KNOWLEDGE ENGINEERING: PRINCIPLES AND TECHNIQUES
Proceedings of the International Conference on Knowledge Engineering,
Principles and Techniques, KEPT2009
Cluj-Napoca (Romania), July 2–4, 2009, pp. 25–28

SEMI-SUPERVISED FEATURE SELECTION WITH SVMS

ZALÁN BODÓ AND ZSOLT MINIER[1]

Abstract. Feature selection plays and important role in machine learning: eliminates irrelevant dimensions thus turning the learner into a better, more efficient system. In this paper we use non-linear semi-supervised SVMs for feature selection and through experiments we demonstrate the efficiency of the methods, showing how unlabeled data can lead to a better reduction. Semi-supervised feature selection is achieved by using semi-supervised/cluster kernels, that is embedding the information provided by the unlabeled data into the kernel, and applying dimensionality reduction methods developed for non-linear SVMs.

1. Introduction

Semi-supervised learning (SSL) is a special case of classification; it is halfway between classification and clustering. In SSL the training data is augmented by a set of unlabeled data samples, that is we have $\{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_\ell, y_\ell)\} \cup \{\mathbf{x}_{\ell+1}, \mathbf{x}_{\ell+2}, \ldots, \mathbf{x}_{N:=\ell+u}\}$, where usually there are far less labeled data than unlabeled ones, i.e. $\ell \ll u$. Semi-supervised learning is the problem of assigning labels to the unlabeled samples of the data set using the information provided by both the labeled and the unlabeled data.

Feature selection or dimensionality reduction methods are important for machine learning algorithms, because many problems usually deal with thousands of features, some of them representing only noise, others being strongly correlated, etc. Therefore in order to handle this high dimensionality and build efficient learners the elimination of irrelevant features is needed. Feature selection methods can be classified as follows [4]: wrappers, embedded methods and filters. Wrappers use an arbitrary machine learning technique with some heuristics for choosing the best feature set observing the performance of the classifier for these feature sets. Embedded methods are implicitly built into the learning technique, i.e. learning the optimal decision function and finding the best feature subset are performed simultaneously. Perhaps the most successful feature selection methods are filters, which choose a subset of features according to a particular measure.

2000 *Mathematics Subject Classification.* 68T05, 68T30.
Key words and phrases. semi-supervised learning, feature selection, kernel methods.

The most popular feature selection techniques are either supervised or unsupervised methods. We expect that by using semi-supervised learners (SVMs) for feature selection we can achieve a better dimensionality reduction, that is the larger unlabeled data set induces a better decision boundary and thus a smaller feature set.

2. Semi-supervised support vector machines

The Laplacian support vector machine [1] approaches the semi-supervised learning problem by introducing an additional regularization term involving the graph Laplacian $\mathbf{L}$, that reflects the intrinsic geometry of the data:

$$\min F(\mathbf{w}, b, \boldsymbol{\xi}) = \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{\ell}\xi_i + \gamma_I \mathbf{f}'\mathbf{L}\mathbf{f}$$
$$\text{such that} \quad y_i(\mathbf{w}'\phi(\mathbf{x}_i) + b) \geq 1 - \xi_i,\ \xi_i \geq 0, \quad i = 1, \ldots, \ell$$

where $\mathbf{f} = [f(\mathbf{x}_1) \ldots f(\mathbf{x}_N)]'$. Interestingly this can be solved by solving the supervised SVM problem using the kernel [2]

$$\tilde{k}(\mathbf{x}_1, \mathbf{x}_2) = k(\mathbf{x}_1, \mathbf{x}_2) - \mathbf{k}'_{\mathbf{x}_1}(\frac{1}{4\gamma_I}\mathbf{I} + \mathbf{L}\mathbf{K})^{-1}\mathbf{L}\mathbf{k}_{\mathbf{x}_2}$$

Other possibilities to use SVMs for semi-supervised learning exist. One example is to build a kernel based on the information provided by the labeled and unlabeled data, that is to use the unlabeled data to build a new representation of the samples [2], then train SVMs in a supervised setting with this kernel.

3. Semi-supervised feature selection

SVMs and linear classifiers in general, $\mathbf{w}'\mathbf{x}$, can be used for feature selection by eliminating those dimensions for which the magnitude of w_i is small. The feature selection method proposed in [5] ranks the dimensions by w_i and eliminates one or more features with small rank. This can be argued by the simple fact that large magnitude dimensions contribute more to the final decision. However this works only in the input space; introducing kernels for non-linear decision functions this selection should be done in the feature space, which could be of infinite dimensionality, therefore no straightforward solution is possible.

For eliminating the dimension with the smallest influence on the predicted label in the non-linear case, we calculate the "ranking criterion" as the variation of $\|\mathbf{w}\|^2$ caused by the removal [5]:

$$\left|\|\mathbf{w}^2 - \|\mathbf{w}^{(i)}\|^2\right| = \left|\boldsymbol{\beta}'\mathbf{K}\boldsymbol{\beta} - \boldsymbol{\beta}^{(i)\prime}\mathbf{K}^{(i)}\boldsymbol{\beta}^{(i)}\right| \tag{1}$$

where $\boldsymbol{\beta}_j = \alpha_j y_j$, $\mathbf{K}^{(i)}$ denotes the kernel matrix after removing the ith feature, and $\boldsymbol{\beta}^{(i)}$ denotes the vector corresponding to the solution $\boldsymbol{\alpha}^{(i)}$. This approach is called the SVM-RFE algorithm. For complexity reasons usually $\boldsymbol{\beta}$ is used instead

Algorithm 1 SVM-based feature selection using backward elimination

1: $F = \{1, 2, \ldots, d\}$
2: **repeat**
3: Train an SVM (using a semi-supervised/cluster kernel) with all the training data using the features from F
4: **for** $f_i \in F$ **do**
5: Evaluate the ranking criterion $R_c(f_i)$
6: **end for**
7: $f_r = \text{argmin}_i R_c$ ▷ Determine most irrelevant dimension
8: $F = F \setminus \{f_r\}$ ▷ Remove most irrelevant dimension
9: **until** $F = \emptyset$

of $\boldsymbol{\beta}^{(i)}$. In the experiments both approaches are tested. Other methods include using leave-one-out error (L) bounds for SVMs for the ranking criterion, like the radius-margin bound of Vapnik [6]. Since we are using L1-SVMs we apply a radius-margin bound proposed in [3]:

$$L \leq \left(R^2 + \frac{1}{C}\right)\left(\|\mathbf{w}\|^2 + C\sum_{i=1}^{\ell}\xi_i\right) \tag{2}$$

where R denotes the radius of the smallest sphere containing the data $\phi(\mathbf{x}_i)$ in the feature space. Using a backward elimination scheme the expression from eq. (1) and the right hand side of eq. (2) can be directly used as a ranking criterion for feature elimination: the feature which minimizes the ranking criterion is removed from the feature set. The scheme of the feature selection algorithm used by us is shown in Algorithm 1 [5, 6].

4. Experiments and discussion

In our experiments we tested the methods on a non-linear synthetic data set described in [7]. From the generated 52 dimensions only 2 are relevant. These dimensions are constructed as follows: if $y = -1$ then the two dimensions are drawn from $N(\boldsymbol{\mu}_1, \boldsymbol{\Sigma})$ or $N(\mu_2, \boldsymbol{\Sigma})$ with equal probability, $\boldsymbol{\mu}_1 = [-3/4, -3]'$, $\boldsymbol{\mu}_2 = [3/4, 3]'$, $\boldsymbol{\Sigma} = \mathbf{I}$, if $y = 1$ then the dimensions are drawn again from two normal distributions with equal probability having the parameters $\boldsymbol{\mu}_1 = [3, -3]'$, $\boldsymbol{\mu}_2 = [-3, 3]'$, and $\boldsymbol{\Sigma} = \mathbf{I}$. The remaining 50 features are noise, each one is generated from the normal distribution $N(0, 20)$.

Table 1 shows the obtained results. Because of lack of space the complete settings of the test are not described here. SVM-RFE and SVM-RMB denote the methods using equations (1) and (2) with supervised SVMs, LapSVM-RFE and LapSVM-RMB denote that a Laplacian SVM was used for learning, while the ^{r} at the end means that retraining was performed.

Methods	Training set size			
	10	**30**	**50**	**100**
SVM-RFE	$47.06 \pm 8\%$	$25.57 \pm 20\%$	$9.63 \pm 12\%$	$5.78 \pm 3\%$
SVM-RFE^r	$49.78 \pm 5\%$	$32.47 \pm 19\%$	$24.48 \pm 19\%$	$18.72 \pm 18\%$
SVM-RMB	$51.02 \pm 4\%$	$46.85 \pm 8\%$	$50.67 \pm 5\%$	$47.28 \pm 8\%$
SVM-RMB^r	$48.52 \pm 6\%$	$36.30 \pm 18\%$	$14.07 \pm 16\%$	$6.65 \pm 8\%$
LapSVM-RFE	$48.47 \pm 6\%$	$43.83 \pm 13\%$	$43.27 \pm 13\%$	$48.50 \pm 7\%$
LapSVM-RFE^r	$48.35 \pm 10\%$	$46.95 \pm 9\%$	$41.40 \pm 14\%$	$41.27 \pm 15\%$
LapSVM-RMB	$48.80 \pm 6\%$	$44.30 \pm 12\%$	$37.95 \pm 15\%$	$30.57 \pm 19\%$
LapSVM-RMB^r	$49.43 \pm 4\%$	$40.57 \pm 13\%$	$28.25 \pm 16\%$	$7.13 \pm 9\%$

TABLE 1. Test errors (mean and standard deviation) for the non-linear synthetic problem using SVMs and Laplacian SVMs for feature selection.

The results show that feature selection methods using SVMs clearly outperform LapSVMs on the synthetic non-linear data set used. Only the results of LapSVM-RMB^r with 100 points can be considered as acceptable. The methods require further tests on data sets where semi-supervised assumptions hold, and detailed analysis of the results obtained.

REFERENCES

[1] Mikhail Belkin, Partha Niyogi, and Vikas Sindhwani. Manifold regularization: A geometric framework for learning from labeled and unlabeled examples. *Journal of Machine Learning Research*, 7:2399–2434, 2006.

[2] Olivier Chapelle, Bernhard Schölkopf, and Alexander Zien. *Semi-Supervised Learning*. MIT Press, September 2006.

[3] Kai-Min Chung, Wei-Chun Kao, Chia-Liang Sun, Li-Lun Wang, and Chih-Jen Lin. Radius margin bounds for support vector machines with the RBF kernel. *Neural Computation*, 15(11):2643–2681, November 2003.

[4] Isabelle Guyon and André Elisseeff. An introduction to variable and feature selection. *Journal of Machine Learning Research*, 3:1157–1182, 2003.

[5] Isabelle Guyon, Jason Weston, Stephen Barnhill, and Vladimir Vapnik. Gene selection for cancer classification using support vector machines. *MACHLEARN: Machine Learning*, 46, 2002.

[6] Alain Rakotomamonjy. Variable selection using SVM-based criteria. *Journal of Machine Learning Research*, 3:1357–1370, 2003.

[7] Jason Weston, Sayan Mukherjee, Olivier Chapelle, Massimiliano Pontil, Tomaso Poggio, and Vladimir Vapnik. Feature selection for SVMs. In Todd K. Leen, Thomas G. Dietterich, and Volker Tresp, editors, *NIPS*, pages 668–674. MIT Press, 2000.

[1] FACULTY OF MATHEMATICS AND COMPUTER SCIENCE, BABEŞ–BOLYAI UNIVERSITY, CLUJ-NAPOCA

E-mail address: {zbodo, minier}@cs.ubbcluj.ro