2012 IEEE INTERNATIONAL WORKSHOP ON MACHINE LEARNING FOR SIGNAL PROCESSING, SEPT. 23–26, 2012, SANTANDER, SPAIN

UNSUPERVISED FEATURE SELECTION BASED ON NON-PARAMETRIC MUTUAL INFORMATION

Lev Faivishevsky and Jacob Goldberger

Engineering Faculty, Bar-Ilan University, Ramat-Gan 52900, Israel

levtemp@gmail.com goldbej@eng.biu.ac.il

ABSTRACT

We present a novel filter approach to unsupervised feature selection based on the mutual information estimation between features. Our feature selection approach does not impose a parametric model on the data and no clustering structure is estimated. Instead, to measure the statistical dependence between features, we employ a mutual information criterion, which is computed by using a non-parametric method, and remove uncorrelated features. Numerical experiments on synthetic and real world tasks show that the performance of our algorithm is comparable to previously suggested state-of-the-art methods.

Index Terms— feature selection, mutual information

1. INTRODUCTION

Feature selection aims at selecting the most relevant feature subset for more efficient training and improved accuracy. Feature selection methods can be classified into supervised and unsupervised methods based on whether label information is available or not. Supervised feature selection methods usually evaluate the importance of a feature in terms of its statistical dependence with the class label. There is usually no shortage of unlabeled data but labels are expensive. Hence, it is of great significance to develop unsupervised feature selection algorithms that can make use of unlabeled data points. In this paper, we consider the problem of selecting features in unsupervised learning scenarios, which is a much harder problem due to the absence of class labels that can guide the search for relevant information.

Recently, there have been several attempts to devise models to deal with feature selection in unsupervised learning tasks, most of them are wrapper methods that combine the feature selection task with clustering, e.g., [1, 2, 3, 4, 5, 6]. The Laplacian score (LS) approach [7, 8, 9] is a filter based approach which is based on the observation that in many real world classification problems, data from the same class are often close to each other in the feature space. The importance of a feature is evaluated by its power of locality preservation. The Multi-Cluster Feature Selection (MCFS) [10] selects features that best preserve the multi-cluster structure of the data. The corresponding optimization problem can be efficiently solved since it only involves a sparse eigen-problem and an L_1-regularized least squares problem. The Q-α [11] algorithm optimizes over a least-squares criterion function which measures the clusterability of the input data points projected onto the selected coordinates.

In this paper we propose a novel mutual information-based approach for the task of unsupervised feature selection. Our method is based on the observation that informative features that are related to the learning task, are expected to be highly correlated with the rest of the features whereas non-informative features are less correlated. In the supervised case, given the labels, a feature might be independent on the other features and still contribute important information. Actually, the Naive Bayes classifier is explicitly based on the assumption of conditional feature independence. When the labels are given, this independence assumption is naive but makes sense and can be adapted to obtain a simple supervised algorithm. In the unsupervised case, this assumption is unrealistic since all the relevant features are correlated with the unknown data labels and hence correlated with each other. This distinction between unsupervised and supervised setups is closely related to the difference between the information theory concepts of mutual information and conditional mutual information [12] (conditioned on the labels). Our feature selection approach does not impose a parametric model on the data and no clustering structure is assumed. Instead, to measure the statistical dependence between features, we employ a mutual information criterion which is computed by using a non-parametric method.

The rest of paper is organized as follows. In Section 2 we formally defined the unsupervised feature selection problem and Section 3 describes the proposed method. In Section 4 we provide extensive experiments on real world data that validate our assumption that relevant features are indeed statistically dependent.

2. UNSUPERVISED FEATURE SELECTION

The general form of unsupervised feature selection is the following. Given a set of points $\{x_1, x_2, ..., x_n\}, x_i \in R^m$, find a feature subset with size $d < m$ which contains the most informative features. In other words, in the data matrix X of size $m \times n$ we would like to select d rows to obtain a submatrix F of size $d \times n$. We use the notation $f_i \in R^n$ to denote the i-th feature vector. f_i is the i-th row of data matrix X, i.e.,

$$X = (x_1, \ldots, x_n) = \begin{pmatrix} f_1 \\ \vdots \\ f_m \end{pmatrix}$$

Abusing notation, we also use f_i to denote the scalar random variable associated with the i-th feature data.

In this study we concentrate on information theoretic formulation of the feature selection problem. In supervised learning we are given a label c_i for each data point x_i. Such labels form a new discrete random variable C. Considering data points and labels as samples from a joint distribution $p(x, c)$, we can define the mutual information between data set X and label set C:

$$I(C; X) = \sum_c \sum_x p(x, c) \log(\frac{p(x|c)}{p(x)})$$

978-1-4673-1026-0/12/$31.00

In case the features are discrete, we can estimate he joint distribution from the data:

$$p(x,c) = \frac{1}{n}|\{i|x = x_i, c = c_i\}|$$

In this study we concentrate on the more frequent case where the features are continuous. In this case it is not obvious how to estimate the mutual information $I(C;X)$ from the data.

Utilizing the mutual information criterion, we can define a supervised feature selection method that chooses feature subset such that the data points represented by those features have the maximal mutual information with the labels [13]:

$$F^* = \arg\max_F I(C;F) \tag{1}$$

where the maximization is done over all the subsets F of d features. Even in this relatively simple case it is not straightforward how to compute $I(C;F)$ and how to carry out the optimization. In our case of unsupervised learning no labels are available. They could be estimated in a pre-processing step of clustering that yields an estimated label-set $\hat{C}$. We can also combine clustering with feature selection and for each feature subset choose the best clustering that can be obtained using only this feature subset, i.e.,

$$F^* = \arg\max_F \max_C I(C;F) \tag{2}$$

where C goes over all the possible data clusterings. However, this approach would lead to a wrapper method for feature selection. In wrapper techniques the feature selection is performed by utilizing a specific clustering algorithm that optimizes its target function for a possible selection of features. The drawbacks of wrapper methods are lack of generality due to dependency on a specific clustering algorithm and also their high computational complexity.

In this paper we focus on a filter approach for unsupervised feature selection. In filter methods feature selection is done by optimization of a numerical criterion based on the input data which thus avoids the need for a step of explicitly applying a clustering algorithm. This means that we cannot utilize the mutual information between features and data labels, as in (1), because we neither have the original labels of data points nor their estimated values.

We would like to keep only the most informative features to improve performance and efficiency. In real world tasks the least informative features usually emerge due to inevitable noise in the feature computation. We expect that the best features describe the actual problem whereas the worst features are due to numerous noise factors. Hence, in the unsupervised case, we expect the good features to be related to the task and therefore to be statistically dependent. As a consequence a good feature is expected to be dependent on the rest of features. By contrast, a bad feature is a description of a noise factor; hence its dependency on the rest of features is small. These considerations lead to a feature selection approach based on dependency evaluation between features.

To motivate our assumption that non-important features are less dependent on the rest of the features we formally validate it for the following simple model corresponding to the naive Bayes classifier. Let an unobserved label c be a binary random variable and let $S = \{f_1, ..., f_m\}$ be a set of conditionally independent binary variables such that $p(f_i = c) = p_i \in [0.5, 1]$. The probabilistic model, therefore, is:

$$\log p(S|c) = \sum_{i=1}^{m} (\log(p_i) 1_{\{f_i = c\}} + \log(1-p_i) 1_{\{f_i \neq c\}}).$$

Denote the variable set S excluding f_i by $f_{\setminus i}$ and the variables set f excluding f_i, f_j by $f_{\setminus ij}$.

If $p_i > p_j$, then $I(f_i;c) > I(f_j;c)$ and therefore the feature f_i is more informative than f_j. In the supervised case the features are independent, i.e. $I(f_i;f_j|c) = 0$. We prove next that in the unsupervised case if $p_i > p_j$ then f_i is more correlated to the rest of the features than f_j, i.e. $I(f_i;f_{\setminus i}) > I(f_j;f_{\setminus j})$. The chain rule for mutual information [12] yields:

$$\begin{aligned} I(f_i;f_{\setminus i}) &= I(f_i;f_j) + I(f_i;f_{\setminus ij}|f_j) \\ &= I(f_i;f_j) + H(f_{\setminus ij}|f_j) - H(f_{\setminus ij}|f_i,f_j) \end{aligned}$$

and similarly

$$I(f_j;f_{\setminus j}) = I(f_i;f_j) + H(f_{\setminus ij}|f_i) - H(f_{\setminus ij}|f_i,f_j)$$

Hence,

$$I(f_i;f_{\setminus i}) - I(f_j;f_{\setminus j}) = H(f_{\setminus ij}|f_j) - H(f_{\setminus ij}|f_i) \tag{3}$$

We can simulate the creation of the feature f_j given the label c as a two step process. In the first step we sample a binary r.v. z such that $p(z = c) = p_i$. In the second step $f_j|z$ is sampled such that $p(f_j = z) = \frac{p_i + p_j - 1}{p_i + p_i - 1}$. It can be easily verified that we obtain exactly the same model and $p(f_j = c) = p_j$.

The two Markov chains $f_i \to c \to f_{\setminus ij}$ and $z \to c \to f_{\setminus ij}$ have the same joint distribution, therefore

$$H(f_{\setminus ij}|z) = H(f_{\setminus ij}|f_i). \tag{4}$$

Applying the Data Processing Lemma [12] to the Markov chain $f_j \to z \to c \to f_{\setminus ij}$, we obtain that

$$H(f_{\setminus ij}|f_j) > H(f_{\setminus ij}|z). \tag{5}$$

Combining (3), (4) and (5), we obtain that the more informative feature f_i is more correlated to the rest of the features, i.e., $I(f_i;f_{\setminus i}) > I(f_j;f_{\setminus j})$.

Note that we do not prove and we do not even claim that always the most relevant features are also the most statistically dependent on the other features. Counterexamples can be easily made by constructing features that will not help learning. We argue, however, that in many situations this is indeed the case. Here we suggest an intuitive feature selection criterion that can be efficiently computed and is shown to be useful in many cases.

3. NON-PARAMETRIC MUTUAL INFORMATION

Utilizing our assumption that less informative and noisy features are less correlated with the rest of the features, we propose an algorithm that selects the most important features. We measure the importance of feature f by directly evaluating the dependency between it and the rest of the features. The only question that needs to be answered to fully specify the algorithm is how to measure dependency between features. Mutual information [12] is a natural measure of dependency between random variables. Mutual information is always nonnegative and equals to zero if and only if the variables are independent and it is scale invariant. Moreover, mutual information can be measured between variables of different dimensions, one of them or both may be continuous, discrete or categorical.

Denote the set of all features by $S = \{f_1, ..., f_m\}$. We also use S to denote the multivariate random variable that was used to sample the data points. For a given feature f_i, denote the set $S \setminus \{f_i\}$ by $f_{\setminus i}$. There are several ways to compute the mutual information $I(f_i;f_{\setminus i})$

between a feature f_i and the rest of the features $f_{\setminus i}$. If the feature-set joint probability density is known then the mutual information can be computed through the information theoretic identity:

$$I(f_i; f_{\setminus i}) = h(f_i) + h(f_{\setminus i}) - h(S) \tag{6}$$

where $h(\cdot)$ is the differential entropy [12]:

$$h(g(x)) = -\int g(x) \log g(x) dx$$

For example if all features are jointly multivariate Gaussian such that $\text{var}(S) = \Sigma$ and $\text{var}(f_{\setminus i}) = \Sigma_i$ then [12]:

$$I(f_i; f_{\setminus i}) = -\frac{1}{2} \log \left(\frac{|\Sigma|}{\text{var}(f_i)|\Sigma_i|} \right). \tag{7}$$

Denote the variance of the optimal linear estimator of f_i given $f_{\setminus i}$ by $\text{var}(f_i|f_{\setminus i})$ [14]. If f_i is a deterministic linear function of $f_{\setminus i}$ then $\text{var}(f_i|f_{\setminus i}) = 0$. In the general case:

$$0 \leq \frac{\text{var}(f_i) - \text{var}(f_i|f_{\setminus i})}{\text{var}(f_i)} \leq 1$$

This quotient (which is the square of the correlation coefficient between f_i and $f_{\setminus i}$), is a measure of how well f_i is linearly approximated by $f_{\setminus i}$. It indicates how much of the uncertainty in f_i is linearly explained by $f_{\setminus i}$. Applying the block-matrix determinant formula, it can be easily verified that:

$$\frac{|\Sigma|}{\text{var}(f_i)|\Sigma_i|} = \frac{|\Sigma_f|\text{var}(f_i|f_{\setminus i})}{\text{var}(f_i)|\Sigma_i|} = \frac{\text{var}(f_i|f_{\setminus i})}{\text{var}(f_i)}. \tag{8}$$

Substituting (8) in (7), we obtain that in the Gaussian case $I(f_i; f_{\setminus i})$ measures the linear correlation between f_i and the rest of the features, e.g., if f_i is a linear function of $f_{\setminus i}$ then $I(f_i; f_{\setminus i}) = \infty$. However, assuming that the data distribution is Gaussian is not always a good choice since by utilizing a Gaussian distribution to describe the density we implicitly assume a unimodal blob type shape which is not always the case. Also, even if f_i and $f_{\setminus i}$ are dependent, this dependency is not necessarily linear.

In a more general case where the joint distribution of features is not known the mutual information should be estimated numerically. Classical methods for estimating the mutual information $I(f_i; f_{\setminus i})$ require the estimation of the joint probability density function of $(f_i, f_{\setminus i})$. This estimation must be carried out on the given dataset $\{x_1, ..., x_n\}$. Histogram- and kernel-based (Parzen windows) pdf estimations are among the most commonly used methods [13]. Their use is usually restricted to one- or two-dimensional probability density functions (i.e. pdf of one or two variables). However, for high-dimensional variables histogram- and kernel-based estimators suffer dramatically from the curse of dimensionality; in other words, the number of samples needed to estimate the pdf grows exponentially with the number of variables. An additional difficulty in kernel based estimation lies in the choice of kernel width.

Other methods that are used to estimate the mutual information are based on k-nearest neighbor statistics (see e.g. [15]). A nice property of these estimators is that they can be easily utilized for high dimensional random vectors and no parameters need to be predefined or separately tuned for each clustering problem (other than determining the value of k). There are a number of non-parametric techniques for (differential) entropy estimation, based on

Input: Data vectors $\{x_1, x_2, ..., x_n\} \subset R^m$, required number of features $d < m$.

Output: Selected features $\{f_1^*, f_2^*, ..., f_d^*\}$

Method:

- For each feature $i = 1, ..., m$ compute the score function $\text{score}_{MI}(f_i)$ between the feature f_i and the rest of the features using (11).
- Choose the d features that achieve the highest score.

Fig. 1. Unsupervised Feature Selection based on the Mutual Information algorithm (UFSMI).

the samples $x_1, ..., x_n \in R^m$, which are all variants of the following nearest-neighbor based estimator [16]:

$$H_1 = \frac{m}{n} \sum_{i=1}^{n} \min_{j \neq i} (\log \|x_i - x_j\|) + \psi(n) - \psi(1) + \log(c_m) \tag{9}$$

where $\psi(\cdot)$ is the digamma function (the logarithmic derivative of the gamma function) and c_m is the volume of the m-dimensional unit ball. A more general version is based on k-th nearest neighbors:

$$H_k = \frac{m}{n} \sum_{i=1}^{n} \log \epsilon_{ik} + \psi(n) - \psi(k) + \log(c_m) \tag{10}$$

where ϵ_{ik} is the Euclidean distance from x_i to its k-th nearest neighbor. The H_k entropy estimator is consistent in the sense that both the bias and the variance vanish as the sample sizes increase. The consistency of the 1-NN estimator was proven in [16] and the consistency of the general k-NN version was shown in [17].

In the information theoretic identity described in (6), the joint entropy $h(S)$ corresponds to the entropy of all features together and therefore does not depend on the specific choice of a feature f. This implies that the term $h(S)$ can be eliminated from the computation of the mutual information estimator. Plugging the entropy estimators of (10) for the one-dimensional random variable f and the $(m-1)$-dimensional random vector $f_{\setminus i}$ into (6) we obtain the score function of the feature f by the following formula:

$$\text{score}_{MI}(f) = \frac{1}{n} \sum_{i=1}^{n} \log \epsilon_{ik} + \frac{m-1}{n} \sum_{i=1}^{n} \log \delta_{ik} \tag{11}$$

where ϵ_{ik} is the Euclidean distance from point x_i to its k-th nearest neighbor in the one-dimensional subspace, corresponding to feature f, and δ_{ik} is the Euclidean distance from point x_i to its k-th nearest neighbor in the $(m-1)$-dimensional subspace, induced by features $f_{\setminus i}$. After computing the mutual information score $\text{score}_{MI}(f)$ for all the features, we select the d features that obtained the highest scores.

From the point of view of computational complexity the proposed algorithm involves m estimations of the mutual information by the score function $\text{score}_{MI}(f)$ (11). In each such estimation the k-th nearest neighbor should be evaluated for each of n data points. In the naive approach the nearest neighbor computation is done by estimating all pairwise distances between data points, which requires $O(mn^2)$. The overall complexity of the proposed algorithm is then $O(m^2n^2)$. As an alternative, approximate nearest neighbor (ANN) computation techniques can be implemented in the mutual information estimation, see e.g. [18]. These methods make it possible to reduce the complexity of the nearest neighbor estimation to

$O(n(\log n + \epsilon^{-m}))$, where ϵ is the accuracy of the approximation, see e.g. [19]. Therefore for tasks that includie a moderate number of dimensions and, possibly, a lot of data points the overall complexity of the proposed feature selection algorithm can be reduced to $O(m^2 n \log n)$.

In the proposed method we simultaneously select the d features with the highest score. Alternatively we could use a backward selection scheme [20] that removes the worst feature at each step until the required number of remaining features is achieved. Backward selection is compatible with our method, since in every step we remove the worst feature and keep the relevant features. Hence, it is more likely that a good feature would be correlated with the remaining features. However, choosing all the features in a single round of computing mutual-information scores is much faster and we have found empirically that there is no significant difference between our method and a backward selection scheme. Note that utilizing forward selection schemes, that pick the best feature at each step and next finding the best feature from the remaining features does not make sense in our approach since we avoid using the most relevant features when computing the mutual-information score to select the next feature. The algorithm that we dub Unsupervised Feature Selection based on Mutual Information (UFSMI) is summarized in Fig. 1.

4. EXPERIMENTAL RESULTS

4.1. Synthetic data

First we conducted a synthetic data experiment to illustrate the essence of the proposed method. To demonstrate the UFSMI method we simulated a three-dimensional two cluster dataset by uniform sampling vectors $f = (f_1, f_2, f_3)$ from a 3-D unit cube, satisfying either $f_1 + f_2 + f_3 \leq 1$ or $f_1 + f_2 + f_3 \geq 2$. The data is trivially linearly separated into two disjoint groups. In this example all features are equivalently relevant and the dataset is perfectly separable into two clusters. Then we added Gaussian noise to the first feature, f_1, of each data point . After this noise addition the dataset was no longer perfectly separable. Note that class labels are used to create the data (and to measure the feature selection quality) but are assumed to be unknown in the feature selection process. In this case an unsupervised feature selection applied to the noisy 3-D set should remove the first feature and keep the second and the third ones.

To assess the performance of the algorithm UFSMI we ran it on the noisy dataset with different noise standard deviations values (0, 0.02, 0.04, 0.05, 0.1). For each standard deviation we repeated the experiment 100 times. We measured the values of the UFSMI score function (5) for each of the three features, i.e., we estimated the mutual information between each feature and the remaining two features. The results (mean and std) are shown in Fig. 2(a). As can be expected, the higher level of noise led to smaller average values of the score function, because the addition of noise reduces mutual information between features. In addition, the score of the corrupted first feature was smaller than the scores of the other two features and the score difference of f_1 monotonically increased with the noise level. That is a desired property of the feature selection algorithm UFSMI, which aimed at removing noisy features.

Finally we compared the performance of the UFSMI with two state-of-the-art unsupervised feature selection methods, Laplacian Score (LS) [7] and Multi Cluster Feature Selection (MCFS) [10]. The feature selection results for each method were produced by a publicly available implementations. We measured the probability of correct feature selection, which was evaluated as the fraction of cases in which an algorithm removed the noisy first feature, see Fig. 2(b). The performance of the proposed algorithm improved with the growth of the noise in the first feature in contrast to the other two algorithms that did not manage to select the feature correctly in this case. Moreover, their performance deteriorated with the increase in noise energy. The LS method fails because it selects a feature if it is correlated with the nearest-neighbor graph that is built in a preprocessing step. As the variance of the noise increases, the graph tends to represent the noise and therefore the noisy feature is most correlated with the graph. The MCFS is also based on the nearest-neighbor graph.

4.2. UCI datasets

We evaluated the performance of the proposed algorithm on standard real world datasets from the UCI Machine Learning Repository [21]. We used the Abalone, Segmentation, Vowel and Yeast datasets. The Abalone dataset is composed of 3 classes with total 4177 instances. Each instance is given by 8 attributes and predicts the age of abalone from physical measurements. The Segmentation dataset contains 7 classes with total 2310 instances. Each instance is presented by 19 features, it corresponds to a 3×3 region of one of 7 outdoor images. The Yeast dataset contains 10 classes with total 1462 instances. Each instance is composed of 8 predictive attributes and defines a localization site. The vowel dataset records 640 time series of 12 LPC coefficients taken from nine male speakers.

In each dataset we randomly picked 90% of the data and performed feature selection with different numbers of selected features. To assess the quality of the feature selection we performed k-means clustering, with the best score out of 10 random restarts. For each dataset we computed the average performance for 10 repetitions in terms of the Rand index score [22] of the resulting clustering assignment. The Rand score is standard non-parametric measure of clustering quality. Let C_1 and C_2 be two clusterings of the same set (not necessarily with the same number of clusters). Then:

$$\text{Rand Score}(C_1, C_2) = \frac{2(n_{diff} + n_{same})}{n(n-1)}$$

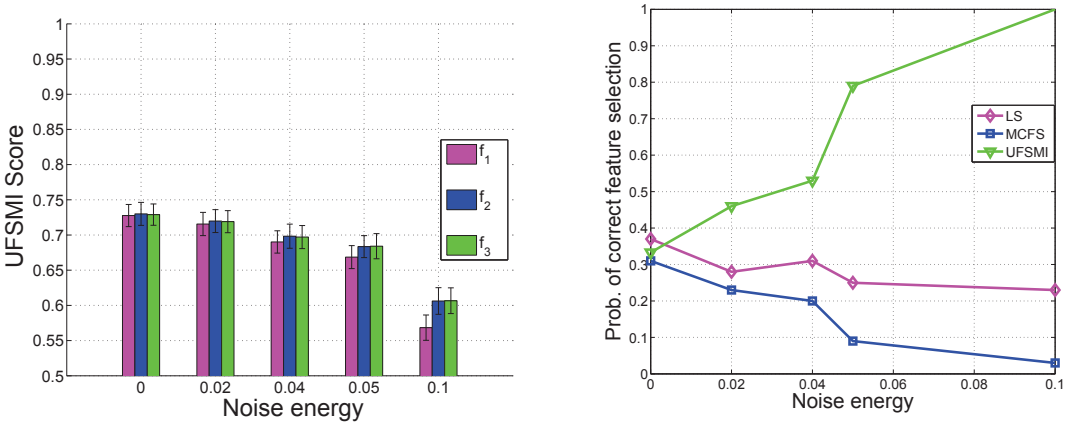
where n is the dataset size, n_{diff} is the number of pairs of points that belong to different clusters in both C_1 and C_2 and n_{same} is the number of pairs of points that belong to the same cluster in both C_1 and C_2.

We compared our method (UFSMI) with LS [7] and MCFS [10], see Fig. 3. UFSMI performed as well as or better than these methods.

4.3. Face clustering on the PIE dataset

The face clustering problem constitutes a common test for unsupervised feature selection methods, see e.g. [7]. We used the CMU PIE face database in this experiment. It contains 68 subjects with 41,368 face images in total. Preprocessing to locate the faces was applied. The original images were normalized (in scale and orientation) such that the two eyes were aligned in the same position. Then, the facial areas were cropped to the final images for matching. The size of each cropped image was 32×32 pixels, with 256 grey levels per pixel. Thus, each image was represented by a 1024-dimensional vector. No further preprocessing was done. In this experiment, we fixed the pose and expression. Thus, for each subject, we obtained 24 images under different lighting conditions.

Each time, 10 classes were randomly selected from the face database. This process was repeated 10 times and the average per-



(a) (b)

Fig. 2. Analysis of feature selection performance on the synthetic 3-D example. (a) UFSMI score function values for feature discrimination with different levels of noise. (b) Probability of correct feature selection for different levels of noise. Statistics are shown for 100 repetitions.

formance was computed. As above, for each test the three algorithms, LS, MCFS and UFSMI were used to select the features. The k-means clustering algorithms was then performed in the selected feature subspace. The k-means was repeated 10 times with different initializations and the best result in terms of the objective function of k-means was recorded. The performance was measured using the Rand score of the resulting clustering assignment. The results are shown in Fig. 4. In the experiments the proposed method achieved better performance than the state-of-the-art methods for the case of the small number of features. As the number of selected features grew, the UFSMI performed as well as the LS method, both the methods achieved better results than the MCFS approach.

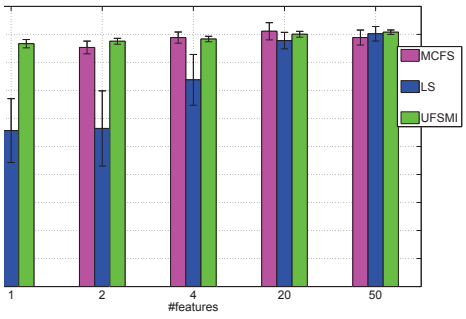


Fig. 4. Comparison of unsupervised feature selection methods. Face clustering problem: CMU PIE dataset. Statistics (mean and std) are shown for 10 repetitions.

To conclude, in this paper we proposed a novel unsupervised feature selection method called Unsupervised Feature Selection based On Mutual Information (UFSMI). It is derived from the observation that good features share information in common. By contrast, non-relevant features originate from noise and therefore are unlikely to possess common information and are less correlated with the other features. Mutual information is then used as a natural discriminator between these two types of features. The improved performance of UFSMI with respect to state-of-the-art methods was demonstrated on synthetic and real world standard datasets. Possible future research can concentrate on utilization of other mutual information estimators, which may enhance the algorithm performance. In addition, the algorithm is naturally cast as a feature ranking approach and subsequent applications can be tested as well.

.

5. REFERENCES

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

[2] V. Roth and T. Lange, "Feature selection in clustering problems," *Advances in Neural Information Processing Systems*, 2004.

[3] M. Law, M. Figueiredo, and A. Jain, "Simultaneous feature selection and clustering using mixture models," *IEEE Trans. on Pattern Analysis and Machine Int.*, pp. 1154–1166, 2004.

[4] C. Boutsidis, M. Mahoney, and P. Drineas, "Unsupervised feature selection for the k-means clustering problem," *Advances in Neural Information Processing Systems*, 2009.

[5] Y. Guan, J. Dy, and M. Jordan, "A unified probabilistic model for global and local unsupervised feature selection," *Int'l Conf. on Machine Learning*, 2011.

[6] S. Chang, N. Dasgupta, and L. Carin, "A bayesian approach to unsupervised feature selection and density estimation using expectation propagation," *IEEE Conf. on Computer Vision and Pattern Recognition*, 2005.

[7] X. He, D. Cai, and P. Niyogi, "Laplacian score for feature selection," *Advances in Neural Information Processing Systems*, 2005.

[8] Z. Zhao and H. Liu, "Spectral feature selection for supervised and unsupervised learning," *Int. Conference on Machine Learning*, 2007.

[9] Z. Zhao, L. Wang, and H. Liu, "Efficient spectral feature selection with minimum redundancy," *AAAI*, pp. 673–678, 2010.

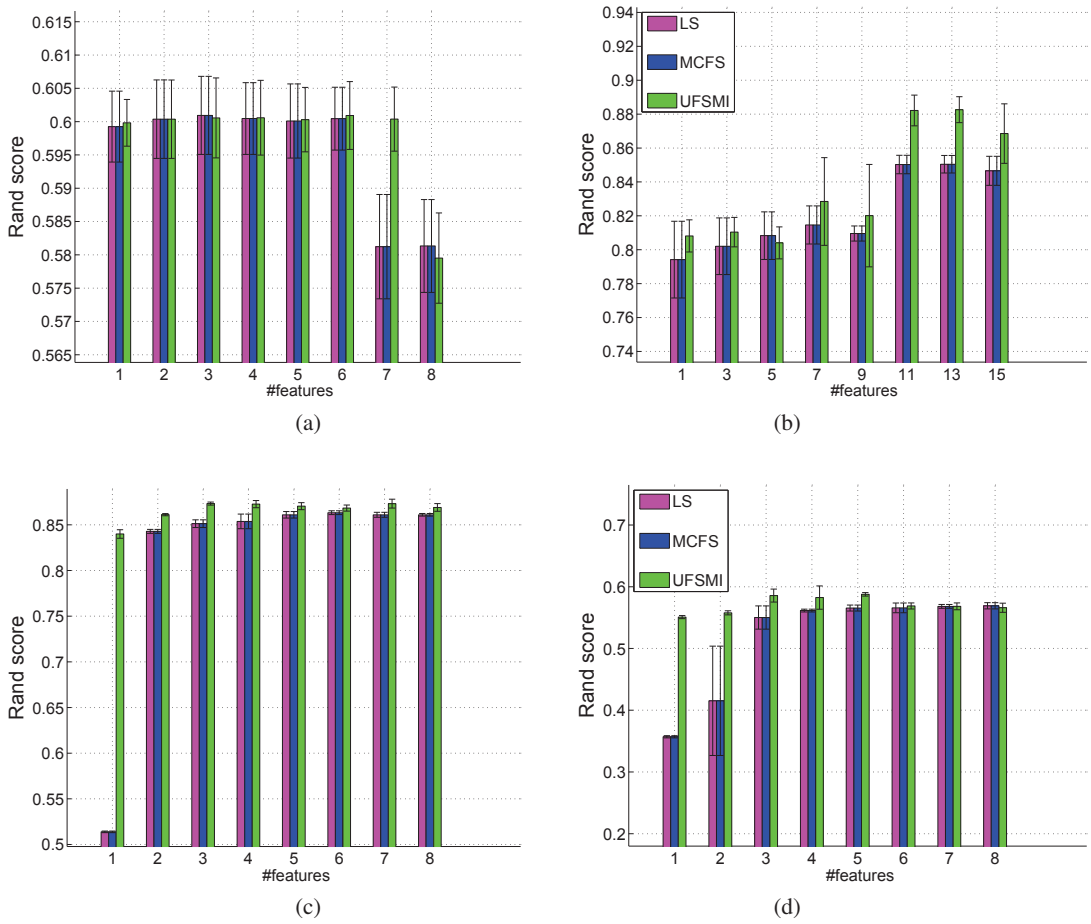


Fig. 3. Comparison of unsupervised feature selection methods. UCI datasets: (a) Abalone, (b) Segmentation, (c) Vowel, (d) Yeast. Statistics (mean and std) are shown for 10 repetitions.

[10] D. Cai, C. Zhang, and X. He, "Unsupervised feature selection for multi-cluster data," *ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD)*, 2010.

[11] L. Wolf and A. Shashua, "Feature selection for unsupervised and supervised inference: The emergence of sparsity in a weight-based approach," *Journal of Machine Learning Research*, pp. 1855–1887, 2005.

[12] T.M. Cover and J.A. Thomas, *Elements of Information Theory*, Wiley-Interscience, 1991.

[13] K. Torkkola, "Feature extraction by non-parametric mutual informtaion maximization," *Journal of Machine Learning Research*, pp. 1415–1438, 2003.

[14] S. M. Kay, "Fundamentals of statistical signal processing," 1993.

[15] J. D. Victor, "Binless strategies for estimation of information from neural data," *Physical Review*, 2002.

[16] L. Kozachenko and N. Leonenko, "On statistical estimation of entropy of random vector," *Problems Infor. Transmiss.*, pp. 95–101, 1987.

[17] M. Goria, N. Leonenko, V. Mergel, and P. Inverardi, "A new class of random vector entropy estimators and its applications in testing statistical hypotheses," *J. Nonparam. Statist.*, pp. 277–297, 2005.

[18] A. Kraskov, H. Stögbauer, and P. Grassberger, "Estimating mutual information," *Physical Review E*, p. 066138, 2004.

[19] S. Arya, G. D. da Fonseca, and D. M. Mount, "A unified approach to approximate proximity searching," *Proc. 18th Annu. European Sympos. Algorithms (ESA'10)*, 2010.

[20] I. Guyon and A. Elisseeff, "An introduction to variable and feature selection," *Journal of Machine Learning Research*, pp. 1157–1182, 2003.

[21] A. Frank and A. Asuncion, "UCI machine learning repository," 2010.

[22] W. Rand, "Objective criteria for the evaluation of clustering methods," *J. Amer. Statist. Assoc. 66*, pp. 846–850, 1971.