Optimization of Obsolescence Forecasting Using New Hybrid Approach Based on the RF Method and the Meta-heuristic Genetic Algorithm

Yosra Grichi
École de Technologie Supérieure

Yvan Beauregard
École de Technologie Supérieure

Thien-My Dao
École de Technologie Supérieure

Obsolescence is highly complex problems due to the influence of many factors such as technological advancement. However, prediction of obsolescence appears to be one of the most efficient solutions. This paper proposes a novel approach known as GA-RF for obsolescence forecasting. Genetic algorithm (GA) searches for optimal parameters and feature selection to construct a random forest (RF) in order to improve the classification of RF. To examine the feasibility of this approach, this paper presents a comparison between GA-RF, RF, Stepwise logistic regression, and stochastic gradient boosting. Experimental results show that GA-RF outperformed the other methods with 93.3% of accuracy, 90.4% of sensitivity and 95.4% of specificity.

INTRODUCTION

Obsolescence is a major problem caused by the increasing development of technologies. Various researchers define obsolescence in various ways (Rojo, Roy, & Shehab, 2010). A component becomes obsolete when the technology used to manufacture it is no longer available, supported or produced by the supplier (Rojo, Roy, & Kelly, 2012; Shen & Willems, 2014). In other words, obsolescence can be characterized by the loss of a supplier or raw materials (Sandborn, 2013; Singh et al., 2002). The negative effects of technological obsolescence on the production performances have been studied in the literature and represent a major challenge in the long term. Technological obsolescence causes problems in the supply chain and in the management of electronic systems (Sandborn, Prabhakar, & Ahmad, 2011). Rapid technological progress is one of the factors that increase the rate of obsolescence. When one hears about technological obsolescence, the first thing that comes to mind is electronics. The electronic industry has emerged as the fastest growing sector, and has spread widely around the world. As defined by Moore's law, the rapid evolution of electronic components continues to grow, which stipulates that semiconductor density doubles approximately every 18 months (Homchalee & Sessomboon, 2013; Sandborn, 2008; Tomczykowski, 2003). This evolution creates new electronic components every year

with short lifetimes. In the USA, the industry has grown at a rapid rate since the 1990s. New technologies are introduced in the market at increasing rates. Today, the short technological lifecycle and the lack of forecasting represent a challenge for several companies that need to take into account the risk of obsolescence. However, obsolescence forecasting appears to be one of the best solutions in the management obsolescence as it assists manufacturers to identify part obsolescence. Through obsolescence forecasting, companies can ensure support for parts in service. There are two types of obsolescence forecasting: long-term forecasting (1 year or longer), which allows a proactive management and life cycle planning to support a system, and short-term forecasting, which can be observed from the supply chain. Short-term forecasting may involve reducing the number of sources, reducing inventories, and increasing the price.

To overcome the problems caused by obsolescence, many studies have been conducted to create models that can effectively forecast obsolescence. Statistical methods such as regression, Partial least square regression (PLS), logistical regression and Gaussian method have been previously employed in many works (Gao, Liu, & Wang, 2011; Sandborn et al., 2011; Solomon, Sandborn, & Pecht, 2000; Jungmok & Namhun, 2017). However, to conduct life cycle forecasting for thousands of components that affect day-to-day lives, there is a need to modify these methods. In this context, there are precise and stable forecasting approaches whose performance has been justified in the literature. In fact, results show that these approaches could be used for nonlinear predictions with high accuracy and without human manipulation (Zurada, 1992). In the past few decades, machine learning has attracted the attention of many researchers in various disciplines, and has been applied in many areas. According to Wu et al. (2007, p. 1), among the top ten algorithms identified by the Institute of Electrical and Electronics Engineers (IEEE) are AdaBoost, SVM, K-Means, decision tree, and Naïve Bayes. These algorithms have been proven to be good predictors. In addition, applying machine learning in obsolescence forecasting has received a lot of attention in the last two years (Jennings et al., 2016).

In previous work, benchmarking studies have shown that the RF performs the best among current classification techniques and numerous experiments have shown that RF has a high degree of satisfactory classifications accuracies (Grichi et al. 2017; Jennings et al., 2016). On the other hand, features selection has an important effect on the classification accuracy, because not all of them are useful for classification. Therefore selecting the best features can achieve a better performance.

To improve the accuracy of machine learning algorithms for obsolescence forecasting, this study presents a novel random forest approach based on genetic algorithm to perform features selection and parameters optimization of random forest algorithm based on supervised learning. This paper also examines the feasibility of applying this proposed method by comparing it to other algorithms which are random forest, stepwise logistic regression and stochastic gradient boosting. To the best of the researcher's knowledge, this is the first time the RF algorithm based on genetic algorithm is applied to predict obsolescence of technological components.

The rest of this paper is organized as follows. Section 2 introduces the potential obsolescence forecasting strategies while Section 3 presents the proposed framework for obsolescence forecasting. Numerical case study is presented in section 4. Finally, Section 5 presents the conclusion and recommendations for future research.

LITTERATURE REVIEW

Obsolescence Forecasting

Forecasting obsolescence is reactive in nature and is based on the resolution of the problem once noticed. The most classical reactive approaches are last time buy (LTB) and exiting stock (Rojo et al., 2010). However, there are two types of forecasting methods, namely forecasting of the obsolescence risk and forecasting of the obsolescence date (life cycle forecasting). Obsolescence risk is used to predict the probability that a component still in production (Josias, Terpenny, & McLean, 2004; Rojo et al., 2012; Van Jaarsveld & Dekker, 2011). A few researchers focus on the prediction of the risk of obsolescence. In this context, Rojo et al. (2012) conducted a Delphi study to analyze the risk of

obsolescence. They developed a risk index using some indicators, which are; years to end of life, the number of sources available, and the consumption rate versus availability of the stock. Another approach developed by (Josias et al., 2004) aims to create a risk index by measuring the manufacturers' market share, number of manufacturers, life cycle stage, and company's risk level. Finally, (Van Jaarsveld et al., 2011) used product demand data history to estimate the risk of obsolescence. Alternatively, for the life cycle forecasting, Solomon et al. (2000) was the first to introduce the life cycle forecasting method. In his paper, Solomon conducted a study to predict the obsolescence date from the life cycle curve, which included six stages: introduction, growth, maturity, saturation, decline, and obsolescence. Another method based on data mining was developed by (Sandborn, et al., 2007). The obsolescence date was obtained by applying Gaussian method. Moreover, other researchers have introduced regression analysis to predict the date of obsolescence (Gao et al., 2011). Last, (Grichi et al., 2017; Jennings et al., 2016) have used data-driven method by create machine learning algorithms to forecast the obsolescence.

Data-Driven Methods of Obsolescence Forecasting

Today, the short life cycle of technology and the lack of forecasting represent a major challenge for the electronics industry. In fact, rapid technological progress leads to the production of technological components with very short lifetimes (Ward & Sohns, 2011). However, with this advancement of technology, obsolescence should be forecasted more accurately. Data-driven methods appear to be the most efficient solutions, we can find for example neural network (Guoqiang, Patuwo, & Hu, 1998). In fact, these networks are capable of assimilating complex relationships between several variables, neural networks show good stability, better accuracy, and the ability to predict and manage a large data (Memmedli & Ozdemir, 2011). Furthermore, we find the support vector machine (SVM) algorithm. This algorithm was applied in several areas such as forecasting and optimization, SVM is considered one of the most precise and robust approaches within all the recognized algorithms. Only a few examples are required to understand the model, and it does not depend on the number of dimensions (Wu et al., 2007). The following are some of machine learning techniques that will be used in experimental results.

Random Forest

Introduced by Breiman (2001), Random forests are an integration of tree predictors where every tree depends on the values of a random vector separately. A similar distribution applies for all the trees in the forest. The tree classifier of a forest has a generalization error which relies on the strong correlation between all trees in the forest. Classification accuracy increases significantly when the group of trees is enlarged. A primary example is bagging, where to raise every tree, an arbitrary selection (without replacement) is done from the set examples. Another example is random split selection where arbitrarily, the split is selected from among the K best splits at every single node. A random forest algorithm consists of rotating many decision trees that are randomly constructed and then generating them. Bootstrap sampling (OOB: Out-Of-Bag sampling) is used in RF to have a better estimate of the distribution of the original data set (Breiman, 2001). Indeed, bootstrapping means randomly selecting a subset of the data for each tree rather than using all the data to build the trees. In statistical terms, if the trees are uncorrelated, this reduces the forecast variance. The main advantage of random forests is their resistance to variances and biases.

The random forest algorithm is used in the regression case to predict a continuous dependent variable, and in the classification case to predict a categorical dependent variable. For the regression type, a random forest consists of a set of simple prediction trees; each is capable of producing a numerical response when presented with a subset of explanatory variables or predictors. The error in this forecast is called Out Of Bag (OOB) error and is calculated as follows:

$$OOb = \frac{1}{n}\sum_{k=1}^{n}(y_k - \widehat{y_k})^2 \quad (1)$$

For the classification type, a categorical variable with N modalities is broken down into a disjunctive array (with N-1 variables) according to a 0-1 coding scheme. Thus, a categorical variable with N modalities can be considered as a set of N-1 variables, of which only one will assume the value 1 for a given observation. In fact, the ability to make predictions on a random subset of predictive variables is one of the strengths of the Random Forest module, which makes it particularly well suited to processing data sets with extremely high predictive variables. This random feature selection encourages systems diversity, and by the end, it enhances classification performance. The random forest is constructed by sampling arbitrarily the features subset as well as the training subset with regard to every system. The majority vote provides the final prediction. Finally, the random forest attains a favorable and vigorous performance with various applications (Cheng, Chan, & Qiu, 2012; Friedman, Hastie, & Tibshirani, 2017).

Genetic Algorithm

Holland (1975) was the first person to introduce the canonical genetic algorithm (CGA). Genetic algorithms are optimization algorithms based on techniques derived from genetics and natural evolution mechanisms. While GA can be used in features selection and classification, it can also solve nonlinear problems and find the best solution through some operations to obtain a set of desired design parameters. These operations are defined as follows:

Selection involves choosing the individuals of the current population that will survive and reproduce. This operation is based on the value of the fitness function that evaluates the solutions. Crossover is defined as individuals that are randomly divided into pairs. The chromosomes are then copied and recombined to form two offspring with characteristics from both parents. Mutation presents a random modification of the value of an allele in a chromosome. This operator avoids premature convergence towards local optima. It is applied with a fixed probability, P_m.

GA_s can adapt to any search space. Because of their modular nature, these evolutionary algorithms are almost completely independent of the problem to be optimized. They require a measure of the quality of the solution and a definition of space by a code and operators that allow it to browse effectively. With these advantages and the power of this algorithm, it is necessary to take into account the specificities of the problem to design or choose the code and adequate operators. It is necessary to perform several experiments to adjust such parameters of the algorithm as the size of the population, the probability of crossing and mutation, the generation number, and the replacement technique among others.

A simple GA works with a set of candidate solutions called a population, in which each chromosome (individual) represents a possible solution to the given problem. The individuals are obtained through a fitness function. The probability of having duplicates for each gene was defined by Holland (1975) as:

$$Fitness = \frac{Cost}{average\ population} \quad (2)$$

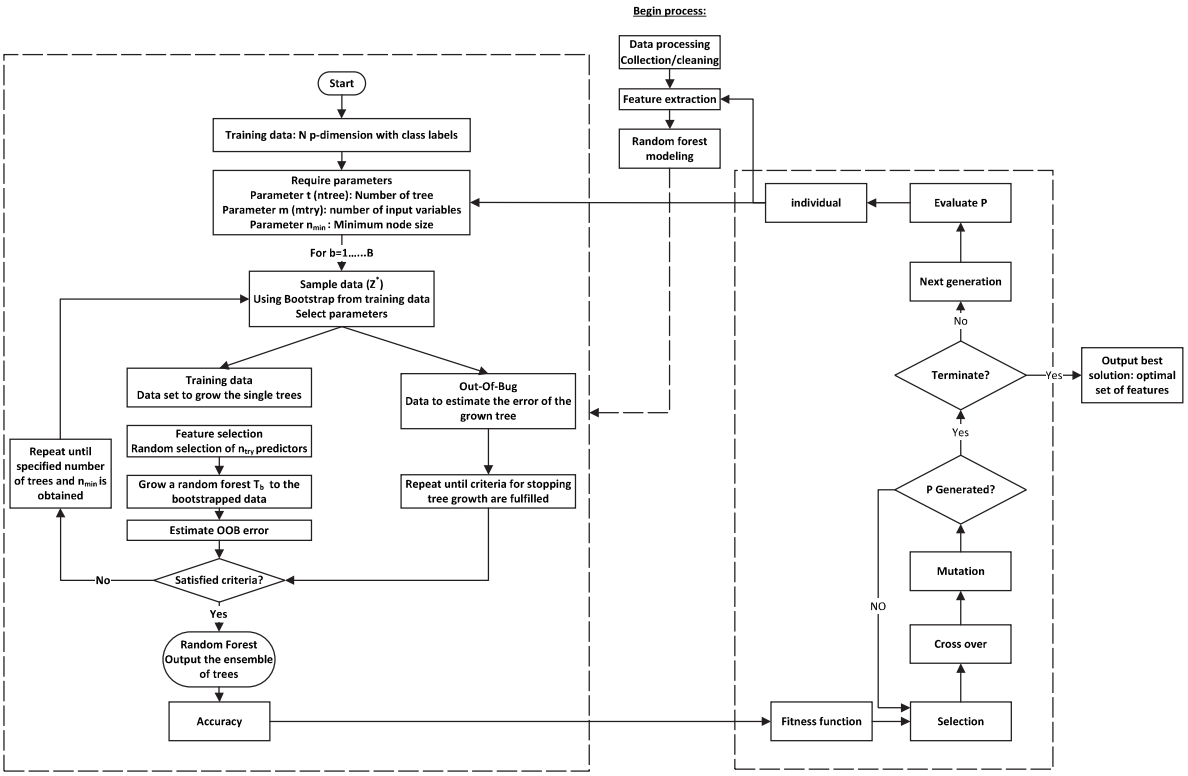
The first phase of the algorithm involves building an initial population randomly. After that, the GA selects at each step the best individuals (parents) in the current generation to create the children of the next generation. The selection of individuals is done based on the results of every individual's fitness function. After that, the next generation's chromosome is selected by choosing the most adapted elements of the new generation. The next operation involves choosing a random point and exchanging the codes (bits) of chromosome pairs and implementing mutation by replacing the element of the current chromosomes with a new generated element. This step is known as crossover and mutation. The reproduction phase of a new population is constructed from the selected individuals though crossover and mutation operators. The algorithm stops once the required number of iterations or the execution time is achieved. The current solution is taken when this test is checked; otherwise, the steps are repeated all over again.

PROPOSED FRAMEWORK

Throughout this research, the literature has showed how to evaluate the most promising approaches to obsolescence prediction based on selection criteria such as the capacity and stability of the algorithms. Based on these criteria, some of machine learning algorithms have been the subject of this research. This work presents a random forest algorithm based on genetic algorithm for obsolescence forecasting. The RF classifier is chosen because it represents a successful ensemble learning algorithms which has been proven as the best predictor for forecasting obsolescence risk based on a previous comparative study with a high degree of accuracy.

The main objective is to apply genetic algorithm to enhance the classification performance of random forest. The framework of predicting obsolescence of technological components using RF based on GA is illustrated in Figure 1.

FIGURE 1
A FLOW CHART OF THE PROPOSED FRAMEWORK



Modeling of Random Forest

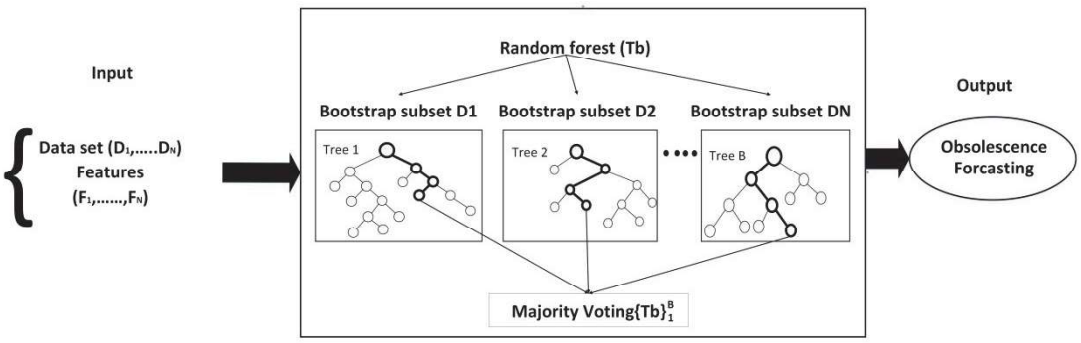
The bootstrap adopted in this research is based on Breiman's (2001) work. Each tree is constructed using a different bootstrap sample from the data sets, each having different subsets of attributes and parameters. Every final node gives a classification, and the forest chooses the classification by the majority vote (Friedman et al., 2017).

The first step of RF algorithm is the extraction of the data from an available database. Then, the data is extrapolated by choosing the attributes and specifications required by the model. After that, the data is randomly subdivided into two subsets. The first subset consists of 70% of the data and is used to form the models; this is called the learning set. The second subset consists of 30% of the data, which is used to evaluate and validate the model. Once the data is collected and the features are extracted, a sample data

that uses a bootstrap that grows from the training data is presented. Then, the learning set is introduced into the RF to create the predictive model, whose parameters are also estimated. For each terminal node of the tree, these steps are repeated until the specified number of trees is reached and the minimum node size is obtained. Next, the OOB error for the model is estimated. Finally, the output is represented as an ensemble of trees$\{T_b\}$.

To make prediction at a new point x, let $\widehat{c_b}(x)$ be the class predicting of the b[th] random forest tree, then $\widehat{C_{rf}^b} = majority\ vote\{\widehat{c_b}(x)\}$. The flow chart of random forest is shown in Figure 2.

FIGURE 2
A FLOWCHART OF RANDOM FOREST



For the RF model, three parameters are required. At each node, the variables are drawn uniformly and without replacement among all the p explanatory variables (each variable has a probability 1/p to be chosen). The number m ($m \leq p$) is fixed at the beginning of the forest construction and is identical for all trees. By default, the number of trees (n_{tree}) is 500, and m is equal to $p^{1/2}$ as suggested by (Breiman, 2001). Here, p presents the total number of variables/features while m is the number of variables in each split.

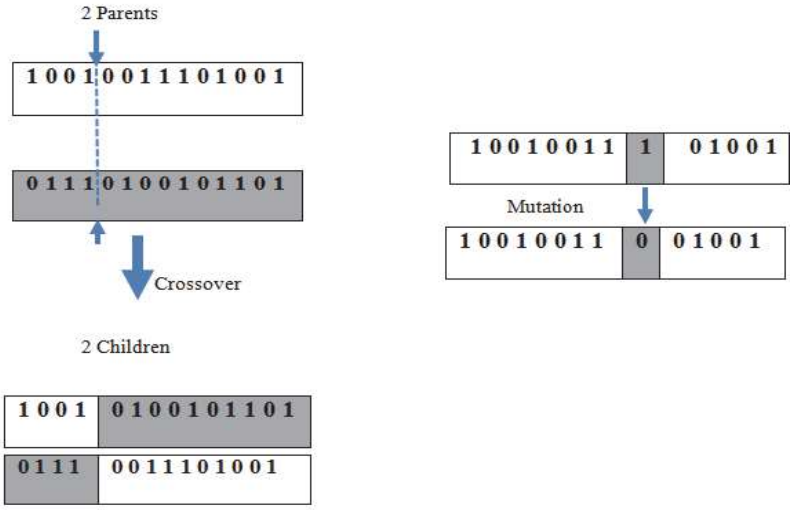
Modeling of Genetic Algorithm

The main objective of applying GA is to enhance the classification performance of random forests for obsolescence forecasting. However, evolutionary algorithms such as genetic algorithm give a promising approach to multi-criteria optimization problems (Yang & Honavar, 1998). The individual in GA is coded with bit strings. First, the population is initialized. After that, the evaluation of the fitness function is conducted where the classification accuracy of the random forest is used to evaluate the fitness of individuals.

Fitness (x) = accuracy (x) (3)

The individual with the highest fitness is selected as the results of the RF algorithm when the evolution is over. Each individual (chromosome) in the population represents a solution to the feature subset problem. Although the proposed genetic algorithm proceeds with the next generation, the operation of genetic includes selection, crossover and mutation. In addition, the crossover operation is done by exchanging the bit strings of a pair of selected individuals at random positions to generate two new offspring, while each bit of the chromosome is treated as a gene. Mutation operation involves replacing one bit with a randomly generated one in a chromosome. This operation avoids premature convergence towards local optima (See Figure3).

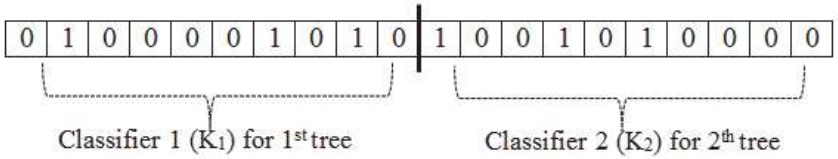
FIGURE 3
CROSSOVER AND MUTATION OPERATIONS



Let F be the total number of the ensemble features available to choose. Each feature in the GA represents a binary bit (0 or 1), where 1 represents a selected feature while 0 represents a feature that is not selected. Also, let K be the total number of classifiers. Each tree constructed contains two parameters to be optimized, which are m (the number of variables in each split) and n_{min} (the minimum node size).

Figure 4 presents a sample individual composed of two classifiers and a dataset with 10 features (e.g. for the first tree, the features 2, 7, and 9 are assigned).

FIGURE 4
AN EXAMPLE OF SELECTION FEATURES



Integration of GA-RF

As previously discussed, the method of getting the decision function (ensemble of trees) uses GA to evolve. In this phase, the decision functions can be evolved using GA following the flowchart shown in Figure 1.

The steps of GA-RF algorithm are as follows: First, the input is initialized. For the RF, the number of trees (defined by the users), the training data set, and the feature set are initialized. For the GA, the number of generations, population size, and the probability of crossover, mutation and selection, are initialized. The output presents the random forest model for obsolescence classification. The second step involves grouping the features and the data set by drawing bootstrap samples from the original data. In step 3, a tree is evolved for feature set and RF parameters combined using GA. Thereafter, a population is initialized. After population initialization, fitness evaluation for each individual is carried out. The fitness of the individual are defined as the accuracy of the classification result. For every individual, a classification label can be obtained when a data record comes. When all the training data records are collected, the accuracy of the individual can be obtained by taking into account all the classification results. After fitness evaluation, GA conducts other genetic operations and produces a new population by selecting the best individual in each generation using the elitism strategy. Elitism consists to Copy one or more of the best chromosomes in the new generation to not lose the best solution.

NUMERICAL CASE STUDY

Data Processing

A total of 999 cell phones characteristics were used in this study, obtained from (Connor et al. 2016). The data provided information such as the brand, date of introduction, and the status (production/end-of-life) of different cellphones. Also, the data contained many technical specifications such as the screen size, GPS, and keyboard. While some data used in this paper are categorical variables (yes or no), others are continuous. These predictive variables (13 features) present the RF input. The target variable (dependent variable) is the obsolescence risk.

About 70% of the data (700 cellphones) is used for training while 20% (299 cellphones) is used for testing. To obtain a better estimation of classification accuracy, a 5-fold cross-validation method was applied.

Experimental Results

The Rstudio software is used to develop the model (the code is available on request from the main author). The characteristic of GA and RF (based on R output) are as follows:

Maximum generations: 20, Population per generation: 50, Crossover probability: 0.8, Mutation probability: 0.1, Elitism: 0, Number of trees: 500.

The most frequently used metrics to describe these types of results are accuracy and error rate. The results in this manuscript are described in terms of accuracy (AC), Error rate, sensitivity (SE) and specificity (SP), and Cohen's KAPPA given by the following equations.

$$AC = \frac{TP+TN}{TP+TN+FP+FN} \tag{4}$$

$$error\ rate = 1 - AC \tag{5}$$

$$SE = \frac{TP}{TP+FN} \tag{6}$$

$$SP = \frac{TN}{TN+FP} \tag{7}$$

$$K = \frac{\Pr(a)-\Pr(e)}{1-\Pr(e)} \tag{8}$$

TP, TN, FP, and FN were defined as true positive, true negative, false positive and false negative respectively. Pr(a) presents the probability of success of classification (accuracy) and Pr(e) presents the probability of success due to chance.

The accuracy of GA-RF was initially present by a confusion matrix (See Table 1) for training and testing set.

TABLE 1
CONFUSION MATRIX OF GA-RF ALGORITHM (TRAINING AND TESTING SET)

		Predict	
	Actual	Available	Discontinued
Training	Available	**404**	6
	Obsolete	4	**286**
Testing	Available	**166**	12
	Obsolete	8	**113**

To validate the results, the GA-RF algorithm is benchmarked with the RF algorithm, Stochastic Gradient Boosting (GBM), as well as the Stepwise GLM (stepwise logistic regression) model, which selected the best logistic regression using the stepwise method and AIC as the criterion for selecting the best model. Parameters of the GA-RF and the RF algorithms were optimized on the training sample using 5-fold cross validation. The same 5 folds were used in both procedures for consistency. The comparisons of the algorithms are shown in Table 2 and 3.

TABLE 2
THE ACCURACY MEASURES COMPARISON ACROSS ALGORITHMS (TESTING SAMPLE)

Accuracy measure	RF	GA-RF	Stepwise GLM	Stochastic Gradient Boosting
Accuracy	0.913	**0.933**	0.906	0.923
No Information Rate	0.582	0.582	0.582	0.582
Kappa	0.821	**0.862**	0.807	0.841
Sensitivity	0.880	**0.904**	0.880	0.896
Specificity	0.937	**0.954**	0.925	0.943
Error rate	0.087	**0.067**	0.094	0.077
Balanced Accuracy ((Sensitivity+ Specificity)/2)	0.908	**0.929**	0.903	0.919

TABLE 3
THE ACCURACY MEASURES COMPARISON ACROSS ALGORITHMS (TRAINING SAMPLE)

Accuracy measure	RF	GA-RF	Stepwise GLM	Stochastic Gradient Boosting
Accuracy	0.947	**0.986**	0.914	0.936
No Information Rate	0.583	0.583	0.583	0.583
Kappa	0.891	**0.971**	0.824	0.867
Sensitivity	0.914	**0.980**	0.901	0.904
Specificity	0.971	**0.990**	0.924	0.958
Error rate	0.053	**0.014**	0.086	0.064
Balanced Accuracy	0.943	**0.985**	0.912	0.931

The accuracy of all algorithms was significantly higher than no information rate, which equals to 0.582 (proportion of discontinued products in the validation sample). Random Forest with feature selection based on the genetic algorithm outperformed stepwise logistic regression, the random forest algorithm and gradient boosting, the performance of which was already outstanding. All commonly used accuracy measures are higher for the GA-RF approach. The error rate was 6.7% for GA-RF, 8.7% for RF, 9.4% for GLM, while the GBM error was 7.7%.

Figure 5 presents a chart comparing the training and testing of each model. For the training set, GA-RF outperforms RF by 3.9%, GLM by 7.2% and GBM by 5%. For the testing set, GA-RF also outperforms GA, GLM and GBM by 2%, 2.7% and 1% respectively.

FIGURE 5
ACCURACY OF ALGORITHMS IN THE TRAINING AND TESTING SET

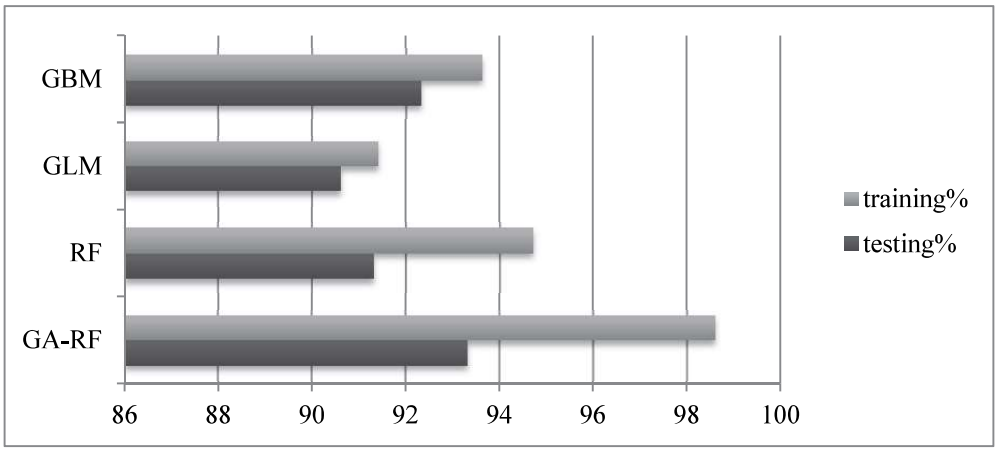
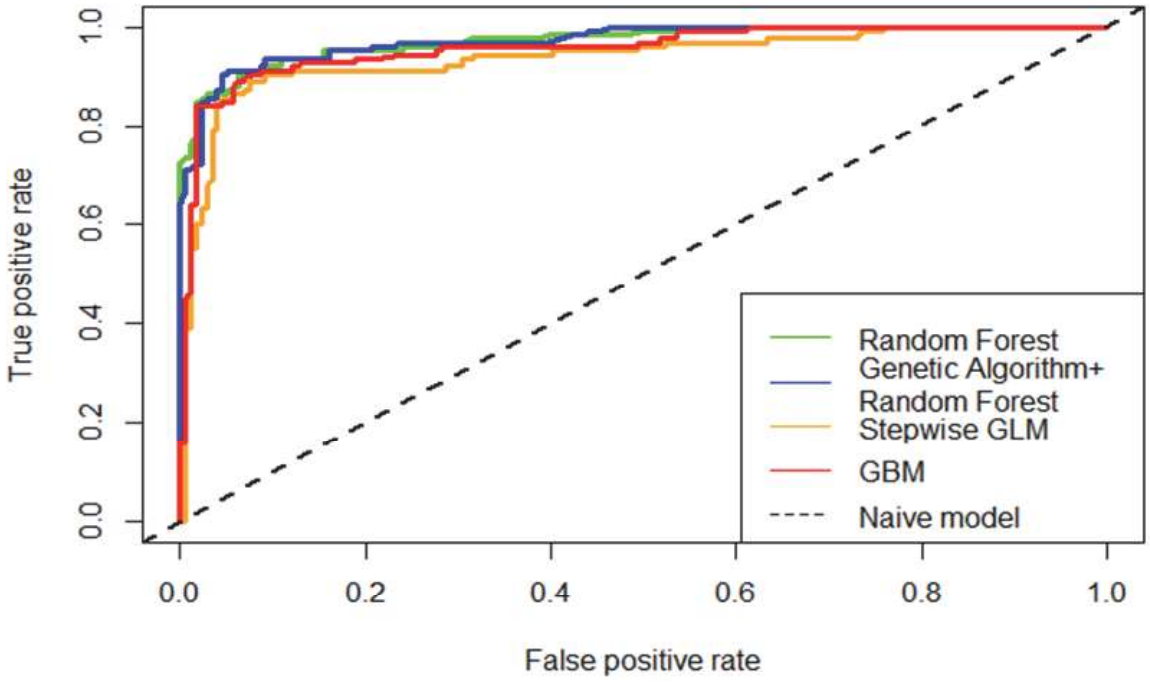


Figure 6 presents a ROC (Receiver Operator Characteristic) curve. The curve compares the false positive rate to the true positive rate. We can see from the above curve that the proposed method presented the highest accuracy of classification with a small difference compared to the others models. However, the accuracy, specificity and sensitivity in the table 1 and 2 are more impressive.

FIGURE 6
RECEIVER OPERATOR CHARACTERISTIC CURVE



CONCLUSION

Research on obsolescence is growing due to the serious challenges that complex systems face. Basically, obsolescence problems are often sudden and not planned, causing delays and extra cost. This research presents a method that can be used to estimate the risk of obsolescence using machine learning

with a high degree of accuracy. In fact, random forest method is one of a kind machine learning method, and appears as the best predictor for forecasting obsolescence risk.

This paper presents a novel approach called GA-RF algorithms, which combines a random forest and genetic algorithm for predicting obsolescence. GA is used to improve the classification of random forest by select an optimal set of features and parameters optimization for classification. The optimal set of features was adopted for the training and testing of the model to obtain the optimal outcomes in classification. In order to validate this approach, GA-RF was compared to random forest algorithm, Stepwise logistic regression and stochastic gradient boosting. GA-RF was found to be more efficient with a high degree of accuracy by 93.3% of accuracy, 90.4% of sensitivity and 95.4% of specificity after the five-fold cross-validation. Therefore, the GA-RF with feature selection subset has better performance than the others methods.

Furthermore, this model can be helpful for manufacturers and researchers that interest to apply evolutionary algorithm in predictive machine learning models. However, the exploitation of this model can be carried out on new data.

For future work, a larger data can give more accurate results. Other optimization algorithms can also be used, such as Particle Swarm Optimization (PSO), which is widely used for optimization, characterized by its robustness and efficiency. In fact, no works have been done before using this approach. Therefore, the researcher intends to include it in the future work by comparing the performance of PSO-RF and GA-RF.

REFERENCES

Breiman, L. (2001). Random forests. *Machine Learning*, *45*(1), 5–32. doi:10.1023/A:1010933404324

Cheng, Y. Y., Chan, P. P., & Qiu, Z. W. (2012). Random forest based ensemble system for short term load forecasting. *2012 International Conference on Machine Learning and Cybernetics*, 52-56. doi:10.1109/icmlc.2012.6358885

Friedman, J. H., Hastie, T., & Tibshirani, R. (2017). *The elements of statistical learning: Data mining, inference, and prediction* (2nd ed.). New York: Springer-Verlag.

Gao, C., Liu, X., & Wang, X. (2011). A model for predicting the obsolescence trend of FPGA. *The Proceedings of 2011 9th International Conference on Reliability, Maintainability and Safety*, 1354-1358. doi:10.1109/icrms.2011.5979481

Grichi, Y., Beauregard, Y., & Dao, T. M. (2017, 10-13 Dec. 2017). A random forest method for obsolescence forecasting. *2017 IEEE International Conference on Industrial Engineering and Engineering Management (IEEM),* 1602-1606.

Guoqiang, Z., Patuwo, B. E., & Hu, M. Y. (1998). Forecasting with artificial neural networks: the state of the art. *International Journal of Forecasting*, *14*(1), 35-62. doi:10.1016/S0169-2070(97)00044-7

Holland, J. H. (1975). *Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence*. Ann Arbor, MI: The University of Michigan Press.

Homchalee, R., & Sessomboon, W. (2013). A comparison of forecasting models using multiple regression and artificial neural networks for the supply and demand of Thai ethanol. *2013 IEEE International Conference on Industrial Engineering and Engineering Management*. doi:10.1109/ieem.2013.6962554

Jennings, C., Wu, D., & Terpenny, J. (2016). Forecasting obsolescence risk and product life cycle with machine learning. *IEEE Transactions on Components, Packaging and Manufacturing Technology*, *6*(9), 1428-1439. doi:10.1109/tcpmt.2016.2589206

Josias, C., Terpenny, J. P., & McLean, K. J. (2004). Component obsolescence risk assessment. In *IIE Annual Conference and Exhibition 2004,* (p. 1). Houston, TX: Component obsolescence risk assessment.

Jungmok, M., & Namhun, K. (2017). Electronic part obsolescence forecasting based on time series modeling. *International Journal of Precision Engineering and Manufacturing, 18*(5), 771-777. doi: 10.1007/s12541-017-0092-6

Memmedli, M., & Ozdemir, O. (2011). Forecasting neural network-based fuzzy time series with different neural network models. In *11th WSEAS International Conference on Signal Processing, Computational Geometry and Artificial Vision, ISCGAV'11, 11th WSEAS International Conference on Systems Theory and Scientific Computation, ISTASC'11, August 23, 2011 - August 25, 2011*, Florence, Italy.

Rojo, F. J., Roy, R., & Kelly, S. (2012). Obsolescence risk assessment process best practice. *Journal of Physics: Conference Series*, *364*, 012095. doi:10.1088/1742-6596/364/1/012095

Rojo, F. J., Roy, R., & Shehab, E. (2009). Obsolescence management for long-life contracts: state of the art and future trends. *The International Journal of Advanced Manufacturing Technology*, *49*(9-12), 1235-1250. doi:10.1007/s00170-009-2471-3

Sandborn, P. (2008). Strategic management of DMSMS in systems. *DSP Journal*, 24-30.

Sandborn, P. (2013). Design for obsolescence risk management. *Procedia CIRP*, *11*, 15-22. doi:10.1016/j.procir.2013.07.073

Sandborn, P. A., Mauro, F., & Knox, R. (2007). A data mining based approach to electronic part obsolescence forecasting. *IEEE Transactions on Components and Packaging Technologies*, *30*(3), 397-401. doi:10.1109/tcapt.2007.900058

Sandborn, P., Prabhakar, V., & Ahmad, O. (2011). Forecasting electronic part procurement lifetimes to enable the management of DMSMS obsolescence. *Microelectronics Reliability*, *51*(2), 392-399. doi:10.1016/j.microrel.2010.08.005

Shen, Y., & Willems, S. P. (2014). Modeling sourcing strategies to mitigate part obsolescence. *European Journal of Operational Research*, *236*(2), 522-533. doi:10.1016/j.ejor.2014.01.025

Singh, P., Sandborn, P., Lorenson, D., & Geiser, T. (2002). Determining optimum redesign plans for avionics based on electronic part obsolescence forecasts. *SAE Technical Paper Series*. doi:10.4271/2002-01-3012

Solomon, R, Sandborn, P., & Pecht, M. (2000). Electronic part life cycle concepts and obsolescence forecasting. *IEEE Transactions on Components and Packaging Technologies*, *23*(4), 707-717. doi:10.1109/6144.888857

Tomczykowski, W. (2003). A study on component obsolescence mitigation strategies and their impact on R&M. *Annual Reliability and Maintainability Symposium*, 332-338. doi:10.1109/rams.2003.1182011

Tu, K. J., Huang, Y. W., Lu, C. L., & Chu, K. H. (2008). Predicting the operation and maintenance costs of apartment buildings at preliminary design stage: Comparing statistical regression and artificial neural network methods. In *CME 25 Conference construction management and economics: Past, present and future: 16th-18th July 2007,* University of Reading, UK.

Van Jaarsveld, W., & Dekker, R. (2011). Estimating obsolescence risk from demand data to enhance inventory control: A case study. *International Journal of Production Economics*, *133*(1), 423-431. doi:10.1016/j.ijpe.2010.06.014

Ward, C. D., & Sohns, C. W. (2011). Electronic component obsolescence. *IEEE Instrumentation and Measurement Magazine*, *14*(6), 8-12. doi:10.1109/MIM.2011.6086892

Wu, X., Kumar, V., Ross Quinlan J., Ghosh, J., Yang, Q., Motoda, H. … Steinberg, D. (2007). Top 10 algorithms in data mining. *Knowledge and Information Systems*, *14*(1), 1-37. doi:10.1007/s10115-007-0114-2

Yang, J., & Honavar, V. (1998). Feature subset selection using a genetic algorithm. *IEEE Intelligent Systems*, *13*(2), 44-49. doi:10.1109/5254.671091

Zurada, J. M. (1992). *Introduction to artificial neural systems*. St. Paul, NY: West Publishing Company.