Income Prediction via Support Vector Machine

Alina Lazar
Computer Science and Information Systems
Department
Youngstown State University
Youngstown, OH 44555
alazar@cis.ysu.edu

Abstract – *Principal component analysis and support vector machine methods are employed to generate and evaluate income prediction data based on the Current Population Survey provided by the U.S. Census Bureau. A detailed statistical study targeted for relevant feature selection is found to increase efficiency and even improve classification accuracy. A systematic study is performed on the influence of this statistical narrowing on the grid parameter search, training time, accuracy, and number of support vectors. Accuracy values as high as 84%, when compared against a test population, are obtained with a reduced set of parameters while the computational time is reduced by 60%. Tailoring computational methods around specific real data sets is critical in designing powerful algorithms.*

Keywords: Support vector machine, principal component analysis, classification, accuracy, ROC curve.

1 Introduction

Supervised learning methods based on statistical learning theory, for classification and regression, provide good generalization and classification accuracy on real data. However, their inherent trade-off is their computational expense. Recently, support vector machines (SVM) [1]-[4] have become a popular tool for learning methods since they translate the input data into a larger feature space where the instances are linear separable, thus increasing efficiency. In SVM methods a kernel which can be considered a similarity measure is used to recode the input data. The kernel is used accompanied by a map function Φ.

Even if the mathematics behind the SVM is straight forward, finding the best choices for the kernel function and parameters can be challenging, when applied to real data sets. Usually, the recommended kernel function for nonlinear problems is the Gaussian radial basis function, because it resembles the sigmoid kernel for certain parameters and it requires less parameters than a polynomial kernel. The kernel function parameter g and the parameter C, which control the complexity of the decision function versus the training error minimization, can be determined by running a 2 dimensional grid search, which means that the values for pairs of parameters (C, g) are generated in a predefined interval with a fixed step. The performance of each combination is computed and used to determine the best pair of parameters. However, due to memory limitations and the quadratical grow of the kernel with the number of training examples, it is not practical to grid search for SVM's parameters by using datasets with more than 10^3 data instances. Also the non-sparse property of the solution leads to a really slow evaluation process. Thus, for larger datasets only a randomly selected subset of training instances is used for the grid search. A supplementary data reduction [5] can be done in terms of variables or features of the data set considered. Redundant or highly correlated features can be replaced with a smaller uncorrelated number of features capturing the entire information. This can be done by applying a method called Principal Component Analysis (PCA) before using the SVM algorithm.

The experiments presented in this paper used the Current Population Survey (CPS) database provided by the U.S. Census Bureau [6]. The CPS survey was conducted for more than 50 years and collects information about the social, demographic and economic characteristics of the labor force 16 years and older of the U.S. population. The data collected each month is used to compute reports about employment, unemployment, and earnings. It also includes statistics about various social factors from voting to smoking. The government policymakers and legislators use the statistics generated from the CPS data as indicators about the economic and social situation and for the planning and evaluation of many government programs. The data is publicly available and free of charge, fact that encouraged its use in various social and economic studies [7], [8]. Due to the large number of variables included and its implicit large size it is farfetched to believe that its entire value has been fully exploited. This has motivated its active use by the machine learning and knowledge discovery communities, as a platform for testing various data mining methods including, neural networks, nearest neighbor, decision tree and lately support vector machines.

In the present report the CPS data is used as a testing and evaluation platform, in order to investigate what is the consequence in terms of accuracy and computing time of reducing the input data vertically in terms of the number of training examples, and horizontally in terms of the number of features considered. In this context one problem is to determine the ideal ratio between the number of instances used for grid search and training, respectively.

The present paper is structured as follows: the first two sections are dedicated to the introduction of the fundamentals for the SVM methods and the principal component analysis for feature selections, respectively. The next section presents the experimental results applied on different instances of the adult dataset with different parameter sets. Finally, the last section it is dedicated to conclusions and future work.

2 Support Vector Machine

The machine learning algorithms named support vector machine proposed by Vapnik [9] consist of two important steps. Firstly, the dot product of the data points in the feature space, called kernel, is computed. Secondly, a hyperplan learning algorithm is applied to the kernel.

Let (x_i, y_i), $i = 1,\ldots,l$, be the training set of examples. The decision $y_i \in \{-1, 1\}$ is associated with each input instance $x_i \in R^N$ for a binary classification task. In order to find a linear separating hyperplan with good generalization abilities, for the input data points, the set of hyperplanes $\langle w,\mathbf{x}\rangle+b=0$ is consider. The optimal hyperplan can be determined by maximizing the distance between the hyperplan and the closest input data points. The hyperplan is the solution of the following problem:

$$\min_{w\in R^l\times R^l, b\in R} t(w) = \frac{1}{2}\|w\|^2 \quad (1)$$

when $y_i(\langle w, x_i\rangle + b) \geq 1$ for all $i = 1,\ldots, l$.

One challenge is that in practice an ideal separating hyperplan may not exist due to a large overlap between input data points from the two classes. In order to make the algorithm flexible a noise variable $e_i \geq 0$ for all $i = 1,\ldots,l$, is introduced in the objective function as follows:

$$\min_{w\in R^l\times R^l, b\in R} t(w,e_i) = \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{l} e_i \quad (2)$$

when $y_i(\langle w, x_i\rangle + b) \geq 1 - e_i$ for all $i = 1,\ldots, l$.

By using Lagrange multipliers the previous problem can be formulated as the following convex maximization problem [10]:

$$W(a) = \sum_{i=1}^{l} a_i - \frac{1}{2}\sum_{i,j=1}^{l} y_i y_j a_i a_j K(x_i, x_j) \quad (3)$$

when the following conditions are true, $0 \leq a_i \leq C$ for all $i = 1,\ldots, l$, and $\sum_{i=1}^{l} a_i y_i = 0$. Here the positive constant C controls the trade-off between the maximization of (3) and the training error minimization, $\sum e_i$.

From the optimal hyperplan equation the decision function for classification can be generated. For any unknown instance x the decision will be made based on:

$$f(x) = sign\left(\sum_{i=1}^{l} y_i a_i K(x_i, x) + b\right) \quad (4)$$

which geometrically corresponds to the distance of the unknown instance to the hyperplan.

The method described until now works well on linear problems. Function K, the kernel from (4) enables good results for nonlinear decision problems. The dot product of the initial input space is called the new higher-dimensional feature space.

$$K : R^l \times R^l \rightarrow R, K(x_i, x_j) = \langle f(x_i), f(x_j)\rangle \quad (5)$$

A polynomial kernel, the Gaussian and the sigmoid function are suitable kernels with similar behavior in terms of the resulting accuracy and they can be tuned by changing the values of the parameters. There is no good method to choose the best kernel function. The results reported in this paper were obtained by using the following Gaussian radial basis function [11] as kernel.

$$K(x_i, x_j) = \exp\left(-\frac{\|x_i - x_j\|^2}{2g^2}\right) \quad (6)$$

3 Principal Component Analysis

Principal component analysis [12] (PCA) is a well known powerful multivariate analysis approach, which allows us to reduce the size of a dataset with large number of independent variables. The smaller set will summarize best the larger set. The method is performed by solving an eigenvector problem or by using iterative algorithms and the result is a set of orthogonal vectors called principal components. The mapping of the larger set into the new smaller set is done by projecting the initial instances on

the principal components. The first principal component is defined as the direction given by a linear regression fit through the input data. This direction will hold the maximum variance in the input data. The second component is orthogonal on the first vector, uncorrelated and it is defined to maximize the remaining variance. This procedure is repeated until the last vector is obtained.

Given a set of instances $x_i \in R^N$, $1 \le i \le l$, with the mean equal to 0, $\sum_{i=1}^{l} x_i = 0$ PCA uses the covariance :

$$Cov = \frac{1}{l}\sum_{i=1}^{l} x_i x_i^T \qquad (7)$$

or the correlation matrix to compute the principal components. The *Cov* matrix is diagonalizable with nonnegative eigenvalues because it is positive definite. The set of all the eigenvalues of the *Cov* matrix is the spectrum of *Cov* denoted by Λ. The eigenvalue decomposition $Cov = VLV^{-1}$ that can be reformulated as $CovV = VL$. Also, if v_i is the ith column of V and l_i is the ith diagonal entry of L, then

$$\text{Cov} \cdot v_i = l_i v_i \qquad (8)$$

where v_i is an eigenvector of *Cov* and l_i is the corresponding eigenvalue. Simple mathematics gives the following equation for the eigenvalues and eigenvectors:

$$l v = Cov \cdot v = \frac{1}{l}\sum_{i=1}^{l} \langle x_i, v \rangle x_l \qquad (9)$$

The eigenvalues l_i, represent the variance of the new extracted components and give a good indication on how many components are important and how many can be discarded without loosing pertinent information. The components retained for the SVM are:

- components with eigenvalues greater than 1.
- components that fell on the maximum slop side of the eigenvalues plot (See Figure 3 for the data set studied).

Thus the main output of the PCA method is to reduce the feature space to be employed in the SVM algorithm. Also, the PCA produces a new set of eigenvectors which inherently are guaranteed to satisfy the orthogonality relations. In the following income prediction results are generated and analyzed using both the reduced original set as well as the new generated eigenvector space.

4 Income Data

One income dataset was previously extracted from the CPS data and posted on the University of California Irvine (UCI) repository [13]. The dataset, posted on the UCI machine learning repository and named "adult dataset", was extracted from the 1994 CPS data. Records with unreal values were eliminated and finally the 48,842 instances were divided into two files: a training file and a testing one. Fourteen attributes, eight categorical and six continuous were chosen. They were age, work class, weight, education, education number (continuos), marital status, occupation, relationship, race, sex, capital gain, capital loss, hours per week and native country. In the present study the decisions in the adult dataset are tailored to predict if a person income is more or less than 50K.

4.1 PCA

Before applying the PCA method, a good visual representation of the relationship between two distinct features can be obtained using a scatterplot. In a scatterplot individual points are represented by using the two features as the axes of a two-dimensional plot. If the scatterplot of two variables can be easily approximated by a regression line, it means that a new variable can be defined as a linear combination of the two plotted features. A correlation coefficient close to 1 (0.881) is characteristic to highly correlated variables. For the two variables represented in figure 1 education and education number the correlation is 0.881.

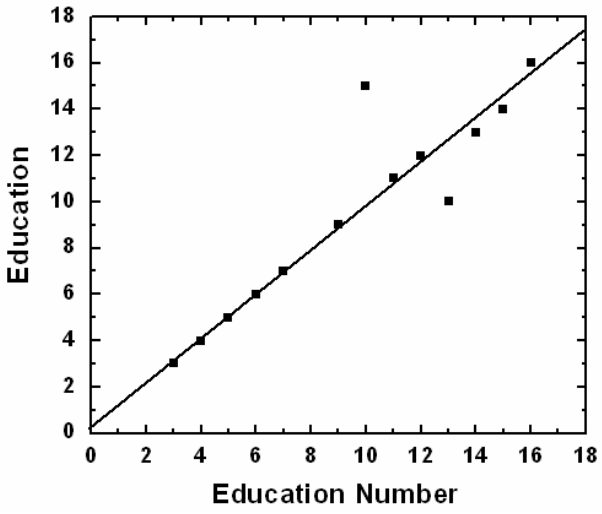

Figure 1. Scatterplot for Education Number versus Education

On the other side if the data points are widely scattered that means that no relation exists between the two features. As an example, Age and Education in this

case are virtually independent and are characterized by a low correlation -0.2.

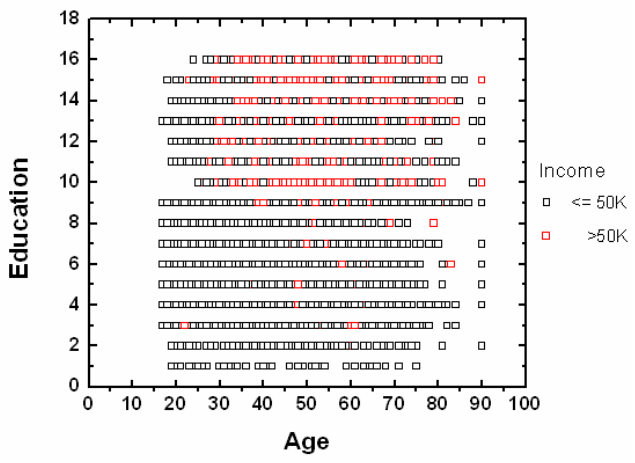

Figure 2. Age versus Education Scatterplot

Table 1 summarizes the calculated number of principal components and the cumulative percent variation for three different values of eigenvalues, after applying the PCA method.

Table 1. Eigenvalues lower limit and number of components

Eigenvalues ≥	# of components selected	%
1.0	6	59.100
0.8	10	85.558
0.0	13	99.175

The eigenvalues versus the component number are represented in Figure 3 as a simple line plot which allows their isolation in respect with their relative importance. Based on this results four new data sets are build: adult_13, adult_10, adult_6 and adult_eig. For adult_13 based on the relationship discovered between the two features education and education number, only education number was removed. Next, adult_10 was built by removing only four variables: age, education number, marital status, sex. For adult_6 we keep only the components associated with the eigenvalues greater than one. The features correlated with these factors are: work class, weight, education number, marital status, capital gain and hours per week. The 6 features capture only 59.1% (table 1) from the initial information represented by all 14 features. The last dataset adult_eig, contains the 13 principal components generated by the PCA method.

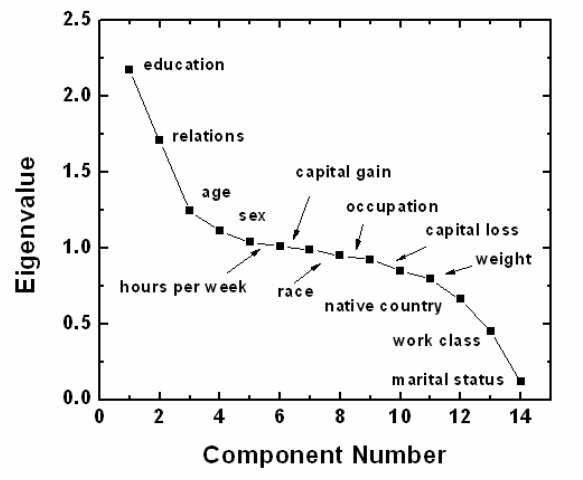

Figure 3. Eigenvalues and Components

4.2 Data Preprocessing

Before applying any machine learning algorithms the data is preprocessed. The range for each feature should not be too small or too large. Linear scaling is a simple procedure which translates the features into a given numerical range, i.e. [-1, 1]. By applying this procedure we make sure that the features with small numerical range will not be dominated by variables with large numerical ranges. Also, very small or very large values can cause computational problems during the calculation of the kernel. Both the training and the testing data are scaled in the same ranges.

Previous research efforts using the adult data set [7] and [8] used a discretization method for the data preprocessing, in which the six continuous variables were quantified into 123 binary features. However, because SVM is a method which deals very well with real numbers, scaling is a much better approach than discretization.

4.3 SVM Grid Search

As mentioned before in the SVM section, in order to use the SVM learning algorithm with the Gaussian kernel to train it efficiently on the input data sets, values for two parameters (C, g) are needed [14]. There are no universal best values for all the problems and very often the two parameters are determined empirically. A straightforward method is to use cross-validation on the training dataset and perform a grid search . Cross-validation means that the training dataset is divided into n subsets of equal size. The classifier obtained after training on n-1 subsets is tested on the remaining subset. In order to find the best training, the parameters, (C, g) are generated in the $[2^{-5}, 2^{15}]\times[2^{-15}, 2^{3}]$

interval with a 2 step for C and -2 step for g, and the pair with the best cross validation classification rate is chosen. Because the grid-search with cross validation is time consuming it is hard to apply on data sets with the magnitude order larger than 10^3. Since the adult training dataset has over 32,561 instances a subset of cases is randomly selected.

The grid search was applied to two training sets, one containing 5% of the initial training set and the second containing 10%. Since the classification accuracy on the testing data improved less than 0.1 and the time increased significantly (more than 30 times) when using the 10 % set, compared with the 5 % set, it is inferred that for all purposes a grid search subset of 5 % is sufficient. The grid for a 5 % randomly chosen adult data set is shown in figure 4 with the best (C, g) pair $(2^5, 2^{-3})$, resulting in a cross-validation rate of 83.58%.

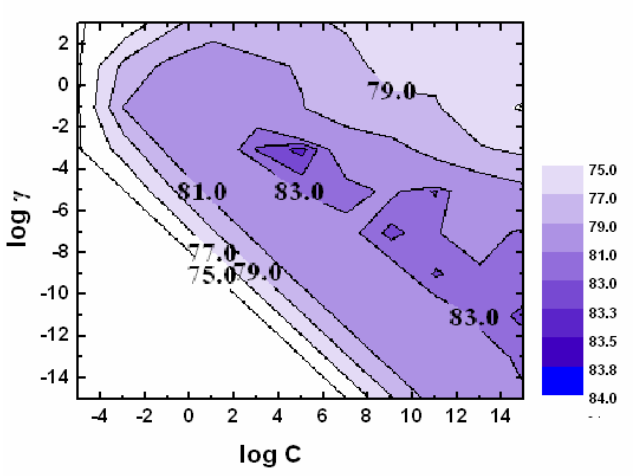

Figure 4. Grid Search

The same best parameters $C=2^5$ and $g=2^{-3}$ were obtained for all the features sets used: adult_13, adult_10, adult_6 and adult_eig showing the efficiency of the PCA analysis in identifying the most important variables of the data set for the specified problem.

5 Experiments and Results

Experiments were run using svm-train and svm-predict from LIBSVM [15] on six datasets derived from the adult dataset. First dataset, named adult, contains all the 32561 instances for training and 16281 instances for testing and all 14 variables from the initial dataset. The next data sets used where the ones described in subsection 4.1. adult_13, adult_10, adult_6 and adult_eig. For the last data set a filtering routine was implemented to remove the instances with unknown values and named adult_mis. This final set contains 30162 training instances and 15060 testing instances. Scaling was applied to the 6 datasets with the same ranges for training and testing. The SVM learning algorithm with parameters $C=2^5$ and $g=2^{-3}$ was separately trained for all the 6 datasets. The timing performance and the number of support vectors are shown in Tble 2. The total number of support vectors is similar for the 6 datasets and also similar with the one reported in [16]. Excluding one feature or the instances with missing value does not speed the training process. The training time for the SVM learning algorithm decreased significantly (2.5 times) for the adult_6 dataset which contains only 6 features, compared to the initial dataset. In the same time the accuracy drops only 0.5%.

Table 2. Training Time and Number of SV

	SVM time (s)	# of SV	# of bound SV
adult	932.0	11155	10632
adult_13	876.0	11582	11150
adult_10	714.3	12966	12671
adult_6	377.7	11687	11591
adult_eig	792.3	11386	11010
adult_mis	898.3	10584	9981

The resulting classifiers are compared in terms of accuracy and area under the curve for the receiver operating characteristics (ROC) (Table 3). The ROC curve is the plot of the probability of correct classification rate versus the probability of false-positive and it displays the model's discriminatory performance across a spectrum of thresholds.

Table 3. Classifier accuracy and area under the ROC curve

	Accuracy %	AUC
adult	84.9272	0.8911
adult_13	84.4481	0.8858
adult_10	81.8009	0.8361
adult_6	84.4174	0.8799
adult_eig	84.8900	0.8866
adult_mis	84.3900	0.8893

There are some interesting observations to make. The accuracy for adult_eig is almost the same with the accuracy obtained for the initial dataset and it is better that the one for adult_13. This happens because the PCA components capture most of the information available in the initial features.

The accuracy for adult_mis is less than the accuracy for the entire data set which means that SVM deals very well with missing data. Thus, it is better in terms of time and accuracy to reduce the data vertically and not horizontally.

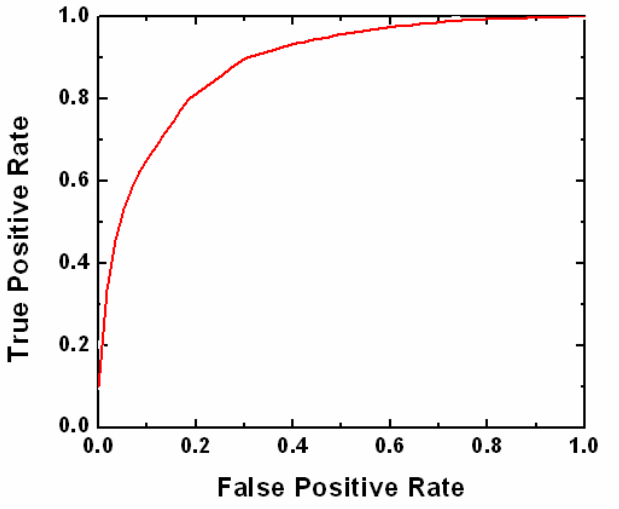

Figure 5. Adult ROC Curve

6 Conclusions and Future Work

In this paper, the effects of data reduction on the classification results of the SVM algorithm are presented. The training time and the performance of a SVM classifier where compared for six different subsets of the adult data set. In spite of the vertically reduced datasets, good classification accuracy was obtained in faster time. The conclusion is very important especially for datasets with a large number of variables that can be reduced by using the PCA method.

The classification error for the adult dataset is relatively large (~15%) because multiple similar instances have different decision values. Future work will identify these instances in a new separate class of unclassified instances. Also the classification rate can be improved by using the kernel PCA method. This method consists in the classical PCA method applied to the kernel feature space. A speed up of the learning algorithm should result since most of the calculations are done at the kernel level.

References

[1] Trafalis, T.B., Santosa B., Richman, M.B.: Tornado Detection with Kernel-Based Classifiers From WSR-D88 Radar. Submitted to: Darema, F. (ed.) Dynamic Data Driven Application Systems, Kluwer, 2004.

[2] H. Nuncz, C. Angulo and A. Catala, "Rule Extraction from Support Vector Machine", Proccedings of ESANN'2002 – European Symposium on Artificial Neural Networks, Bruges (Belgium), pp. 107-112, 24-26 April 2002.

[3] G. M. Fung,, O. L. Mangasarian, and J. W. Shavlik, "Knowledge-based Nonlinear Kernel Classifiers", Data Mining Institute Technical Report 03-02. Computer Sciences Department, University of Wisconsin, 2003.

[4] J. Wang and C. Zhang, "Support Vector Machines Based on Set Covering", Proc. Of the 2nd International Conference on Information Technology for Application (ICITA), Harbin, China, January 2004.

[5] P. S. Bradley and O. L. Mangasarian, "Feature Selection via Concave Minimization and Support Vector Machines". In Machine Learning Proceedings of the Fifteenth International Conference(ICML '98), J. Shavlik, editor, Morgan Kaufmann, San Francisco, California, 82-90, 1998.

[6] U.S. Census Bureau, United States Department of Commerce. Retrieved from http://www.census.gov/ on 11/16/03.

[7] T. Joachims, "Making large-scale SVM learning practical.", In B. Scholkopf, C. J. C. Burges and A. j. Smola, editors, *Advances in Kernel Methods – Support Vector Learning*, pp. 169-184, MIT Press, , Cambrige, MA, 1999.

[8] J. Platt, "Sequential Minimal Optimization: A Fast Algorithm for Training Support Vector Machines". Technical Report MSR-TR-98-14, Microsoft Research, 1998.

[9] V. N. Vapnik, *The Nature of Statistical Learning Theory*, 2nd edition, Springer-Verlag, New York, NY, 1999.

[10] B. Schölkopf and A. Smola, *Learning with Kernels*. MIT Press, Cambridge Massachusetts, 2002.

[11] N. Cristianini and J. Shawe-Taylor, *An Introduction to Support Vector Machines and Other Kernel-based Learning Methods*, Cambridge University Press, Cambridge, England, 2000.

[12] I.T. Jolliffe *Principal Component Analuysi*, Springer-Verlag, New-York, NY, 1986.

[13] The UCI ML and KDD Archive. Retrieve from http://www.ics.uci.edu/~mlearn/MLRepository.html and http://kdd.ics.uci.edu on 06/06/04, Irvine, CA: University of California, Department of Information and Computer Science.

[14] F. Friedrichs and C. Igel, "Evolutionary Tuning of Multiple SVM Parameters", Proc of the 12th European Symposium on Artificial Neural Networks (ESANN), Evere, Belgium, April, 2004.

[15] C.-C. Chang, and C.-J. Lin, "LIBSVM: a library for support vector machines", Software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm, 2001.

[16] Y.-J. Lee and O.L. Mangasarian, "RSVM: Reduced Support Vector Machines", Proc. Of the First SIAM International Conference on Data Mining, Chicago, April 5-7, 2001.