STATISTICS,OPTIMIZATION AND INFORMATION COMPUTING
Stat., Optim. Inf. Comput., Vol. **2**, June 2014, pp 161–175.
Published online in International Academic Press (www.IAPress.org)

Supersaturated plans for variable selection in large databases

C. Parpoula [1], C. Koukouvinos [1,*], D.E. Simos [2] and S. Stylianou [3]

[1] *Department of Mathematics, National Technical University of Athens, Athens, Greece*
[2] *SBA Research, 1040 Vienna, Austria.*
[3] *Department of Mathematics, University of the Aegean, Samos, Greece*

Received 7 April 2014; Accepted 22 May 2014
Editor: Paulo Canas Rodrigues

Abstract Over the last decades, the collection and storage of data has become massive with the advance of technology and variable selection has become a fundamental tool to large dimensional statistical modelling problems. In this study we implement data mining techniques, metaheuristics and use experimental designs in databases in order to determine the most relevant variables for classification in regression problems in cases where observations and labels of a large database are available. We propose a database-driven scheme for the encryption of specific fields of a database in order to select an optimal supersaturated design consisting of the variables of a large database which have been found to influence significantly the response outcome. The proposed design selection approach is quite promising, since we are able to retrieve an optimal supersaturated plan using a very small percentage of the available runs, a fact that makes the statistical analysis of a large database computationally feasible and affordable.

Keywords Association rule mining, Design of Experiments,
Large Dimensional Data, Metaheuristics, Sensitivity Analysis,
Support Vector Machines, Variable Selection

AMS Subject Classification: 62-07, 62K15, 62P10

DOI: 10.19139/soic.v2i2.75

*Correspondence to: Department of Mathematics, National Technical University of Athens, Zografou 15773, Athens, Greece. Email:ckoukouv@math.ntua.gr

ISSN 2310-5070 (online) ISSN 2311-004X (print)

1. Introduction

The advent of new technologies has enabled scientists to measure the class label of hundreds of variables simultaneously and large dimensional problems are becoming more and more common since large amounts of data are increasingly produced and stored. Therefore, factor screening has become a challenge that many statisticians face in large-dimensional problems, and an essential activity in which the main goal is to identify correctly and parsimoniously the factors that have an important influence on the measured response.

Large databases exist in diverse fields of science, and extensive research into variable selection has been carried out over the last decades (see, for example [6] and [18]). Stepwise deletion and subset selection [21] are some of the existing traditional variable selection techniques which are useful for exploratory investigations but are very time-consuming or even impossible in cases where the number of predictor variables of interest is large. Variable selection procedures via penalized likelihood (see, for example [5] and [16]) are easily and quickly implemented even in a large-dimensional problem, but they remain very time-consuming when they are applied during a large dimensional statistical analysis. This computational difficulty prevents these methods from being widely used when there is a large number of predictors in real life problems.

In this paper, we extend the idea of using heuristic algorithms for variable selection from the data of a database, as presented in [15]. In this paper, we deal with a large-dimensional statistical modelling problem, and study the variable selection issue considering an alternative approach. We propose a step-by-step database-driven design selection scheme for the encryption of specific fields of a database which correspond to the significant variables of a regression problem in cases where observations and labels of a database are available. The proposed data-driven scheme is a combination of metaheuristics and data mining techniques, and enables the experimenter to identify the optimal supersaturated plan retrieved from a database for variable selection purposes. The close interaction between data mining and statistical analysis enabled the successful transition from collecting data, through modeling the underlying structures, to understandable and profitable results.

The rest of this paper is organized as follows. In Section 2, we discuss the need of considering design of experiments, and specifically supersaturated designs (SSDs) for variable selection issues. In Section 3, we describe the statistical methods employed in this work. We also present the proposed method of identifying an optimal supersaturated plan given a database. In Section 4, we describe the criteria used for performance evaluation; all the above procedures are applied to the real medical data, and the merits of the alternative approach using SSDs are presented. Finally in Section 5, the obtained results are discussed and some concluding remarks are made.

2. The use of SSDs for variable selection

Recently the experimental designs have been used for variable selection purposes. Pumplün et al. (see, [24, 25]) introduced the use of experimental designs for variable selection problems given a database of observations However, in that case the retrieved plan arose from the class of D-optimal designs while in our case we are interested in supersaturated designs. Schiffner and Weihs [27] extended the study of [24] and investigated the appropri- ateness of D-optimal plans for training classification methods. Rüping and Weihs [26] dealt with the variable selection issue given a database of observations using statistical design of experiments and kernel methods.

Since nowadays massive data sets become available without predefined purposes, it is usually preferable to identify some important features in the data sets that will provide valuable information to support decision making [18]. For situations where there is no prior knowledge of the factor effects, but factor sparsity holds [1], and where the aim is to identify any dominant factors, experimenters should seriously consider using SSDs as suggested in [9]. SSDs are widely used in experimental situations in which a large number of factors are studied and only a few of them are expected to influence significantly the measured response. SSDs can be generally described as fractional factorial designs in which the number of factors m to be estimated exceeds the number of experimental runs n ($m \geq n$). Recent research has targeted on the class of SSDs due to their mathematical novelty and their run-size economy. Parpoula et al. [22] presented a new variable selection approach inspired by SSDs given a dataset of observations, and dealt with the large dimensional statistical modelling problem by employing nonconcave penalized likelihood methods and best subset techniques. Since this work focuses on the idea of implementing SSDs for variable selection issues, we do not present here more details on construction and analysis methods of SSDs, and we refer the interested reader to recent reviews (see, for example [9, 14] for construction methods of SSDs; see, for example [17, 11, 8] for analysis methods of SSDs). The interested reader may also refer to [9, 13] regarding the practical use of SSDs in real life problems.

3. The employed methods

3.1. Association rule mining

Association rule mining finds interesting associations and/or correlation relationships among large set of data items. Association rules show attribute value conditions that occur frequently together in a given data set. Association rules provide information of this type in the form of "if-then" statements. In addition to the antecedent (the "if" part) and the consequent (the "then" part), an association

rule has two criteria that express the degree of uncertainty about the rule. The first criterion is called *support* which refers to the percentage of records in the training data for which the antecedents (the "if" part of the rule) are true. The other criterion is known as *confidence* which is based on the records for which the rule's antecedents are true, and is the percentage of those records for which the consequent(s) are also true. In other words, it is the percentage of predictions based on the rule that are correct; rules with lower confidence than the specified criterion are discarded.

Generalized Rule Induction (GRI) algorithm is the most suitable methodology for our study because it can handle categorical or numerical variables as inputs, and requires categorical variables as outputs. GRI applies an information-theoretic approach [28] to determine the interestingness of a candidate association rule using the quantitative measure J GRI uses this quantitative measure J to calculate how interesting a rule may be and uses bounds on the possible values this measure may take to constrain the rule search space. The association rules generated from GRI take the form "If $X = x$ then $Y = y$" where X and Y are two fields (attributes) and x and y are values for those fields. GRI extracts rules with the highest information content based on the J index that takes both the generality (support) and accuracy (confidence) of rules into account.

This preliminary stage of the statistical analysis is very important since it enables the experimenter to locate fields and records that are most likely to be of interest in modeling, and create a database-driven scheme for the encryption of specific fields of a database.

3.2. Simple genetic algorithm

In this paper, we assume some basic familiarity with genetic algorithm concepts like reproduction, mutation and crossover and will only describe what is needed for our approach. The interested reader may refer to Goldberg [10] and to Davis [4] for more details concerning with the necessary concepts for a description of the Simple Genetic Algorithm (SGA).

In this paper, an SGA is implemented. We correspond records of the database to chromosomes of the SGA where we are interested in the case where the chromosome length is very small when compared to the total number of attributes of the database. The database is encoded in $\{-1, 1\}$ values, and we use the same encoding for SGA. Every optimization algorithm depends on a certain objective function (OF) that needs to be optimized. For SSDs, a widely used optimality criterion is the r_{max} criterion. We give the definition, below.

r_{max} **criterion** A reasonable criterion for comparing supersaturated designs is the minimization of $max_{i<j}|s_{ij}/n|$ where $s_{ij}/n = r_{ij}$ is the correlation of two columns $\mathbf{c}_i$, $\mathbf{c}_j$ The largest absolute value of r_{ij} between all pairs of columns is denoted by r_{max} More details can be found in Lin [19].

3.3. L_1-norm support vector machine

Support vector machines (SVMs) are a supervised learning classification method based on ideas originated in statistical learning theory [30, 3]. SVMs use the training data in order to generate input-output mapping functions and determine the maximal margin hyperplane. Since in our study we deal with a binary classification problem, the optimal hyperplane in terms of classification performance, is the one with the maximal margin of separation between the two classes [30]. SVMs can also be formulated as a regularized function estimation problem, corresponding to a hinge loss function plus a regularization term on the fitted coefficients [34]. The Least Absolute Shrinkage and Selection Operator (LASSO) method [29] is one of the most common approaches in regression for parameter estimation. The L_1 penalty term (LASSO) was adapted to SVM methodology in order to perform automatical variable selection to classification problems (see, for example [2, 35, 7]). Recently the elastic net penalty term [36] was adapted to SVMs by using a mix of the L_1-norm and the L_2-norm penalties (see, [31, 32]). The elastic net SVM is especially useful for cases in which the number of variables exceeds the sample size. The L_1-norm SVM is suitable for our real data analysis, since the dimension of the data (m=44 input variables) is not larger than the number of training samples (n=8862 observations).

3.4. The proposed method

In this section, we present analytically the proposed method for harvesting an optimal super- saturated design from the records and specific fields (attributes) of a database. The proposed method proceeds as follows:

1. Let y_i denote the i-th response in the data set and x_i denote the m vector of explanatory variables of the data set.
2. Split the data set into training (90%) and test (10%) set. For the partitioning, the total observations were randomly selected to create the training and test set, according to their predefined size.
3. Perform GRI algorithm to $(x_i, y_i)_{i=1,\ldots,n}$ and generate the association rules which are used for initializing the random chromosomes that are used in the mating pool of an SGA. In particular, apply the GRI mining task iteratively i.e., run an "up" search on the entire training set and then run a "down" search on the remainder to weed out low-performing segments.
4. Select the initial runs for the optimal plan by matching the generated GRI rules with fields of the database. If n_d are the runs retrieved using the GRI rules then the algorithmic procedure proceeds by fixing the total number of runs n of the supersaturated design as $n = n_d + n_h$ where n_h are the runs that are being selected from the SGA.
5. Use 1-point crossover and keep track of the selected attributes using the GRI rules without changing their values.

6. Begin a heuristic search using as an initial seed the selected factors of the optimal plan. In particular, guide the heuristic to search for k factors, where a subset of them is always keep fixed and the others are randomized.
7. Retrieve the remaining factors of the optimal plan subject to r_{max} optimality criterion. The implemented SGA outputs an optimal plan belonging to the class of supersaturated designs when a value of r_{max} <1 is detected.

4. Application and experimental results

4.1. Medical data

The Trauma data set used here was collected in an annual registry conducted during the period 01/01/2005 - 31/12/2005 by the Hellenic Trauma and Emergency Surgery Society involving 30 General Hospitals in Greece. Altogether, 8862 patients were recorded and for each of them the binary response variable y taking only two possible outcomes, denoted by - 1 and 1 for "survival"and "death", respectively was reported. The Trauma data set which is used for further analysis, includes all of the 8862 available patients and the 44 input explicative variables (see Appendix, Table 3 for the list of the variables), that include demographic, transport and intrahospital data. After medical advice, all of the factors are treated equally during the data mining approach, meaning that there was no factor that should be always maintained in the model. The data set was split into training (90%) (=7975 patterns) and test (10%) (=887 patterns) sets for classification and association analysis. All results were derived from SPSS Clementine 12.0 and MATLAB software.

4.2. Performance criteria

Assessing the reliability of a classifier is essential to ensure data quality The most common criterion to assess the quality of a classification model is discrimination which measures how well the two classes in the data set are separated [33]. We consider the most commonly used measures of discrimination for evaluating the performance of the employed method To discuss these performance criteria, we adopt the standard definitions used in binary classification; given a classifier and a record, there are four possible scenarios. Positive records are correctly predicted as positive (True Positive-TP), positive records are incorrectly identified as negative (False Negative-FN), negative records are classified as positive ones (False Positive-FP) and finally negative records are correctly identified as negative (True Negative-TN).

The classification accuracy is used as first criterion. Accuracy is defined as the percentage of correct classified records in the test set for every used method. The other two criteria used are the sensitivity and specificity which are two statistical

measures of the performance of a binary classification test and are closely related to the concepts of Type I and Type II errors. Sensitivity measures the proportion of actual positives which are correctly identified as such whereas specificity measures the proportion of negatives which are correctly identified. The other used criteria in information retrieval are the recall which corresponds to sensitivity and precision which is the proportion of true positives among all predicted positives. In classification tasks, another widely used performance metric is the geometric mean of class accuracies which puts all classes on an equal footing, and does not give higher priority to the rare positive classes. A performance metric that allows for this is the F-measure, which does not take account of performance on the negative classes.

In two-class problems, the above mentioned performance measures are defined as follows:

- Accuracy = $\frac{TP+TN}{TP+FP+TN+FN}$
- Sensitivity (Recall) = $\frac{TP}{TP+FN}$
- Specificity = $\frac{TN}{TN+FP}$
- Precision = $\frac{TP}{FP}$
- G-mean = $\sqrt{\text{sensitivity} * \text{specificity}}$
- F - measure = $\frac{\text{precision}*\text{recall}}{\beta\text{precision}+(1-\beta)\text{recall}}$

where the β parameter, $0 < \beta < 1$, allows the user to assign relative weights to precision and recall, with 0.5 giving them equal importance.

Another popular statistical tool is the Receiver Operating Characteristic (ROC) curve [23] which by definition is used to evaluate the performance of a system with dichotomous outcomes. An ROC curve is presented as a plot of Sensitivity as a function of (1-Specificity) for all the possible cutoffs. Traditionally the Area Under the ROC Curve (AUC) is used as a summary index of test accuracy [12] and is useful as a descriptive of overall test performance. Statistically speaking, the AUC of a classifier is the probability that a classifier will rank a randomly chosen positive instance higher than a randomly chosen negative instance.

As with many decision problems, errors of various types must be balanced against costs. In screening designs, there is a cost of declaring an inactive factor to be active (Type I error), and also a cost of declaring an active effect to be inactive (Type II error). Type II errors are troublesome, as addressed in [20], as well as Type I errors, since they can result in unnecessary cost in follow-up experiments and can cause detrimental actions if the experiment has immediate impact on practice. Under situations of effect sparsity Type I errors are very likely to occur.

Sensitivity and Specificity can be alternatively described as follows:

$$\text{Sensitivity} = \text{P}(\widehat{\text{y}} = 1|\text{y} = 1) = 1 - \text{Type II error}$$

$$\text{Sensitivity} = \text{P}(\widehat{\text{y}} = 0|\text{y} = 0) = 1 - \text{Type I error}$$

4.3. The optimal supersaturated plan

For initializing the random chromosomes that were used in the mating pool of an SGA, we employed the following GRI rules.

GRI	Rules Antecedents	Consequent	Support	Confidence
If	x1 = 1.0 and x3 = 1.0 and x14 = 1.0 and x15 = 1.0 and x16 = 1.0	then y=1.0	0.21	94.12
If	x14 = 1.0 and x16 = 1.0	then y=1.0	0.49	94.87
If	x5 = 1.0 and x15 = 1.0 and x16 = 1.0	then y=1.0	0.21	94.12

Thus, we selected three initial runs for the optimal plan by matching these rules with fields of the database. We give below the optimal plan selected using the GRI rules and the SGA in 6 runs and 8 factors. This is interpreted as selecting 6 records with 8 field attributes.

record ID	x8	x5	x7	x13	x6	x2	x14	x11	y
968	1	-1	-1	1	-1	-1	1	-1	1
936	-1	1	-1	1	-1	-1	1	-1	1
5587	1	1	-1	1	-1	-1	1	1	1
7344	-1	1	-1	1	-1	-1	-1	-1	-1
3806	-1	1	-1	-1	-1	1	-1	-1	-1
2530	-1	-1	-1	-1	1	-1	-1	-1	-1

The plan formed by [x8,x5,x7,x13,x6,x2,x14,x11] has a value of r_{max}= 0.667.

We note here that the SGA also detected another optimal plan belonging to the class of SSDs with a value of r_{max} <1. We give below the second optimal plan selected using the GRI rules and the SGA in 6 runs and 8 factors. This is also interpreted as selecting 6 records with 8 field attributes.

record ID	x8	x5	x7	x13	x6	x2	x16	x11	y
968	1	-1	-1	1	-1	-1	1	-1	1
936	-1	1	-1	1	-1	-1	1	-1	1
5587	1	1	-1	1	-1	-1	1	1	1
7344	-1	1	-1	1	-1	-1	-1	-1	-1
3806	-1	1	-1	-1	-1	1	-1	-1	-1
2530	-1	-1	-1	-1	1	-1	-1	-1	-1

The plan formed by [x8,x5,x7,x13,x6,x2,x16,x11] has also a value of $r_{max} = 0.667$.

Note here that the $x7$ column, in both design matrices, is fully aliased to the mean since all of its values are equal to - 1. Hence, its effect is not estimable. This fact is not troublesome for our study which is designed for screening purposes. We observe that the only difference between the first and second optimal plan retrieved from the database is the selection of the 7-th column, i.e., $x14$ instead of $x16$ respectively Since the ± 1 signs are identically allocated to both $x14$ and $x16$ columns of the respective design matrices, the derived results will be totally identical either we consider to employ the first or the second detected optimal plan for further analysis. The fact that both plans retrieved from the database have identical ± 1 signs allocated to each column of the design matrix is crucial, since it provides strong evidence that the desired optimal plan is indeed detected correctly (either the plan is formed by [x8,x5,x7,x13,x6,x2,x14,x11] or by [x8,x5,x7,x13,x6,x2,x16,x11]).

4.4. Comparative results

The L_1-norm SVM methodology is used to evaluate the performance of the proposed method. We firstly perform L_1-norm SVM method to the whole trauma dataset consisting of 8862 patients and 44 possible risk factors (*Model I*). The fact that L_1-norm SVMs are fast in training, classifying, and are not computationally time-expensive allowed us to apply this method to the whole large dimensional dataset. We then perform L_1-norm SVM to the proposed method, using as design matrix the identified $SSD_{(6,8)}$and as response its corresponding outcome class labels (*Model II*).

Table 1: Advanced comparison of models performance

Criterion	*Model I*	*Model II*
Training error	0.04%	0.01%
Accuracy	96%	99%
Sensitivity	20%	100%
Recall	20%	100%
Specificity	99%	100%
Precision	68%	100%
G-mean	45%	100%
F-measure	31%	100%
AUC	0.57	0.88

A perfect predictor would be described as 100% sensitive and 100% specific. Sensitivity and specificity relate to the test's ability to identify positive and negative results, respectively. A test with high sensitivity and high specificity can be considered as a reliable indicator of a test with which has a low Type II error rate and a low Type I error rate, respectively.

The L_1-norm SVM (*Model I*) reaches the percentage of 99% for specificity which means that the classifier recognizes almost all actual negatives; in other words this means that it has a low Type I error rate. This measure alone does not tell us how well the classifier recognizes positive cases and so it is necessary to take into consideration both sensitivity of the used classifier. When the L_1-norm SVM (*Model I*) is evaluated against the sensitivity has a clear disadvantage having lowest percentages, which means that the Type II error rates are higher. In general, Table 1 shows that the L_1-norm SVM (*Model I*) tends to declare at a lower rate inactive variables to be active, and at a higher rate active variables to be inactive. The AUC for *Model I* takes the lowest value (AUC_{ModelI}=0.57) compared to *Model II* ($\text{AUC}_{ModelII}$=0.88).

The L_1-norm SVM applied to the proposed method (*Model II*) has clearly better clas- sification accuracy sensitivity and specificity which almost reaches the absolute percentage of 100%. In general, Table 1 shows that the proposed method (*Model II*) has very low Type I errors (Type I error rate is almost 0, and this corresponds to cases where almost none of the inactive effects are declared as active) and very low Type II errors (Type II error rate is almost 0, and this corresponds to cases where almost none of the active effects are detected wrongly). In other words this means that the proposed method tends to declare at a low rate inactive variables to be active, and active variables to be inactive. Therefore, the proposed method is indeed stable in this sense.

Sensitivity and specificity values alone cannot be used to determine whether a test is useful in practice, and may be misleading. In medical diagnostics, sensitivity is the ability of a test to correctly identify those with the disease (true positive rate), whereas specificity is the ability of the test to correctly identify those without the disease (true negative rate). The "worst-case" sensitivity or specificity must be calculated in order to avoid reliance on experiments with few results. A common way to do this is to calculate the binomial proportion confidence intervals for sensitivity and specificity giving the range of values within which the correct value lies at a given confidence level (95%). Table 2 shows that sensitivity and specificity values for both *Model I* and *Model II* are satisfactory since the estimated values lie at the corresponding estimated confidence interval. Note here that the three basic things that usually impact the width of a confidence interval are the confidence level, variability and sample size. The fact that the confidence intervals in Table 2, for *Model II* are much broader than *Model I* is not surprising since smaller sample sizes generate wider intervals.

Table 2: Confidence Interval (95%)

	Sensitivity			Specificity		
Model	Estimated Value	Lower Limit	Upper Limit	Estimated Value	Lower Limit	Upper Limit
Model I	0.994891	0.993057	0.996255	0.204036	0.168201	0.245076
Model II	0.999999	0.309989	1.00000	0.999999	0.309989	1.000000

The L_1-norm SVM applied to the whole dataset (*Model I*) detected a set of 33 out of 44 variables as statistically significant. The generated *Model I* excluded 11 variables as unimportant, i.e., $x19$, $x20$, $x23$, $x24$, $x29$, $x30$, $x31$, $x38$, $x40$, $x42$, $x43$. The proposed method (*Model II*) also succeeded to exclude these 11 variables as unimportant. Moreover, the proposed method (*Model II*) achieved to exclude more unimportant variables leading to a more parsimonious model compared to *Model I*. The proposed method (*Model II*) detected a set of 8 out of 44 variables as statistically significant i.e., $x2$, $x5$, $x6$, $x7$, $x8$, $x11$, $x13$, $x14$ (or $x16$). Since the decision between $x14$ and $x16$ seems to be arbitrary and in order to avoid excluding a variable and missing probably a very important factor we further performed a subsequent analysis examining three new models, firstly including only $x14$, secondly including only $x16$ and finally a model including both variables (x14 and x16). The derived results are presented below.

4.5. Subsequent Analysis

The L_1-norm SVM methodology is also used to evaluate the performance of the new models considered for subsequent analysis. We firstly perform L_1-norm SVM method to the dataset consisting of m= 8 possible risk factors, i.e., [x8,x5,x7,x13,x6,x2,x14,x11] and n= 8862 patients (*Model A*). We secondly perform L_1-norm SVM method to the dataset consisting of m= 8 possible risk factors, i.e., [x8,x5,x7,x13,x6,x2,x16,x11] and n= 8862 patients (*Model B*). We finally perform L_1-norm SVM method to the dataset consisting of m= 9 possible risk factors, i.e., [x8,x5,x7,x13,x6,x2,x14,x16,x11] and n= 8862 patients (*Model C*).

Table 3: Advanced comparison of models performance

Criterion	*Model A*	*Model B*	*Model C*
Training error	0.04976 %	0.04513 %	0.04513 %
Accuracy	0.95024 %	0.95486 %	0.95486 %
Sensitivity	0.02017 %	0.11211 %	0.11435 %
Recall	0.02017 %	0.11211 %	0.11435 %
Specificity	0.99952 %	0.99952 %	0.99941 %
Precision	0.69231 %	0.92593 %	0.91071 %
G-mean	0.14202 %	0.33475 %	0.33806 %
F-measure	0.03921 %	0.20000 %	0.20319 %
AUC	0.70096	0.69024	0.68281

Table 4: Confidence Interval (95%)

	Sensitivity			Specificity		
Model	Estimated Value	Lower Limit	Upper Limit	Estimated Value	Lower Limit	Upper Limit
Model A	0.02017	0.009879	0.039343	0.99952	0.998694	0.999848
Model B	0.11211	0.085092	0.145999	0.99952	0.998694	0.999848
Model C	0.11435	0.087064	0.148470	0.99941	0.998527	0.999781

We observe from Table 3 that *Model B* or *Model C* clearly outperform *Model A* in terms of training error, sensitivity recall, precision, G-mean and F-measure values. *Model B* and *Model C* have similar performance with small differences in values of the examined criteria. Moreover, we observe from Table 4 that sensitivity and specificity values for all three models are satisfactory since the estimated values lie at the corresponding estimated confidence interval. Since the decision between selecting *Model B* (including only $x16$) or *Model C* (including both $x14$ and $x16$) seems not to be clear and might be arbitrary due to the similar performance of both models, we recommend selecting *Model C* as the "best" model derived, in order to avoid excluding a variable and missing probably an important factor.

5. Concluding Remarks

The innovative nature of our study lies on using the class of SSDs in conjunction with data mining methods and genetic algorithms that enabled us to deal with the problem of variable selection in a large-dimensional database with a feasible computational effort. The proposed method achieved to declare at a low rate inactive variables to be active, and active variables to be inactive. Therefore, the proposed method is indeed stable in this sense. The proposed method using SSDs is very important for the statistical analysis of large data, since it allowed us to identify effectively and parsimoniously the important prognostic factors (9 statistically significant out of the available 44 predictor variables) using only few runs (6 runs of the available 8862 runs). We recommend this alternative approach for variable selection purposes given a database of observations since it enables the experimenter to use only a very small percentage of the available runs, a fact that makes the statistical analysis of a large database computationally feasible and affordable. The proposed approach might be preferable to practitioners who find it expensive to consider levels of certain factors which are associated with high costs and would like to achieve specific levels of certain factors in order to minimize the experimental cost. In the proposed method, only main effects models have been considered. It will naturally be of interest to incorporate interaction effects in the models, and then to develop a suitable method for subsequent analysis. Work is currently under progress in this direction and we hope to report these findings in a future paper.

Appendix

Table 5: Trauma Study

Covariates
$x1$: ambulance (-1 = no, 1 = yes)
$x2$: comorbidities (-1 = no, 1 = yes)
$x3$: sex (-1 = female, 1 = male)
$x4$: evacuation (-1 = no, 1 = yes)
$x5$: major doctor (-1 = no, 1 = yes)
$x6$: A.T.L.S (-1 = no, 1 = yes)
$x7$: capillary refill (-1 = no, 1 = yes)
$x8$: pale (-1 = no, 1 = yes)
$x9$: intrahospital transport (-1 = no, 1 = yes)
$x10$: dysphoria (-1 = no, 1 = yes)
$x11$: central cyanosis (-1 = no, 1 = yes)
$x12$: peritoneum points (-1 = no, 1 = yes)
$x13$: oxygen (-1 = no, 1 = yes)
$x14$: intubation (-1 = no, 1 = yes)
$x15$: mechanical ventilation (-1 = no, 1 = yes)
$x16$: cardiopulmonary resuscitation (-1 = no, 1 = yes)
$x17$: chest drainage (-1 = no, 1 = yes)
$x18$: pericardiocentesis (-1 = no, 1 = yes)
$x19$: catheter (-1 = no, 1 = yes)
$x20$: nasogastric tube (-1 = no, 1 = yes)
$x21$: collar (-1 = no, 1 = yes)
$x22$: spinal immobilisation (-1 = no, 1 = yes)
$x23$: pelvic immobilisation (-1 = no, 1 = yes)
$x24$: limb immobilisation (-1 = no, 1 = yes)
$x25$: fluids (-1 = no, 1 = yes)
$x26$: blood (-1 = no, 1 = yes)
$x27$: ICP monitoring (-1 = no, 1 = yes)
$x28$: thoracotomy (-1 = no, 1 = yes)
$x29$: angiography (-1 = no, 1 = yes)
$x30$: embolism (-1 = no, 1 = yes)
$x31$: diagnostic peritoneal lavage (DPL) (-1 = no, 1 = yes)
$x32$: gases (vacuum phenomenon) (-1 = no, 1 = yes)
$x33$: Radiograph E.R. (-1 = no, 1 = yes)
$x34$: computed tomography (CT) (-1 = no, 1 = yes)
$x35$: ultrasound (US) (-1 = no, 1 = yes)
$x36$: urea testing (-1 = no, 1 = yes)
$x37$: toxicology testing (-1 = no, 1 = yes)
$x38$: surgical intervention (-1 = no, 1 = yes)
$x39$: intrahospital CT (-1 = no, 1 = yes)
$x40$: intrahospital US (-1 = no, 1 = yes)
$x41$: intrahospital M.R.I (-1 = no, 1 = yes)
$x42$: intrahospital angiography (-1 = no, 1 = yes)
$x43$: complications (-1 = no, 1 = yes)
$x44$: Intensive Care Unit (ICU) stay (-1 = no, 1 = yes)

Acknowledgments

The authors would like to thank the First Propedeutic Surgical Clinic, in Hippocratio Hospital for giving the real medical data. The authors would also like to thank the Associate Editor and the referees for their constructive and useful suggestions which resulted in an improvement on an earlier version of this manuscript. The research of the first author was financially supported by a scholarship awarded by the Secretariat of the Research Committee of National Technical University of Athens. The work of the third author was carried out

during the tenure of an ERCIM "Alain Bensoussan" Fellowship Programme. This Programme is supported by the Marie Curie Cofunding of Regional, National and International Programmes (COFUND) of the European Commission. In addition, the research of the third author was funded by COMET K1, FFG Austrian Research Promotion Agency.

REFERENCES

1. G.E.P Box and R.D. Meyer (1986). An analysis for unreplicated fractional factorials, *Technometrics* **28**, 11-18.
2. P.S. Bradley and O.L. Mangasarian (1998). Feature Selection via Concave Minimization and Support Vector Machines, In Shavlik, J. (ed.) *Machine Learning Proceedings of the Fifteenth International Conference(ICML '98)*, Morgan Kaufmann, San Fransisco, CA, 82-90.
3. C.J.C. Burges (1998). A tutorial on Support Vector Machines for Pattern Recognition, *Data Mining and Knowledge Discovery*, **2**, 121-167.
4. L. Davis (1991). *Handbook of Genetic Algorithms*, Van Nostrand, Reinhold.
5. J. Fan and R. Li (2001). Variable selection via nonconcave penalized likelihood and its oracle properties, *J. Amer. Statist. Assoc.*, **96**, 1348-1360.
6. J. Fan and R. Li (2006). Statistical challenges with high dimensionality: feature selection in knowledge discovery *Proceedings of the International Congress of Mathematicians* ,(M. Sanz-Sole, J. Soria, J.L. Varona, J. Verdera, eds.), Vol. III, 595-622.
7. G. Fung and O.L. Mangasarian (2004). A feature selection newton method for support vector machine classification, *Comput. Optim. Appl. J.,* **28**, 185-202.
8. S.D. Georgiou (2014). Supersaturated designs: A review of their construction and analysis. *Journal of Statistical Planning and Inference* ,**144**, 92-109.
9. S.G. Gilmour (2006). Factor Screening via Supersaturated Designs, In: A. Dean, and S. Lewis, (Eds.), *Screening Methods for Experimentation in Industry, Drug Discovery, and Genetics*, Springer-Verlag, New York, 169-190.
10. D.E. Goldberg (1989). *Genetic Algorithms in Search, Optimization, and Machine Learning*, Addison-Wesley Reading, MA.
11. S. Gupta and P Kohli (2008). Analysis of supersaturated designs: a review, *Journal of Indian Society of Agricultural Statistics*, **62**, 156-168.
12. J.A. Hanley and B.J. McNeil (1982). The meaning and use of the area under a receiver operating characteristic (ROC) curve, *Radiology*, **143**, 29-36.
13. D.R. Holcomb, D.C. Montgomery and W.M. Carlyle (2007). The use of supersaturated experiments in turbine engine development, *Quality Engineering*, **19**, 17-27.
14. B. Kole, J. Gangwani, V.K. Gupta and R. Parsad (2010). Two level supersaturated designs: a review, *Journal of Statistical Theory and Practice*, **4**, 598-608.
15. C. Koukouvinos, C. Parpoula and D.E Simos (2013). Genetic algorithm and data mining techniques for design selection in databases, *In Proceedings of the 2013 IEEE International Conference on Availability, Reliability and Security, (2013)*, 743-746, DOI: 10.1109/ARES.2013.98.
16. R. Li and D.K.J. Lin (2002). Data analysis in supersaturated designs, *Statist. Probab. Lett.*, **59**, 135-144.
17. R. Li and D.K.J. Lin (2003). Analysis methods for supersaturated designs: Some comparisons, *Journal of Data Science*, **1**, 249-260.
18. R. Li, D.K.J. Lin and B. Li (2013). Statistical inference in massive data sets, *Applied Stochastic Models in Business and Industry*, **29**, 399-409.
19. D.K.J. Lin (1993). Another look at first-order saturated designs: the p-efficient designs, *Technometrics*, **35**, 284-292.
20. D.K.J. Lin (1995). Generating systematic supersaturated designs, *Technometrics* ,**37**, 213-225.
21. A.J. Miller (2002). *Subset Selection in Regression*, Chapman & Hall/CRC, Boca Raton.

22. C. Parpoula, K. Drosou, C. Koukouvinos and K. Mylona (2014). A new variable selection approach inspired by supersaturated designs given a large-dimensional dataset, *Journal of Data Science*, **12**, 35-52.
23. M.S. Pepe (2000). Receiver operating characteristic methodology *J. Am. Statist. Assoc.*, **95**, 308-311.
24. C. Pumplün, S. Rüping, K. Morik and C. Weihs (2005a). *D-optimal plans in observational studies*, Technical Report 44/2005, SFB 475, Complexity reduction in multivariate data structures, Technische Universität Dortmund, 44221 Dortmund, Germany, URL http://www.statistik.tu-dortmund.de/sfb-tr2005.html.
25. C. Pumplün, C. Weihs and A. Preusser (2005b). Experimental design for variable selection in data bases, In C. Weihs and W. Gaul, editors, *Classification The Ubiquitous Challenge*, 192-199, Springer, Berlin.
26. S. Rüping and C. Weihs (2009). *Kernelized design of experiments*, Technical Report, Sonderforschungsbereich 475, Komplexitätsreduktion in Multivariaten Datenstrukturen, Universität Dortmund, No. 2009, 02, http://hdl.handle.net/10419/36602.
27. J. Schiffner and C. Weihs (2009). *D-optimal plans for variable selection in data bases*, Technical Report, Sonderforschungsbereich 475, Komplexitätsreduktion in Multivariaten Datenstrukturen, Universität Dortmund, No. 2009, 14, http://hdl.handle.net/10419/41052.
28. P Smyth and R.M. Goodman (1992). An Information Theoretic Approach to Rule Induction from Databases, *IEEE Trans. Knowl. Data Eng.* ,**4**, 301-316.
29. R. Tibshirani (1996). Regression shrinkage and selection via the lasso, *J. Roy. Statist.*, *Soc. Ser. B*, **58**, 267-288.
30. V. Vapnik (1998). *Statistical Learning Theory*, Wiley New York.Vapnik 1998.
31. L. Wang, J. Zhu and H. Zou (2006). The doubly regularized support vector machine, *Statistica Sinica*, **16**(2), 589-615.
32. L. Wang, J. Zhu and H. Zou (2008). Hybrid huberized support vector machines for microarray classification and gene selection, *Bioinformatics*, **24**(3), 412-419.
33. I.H. Witten and E. Frank (2005). *Data Mining: Practical Machine learning Tools and Techniques with Java Implementations*, 2nd edn, Morgan Kaufmann Publishers San Francisco.
34. G.-B. Ye, Y. Chen and X. Xie (2011). Efficient variable selection in support vector machines via the alternating direction method of multipliers, In *Proceedings of the* 14^{th} *International Conference on Artificial Intelligence and Statistics (AISTATS) 2011*, Fort Lauderdale, FL, USA, JMLR W&CP **15**, 832-840.
35. J. Zhu, S. Rosset, T. Hastie and R. Tibshirani (2003). 1-norm support vector machines, In *Advances in Neural Information Processing Systems 16, Proceedings of the 2003 Conference*.
36. H. Zou and T. Hastie (2005). Regularization and variable selection via the elastic net, *J. Roy. Statist. Soc. Ser. B*, **67**, 301-320.