol{u}_b)$ and $f(\boldsymbol{u}_a, \mathrm{u}_c)$ for N hidden layers are

$$f(\boldsymbol{u}_a, \boldsymbol{u}_b) = \tanh(\boldsymbol{W}^\top \boldsymbol{o}^{N1} + b),$$
$$f(\boldsymbol{u}_a, \boldsymbol{u}_c) = \tanh(\boldsymbol{W}^\top \boldsymbol{o}^{N2} + b).$$

The objective function of the deep network model is Eq. (5), where $\mathfrak{R}(\theta)$ can be defined as follows:

$$\mathfrak{R}(\theta) = \|\boldsymbol{W}^{11}\|_F^2 + \|\boldsymbol{W}^{12}\|_F^2 + \|\boldsymbol{W}^2\|_F^2 + \ldots + \|\boldsymbol{W}^N\|_F^2 + \|\boldsymbol{W}\|_2^2 + \|\boldsymbol{b}^1\|_2^2 + \ldots + \|\boldsymbol{b}^N\|_2^2 + b^2.$$

Then, we employ back propagation to optimize the objective function of the deep network model (Eq. (5)), update the parameters, and obtain the user latent representation low-dimensional matrix $\boldsymbol{U}$.

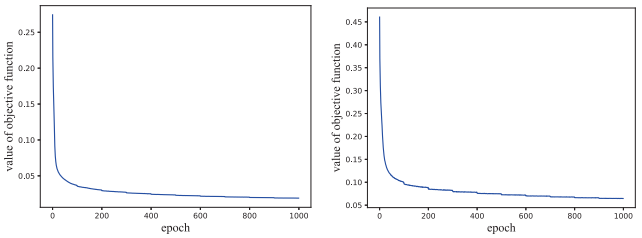
Once we obtain the user latent representation matrix, we select features by the SignedLasso.

5 Experiments

In this section, we report experiments on two real world signed network datasets that compared the framework proposed in this paper with five state-of-the-art unsupervised feature selection methods. We first describe the two real-world signed networks and baselines briefly. Then, we introduce settings of the experiments and present the comparison results between SignedLasso and the state-of-the-art unsupervised feature selection methods. Finally, we analyze the run time and memory cost of SignedLasso and two feature selection algorithms which are designed for network data. All codes are implemented by Python and we use Intel(R) Xeon(R) CPU E7- 4870 @ 2.40GHz and 64GB memory.

Table 1: Statistics of Epinions and Wiki-RfA.

Datasets	Epinions	Wiki-RfA
# of Classes	27	2
# of Users	5016	8044
# of Features	7010	8267
# of Positive links	280838	84425
# of Negative links	41246	25924



(a) Epinions. (b) Wiki-RfA.

Figure 3: Convergence on Epinions & Wiki-RfA.

5.1 Datasets

We use two real world signed network datasets: Epinions and Wiki-RfA.

Epinions[2]**:** Epinions is a product review website on which users can share their comments on different products. Users can establish positive or negative links with others. The signs of links can be 1, -1 or 0, which donate friend, foe, and no links relationship, respectively. On the website, each user can write comments or reviews for items, such as products. We take the words that appear in comments or reviews as features of users, their frequencies as feature values, and the category in which a user comments or reviews most as the ground truth of class labels.

Wiki-RfA[3]**:** Wiki-RfA is a signed network dataset concerning the election of editors on Wikipedia. Anyone can be a candidate for the chief editor or vote for somebody else. In the network, nodes represent the members and the edges represent votes. Links are treated the same for this dataset as they are for the Epinions dataset. Every vote is accompanied with a short comment that can be modeled as a user feature. Similarly, each feature value is the frequency that a word appears in a short comment. Whether a user is rejected or accepted is regarded as the ground truth of class labels.

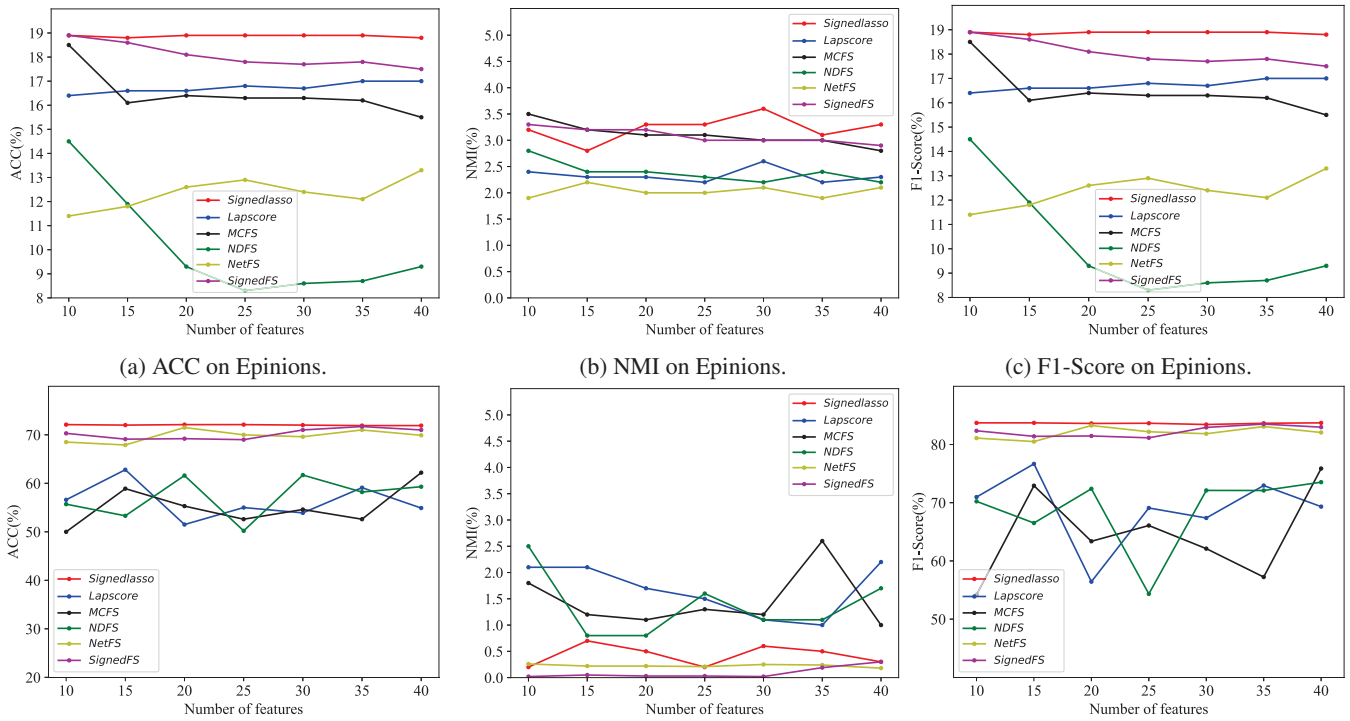
More statistical details of the Epinions and Wiki-RfA data are given on Table 1.

5.2 Baselines

We compared SignedLasso with five state-of-the-art unsupervised feature selection methods, including traditional algorithms and methods for social networks.

- Laplacian Score (He, Cai, and Niyogi 2006).

[2]http://www.cse.msu.edu/ tangjili/trust.html
[3]http://snap.stanford.edu/data/wiki-RfA.html



(a) ACC on Epinions. (b) NMI on Epinions. (c) F1-Score on Epinions.

(d) ACC on Wiki-RfA. (e) NMI on Wiki-RfA. (f) F1-Score on Wiki-RfA.

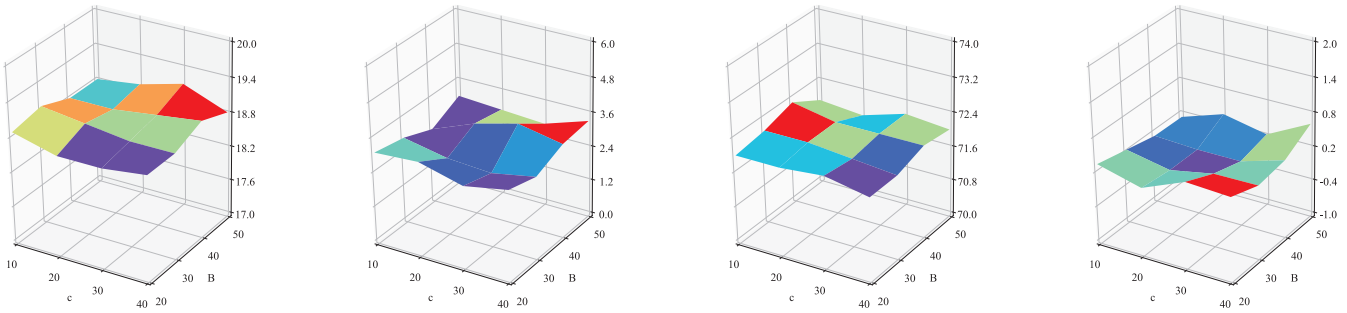
Figure 4: The results of experiments for two real signed networks, the horizontal axis denotes the number of selected features, and the vertical axis denotes the mean value of evaluation metrics: (a) accuracy on Epinions, (b) NMI on Epinions, (c) F1-Score on Epinion, (d) accuracy on Wiki-RfA, (e) NMI on Wiki-RfA, (f) F1-Score on Wiki-RfA.

- MultiCluster Feature Selection (MCFS) (Cai, Zhang, and He 2010).
- Nonnegative Discriminative Feature Selection (NDFS) (Li et al. 2012).
- Robust Unsupervised Feature Selection for Networked Data (NetFS) (Li et al. 2016).
- Feature Selection Framework in Signed Social Network (SignedFS).(Cheng, Li, and Liu 2017)

Of the five feature selection technologies described above, Laplacian Score, MCFS, and NDFS are traditional unsupervised feature selection methods, which only consider data matrix X without extra information, NetFS is an unsupervised feature selection method that utilizes positive links among users in an unsigned network. SignedFS is an unsupervised method that considers both positive and negative links simultaneously in a signed network. For a fair comparison, we use the best cluster performance of all feature selection methods in different parameters by a grid search strategy.

For the framework proposed in this paper, we set the block size as $B = 50$, the dimension of latent representation vector as $c = 10$, the controllers of similarity as $\eta = 1$ and $\eta_0 = 1$, the number of hidden layers as $N = 3$, and select the top 10, 15, 20, 25, 30, 35 and 40 features, respectively. The results of various feature selection methods on Wiki-RfA and Epinions are shown in Figure 4. By analyzing the results of different methods, we determined the following results:

- SignedLasso clearly outperforms the traditional feature selection methods of Laplacian Score, MCFS, and NDFS, which ignore the link and structure information in most cases of different numbers of selected features. This shows that feature selection methods considering link and structure information are better than traditional methods based on the i.i.d assumption.
- SignedLasso and SignedFS have better clustering performance than NetFS. The most important reason for this is that NetFS only considers the positive link information, while SignedLasso and SignedFS utilize both positive and negative links between users. This observation demonstrates that negative links have added value over positive links for signed networks.
- SignedLasso also outperforms SignedFS, which is a linear feature selection algorithm for signed networks. This is because SignedFS is a linear feature selection model that can only capture linear dependencies between features and outputs, while SignedLasso considers a feature-wise kernelized Lasso to capture nonlinear dependency.
- The proposed method is more stable than other feature selection methods as its performance is steady as the number of selected features changes. The performance of other methods, such as NDFS or MCFS, degrades as the number of selected features increases because of noise in



(a) ACC (%) on Epinions. (b) NMI (%) on Epinions. (c) ACC (%) on Wiki-RfA. (d) NMI (%) on Wiki-RfA.

Figure 5: Impact of B and c on SingnedLasso.

the social network data. This means that SignedLasso is not sensitive to noise and has stronger stabilization.

5.3 Computing and Parameter Analysis

As mentioned before, we use the block SignedLasso to guide the unsupervised feature selection on a high-dimensional signed social network. In this subsection, we compare the run time and memory cost between block SignedLasso and the other two feature selection algorithms NetFS and SignedFS. We omit Laplacian Score, MCFS and NDFS from the analysis because they are not comparable to those methods for network data.

The main time consumption of SignedLasso is in the process of user latent embedding and feature selection. Therefore, we first investigate the size of epoch to converge for the objective function of embedding of SignedLasso. Figure 3 shows the value of objective function in each epoch for Epinions and Wiki-RfA, respectively. We can see that the value of the objective function decreases quickly in the first 200 epochs and then converges. Therefore we set epoch = 200 to calculate the time consumption of SignedLasso.

We next compare the run time and peak memory usage of SignedLasso with those of NetFS and SignedFS. We set the number of training steps for NetFS and SignedFS as 100 and 200, respectively. Tables 2 and 3 shows the run times and memory peaks of the proposed method and the other two feature selection methods on Epinions and Wiki-RfA. In Tables 2 and 3, we can see that SignedLasso has much less run time than NetFS and SignedFS. The models of SignedFS and NetFS are based on complex matrix factorization, so their computations are very expensive and convergence is so slow when the dimension of data is high. However, the proposed method, which utilizes blocking, converges an order of magnitude faster than SignedFS and NetFS, but the required memory is just several times larger that of the other two methods.

Secondly, we investigate the impacts of parameters block size B and embedding dimension c which are critical parts of SignedLasso. More specifically, we set the number of selected features as 20 and vary the values of B as $\{20, 30, 40, 50\}$ and c as $\{10, 20, 30, 40\}$, while we keep the rest of parameter values the same. The results in terms of ACC and NMI on Epinions and Wiki-RfA under different combinations of B and c are shown in Figure 5. As can be seen, SignedLasso is not sensitive to B and c.

Table 2: Run time and Peak memory on Epinions.

Methods	NetFS	SignedFS	SignedLasso
Time(s)	36704	7428	1837
Memory peak(MiB)	2828	3249	6487

Table 3: Run time and Peak memory on Wiki-RfA.

Methods	NetFS	SignedFS	SignedLasso
Time(s)	68256	16860	636
Memory peak(MiB)	4933	8821	11350

6 Conclusion and Future Work

In this paper, we proposed a novel unsupervised nonlinear feature selection framework called SignedLasso to find the most relevant and least redundant feature subset for a signed network. First, combing social theories and the advantage of deep learning, we embedded the latent representations of users by modeling two types of training sets consisting of positive and negative links. Then, we replaced the label information with the latent representations and applied HSIC Lasso for effective feature selection by capturing the nonlinear dependencies between features and outputs. It is worthy mentioning that the number of selected features is much less than that for traditional methods. Besides, we use a block policy to reduce the cost of computation so that we can extend the method to larger scale and higher dimensional data. We implemented SignedLasso on two real world signed networks, and the experimental results show that the proposed method is promising.

In future work, we can focus on two aspects. First, we would like to adapt SignedLasso to other signed networks such as gene networks to capture nonlinear dependency based on their unique characteristics and then find the most representative feature subset. Second, we would like to investigate how to apply the proposed method to more signed network scenarios to lower data dimension greatly so that subsequent tasks can be more effective.

Acknowledgements

This work was supported in part by the National Natural Science Foundation of China (No.61976102). M.Y. was supported by the JST PRESTO program JPMJPR165A and partly supported by MEXT KAKENHI 16H06299 and the RIKEN engineering network fund.

References

Belkin, M., and Niyogi, P. 2002. Laplacian eigenmaps and spectral techniques for embedding and clustering. In *NIPS*.

Cai, D.; Zhang, C.; and He, X. 2010. Unsupervised feature selection for multi-cluster data. In *KDD*.

Cartwright, D., and Harary, F. 1956. Structural balance: a generalization of heider's theory. *Psychological review* 63(5):277.

Cheng, K.; Li, J.; and Liu, H. 2017. Unsupervised feature selection in signed social networks. In *KDD*.

Climente-González, H.; Azencott, C.-A.; Kaski, S.; and Yamada, M. 2019. Block HSIC Lasso: model-free biomarker detection for ultra-high dimensional data. *Bioinformatics* 35(14):i427–i435.

Ding, C., and Peng, H. 2005. Minimum redundancy feature selection from microarray gene expression data. *Journal of Bioinformatics and Computational Biology* 3(02):185–205.

Duda, R. O.; Hart, P. E.; and Stork, D. G. 2012. *Pattern Classification*. John Wiley & Sons.

Gretton, A.; Bousquet, O.; Smola, A.; and Schölkopf, B. 2005. Measuring statistical dependence with hilbert-schmidt norms. In *ALT*.

He, X., and Niyogi, P. 2004. Locality preserving projections. In *NIPS*.

He, X.; Cai, D.; and Niyogi, P. 2006. Laplacian score for feature selection. In *NIPS*.

Li, Z.; Yang, Y.; Liu, J.; Zhou, X.; and Lu, H. 2012. Unsupervised feature selection using nonnegative spectral analysis. In *AAAI*.

Li, J.; Hu, X.; Tang, J.; and Liu, H. 2015. Unsupervised streaming feature selection in social media. In *CIKM*.

Li, J.; Hu, X.; Wu, L.; and Liu, H. 2016. Robust unsupervised feature selection on networked data. In *SDM*.

Li, J.; Zhang, S.; Zhang, L.; Lei, C.; and Zhang, J. 2018. Unsupervised nonlinear feature selection algorithm via kernel function. *Neural Computing and Applications* 1–12.

Liu, H.; Wasserman, L.; and Lafferty, J. D. 2009. Nonparametric regression and classification with joint sparsity constraints. In *NIPS*.

Masaeli, M.; Dy, J. G.; and Fung, G. M. 2010. From transformation-based dimensionality reduction to feature selection. In *ICML*.

Nie, F.; Huang, H.; Cai, X.; and Ding, C. H. 2010. Efficient and robust feature selection via joint l2, 1-norms minimization. In *NIPS*.

Peng, H.; Long, F.; and Ding, C. 2005. Feature selection based on mutual information: criteria of max-dependency, max-relevance, and min-redundancy. *IEEE Transactions on Pattern Analysis & Machine Intelligence* (8):1226–1238.

Raskutti, G.; Wainwright, M. J.; and Yu, B. 2012. Minimax-optimal rates for sparse additive models over kernel classes via convex programming. *Journal of Machine Learning Research* 13(Feb):389–427.

Rodriguez-Lujan, I.; Huerta, R.; Elkan, C.; and Cruz, C. S. 2010. Quadratic programming feature selection. *Journal of Machine Learning Research* 11(Apr):1491–1516.

Tang, J., and Liu, H. 2012. Feature selection with linked data in social media. In *SDM*.

Tang, J., and Liu, H. 2014. An unsupervised feature selection framework for social media data. *IEEE Transactions on Knowledge and Data Engineering* 26(12):2914–2927.

Tang, J.; Chang, Y.; Aggarwal, C.; and Liu, H. 2016. A survey of signed network mining in social media. *ACM Computing Surveys (CSUR)* 49(3):42.

Tang, J.; Hu, X.; and Liu, H. 2014. Is distrust the negation of trust?: the value of distrust in social media. In *HT*.

Tibshirani, R. 1996. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B (Methodological)* 58(1):267–288.

Wang, S.; Tang, J.; Aggarwal, C.; Chang, Y.; and Liu, H. 2017. Signed network embedding in social media. In *SDM*.

Yamada, M.; Jitkrittum, W.; Sigal, L.; Xing, E. P.; and Sugiyama, M. 2014. High-dimensional feature selection by feature-wise kernelized lasso. *Neural Computation* 26(1):185–207.

Yamada, M.; Tang, J.; Lugo-Martinez, J.; Hodzic, E.; Shrestha, R.; Saha, A.; Ouyang, H.; Yin, D.; Mamitsuka, H.; Sahinalp, C.; et al. 2018. Ultra high-dimensional nonlinear feature selection for big biological data. *IEEE Transactions on Knowledge and Data Engineering* 30(7):1352–1365.

Zhang, Q.; Filippi, S.; Gretton, A.; and Sejdinovic, D. 2018. Large-scale kernel methods for independence testing. *Statistics and Computing* 28(1):113–130.