Graph Laplacian for Semi-supervised Feature Selection in Regression Problems

Gauthier Doquire* and Michel Verleysen

Université catholique de Louvain, ICTEAM - Machine Learning Group
Place du Levant, 3, 1348 Louvain-la-Neuve, Belgium
{gauthier.doquire,michel.verleysen}@uclouvain.be
http://www.ucl.ac.be/mlg

Abstract. Feature selection is fundamental in many data mining or machine learning applications. Most of the algorithms proposed for this task make the assumption that the data are either supervised or unsupervised, while in practice supervised and unsupervised samples are often simultaneously available. Semi-supervised feature selection is thus needed, and has been studied quite intensively these past few years almost exclusively for classification problems. In this paper, a supervised then a semi-supervised feature selection algorithms specially designed for regression problems are presented. Both are based on the Laplacian Score, a quantity recently introduced in the unsupervised framework. Experimental evidences show the efficiency of the two algorithms.

Keywords: Feature selection, semi-supervised learning, Graph Laplacian.

1 Introduction

Feature selection is an important task for many applications involving the mining of high dimensionnal datasets. Indeed, many features are often either redundant or totally uninformative and can harm learning algorithms, making them prone to overfitting [1]. Moreover, the elimination of such useless features is generally benefical both for the learning time and the interpretation of models.

Traditionally, feature selection algorithms are said to be *supervised*, in the sense that they assume the knowledge of the output (a class label for classification problems and a continuous value for regression ones) associated with each training sample [2]. On the other hand, *unsupervised* feature selection methods have also been developped, whose most obvious example is simply the evaluation of each feature variance. More complex unsupervised algorithms include for example an approach using feature similarity [3] or a graph Laplacian based ranking [4] which will be desribed later as it is the base of this paper.

Halfway between those two situations, a more realistic assumption is that, in many real-world problems, unsupervised samples are often easy to obtain and

* Gauthier Doquire is funded by a Belgian FRIA grant.

J. Cabestany, I. Rojas, and G. Joya (Eds.): IWANN 2011, Part I, LNCS 6691, pp. 248–255, 2011.

thus numerous, while only a few labeled samples are typically available. This limitation is mainly due to the cost associated to the obtention of the desired outputs (human expertise, destructive test...). These considerations naturally led to the development of *semi-supervised* learning, in which the few available information about the output is used to improve learning algorithms based on the unsupervised part only [5,6].

In this context, many feature selection algorithms have been proposed recently. Among others, Zhao et Liu proposed an approach using spectral analysis [7] while Quinzan et al. introduced an algorithm based on feature clustering, conditional mutual information and conditional entropy [8]. These two workss as well as the very large majority of semi-supervised feature selection algorithms are basically designed to handle classification problems, while, to the best of our knowledge, almost no work has been done to develop algorithms specific to regression problems.

This paper first introduces a supervised feature selection algorithms, which is then used to achieve semi-supervised feature selection. Both are specifically designed to handle continuous outputs. They extend the unsupervised concepts in [4]. Within the unsupervised framework, this algorithm scores features according to their locality preserving power. Roughly speaking, good features have close values for close samples and thus preserve the local structure. In this work, the idea is extended by using distance information between the output of supervised samples.

The rest of the paper is organized as follows. Section 2 briefly presents the original unsupervised Laplacian Score. Section 3 presents the supervised feature selection criterion. Section 4 introduces the semi-supervised algorithm which combines in a simple way the information from supervised and unsupervised samples. Experimental evidences of its efficiency are presented in Section 5. Eventually, Section 6 gives some concluding remarks and directions for future work.

2 Laplacian Score

This section briefly presents the Laplacian Score, as introduced by He et al. [4] for unsupervised feature selection. As already discussed, the method selects features according to their locality preserving power.

Consider a dataset X. Let f_{ri} denote the r^{th} feature of the i^{th} sample ($i = 1 \dots m$), $\mathbf{x_i}$ the i^{th} data point and $\mathbf{f_r}$ the r^{th} feature. A proximity graph with m nodes is built, which contains an edge between node i and node j if the corresponding points $\mathbf{x_i}$ and $\mathbf{x_j}$ are close, i.e. if $\mathbf{x_i}$ is among the k nearest neighbors of $\mathbf{x_j}$ or conversely. Throughout this paper, the proximity measure used to compute the nearest neighbors of a point is always the Euclidean distance.

From the proximity graph, a matrix S^{uns} is built by setting

$$S^{uns}_{i,j} = \begin{cases} e^{-\frac{\left\|\mathbf{x_i}-\mathbf{x_j}\right\|_2}{t}} & \text{if } \mathbf{x_i} \text{ and } \mathbf{x_j} \text{ are close} \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

where t is a suitable constant. $D^{uns} = diag(S^{uns}\mathbf{1})$, with $\mathbf{1} = [1 \dots 1]^T$, is defined, as well as the graph Laplacian $L^{uns} = D^{uns} - S^{uns}$ [9].

The mean (weighted by the local density of data points) of each feature $\mathbf{f_r}$ is then removed: the new features are called $\tilde{\mathbf{f}}_\mathbf{r} = \mathbf{f_r} - \frac{\mathbf{f_r^T} D^{uns} \mathbf{1}}{\mathbf{1^T} D^{uns} \mathbf{1}} \mathbf{1}$. This is done to prevent a non-zero constant vector such as $\mathbf{1}$ to be assigned a zero Laplacian score as such a feature obviously does not contain any information.

Eventually the Laplacian score of each feature $\mathbf{f_r}$ is computed as

$$L_r = \frac{\tilde{\mathbf{f}}_\mathbf{r}^{\,T} L^{uns} \tilde{\mathbf{f}}_\mathbf{r}}{\tilde{\mathbf{f}}_\mathbf{r}^{\,T} D^{uns} \tilde{\mathbf{f}}_\mathbf{r}} \tag{2}$$

and features are ranked according to this score, in increasing order.

As a convincing justification of this criterion for feature selection, one can notice that

$$L_r = \frac{\sum_{ij} (f_{ri} - f_{rj})^2 S_{ij}^{uns}}{Var(\mathbf{f_r})}. \tag{3}$$

The numerator thus penalizes features belonging to close samples and however having very different values. $Var(\mathbf{f_r})$ can be seen as the estimated weighted variance of feature r where the D matrix models the importance of the data points. Features with a high variance are thus preferred, as they are expected to have a higher discriminative power. More details can be found in [4].

3 Supervised Laplacian Score

3.1 Definitions

A formalism similar to the one described in the previous section can also be derived for *supervised* feature selection.

Consider again the training set X containing m samples $\mathbf{x_i}$ described by n features. In case of a supervised regression problem, an output vector $Y = [y_1 \dots y_m] \in \Re^m$ is also given. Under the assumption that the output Y is generated by a continuous and smooth enough function of X, it is natural to expect close samples $\mathbf{x_i}$ and $\mathbf{x_j}$ to have close output values y_i and y_j. Consequently, good features are expected to have close values for data points whose outputs are close too.

Define the matrix S^{sup} as:

$$S_{i,j}^{sup} = \begin{cases} e^{-\frac{(y_i - y_j)^2}{t}} & \text{if } y_i \text{ and } y_j \text{ are close} \\ 0 & \text{otherwise} \end{cases} \tag{4}$$

and $D^{sup} = diag(S^{sup}\mathbf{1})$, $L^{sup} = D^{sup} - S^{sup}$, $\tilde{\mathbf{f}}_\mathbf{r} = \mathbf{f_r} - \frac{\mathbf{f_r^T} D^{sup} \mathbf{1}}{\mathbf{1^T} D^{sup} \mathbf{1}} \mathbf{1}$. Criterion (2) can again be used to rank features by computing a so-called Supervised Laplacian Score (SLS):

$$SLS_r = \frac{\tilde{\mathbf{f}}_\mathbf{r}^{\,T} L^{sup} \tilde{\mathbf{f}}_\mathbf{r}}{\tilde{\mathbf{f}}_\mathbf{r}^{\,T} D^{sup} \tilde{\mathbf{f}}_\mathbf{r}} \tag{5}$$

Here again, in Equation (4), two points are considered to be close if one is among the k nearest neighbors of the other while t is a suitable (positive) constant.

Expression (3) can of course be derived with $S_{i,j}^{sup}$ and thus $Var(\mathbf{f_r})$ adapted to the supervised case.

3.2 Illustration

SLS is briefly compared with the correlation coefficient, a widely used criterion for feature selection. The objective is to show the ability of the method to achieve feature selection and its greater capability detect non linear relationships between each feature and the output. Two artificial problems are considered.

The first one consists of 8 features $X_1 \dots X_8$ uniformly distributed on $[0; 1]$. The output is defined as:

$$Y_1 = \cos(2\,\pi\,X_1\,X_2)\sin(2\,\pi\,X_3\,X_4). \tag{6}$$

The second consists of 4 features $X_1 \dots X_4$ uniformly distributed on $[0; 1]$. The output is defined as:

$$Y_2 = X_1^2\,X_2^{-2}. \tag{7}$$

The sample size is 1000 in both cases and 1000 datasets are randomly generated for each problem. The comparison criterion is the percentage of cases for which the 4 (2) informative features are the 4 (2) best rated.

For the first problem this percentage is 93% with SLS and 25% with the correlation coefficient. For the second problem, the percentages are 100% and 32% respectively. The advantage of SLS in these two simple cases is thus obvious.

4 Semi-supervised Laplacian Score

First experiments showed promising results concerning the use of SLS for supervised regression problems with a large number of data points. When the number of supervised samples is small, however, unsupervised samples have also to be taken into account.

The semi-supervised feature selection algorithm proposed in this paper is based on the developments in the two previous sections. More precisely, LS and SLS are both based on the locality preserving power of the features. The difference comes from the fact that locality (measured by the distance between samples) is defined from the unsupervised part of data for LS and from the output for SLS. A quite intuitive idea is thus to compute the distance between two samples from their outputs if both are known, and from the unsupervised part of the data otherwise.

Consider a semi-supervised regression problem consisting in the training set X and an output vector $Y = [y_1 \dots y_s] \in \Re^s$, $s << m$.

The first step is to define a matrix d of distances between each pair of data points:

$$d_{i,j} = \begin{cases} (y_i - y_j)^2 & \text{if } y_i \text{ and } y_j \text{ are known} \\ \frac{1}{n}\sum_{k=1}^{n}(f_{k,i} - f_{k,j})^2 & \text{otherwise.} \end{cases} \tag{8}$$

In the second case, the distance is normalized by the number of features n in order to keep it comparable to the distance computed from the output.

A matrix S^{semi} is then built as follows:

$$S_{i,j}^{semi} = \begin{cases} e^{-\frac{d_{i,j}}{t}} & \text{if } \mathbf{x_i} \text{ and } \mathbf{x_j} \text{ are close and } y_i \text{ or } y_j \text{ is unknown,} \\ C \times e^{-\frac{d_{i,j}}{t}} & \text{if } \mathbf{x_i} \text{ and } \mathbf{x_j} \text{ are close and } y_i \text{ and } y_j \text{ are known,} \\ 0 & \text{otherwise.} \end{cases} \quad (9)$$

Two points are considered as close if one is among the k nearest neighbors of the other one.

The (positive) constant C allows us to give more weight to the information coming from the supervised part of the data, as it is believed to be more important than the unsupervised one for the feature selection problem.

One can then define $D^{semi} = diag(S^{semi}\mathbf{1})$, $L^{semi} = D^{semi} - S^{semi}$ and $\tilde{\mathbf{f}}_\mathbf{r} = \mathbf{f_r} - \frac{\mathbf{f_r^T} D^{semi}\mathbf{1}}{\mathbf{1^T} D^{semi}\mathbf{1}}\mathbf{1}$.

The criterion for semi-supervised feature selection, called Semi-Supervised Laplacian Score (SSLS), is eventually:

$$SSLS_r = \frac{\tilde{\mathbf{f}}_\mathbf{r}^T L^{semi} \tilde{\mathbf{f}}_\mathbf{r}}{\tilde{\mathbf{f}}_\mathbf{r}^T D^{semi} \tilde{\mathbf{f}}_\mathbf{r}} \times SLS_r, \quad (10)$$

where SLS_r is the Supervised Laplacian Score (computed on the supervised samples only). The criterion thus combines the influence of both the unsupervised and the supervised part of the data, giving however this last part more importance.

5 Experimental Results

In this section the interest of the proposed semi-supervised approach is illsutrated on three real-world data sets.

The first one is the Juice dataset. The goal is to estimate the level of saccharose of an orange juice from its observed near-infrared spectrum. 218 spectra samples with 700 points are available. The dataset can be downloaded from the website of the UCL's Machince Learning Group [1] .

The second one is the Nitrogen dataset, containing originally 141 spectra discretized at 1050 different wavelengths. The objective is the prediction of the nitrogen content of a grass sample. The data can be obtained from the Analytical Spectroscopy Research Group of the University of Kentucky[2] . In order to reduce the huge number of features, each spectrum is represented by its coordinates in a B-splines base as a preprocessing [10]. 105 features are built this way.

The last one is the Delve-Census data set, for which only the 2048 fisrt samples are considered. The data is available from the University of Toronto[3] . Originally,

[1] http://www.ucl.ac.be/mlg/

[2] http://kerouac.pharm.uky.edu/asrg/cnirs/

[3] http://www.cs.toronto.edu/ delve/data/census-house/desc.html

each sample consists in 139 demographic features about a small region and the objective is to predict the median price of houses in each region. However, only 104 features are considered here, since those which are too correlated with the output have been removed for the experiments.

The performances of the feature selection algorithms are evaluated by the root mean squared error (RMSE) of a 5 nearest neighbors prediction model.

First features are selected on the training set with only a few randomly selected supervised samples. The model is then used to predict the output of the points of an independent test set. For the prediction step, as a too small number of labeled data would not allow the model to perform correctly, all the samples in the training set are supposed to be labelled.

This procedure ensures that the performances reflect the quality of the feature selection itself, and are not too much influenced by the prediction model. The algorithms are tested with 7 and 10 supervised samples for the two first (smaller) data sets, and with 70 and 100 supervised samples for the larger Delve Census dataset. The RMSE is estimated through a 5-fold cross validation procedure repeated 10 times.

Parameter t is set to 1, 5 neighbors are considered for computing the unsupervised (1) and supervised (4) score, while 30 neighbors are considered for the semi-supervised score (9). Indeed, the number of supervised samples being small, increasing the number of neighbors considered in the analysis allows to take such samples into account. The parameter C in (9) is set to 5. This moderate value gives a large importance to the supervised samples, but still gives a significant weight to the information coming from the unsupervised data points. The maximum number of selected features is 100. Before any distance computation, features are normalized by removing their mean and dividing them by their standard deviation.

Figure 1 first shows how the use of a few labeled data can improve the feature selection procedure for regression purposes when compared with the unsupervised approach. Indeed, as expected, the unsupervised approach (LS) is obviously the one performing the worse on all three datasets and the prediction performances are greatly improved by the use of only a small number of supervised samples.

Moreover, the proposed SSLS also performs better than the correlation coefficient for the three examples. This is particularly obvious for the Juice and the Delve Census data sets where the RMSE obtained with the SSLS is never larger than the one obtained with the correlation coefficient.

Eventually, results on the Juice data set underline the interest of the semi-supervised SSLS approach over the supervised SLS method. This indicates that the knowledge coming from the unsupervised samples is efficiently taken into account in the feature selection procedure. Results on the Nitrogen data set are also in favour of the SSLS, which leads in most of the cases to a smaller RMSE. It also reaches the lowest global RMSE with both 7 and 10 supervised samples. Results on the Delve Census dataset are slightly better for the supervised approach.

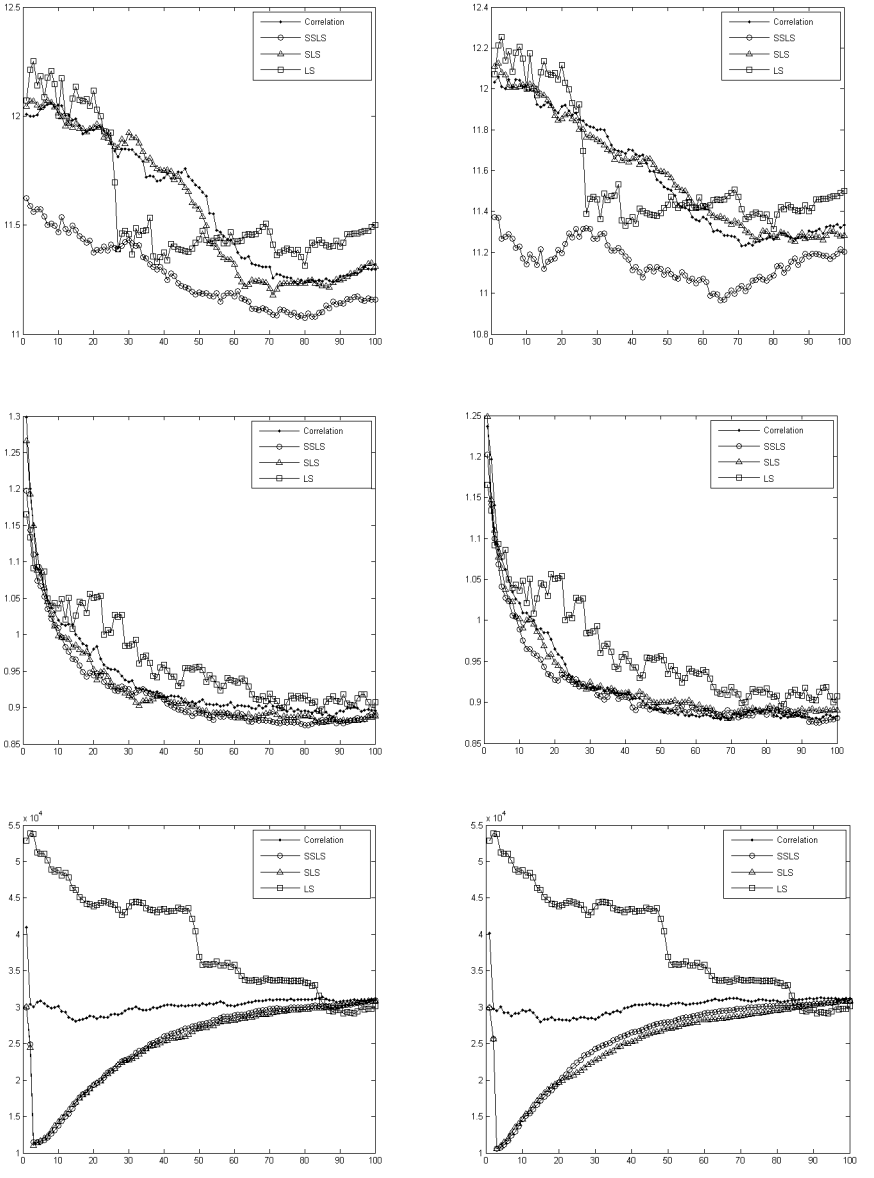

Fig. 1. RMSE as a function of the number of selected features with 7 (left) and 10 (right) supervised samples. From top to bottom: Juice, Nitrogen and Delve Census data set.

6 Conclusions and Future Work

In this paper, two feature selection algorithms are introduced for regression problems. Both are inspired by the Laplacian Score (LS), a recently introduced unsupervised feature selection criterion and are based on the locality preserving power of the features. In other words, the algorithms first select feature which are coherent with a distance measure between samples.

In supervised learning, the distances are evaluated with the output only, leading to the Supervised Laplacian Score (SLS). In the case of semi-supervised learning, distances are computed with the output if they are known or with the data points otherwise. The semi-supervised score obtained this way is then combined with the SLS to produce the semi-supervised Laplacian score (SSLS).

Experiments demonstrate the interest of the proposed approach, especially for the semi-supervised feature selection problem. More precisely, for the problems considered here, SSLS is shown to be superior to the correlation coefficient and to the unsupervised approach. Moreover, it also outperforms its supervised version on two datasets, showing the interest of considering unsupervised samples for the feature selection when the number of supervised points is too low.

Further work could be focused on new ways to combine the information coming from the supervised and unsupervised part of the data, as only the product has been considered here.

References

1. Guyon, I., Elisseeff, A.: An Introduction to Variable and Feature Selection. J. Mach. Lear. Res. 3, 1157–1182 (2003)
2. Battiti, R.: Using Mutual Information for Selecting Features in Supervised Neural Net Learning. IEEE T. Neural. Networ. 5, 537–550 (1994)
3. Mitra, P., Murthy, C.A., Pal, S.K.: Unsupervised Feature Selection Using Feature Similarity. IEEE T. Pattern. Anal. 24 (2002)
4. He, X., Cai, D., Niyogi, P.: Laplacian Score for Feature Selection. In: Advances in Neural Information Processing Systems (NIPS), vol. 17 (2005)
5. Chapelle, O., Schölkopf, B., Zien, A.: Semi-Supervised Learning. MIT Press, Cambridge (2007)
6. Zhu, X., Goldberg, A.B.: Introduction to Semi-Supervised Learning. Morgan & Claypool Publishers, San Francisco (2009)
7. Zhao, Z., Liu, H.: Semi-supervised Feature Selection via Spectral Analysis. In: 7th SIAM International Conference on Data Mining (2007)
8. Quinzán, I., Sotoca, J.M., Pla, F.: Clustering-Based Feature Selection in Semi-supervised Problems. In: 9th International Conference on Intelligent Systems Design and Applications, pp. 535–540 (2009)
9. Chung, F.R.K.: Spectral Graph Theory. CBMS Regional Conference Series in Mathematics 92. American Mathematical Society, Providence (1997)
10. Rossi, F., Delannay, N., Conan-Guez, B., Verleysen, M.: Representation of Functional Data in Neural Networks. Neurocomputing 64, 183–210 (2005)