2014 International Joint Conference on Neural Networks (IJCNN)
July 6-11, 2014, Beijing, China

Unsupervised Robust Bayesian Feature Selection

Jianyong Sun
School of Arts, Media and Computer Games
The Abertay University
Dundee, Scotland, DD1 1HG
Email: j.sun@abertay.ac.uk

Aimin Zhou
Department of Computer Science and Technology
East China Normal University
500 Dongchuan Road, Shanghai, 200241, China.
Email: amzhou@cs.ecnu.edu.cn

***Abstract*—In this paper, we proposed a generative graphical model for unsupervised robust feature selection. The model assumes that the data are independent and identically sampled from a finite mixture of Student-t distribution for dealing with outliers. The Student *t*-distribution works as the building block for robust clustering and outlier detection. Random variables that represent the features' saliency are included in the model for feature selection. As a result, the model is expected to simultaneously realise unsupervised clustering, feature selection and outlier detection. The inference is carried out by a tree-structured variational Bayes (VB) algorithm. The feature selection capability is realised by estimating the feature saliencies associated with the features. The adoption of full Bayesian treatment in the model realises automatic model selection. Experimental studies showed that the developed algorithm compares favourably against existing unsupervised Bayesian feature selection algorithm in terms of commonly-used internal and external cluster validity indices on controlled experimental settings and benchmark data sets. The controlled experimental study also showed that the developed algorithm is capable of exposing the outliers and finding the optimal number of components (model selection) accurately.**

I. Introduction

Competitive performances of clustering algorithms cannot be expected on high-dimensional datasets due to the curse of dimensionality problem. A subset of features, if properly selected, could improve the clustering performance [1].

Existing learning algorithms for feature selection in the literature were proposed to tackle data sets with labels (supervised feature selection) and without labels (unsupervised feature selection) [1]. In the paper, we focus on unsupervised feature selection. As discussed in [1], feature selection is to select a subset of most informative features (or attributes, variables) rather than selecting a combination of features, such as PCA, ICA, and so on.

Mathematically, the feature selection problem can be formulated as follows: given a set Y of features of size $D = |Y|$, denote $\mathcal{X}_d$ the set of all possible variables. The unsupervised feature selection is to select an optimal subset of features in terms of a criterion function $J(x)$:

$$x_{opt} = \arg\max_{x \subset \mathcal{X}_d} J(x)$$

Basically, the feature selection methods can be classified into three categories, namely filter, wrapper and hybrid methods with different criterion function. Interested readers please refer to [2] for a detailed survey on feature selection.

Various feature selection methods for unsupervised learning have been developed. These methods can be categorised into several groups[1]. In [3], consistency based feature selection methods were evaluated. Methods based on information entropy and correlation have also been developed such as in [4], [5], [6], [7], [8], [9]. Local learning-based feature selection methods have been extensively studies, especially recently. For examples, in [10], [11], [12], [13], nonnegative matrix factorisation is used, where the loading matrix is penalised by L_2 or L_1 norms. Moreover, L_2 and L_1 norms have been widely applied in various feature selection methods, such as in [14], [15], [16], [17]. In [18], [19], [20], [21], spectral analysis is applied. These algorithms have shown their capability in feature selection. But those mentioned algorithms are not able to select the optimal number of clusters in a principled way.

Alternatively, the probability model based feature selection methods have been developed in [22], [23], [24], [25], which are based on a finite mixture of Gaussians. These methods can simultaneously realise the feature selection and clustering. As well known, the mixture of Gaussians are not able to deal with outliers properly. Outliers or scattered objects exist elsewhere in real datasets. The outliers, if not appropriately tackled, should seriously deteriorate the performances of learning algorithms, such as the case in the finite mixture. Moreover, the outliers could also make the optimal selection of subset features get much more difficult. It is thus indispensable to propose a principled approach to realise the selection of the most informative features, whilst eliminating the effects of outliers and improving the clustering performance.

In this paper, we propose a hierarchical latent variable model so that all the problems mentioned above can be dealt simultaneously. That is, a finite mixture of Student t-distributions is the backbone of the model, and the features are associated with variables that represent the importance of the features. The Student t-distribution has a heavy tail so that the outliers can be properly accounted. The feature saliencies are able to select the most informative features. A Bayesian variational framework is presented for training the model that maximise a lower bound of the marginal likelihood. To increase the efficiency of the maximisation, a tree-structured factorisation of the latent variables is proposed.

In the rest of the paper, Section II presented the proposed

[1]This does not intent to a comprehensive review.

978-1-4799-1484-5/14/$31.00 ©2014 IEEE

latent variable. The inference was presented in Section III, in which the tree-structured variational Bayes algorithm is described. The experimental study was presented in Section V. The interpretation of the model was described in Section IV. In the study, controlled experiments were firstly conducted to justify the outperformance of the developed model over the model using Gaussian distributions. Then the developed algorithm was compared with a full-factorised VB algorithm to justify the outperformances of the tree-structured factorisation. Section VI concludes the paper.

II. Model

Based on the definition of irrelevance of features as seen in [23], the ℓ-th feature is irrelevant if the ℓ-th feature comes from a common distribution. Here, we use a random binary variable vector $\Phi = (\phi_1, \cdots, \phi_d)$ to denote the relevance of the features, i.e. $\phi_\ell = 1$ if the ℓ-th feature is relevant, and 0 otherwise. Thinking of a mixture of Student *t*-distribution and taking the features' relevance / irrelevance into consideration, we have the following model:

$$\begin{aligned} p(\mathbf{y}|\Phi) &= \sum_{k=1}^{K} \pi_k \left\{ \prod_{\ell=1}^{d} \left[S_t(y_\ell|\theta_{k\ell}) \right]^{\phi_\ell} \left[S_t(y_\ell|\theta_{0\ell}) \right]^{1-\phi_\ell} \right\} \\ &= \sum_{k=1}^{K} \pi_k \left\{ \prod_{\ell=1}^{d} \phi_\ell S_t(y_\ell|\theta_{k\ell}) + (1-\phi_\ell) S_t(y_\ell|\theta_{0\ell}) \right\} \end{aligned}$$

where S_t is the Student *t*-distribution. From the above equation, it can be seen that if some features with $\phi_\ell = 0$, they are assumed to follow a distribution which is independent of the class assignment, i.e. a common distribution.

Note that the Student *t*-distribution can be written as a convolution of a Gaussian and a gamma distribution as follows:

$$S_t(y|\theta) = \int \mathcal{N}(y|\mu, \sigma u)\, \mathcal{G}\left(u|\frac{\nu}{2}, \frac{\nu}{2}\right) du$$

where σ is the precision and $\theta = (\mu, \sigma, \nu)$ is the parameters, and $\mathcal{G}(x|a,b) = b^a x^{a-1} \exp(-bx)/\Gamma(a)$.

In our model, considering the hierarchical representation of the Student *t*-distribution, we introduce latent variables $\mathbf{u_n} = (u_{n1}, \cdots, u_{nd})$ and $\Phi_n = (\phi_{n1}, \cdots, \phi_{nd})$, where each $u_{n\ell}$ and $\Phi_{n\ell}$ correspond to each data and feature. Moreover, if we let $\mathbf{z}_n$ is the discrete latent variable to specify which cluster that the n-th data belongs to, the model can be written hierarchically as follows:

$$\begin{aligned} p(\mathbf{y}_n|\Phi_n, \mathbf{u_n}, \mathbf{z}_n = j) &= \prod_{\ell=1}^{d} p(y_{n\ell}|\phi_{n\ell}, u_{n\ell}, \mathbf{z}_n = j) \\ p(\mathbf{u_n}|\Phi_n, \mathbf{z}_n = j) &= \prod_{\ell=1}^{d} p(u_{n\ell}|\phi_{n\ell}, \mathbf{z}_n = j) \end{aligned}$$

where for the right hand side probabilities, we have the following probabilities:

$$p(y_{n\ell}|\phi_{n\ell}, u_{n\ell}, \mathbf{z}_n = j) = \begin{cases} \mathcal{N}(y_{n\ell}|\mu_{j\ell}, \sigma_{j\ell} u_{n\ell}) & : \phi_{n\ell} = 1 \\ \mathcal{N}(y_{n\ell}|\chi_\ell, \tau_\ell u_{n\ell}) & : \phi_{n\ell} = 0 \end{cases}$$

and

$$p(u_{n\ell}|\phi_{n\ell}, \mathbf{z}_n = j) = \begin{cases} \mathcal{G}\left(u_{n\ell}|\frac{\nu_{j\ell}}{2}, \frac{\nu_{j\ell}}{2}\right) & : \phi_{n\ell} = 1 \\ \mathcal{G}\left(u_{n\ell}|\frac{\gamma_\ell}{2}, \frac{\gamma_\ell}{2}\right) & : \phi_{n\ell} = 0 \end{cases}$$

Integrating over $\mathbf{u}_n$ given $\mathbf{z}_n = j$ and Φ_n, we can derive the integration as shown in Fig. 1.

Integrating $p(\mathbf{y}_n|\mathbf{z}_n = j, \Phi_n)$ over $\mathbf{z}_n$, we recover $p(\mathbf{y}_n|\Phi_n)$. Given Bernoulli prior probability over Φ_n as

$$p(\Phi_n|\beta) = \prod_{\ell=1}^{d} \beta_\ell^{\phi_{n\ell}} (1-\beta_\ell)^{1-\phi_{n\ell}}$$

and integration Φ_n, we obtain the following likelihood function

$$p(\mathbf{y}_n|\beta) = \sum_{j=1}^{J} \pi_j \left\{ \prod_{\ell=1}^{d} \beta_\ell S_t(y_\ell|\theta_{k\ell}) + (1-\beta_\ell) S_t(y_\ell|\theta_{0\ell}) \right\}$$

Here β_ℓ is called feature saliency in [23].

To realise model selection, i.e. selecting the optimal number of components, we apply the full Bayesian treatment. In [22], the authors developed a *semi-Bayesian* treatment to the method developed The obvious disadvantage of the method in [22] is its capability to deal with outliers. Moreover, it is also sceptical to use a semi-Bayesian for the purpose of model selection.

In the proposed model, bear in mind that the probability distributions $p(\mathbf{y}|\mathbf{z}_n, \Phi, \Theta)$ ($\Theta = \{\mu, \sigma, \chi, \tau, \pi, \beta\}$) and $p(\mathbf{u}_n|\mathbf{z}_n, \Phi_n)$ ($\mathbf{v} = \{\nu_j, \nu_0\}$) are fully factorized. In the sequel, we denote the latent variables as $\mathbf{h}_n = \{\mathbf{u}_n, \mathbf{z}_n, \Phi_n, 1 \leq n \leq N\}$. According to the model, the complete likelihood of a data $\mathbf{y}_n$ can be written as follows:

$$L_{\mathcal{C}} = p(\mathbf{y}_n, \mathbf{u}_n, \mathbf{z}_n, \mathbf{\Phi}_n) p(\Theta) \tag{1}$$

where

$$\begin{aligned} p(\mathbf{y}_n, \mathbf{h}_n) &= p(\mathbf{y}_n|\mathbf{u}_n, \mathbf{z}_n, \Phi_n) p(\mathbf{u}_n|\mathbf{z}_n, \Phi_n) p(\mathbf{z}_n) p(\Phi_n) \\ p(\Theta) &= p(\mu) p(\sigma) p(\beta) p(\pi) p(\chi) p(\tau) \end{aligned}$$

Note that we assume the same hyper-parameters of the prior distributions corresponding to the parameters θ, θ_0. We do not assume any priors for ν and ν_0 since there are no conjugate priors.

In the following, we use n, ℓ, j to denote the indices of training data point, the features and the mixing components. We also omit the typeset of parameters in the formula.

Fig. 2 shows the proposed graphical model.

The prior probabilities for the latent variables are defined as follows:

$$\begin{aligned} p(\mathbf{y}_n|\mathbf{z}_n, \mathbf{u}_n, \Phi_n) &= \prod_{j=1}^{K} \left[\prod_{\ell=1}^{d} p(y_{n\ell}|\phi_{n\ell}, u_{n\ell}, j) \right]^{1_{\mathbf{z}_n=j}} \\ p(\mathbf{u}_n|\mathbf{z}_n, \Phi_n) &= \prod_{j=1}^{K} \left[\prod_{\ell=1}^{d} p(u_{n\ell}|\phi_{n\ell}, j) \right]^{1_{\mathbf{z}_n=j}} \end{aligned}$$

and $p(\mathbf{z}_n|\pi) = \prod_{j=1}^{K} \pi_k^{1_{\mathbf{z}_n=j}}$ where $1_{\mathbf{z}_n=j}$ is the Kronecker delta.

$$
\begin{aligned}
p(\mathbf{y}_n|j,\Phi_n) &= \int \prod_{\ell} \left[\mathcal{N}\left(y_{n\ell}|\mu_{j\ell},\sigma_{j\ell}u_{n\ell}\right)\mathcal{G}\left(u_{n\ell}|\frac{\nu_{j\ell}}{2},\frac{\nu_{j\ell}}{2}\right)\right]^{\phi_{n\ell}} \left[\mathcal{N}\left(y_{n\ell}|\chi_{\ell},\tau_{\ell}u_{n\ell}\right)\mathcal{G}\left(u_{n\ell}|\frac{\gamma_{\ell}}{2},\frac{\gamma_{\ell}}{2}\right)\right]^{1-\phi_{n\ell}} du_{n\ell} \\
&= \prod_{\ell}\int \left\{\phi_{n\ell}\left[\mathcal{N}\left(y_{n\ell}|\mu_{j\ell},\sigma_{j\ell}u_{n\ell}\right)\mathcal{G}\left(u_{n\ell}|\frac{\nu_{j\ell}}{2},\frac{\nu_{j\ell}}{2}\right)\right] + (1-\phi_{n\ell})\left[\mathcal{N}\left(y_{n\ell}|\chi_{\ell},\tau_{\ell}u_{n\ell}\right)\mathcal{G}\left(u_{n\ell}|\frac{\gamma_{\ell}}{2},\frac{\gamma_{\ell}}{2}\right)\right]\right\} du \\
&= \prod_{\ell}\left\{\prod_{\ell=1}^{d}\phi_{n\ell}S_t(y_{\ell}|\theta_{k\ell}) + (1-\phi_{n\ell})S_t(y_{\ell}|\theta_{0\ell})\right\}
\end{aligned}
$$

Fig. 1. The integration of the hierarchical probability model.

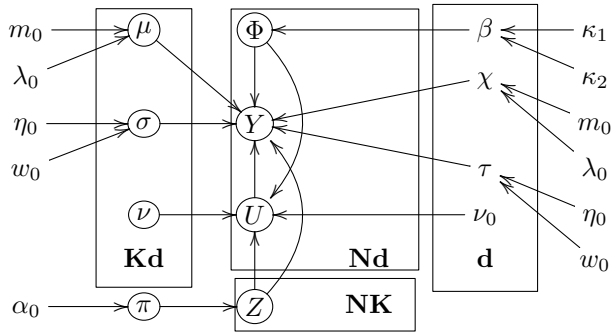


Fig. 2. The hierarchical graphical model.

The prior probabilities associated with the parameters are as follows:

$$
\begin{aligned}
p(\beta) &= \prod_{\ell=1}^{d} Be(\beta_{\ell}|\kappa_1,\kappa_2) \\
p(\sigma) &= \prod_{j}\prod_{\ell} p(\sigma_{j\ell}) = \prod_{j}\prod_{\ell}\mathcal{G}(\sigma_{j\ell}|\frac{\eta_0}{2},\frac{\omega_0}{2}) \\
p(\mu) &= \prod_{j}\prod_{\ell} p(\mu_{j\ell}) = \prod_{j}\prod_{\ell}\mathcal{N}(\mu_{j\ell}|m_0,\lambda_0) \\
p(\chi) &= \prod_{\ell} p(\chi_{\ell}) = \prod_{\ell}\mathcal{N}(\chi_{\ell}|m_0,\lambda_0) \\
p(\tau) &= \prod_{\ell} p(\tau_{\ell}) = \prod_{\ell}\mathcal{G}(\tau_{\ell}|\frac{\eta_0}{2},\frac{\omega_0}{2}) \\
p(\pi) &= Dir(\pi|\alpha_0)
\end{aligned} \tag{2}
$$

where $Be(x|a,b)$ represents the Beta density function $Be(x|a,b) = x^{a-1}(1-x)^{b-1}/B(a,b)$ where $B(a,b)$ is the beta function, $\mathcal{G}(x|a,b)$ is the gamma distribution, and $Dir(\pi|\alpha_0) = C_{\mathcal{D}}\prod_{k=1}^{K}\pi_k^{\alpha_k^0-1}$ where

$$
C_{\mathcal{D}} = \Gamma\left(\sum_{k=1}^{K}\alpha_k^0\right) / \prod_{k=1}^{K}\Gamma\left(\alpha_k^0\right)
$$

III. Inference

In this section, we derive the VB algorithm to train the model. The auxiliary posterior distributions are factorised as a tree-like structure. Tree-like structural factorisation in VB has been shown to be superior over the full factorisation scheme [26], [27]. Its form is of the following:

$$
\begin{aligned}
&q(\mathbf{z}_n,\mathbf{u}_n,\Phi_n,\pi,\beta,\{\mu_j,\sigma_j\},\{\chi_{\ell},\tau_{\ell}\}) = \\
&\quad q(\mathbf{u}_n|\mathbf{z}_n,\Phi_n)q(\mathbf{z}_n)q(\Phi_n|\mathbf{z}_n)q(\pi)q(\beta) \\
&\quad q\left(\{\mu_j,\sigma_j\}\right)q\left(\{\chi_{\ell},\tau_{\ell}\}\right)
\end{aligned}
$$

Or specifically, due to the conjugate priors we used, it can be seen that

$$
\begin{aligned}
&q(\mathbf{z}_n = j,\mathbf{u}_n,\Phi_n,\pi,\beta,\{\mu_j,\sigma_j\},\{\chi_{\ell},\tau_{\ell}\}) = \\
&\quad q(\mathbf{u}_n|j,\Phi)q(\mathbf{z}_n=j)q(\Phi_n|j)q(\pi)q(\beta) \\
&\quad q\left(\{\mu_j,\sigma_j\}\right)q\left(\{\chi_{\ell},\tau_{\ell}\}\right) \\
&\quad = \Big[\prod_{\ell=1}^{d} q(u_{n\ell}|j,\phi_{n\ell})q(\phi_{n\ell}|j)\Big]q(\mathbf{z}_n=j)q(\pi)\prod_{\ell=1}^{d}q(\beta_{\ell})\cdot \\
&\quad \Big[\prod_{j,\ell}^{K,d} q(\mu_{j\ell})q(\sigma_{j\ell})\Big]\Big[\prod_{\ell=1}^{d} q(\chi_{\ell})q(\tau_{\ell})\Big]
\end{aligned}
$$

The free energy function $\mathcal{F}$ can then be written in Fig. 3. We need to highlight the following expectations, that is:

$$
\begin{aligned}
\Big\langle \log p(\phi_{n\ell}|\beta_{\ell})\Big\rangle_j &= q(\phi_{n\ell}=1|j)\langle\log\beta_{\ell}\rangle + \\
&\quad q(\phi_{n\ell}=0|j)\langle\log(1-\beta_{\ell})\rangle
\end{aligned}
$$

and

$$
\begin{aligned}
\Big\langle \log q(\phi_{n\ell}|j)\Big\rangle_j &= q(\phi_{n\ell}=1|j)\langle\log q(\phi_{n\ell}=1|j)\rangle + \\
&\quad q(\phi_{n\ell}=0|j)\langle\log q(\phi_{n\ell}=0|j)\rangle
\end{aligned}
$$

A. Auxiliary posteriors of the latent variables: the VB-E step

1) $q(\mathbf{u}_n|\mathbf{z}_n,\Phi_n)$: First of all, the free energy associated with the auxiliary posterior $q(\mathbf{u}_n|\mathbf{z}_n,\Phi_n)$ can be read as follows:

$$
\mathcal{F} = \Big\langle \log\left[p(\mathbf{y}_n,h_n)\right] - \log q(\mathbf{u}_n|\mathbf{z}_n,\Phi_n)\Big\rangle_q
$$

According to the KKT condition, and using the Lagrange multiplier, we obtain:

$$
\begin{aligned}
q(\mathbf{u}_n|\mathbf{z}_n,\Phi_n) &\propto e^{\Big\langle \log[p(\mathbf{y}_n|\mathbf{u}_n,\mathbf{z}_n,\Phi_n)p(\mathbf{u}_n|\mathbf{z}_n,\Phi_n)]\Big\rangle} \\
&\propto \prod_{\ell=1}^{d}\exp\Big\langle \log\left[p(y_{n\ell}|u_{n\ell},\mathbf{z}_n,\phi_{n\ell})p(u_{n\ell}|\mathbf{z}_n,\phi_{n\ell})\right]\Big\rangle
\end{aligned}
$$

$$
\begin{aligned}
\mathcal{F} &= \sum_{n,j} q(\mathbf{z}_n = j) \left\{ \sum_{\ell} \Big\langle \log p(y_{n\ell}|u_{n\ell}, \phi_{n\ell}) p(u_{n\ell}|\phi_{n\ell}) \Big\rangle_j \right\} + \sum_{n,j} q(\mathbf{z}_n = j) \left\{ \sum_{\ell} \Big\langle \log p(\phi_{n\ell}|\beta_\ell) \Big\rangle_j \right\} + \\
& \sum_{n,j} q(\mathbf{z}_n = j) \Big\langle \log p(\mathbf{z}_n = j) - \log q(\mathbf{z}_n = j) \Big\rangle + \sum_{j} \Big\langle \log p(\mu_j) + \log p(\sigma_j) - \log q(\mu_j) - \log q(\sigma_j) \Big\rangle + \\
& \Big\langle \log p(\chi) + \log p(\tau) - \log q(\chi) - \log q(\tau) \Big\rangle + \Big\langle \log p(\pi) - \log q(\pi) \Big\rangle + \Big\langle \log p(\beta) - \log q(\beta) \Big\rangle - \\
& \sum_{n,j} q(\mathbf{z}_n = j) \left\{ \sum_{\ell} \Big\langle \log q(u_{n\ell}|j, \phi_{n\ell}) \Big\rangle_j \right\} - \sum_{n,j} q(\mathbf{z}_n = j) \left\{ \sum_{\ell} \Big\langle \log q(\phi_{n\ell}|j) \Big\rangle_j \right\}
\end{aligned}
$$

Fig. 3. The free energy of the proposed model.

This shows that $q(\mathbf{u}_n|\mathbf{z}_n, \Phi_n) = \prod_{\ell=1}^{d} q(u_{n\ell}|\mathbf{z}_n, \phi_{n\ell})$. Through mathematical manipulation, we can obtain:

$$
\begin{aligned}
q(u_{n\ell}|\mathbf{z}_n = j, \phi_{n\ell} = 1) &= \mathcal{G}(u_{n\ell}|\bar{a}_{jn\ell}, \bar{b}_{jn\ell}) \\
q(u_{n\ell}|\mathbf{z}_n = j, \phi_{n\ell} = 0) &= \mathcal{G}(u_{n\ell}|\bar{s}_{n\ell}, \bar{t}_{n\ell})
\end{aligned}
$$

where

$$
\begin{aligned}
\bar{a}_{jn\ell} &= \frac{\nu_{j\ell} + 1}{2}; \bar{s}_{n\ell} = \frac{\gamma_\ell + 1}{2} \\
\bar{b}_{jn\ell} &= \frac{\nu_{j\ell} + \langle (y_{n\ell} - \mu_{j\ell})^2 \sigma_{j\ell} \rangle}{2} \\
\bar{t}_{n\ell} &= \frac{\gamma_\ell + \langle (y_{n\ell} - \chi_\ell)^2 \tau_\ell \rangle}{2}
\end{aligned}
$$

or concisely, we see that

$$
q(u_{n\ell}|\mathbf{z}_n = j, \phi_n) = \mathcal{G}(u_{n\ell}|\bar{a}_{jn\ell}, \bar{b}_{jn\ell})^{\phi_{n\ell}} \mathcal{G}(u_{n\ell}|\bar{s}_{n\ell}, \bar{t}_{n\ell})^{1-\phi_{n\ell}} \tag{3}
$$

Note that in case $\phi_{n\ell} = 0$, the auxiliary distribution of $u_{n\ell}$ is independent of $\mathbf{z}_n$.

Note that $\langle (y_{n\ell} - \mu_{j\ell})^2 \rangle = (y_{n\ell} - \langle \mu_{j\ell} \rangle)^2 + \bar{\sigma}_{j\ell}$ and $\langle (y_{n\ell} - \chi_\ell)^2 \rangle = (y_{n\ell} - \langle \chi_\ell \rangle)^2 + \bar{\psi}_\ell$ where $\bar{\sigma}_{j\ell}$ and $\bar{\psi}_\ell$ are the standard deviations of the posterior $q(\mu_{j\ell})$ and $q(\chi_\ell)$, respectively.

In the sequel, we denote the expectation of $q(u_{n\ell}|\phi_{n\ell} = 1, \mathbf{z}_n = j)$ as $\langle u_{n\ell} \rangle_j^1$ and the expectation of $q(u_{n\ell}|\phi_{n\ell} = 0, \mathbf{z}_n = j)$ as $\langle u_{n\ell} \rangle_j^0$.

2) $q(\Phi_n|\mathbf{z}_n = j)$: If we let

$$
\begin{aligned}
A = \Big[\langle \log \mathcal{N}(y_{n\ell}|\mu_{j\ell}, u_{n\ell}\sigma_{j\ell}) \rangle + \langle \log \mathcal{G}(u_{n\ell}|\frac{\nu_{j\ell}}{2}, \frac{\nu_{j\ell}}{2}) \rangle \Big] + \\
\langle \log \beta_\ell \rangle - \langle \log q(u_{n\ell}|j, \phi_{n\ell} = 1) \rangle
\end{aligned} \tag{4}
$$

and

$$
\begin{aligned}
B = \Big[\langle \log \mathcal{N}(y_{n\ell}|\chi_\ell, u_{n\ell}\tau_\ell) \rangle + \langle \log \mathcal{G}(u_{n\ell}|\frac{\gamma_\ell}{2}, \frac{\gamma_\ell}{2}) \rangle \Big] + \\
\langle \log(1 - \beta_\ell) \rangle - \langle \log q(u_{n\ell}|j, \phi_{n\ell} = 0) \rangle
\end{aligned} \tag{5}
$$

then $q(\phi_{n\ell} = 1|j)$ can be written as:

$$
q(\phi_{n\ell} = 1|j) = \frac{\exp\{A\}}{\exp\{A\} + \exp\{B\}}
$$

and $q(\phi_{n\ell} = 0|j)$ is

$$
q(\phi_{n\ell} = 0|j) = \frac{\exp\{B\}}{\exp\{A\} + \exp\{B\}}
$$

In the sequel, we use $\langle \phi_{n\ell} \rangle_j^0$ and $\langle \phi_{n\ell} \rangle_j^1$ to denote $q(\phi_{n\ell} = 0|j)$ and $q(\phi_{n\ell} = 1|j)$, respectively.

3) $q(\mathbf{z}_n)$: If we define the quantity,

$$
\begin{aligned}
R_{n,j} &= \Big\langle \log \big[p(\mathbf{y}_n|\mathbf{z}_n = j, \mathbf{u}_n, \Phi_n) \big] \Big\rangle + \\
& \Big\langle \log p(\mathbf{u}_n|\mathbf{z}_n = j, \Phi_n) \Big\rangle + \\
& \Big\langle \log p(\mathbf{z}_n = j|\pi) \Big\rangle + \\
& \sum_{\ell} \big[\langle \phi_{n\ell} \rangle_j^1 \langle \log \beta_\ell \rangle + \langle \phi_{n\ell} \rangle_j^0 \langle \log(1 - \beta_\ell) \rangle \big] - \\
& \Big\langle \log \big[q(\mathbf{u}_n|\mathbf{z}_n = j, \Phi_n) \big] \Big\rangle - \\
& \Big\langle \log \big[q(\Phi_n|\mathbf{z}_n = j) \big] \Big\rangle
\end{aligned} \tag{6}
$$

Then the responsibility $q(\mathbf{z}_n = j)$ can be calculated as follows:

$$
q(\mathbf{z}_n = j) = \frac{\exp\{R_{n,j}\}}{\sum_k \exp\{R_{n,k}\}}
$$

In the sequel, we use $\langle \mathbf{z}_n \rangle_j$ to denote $q(\mathbf{z}_n = j)$.

B. Auxiliary posteriors of the parameters: the VB-M step

On the other hand, for the parameters, the posteriors are as follows.

1) $q(\pi)$:

$$
q(\pi) = \mathcal{D}(\pi|\hat{\alpha})
$$

where $\hat{\alpha}_j = \sum_n^N q(\mathbf{z}_n = j) + \alpha_0$ and $\hat{\alpha}_0 = \sum_j \hat{\alpha}_j$ and

$$
\log \pi_j = \Psi(\hat{\alpha}_j) - \Psi(\hat{\alpha}_0)
$$

2) $q(\beta)$: For $q(\beta) = \prod_\ell q(\beta_\ell)$, we have:

$$
q(\beta_\ell) = Be(\beta_\ell|\bar{\kappa}_{1\ell}, \bar{\kappa}_{2\ell})
$$

where

$$
\begin{aligned}
\bar{\kappa}_{1\ell} &= \kappa_1 + \sum_{n,j} q(\phi_{n\ell} = 1|j) \langle \mathbf{z}_n \rangle_j \\
\bar{\kappa}_{2\ell} &= \kappa_2 + \sum_{n,j} q(\phi_{n\ell} = 0|j) \langle \mathbf{z}_n \rangle_j
\end{aligned}
$$

The expectation $\langle \log \beta_\ell \rangle$ and $\langle \log(1 - \beta_\ell) \rangle$ as used in the calculation of $q(\Phi_n|j)$ can be obtained as:

$$\langle \log \beta_\ell \rangle = \psi(\bar{\kappa}_{1\ell}) - \psi(\bar{\kappa}_{1\ell} + \bar{\kappa}_{2\ell})$$
$$\langle \log(1-\beta_\ell) \rangle = \psi(\bar{\kappa}_{2\ell}) - \psi(\bar{\kappa}_{1\ell} + \bar{\kappa}_{2\ell})$$

3) $q(\sigma_j)$: For $q(\sigma_{j\ell})$, we have:

$$q(\sigma_{j\ell}) = \mathcal{G}(\sigma_{j\ell}|\bar{\eta}_{j\ell}, \bar{\omega}_{j\ell})$$

where

$$\bar{\eta}_{j\ell} = \frac{\eta_0 + \sum_n \langle \mathbf{z}_n \rangle_j \langle \phi_{n\ell} \rangle_j^1}{2}$$
$$\bar{\omega}_{j\ell} = \frac{\omega_0 + \sum_n \langle \mathbf{z}_n \rangle_j \langle \phi_{n\ell} \rangle_j^1 \langle (y_{n\ell} - \mu_{j\ell})^2 \rangle \langle u_{n\ell} \rangle_j^1}{2}$$

4) $q(\tau)$: For $q(\tau_\ell)$, we have:

$$q(\tau_\ell) = \mathcal{G}(\tau_\ell|\bar{\psi}_\ell, \bar{\xi}_\ell)$$

where

$$\bar{\psi}_\ell = \frac{\eta_0 + \sum_{n,j} \langle \mathbf{z}_n \rangle_j \langle \phi_{n\ell} \rangle_j^0}{2}$$
$$\bar{\xi}_\ell = \frac{\omega_0 + \sum_{n,j} \langle \mathbf{z}_n \rangle_j \langle \phi_{n\ell} \rangle_j^0 \langle (y_{n\ell} - \chi_\ell)^2 \rangle \langle u_{n\ell} \rangle_j^0}{2}$$

5) $q(\mu_j)$: For $q(\mu_{j\ell})$, we have:

$$q(\mu_{j\ell}) = \mathcal{N}(\mu_{j\ell}|\bar{\mu}_{j\ell}, \bar{\sigma}_{j\ell})$$

where

$$\bar{\sigma}_{j\ell} = \langle \sigma_{j\ell} \rangle \sum_n \langle \mathbf{z}_n \rangle_j \langle \phi_{n\ell} \rangle_j^1 \langle u_{n\ell} \rangle_j^1 + \lambda_0$$
$$\bar{\mu}_{j\ell} = \bar{\sigma}_{j\ell}^{-1} \left(\langle \sigma_{j\ell} \rangle \sum_n \langle \mathbf{z}_n \rangle_j \langle \phi_{n\ell} \rangle_j^1 \langle u_{n\ell} \rangle_j^1 y_{n\ell} + \lambda_0 \mu_0 \right)$$

6) $q(\chi_\ell)$: For $q(\chi_\ell)$, we have:

$$q(\chi_\ell) = \mathcal{N}(\mu_\ell|\bar{\mu}_\ell.\bar{\sigma}_\ell)$$

where

$$\bar{\sigma}_\ell = \langle \tau_\ell \rangle \sum_{n,j} q(\mathbf{z}_n = j) \langle \phi_{n\ell} \rangle_j^0 \langle u_{n\ell} \rangle_j^0 + \lambda_0$$
$$\bar{\mu}_\ell = \bar{\sigma}_\ell^{-1} \left(\langle \tau_\ell \rangle \sum_{n,j} q(\mathbf{z}_n = j) \langle \phi_{n\ell} \rangle_j^0 \langle u_{n\ell} \rangle_j^0 y_{n\ell} + \lambda_0 \mu_0 \right)$$

C. The optimization of the degree of freedom parameters

The degree of freedom $\nu_{j\ell}, 1 \leq j \leq d, \gamma_\ell, 1 \leq \ell \leq K$ can be obtained by solving the following non-linear Eqs. (7) and (8), respectively. In the equations, $\langle \log u_{n\ell} \rangle_j^0$ and $\langle \log u_{n\ell} \rangle_j^1$ denote the expectations of $\log q(u_{n\ell}|\phi_{n\ell} = 0, j)$ and $\log q(u_{n\ell}|\phi_{n\ell} = 1, j)$, respectively.

$$\sum_n \langle \mathbf{z}_n \rangle_j \langle \phi_{n\ell} \rangle_j^1 \Big[1 + \log \frac{\nu_{j\ell}}{2} + \langle \log u_{n\ell} \rangle_j^1 - \langle u_{n\ell} \rangle_j^1 - \psi(\frac{\nu_{j\ell}}{2}) \Big] = 0 \quad (7)$$

$$\sum_{n,j} \langle \mathbf{z}_n \rangle_j \langle \phi_{n\ell} \rangle_j^0 \Big[\langle \log u_{n\ell} - u_{n\ell} \rangle_j^0 + 1 + \log \frac{\gamma_\ell}{2} - \psi(\frac{\gamma_\ell}{2}) \Big] = 0 \quad (8)$$

D. The log-likelihood bound

The optimisation process of the variational Bayes can be monitored by the log-likelihood bound as shown in Fig. 3. Due to the length limit of the paper, we omit the evaluation.

IV. Interpreting the model

A. The outlier detection

Due to the application of Student t-distribution, the developed algorithm is supposed to deal with outliers. To detect outliers, the expectation of the $\mathbf{u_n}$ is applied as the outlier criterion. Note that

$$q(\mathbf{u}_n) = \sum_j q(\mathbf{z}_n = j) \int q(\mathbf{u}_n|\mathbf{z}_n = j, \Phi_n) q(\Phi_n|\mathbf{z}_n = j) d\Phi_n$$

The expectation of $q(\mathbf{u}_n)$ can then be calculated as

$$\langle \mathbf{u}_n \rangle = \sum_j \langle z_{nj} \rangle \sum_\ell \left[\langle \phi_{n\ell} \rangle_j^1 \frac{\bar{a}_{jn\ell}}{\bar{b}_{jn\ell}} + \langle \phi_{n\ell} \rangle_j^0 \frac{\bar{s}_{n\ell}}{\bar{t}_{n\ell}} \right]$$

The smaller the value of $\langle \mathbf{u}_n \rangle$ of the data $\mathbf{y}_n$, the higher chance that the data point is an outlier.

B. Selecting proper partition

An important issue is to decide the number of clusters. In the Bayesian framework, the optimal number of clusters can be obtained automatically. Given a large number K, during the optimisation process, some clusters that do not have enough supportive data points will be pruned.

C. The Feature Saliencies

The saliencies of each feature can be measured by the expectation of the variable β, which can be obtained as follows:

$$\langle \beta_\ell \rangle = \frac{\bar{\kappa}_{1\ell}}{\bar{\kappa}_{1\ell} + \bar{\kappa}_{2\ell}}$$

The higher the value, the more importance of feature.

V. Experimental Study

In this section, we studied the developed algorithm in controlled experiments. We generate synthetic data sets to demonstrate the performance of the developed algorithm in terms of feature selection, clustering performance and the outlier detection rate. The developed algorithm was compared with the semi-Bayesian feature selection algorithm [22], called varFnMS. varFnMS is based on variational Bayes algorithm and the associated model is similar to the hierarchical latent variable model proposed in the paper. The difference is that a finite mixture of Gaussian is adopted in the varFnMS model, and a full-factorised variational Bayes is applied.

To demonstrate the developed algorithm, synthetic data are generated similar to that in [23]. That is, a set of data points are sampled from four well-separated bi-variate clusters. The centres and covariance matries are $[0\ \ 3]^T, [1\ \ 9]^T, [6\ \ 4]^T, [7\ \ 10]^T$ and $\mathbf{I}$. Eight "noisy" features (sampled from $\mathcal{N}(0,1)$ density) are then appended to this data, resulting in a 10-dimensional patterns. The synthetic data set consists of 800 data points

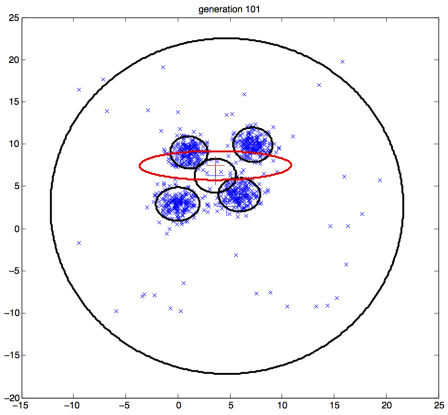


Fig. 5. A typical run of the semi-Bayesian feature selection algorithm. The plot shows the data on the first two coordinates.

from the four Gaussians, and a set of outliers are sampled from the range $[-10\ 30]^{10}$. To test the performance of the algorithm on outlier detection, various percentages of outliers are specified in the following experiments.

We run the proposed algorithm 10 times, each initialised with $K = 10$. The K-means clustering algorithm is used to initialise the mean of the posterior $q(\mu_j)$, and the feature saliency variable is initialised to be 0.5. The hyper parameters including $\kappa_1, \kappa_2, \lambda_0, \alpha_0$ are set to be 10^{-5}, and m_0 is set to be the mean of all data. The algorithm terminates when the difference of log-likelihood bound is less than 10^{-7}.

Fig. 4 shows a typical run of the developed algorithm, while the estimation of the first two-dimension of the parameters is shown at certain iterations. In the run, we initialise the number of components to be 10. From the figure, we can see that the developed algorithm estimates the parameters successfully. Moreover, it can be seen that it automatically prunes some unnecessary components. The last plot is the data shown for the third and fourth variables, and the red circle demonstrates the mean of the posterior $q(\mu_0)$. Fig. 5 shows the results obtained by the semi-Bayesian feature selection algorithm on the variables Y_1 and Y_2. From the figure, it can be seen that the algorithm is not able to find the true cluster centres.

The AUC (area under curve) values obtained through the ROC analysis [28] is obtained to measure the performance of the ability of outlier detection. The results are summarised in Fig. 6. In Fig. 6, the AUC values obtained for the different percentages of outliers are shown. From the figure, it can observed that the developed algorithm can successfully find the outliers. On the other hand, the semi-Bayesian algorithm is not able to find outliers.

Fig. 7 shows the feature saliencies associated with the features retrieved by the proposed algorithm and the semi-Bayesian algorithm. From the figure, we can see that the saliencies of the noisy variables $(Y_3 - Y_8)$ obtained by the proposed algorithm is closer to the ground true than that of the semi-Bayesian algorithm.

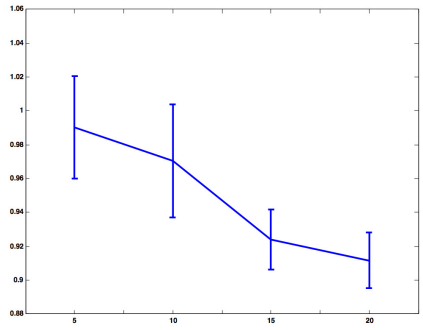
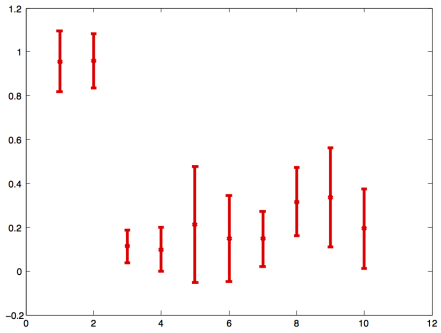
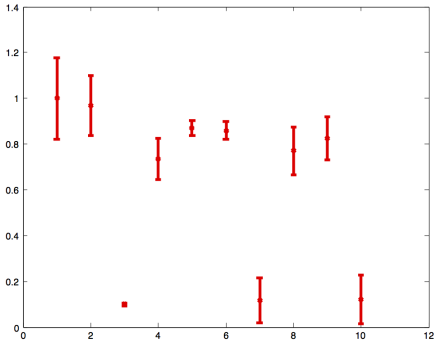


Fig. 6. The AUC values obtained by the developed algorithm for different percentages of outliers with standard deviations shown.



(a)



(b)

Fig. 7. The feature saliencies for the synthetic data with 5% percentage of outliers by a) the proposed algorithm; b) the semi-Bayesian algorithm. The standard deviations of the 10 runs were also shown in the plots.

VI. Conclusion

In this paper, we developed a hierarchical latent variable model for dealing with unsupervised feature selection. The model is also proposed to deal with outliers. The Bayesian treatment of the model makes model selection possible. The Bayesian variational framework was used for the inference. A tree-structured factorisation of the latent variable models was used in the variational framework. The developed algorithm was compared in controlled experiments with the VB algorithm developed for the semi-Bayesian mixture model [22].

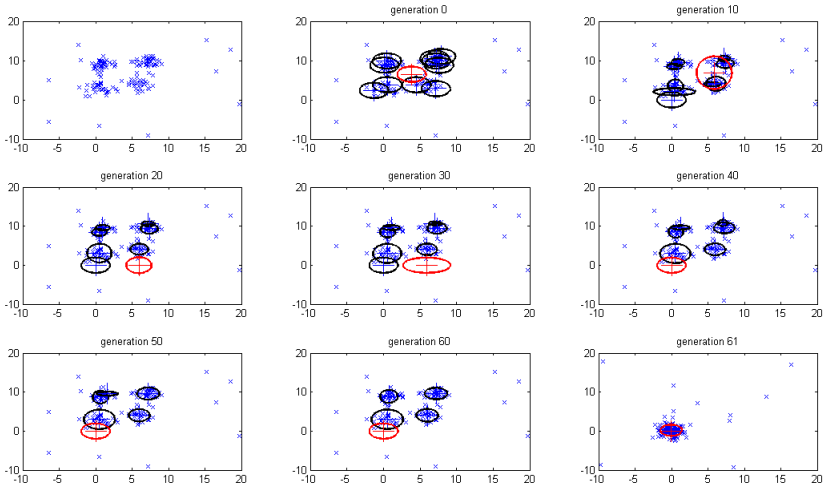


Fig. 4. A typical run of the developed algorithm on the example dataset, while the black circles represent $q(\mu_1|k)$, and the red circles denote $q(\mu_0)$. The first plot shows the dataset on the first two dimensions, while the last plot shows the estimation of the third and fourth dimension.

The experiments have shown that the developed algorithm outperformed the semi-Bayesian algorithm in terms of outlier detection, feature selection and the clustering. This also showed that the proposed model is better than the corresponding Bayesian mixture model.

ACKNOWLEDGEMENT

This work was supported by NSFC grants 61273313.

REFERENCES

[1] J. Dy and C. Brodley, "Feature selection for unsupervised learning," *Journal of Machine Learning Research*, vol. 5, pp. 845–889, 2004.

[2] Y. Saeys, I. Inza, and P. Larranaga, "A review of feature selection techniques in bioinformatics," *Bioinformatics*, vol. 23, no. 19, pp. 2507–2517, 2007.

[3] A. Arauzo-Azofra, J. Benitez, and J. Castro, "Consistency measures for feature selection," *Journal of Intelligent Information Systems*, 2007.

[4] M. Sebban and R. Nock, "A hybrid filter/wrapper approach of feature selection using information theory," *Pattern Recognition*, vol. 35, pp. 835–846, 2002.

[5] D. Francois, F. Rossi, V. Wertz, and M. Verleysen, "Resampling methods for parameter-free and robust feature selection with mutual information," *Neurocomputing*, vol. 70, no. 7-9, pp. 1276–1288, 2007.

[6] M. Hall, "Correlation-based feature selection for machine learning," The University of Waikato, 1999.

[7] M. Houle and N. Grira, "A correlation-based model for unsupervised feature selection," in *Proceedings of the 16th ACM Conference on Information and Knowledge Management*, 2007.

[8] H. Peng, F. Long, and C. Ding, "Feature selection based on mutual information: Criteria of max-dependency, max-relevance and min-redundancy," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 27, no. 8, pp. 1226–1238, Aug. 2005.

[9] L. Yu and H. Liu, "Feature selection for high-dimensional data: A fast correlation-based filter solution," in *Proceedings of the 20th International Conference on Machine Learning*, 2003.

[10] Q. Gu and J. Zhou, "Local learning regularized nonnegative matrix factorization," in *Proceedings of the 21st International Joint Conference on Artificial Intelligence*, 2009.

[11] Q. Gu, Z. Li, and J. Han, "Joint feature selection and subspace learning," in *Proceedings of the 22nd International Joint Conference on Artificial Intelligence*, 2011.

[12] D. Kong, C. Ding, and H. Huang, "Robust nonnegative matrix factorization using l_21-norm," in *Proceedings of the 20-th ACM International Conference on Information and Knowledge Management*. ACM, 2011, pp. 673–682.

[13] H. Zeng and Y.-M. Cheung, "Feature selection and kernel learning for local learning-based clustering," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 33, no. 8, pp. 1532–1547, Aug. 2011.

[14] A. Ng, "Feature selection, l_1 vs. l_2 regularization, and rotational invariance," in *Proceedings of the 21st International Conference on Machine Learning*, 2004.

[15] F. Nie, H. Huang, X. Cai, and C. Ding, "Efficient and robust feature selection via joint l_12-norms minimization," in *Advances in Neural Information Processing Systems*, vol. 23, 2010, pp. 1813–1821.

[16] M. Qian and C. Zhai, "Robust unsupervised feature selection," in *Proceedings of the 23rd International Joint Conference on Artificial Intelligence*, 2013.

[17] Y. Yang, H. Shen, Z. Ma, and Z. Huang, "L_21-norm regularized discriminative feature selection for unsupervised learning," in *Proceedings of the 22nd International Joint Conference on Artificial Intelligence*, 2011.

[18] Z. Li, Y. Yang, J. Liu, X. Zhou, and H. Lu, "Unsupervised feature selection using nonnegative spectral analysis," in *Proceedings of the 26-th AAAI Conference on Artificial Intelligence*, 2012.

[19] M. Wu and B. Scholkopf, "A local learning approach for clustering," in *Advances in Neural Information Processing Systems*, vol. 19, 2007, p. 1529.

[20] Z. Zhao and H. Liu, "Spectral feature selection for supervised and unsupervised learning," in *Proceedings of the 24-th International Conference on Machine Learning*, 2007, pp. 1151–1157.

[21] ——, "Semi-supervised feature selection via spectral analysis," in *Proceedings of the 7th SIAM International Conference on Data Mining*, 2007.

[22] C. Constantinopoulos, M. Titsias, and A. Likas, "Bayesian feature and model selection for Gaussian mixture models," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 28, no. 6, pp. 1013–1018, Jun. 2006.

[23] M. Law, M. Figueiredo, and A. Jain, "Simultaneous feature selection and clustering using mixture models," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 26, no. 9, pp. 1154–1166, Sep. 2004.

[24] V. Roth and L. Tilman, "Feature selection in clustering problems," in *Advances in Neural Information Processing Systems*, 2003.

[25] S.-H. Yang and B.-G. Hu, "Feature selection by nonparametric bayes error minimization," in *PAKDD*, ser. LNAI, T. W. et al., Ed., vol. 5012. Springer-Verlag Berlin Heidelberg, 2008, pp. 417–428.

[26] J. Sun and A. Kaban, "A fast algorithm for robust mixtures in the presence of measurement errors," *IEEE Trans. on Neural Networks*, vol. 21, no. 8, pp. 1206–1220, 2010.

[27] J. Sun, J. Garibaldi, and K. Kenobi, "Robust Bayesian clustering fro datasets with repeated measures," *IEEE/ACM Trans. on Computational Biology and Bioinformatics*, vol. 9, no. 5, pp. 1504–1514, Sep. 2012.

[28] T. Fawcett, "ROC graphs: Notes and practical considerations for researchers," HP Labs, Tech. Rep. HPL-2003-4, 2004.