Unsupervised Personalized Feature Selection

Jundong Li, Liang Wu, Harsh Dani, Huan Liu
Computer Science and Engineering, Arizona State University, USA
{jundong.li,wuliang,hdani,huan.liu}@asu.edu

Abstract

Feature selection is effective in preparing high-dimensional data for a variety of learning tasks such as classification, clustering and anomaly detection. A vast majority of existing feature selection methods assume that all instances share some common patterns manifested in a subset of shared features. However, this assumption is not necessarily true in many domains where data instances could show high individuality. For example, in the medical domain, we need to capture the heterogeneous nature of patients for personalized predictive modeling, which could be characterized by a subset of instance-specific features. Motivated by this, we propose to study a novel problem of personalized feature selection. In particular, we investigate the problem in an unsupervised scenario as label information is usually hard to obtain in practice. To be specific, we present a novel unsupervised personalized feature selection framework UPFS to find some shared features by all instances and instance-specific features tailored to each instance. We formulate the problem into a principled optimization framework and provide an effective algorithm to solve it. Experimental results on real-world datasets verify the effectiveness of the proposed UPFS framework.

Introduction

Recent years have witnessed huge amounts of high-dimensional data in many data mining, machine learning, computer vision and natural language processing applications. High-dimensional data not only demands more on the computational and storage requirements, but also degenerates many learning algorithms due to the *curse of dimensionality* (Friedman, Hastie, and Tibshirani 2001; Guyon and Elisseeff 2003; Liu and Motoda 2007; Li et al. 2016; Li and Liu 2017). Feature selection is one of the most effective data preprocessing strategies to deal with these high-dimensional data. It directly selects a subset of relevant features from the original high-dimensional feature space for a compact and accurate representation. Feature selection helps build simpler and more comprehensive learning models, improve learning performance, and prepare clean and understandable data.

According to the availability of label information, feature selection algorithms can be divided into supervised methods and unsupervised methods. Supervised methods are mainly designed for classification or regression problems, they attempt to find relevant features that can discriminate instances from different classes. On the other hand, without label information to guide the selection process, unsupervised methods employ alternative criteria to assess the importance of features such as data similarity (He, Cai, and Niyogi 2005; Cai, Zhang, and He 2010), local discriminative information (Yang et al. 2011; Li et al. 2012) or data reconstruction error (Farahat, Ghodsi, and Kamel 2011; Li, Tang, and Liu 2017). As most real-world data is unlabeled and label information is time consuming and labor intensive to obtain, unsupervised feature selection is more appealing in practical usage. Recently, sparse learning based feature selection methods (Tibshirani 1996; Liu, Ji, and Ye 2009; Cai, Zhang, and He 2010; Nie et al. 2010) received increasing attention as they can embed feature selection into the model construction phase, which often gives good learning performance and model interpretability. Through a ℓ_1-norm or a $\ell_{2,1}$-norm sparse regularization term, feature sparsity is achieved for all instances. The features with small or zero weights then can be directly eliminated as they have limited contribution to the model construction.

Most of the sparse learning based feature selection methods proposed so far, overwhelmingly build a single global model (i.e., feature weight) for all data instances. Despite its empirical success in terms of high prediction accuracy (either classification or clustering), this kind of global models inevitably ignore the individuality or personality of each individual data instance. In many cases, instances could be highly idiosyncratic. For example, posting behaviors of users in social media sites could differ remarkably. Based on their personal characters and interests, the words and phrases they frequently use are rather diverse, with different social foci. In this regard, the key challenge centers around if we can tailor the feature selection for each instance such that important features for different instances can be different. Although it is important to find personalized features, different instances more or less have some commonality. For example, in medical predictive modeling, despite the fact that health conditions of patients could vary a lot, they may share a certain amount of common symptoms for a specific disease. Hence, it is also of vital importance to leverage these common patterns for learning by finding some shared features across all data instances.

Motivated by the above observations, we propose to perform personalized feature selection for each instance in an unsupervised fashion. To be specific, we would like to customize the feature selection process for each instance in finding a subset of shared features and certain instance-specific features. An illustration of the proposed unsupervised personalized feature selection is shown in Figure 1. In essence, we study (1) how to mathematically model the common patterns of all instances and personalized patterns of each specific data instance for feature selection. (2) how to find both shared features and instance-specific features when label information is not available. To answer these two research questions, we propose an unsupervised personalized feature selection framework UPFS. The major contributions of this work is summarized as follows:

- We formally define the problem of unsupervised personalized feature selection.
- We propose a principled way to capture common and individualized patterns by finding (1) shared features; and (2) discriminative features customized for each instance.
- We present an effective alternating algorithm to solve the optimization problem of the proposed UPFS framework.
- We validate the effectiveness of the UPFS framework on real-world datasets of different types.

Problem Statement

We first summarize the notations used in this work. We use bold uppercase letters for matrices (e.g., $\mathbf{A}$), bold lowercase letters for vectors (e.g., $\mathbf{a}$), normal lowercase letters for scalars (e.g., a). Also, we represent the i-th row of the matrix $\mathbf{A} \in \mathbb{R}^{n \times d}$ as $\mathbf{A}_{i*}$, the j-th column as $\mathbf{A}_{*j}$, the (i,j)-th entry as $\mathbf{A}_{ij}$. We denote the transpose of the matrix $\mathbf{A}$ as $\mathbf{A}^T$, the trace of $\mathbf{A}$ as $tr(\mathbf{A})$ if it is a square matrix. The Frobenius norm of matrix $\mathbf{A}$ is defined as $||\mathbf{A}||_F$, and its $\ell_{2,1}$-norm is defined as $\|\mathbf{A}\|_{2,1}$. More specifically, $\|\mathbf{A}\|_F = \sqrt{\sum_{i=1}^n \sum_{j=1}^d \mathbf{A}_{ij}^2}$ and $||\mathbf{A}||_{2,1} = \sum_{i=1}^n \sqrt{\sum_{j=1}^d \mathbf{A}_{ij}^2}$. $\mathbf{A} \otimes \mathbf{B}$ denotes the Kronecker product between matrices $\mathbf{A}$ and $\mathbf{B}$. The identity matrix of size d-by-d is denoted by $\mathbf{I}_d$ and $\mathbf{1}$ is a vector whose elements are all 1.

With the above mentioned notations, the problem of *unsupervised personalized feature selection* can be formally defined as follows. Given a dataset $\mathbf{X} \in \mathbb{R}^{n \times d}$ with n instances and d features, the task is to tailor the unsupervised feature selection phase for each instance by finding (1) a subset of discriminative features specific for each instance $\mathbf{x}_i$; (2) some shared features for all instances $\{\mathbf{x}_1, ..., \mathbf{x}_n\}$.

Unsupervised Personalized Feature Selection Framework - UPFS

In this section, we present the proposed unsupervised personalized feature selection framework - UPFS in detail. Let $\mathbf{X}$ be the unlabeled dataset where each instance $\mathbf{x}_i \in \mathbb{R}^d$ is in a d-dimensional feature space (d could be very large). To tackle the challenges resulted from the lack of labels in guiding feature selection, we introduce the concept of pseudo labels. More specifically, we assume that these n data instances are sampled from c different classes. Let $\mathbf{F} \in \{0,1\}^{n \times c}$ denote the one-hot class matrix such that $\mathbf{F}_{ij} = 1$ if $\mathbf{x}_i$ is assigned with the class j (we assume that each instance only belongs to one class), otherwise $\mathbf{F}_{ij} = 0$.

With the definition of pseudo labels, we can perform feature selection to find some shared features that can discriminate instances from different classes. One way to achieve this target is to build a least squares classification model with a sparse regularization term:

$$\min_{\mathbf{W}} \sum_{i=1}^{n} \|\mathbf{x}_i \mathbf{W} - \mathbf{F}_{i*}\|_2^2 + \alpha \|\mathbf{W}\|_{2,1}, \tag{1}$$

where $\mathbf{W} \in \mathbb{R}^{d \times c}$ is the global feature weight and α is to control the sparsity of the global feature weight $\mathbf{W}$. We impose a $\ell_{2,1}$-norm penalty on $\mathbf{W}$ to achieve joint feature sparsity across c different classes.

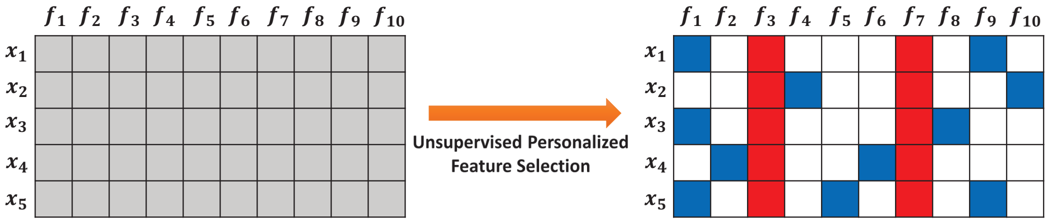
Above formulation assumes that feature weights are consistent across all data instances. However, as mentioned previously, in many cases, feature importance for different data instances could vary remarkably. Hence, it would be more appealing to tailor the feature selection for each instance in finding a subset of instance-specific features. To this end, we propose to use a conjunction of the global feature weight and a localized feature weight to perform the pseudo label prediction for each instance, resulting in the following formulation:

$$\min_{\mathbf{W}, \mathbf{U}^i} \sum_{i=1}^{n} \|\mathbf{x}_i (\mathbf{W} + \mathbf{U}^i) - \mathbf{F}_{i*}\|_2^2 + \alpha \|\mathbf{W}\|_{2,1}, \tag{2}$$

where $\mathbf{U}^i \in \mathbb{R}^{d \times c}$ is a localized feature weight for instance $\mathbf{x}_i$. In order to find a subset of discriminative features for each instance, we also would like to achieve feature sparsity for the localized feature weight $\mathbf{U}^i$. In particular, the joint feature sparsity across c pseudo class labels is desired. For that purpose, we formulate the problem as an exclusive group lasso problem (Kong et al. 2014; 2016). Specifically, we regard each localized feature weight $\mathbf{U}^i$ as a group. As we attempt to find discriminative features customized for each instance, we incentivize competition within each group but discourage competition between groups. In this way, no groups will dominate others and it enables us to find discriminative features for each instance. Mathematically, we first impose a $\ell_{2,1}$-norm penalty on each $\mathbf{U}^i$ for joint feature sparsity across c pseudo labels. Afterwards, we introduce a ℓ_2-norm at the inter-group level for non-sparsity. The objective function now is:

$$\min_{\mathbf{W}} \sum_{i=1}^{n} \|\mathbf{x}_i (\mathbf{W} + \mathbf{U}^i) - \mathbf{F}_{i*}\|_2^2 + \alpha (\|\mathbf{W}\|_{2,1} + \sum_{i=1}^{n} \|\mathbf{U}^i\|_{2,1}^2). \tag{3}$$

The above formulation enables us to perform unsupervised personalized feature selection to obtain a number of shared features across all instances and instance-specific discriminative features. However, building a personalized model for each instance is computationally expensive. In addition, we could only leverage one instance $\mathbf{x}_i$ to train the localized feature weight $\mathbf{U}^i$, thus the model learning process



f_1 f_2 f_3 f_4 f_5 f_6 f_7 f_8 f_9 f_{10}
x_1
x_2
x_3
x_4
x_5
Unsupervised Personalized Feature Selection
f_1 f_2 f_3 f_4 f_5 f_6 f_7 f_8 f_9 f_{10}
x_1
x_2
x_3
x_4
x_5

Figure 1: Illustration of the unsupervised personalized feature selection. Red cells indicate the shared features by all instances while the blue cells represent the instance-specific features. White columns are the unselected features.

can easily overfit with poor generalization ability. To alleviate this critical issue, we force instance to borrow strength from its neighbors to learn the localized feature weight. In particular, we first build a nearest neighbor affinity graph of input data instances to realize the local geometry structure. The nearest neighbor affinity graph $\mathbf{S} \in \mathbb{R}^{n \times n}$ is created as follows:

$$\mathbf{S}_{ij} = \begin{cases} \exp(-\frac{\|\mathbf{x}_i - \mathbf{x}_j\|_2^2}{\sigma^2}) & \text{if } \mathbf{x}_i \in \mathcal{N}_k(\mathbf{x}_j) \text{ or } \mathbf{x}_j \in \mathcal{N}_k(\mathbf{x}_i) \\ 0 & \text{otherwise,} \end{cases} \tag{4}$$

where $\mathcal{N}_p(\mathbf{x}_i)$ is the set of k-nearest neighbors of $\mathbf{x}_i$, σ is a predefined parameter. Above formulation indicates that if $\mathbf{x}_i$ or $\mathbf{x}_j$ is among the k-nearest neighbors of the other, their similarity can be obtained by the RBF kernel, otherwise their similarity is 0. With this, we force connected data instances borrow strength from each other in learning the localized feature weight by the network lasso penalty (Hallac, Leskovec, and Boyd 2015):

$$\min_{\mathbf{U}^i(i=1,\ldots,n)} \sum_{i,j=1}^{n} \mathbf{S}_{ij} \|\mathbf{U}^i - \mathbf{U}^j\|_F. \tag{5}$$

The above network lasso penalty is similar but different from the graph Laplacian. Without the square on the difference between $\mathbf{U}^i$ and $\mathbf{U}^j$, Eq. (5) forces $\mathbf{U}^i$ to be the same as $\mathbf{U}^j$ if they have high similarity. Hence, it can greatly reduce the number of model parameters for the personalized feature selection and also mitigate the overfitting problem.

Furthermore, according to the spectral theory (Ng et al. 2001; Von Luxburg 2007), a rational choice of pseudo class labels should preserve the local geometry structure of data such that instances that are close to each other in the original feature space should also have the same pseudo class labels. Since the data local geometry structure has been modeled by the affinity graph defined in Eq. (4), we make the pseudo class labels $\mathbf{F}$ be smooth over the affinity graph $\mathbf{S}$, resulting in the following term:

$$\min_{\mathbf{F}} \frac{1}{2} \sum_{i,j=1}^{n} \mathbf{S}_{ij} \| \frac{\mathbf{F}_{i*}}{\sqrt{\mathbf{D}_{ii}}} - \frac{\mathbf{F}_{j*}}{\sqrt{\mathbf{D}_{jj}}} \|_2^2 = tr(\mathbf{F}^T \mathbf{L} \mathbf{F}), \tag{6}$$

where $\mathbf{D}$ is a diagonal matrix with $\mathbf{D}_{ii} = \sum_{j=1}^{n} \mathbf{S}_{ij}$. The normalized Laplacian matrix $\mathbf{L} = \mathbf{D}^{-\frac{1}{2}}(\mathbf{D} - \mathbf{S})\mathbf{D}^{-\frac{1}{2}}$.

By combining the network lasso term in Eq. (5) and the spectral analysis in Eq. (6), the final objective function of the unsupervised personalized feature selection is as follows:

$$\begin{gathered} \min_{\mathbf{W},\mathbf{U},\mathbf{F}} \sum_{i=1}^{n} \|\mathbf{x}_i(\mathbf{W} + \mathbf{U}^i) - \mathbf{F}_{i*}\|_2^2 + \alpha(\|\mathbf{W}\|_{2,1} + \sum_{i=1}^{n} \|\mathbf{U}^i\|_{2,1}^2) \\ +\beta \sum_{i,j=1}^{n} \mathbf{S}_{ij} \|\mathbf{U}^i - \mathbf{U}^j\|_F + \gamma\, tr(\mathbf{F}^T \mathbf{L} \mathbf{F}) \\ \text{s.t. } \mathbf{F} \in \{0,1\}^{n \times c},\ \mathbf{F}^T \mathbf{1} = \mathbf{1}, \end{gathered} \tag{7}$$

where $\mathbf{U} = [\mathbf{U}^1; \ldots; \mathbf{U}^n]$ is the concatenation of all localized feature weights. β and γ are two regularization parameters. In detail, β controls to what extent instances can borrow strength from neighbors in learning the localized feature weight; and γ controls how well the pseudo labels preserve the local geometry structure of the data. The above objective function is difficult to solve due to the discrete constraint, rendering it as an integer programming problem (Nemhauser 1998). Motivated by (Von Luxburg 2007), we relax the constraints with an orthogonal condition:

$$\begin{gathered} \min_{\mathbf{W},\mathbf{U},\mathbf{F}} \sum_{i=1}^{n} \|\mathbf{x}_i(\mathbf{W} + \mathbf{U}^i) - \mathbf{F}_{i*}\|_2^2 + \alpha(\|\mathbf{W}\|_{2,1} + \sum_{i=1}^{n} \|\mathbf{U}^i\|_{2,1}^2) \\ +\beta \sum_{i,j=1}^{n} \mathbf{S}_{ij} \|\mathbf{U}^i - \mathbf{U}^j\|_F + \gamma\, tr(\mathbf{F}^T \mathbf{L} \mathbf{F}) \\ \text{s.t. } \mathbf{F}^T \mathbf{F} = \mathbf{I}_c,\ \mathbf{F} \geq 0. \end{gathered} \tag{8}$$

We can further rewrite the above objective function as:

$$\begin{gathered} \min_{\mathbf{W},\mathbf{U},\mathbf{F}} \sum_{i=1}^{n} \|\mathbf{x}_i(\mathbf{W} + \mathbf{U}^i) - \mathbf{F}_{i*}\|_2^2 + \alpha(\|\mathbf{W}\|_{2,1} + \sum_{i=1}^{n} \|\mathbf{U}^i\|_{2,1}^2) \\ +\beta \sum_{i,j=1}^{n} \mathbf{S}_{ij} \|\mathbf{U}^i - \mathbf{U}^j\|_F + \gamma\, tr(\mathbf{F}^T \mathbf{L} \mathbf{F}) + \frac{\theta}{2} \|\mathbf{F}^T \mathbf{F} - \mathbf{I}_c\|_F^2 \\ \text{s.t. } \mathbf{F} \geq 0. \end{gathered} \tag{9}$$

Here we introduce another parameter θ to ensure that the orthogonal condition is satisfied. Normally, we set it as a constant large number (e.g., 10^8) to ensure that the orthogonal condition is satisfied.

After we solve the above objective function to obtain the model parameters $\mathbf{W}$, $\mathbf{U}$ and $\mathbf{F}$, we can perform the pseudo class label prediction. For each instance $\mathbf{x}_i$, the classifier is a conjunction of the global feature weight $\mathbf{W}$ and the localized feature weight $\mathbf{U}^i$. In particular, for each data instance

$\mathbf{x}_i$, we define feature score of the j-th feature as $\|\mathbf{K}_{j*}\|_2^2$, where $\mathbf{K} = \mathbf{W} + \mathbf{U}^i$. After computing all feature scores, we rank them in a descending order and return the top m ranked features where m is the number of selected features we want to select.

Optimization Algorithm for UPFS

The optimization problem of the proposed UPFS framework in Eq. (9) is not convex w.r.t. three variables $\mathbf{W}$, $\mathbf{U}$ and $\mathbf{F}$ simultaneously. Fortunately, it is a convex optimization problem if we fix two model parameters and update the other one. Therefore, we propose to solve the optimization problem of UPFS by an alternating optimization algorithm until the objective function in Eq. (9) converges. The detailed model update procedures are presented as follows.

Update Global Feature Weight W

First, we attempt to update the global feature $\mathbf{W}$ when the other two model parameters $\mathbf{U}$ and $\mathbf{F}$ are fixed. We specify $\mathbf{Y} = [\text{diag}(\mathbf{X}_{*1}), \text{diag}(\mathbf{X}_{*2}), ..., \text{diag}(\mathbf{X}_{*n})]$, where diag(.) denotes the diagonalization of a vector to a diagonal matrix. Then by removing the terms that are irrelevant to $\mathbf{W}$, the objective function can be rewritten as follows:

$$\min_{\mathbf{W}} \mathcal{L}(\mathbf{W}) = \|\mathbf{XW} + \mathbf{YU} - \mathbf{F}\|_F^2 + \alpha\|\mathbf{W}\|_{2,1}, \tag{10}$$

It is easy to verify that $\mathcal{L}(\mathbf{W})$ is a convex function, thus we can obtain the optimal solution of $\mathbf{W}$ by taking the derivative of $\mathcal{L}(\mathbf{W})$ w.r.t. the variable $\mathbf{W}$ and set it to be zero:

$$\frac{\partial \mathcal{L}(\mathbf{W})}{\partial \mathbf{W}} = 2\mathbf{X}^T(\mathbf{XW} + \mathbf{YU} - \mathbf{F}) + 2\alpha\mathbf{CW} = 0, \tag{11}$$

where $\mathbf{C}$ is a diagonal matrix whose diagonal element is $\mathbf{C}_{ii} = 1/2\|\mathbf{W}_{i*}\|_2$. It should be noted that in practice, $\|\mathbf{W}_{i*}\|_2$ could be zero technically. To tackle this problem, we add a very small constant ϵ on the denominator and make $\mathbf{C}_{ii} = 1/(2\|\mathbf{W}_{i*}\|_2 + \epsilon)$. From Eq. (11), we obtain the closed-form solution of $\mathbf{W}$ as:

$$\mathbf{W} = -(\mathbf{X}^T\mathbf{X} + \alpha\mathbf{C})^{-1}\mathbf{X}^T(\mathbf{YU} - \mathbf{F}). \tag{12}$$

Update Local Feature Weight U

Likewise, we update the local feature weight $\mathbf{U}$ when $\mathbf{W}$ and $\mathbf{F}$ are fixed. We convert the optimization problem in Eq. (9) into the following problem when we remove the terms that are irrelevant to $\mathbf{U}$:

$$\begin{aligned}\min_{\mathbf{U}} \mathcal{L}(\mathbf{U}) = \|\mathbf{XW} + \mathbf{YU} - \mathbf{F}\|_F^2 + \alpha\sum_{i=1}^{n}\|\mathbf{U}^i\|_{2,1} \\ + \beta\sum_{i,j=1}\mathbf{S}_{ij}\|\mathbf{U}^i - \mathbf{U}^j\|_F.\end{aligned} \tag{13}$$

We denote $\|\mathbf{XW}+\mathbf{YU}-\mathbf{F}\|_F^2$, $\sum_{i,j=1}\mathbf{S}_{ij}\|\mathbf{U}^i-\mathbf{U}^j\|_F$, and $\sum_{i=1}^n\|\mathbf{U}^i\|_{2,1}^2$ as $\mathcal{L}_1(\mathbf{U})$, $\mathcal{L}_2(\mathbf{U})$, and $\mathcal{L}_3(\mathbf{U})$, respectively.

Eq. (13) is convex w.r.t. $\mathbf{U}$. Hence, we take the derivative of $\mathcal{L}(\mathbf{U})$ w.r.t. $\mathbf{U}$ and set it to be zero. The derivative of $\mathcal{L}_1(\mathbf{U})$ can be computed as follows:

$$\frac{\partial \mathcal{L}_1(\mathbf{U})}{\partial \mathbf{U}} = 2\mathbf{Y}^T(\mathbf{YU} + \mathbf{XW} - \mathbf{F}). \tag{14}$$

The derivative of $\mathcal{L}_2(\mathbf{U})$ w.r.t. each variable $\mathbf{U}^i$ (a block of $\mathbf{U}$) can be computed as follows:

$$\frac{\partial \mathcal{L}_2(\mathbf{U})}{\partial \mathbf{U}^i} = \sum_{j=1}^{n}\frac{\mathbf{S}_{ji}(\mathbf{U}^i - \mathbf{U}^j)}{\|\mathbf{U}^j - \mathbf{U}^i\|_F} + \sum_{j=1}^{n}\frac{\mathbf{S}_{ij}(\mathbf{U}^i - \mathbf{U}^j)}{\|\mathbf{U}^i - \mathbf{U}^j\|_F}. \tag{15}$$

By putting all the derivative from different blocks together, we can obtain the derivative of $\mathcal{L}_2(\mathbf{U})$ w.r.t. $\mathbf{U}$ as follows:

$$\frac{\partial \mathcal{L}_2(\mathbf{U})}{\partial \mathbf{U}} = 2(\mathbf{E} \otimes \mathbf{I}_d)\mathbf{U}, \tag{16}$$

where $\mathbf{E}$ is a n-by-n square matrix with its element as:

$$\mathbf{E}_{ij} = \begin{cases} \sum_{l=1}^{n}\frac{\mathbf{S}_{il}}{\|\mathbf{U}^i - \mathbf{U}^l\|_F + \epsilon} - \frac{\mathbf{S}_{ij}}{\|\mathbf{U}^i - \mathbf{U}^j\|_F + \epsilon} & (i = j) \\ \frac{-\mathbf{S}_{ij}}{\|\mathbf{U}^i - \mathbf{U}^j\|_F + \epsilon} & (i \neq j). \end{cases} \tag{17}$$

In the above formulation, as $\mathcal{L}_2(\mathbf{U})$ is not smooth at some certain points due to the existence of $\ell_{2,1}$-norm sparse regularization, we impose a very small constant ϵ (as shown in Eq. (17)) to make sure the derivative of $\mathcal{L}_2(\mathbf{U})$ can be obtained at all points in its feasible region.

At last, we show the derivative of $\mathcal{L}_3(\mathbf{U})$ w.r.t. $\mathbf{U}$. The derivative is presented in a matrix format:

$$\frac{\partial \mathcal{L}_3}{\partial \mathbf{U}} = 2\mathbf{GU}, \tag{18}$$

where $\mathbf{G}$ is a nd-by-nd diagonal matrix. Its diagonal element is defined as follows:

$$\mathbf{G}_{ii} = \sum_{j=1}^{n}\frac{\mathbb{I}_{i,j}\|\mathbf{U}^j\|_{2,1}}{\|\mathbf{U}_{i*}\|_2 + \epsilon}, \tag{19}$$

where $\mathbb{I}_{i,j}$ is an indicator function such that $\mathbb{I}_{i,j} = 1$ if $\mathbf{U}_{i*}$ is from the block $\mathbf{U}^j$, otherwise $\mathbb{I}_{i,j} = 0$. As before, a very small constant ϵ is introduced to make the derivative computable at all points.

Combing the derivative of $\mathcal{L}_1(\mathbf{U})$, $\mathcal{L}_2(\mathbf{U})$, $\mathcal{L}_3(\mathbf{U})$ w.r.t. $\mathbf{U}$ together and set it the derivative to be zero, we get a closed-form solution for the concatenation of all local feature weights, which is:

$$\mathbf{U} = (\mathbf{Y}^T\mathbf{Y} + \beta\mathbf{E} \otimes \mathbf{I}_d + \alpha\mathbf{G})^{-1}\mathbf{Y}^T(\mathbf{F} - \mathbf{XW}). \tag{20}$$

Update Pseudo Class Labels F

At last, we discuss how to update the pseudo class labels $\mathbf{F}$. Specifically, when the other two model parameters $\mathbf{W}$ and $\mathbf{U}$ are fixed, the objective function now can be reformulated as follows:

$$\begin{aligned}\min_{\mathbf{F}} \; tr(\mathbf{F}^T(\mathbf{I}_n + \gamma\mathbf{L})\mathbf{F}) - 2tr(\mathbf{H}^T\mathbf{F}) + \frac{\theta}{2}\|\mathbf{F}^T\mathbf{F} - \mathbf{I}_c\|_F^2 \\ s.t. \; \mathbf{F} \geq 0,\end{aligned} \tag{21}$$

where $\mathbf{H} = \mathbf{XW} + \mathbf{YU}$. The above optimization problem is a convex optimization problem with nonnegative constraint. We introduce the Lagrangian multiplier $\mathbf{\Delta}_{ij}$ for the constraint $\mathbf{F}_{ij} \geq 0$. Then the Lagrangian function is given as follows:

$$\begin{aligned}\min_{\mathbf{F}} \mathcal{L}_\Delta(\mathbf{F}) = & \; tr(\mathbf{F}^T(\mathbf{I}_n + \gamma\mathbf{L})\mathbf{F}) - 2tr(\mathbf{H}^T\mathbf{F}) \\ & + \frac{\theta}{2}\|\mathbf{F}^T\mathbf{F} - \mathbf{I}_c\|_F^2 + tr(\Delta\mathbf{F}^T).\end{aligned} \tag{22}$$

Algorithm 1 Unsupervised Personalized Feature Selection

```
Input: $\mathbf{X} \in \mathbb{R}^{n \times d}$, $\alpha, \beta, \gamma, \theta = 10^8, m$.
Output: $\mathbf{W} \in \mathbb{R}^{d \times c}$, $\mathbf{U} \in \mathbb{R}^{nd \times c}$, $\mathbf{F} \in \mathbb{R}^{n \times c}$.
Initialize $\mathbf{W}$, $\mathbf{U}$, $\mathbf{F}$;
Compute affinity graph $\mathbf{S}$ and Laplacian matrix $\mathbf{L}$;
while objective function of UPFS in Eq. (9) not converge do
    Compute matrix $\mathbf{C}$;
    Update $\mathbf{W}$ by Eq. (12);
    Compute matrix $\mathbf{E}$ and $\mathbf{G}$;
    Update $\mathbf{U}$ by Eq. (20);
    Compute matrix $\mathbf{H}$;
    Update $\mathbf{F}$ by Eq. (25);
    Normalize $\mathbf{F}$;
end while
for each instance $\mathbf{x}_i$ do
    Compute $\mathbf{K} = \mathbf{W} + \mathbf{U}_i$
    Rank features according to $\|\mathbf{K}_{j*}\|_2^2$ in a descending order;
    Return the top $m$ ranked features for $\mathbf{x}_i$;
end for
```

By setting the derivative of $\mathcal{L}_{\triangle}(\mathbf{F})$ w.r.t. $\mathbf{F}$ to be zero, we get the following formulation:

$$\mathbf{\Delta} = 2\theta\mathbf{F} - 2\theta\mathbf{F}\mathbf{F}^T\mathbf{F} - 2(\mathbf{I}_n + \gamma\mathbf{L})\mathbf{F} + 2\mathbf{H}. \tag{23}$$

The Karush-Kuhn-Tuckre (KKT) condition (Boyd and Vandenberghe 2004) for the nonnegative constraint of $\mathbf{F}$ gives:

$$[\theta\mathbf{F} - \theta\mathbf{F}\mathbf{F}^T\mathbf{F} - (\mathbf{I}_n + \gamma\mathbf{L})\mathbf{F} + \mathbf{H}]_{ij}\mathbf{F}_{ij} = 0. \tag{24}$$

Therefore, the multiplicative update rule for $\mathbf{F}$ is as follows:

$$\mathbf{F}_{ij} \leftarrow \mathbf{F}_{ij}\sqrt{\frac{\theta\mathbf{F} + \mathbf{H}}{(\mathbf{I}_n + \gamma\mathbf{L})\mathbf{F} + \theta\mathbf{F}\mathbf{F}^T\mathbf{F}}}. \tag{25}$$

After that, we normalize $\mathbf{F}$ to further ensure that it satisfies the orthogonal constraint such that $\mathbf{F}^T\mathbf{F} = \mathbf{I}_c$.

With the above updating rules, the pseudo code of the proposed UPFS framework is summarized in Algorithm 1.

Convergence Analysis

In each iteration, when we update the global feature weight $\mathbf{W}$ and localized feature weight $\mathbf{U}$, we have a closed form solution. Thus, the objective function is guaranteed to decrease. In updating the pseudo class labels $\mathbf{F}$, we rely on the multiplicative update rule which is widely used in solving the nonnegative matrix factorization problems. Its convergence has been proven in (Lee and Seung 2001). Thus, the objective function also decreases when we update $\mathbf{F}$. In conclusion, as the objective function value is guaranteed to decrease in each iteration, its convergence is proved. Empirically, our proposed algorithm converges within 100 iterations of datasets used in this paper.

Experiments

In this section, we conduct experiments to verify the effectiveness of the proposed unsupervised personalized feature selection framework - UPFS. Before presenting the detailed experimental results, we first introduce the experimental settings. Further experiments are performed to analyze the impact of model parameters on UPFS.

Table 1: Dataset description

Type	Data	# instance	# features	# classes
Text	CNNStory	142	8,682	10
	BlogCatalog	232	5,196	6
	Flickr	150	8,189	9
	DBLP	327	5,665	24
Image	Yale	165	1,024	15
	warpPIE10P	210	2,420	10
Biology	Carcinoma	174	9,182	11
	Prostate_GE	102	5,966	2
	TOX171	171	5,748	4

Experimental Settings

We choose 9 datasets from various domains, including (1) four text datasets: CNNStory, BlogCatalog, Flickr and DBLP; (2) two image datasets: Yale and warpPIE10P; (3) three biology datasets: Carcinoma, Prostate_GE and TOX-171. Normally, datasets that demand feature selection are often in the "fat" shape: with a small number of instances but a large number of features. Therefore, we conduct experiments on this kind of datasets. Detailed statistics of the used datasets are shown in Table 1.

Following the common experiment settings of unsupervised feature selection (Cai, Zhang, and He 2010; Du and Shen 2015), we assess the performance of feature selection by the clustering performance in terms of Clustering Accuracy (ACC) and Normalized Mutual Information (NMI). Suppose C and C' are the clustering results from ground truth class labels and the predicted cluster labels, respectively. The mutual information between C and C' can be defined as:

$$MI(C, C') = \sum_{c_i \in C, c'_j \in C'} p(c_i, c'_j) log \frac{p(c_i, c'_j)}{p(c_i)p(c'_j)}, \tag{26}$$

where $p(c_i)$ and $p(c'_j)$ indicate the probabilities of instances in cluster c_i and c'_j, respectively. $p(c_i, c'_j)$ indicates the joint probability of instances in cluster c_i and in c'_j simultaneously. NMI is further defined based on MI:

$$NMI(C, C') = \frac{MI(C, C')}{max(H(C), H(C'))}, \tag{27}$$

where $H(C)$ and $H(C')$ are the entropies of two clusters C and C', respectively. Suppose g_i and h_i denote the clustering result and the ground truth label for instance $\mathbf{x}_i$, respectively. Then, accuracy (ACC) is defined as:

$$ACC = \frac{1}{n}\sum_{i=1}^{n} \delta(h_i, map(g_i)), \tag{28}$$

where $\delta(.)$ is an indicator function with $\delta(x, y) = 1$ if $x = y$, otherwise $\delta(x, y) = 0$. $map(x)$ permutes the predicted cluster labels to match the ground truth maximally.

Specifically, we first apply the unsupervised feature selection algorithms to select features and then employ k-means clustering algorithm on the selected features. As k-means may converge to different local optima because of different initialization, we perform the experiments 20 times and report the average clustering performance with the standard deviation. In the proposed UPFS framework, for each instance $\mathbf{x}_i$, we first select the top m features according to the corresponding entry in $\mathbf{K} = \mathbf{W} + \mathbf{U}^i$, then we make the feature values of the bottom ranked $d - m$ features in $\mathbf{x}_i$ to be zero. Afterwards, we employ the refined data matrix for clustering evaluation. The higher the clustering performance, the better the selected features are.

Performance Evaluation

We compare the proposed UPFS framework with the following representative unsupervised feature selection algorithms:

- All: all features are employed for clustering.
- Laplacian Score (LS): selects features that best preserve the local manifold structure of data (He, Cai, and Niyogi 2005).
- SPEC: selects features based on spectral analysis (Zhao and Liu 2007).
- MCFS: selects features based on spectral analysis and ℓ_1-norm sparse regression (Cai, Zhang, and He 2010).
- NDFS: selects features with nonnegative spectral analysis and $\ell_{2,1}$-norm sparse regression (Li et al. 2012).
- RUFS: performs robust label learning and robust feature selection simultaneously (Qian and Zhai 2013).
- FSASL: performs data structure learning and feature selection in a joint framework (Du and Shen 2015).

In Laplacian Score, MCFS, NDFS, FSASL and UPFS, we specify the number of nearest neighbors k as 5. These baseline methods and the proposed UPFS framework have different sets of parameters. To have a fair comparison, we tune these parameters by the "grid-search" strategy. It is still an open question to decide the optimal number of selected features in feature selection research. Thus, we set the number of selected features among $\{10, 20, ..., 300\}$ and report the best clustering results. The clustering results in terms of ACC and NMI are shown in Table 2 and Table 3. The following observations are induced from these two tables:

- Feature selection is necessary in most cases. As indicated in the tables, when we use feature selection algorithms to find discriminative features, it can improve the clustering performance.
- The proposed UPFS framework outperforms baseline methods in many cases, thus its effectiveness is verified. We also perform a two-sample one tail t-test between UPFS and other methods, the results show that UPFS is significantly better with a significance level of 0.05. The improvements can be attributed as follows: (1) instances in a dataset can be highly idiosyncratic while a global feature weight cannot fully capture the individuality of instances; (2) instances more or less share some commonality, hence, it would be beneficial to use a conjunction of global feature weight and localized feature weight for the pseudo label prediction.
- The proposed UPFS performs better on text data and biology data than image data. The reason is that in these two domains, instances are more likely to show high individuality, which can be characterized by a number of personalized features.
- NDFS is a special case of the UPFS by eliminating the exclusive group lasso term and the network lasso term. NDFS only employs one single global feature weight, the improvement of UPFS over NDFS indicates that it is indeed helpful to find instances-specific features.

Parameter Study

We investigate the impacts of parameters of α, β and γ on the performance of UPFS. Among them, α controls the feature sparsity, β controls to what extent instances can borrow strength from neighbors in learning localized feature weight, γ controls how well the pseudo class labels preserve the local geometry structure of data. To study how its variation affects the feature selection performance, we fix two parameters each time and vary the third one in the range of $\{0.001, 0.01, 1, 10, 100, 1000\}$. The performance variation of these parameters w.r.t. the number of selected features are shown in Figure 2. Due to space limit, we only show the parameter study results in terms of ACC on the BlogCatalog dataset as we have similar observations on other datasets. As can be observed, the best clustering performance is achieved when α, β and γ is in the range of 0.1 to 10. Generally speaking, the proposed UPFS framework is not very sensitive to these model parameters, and it is safe to tune them in a wide range, which is appealing in practice.

Related Work

Feature selection has shown its effectiveness in preparing high-dimensional data for many learning tasks. As label information is rather difficult to obtain in many applications, there is a surge of research studying unsupervised feature selection. Without label information to assess feature relevance, unsupervised feature selection methods employ some alternative criteria such as data similarity (He, Cai, and Niyogi 2005; Zhao and Liu 2007), local discriminative information (Yang et al. 2011; Li et al. 2012; Qian and Zhai 2013) and data reconstruction error (Masaeli et al. 2010; Farahat, Ghodsi, and Kamel 2013; Li, Tang, and Liu 2017). Recently, sparse learning based unsupervised learning with ℓ_1-norm regularization or $\ell_{2,1}$-norm regularization has received increasingly attention. However, these methods predominately assume that all instances share the same feature weights, which is not necessarily true in practice.

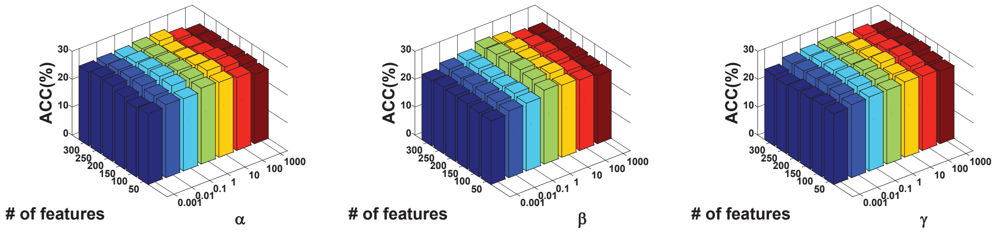
Personalized feature learning has significant implications in many real-world applications such as medical predictive modeling (Xu, Zhou, and Tan 2015), node classification (Li et al. 2017) and sentiment classification (Gong, Al Boni,

Table 2: Clustering results (ACC%± std) of different unsupervised feature selection algorithms.

Data	All	LS	SPEC	MCFS	NDFS	RUFS	FSASL	UPFS
CNNStory	50.70±3.83	54.02±6.83	24.13±1.33	54.61±7.33	51.17±6.68	53.24±5.58	53.81±5.02	**54.82±3.92**
BlogCatalog	24.78±1.85	25.08±3.80	23.28±0.20	26.25±2.02	26.51±0.63	27.91±2.76	27.01±1.55	**28.30±0.85**
Flickr	18.70±0.82	22.83±0.41	18.67±1.05	19.37±0.98	22.40±1.99	20.53±1.83	22.61±2.32	**22.97±0.39**
DBLP	34.01±1.07	35.62±2.56	35.17±2.33	38.07±3.89	35.60±5.84	38.01±3.09	39.27±0.50	**39.72±1.22**
Yale	38.30±3.83	41.30±3.01	38.52±3.42	40.18±1.86	42.48±3.36	42.61±2.96	**42.87±2.47**	39.92±2.60
warpPIE10P	26.67±2.03	38.69±2.63	**40.38±2.38**	27.29±1.98	37.60±3.56	38.83±2.45	33.40±1.05	28.74±1.48
Carcinoma	59.57±7.42	72.87±6.66	53.85±4.03	76.49±8.33	73.11±7.8	76.24±5.41	75.16±4.81	**76.98±4.69**
Prostate-GE	58.28±0.50	57.50±0.47	57.45±0.48	58.82±1.12	57.81±1.11	57.84±1.12	57.97±0.22	**59.37±0.46**
TOX-171	41.87±3.97	41.96±1.61	42.84±4.40	43.45±3.87	47.89±0.34	48.26±2.30	45.49±0.76	**49.98±1.70**

Table 3: Clustering results (NMI%± std) of different unsupervised feature selection algorithms.

Data	All	LS	SPEC	MCFS	NDFS	RUFS	FSASL	UPFS
CNNStory	50.65±4.82	57.25±3.02	36.02±0.79	52.88±8.45	51.63±5.8	53.24±4.49	53.29±3.38	**59.71±4.08**
BlogCatalog	4.14±2.47	5.75±1.77	2.18±0.72	6.27±0.99	5.11±2.07	6.46±2.93	5.94±1.04	**7.20±0.08**
Flickr	5.74±0.39	11.29±0.74	5.43±0.21	7.17±1.60	11.13±1.56	8.42±1.76	9.05±0.13	**11.56±1.82**
DBLP	3.86±1.63	7.06±2.66	1.19±0.30	6.64±1.98	6.05±2.50	5.43±2.44	7.99±2.37	**9.86±0.44**
Yale	43.89±3.93	46.71±2.17	44.16±2.12	44.90±1.40	50.04±2.10	51.83±2.19	**53.09±0.95**	45.49±3.53
warpPIE10P	25.61±4.18	38.11±3.49	42.81±2.73	26.16±3.59	**43.59±5.20**	35.86±4.16	32.68±6.02	29.54±0.82
Carcinoma	62.56±7.46	78.23±4.28	57.21±3.57	80.99±4.69	77.12±4.72	79.82±3.86	78.04±5.5	**81.44±3.67**
Prostate_GE	2.07±0.31	1.61±0.20	1.59±0.21	3.09±2.13	1.80±0.34	1.80±0.34	2.65±0.08	**3.64±0.63**
TOX171	13.37±3.71	12.04±2.54	16.10±1.65	14.12±1.89	18.40±0.21	16.30±2.63	16.00±0.71	**18.95±1.28**



(a) Effect of α ($\beta = 1, \gamma = 0.1$) (b) Effect of β ($\alpha = 0.1, \gamma = 0.1$) (c) Effect of γ ($\alpha = 0.1, \beta = 1$)

Figure 2: Parameter study on BlogCatalog dataset in terms of ACC.

and Wang 2016; Wu and Huang 2016). In (Xu, Zhou, and Tan 2015), the authors regard the personalized feature learning as a multi-task learning task. The proposed personalized models are assumed to share task relatedness by low rankness and the low rank matrices are forced to be sparse. As features could be high-dimensional, (Yamada et al. 2016) studies how to build a high-dimensional localized regression model. It assumes that different instances have different sets of localized feature weights, the proposed model is shown to outperform traditional lasso methods. Our work differs from these as: (1) our model uses as a conjunction of global model and localized model for feature learning; (2) to the best of our knowledge, we are the first how to perform study personalized feature selection in an unsupervised scenario.

Conclusions and Future Work

Real-world high-dimensional data is often unlabeled. Without label information to assess feature relevance, unsupervised feature selection is more appealing in practical usage. Existing unsupervised feature selection algorithms attempt to find the same set of discriminative features for all instances. However, these methods inevitably ignore the individuality of instances as important features for different instances could vary significantly. To tackle this problem, we study a novel problem of unsupervised personalized feature selection. Specifically, we propose a principled framework UPFS to find a subset of shared features and instance-specific discriminative features for each instance. Experimental results on real-world datasets corroborate the effectiveness of the proposed framework. Future work can be focused on designing more efficient distributed optimization algorithm for UPFS and deploy it on real applications.

Acknowledgements

This material is supported by the National Science Foundation (NSF) grant 1614576, and the Office of Naval Research (ONR) grant N00014-16-1-2257.

References

Boyd, S., and Vandenberghe, L. 2004. *Convex optimization*. Cambridge university press.

Cai, D.; Zhang, C.; and He, X. 2010. Unsupervised feature selection for multi-cluster data. In *Proceedings of the 16th ACM SIGKDD International conference on Knowledge Discovery and Data Mining*, 333–342. ACM.

Du, L., and Shen, Y.-D. 2015. Unsupervised feature selection with adaptive structure learning. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 209–218. ACM.

Farahat, A. K.; Ghodsi, A.; and Kamel, M. S. 2011. An efficient greedy method for unsupervised feature selection. In *Proceedings of the 2011 IEEE 11th International Conference on Data Mining*, 161–170. IEEE.

Farahat, A. K.; Ghodsi, A.; and Kamel, M. S. 2013. Efficient greedy feature selection for unsupervised learning. *Knowledge and information systems* 35(2):285–310.

Friedman, J.; Hastie, T.; and Tibshirani, R. 2001. *The elements of statistical learning*. Springer series in statistics New York.

Gong, L.; Al Boni, M.; and Wang, H. 2016. Modeling social norms evolution for personalized sentiment classification. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, 855–865.

Guyon, I., and Elisseeff, A. 2003. An introduction to variable and feature selection. *Journal of machine learning research* 3(Mar):1157–1182.

Hallac, D.; Leskovec, J.; and Boyd, S. 2015. Network lasso: Clustering and optimization in large graphs. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 387–396. ACM.

He, X.; Cai, D.; and Niyogi, P. 2005. Laplacian score for feature selection. In *Advances in Neural Information Processing Systems*, 507–514.

Kong, D.; Fujimaki, R.; Liu, J.; Nie, F.; and Ding, C. 2014. Exclusive feature learning on arbitrary structures via l1, 2-norm. In *Advances in Neural Information Processing Systems*, 1655–1663.

Kong, D.; Liu, J.; Liu, B.; and Bao, X. 2016. Uncorrelated group lasso. In *Proceedings of the 30th AAAI Conference on Artificial Intelligence*, 1765–1771. AAAI Press.

Lee, D. D., and Seung, H. S. 2001. Algorithms for nonnegative matrix factorization. In *Advances in Neural Information Processing Systems*, 556–562.

Li, J., and Liu, H. 2017. Challenges of feature selection for big data analytics. *IEEE Intelligent Systems* 32(2):9–15.

Li, Z.; Yang, Y.; Liu, J.; Zhou, X.; Lu, H.; et al. 2012. Unsupervised feature selection using nonnegative spectral analysis. In *Proceedings of the 26th AAAI Conference on Artificial Intelligence*, 1026–1032. AAAI Press.

Li, J.; Cheng, K.; Wang, S.; Morstatter, F.; Robert, T.; Tang, J.; and Liu, H. 2016. Feature selection: A data perspective. *arXiv:1601.07996*.

Li, J.; Wu, L.; Zaïane, O. R.; and Liu, H. 2017. Toward personalized relational learning. In *Proceedings of the 2017 SIAM International Conference on Data Mining*, 444–452. SIAM.

Li, J.; Tang, J.; and Liu, H. 2017. Reconstruction-based unsupervised feature selection: an embedded approach. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence*, 2159–2165. IJCAI/AAAI.

Liu, H., and Motoda, H. 2007. *Computational methods of feature selection*. CRC Press.

Liu, J.; Ji, S.; and Ye, J. 2009. Multi-task feature learning via efficient l2,1-norm minimization. In *Proceedings of the 25th conference on Uncertainty in Artificial Intelligence*, 339–348. AUAI Press.

Masaeli, M.; Yan, Y.; Cui, Y.; Fung, G.; and Dy, J. G. 2010. Convex principal feature selection. In *Proceedings of the 2010 SIAM International Conference on Data Mining*, 619–628. SIAM.

Nemhauser, G. L. 1998. *Integer programming and combinatorial optimization*. Springer.

Ng, A. Y.; Jordan, M. I.; Weiss, Y.; et al. 2001. On spectral clustering: Analysis and an algorithm. In *Advances in Neural Information Processing Systems*, 849–856.

Nie, F.; Huang, H.; Cai, X.; and Ding, C. H. 2010. Efficient and robust feature selection via joint l2, 1-norms minimization. In *Advances in Neural Information Processing Systems*, 1813–1821.

Qian, M., and Zhai, C. 2013. Robust unsupervised feature selection. In *Proceedings of the 23th International Joint Conference on Artificial Intelligence*, 1621–1627. IJCAI/AAAI.

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

Von Luxburg, U. 2007. A tutorial on spectral clustering. *Statistics and computing* 17(4):395–416.

Wu, F., and Huang, Y. 2016. Personalized microblog sentiment classification via multi-task learning. In *Proceedings of the 30th AAAI Conference on Artificial Intelligence*, 3059–3065. AAAI Press.

Xu, J.; Zhou, J.; and Tan, P.-N. 2015. Formula: Factorized multi-task learning for task discovery in personalized medical models. In *Proceedings of the 2015 SIAM International Conference on Data Mining*, 496–504. SIAM.

Yamada, M.; Takeuchi, K.; Iwata, T.; Shawe-Taylor, J.; and Kaski, S. 2016. Localized lasso for high-dimensional regression. *arXiv preprint arXiv:1603.06743*.

Yang, Y.; Shen, H. T.; Ma, Z.; Huang, Z.; and Zhou, X. 2011. l2,1-norm regularized discriminative feature selection for unsupervised learning. In *Proceedings of the 22nd International Joint Conference on Artificial Intelligence*, 1589–1594. AAAI Press.

Zhao, Z., and Liu, H. 2007. Spectral feature selection for supervised and unsupervised learning. In *Proceedings of the 24th International Conference on Machine Learning*, 1151–1157. ACM.