

# Machine-learning-based Mechanical Properties Prediction in Foundry Production

Igor Santos, Javier Nieves, Yoseba K. Peña and Pablo G. Bringas

S<sup>3</sup>Lab, Deusto Technology Foundation, Bilbao, Basque Country  
(E-mail: {isantos, jnieves, ypenya, pgb}@tecnologico.deusto.es)

**Abstract:** Ultimate tensile strength (UTS) is the force a material can resist until it breaks. The only way to examine this mechanical property is the employment of destructive inspections with the subsequent cost increment. Modelling the foundry process as an expert knowledge cloud allows properly-trained machine-learning algorithms to foresee the value of UTS. Extending previous research that presented outstanding results with a Bayesian-network-based approach, we have adapted an ANN and K-nearest-neighbour algorithm for the same objective. We compare the obtained results and show that artificial neural networks are more suitable than the rest of counterparts for the prediction of UTS.

**Keywords:** Machine learning, data mining, fault prediction.

## 1. INTRODUCTION

Foundry is an ancient magic-surrounded activity that has evolved to become a strong industry that maintains the society as we know it. In this way, foundry supplies important pieces to automotive, naval, aeronautic and weapon industries. These pieces often play a crucial role in more complex systems, as for instance, brakes, wind castings or aeroplane components. Therefore, the foundry processes are subject to very strict safety controls in order to ensure the quality of the manufactured products, since, as one may think, the tiniest error may become fatal.

Nowadays, the most used techniques for the assurance of failure-free foundry processes are exhaustive production control and diverse simulation techniques [1] but they are extremely expensive and only achieve good results in an a posteriori fashion. Therefore, providing effective *ex-ante* methods can help to increase the quality standards and to save resources in the process (i.e. saving money).

Mechanical properties are the ability of the material of a piece to take several forces and tensions. In this paper we focus on the so-called *ultimate tensile strength* that is the force that a casting can withstand until it breaks. Therefore, the foundry castings that are manufactured have to reach a certain value or threshold of ultimate tensile strength in order to pass the strict quality tests. To this extent, current standard procedures to determine that value are the employment of destructive inspections. Unfortunately, these procedures make the piece useless after the inspection and thus, they incur a cost increment.

In a previous work, we presented a mechanical properties prediction system based on a Bayesian network. After a training period, the Bayesian network learned the behaviour of the model and, thereafter it was able to foresee its outcome [2] (i.e. the value of ultimate tensile strength) showing how computer science can help enhance foundry production techniques.

Still, similar *machine-learning* classifiers have been applied in domains alike with outstanding results, for in-

stance, neural networks [3] or the K-nearest neighbour algorithm [4]. In this way, successful applications of artificial neural networks include for instance spam filtering [5], intrusion detection [6], or industrial fault diagnosis [7]. Similarly, K-nearest neighbour algorithm, despite its simplicity, has been applied for instance to visual category recognition [8], weather forecasting [9], malware detection [10] or image retrieval [11].

Against this background, this paper advances the state of the art in two main ways. First, we propose a methodology to adapt machine learning classifiers to the prediction of ultimate tensile strength and we describe the method for training them. Second, we evaluate the classifiers with a historical dataset from a real foundry process in order to compare the accuracy and suitability of each method.

The remainder of this paper is organised as follows. Section 2 discusses related work. Section 3 details mechanical properties of iron castings, focusing on the ultimate tensile strength. Section 4 introduces and describes the *machine-learning algorithms* we tailor to iron foundries. Section 5 describes the experiments performed and section 6 examines the obtained results and explains feasible enhancements. Finally, section 7 concludes and outlines the avenues of future work.

## 2. RELATED WORK

There has been a hectic activity around the applications of neural networks to several other problems of foundry process, for instance on the prediction of the ferrite number in stainless steel arc welds [12]. Similarly, successful experiments involving K-nearest neighbour algorithm include fault detection of semiconductor manufacturing processes [13].

In a verge closer to our view, neural networks have been used in several works, for instance, classifying foundry pieces [14], optimising casting parameters [15], detection of causes of casting defects [16] and in other problems [17]. Nevertheless, Bayesian networks are used as previous methods in Bayesian neural networks methodology (i.e. predicting the ferrite number in stain-

less steel [18]). In addition, K-nearest neighbour algorithm and artificial neural networks have been applied for enhance quality of steel [19] that achieves an overall root mean square error of 0.38.

The good results obtained for these works have encouraged us to tailor these approaches into our concrete problem domain.

### 3. MECHANICAL PROPERTIES OF IRON CASTINGS

There are several factors that make foundry production a very complex process, such as, the extreme conditions in which it is performed. In this way, starting from the raw material to the manufactured item, this procedure involves numerous phases, some of which may be performed in parallel. More accurately, when it refers to iron ductile castings, this process presents the following phases:

- **Melting and pouring:** The raw metals are melt, mixed and poured onto the sand shapes.
- **Moulding:** The moulding machine forms and prepares the sand moulds.
- **Cooling:** The solidification of the castings is controlled in the cooling lines until this process is finished.

Fig. 1 shows the moulding and cooling phases. Once the raw material is melt, it is poured onto the moulds (made out of sand mixed in the sand-mill) and shaped in (1). The cooling lines (2) accelerate the natural cooling process of the castings. When they are properly solidified, the sand moulds are detached from them and return to the sand-mill, so the sand can be reused to mould further castings.

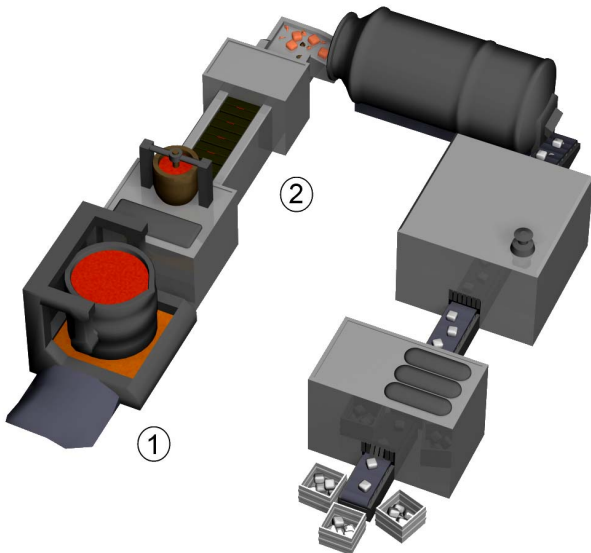


Fig. 1 Moulding and cooling in the casting production

Once these phases are accomplished, the final casting are subject to forces (loads). Engineers have to calculate the value of these forces and how the material deforms or breaks as a function of applied load, time or other conditions. Hence, it is a very important theme knowing how mechanical properties affect to iron castings [20], since

they directly affect the quality of the final piece. More accurately, the most important mechanical properties of foundry materials are the following ones [21]:

- **Strength:** it is the property that enables a metal to resist deformation under load. There are many kinds of strength such as ultimate strength and ultimate tensile strength.
- **Hardness:** it is the property to resist permanent indentation.
- **Toughness:** it is the property that enables a material to withstand shock and to be deformed without rupturing. This property is considered as a combination as strength and plasticity.
- **Resilience:** it is the property of a material to absorb energy when it is deformed elastically.
- **Elasticity:** it is the ability of a material to return to its original shape after the load is removed.
- **Plasticity:** it is the ability of a material to deform permanently without breaking or rupturing. This property is the opposite of strength.
- **Brittleness:** it is the opposite of plasticity. A brittle metal is one that breaks or shatters before it deforms. Generally, brittle metals have a high value in compressive strength but a low value in tensile strength.
- **Ductility:** it is the property that enables a material to stretch, bend or twist without cracking or breaking.
- **Malleability:** in comparison with ductility, it is the property that enables a material to deform by compressive forces without developing defects. A malleable material can be stamped, hammered, forged, pressed or rolled into thin sheets.

Furthermore, there are common or standard procedures (i.e. ASTM standards [22, 23]) for testing the value of mechanical properties of the materials in a laboratory. Unfortunately, they need to use destructive inspections and they are *ex-post* (i.e. performed after production). Moreover, the process requires suitable devices, specialised staff and quite a long time to analyse the materials.

Regarding the ultimate tensile strength, which we focus here on, its testing method is conducted as follows. First, a scientist prepares a testing specimen from the original casting (see (1) in Fig. 2). Second, the specimen is placed on the tensile testing machine (2). And, finally, this machine pulls the sample from both ends and measures the force required to break the specimen apart and how much the sample stretches before breaking.

Moreover, there are some variables that may influence the mechanical properties of the metal during the foundry process, such as the composition [24], the size of the casting, the cooling speed and thermal treatment [20, 25]. The system must take into account all of them in order to issue a prediction on those mechanical properties. In this way, the machine-learning models used in our experiments are composed of about 25 variables.

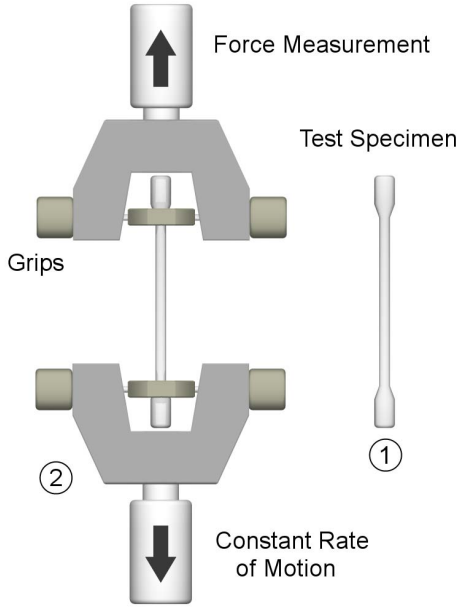


Fig. 2 Ultimate Tensile Strength Test

## 4. MACHINE-LEARNING CLASSIFIERS

### 4.1 Bayesian networks

“Bayes’ theorem” is the basis of the so-called Bayesian inference, a statistical inference method that allows, upon a number of observations, to obtain the probability that a hypothesis may be true. In this way, “Bayes’ theorem” adjusts the probabilities as new informations on evidences appear. According to its classical formulation, given two events  $A$  and  $B$ , the conditional probability  $P(A|B)$  that  $A$  occurs if  $B$  occurs can be obtained if we know the probability that  $A$  occurs,  $P(A)$ , the probability that  $B$  occurs,  $P(B)$ , and the conditional probability of  $B$  given  $A$ ,  $P(B|A)$  (shown in equation 1).

$$P(A|B) = \frac{P(B|A) \cdot P(A)}{P(B)} \quad (1)$$

Bayesian networks are probabilistic models for multivariate analysis. Formally, they are directed acyclic graphs associated to a probability distribution function [21]. Nodes in the graph represent variables (i.e. they can be either a premise or a conclusion), and the arcs represent conditional dependencies between such variables. Further, the probability function illustrates the strength of these relationships (i.e. arcs) in the graph [26].

To our needs, the most important ability of Bayesian networks is their capability of inferring the probability that a certain hypothesis becomes true (i.e. the probability of a certain value of *ultimate tensile strength*).

### 4.2 K-Nearest Neighbours

The  $K$ -nearest neighbour [4] algorithm is one of the simplest supervised machine learning algorithms for classifying instances. This algorithm is based on the class of the most nearest instances of an unknown instance (see Fig. 3).

Specifically, the training phase of this algorithm comprises representing a set of training data instances  $S = s_1, s_2, \dots, s_m$  in a  $n$ -dimensional space where  $n$  is the amount of variables for each instance (i.e. in our case the variables of the casting production).

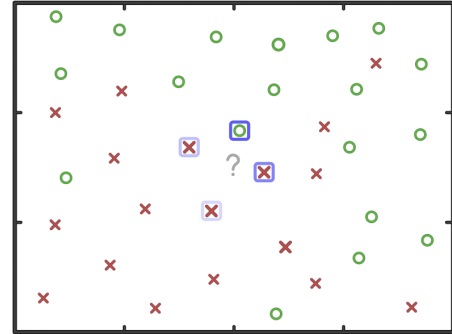


Fig. 3 Example of a knn classifier in a bidimensional space

Furthermore, the classification phase of an unknown instance (whose class is not known) is performed by measuring the distance between the training instances and the unknown instance. In this way, establishing the distance between two points  $X$  and  $Y$  in a  $n$ -dimensional space, can be achieved by using any distance measure, in our case we used Euclidean distance (shown in equation 2).

$$\sqrt{\sum_{i=0}^n (X_i - Y_i)^2} \quad (2)$$

Finally, there are several ways to choose the class of the unknown instance, the most used technique is to classify the unknown instance as the most common class amongst the  $K$ -nearest neighbors.

### 4.3 Artificial neural networks

*Artificial neural networks* (ANN) is a machine learning model that simulates the behaviour of neurons in the human brain [3]. Formally, a neuronal network consists on interconnected neurons. In this way, the activation of a neuron depends on its set of inputs, where  $y_i$  is the activation of the current neuron,  $f_i$  is the activation function,  $W_{j,i}$  is the weight of the neuron and  $a_j$  is the activation of the input neuron (shown in equation 3).

$$y_i = f_i \left( \sum_{j=0}^n W_{j,i} \cdot a_j \right) \quad (3)$$

More accurately, *Multilayer perceptron* (MLP) is a kind of artificial neural network model of simple neurons called perceptrons that are structured in layers. The layers can be classified as *input layers*, *hidden layers* and *output layer*. We perform the training of the model using *backpropagation algorithm* [27] that calculates the weights  $W_j$  of the activation function for each neuron.

## 5. EXPERIMENTS

We have collected data from a foundry specialised in safety and precision components for the automotive in-

dustry, principally in disk-brake support with a production over 45000 tons a year. These experiments are focused exclusively in the ultimate tensile strength prediction. Note that, as aforementioned, the only way to examine the mechanical properties is the employment of destructive inspections, therefore, the evaluation must be performed after the production is done.

Moreover, the acceptance/rejection criterion of the studied models resembles the one applied by the final requirements of the customer (i.e. in the examined cases, the automotive industry). According to the very restrictive quality standards imposed by these clients, pieces flawed with an invalid ultimate tensile strength must be rejected.

In these experiments, the machine-learning models have been built with the aforementioned 25 variables. We have worked with 11 different references (i.e. type of pieces) and, in order to test the accuracy of the predictions, with the results of the destructive inspection from 889 castings (note that each reference may involve several castings or pieces) performed in beforehand. Still, the machine-learning models show a different performance depending on the quality of the training. Therefore, we have examined it with datasets of diverse sizes ( $n$ ). In this way, we have carried out experiments with  $n = 100$ ,  $n = 200$ ,  $n = 300$ ,  $n = 400$ ,  $n = 500$ ,  $n = 600$ ,  $n = 700$ ,  $n = 800$ , and with the full original dataset ( $n = 889$ ). The testing procedure was always the same: the machine-learning models were trained with the 66% of the dataset (e.g. 66 castings with  $n=100$ ) and then, we infer the rest of the dataset (e.g. 34 castings with  $n=100$ ).

Specifically, we conducted the next methodology in order to evaluate properly the machine-learning classifiers:

- **Cross validation:** For each different  $n$  we have performed a  $k$ -fold cross validation [28] with  $k = 10$ . In this way, our dataset is 10 times split into 10 different sets of learning (66% of the total dataset) and testing (34% of the total data).

- **Learning the model:** For each fold, we have made the learning phase of each algorithm with each training dataset, applying different parameters or learning algorithms depending on the model. More accurately, we have use this three different models:

- *Bayesian networks:* In order to train Bayesian Networks we have used different structural learning algorithms; K2 [29], Hill Climber [30] and Tree Augmented Naive (TAN) [31]. Moreover, we have also performed experiments with a Naïve Bayes Classifier.

- *K-nearest Neighbour:* For  $knn$  we have performed experiments with  $k = 1$ ,  $k = 2$ ,  $k = 3$ ,  $k = 4$ , and  $k = 5$ .

- *Artificial neural networks:* We have used a Multilayer Perceptron (MLP) learned with *backpropagation* algorithm.

- **Testing the model:** For each fold, we evaluated the error rate between the predicted value set  $X$  and the real value set  $Y$  (both with size of the testing dataset  $m$ ) with

mean absolute error (MAE) (shown in equation 4).

$$MAE(X, Y) = \sum_{i=1}^m \frac{|X_i - Y_i|}{m} \quad (4)$$

Similarly, we have used root mean square error (RMSE) (shown in equation 5)

$$RMSE(X, Y) = \frac{1}{m} \cdot \sqrt{\sum_{i=1}^m (X_i - Y_i)^2} \quad (5)$$

## 6. RESULTS

As we mentioned before, we evaluated the classifiers in terms of prediction accuracy and error (i.e. MAE and RMSE). In this way, Fig. 4 shows the obtained results in terms of prediction accuracy. For a size of the training dataset of 100 the overall prediction of the *machine-learning classifiers* is low, however, Bayesian networks trained with *Hill Climbing* outperformed the rest of classifiers with a prediction accuracy of 81.82%. Despite these results, this approach for Bayesian networks did not perform well when increasing the size of the dataset. On the other hand, the rest of classifiers, obtained their best results for a training size between 700 and 800 instances and not for the whole training dataset ( $n = 889$ ) where there is an interesting accuracy reduction. This fact may be result of the data acquiring phase that is performed manually and, thus, it is subject to numerous errors that can include noise in the dataset. Therefore, techniques for reducing the noise of the dataset such as Principal Component Analysis [32] will be studied in further work.

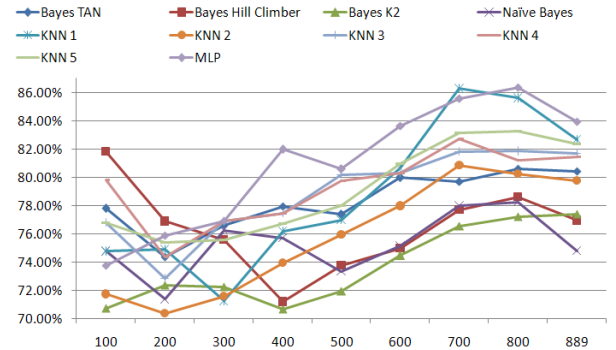


Fig. 4 Accuracy of the evaluated classifiers

More accurately, the algorithms showed quite different performances. In this way, *k-nearest neighbour* did it surprisingly well for being a lazy algorithm. Specifically, for a value of  $k = 1$ , *k-nearest neighbour* achieved the best results and the second best classifier tested. Second, Bayesian networks do achieve overall good results, specifically, Bayesian networks trained with TAN perform as expected and the results were similar to the ones obtained in [2]. Finally, *MLP* outperformed the rest of the classifiers, showing that can be an interesting classifier for predicting the values of the mechanical properties.

Furthermore, Fig. 5 shows the Mean Absolute Error and Fig. 6 shows the Root Absolute Error. In this way, the results obtained are similar to the ones of prediction accuracy and *MLP* also outperformed the rest of algorithms in terms of error.

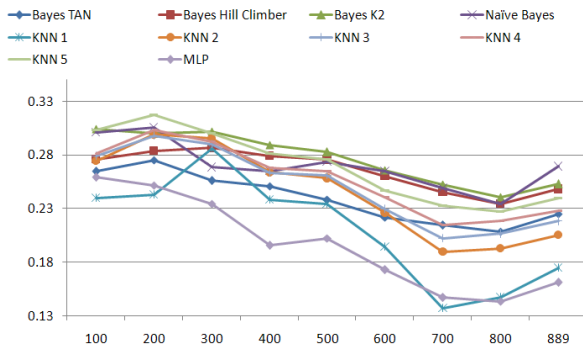


Fig. 5 Mean absolute error of the evaluated classifiers

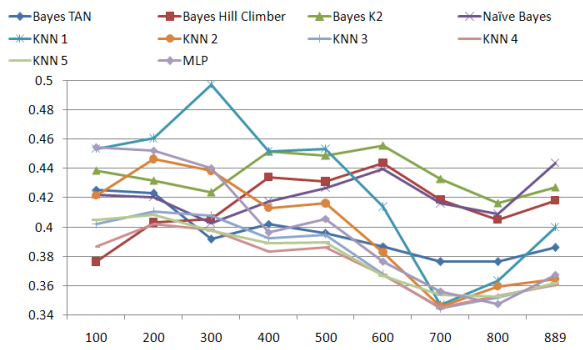


Fig. 6 Root mean square error of the evaluated classifiers

In addition, the classifiers have interesting results and can be used in a high-precision foundry. Remarkably, the outstanding results achieved by Multilayer Perceptron show that it can be used in a similar way as we have used the Bayesian networks in previous works.

Still, the sensitive module [33] (SM), that we used for mechanical properties [2], provided a decision support system for the operators in the foundry rely on Bayesian theory. Specifically, SM studies the different values that each variable adopts in order to trace the influence of such values in the apparition of a range of the ultimate tensile strength. Note that a variable may represent, for example, the use of one or another product in a certain phase of the process, applying one certain methodology or not, and so on. This is, the SM evaluates the results obtained by the Bayesian network and calculates the causal relationship between each amount of magnesium and the probability that a range of ultimate tensile strength appears

Therefore, with the combination of the sensitivity module and the best classifiers we can reduce in a significative manner the cost and the duration of the actual testing methods, apart from providing an effective *ex-ante* method.

## 7. CONCLUSION

The ultimate tensile strength is the capacity of a metal to resist deformation when subject to a certain load. When a manufactured piece does not resist a certain threshold, it must be discarded in order to avoid breaking afterwards. Foreseeing the value of ultimate tensile strength renders as one of the hardest issues in foundry

production, due to many different circumstances and variables that are involved in the casting process and determine it.

Our previous work [34, 35] pioneers the application of Artificial Intelligence to the prediction of microshrinkages. Here, we have extended that model to the prediction of mechanical properties [2]. More accurately, we focus on comparing *machine-learning* classifiers used for the prediction of ultimate tensile strength. Specifically, we have included and adapted to our particular problem domain three classifiers that have a widely use in similar issues. All of them behave well, but artificial neural networks outperformed the rest of the classifiers. Still, the ability of Bayesian theory and specifically, the sensitivity module cannot be ignored, since it is an effective method that provides a decision support system for the operators in the foundry plant.

The future development of this predictive tool is oriented in three main directions. First, we plan to extend our analysis to the prediction of other defects in order to develop a global system of incident analysis. Second, we will compare more supervised and semi-supervised machine learning algorithms in order to prove their effectiveness to predict foundry defects. Finally, we plan to integrate the best classifiers in meta-classifier combining the partial results.

## REFERENCES

- [1] J. Sertucha, A. Loizaga, and R. Su'arez, "Improvement opportunities for simulation tools," in *Proceedings of the 16th European Conference and Exhibition on Digital Simulation for Virtual Engineering*, 2006, invited talk.
- [2] J. Nieves, I. Santos, Y. K. Penya, S. Rojas, M. Salazar, and P. G. Bringas, "Mechanical properties prediction in high-precision foundry production." in *Proceedings of the 7<sup>th</sup> IEEE International Conference on Industrial Informatics (INDIN 09)*, 2009, in press.
- [3] C. M. Bishop, *Neural Networks for Pattern Recognition*. Oxford University Press, 1995.
- [4] E. Fix and J. L. Hodges, "Discriminatory analysis: Nonparametric discrimination: Small sample performance," *Technical Report Project 21-49-004, Report Number 11*, 1952.
- [5] S. Elfayoumy, Y. Yang, and S. Ahuja, "Anti-spam filtering using neural networks," in *Proceedings of the International Conference on Artificial Intelligence*, vol. 4. IC-AI, pp. 984–989.
- [6] C. Zhang, J. Jiang, and M. Kamel, "Intrusion detection using hierarchical neural networks," *Pattern Recognition Letters*, vol. 26, no. 6, pp. 779–791, 2005.
- [7] S. Simani and C. Fantuzzi, "Neural networks for fault diagnosis and identification of industrial processes," in *Proceedings of 10<sup>th</sup> Euroean Symposium on Artificial Neural Networks (ESANN)*, 2002, pp. 489–494.

- [8] H. Zhang, A. Berg, M. Maire, and J. Malik, "Svm-knn: Discriminative nearest neighbor classification for visual category recognition," in *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 2, 2006, pp. 2126–2136.
- [9] M. Sharif and D. Burn, "Improved k-nearest neighbor weather generating model," *Journal of Hydrologic Engineering*, vol. 12, p. 42, 2007.
- [10] I. Santos, Y. K. Peña, J. Devesa, and P. G. Bringas, "N-grams-based file signatures for malware detection," in *Proceedings of the 11<sup>th</sup> International Conference on Enterprise Information Systems (ICEIS), Volume AIDSS*, 2009, pp. 317–320.
- [11] Z. Lu and H. Burkhardt, "Fast Image Retrieval Based on Equal-average Equal-variance K-Nearest Neighbour Search," in *Proceedings of 18<sup>th</sup> International Conference on Pattern Recognition*, vol. 2.
- [12] J. Vitek, S. David, and C. Hinman, "Improved ferrite number prediction model that accounts for cooling rate effects part 1 model development," *Welding Journal*, vol. 82, no. 10.
- [13] H. Peter and J. Wang, "Fault detection using the k-nearest neighbor rule for semiconductor manufacturing processes," *IEEE Transactions on Semiconductor Manufacturing*, vol. 20, no. 4.
- [14] A. Lazaro, I. Serrano, J. Oria, and C. de Miguel, "Ultrasonic sensing classification of foundry pieces applying neural networks," in *5<sup>th</sup> International Workshop on Advanced Motion Control*, 1998, pp. 653–658.
- [15] P. Zhang, Z. Xu, and F. Du, "Optimizing casting parameters of ingot based on neural network and genetic algorithm," in *ICNC '08: Proceedings of the 2008 Fourth International Conference on Natural Computation*. Washington, DC, USA: IEEE Computer Society, 2008, pp. 545–548.
- [16] M. Perzyk and A. Kochanski, "Detection of causes of casting defects assisted by artificial neural networks," in *Proceedings of the Institution of Mechanical Engineers Part B Journal of Engineering Manufacture*, vol. 217, no. 9, 2003, pp. 1279–1284.
- [17] H. Bhadeshia, "Neural networks in materials science," *ISIJ international*, vol. 39, no. 10, pp. 966–979, 1999.
- [18] M. Vasudevan, M. Muruganath, and A. K. Bhaduri, "Application of bayesian neural network for modelling and prediction of ferrite number in austenitic stainless steel welds," *ser. Mathematical Modelling of Weld Phenomena - VI. London: Institute of Materials*, pp. 1079–1100, 2002.
- [19] K. Schnelle and R. Mah, "Product quality management using a real-time expert system," *ISIJ International*, vol. 34, no. 10, pp. 815–821, 1994.
- [20] R. Gonzaga-Cinco and J. Fernández-Carrasquilla, "Mechanical properties dependency on chemical composition of spheroidal graphite cast iron," *Revista de Metalurgia*, vol. 42, pp. 91–102, March–April 2006.
- [21] C. W. Lung and N. H. March, *Mechanical Properties of Metals: Atomistic and Fractal Continuum Approaches*. World Scientific Pub Co Inc, 1992.
- [22] "ASTM D1062 - Standard Test Method for Cleavage Strength of Metal-to-Metal Adhesive Bonds," 2008.
- [23] "ASTM B489 - e1 Standard Practice for Bend Test for Ductility of Electrodeposited and Autocatalytically Deposited Metal Coatings on Metals," 2008.
- [24] C. J. F. and R. Ríos, "A fracture mechanics study of nodular iron."
- [25] M. Hecht and F. Condet, "Shape of graphite and usual tensile properties of sg cast iron: Part 1," *Fonderie, Fondateur d'aujourd'hui*, vol. 212, pp. 14–28, 2002.
- [26] E. Castillo, J. M. Gutiérrez, and A. S. Hadi, *Expert Systems and Probabilistic Network Models*. Springer-Verlag, 1997.
- [27] D. Michie, D. J. Spiegelhalter, C. C. Taylor, and J. Campbell, Eds., *Machine learning, neural and statistical classification*. Upper Saddle River, NJ, USA: Ellis Horwood, 1994.
- [28] R. Kohavi, "A study of cross-validation and bootstrap for accuracy estimation and model selection," in *International Joint Conference on Artificial Intelligence*, vol. 14, 1995, pp. 1137–1145.
- [29] G. F. Cooper and E. Herskovits, "A bayesian method for constructing bayesian belief networks from databases," in *Proceedings of the 7<sup>th</sup> conference on Uncertainty in artificial intelligence*, 1991.
- [30] S. J. Russell and Norvig, *Artificial Intelligence: A Modern Approach (Second Edition)*. Prentice Hall, 2003.
- [31] D. Geiger, M. Goldszmidt, G. Provan, P. Langley, and P. Smyth, "Bayesian network classifiers," in *Machine Learning*, 1997, pp. 131–163.
- [32] C. Thomas, R. Harshman, and R. Menon, "Noise reduction in BOLD-based fMRI using component analysis," *Neuroimage*, vol. 17, no. 3, pp. 1521–1537, 2002.
- [33] C. K. Saltelli, A. and S. E. M., *Sensitivity Analysis: Gauging the Worth of Scientific Models*, wiley ed. New York, NY, USA: Springer, August 2000.
- [34] Y. Peña, P. García Bringas, and A. , Zabala, "Advanced fault prediction in high-precision foundry production," in *Proceedings of the 6<sup>th</sup> IEEE International Conference on Industrial Informatics*, 2008, pp. 1673–1677.
- [35] I. Santos, J. Nieves, Y. K. Peña, and P. G. Bringas, "Optimising machine-learning-based fault prediction in foundry production." in *Proceedings of the 2<sup>nd</sup> International Symposium on Distributed Computing and Artificial Intelligence (DCAI)*, 2009, in press.