Machine Learning and Pattern Recognition Techniques for Information Extraction to Improve Production Control and Design Decisions

Carlos A. Escobar[1,2], Ruben Morales-Menendez[2]

[1] General Motors, Research and Development, Warren MI 48092, USA,
carlos.1.escobar@gm.com
[2] Tecnológico de Monterrey, Monterrey, NL. 64849, México,
rmm@itesm.mx

Abstract In today's highly competitive global market, winning requires near-perfect quality. Although most mature organizations operate their processes at very low defects per million opportunities, customers expect completely defect-free products. Therefore, the prompt detection of rare quality events has become an issue of paramount importance and an opportunity for manufacturing companies to move quality standards forward. This paper presents the learning process and pattern recognition strategy for a knowledge-based intelligent supervisory system; in which the main goal is the detection of rare quality events through binary classification. The proposed strategy is validated using data derived from an automotive manufacturing systems. The l_1-regularized logistic regression is used as the learning algorithm for the classification task and to select the features that contain the most relevant information about the quality of the process. According to experimental results, 100% of defects can be detected effectively.

Keywords l_1-regularized logistic regression · Defect detection · Intelligent supervisory system · Quality Control · Model selection criterion

1 Introduction

In today's highly competitive global market, winning requires near perfect quality, since intense competition has led organizations to low profit margins. Consequently, a warranty event could make the difference between profit and loss. Moreover, customers use internet and social media tools (e.g., Google product review) to share their experiences, leaving organizations little flexibility to recover from their mistakes. A single bad customer experience can immediately affect companies' reputations and customers' loyalty.

In the quality domain, most mature organizations have merged business excellence, lean production, standards conformity, six sigma, design for six sigma, and other quality-oriented philosophies to create a more coherent approach [1]. Therefore, the manufacturing processes of these organizations only generate a

P. Perner (Ed.): ICDM 2017, LNAI 10357, pp. 286–300, 2017.
DOI: 10.1007/978-3-319-62701-4_23

Acronyms	Definition
AI	Artificial Intelligence
CM	Confusion Matrix
CV	Cross-Validation
FN	False Negatives
FP	False Positives
FS	Feature Selection
ISCS	Intelligent Supervisory Control Systems
KB	Knowledge-Based
LASSO	Least Absolute Shrinkage and Selection Operator
LR	Logistic Regression
LP	Learning Process
ML	Machine Learning
MPCD	Maximum Probability of Correct Decision
PR	Pattern Recognition
TN	True Negatives
TP	True Positives
UMW	Ultrasonically Metal Welding

Table 1: Acronyms Table

few defects per million of opportunities. The detection of these rare quality events represents not only a research challenge, but also an opportunity to move manufacturing quality standards forward.

Impressive progress has been made in recent years, driven by exponential increases in computer power, database technologies, *Machine Learning (ML)* algorithms, optimization methods and big data [2]. From the point of view of manufacturing, the ability to efficiently capture and analyze big data has the potential to enhance traditional quality and productivity systems. The primary goal behind the generation and analysis of big data in industrial applications is to achieve fault-free (defect-free) processes [3, 4], through *Intelligent Supervisory Control Systems (ISCS)* [5].

A *Learning Process (LP)* and *Pattern Recognition (PR)* strategy for a knowledge-based *ISCS* is presented, aimed at detecting rare quality events from manufacturing systems. The defect detection is formulated as a binary classification problem, in which the l_1-regularized logistic regression is used as the learning algorithm. The outcome is a parsimonious predictive model that contains the most relevant features.

The proposed strategy is validated using data derived from an automotive manufacturing systems; (1) *Ultrasonically Metal Welding (UMW)* battery tabs from a battery assembly process. The main objective is to detect low-quality welds (*bad*).

The rest of this paper is organized as follows. It starts with a review of the theoretical background of this research in section 2. Then, section 3 describes the pattern recognition strategy, followed by an empirical study in section 4. Finally, section 5 concludes the paper.

2 Theoretical Background

The theoretical background of this research is briefly reviewed.

2.1 Machine Learning and Pattern Recognition

As discussed by [6], "As an intrinsic part of *Artificial Intelligence (AI)*, *ML* refers to the software research area that enables algorithms to improve through self-learning from data without any human intervention". *ML* algorithms learn information directly from data without assuming a predetermined equation or model. The two most basic assumptions underlying most *ML* analyses are that the examples are independent and identically distributed, according to an unknown probability distribution. On the other hand, *PR* is a scientific discipline that "deals with the automatic classification of a given object into one from a number of different categories (e.g., classes)" [7].

In *ML* and *PR* domains, generalization refers to the prediction ability of a learning algorithm model [8]. The generalization error is a function that measures how well a trained algorithm generalizes to unseen data.

In general, the *PR* problem can be widely broken down into four stages: (1) Feature space reduction, (2) Classifier design, (3) Classifier selection, and (4) Classifier assessment.

2.2 Feature Space Reduction

The world of big data is changing dramatically, and feature access has grown from tens to thousands, a trend that presents enormous challenges in the *Feature Selection (FS)* context. Empirical evidence from FS literature exhibits that discarding irrelevant or redundant features improves generalization, helps in understanding the system, eases data collection, reduces running time requirements, and reduces the effect of dimensionality [9–14]. This problem representation highlights the importance of finding an optimal feature subset. This task can be accomplished by *FS* or regularization.

Feature Selection The *FS* methods broadly fall into two classes: filters and wrappers [15].

Filter methods select variables independently of the classification algorithm or its error criteria, they assign weights to features individually and rank them based on their relevance to the class labels. A feature is considered good and thus selected if its associated weight is greater than the user-specified threshold [9]. The advantages of feature ranking algorithms are that they do not over-fit the data and are computationally faster than wrappers, and hence they can be efficiently applied to big data sets containing many features [10].

ReliefF is a well-know rank-based algorithm, the basic idea for numerical features is to estimate the quality of each according to how well their values distinguish between instances of the same and different class labels. *ReliefF* searches

for a k of its nearest neighbors from the same class, called nearest *hits*, and also a k nearest neighbors from each of the different classes, called nearest *misses*, this procedure is repeated m times, which is the number of randomly selected instances. Thus, features are weighted and ranked by the average of the distances (*Manhattan* distance) of all *hits* and all *misses* [16] to select the most important features [17], developing a significance threshold τ. Features with an estimated weight below τ are considered irrelevant and therefore eliminated. The proposed limits for τ are $0 < \tau \leq 1/\sqrt{\alpha m}$ [16]; where α is the probability of accepting an irrelevant feature as relevant.

Regularization Another approach for feature space reduction is l_1-regularization. This method trims the hypothesis space by constraining the magnitudes of the parameters [18]. Regularization adds a penalty term to the least square function to prevent over-fitting [19]. l_1-norm formulations have the advantage of generating very sparse solutions while maintaining accuracy. The classifier-fitted parameters θ_i are multiplied by a coefficient λ to shrink them toward zero. This procedure effectively reduces the feature space and protects against over-fitting with irrelevant and redundant features. The value of λ can be tuned through validation or *CV*. Regularization methods may perform better than *FS* methods [20].

2.3 Classifier Design, Selection and Assessment

A classifier is a supervised learning algorithm that analyzes the training data (e.g., data with a class label) and fits a model. In a typical *PR* analysis, the training data set is used to train a set of candidate models using different tuning parameters.

It is important to choose an appropriate validation or *CV* method to evaluate the generalization ability of each candidate model, and select the *best*, according to a relevant model selection criterion.

For information-theoretic model selection approaches in the analysis of empirical data refer to [21]. Common performance metrics for model selection based on recognition rates —correct decisions made— can be found in [22].

For a data-rich analysis, it is recommended the hold-out validation method, an approach in which a data set is randomly divided into three parts: training set, validation set, and test set. As a typical rule of thumb, 50 percent of the initial data set is allocated to training, 25 percent to validation, and 25 percent to testing [23].

Once the best candidate model has been selected, it is recommended that the model's generalization performance be tested on a new data set before the model is deployed. This can also determine whether the model satisfies the learning requirement [23]. The generalization performance can be efficiently evaluated using a *Confusion Matrix (CM)*.

Confusion Matrix In predictive analytics, a *CM* [22] is a table with two rows and two columns that reports the number of *False Positives (FP)*, *False Negatives (FN)*, *True Positives (TP)*, and *True Negatives (TN)*. This allows more detailed analysis than just the proportion of correct guesses since it is sensitive to the recognition rate by class.

A type-I error may be compared with a *FP* prediction, and it is denoted by the greek letter α. On the other hand, a type-II error may be compared with a false *FN*, and it is denoted by the greek letter β [24]. Alpha, and beta are estimated by:

$$\alpha = \frac{\mathrm{FP}}{\mathrm{FP} + \mathrm{TN}}, \tag{1}$$

$$\beta = \frac{\mathrm{FN}}{\mathrm{FN} + \mathrm{TP}}. \tag{2}$$

2.4 Logistic Regression

Logistic Regression (LR), which uses a transformation of the values of a linear combination of the features, is widely used in classification problems. It is an unconstrained convex problem with a continuous differentiable objective function that can be solved either by the Newton's method or the conjugate gradient. *LR* models the probability distribution of the class label y, given a feature vector x [25].

$$P(y = 1|x; \theta) = \sigma(\theta^T x) = \frac{1}{1 + \exp(-\theta^T x)}. \tag{3}$$

where $\theta \in \mathbb{R}^N$ are the parameters of the *LR* model and $\sigma(\cdot)$ is the sigmoid function. The sigmoid curve (logistic function), maps values in $(-\infty, \infty)$ to $[0, 1]$. The discrimination function itself is not linear anymore, but the decision boundary is still linear.

Under the Laplacian prior $p(\theta) = (\lambda/2)^N \exp(-\lambda||\theta||_1)(\lambda > 0)$, the *Maximum A Posteriori (MAP)* estimate of the parameters θ is:

$$\min_{\theta} \sum_{i=1}^{M} -\log p(y^{(i)}|\mathbf{x}^{(i)}; \theta) + \lambda||\theta||_1. \tag{4}$$

This optimization problem is referred to as l_1-regularized *LR*. This algorithm is widely applied in problems with small training sets or with high dimensional input space. However, adding the l_1-regularization makes the optimization problem computationally more expensive to solve. For solving the l_1-regularized *LR* [26], the *Least Absolute Shrinkage and Selection Operator (LASSO)* is an efficient method.

2.5 Intelligent Supervisory Control Systems

*ISCS*s are computer-based decision support systems that incorporate a variety of *AI* and non-*AI* techniques to monitor, control, and diagnose process variables to

assist operators with the tasks of monitoring, detecting, and diagnosing process anomalies, or in taking appropriate actions to control processes [27]. Developing and deploying an *ISCS* requires a lot of collaborative intellectual work from different engineering disciplines.

There are three general solution approaches for supporting the tasks of monitoring, control, and diagnosis: (1) data-driven, for which the most popular techniques are Principal Component Analysis, Fisher discriminant analysis, and Partial Least-Squares analysis; (2) analytical, an approach founded in first principles or other mathematical models; and (3) *Knowledge-Based (KB)*, founded in *AI*, specifically Expert Systems, Fuzzy Logic, *ML*, and *PR* [27, 28].

Due to the explosion of industrial big data, *KB-ISCS*s have received great attention. Since the scale of the data generated from manufacturing systems cannot be efficiently managed by traditional process monitoring and quality control methods, a *KB* scheme might be an advantageous approach.

3 Learning Process and Pattern Recognition Strategy

The proposed *LP* and *PR* strategy for a *KB-ISCS* considers the l_1-regularized *LR* as the learning algorithm. Fig. 1 displays the proposed strategy, which uses the hold-out data partition method (framed into a 4-stage approach). The input is a set of candidate features, the outcome is a parsimonious predictive model that contains the most relevant features to the quality of the product. This model is used to detect rare quality events in manufacturing systems. The candidate features can be derived from sensor signals following typical feature construction techniques [29] or from process physical knowledge.

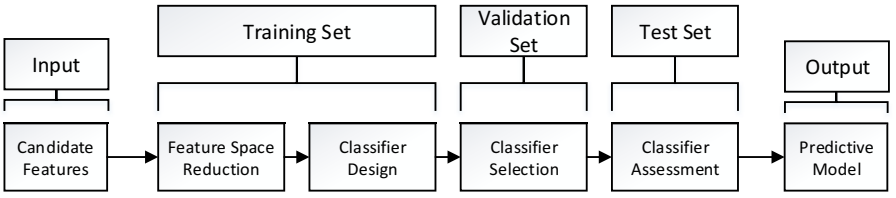

Fig. 1: Learning process and pattern recognition framework.

Two main assumptions that must be satisfied are: (1) the faulty events must be generated during the manufacturing process and captured by the signals; (2) since the *LR* learning algorithm is a linear classifier, the decision boundaries between the two classes must be linear. Due to the dynamic nature of manufacturing systems, the predictive model should be updated constantly to maintain its generalization ability.

3.1 Feature Space Reduction

The goal of this stage is to eliminate irrelevant features from the analysis. For manufacturing processes, massive amounts of data and the lack of a comprehensive physical understanding may result in the development of many irrelevant features. This problem representation highlights the importance of preprocessing the data. The *ReliefF* algorithm is used to obtain the feature ranking, and the associated weight of each feature is compared with τ to eliminate the irrelevant ones.

3.2 Classifier Design

The main goal of this stage is to design the classifier and to identify which features contain the most relevant information to the quality of the product. While the classifier is aimed to detect rare quality events, the features included in the predictive model may provide valuable engineering information. Although feature interpretation is out of the scope of this approach, analyzing the data-derived predictive model from a physics perspective may support engineers in systematically discovering hidden patterns and unknown correlations that may guide them to identify root causes and solve quality problems.

The training set is used to fit n-candidate l_1-regularized *LR* models by varying the penalty value λ. It is recommended to start with the largest value of λ that gives a non-null model (i.e., a model with the intercept only), and from that point decrease the value of λ to develop more candidate models with more features. The rationale behind this approach is that the form of the model is not known in advance; therefore, it can be approximated by generating a set of candidate models. This analysis can be computationally performed using the *LASSO* method in MATLAB or R.

Optimal Classification Threshold The goal of this step is to obtain the classification threshold of each candidate model. Since faulty events rarely occur, the data set is highly unbalanced. Therefore, the 0.5 threshold may not the optimal classification threshold, and accuracy [22] may be a misleading indicator of classification performance. To address this scenario, the *Maximum Probability of Correct Decision (MPCD)* criterion is used [30, 31]. A model selection criterion that tends to be very sensitive to *FN*s — failure to detect a quality event — in highly unbalanced data. *MPCD* is estimated by:

$$\text{MPCD} = (1 - \alpha)(1 - \beta). \tag{5}$$

Since *MPCD* is used as a model selection criterion, *gamma* (γ), the optimal classication threshold with respect to *MPCD* of each candidate model is obtained. γ is found by enumerating all candidate classification thresholds (midpoints between two consecutive examples), and estimating the *MPCD* at each threshold. γ is the maximum value of all candidate classification thresholds, a graphical representation of this procedure is shown in Fig. 4.

3.3 Classifier Selection

In the context of *PR*, the primary purpose of this stage is to select the *best* candidate model with respect to generalization (*MPCD*). The validation data set is used to estimate the *MPCD* of each candidate model, and the model with the highest value should be selected.

3.4 Classifier Assessment

The generalization performance of the selected model is evaluated on the testing set. The classifier must be assessed without the bias induced in the validation stage. This stage ensures that the model satisfies the learning target; due to the nature of the problem, *FN*s are the main concern. The target can be simplified to develop a model that produces zero or nearly zero *FN*s with the least possible number of *FP*s.

4 Empirical study

To show the effectiveness of the proposed strategy, an automotive case study is presented.

4.1 Ultrasonic Metal Welding

UMW is a solid state bonding process that uses high frequency ultrasonic vibration energy to generate oscillating shears between metal sheets clamped under pressure. It is an ideal process for bonding conductive materials such as copper, aluminum, brass, gold, and silver, and for joining dissimilar materials. Recently, ultrasonic metal welding has been adopted for battery tab joining in the manufacturing of vehicle battery packs. Creating reliable joints between battery tabs is critical because one low-quality connection may cause performance degradation or the failure of the entire battery pack. It is important to evaluate the quality of all joints prior to connecting the modules and assembling the battery pack [13].

The data used for this analysis is derived from an *UMW* of battery tabs. A very stable process, that only generates a few defective welds per million of opportunities. However, all the welds in the battery must be good for the unit to function. This problem representation not only highlights the engineering intellectual challenge but also the importance of a zero-defects policy.

The collected data set contains a binary outcome (*good*/*bad*) with 54 features derived from signals (e.g., acoustics, power, and linear variable differential transformers) following typical feature construction techniques [29]. The data set is highly unbalanced since it contains only 36 bad batteries out of 40,000 examples (0.09%). The data set is partitioned following the hold-out validation scheme (including *bads* in each data set): training set (20,000), validation set (10,000), and testing set (10,000).

Feature Space Reduction In order to eliminate irrelevant features, the data set is initially preprocessed using the *ReliefF* algorithm. *ReliefF* is run with $k = 5$ nearest neighbors and a significance threshold of $\tau = 0.031622$, (calculated based on $1/\sqrt{\alpha m}$ — $\alpha = 0.05$, and $m = 20,000$). According to the algorithm, feature 26 is the most important feature, while feature 14 is the lowest quality feature. Fig. 2 summarizes the feature ranking and which features are selected based τ. According to *ReliefF*, 43 features —out of 54— should be selected.

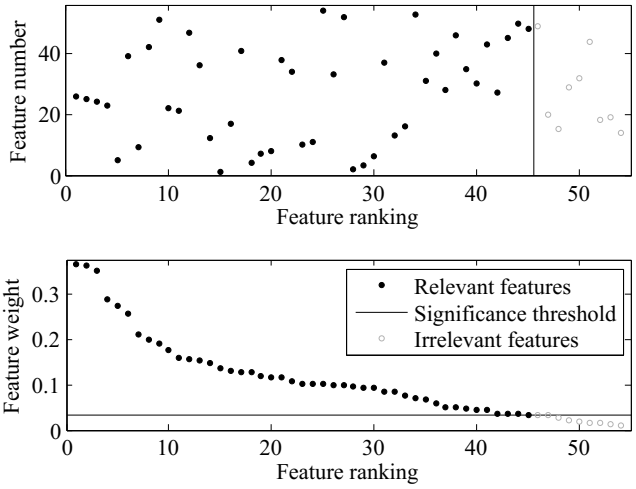

Fig. 2: Feature ranking and selection using ReliefF.

Classifier Design The training set was used to fit 100 regularized *LR* models. The *LASSO* method was applied to estimate the fitted least-squares regression coefficients for a set of 100 regularization coefficients λ, starting with the largest value of λ that gives a non-null model (i.e., a model with the intercept only). However, the non-null model is not included in the analysis since its estimated *MPCD* equals zero. Fig. 3a displays each candidate model's associated value of λ, Fig. 3b the number of features, and Fig. 3c the associated values of γ. As shown by Fig. 3a and Fig. 3b, the number of features decreases as the value of λ increases, therefore, selecting the right model is one of the main challenges.

Optimal Classification Threshold Figure 4 shows the optimal classification threshold search of candidate model 69.

Classifier Selection The goal is to select the candidate model with the highest *MPCD*. In the context of the problem that is being solved, the goal is to detect low-quality welds. Due to the relevance of failing to detect a potential defect, the

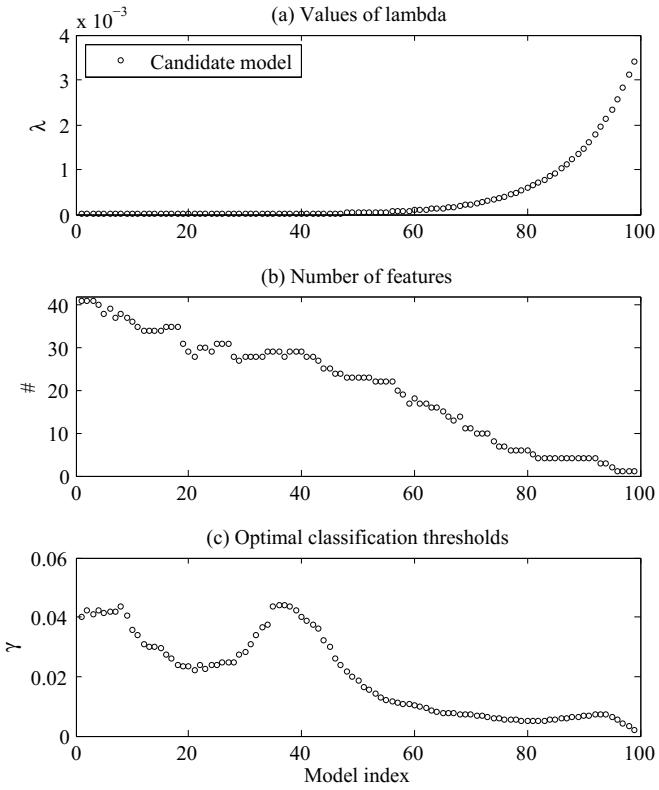

Fig. 3: Candidate model information.

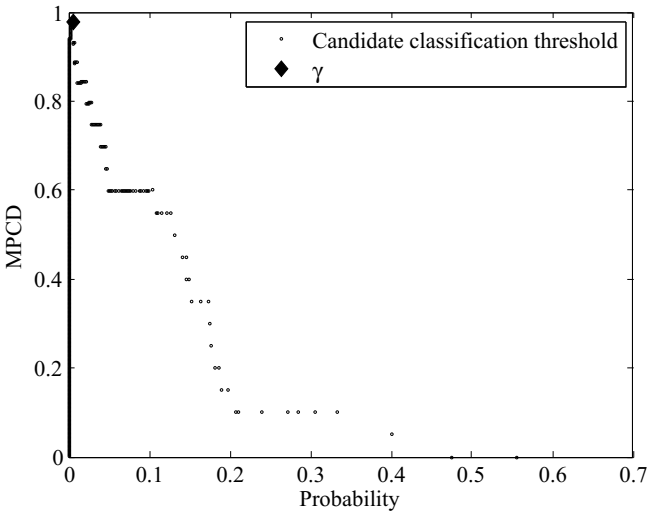

Fig. 4: Optimal classification threshold search of candidate model 69.

type-II error is the main concern of this analysis; for this reason, the *MPCD* is also used as a model selection criteria. The estimated *MPCD*, α, and β of each model are summarized in Fig. 5.

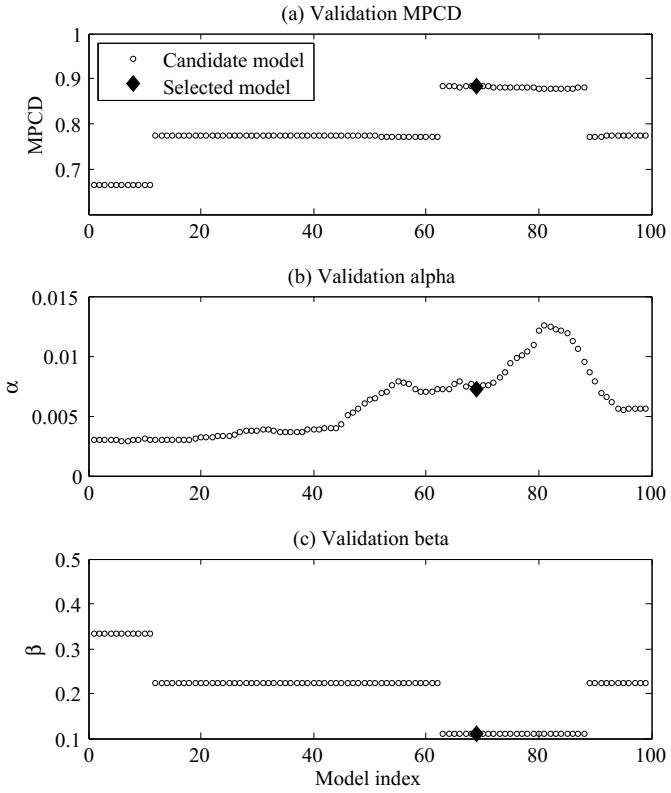

Fig. 5: Generalization performance of candidate models.

According to the selection criteria, model 69 is the best candidate, with an estimated *MPCD* of 0.8825 ($\alpha = 0.0071$, $\beta = 0.1111$) and 11-relevant features, varying the values of λ helped to identify the most relevant features. The coefficients are shown in Table 2. The value of γ for this model is 0.0073, meaning that any value estimated by the logistic function below this threshold will be classified as 0 (i.e., *good*), or 1 (i.e., *bad*) otherwise.

While is clear there are other models with less features and very similar *MPCD*, this is accomplished at the expense of *FP*. Figure 5b and Fig. 3b, show that decreasing the number of features significantly increases α.

Classifier Assessment The importance of this final step is to assess the classifier without the bias induced in the validation stage, and to ensure the model

Coef	Value	Coef	Value	Coef	Value	Coef	Value	Coef	Value	Coef	Value
θ_0	372.24	θ_4	-0.0077	θ_5	29.68	θ_9	3.20	θ_{21}	0.0019	θ_{22}	-0.0015
θ_{26}	-0.0122	θ_{30}	-0.0018	θ_{31}	-0.0035	θ_{32}	0.0536	θ_{36}	0.0108	θ_{48}	1.7122

Table 2: Coefficients of model 69.

satisfies the learning target; due to the nature of the problem, *FN*s are the main concern. Therefore, the goal can be simplified to develop a model that produces zero or nearly zero *FN*s with the least possible number of *FP*s.

The estimated *MPCD* of the final model on the testing data is 0.9956, with an estimated $\beta = 0$ and $\alpha = 0.0044$. The testing set includes approximately 10,000 records, with seven bad batteries. The classifier correctly classified the seven bad units and only misclassified 44 good units. Recognition rates are summarized on Table 3. According to model assessment results, *LR* not only shows high prediction ability, but also did not commit any type-II error.

	Declare good	Declare bad
good	9949	44
bad	0	7

Table 3: Confusion Matrix

5 Conclusions and Future Work

Today's business environment sustains mainly those companies committed to a zero-defects policy. This quality challenge was the main driver of this research, where a *LP* and *PR* strategy was developed for a *KB-ISCS*. The proposed approach was aimed at detecting rare quality events in manufacturing systems and to identify the most relevant features to the quality of the product. The defect detection was formulated as a binary classification problem and validated in a data set derived from an automotive manufacturing system. The main objective was to detect low-quality welds (*bad*) from the *UMW* of battery tabs from a battery assembly process.

The l_1-regularized *LR* was used as the learning algorithm for the classification task and to identify the most important features. Since the form of the model was not known in advance, a set of candidate models were developed —by varying the value of λ— as an effort to approximate the true model. Chosen model exhibited high capacity to detect rare quality events, since 100% of the defective units on the testing set were detected.

The l_1 penalty term helped to identify the features most relevant to the quality of the product. Although the identification of relevant features may support

engineers in systematically discovering hidden patterns and unknown correlations, it was beyond the scope of this research to use a physics-based perspective to analyze the influence of these features over the manufacturing system.

The proposed strategy used *MPCD*, a model selection criterion very sensitive to *FN*s and developed γ, an optimal classification threshold with respect to *MPCD*.

The proposed approach can be adapted and widely applied to manufacturing processes to boost the performance of traditional quality methods and potentially move quality standards forward, where soon virtually no defective product will reach the market.

Since *MPCD* is founded exclusively on recognition rates, future research along this path could focus on adding a penalty term for model complexity.

Acknowledgments Authors would like to express our deepest appreciation to Dr. Debejyo Chakraborty, Diana Wegner and Dr. Xianfeng Hu, who helped us to complete this report. A special gratitude is given to Dr. Jeffrey Abell, whose ideas and contributions illuminated this research.

Bibliography

[1] A. S. of Quality, "Emergence - 2011 Future of Quality Study," *ASQ: The Global Voice of Quality*, 2011.

[2] K. Schwab, "The Fourth Industrial Revolution: What It Means, How to Respond," *World Economic Forum*, 2016.

[3] S. Yin and O. Kaynak, "Big Data for Modern Industry: Challenges and Trends [Point of View]," *Proc of the IEEE*, vol. 103, no. 2, pp. 143–146, 2015.

[4] S. Yin, X. Li, H. Gao, and O. Kaynak, "Data-based Techniques Focused on Modern Industry: An Overview," *IEEE Trans on Industrial Electronics*, vol. 62, no. 1, pp. 657–667, 2015.

[5] V. Venkatasubramanian, R. Rengaswamy, S. Kavuri, and K. Yin, "A Review of Process Fault Detection and Diagnosis: Part III: Process History based Methods," *Computers & Chemical Eng*, vol. 27, no. 3, pp. 327–346, 2003.

[6] P. Ghosh, "A Comparative Roundup: Artificial Intelligence vs. Machine Learning vs. Deep Learning," June 2016. [Online]. Available: www.dataversity.net/ai-vs-machine-learning-vs-deep-learning

[7] S. Theodoridis and K. Koutroumbas, "Pattern Recognition and Neural Networks," in *Machine Learning and its Applications*. Springer, 2001, pp. 169–195.

[8] Z. Zhou, "Ensemble Learning," in *Encyclopedia of Biometrics*. Springer, 2009, pp. 270–273.

[9] L. Yu and H. Liu, "Feature Selection for High-Dimensional Data: A Fast Correlation-based Filter Solution," in *ICML*, vol. 3, 2003, pp. 856–863.

[10] M. Hall, "Correlation-based Feature Selection of Discrete and Numeric Class Machine Learning," in *Proc of the* 17^{th} *Int Conf on Machine Learning*. University of Waikato, 2000, pp. 359–366.

[11] K. Nicodemus and J. Malley, "Predictor Correlation Impacts Machine Learning Algorithms: Implications for Genomic Studies," *Bioinformatics*, vol. 25, no. 15, pp. 1884–1890, 2009.

[12] F. Wang, Y. Yang, X. Lv, J. Xu, and L. Li, "Feature Selection using Feature Ranking, Correlation Analysis and Chaotic Binary Particle Swarm Optimization," in 5^{th} *IEEE Int Conf on Software Engineering and Service Science*. IEEE, 2014, pp. 305–309.

[13] C. Shao, K. Paynabar, T. Kim, J. Jin, S. Hu, J. Spicer, H. Wang, and J. Abell, "Feature Selection for Manufacturing Process Monitoring using Cross-Validation," *J. of Manufacturing Systems*, vol. 10, 2013.

[14] S. Wu, Y. Hu, W. Wang, X. Feng, and W. Shu, "Application of Global Optimization Methods for Feature Selection and Machine learning," *Mathematical Problems in Eng*, 2013.

[15] A. Ng, "On Feature Selection: Learning with Exponentially Many Irrevelant Features as Training Examples," in *Proc of the* 15^{th} *Int Conf on Machine*

Learning. MIT, Dept. of Electrical Engineering and Computer Science, 1998, pp. 404–412.
[16] M. Robnik-Šikonja and I. Kononenko, "Theoretical and Empirical Analysis of ReliefF and RReliefF," *Machine Learning*, vol. 53, no. 1-2, pp. 23–69, 2003.
[17] K. Kira and L. Rendell, "The Feature Selection Problem: Traditional Methods and a New Algorithm," in *AAAI*, vol. 2, 1992, pp. 129–134.
[18] C. Bishop, *Neural Networks for Pattern Recognition*. Oxford University Press, 1995.
[19] A. Ng, "Feature Selection L1 vs L2 Regularization and Rotational Invariance," in *Proc. of the* 21^{st} *Int Conf on Machine Learning*. ACM, 2004, p. 78.
[20] E. Xing, M. Jordan, R. Karp *et al.*, "Feature Selection for High-Dimensional Genomic Microarray Data," in *ICML*, vol. 1, 2001, pp. 601–608.
[21] M. Peruggia, "Model Selection and Multimodel Inference: A Practical Information-Theoretic Approach," *J of the American Statistical Association*, vol. 98, no. 463, pp. 778–779, 2003.
[22] T. Fawcett, "An Introduction to ROC Analysis," *Pattern Recognition Letters*, vol. 27, no. 8, pp. 861–874, 2006.
[23] J. Friedman, T. Hastie, and R. Tibshirani, *The Elements of Statistical Learning*. Statistics Springer, Berlin, 2001, vol. 1.
[24] J. Devore, *Probability and Statistics for Engineering and the Sciences*. Cengage Learning, 2015.
[25] S. Lee, H. Lee, P. Abbeel, and A. Ng, "Efficient L_1 Regularized Logistic Regression," in *Proc of the National Conf on Artificial Intelligence*, vol. 21, no. 1. Cambridge, MA, 2006, p. 401.
[26] R. Tibshirani, "Regression Shrinkage and Selection via the LASSO," *J. of the Royal Statistical Society. Series B (Methodological)*, pp. 267–288, 1996.
[27] V. Uraikul, W. Chan, and P. Tontiwachwuthikul, "Artificial Intelligence for Monitoring and Supervisory Control of Process Systems," *Eng Applications of Artificial Intelligence*, vol. 20, no. 2, pp. 115–131, 2007.
[28] L. Chiang, R. Braatz, and E. Russell, *Fault Detection and Diagnosis in Industrial Systems*. Springer Science & Business Media, 2001.
[29] L. Huan and H. Motoda, "Feature Extraction, Construction and Selection: A Data Mining Perspective," 1998.
[30] J. A. Abell, J. P. Spicer, M. A. Wincek, H. Wang, and D. Chakraborty, "Binary Classification of Items of Interest in a Repeatable Process," *US Patent* , no. US8757469B2, June 2014. [Online]. Available: www.google.com/patents/US20130105556
[31] J. A. Abell, D. Chakraborty, C. A. Escobar, K. H. Im, D. M. Wegner, and M. A. Wincek, "Big data driven manufacturing — process-monitoring-for-quality philosophy," *ASME Journal of Manufacturing Science and Engineering (JMSE) on Data Science-Enhanced Manufacturing*, vol. 139, no. 10, October 2017.