**Preprints of the 19th World Congress**
**The International Federation of Automatic Control**
**Cape Town, South Africa. August 24-29, 2014**

# Root Cause Analysis by a Combined Sparse Classification and Monte Carlo Approach ⋆

**Mattia Zanon** * **Gian Antonio Susto** * and **Seán McLoone** *,**

* *Department of Electronic Engineering, National University of Ireland Maynooth, Maynooth, Ireland, (e-mail: {zanon.mattia; gianantonio.susto}@google.com)*
** *Queen's University Belfast, Belfast, Northern Ireland (e-mail: s.mcloone@ieee.org)*

**Abstract:** Classification methods with embedded feature selection capability are very appealing for the analysis of complex processes since they allow the analysis of root causes even when the number of input variables is high. In this work, we investigate the performance of three techniques for classification within a Monte Carlo strategy with the aim of root cause analysis. We consider the naïve bayes classifier and the logistic regression model with two different implementations for controlling model complexity, namely, a LASSO-like implementation with a $\ell_1$ norm regularization and a fully Bayesian implementation of the logistic model, the so called relevance vector machine. Several challenges can arise when estimating such models mainly linked to the characteristics of the data: a large number of input variables, high correlation among subsets of variables, the situation where the number of variables is higher than the number of available data points and the case of unbalanced datasets. Using an ecological and a semiconductor manufacturing dataset, we show advantages and drawbacks of each method, highlighting the superior performance in term of classification accuracy for the relevance vector machine with respect to the other classifiers. Moreover, we show how the combination of the proposed techniques and the Monte Carlo approach can be used to get more robust insights into the problem under analysis when faced with challenging modelling conditions.

*Keywords:* Logistic Regression, LASSO, Relevance Vector Machine, Naïve Bayes Classifier, Semiconductor Manufacturing, Fault Detection.

## 1. INTRODUCTION

As computers and sensors advance rapidly, the amount of data available is continuously increasing in all the major fields of science and industry, from biomedics [Saeys et al. (2007); Zanon et al. (2013)] to semiconductor manufacturing [Susto et al. (2012)], just to cite a couple. Two factors contributing to this growth are the increasing dimensionality of the measurement space, due to the increasing number of sensors and features used to describe a particular problem, and the increasing capacity to store large volumes of data. Dealing with high dimensionality is a particularly challenging problem, when attempting to estimate models that can be used to provide a better understanding of underlying systems. A key requirement here, is that some form of variable or feature selection is undertaken as part of the modelling process. Previous work [Inza et al. (2004); Saeys et al. (2007); Dash and Liu (1997)] divided the techniques for variable selection into three main categories: *filter*, *wrapper* and *embedded* models. Filter methods generally consist of a two-steps approach, where a pre-processing phase is used to identify a smaller subset of variables which are subsequently used as inputs in a standard modeling algorithm; examples of these techniques are those based on correlation analysis. Wrapper methods instead embed the model search procedure within the feature subset search. In practice, a search procedure in the space of possible features subset is defined, and various subsets of features are generated and evaluated. Perhaps the best known example in this category is the stepwise selection strategy, where variables are added (or deleted) one at the time [Guyon and Elisseeff (2003)]. Finally, embedded methods estimate at the same time the model and select the most useful features. The last class of techniques is the more appealing since the variable selection procedure is embedded in the model estimation phase and does not require additional algorithm steps [Bastani et al. (2012); Tibshirani (1996)].

Here, we consider the analysis of root causes in classification tasks. Often, only poor quality data are available. For example, the classification problem can be ill-conditioned because the number of input variables is greater than the number of data samples. For this reason, we embed sparse classification techniques within a Monte Carlo framework with the aim of dealing with the poor data and returning a more robust analysis of the root causes.

The paper is organized as follows: Section 2 presents an overview of the techniques considered in this work, namely the naïve bayes (NB) classifier, the logistic regression model with $\ell_1$ norm regularization and the relevance vector

⋆ The financial support of the Irish Centre for Manufacturing Research and Enterprise Ireland (grants CC/2010/1001 and CC/2011/2001) are gratefully acknowledged.

machine (RVM), a fully Bayesian treatment of the logistic regression model; Section 3 then describes the Monte Carlo strategy for root cause analysis while in Section 4 the two datasets are presented. Finally, Section 5 highlights the results for the three models and in Section 6 some final remarks are provided.

## 2. SPARSE CLASSIFICATION TECHNIQUES

In this work, we consider the following notation for the data: input variables are collected in an $(n \times p)$ matrix $X$, where $p$ represents the number of variables (or features) and $n$ the number of observations. The symbol $x_i$ indicates the $(1 \times p)$ vector collecting the $p$ variables for the $i$-th observation. The $(n \times 1)$ output vector $Y$ collects the class to which each observation belongs. All the datasets described in Section 4 have two classes ($k = 2$), thus the output variable can assume only two values, i.e. $y_i \in \{0, 1\}$.

### *2.1 Naïve Bayes Classifier*

Optimal classification for a data point $(x_i, y_i)$ with the naïve bayes (NB) classifier is done by calculating the class posterior for that point $p(y_i|x_i)$, namely, the probability that an observation belongs to one of the two classes given the data [Hastie et al. (2009)]. The class posterior for the $k$-th class is calculated by applying Bayes' theorem:

$$p(y_i|x_i) = \frac{f_k(x_i, \theta_k)\pi_k}{\sum_{k=1}^{K} f_k(x_i, \theta_k)\pi_k} \tag{1}$$

where $\pi_k$ is the prior probability of each class and is defined as $\pi_k = n_k/n$ where $n_k$ is the number of observations in the $k$-th class. For continuous input variables, $f_k(x_i, \theta_k)$ can be modeled as a Gaussian distribution parametrized by $\theta_k = (\mu_k, \Sigma_k)$, where $\mu_k$ and $\Sigma_k$ are the sample mean and the covariance matrix calculated from the portion of data matrix $X$ corresponding to the two classes. $f_k(x_i, \theta_k)$ represents the class-conditional density of the data point $x_i$ in one of the two classes $k$ and is also known as the likelihood:

$$f_k(x_i, \theta_k) = p(x_i|y_i) = \\ = \frac{1}{(2\pi)^{p/2}|\Sigma_k|^{1/2}} e^{-\frac{1}{2}(x_i-\mu_k)^T\Sigma_k^{-1}(x_i-\mu_k)} \tag{2}$$

The NB model assumes that the input variables are conditionally independent in each class. This is a strong assumption, not always satisfied, but found to work well in practical applications [Hastie et al. (2009)]). The data point $(x_i, y_i)$ is assigned to the class giving the highest value of the posterior calculated in (1).

The NB model formulation does not include an implicit methodology to perform variable selection. Instead, a *step-wise* (forward or backward [Guyon and Elisseeff (2003)]) strategy can be used. For sake of space, we refer the reader to [Guyon and Elisseeff (2003); Hastie et al. (2009)]. This procedure is performed at step 7 of Algorithm 1.

### *2.2 Logistic Regression and $\ell_1$ Norm Regularization*

A linear regression model is a model where a set of input variables are linearly combined through a set of coefficients with the aim of estimating one (or more) quantitative output variable. In logistic regression, the model output is constrained to assume only a limited number of values, transforming it into a classification problem. In order to constrain the output of the logistic regression model to only two values (since we have two classes), a logistic sigmoid function is considered. This function links the probability of an outcome to the linear combination of the input variables. As an example, if the model output $\sigma(x_i\beta + \beta_0)$ assumes values lower than 0.5, than the point $x_i$ is assigned to class 1, otherwise it belongs to class 2:

$$\sigma(x_i\beta + \beta_0) = \frac{e^{(x_i\beta+\beta_0)}}{e^{(x_i\beta+\beta_0)} + 1} = \frac{1}{e^{-x_i\beta} + 1} \tag{3}$$

Hereafter we assume, for convenience, that the intercept coefficient $\beta_0$ is incorporated into the vector $\beta$. The output variable $y_i$ can be regarded as being Bernoulli distributed, since it can only assume two values. The outcome $y_i$ is thus determined by a probability $p_i$ if $y_i = 1$ or $1 - p_i$ if $y_i = 0$. In a more compact form this can be written as:

$$Pr(y_i|x_i) = \begin{cases} p_i & \text{if } y_i = 1 \\ 1 - p_i & \text{if } y_i = 0 \end{cases} \tag{4}$$

or as:

$$Pr(y_i|x_i) = p_i^{y_i}(1 - p_i)^{1-y_i} \tag{5}$$

The parameter vector $\beta$ of the logistic model is estimated in such a way that the probability in (5) is maximized for each data point. This is better known as the maximum likelihood estimate, namely, find the $\beta$ that is most likely to have generated the data. Formally, the likelihood is given by the product of (5) for each data point $i = 1, \ldots, n$:

$$l(\beta|X, Y) = \prod_{i=1}^{n} p_i^{y_i}(1 - p_i)^{1-y_i}. \tag{6}$$

Substituting (3) in (6) and taking the logarithm we obtain the log-likelihood:

$$\log l(\beta|X, Y) = \sum_{i=1}^{n} y_i \log(\sigma(x_i\beta)) + \\ + (1 - y_i)\log(1 - \sigma(x_i\beta)) \tag{7}$$

that is maximized to obtain an estimate of the parameter vector $\hat{\beta}$. As mentioned in Section 1, we are seeking a model with a sparse vector $\beta$ with few non-zero coefficients in order to improve interpretability and identify only important variables. For this purpose, a term can be added to the likelihood function that has the role of penalizing complex models. In particular we consider an $\ell_1$ term that penalizes the sum of the absolute values of the model coefficients. In the literature, this type of model regularization approach is known as the Least Absolute Shrinkage and Selection Operator (LASSO) [Tibshirani (1996)]. The trade-off between likelihood maximization and model complexity is governed by the regularization parameter $\lambda(\geq 0)$ [Friedman et al. (2010)]:

$$\log l(\beta|X, Y) = \sum_{i=1}^{n} \{y_i \log(\sigma(x_i\beta)) + \\ + (1 - y_i)\log(1 - \sigma(x_i\beta))\} - \lambda \sum_{j=1}^{p} |\beta_j| \tag{8}$$

The logistic regression solution with $\ell_1$ regularization penalty is given by the argument that maximize the cost function (8), which can be found using a coordinate descent-based algorithm [Friedman et al. (2010)], while

the parameter $\lambda$ in (8) is usually chosen through a cross-validation procedure. This step is performed at point 7 of Algorithm 1. For further details see [Hastie et al. (2009)].

### *2.3 Relevance Vector Machine*

The *relevance vector machine* (RVM) is a fully Bayesian treatment of the logistic regression model [Tipping (2001)]. The model is obtained by combining the logistic linking function (3) and the Bernoulli distribution (5) to obtain the likelihood function of eq. (6). It is well known in the literature that estimating $\beta$ to maximize the likelihood can lead to severe overfitting [Hastie et al. (2009)]. To avoid this, a "complexity" penalty term was added to the likelihood function for estimating the previous model. The $\ell_1$ norm controls complexity inducing sparseness on the parameter vector coefficients, shrinking many coefficient of the logistic model to zero. In the RVM model, a fully Bayesian perspective is considered instead. The parameters of the logistic model are constrained by defining a *prior* probability distribution over the coefficients. A popular choice is the zero-mean Gaussian prior:

$$p(\beta|\alpha) = \prod_{j=1}^{p} \mathcal{N}\left(\beta_j|0, \alpha_j^{-1}\right) \tag{9}$$

where $\alpha$ is a vector of *hyperparameters*. Note how there is an individual hyperparameter associated with each coefficient $\beta_j$. In order to complete the specification of the hierarchical prior we should define hyperpriors over the vector $\alpha$. For the purpose of this work, we do not report them here and refer to [Tipping (2001)] for further details. The above formulation of the prior distributions is also known as *automatic relevance determination* (ARD) [MacKay (1992b); Neal (1996)]. The basic idea is that only important variables are selected if there is enough evidence in the data. The priors over the model coefficients are initialized with large values (small $\alpha_j$'s), in such a way that all variables are considered important at the beginning. The evidence from the data will then concentrate the posterior probability at very large values for some of the $\alpha_j$, with the consequence that the posterior probability of the associated weights will be concentrated to zero, effectively "switching off" the corresponding variables. A practical advantage of this formulation is that the model complexity is automatically determined from the data and thus the RVM model does not need cross-validation for model order selection. Estimation of the quantities of interest in a Bayesian framework (i.e., inference) is obtained by computing the posterior over all unknowns given the data following the Bayes' rule:

$$p(\beta, \alpha|Y) = \frac{p(Y|\alpha, \beta)p(\alpha, \beta)}{p(Y)} \tag{10}$$

The posterior (10) cannot be computed directly, thus an approximation procedure based on Laplace's method can be used [MacKay (1992a); Tipping et al. (2003)].

## 3. MONTE CARLO SIMULATIONS FOR ROOT CAUSE ANALYSIS

In statistical modeling, several issues can arise mainly related to the high number of features. In particular, the situation where the number of variables is high compared to the number of available data points ($p > n$) leads to ill-conditioned problems, calling for techniques able to simultaneously handle input selection and model coefficient estimation. Then, from a classification point of view, there is also often the necessity to handle unbalanced datasets (skewed data), where the difference between the number of samples in the two classes can reach one or two orders of magnitude. In this section, we describe a Monte Carlo (MC) strategy to handle the aforementioned issues in order to perform a robust root cause analysis.

### *3.1 Model Setting*

Assessing the quality of the model on a set of observations of the phenomena that has not being used for model construction is essential for a fair evaluation of the prediction performance of the proposed model; for this reason, usually the available dataset of $n$ samples is split into two parts:

- a training dataset ($qn$ samples, where $0 < q < 1$), which is used to construct the model;
- a test dataset ($(1-q)n$ samples), which is used to assess the quality of the built model.

When only a limited number of observations are available, the quality of the model strongly depends on the split of the data. To avoid data related bias when evaluating the performance of models, repeated random sub-sampling validation [Picard and Cook (1984)], also known as Monte Carlo cross-validation (MCCV), can be used. Briefly, as described in Algorithm 1, this involves performing an analysis on $K_1$ different random splits of the available observations into training/test datasets. Thus, $K_1$ different models are built and the performance of the models are assessed as the average model performance over $K_1$ simulations. For consistent results, $K_1$ needs to be a large number (of the order of hundreds/thousands). Moreover, in a classification problem, the dataset can be unbalanced, with one class having a much lower number of instances than the other one. To handle this problem, an *undersampling* strategy can be used where at each MC iteration a new dataset is built randomly undersampling the most numerous class. The cross-validation procedure at step 7 of Algorithm 1 is required in order to estimate the number of variables entering the model for the NB model and the complexity parameter $\lambda$ for logistic regression (see Section 2). Step 7 is repeated $K_2$ times and the median among the $K_2$ complexity values is considered for estimating the model during the $i$-th MC iteration.

Classification accuracy at each MC simulation on the test dataset for the models under consideration is measured in terms of misclassification error (MCE). The MCE is calculated as the number of errors in predicting the outcome variables over the test dataset ($e$) normalized by the number of points in the test dataset ($(1-q)n$):

$$\text{MCE}_k = \frac{e}{(1-q)n} \quad k = 1, \ldots, K_1 \tag{11}$$

Accuracy of the models is also measured by the Receiver Operating Characteristic (ROC) curves, showing the True Positive Rate (fraction of true positives out of the total actual positives) *vs.* the False Positive Rate (fraction of false positives out of the total actual negatives) for various threshold values. The Area Under the Curve (AUC) can

**Algorithm 1:** Monte Carlo cross-validation with undersampling for handling unbalanced datasets.

**Data**: Input matrix $X$, class labels vector $y$
**Result**: Misclassification error (MCE), Frequency vector $F$ of ranked variables

1 Set $K_1$ and $K_2$ ;
2 Set $F = 0^p$;
3 **for** $i = 1$ ***to*** $K_1$ **do**
4  Build a balanced dataset, randomly undersampling the most numerous class;
5  Split the balanced dataset into training and test datasets;
6  **for** $j = 1$ ***to*** $K_2$ **do**
7  Cross-validation for complexity parameter estimation;
8  Choose the complexity parameter as the median among the $K_2$;
9  Identify models using the training dataset;
10  Test algorithms on test dataset;
11  Let $\mathcal{A}$ be the set of parameters selected by the model with non-null coefficients;
12  **for** $jj = 1$ ***to*** $p$ **do**
13  Set $F(jj) = F(jj) + 1$ if the $j$-th variable $x_{jj}$ is in $\mathcal{A}$;

be calculated from the ROC curves, and is considered here to summarize the results obtained with Algorithm 1, since it would be infeasible to plot ROC curves for each $K_1$ MC simulation. The AUC $\in [0, 1]$ where "1" indicates the "perfect" model, while "0.5" indicates a completely random guess.

### *3.2 Ranking*

The techniques illustrated in Section 2 return sparse models that can be used to identify important variables for the current classification problem: only those variables whose coefficients are not shrunk to zero may be considered as the ones that really affect the output, and therefore marked as "important". Variables are thus ranked according to their importance: a "1" or a "0" is assigned to a variable if its coefficient is either different from zero or not. As can be seen in lines 11-13 of Algorithm 1, this is done for each of the $K_1$ MC iterations, providing a count of the number of times a variable enters the model. In this way, a more robust computation of the root causes for the current problem can be obtained. However, when subsets of the input data are highly correlated, a more robust root cause procedure for identifying significant variables can be obtained by counting the frequency with which groups of highly correlated variables are selected, as opposed to individual variables. This strategy compensates for the so-called *grouping effect* affecting techniques based on $\ell_1$ norm regularization [Zou and Hastie (2005); Zanon et al. (2013)]: if there is a group of highly correlated variables then the algorithm tends to select only one of them from the group and does not care which one is selected. Consequently, the procedure in Algorithm 1 may not correctly identify a significant contribution from such groups because the frequency of selection will be spread across the different variables from the group that are selected in each MC model. Thus, variables presenting correlation greater than a certain threshold are first grouped into clusters. The ranking of groups is then obtained by assigning during each MC iteration a "1" to those groups presenting at least one variable in the model.

## 4. DATASETS

Table 1 shows the characteristics of the two datasets considered in this work.

Table 1. Datasets summary. The last column indicates the number of observations and the relative percentage belonging to the two classes.

| Dataset name | Observations ($n$) | Variables ($p$) | Samples distribution ($k = 1, 2$) [ratio] |
|---|---|---|---|
| *Abalone* | 4174 | 8 | 32/4132 [0.8/99.2 %] |
| *Semiconductor* | 2194 | 1988 | 410/1784 [18.6/81.4 %] |

### *4.1 Abalone Dataset*

The age of *abalone* (a sea snail) is usually determined by cutting the shell, staining it, and counting the number of rings through a microscope. Other measurements, like shell weight, diameter and others can instead be collected with the aim of estimating the age [Warwick et al. (1994)]. In this dataset the age is divided into two classes, namely older than or younger than 19 months. This dataset is particularly skewed, with 99.2% of the data corresponding to the second class ($<$ 19 months).

### *4.2 Semiconductor Dataset*

In the semiconductor manufacturing industry, virtual metrology refers to the use of models to predict "costly to measure" key physical variables from more accessible in-line measurements [Susto and Beghi (2013); Susto et al. (2013)]. In this work, we consider the problem of predicting the plasma etch rate (ER) during the production of silicon wafers. The in-line measurements are obtained from optical emission spectroscopy (OES) data consisting of 2048 wavelengths from which statistical moments (mean, variance, skewness, etc.) are calculated. We recast the prediction of ER into a classification problems with the aim of classify out-of-spec ER measurements (outside reference values) and identify input variables that help to identify these events. In order to deal with the potentially high number of features extracted from the data (each statistical moment is calculated for each wavelength), only features presenting a reasonably good correlation with ER are used as candidate variables to build the models [Guyon and Elisseeff (2003)]. This dataset presents high collinearity among subsets of variables, whose number is also substantially greater than the number of out-of-spec measurements, leading to a situation where $p > n$ when we consider the undersampling strategy for dealing with the unbalance between the two classes.

## 5. RESULTS

Each dataset presented in Section 4 underwent the MC analysis reported in Section 3. In order to have statistically

reliable results we chose to perform $K_1 = 1000$ MC simulations. Moreover, the training set dimension was set to 70% of the total data available with the remaining 30% used for testing. The results are reported in terms of: (a) Distribution of the MCE and AUC as defined in Section 3 over the $K_1$ MC simulations. Table 2 and Table 3 summarizes the results for each model on each dataset in terms of the mean and standard deviation of the MCE and AUC respectively; (b) Ranking of the variables for the purpose of root cause analysis.

Table 2. Mean and standard deviation of the misclassification error distribution obtained from the MC analysis.

| | NB-SF | NB-SB | LASSO | RVM |
|---|---|---|---|---|
| *Abalone* | .327 | .325 | .283 | .244 |
| | (.13) | (.128) | (.129) | (.119) |
| *Semiconductor* | - | - | .24 | .218 |
| | (-) | (-) | (.029) | (.028) |

Table 3. Mean and standard deviation of the AUC distribution obtained from the MC analysis.

| | NB-SF | NB-SB | LASSO | RVM |
|---|---|---|---|---|
| *Abalone* | .746 | .745 | .813 | .824 |
| | (.133) | (.133) | (.123) | (.125) |
| *Semiconductor* | - | - | .822 | .873 |
| | (-) | (-) | (.053) | (.025) |

**i) Abalone dataset:** Subsets of input variables are highly correlated, causing the NB methods to suffer the greedy nature of the stepwise strategy. This results in higher error distributions with respect to both LASSO and RVM models (see Table 2 and Table 3). Moreover, RVM outperforms the LASSO in term of classification accuracy. Table 4 shows that LASSO and RVM still agree on the selection of the top four most ranked variables (i.e. *shell weight*, *shucked weight*, *height* and *sex*), while the NB models do not consider *shucked weight*. The stepwise forward strategy returns a more parsimonious model than the backward one, where in general variables are selected less often. **ii) Semiconductor dataset:** For this dataset, the NB models cannot be computed since the balanced dataset built at each MC iteration has $p > n$ and hence the estimate of the covariance matrix $\Sigma_k$ is singular and cannot be inverted (see eq. (2)). Thus, for this dataset, only the results for the LASSO and RVM models will be presented. Table 2 highlights how RVM slightly outperforms the LASSO model in term of classification accuracy. An MCE of 22% is achieved when classifying out-of-spec ER measurements. Moreover, Table 3 shows that RVM has a significantly higher AUC calculated from the ROC curves. In order to analyze the root causes in terms of the parameters that can predict these faulty measurements, ranking of the input variables is required. Given the high level of correlation between subsets of the input variables, we do not consider the ranking of the single variables since they are present in the ranking several times without bringing additional useful information. A better strategy is to consider the ranking

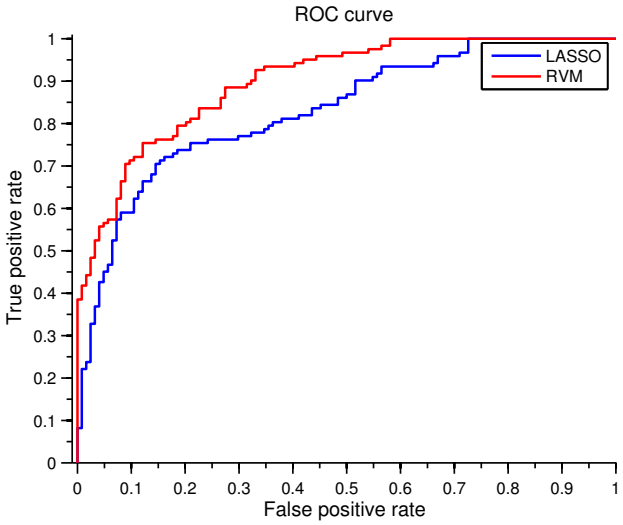


Fig. 1. ROC curves for the LASSO (blue) and RVM (red) models obtained during a MC itaration.

of groups of variables, or clusters, obtained according to the procedure explained in Section 3. Table 5 shows that there is a good agreement among the first ranked clusters for the two models. The ranked clusters with the LASSO model tend to have a lower frequency of selection compared to the RVM model. This is consequence of the fact that LASSO generates more parsimonious models than RVM, with a mean of 8 variables selected compared to 52 with RVM over the MC simulations. This suggests that the RVM model is less prone to the grouping effect than the LASSO model.

Table 5. Variables ranking for the LASSO and RVM models for the semiconductor dataset.

| LASSO | | RVM | |
|---|---|---|---|
| cluster name | rank [%] | cluster name | rank [%] |
| 1 | 100 | 1 | 99.3 |
| 11 | 86.4 | 11 | 99 |
| 31 | 68 | 2 | 96.5 |
| 5 | 57.9 | 15 | 68.5 |
| 2 | 52.3 | 20 | 65.4 |
| 6 | 49.2 | 12 | 63.1 |
| 15 | 28.3 | 33 | 48 |

## 6. CONCLUSIONS

Increasingly high dimensional datasets are being encountered in different fields of study from biomedics to manufacturing. To handle such datasets effectively techniques are required that can integrate variable selection with model building in a robust and efficient manner. In particular, there is a requirement to have *sparse* models to facilitate root cause analysis. Here, three different sparse classification techniques have been investigated and a MC approach proposed as a means of robustly identifying root cause variables when faced with challenging modelling scenarios. The scenarios in question are: (1) a large number of candidate input variables $p$; (2) strong correlation among subsets of input variables; (3) samples distributed unevenly between classes (i.e. skewed datasets). While the

Table 4. Variables ranking for the NB (Stepwise forward and backward), LASSO and RVM models for the abalone dataset.

| NB-SF | | NB-SB | | LASSO | | RVM | |
|---|---|---|---|---|---|---|---|
| var name | rank [%] | var name | rank [%] | var name | rank [%] | var name | rank [%] |
| height | 69.3 | shell weight | 70.8 | shell weight | 82 | shucked weight | 86.2 |
| shell weight | 41.6 | height | 55.3 | shucked weight | 80.1 | shell weight | 63.2 |
| sex | 32.3 | sex | 37.8 | height | 75.1 | height | 61.3 |
| length | 9.3 | viscera weight | 21.5 | sex | 57.3 | sex | 33.9 |
| whole weight | 5.5 | diameter | 13.5 | viscera weight | 19.8 | whole weight | 30.6 |
| diameter | 5 | shucked weight | 12.7 | diameter | 16.1 | viscera weight | 30.4 |
| shucked weight | 3.1 | whole weight | 10 | whole weight | 14.1 | diameter | 20.5 |
| viscera weight | 1.2 | length | 6.7 | length | 12.7 | length | 16.4 |

first two scenarios cause the problem to be ill-conditioned or singular (as was the case with the NB model applied to the semiconductor dataset), the third one is more subtle and must be handled though a suitable sampling strategy. Here, a new dataset is generated at each MC iteration with an equal number of samples in each class by randomly under-sampling the most numerous class. While this approach does exacerbate the numerical issues with model estimation (since the situation $p > n$ is more likely to happen) it ensures unbiased results. Results for two benchmark problems suggest that among the different models, the RVM is the more appealing one for several reasons. First, it shows better classification accuracy with respect to the other models. In particular, the NB models seem to suffer from the greedy nature of the stepwise framework and cannot be computed when $p > n$, as was the case with the semiconductor dataset. Moreover, this model was the worst of the 3 methods (NB, LASSO and RVM) for the ecological dataset. Classification accuracy and ranking of the LASSO model is comparable with that of RVM, but the latter is more appealing for computational reasons since the fully Bayesian treatment obviates the need for cross-validation procedures. Finally, in order to deal with the grouping effect when there is high correlation among variables, as evident in the semiconductor dataset, a cluster ranking algorithm has been proposed which focuses on analysing clusters of similar variables rather than individual variables. This ensures that important root cause patterns distributed over a group of variables are not overlooked.

## REFERENCES

Bastani, K., Kong, Z., Huang, W., Huo, X., and Zhou, Y. (2012). Fault diagnosis using an enhanced relevance vector machine (rvm) for partially diagnosable multi-station assembly processes. *Automation Science and Engineering, IEEE Transactions on*, 10(1), 124–136.

Dash, M. and Liu, H. (1997). Feature selection for classification. *Intelligent data analysis*, 1(3), 131–156.

Friedman, J., Hastie, T., and Tibshirani, R. (2010). Regularization paths for generalized linear models via coordinate descent. *Journal of Stat. Software*, 33(1), 1.

Guyon, I. and Elisseeff, A. (2003). An introduction to variable and feature selection. *The Journal of Machine Learning Research*, 3, 1157–1182.

Hastie, T., Tibshirani, R., and Friedman, J. (2009). *The Elements of Statistical Learning. Data Mining, Inference and Prediction*. Springer.

Inza, I., Larrañaga, P., Blanco, R., and Cerrolaza, A.J. (2004). Filter versus wrapper gene selection approaches in dna microarray domains. *Artificial intelligence in medicine*, 31(2), 91–103.

MacKay, D.J. (1992a). The evidence framework applied to classification networks. *Neural Comp.*, 4(5), 720–736.

MacKay, D.J. (1992b). A practical bayesian framework for backpropagation networks. *Neural Computation*, 4(3), 448–472.

Neal, R.M. (1996). *Bayesian Learning for Neural Networks*. Springer-Verlag New York, Inc.

Picard, R.R. and Cook, R.D. (1984). Cross-validation of regression models. *Journal of the American Statistical Association*, 79(387), 575–583.

Saeys, Y., Inza, I., and Larrañaga, P. (2007). A review of feature selection techniques in bioinformatics. *bioinformatics*, 23(19), 2507–2517.

Susto, G. and Beghi, A. (2013). A virtual metrology system based on least angle regression and statistical clustering. *Applied Stochastic Models in Business and Industry*, 29(4), 362–376.

Susto, G.A., Beghi, A., and De Luca, C. (2012). A predictive maintenance system for epitaxy processes based on filtering and prediction techniques. *Semiconductor Manufacturing, IEEE Transactions on*, 25(4), 638–649.

Susto, G.A., Johnston, A.B., O'Hara, P.G., and McLoone, S. (2013). Virtual metrology enabled early stage prediction for enhanced control of multi-stage fabrication processes. In *Automation Science and Engineering, 2013 IEEE International Conference on*, 201–206.

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, 267–288.

Tipping, M.E. (2001). Sparse bayesian learning and the relevance vector machine. *The Journal of Machine Learning Research*, 1, 211–244.

Tipping, M.E., Faul, A.C., et al. (2003). Fast marginal likelihood maximisation for sparse bayesian models. In *Proceedings of the ninth international workshop on artificial intelligence and statistics*, volume 1. Jan.

Warwick, J., Sellers, T., Talbot, S., Cawthorn, A., and Ford, W. (1994). The population biology of abalone in tasmania. Technical report, Tech. Rept. 48.

Zanon, M., Sparacino, G., Facchinetti, A., Talary, M.S., Caduff, A., and Cobelli, C. (2013). Regularised model identification improves accuracy of multisensor systems for noninvasive continuous glucose monitoring in diabetes management. *Journal of Applied Mathematics*.

Zou, H. and Hastie, T. (2005). Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 67(2), 301–320.