Received October 12, 2020, accepted October 22, 2020, date of publication October 29, 2020, date of current version November 11, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3034680

A Novel Framework for Semiconductor Manufacturing Final Test Yield Classification Using Machine Learning Techniques

DAN JIANG[1,2], WEIHUA LIN[2], AND NAGARAJAN RAGHAVAN[1], (Member, IEEE)

[1]Engineering Product Development (EPD) Pillar, Singapore University of Technology and Design, Singapore 487372

[2]Silicon Laboratories International, Singapore 539775

Corresponding author: Dan Jiang (dan_jiang@mymail.sutd.edu.sg)

This work was supported by the Economic Development Board of Singapore (EDB) and Silicon Laboratories, Inc. (SLAB) through the Industry Postgraduate Program (IPP) under Grant IGIPSILICON1901.

ABSTRACT Advanced data analysis tools and techniques are important for semiconductor companies to gain competitive advantage. In particular, yield prediction tools, which fully utilize production data, help to improve operational efficiency and reduce production costs. This paper introduces a novel and scalable framework for semiconductor manufacturing Final Test (FT) yield prediction leveraging machine learning techniques. This framework is able to predict FT yield at wafer fabrication stage, so that FT low yield problems can be caught at an earlier production stage compared to past studies. Our work presents a robust solution to automatically handle both numerical and categorical production related data without prior knowledge of the low yield root cause. Gaussian Mixture Models, One Hot Encoder and Label Encoder techniques are adopted for data pre-processing. To improve model performance for both binary and multi-class classification, model selection and model ensemble using the F1-macro method is demonstrated. The framework has been applied to three mass production products with different wafer technologies and manufacturing flows. All of them achieved high F1-macro test score indicative of the robustness of our framework.

INDEX TERMS Semiconductor manufacturing, smart manufacturing, yield prediction, final test, Gaussian mixture models, clustering, ensemble methods.

I. INTRODUCTION

The semiconductor manufacturing process flow involves hundreds of processes and the production life-cycle from raw material to packaged chips can take 8-16 weeks in all. In general, Wafer Fabrication (WF), Wafer Sort (WS) and Final Test (FT) are the three major stages where huge amount of production data are generated every day, but most of them are not fully utilized. During WF stage, Wafer Acceptance Test (WAT) is conducted to monitor important WF process related parameters. Wafers that have passed WAT will then proceed to WS stage where functional defects are filtered before assembly. FT is done on packaged chips and it has the largest test coverage and longest test time to make sure defective parts are not shipped to customers. Normally, FT has more low yield problems and higher test cost compared to WAT and WS. Therefore, FT yield control is one of the most important factors which contribute to manufacturing cost and forgone production losses, especially for fabless semiconductor companies.

Current practice for FT low yield analysis is to monitor production FT yield. If there is any low yield problem, engineers need to manually review all related production data and identify the root cause. There are two major categories of root causes. The first one is front-end WF process variation. The second one is backend manufacturing flow problems, involving package types, product configurations, test facilities, human interference, etc. However, due to lack of high dimensional and unstructured data analysis capability, it is very time consuming to carry out manual root cause analysis, which results in a prolonged corrective action process.

In this paper, we propose a holistic framework for FT yield prediction using a suite of machine learning techniques. The framework is able to predict FT yield at the WF stage itself, which implies that FT low yield problems can be identified two months earlier when compared to current practice. The novelty of our framework is that it takes into consideration of all manufacturing related parameters and is able

The associate editor coordinating the review of this manuscript and approving it for publication was Zhaojun Li.

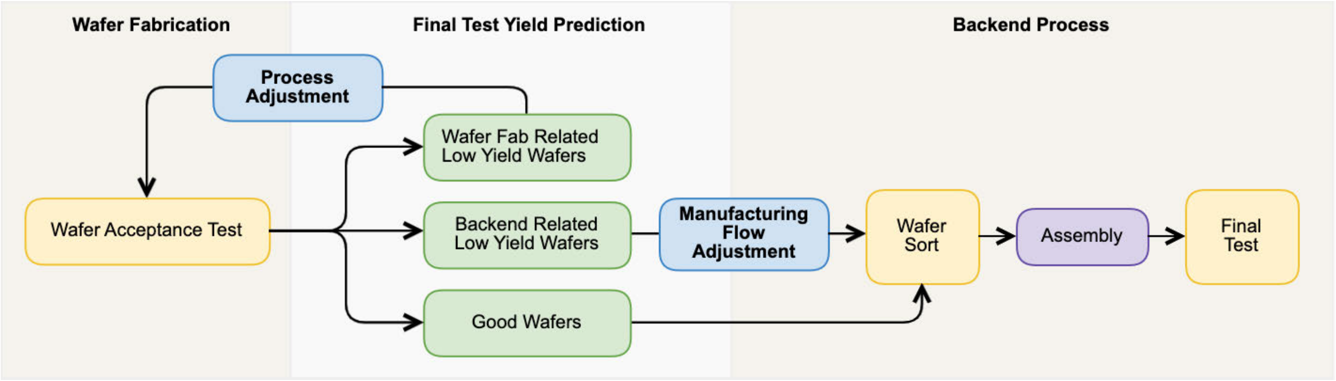

Wafer Fabrication
Final Test Yield Prediction
Backend Process
Process Adjustment
Wafer Fab Related Low Yield Wafers
Wafer Acceptance Test
Backend Related Low Yield Wafers
Manufacturing Flow Adjustment
Wafer Sort
Assembly
Final Test
Good Wafers

FIGURE 1. Semiconductor manufacturing flow and final test yield prediction at wafer fabrication stage.

to automatically handle numeric, categorical, nominal and cardinal type of manufacturing data. Based on the output from the framework, corrective actions can be taken to reduce yield loss at an earlier stage as illustrated in Fig 1. Using the WAT measurements and backend manufacturing flow parameters, our proposed model is able to classify wafer material into different yield sub-populations. Based on the binning or multi-modal classification of the yield, wafer process adjustment or backend related manufacturing flow adjustment can be selectively carried out. For example, low to moderate yield sub-population wafers can be intentionally used for fabrication of low-end non-critical application products (such as home-based security and IoT solutions) and sold to customers in such markets. Moreover, the testing priority can be adjusted for each yield sub-population and resource allocation and shipment forecast made easier based on the prediction results. Our data-driven decision-making process is able to overcome the limitations of manual work for low yield materials' data review and disposition.

II. OVERVIEW OF RELATED WORK

Recent semiconductor yield prediction studies are focusing mainly on WF and WS stages. The prediction targets are wafer map design optimization [1], [2], wafer map defect pattern monitoring [3]–[5] as well as wafer yield prediction [6]. Jang *et al.* [1] introduced a die level yield prediction model with wafer die spatial features as input parameters. The model was used to evaluate the productivity of wafer maps considering yield variations based on the die positions and die sizes of a wafer map. Studies by Kim *et al.* [2] also used wafer die spatial features as input parameters, and the model was focused on evaluation of lithography process related yield problem. Both these studies used the Deep neural network (DNN) algorithm. Convolutional Neural Network (CNN) was also applied by Nakata *et al.* [3] for wafer map failure pattern monitoring. The authors proposed a framework to classify failure patterns taking the wafer map as input. Their results showed that CNN outperforms Support Vector Machine (SVM). A novel DNN model was proposed by J. Wang *et al.* in Ref. [4] to resolve wafer map imbalance data problem. DNN is widely used in wafer map related studies because it is suitable for pattern recognition application and wafer maps can be treated as images [5]. However, DNN is more suitable for large datasets and may not be suitable for yield prediction due to over-fitting problem and poor model visibility as mentioned in [6]. Kong and Ni [6] used SVM and partial least square algorithm to predict functional block based die yield with inline metrology data as input parameters. Their yield model was based on the assumption that wafer yield loss is dominated by inline defects which may not be suitable for other situations in a real production environment. Kim *et al.* in Ref. [7] discussed equipment-related variable selection for WS yield prediction. Partial least squares, least absolute shrinkage and selection operator regression were utilized as prediction models. The limitation of their work is that the proposed model only allowed for prediction of two wafer process parameters' performance instead of the overall yield. Besides, the model does not show the relationship between the two measurements. In practise, if the two measurements are not independent, adjustment of one of the equipment variables will affect the other measurement as well. Therefore, the overall yield may not necessarily increase when the correlation between the measurements is ignored.

There are few studies on FT yield prediction. WS measurements and wafer spatial features were used as input by S. Kang *et al.* in Ref. [8] to predict two types of die level FT yield. To the best of our knowledge, there is no prior study on FT yield prediction using WAT data yet. Besides, most of the past studies mentioned above tend to predict a specific failure mode, which does not cover all failure types. The input data were closely related to low yield problems based on engineers' past experience or prior knowledge on the root cause. Their input data including wafer die features, WS measurements, inline metrology data or process equipment information, only represent certain manufacturing stages. For example, wafer map design is one of the most important factors for wafer low yield problems. However, wafer low yield problem root causes are not limited to wafer map design. It can be related to product design constraints, human error, equipment or subcontractors' performance deviations etc. In our proposed framework, all production related parameters including both numerical and categorical parameters are considered as input parameters to account for all manufacturing stages' factors into the yield model. Moreover, no manual data labelling or filtering is required, the data can be automatically fed into our FT yield prediction model framework.

III. METHODOLOGIES ADOPTED IN OUR FRAMEWORK

A. OUTPUT DISCRETIZATION

Production FT yield is a continuous random variable ranging from $0\% - 100\%$. The yield distribution can sometimes be multi-modal, highly skewed or long tailed. The distribution variation is caused by different wafer fab technology and product-specific manufacturing flow. For individual products, majority materials' FT yield are acceptable to ship to customer when all the processes are well controlled. Only a small portion of low yield material need further analysis and reprocessing. Normally, the low yield threshold is decided based on engineers' past experience and manual review of historical production data. In order to define material as high yield or low yield, output discretization is required to convert the numeric yield into categorical classes. Past semiconductor yield problems used quartile discretization method, which aimed to identify excursions for the points outside the region of $(Q1-k \times IQR, Q3 + k \times IQR)$, where k is a constant and interquartile range (IQR) is the difference between the third and the first quartiles ($Q3$ and $Q1$) of the yield [9]. However, this method is not suitable for multi-modal yield distributions. The equal width and equal frequency binning approaches are also commonly used discretization methods. However, they require users to decide on the number of intervals, k, and then discretize the continuous attributes into k intervals simultaneously. However, the hard coded k may not be suitable for all products. Other discretization algorithms focus on either minimizing the number of identified intervals or maximizing the classification accuracy.

The purpose of our framework is not only to optimize prediction accuracy, but also to correctly identify different yield classes and provide guided material disposition and root cause analysis. Material subject to similar manufacturing flows tend to have similar yield distributions. For products with the same manufacturing flow, but different FT subcontractors, the major low yield root cause is that one of the subcontractor's test performance is worse than the others. The FT yield distribution then becomes bi-modal due to the particular subcontractor. For output discretization, we'll need to differentiate the target beforehand due to subcontractor variations. Therefore, in our framework, we propose using Gaussian Mixture Models (GMM) to automatically cluster and identify optimal number of FT yield classes.

B. GAUSSIAN MIXTURE MODELS (GMM)

GMM is a probabilistic model representing a mixture of Gaussian distributions. It is a popular statistical technique and widely used for clustering problem, heterogeneous populations and multivariate density estimations. Let M denote the number of Gaussian components, which represent FT yield classes in our proposed framework. Assuming we have a training dataset with N number of FT yield values $\{X_1, \ldots, X_N\}$. Let Z be the latent parameter where

$$p(Z = m) = \pi_m, \quad m = 1, \ldots, M, \tag{1}$$

π_m are the mixture weights for the M components and therefore

$$\sum_{m=1}^{M} \pi_m = 1 \tag{2}$$

The joint probability of X with a latent variable Z is

$$P(X, Z) = \sum_{m=1}^{M} \pi_m N(X_i|\mu_m, \Sigma_m) \tag{3}$$

where π_m, μ_m, Σ_m are unknown parameters representing the m^{th} Gaussian component's mixture weight, mean and covariance. Therefore, the log-likelihood is

$$L(\pi, \mu, \Sigma) = \sum_{i=1}^{N} \log \sum_{m=1}^{M} \pi_m N(X_i|\mu_m, \Sigma_m) \tag{4}$$

One of the popular algorithm to estimate maximum likelihood of GMM's unknown parameters is Expectation-Maximization (EM) [10]. The algorithm is used to optimize the log likelihood function $L(\pi, \mu, \Sigma)$ using an iterative approach by repeating the following two steps until there is no more update required for the parameters or the update meets predefined threshold.

1) EXPECTATION-STEP

The first step is to compute posterior distribution of latent variable Z:

$$\begin{aligned} P(Z_i = m|X_i) &= \frac{P(X_i|Z_i = m)P(Z_i = m)}{P(X_i)} \\ &= \frac{\pi_m N(\mu_m, \Sigma_m)}{\sum_{m=1}^{M} \pi_m N(\mu_m, \Sigma_m)} = \gamma_{Z_i}(m) \end{aligned} \tag{5}$$

2) MAXIMIZATION-STEP

Once we compute the value of $\gamma_{Z_i}(m)$, parameters π_m, μ_m, Σ_m can be updated using below equations

$$\pi_m = \frac{\sum_{i=1}^{N} \gamma_{Z_i}(m)}{N} \tag{6}$$

$$\mu_m = \frac{\sum_{i=1}^{N} \gamma_{Z_i}(m) X_i}{\sum_{i=1}^{N} \gamma_{Z_i}(m)} \tag{7}$$

$$\Sigma_m = \frac{\sum_{i=1}^{N} \gamma_{Z_i}(m)(X_i - \mu_m)^2}{\sum_{i=1}^{N} \gamma_{Z_i}(m)} \tag{8}$$

C. ONE HOT ENCODER AND LABEL ENCODER

In our FT yield prediction framework, part of the input data are descriptive text data which need to be converted to a numerical representation in order to fit into the machine learning models. One Hot Encoding and Label Encoding are two of the most popular techniques for categorical parameter pre-processing. Both of them have benefits and drawbacks. It depends on the characteristics of the dataset to decide which one should be used for categorical data pre-processing.

The benefit of the Label Encoder is that it does not increase the dimension of the input data. Label Encoder directly uses integers to represent different text under the same categorical parameter. For example, the product A to be discussed in later section has one categorical parameter named Tester Type, which includes 6 different types. After applying Label Encoder, integers from $0-5$ are used to represent different Tester Types. But one of the drawbacks for using Label Encoder is that it introduces meaningless numerical comparison between the different types. Practically there should be no weight or ordering difference for each tester and they should be treated equally during model training. The ordering problem can bring misinterpretation, thereby affecting model performance.

The One Hot Encoder is able to avoid this problem. It creates new parameters to represent different categorical values, and assigns 0 or 1 to indicate whether or not each data point belongs to the particular category. All the processed columns have an independent relationship. For example, after applying the One Hot Encoder for the Tester Type column, the parameter columns vector size increases from 1 to 6, where each column denotes one type of Tester. For a particular data point, if it uses TESTER_03, then the third element of the vector TESTER_03 is 1 and the entries in the other five columns are all zero (0). However, the disadvantage of One Hot Encoder is that it leads to a huge increase in the input dimension space. For input parameters with high cardinality, this will result in sparse and high dimensional data thereby resulting in poor model performance.

D. F1 MACRO MEASUREMENT METRIC

FT yield prediction is either a binary or a multi-class problem with an imbalanced dataset. Normally, production yield distributions are skewed towards the high yield end, while low yield wafers tend to be the minority. The common model accuracy metric is not suitable in such a case because the majority class will tend to dominate the accuracy result. However, the low yield wafers' analysis and disposition are more important from a cost perspective.

Precision and recall are two effective metrics used for evaluating imbalanced dataset's model performance. Let TP, FP, FN denotes True Positives, False Positives and False Negatives. The precision for any individual class m is defined as

$$P_m = \frac{TP_m}{TP_m + FP_m} \tag{9}$$

It measures the probability of a wafer that is classified as "positive class" is truly positive. It focuses on reducing FP. For example, let positive class stand for low yield wafers. By increasing Precision, it helps reduce cases where high yield class wafers are misclassified as low yield wafers. From a production control point of view, it helps reduce wastage of good material and avoid excessive wafer fabrication.

Recall evaluates the ratio of TP over all Positive Class wafers. It is defined as

$$R_m = \frac{TP_m}{TP_m + FN_m} \tag{10}$$

It aims to reduce FN, which is the case when actual low yield class wafers are not fully identified. Both metrics are equally important as we need to have balance between Precision and Recall to better control production cost and shipment forecast. Therefore, in this framework, we use the F1 metric which takes into account both Precision and Recall. F1 score is defined in Ref. [11] as

$$F1_m = 2 * \frac{P_m \times R_m}{P_m + R_m} \tag{11}$$

For a multi-class classification problem, metrics can be computed using a micro averaging or macro averaging method. Micro averaging method is to compute the probability with total number of all TP, FP or FN. In contrast, the macro averaging method takes the average performances for each class. It is known that macro-averaged scores are more influenced by the performance of rare categories as mentioned by Y. Yang *et al.* in Ref. [12]. Therefore, F1-macro is the apt model evaluation metric for our proposed framework. Since F1-macro treats all classes equally, it can be mathematically defined as

$$F1(macro) = \frac{\sum_{m=1}^{M} F1_m}{M} = \frac{2}{M}\sum_{m=1}^{M}\frac{P_m \times R_m}{P_m + R_m} \tag{12}$$

IV. FINAL TEST YIELD CLASSIFICATION FLOW

The overall FT yield classification flow framework is illustrated in Fig 2. In this section, we will explain the following steps in detail.

A. DATA PREPROCESSING

1) NUMERICAL AND CATEGORICAL INPUT DATA

The numeric input data are the WAT parameters, whose range varies widely between 10^{-13} and 10^{3}. It is important to keep all the numerical values in a similar range of value or magnitude. To do this, first, a WAT parameter standard scaler is generated by fitting with historical production WAT data. For dimension reduction purpose, the Pearson Correlation is calculated for the WAT data and highly correlated WAT parameters are removed if their correlation coefficient exceeds 0.9. Thereby, a WAT standard scaler with reduced dimension is generated. Any new incoming production WAT data are transformed using this scaler.

The categorical input data describe the variety in the manufacturing configurations. In general, the categorical data include wafer technology, RAM/ROM versions, firmware versions, package types, product functionality, fab and test locations, test program versions, tester and handler types etc. They are descriptive string type data and can be either nominal or ordinal. The number of parameters and their values are different across products and production lines. For example, IoT products tend to have many customized firmware versions while audio products' firmware version is relatively more standardized. The most significant categorical parameters are selected using ANOVA analysis. Categorical parameters with significance levels corresponding to a $p-value \geq 0.05$ are removed. To overcome the limitations

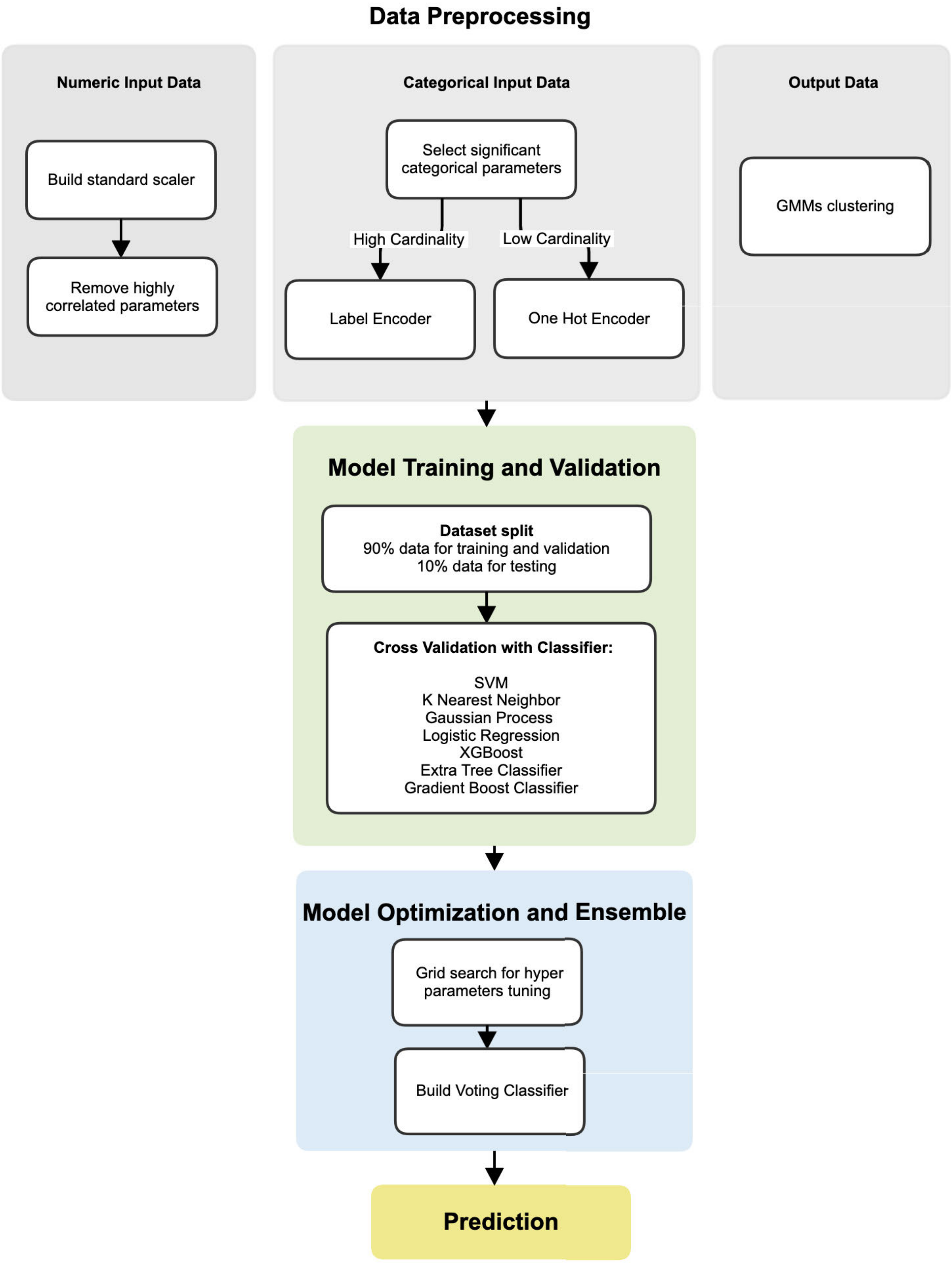


FIGURE 2. Flow chart for final test yield classification.

imposed by the Label Encoder and One Hot Encoder methods as discussed previously, both of them are used for categorical input conversion. Based on the overview from Kline [13], a sample size of at least 10 per parameter is required to obtain trustworthy results. Therefore, we propose here a guideline that when the ratio between the number of data points and the number of input parameters is larger than 10, it is recommended to use the Label Encoder and for all other cases, the One Hot Encoder is preferred.

2) OUTPUT DATA

The output data "FT Yield" is a continuous random variable. To convert yield prediction into a classification problem, we need to carry out output labeling. In this paper, we propose to use the GMM for FT yield clustering and labeling. The number of classes for binning and evaluation can range anywhere from one to four. The GMM model with the lowest Bayesian Information Criterion (BIC) score is selected. BIC is a common model selection criterion which allows for a penalty based approach for statistical mixture distribution fitting to sample data. The penalty term in BIC prevents redundant overfitting of the data. It is easy to interpret by visual inspection and can be used generally with any prior [14]. We avoid considering a class size higher than four as it may result in poor prediction performance due to highly imbalanced datasets and is practically unnecessary for production wafers disposition in the semiconductor domain.

B. MODEL TRAINING AND VALIDATION

After data preprocessing, the train test dataset split is carried out. 90% of the data are used for training and validation while 10% of the data are used for testing at the final stage. A ten split stratified cross validation step is used for model selection, because its bias and variance are relatively lower compared to regular cross validation methods as concluded by Kohavi *et al.* in Ref. [15]. Several popular and diversified classifiers are applied in this step, as listed below along with the justification for their choice.

Support Vector Machine Classifier (SVC) uses hyper plane and kernel tricks to classify different groups. It was used in several semiconductor yield related studies [3], [6]. The K Nearest Neighbor (KNN) determines the result based on a majority vote from K closest neighbors. The neighbors are selected based on the distance metric function. It is a non-parametric algorithm [16] and easy to implement but very sensitive to training samples. It is suitable for classification problems without any prior knowledge on the training dataset because the KNN algorithm does not require any assumptions on the underlying data distribution. Gaussian process is a natural way of defining prior distributions over functions of one or more input variables [17]. It is widely used in statistical settings and machine learning applications due to its high flexibility, ability to render interpretable results and its conceptual simplicity [18].

For model selection here, we use the Gaussian Process Classifier (GP) with a Laplace's method for approximating the Bayesian inference. The reason we choose GP is that it is robust to noisy data and able to work even for small datasets. Moreover, models defined with GP can discover higher-level properties of the data, such as which inputs are relevant to predicting the response [17]. Logistic Regression (LR) models are used to understand data from a wide variety of disciplines. It is best known and used in the medical and healthcare domains. Also, it is commonly used in social sciences, economic research and in physical sciences. LR is one of the statistical tools used in Six Sigma quality control analyses, and it plays an important role in the data mining domain [19]. LR is able to explain the relationship between one dependent data variable and one or more nominal and ordinal independent variables, which is suitable for the FT yield classification. Therefore, it is used as the benchmark for model performance measurement.

The remaining three classifiers used are ensemble learning methods based on decision trees. Extra Tree Classifier (XT) builds an ensemble of unpruned decision trees according to the classical top-down procedure. It essentially consists of randomizing strongly both attribute and cut-point choice while splitting a tree node [20]. The benefit of XT is that the variance is smaller compared to weak randomization methods like Random Forest. The other main strength of the XT algorithm is its high computational efficiency. The Gradient Boost (GB) model utilizes machine learning based boosting method. It can be used as a generic algorithm to find approximate solutions to the additive modeling problem [21]. It improves weaker learner's performance by minimizing the loss function. The reason we choose GB is that it produces competitive, highly robust and interpretable procedures for classification [22]. Finally, XGBoost (XGB) provides for an efficient and scalable implementation of gradient boosting framework with L1 and L2 regularization to improve model generalization [23]. It is widely used by data scientists to achieve state-of-the-art results on many machine learning challenge datasets [24]. XGB is able to handle sparse and noisy data, and the parallel and distributed computing makes execution speed faster than the traditional GB, thereby enabling quicker model exploration.

C. MODEL OPTIMIZATION AND ENSEMBLE

Based on the above cross validation results, the top three high performance models are selected using F1-macro measurements. Grid search with cross validation is applied for hyper parameter tuning and optimization of the top models. Hard voting and soft voting grid search results are compared to define which one of them is to be used as the final voting classifier (VC). The hard voting result is computed based on an average weight for each classifier whereas soft voting is to sum up all the prediction probability values and the prediction result for each classifier. Finally, test result is generated using 10% test dataset.

V. EXPERIMENTS AND RESULTS

We have conducted experiments for three different products in this study. All production manufacturing data are provided by Silicon Laboratories. In this section, we will discuss product A's classification procedure in detail. The other two products prediction flow is similar and therefore, a summary of the results are presented in a tabular format.

Product A has 1887 backend lots, with FT yield ranging from 82.36% to 99.27%. After output data pre-processing,

TABLE 1. Input and output parameters after data pre-processing. Categorical input pre-processing using one hot encoder.

	Total 57 WAT Parameters				Total 86 Categorical Parameters				Output
Backend Lot	CONTI_M1	CONTI_M2	...	CONTI_TM	TESTER_03	TESTER_U1	...	PROGRAM_42	Class
Lot#1	-0.76357	-1.13252	...	0.38751	1	0	...	0	1
Lot#2	-0.07740	0.15604	...	0.13815	0	0	...	0	0
...	...	...	...	...	...	...	...	...	...
Lot#1887	0.12985	-0.27254	...	-0.69959	0	0	...	1	2

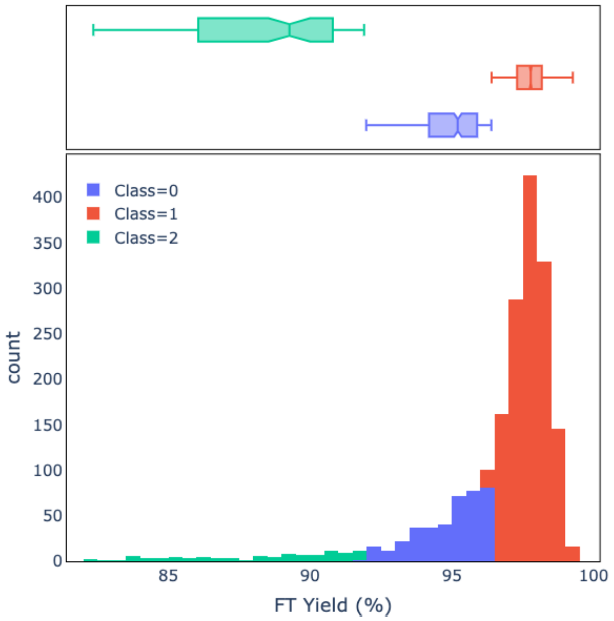


FIGURE 3. Product A - FT yield GMM clustering result.

three classes are identified using GMM clustering method as shown in Fig 3. Each backend lot consists of 74 WAT parameters and 18 categorical parameters. After numerical and categorical input data pre-processing, the number of WAT parameters has been reduced to 57 and categorical parameters reduced to 3. After applying the One Hot Encoder technique to significant categorical parameters, the dimension increases from 3 to 86. The total input dimension is therefore 143, still less than 10% of training dataset size. Therefore, the One Hot Encoder is preferred for this product. Overall input and output data after pre-processing are presented in Table.1.

To compare the performance for different encoding methods, the cross validation F1-macro result comparison between One Hot Encoder and Label Encoder and without using categorical inputs is shown in Fig.4. The F1-macro result for each model is the mean value over 10 split cross validation. By using Label Encoder, total number of input parameters is 60. For the case of no categorical inputs, the input parameters are only the WAT parameters. If we compare the mean F1-macro across all the 7 models, One Hot Encoder method has the highest score of 0.714, while Label Encoder has the lowest score of 0.652. The Label Encoder method performance is worse than without using categorical input whose score is 0.672. The reason for this trend is that product A's WAT parameters are the major root cause factor for low yield problem. However, Label Encoder brings with it the ordering problem for high cardinality categorical parameters, where the encoded number is numerically much larger than WAT parameters which ranges from −3 to 3. SVC using Label Encoder score has the lowest value of 0.345 in Fig.4. As it is based on the Euclidean distance, a high cardinality categorical parameter can distort the decision plane. The KNN is also based on Euclidean distance; however, its performance is better than SVC. The reason is that Product A's dataset is not linearly separable as illustrated by the poor LR results. The KNN in this case is more suitable for the non-linear problem compared to the Support Vector Machine Classifier (SVC) model. Besides, KNN is able to handle noisy datasets [25] which is another reason it outperforms SVC. Tree-based models including GB, XT and XGB performance are better than Non-Tree based models. This is because Tree-based models are better at handling both categorical and continuous numerical parameter values.

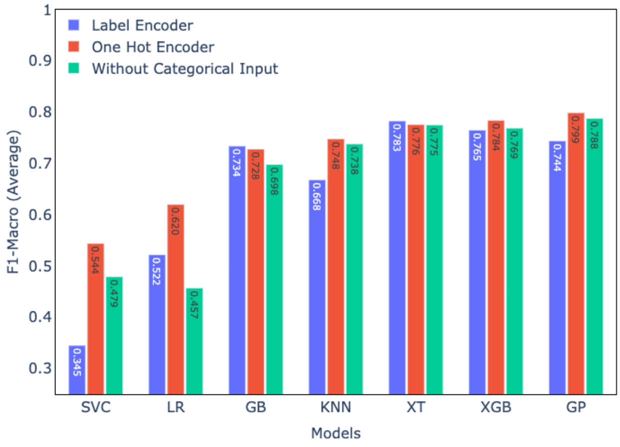


FIGURE 4. Product A cross validation: F1-macro comparison for One Hot Encoder, Label Encoder and without categorical input parameters.

During model validation and selection step, three top models are selected for further optimization. The top 3 models are XT, GP and XGB for both encoding methods based on the validation results in Fig.4. Three models' F1-macro scores are 0.776, 0.799 and 0.784 by using One Hot Encoder. The average score is 0.786. The F1-macro scores are 0.783, 0.744 and 0.765 by using Label Encoder, wherein the average score is 0.764. Although One Hot Encoder model, XT's F1-macro score is slightly lower than that using the Label Encoder, the average performance is 2.88% higher than Label Encoder. Therefore, it is proven that One Hot Encoder method is preferred for product A. The detailed number of input parameters and cross validation results' comparison is summarized in Table.2 for the three products. For Product B and Product C, cross validation of all models' results are presented in Fig.5 and Fig.6.

TABLE 2. Three product input parameters and cross validation results' comparison with different encoding methods.

	Total Number of Input Parameters			7 Models Averaged F1-Macro			Top 3 Models Averaged F1-Macro		
Input Pre-Processing	Product A	Product B	Product C	Product A	Product B	Product C	Product A	Product B	Product C
Without Categorical Input	57	67	61	0.672	0.717	0.711	0.777	0.734	0.772
Label Encoder	60	72	62	0.652	0.751	**0.727**	0.764	**0.799**	**0.800**
One Hot Encoder	143	186	70	**0.714**	**0.772**	0.721	**0.786**	0.798	0.780
Number of data points	1887	802	485						

TABLE 3. Test result for 3 products.

	Product A			Product B			Product C		
Models	Precision Macro	Recall Macro	F1 Macro	Precision Macro	Recall Macro	F1 Macro	Precision Macro	Recall Macro	F1 Macro
VC	**0.859**	0.809	**0.831**	**0.941**	**0.862**	**0.889**	**0.988**	**0.938**	**0.961**
XT	0.847	0.798	0.820	0.922	0.817	0.846	0.819	0.675	0.715
GP	0.786	**0.828**	0.803	0.826	0.832	0.829	0.925	0.925	0.925
XGB	0.844	0.754	0.787	0.932	0.849	0.879	0.905	0.863	0.882
KNN	0.787	0.696	0.734	0.904	0.827	0.852	0.966	0.813	0.867
GB	0.771	0.647	0.694	0.922	0.817	0.846	0.882	0.800	0.833
LR	0.689	0.623	0.648	0.883	0.847	0.859	0.855	0.738	0.778
SVC	0.507	0.441	0.450	0.885	0.691	0.699	0.946	0.688	0.744

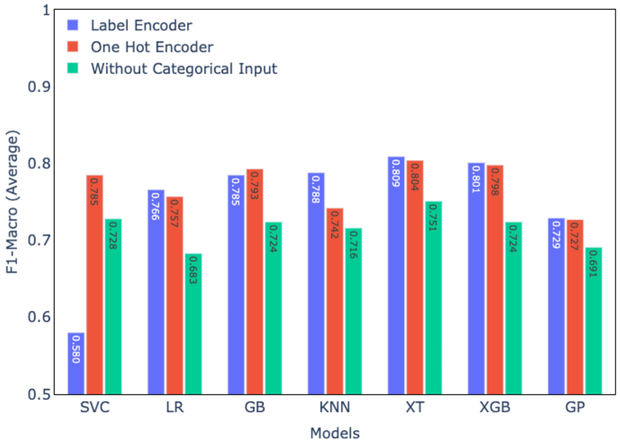


FIGURE 5. Product B cross validation: F1-macro comparison for One Hot Encoder, Label Encoder and without categorical input parameters.

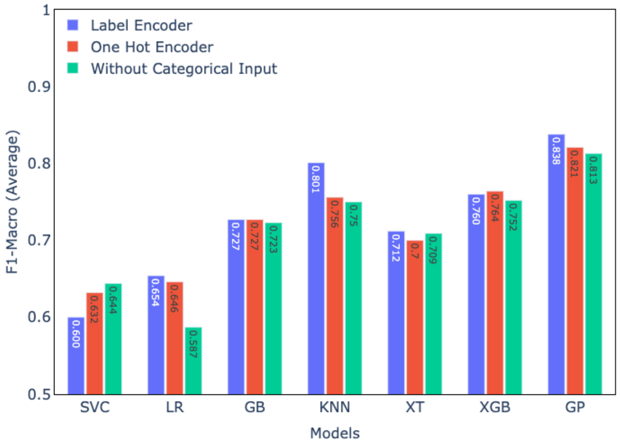


FIGURE 6. Product C cross validation: F1-macro comparison for One Hot Encoder, Label Encoder and without categorical input parameters.

For product B, categorical parameters play an important role for FT yield variation based on engineers' past experience, which is already reflected in the cross validation results. The validation results using categorical input are better than without using it for all models except for SVC. SVC performance is worst when handling a mixture of continuous numeric and categorical inputs. For the One Hot Encoder method, the average F1-macro score of the 7 models is 0.772, higher than Label Encoder's 0.751. If we only evaluate top 3 models' performance, namely XT, KNN and XGB, Label Encoder's averaged result of 0.799 is slightly better than that of the One Hot Encoder's 0.798. For Product C, the Label Encoder F1-macro results are better for both seven models and top three models. Overall, the results prove that if we choose the proper encoding method based on our proposed guideline, we are able to achieve better performance than with other encoding method or without using categorical parameters.

Based on the three products' validation results, it can be seen that the prediction performance varies between different models and different encoding methods. Therefore, when doing data pre-processing and machine learning model selection, it is important to take into consideration of the following factors which provide guidelines as to the device model to be used and its computational demand: number of input parameters, size of training dataset and model's capability to handle mixture type of input.

At the model optimization step, product A's top three model (XT, GP, XGB) hyper parameters are fine tuned with grid search cross-validation method. Finally, VC is generated using the optimized top three models. The 10% test dataset VC F1-macro results are compared with other models as presented in Fig.7. The detailed test results including precision-macro, recall-macro and F1-macro are presented in Table.3. The best outcomes of our analysis are highlighted in bold. In general, VC's performance using all the three measurement metrics are top among all the models. Product A's recall-macro score is 0.809, which is slightly lower than GP's score of 0.828. However its precision-macro score of 0.859 is much higher than GP's 0.786. Therefore, VC's overall performance is still better than GP. It can been inferred from the table that the F1-macro is a suitable metric for model selection and optimization for all the three products.

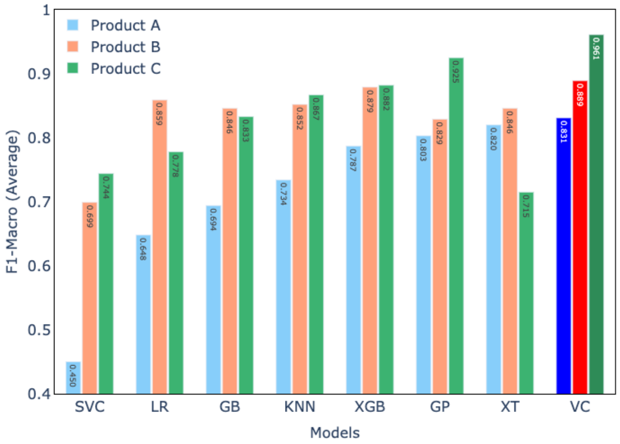


FIGURE 7. Products A, B and C test dataset prediction result with a comparison between VC (voting classifier) and other classifiers.

The VC performance is the best among all the models for all three products with an F1-macro score of 0.831, 0.889 and 0.961. We have achieved 28.17%, 3.44%, 23.48% F1-macro improvement compared to a purely LR performance for Products A, B and C respectively.

VI. FEATURE IMPORTANCE ANALYSIS

Based on the above analysis, we have now generated suitable classifiers for FT yield prediction. With this, we can proceed to carry out a feature importance analysis of the data using Gini importance [26]. Gini importance is a general indicator of feature relevance. It describes the importance of a feature by computing the normalized total reduction of the criterion (which is the objective function, in our case, it is the discrete class value of the yield sub-population - (0, 1, 2)) brought about by that feature [26]. The best classifier for product A turns out to be XT. We carry out the feature importance ranking and visualisation of the fitted XT model using the method prescribed in Ref. [27]. The resultant most important 15 features are plotted in Fig 8. It can be seen that the top three features are all categorical features - *PACKAGE_MSOP_3, PROGRAM_37,TESTER_024*. Based on the production data investigation, we confirm that the material tested with *PROGRAM_37* and *TESTER_024* did exhibit low yield problems. Corrective actions can now be taken to modify the test program and the tester as well for yield improvements.

One of the three important features identified viz. *PACKAGE_MSOP3* stands for the assembly package type, which is in turn related to the product functionality. Yield variations between different product functionality may most likely be caused by different WAT parameters. Therefore, WAT parameter feature importance analysis can now be done under two conditions. First condition is considering the dataset with assembly package *PACKAGE_MSOP3* only and the second one is using dataset without *PACKAGE_MSOP3*. The same model optimization process is then repeated for these two conditions and the resulting top 15 feature importance results are shown in Fig 9(a) and Fig 9(b). The top three most important features' names are modified with descriptive names for better understanding in Fig 9. It is worth noting that the top three most important features are completely different between these two conditions. For product with only one type of package, *PACKAGE_MSOP3*, the top three WAT parameters are all contact resistance related parameters, including *Contact_Resistance_DNW, Contact_Resistance_Nw and Contact_Resistance_TV*. While for the other package types, the top three WAT parameters are *Continuity_M6, Threshold_Voltage_N4H and Contact_Resistance_V3*. These results can now be used for fine tuning of the top three WAT parameters, separately for the different package types, so as to optimize the product yield further.

VII. CONCLUSION

In this paper, we have introduced a novel framework for final test yield prediction at the wafer fabrication stage. This is a challenging task since there are many unknown factors in between WF and FT that can cause FT low yield problems.

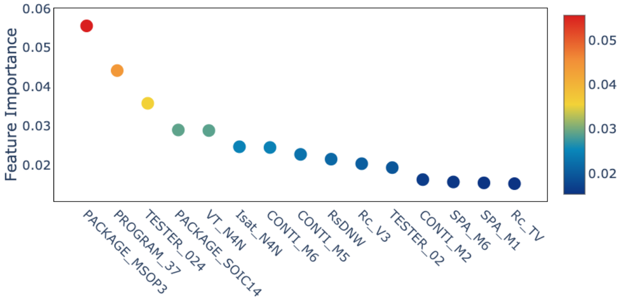


FIGURE 8. Product A ExtraTree Classifier Top 15 Feature Importance Ranking.

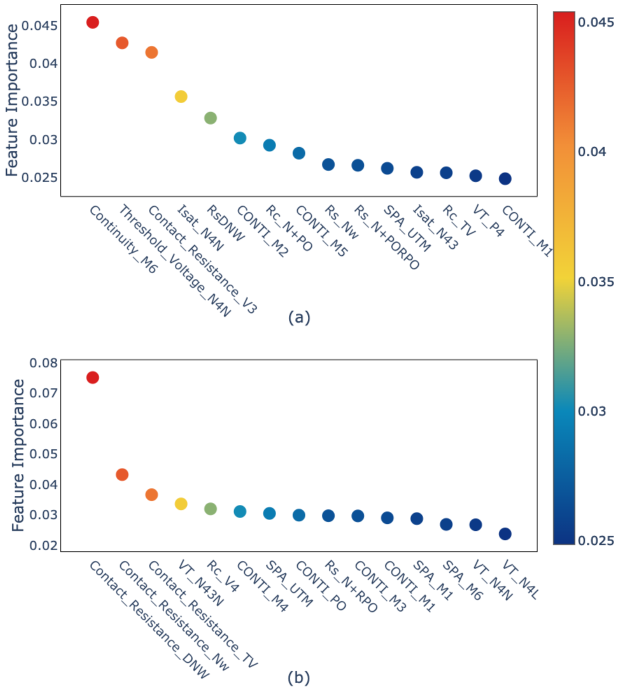


FIGURE 9. Product A ExtraTree Classifier based feature importance ranking analysis results (a) without *PACKAGE_MSOP3* and (b) with *PACKAGE_MSOP3* showing the 15 most sensitive input parameters. There is a significant change in the order of importance of the input parameter for these two cases.

Three different products' production data were fitted into the framework and all of them managed to achieve a good F1-macro score. This framework can be extended to predict any semiconductor production stage's yield based on the production data prior to the stage. The major contribution of our framework is to automatically convert semiconductor production related data into the yield prediction model without prior knowledge of the data. Furthermore, the framework introduced a model selection and ensemble method to achieve good model performance for both binary and multi-class problems.

The novelty of our study is that we propose a generic framework that can be applied to address several semiconductor manufacturing yield problems for advanced logic / memory technology nodes. The framework is robust, scalable and configurable to include both numerical as well as categorical inputs and map their relationships to the output yield of multiple product lines and also allows for automated feature importance, sensitivity analysis and multi-modal yield classification. Our framework takes into account all the manufacturing related parameters as input data, with no necessity for any manual filtering as compared to existing yield prediction models [9], [28]. Additional machine learning models can also be added as candidate models during the model selection step to accommodate other types of yield problems.

Future work will involve further methodological explorations into improving the model performance and enabling dedicated low yield root cause analysis. The low yield root cause analysis task should be able to automatically identify whether the causal factor is WAT related or production flow related so as to provide corrective and effective recommendations for faster turn around yield improvements.

REFERENCES

[1] S.-J. Jang, J.-H. Lee, T.-W. Kim, J.-S. Kim, H.-J. Lee, and J.-B. Lee, "A wafer map yield model based on deep learning for wafer productivity enhancement," in *Proc. 29th Annu. SEMI Adv. Semiconductor Manuf. Conf. (ASMC)*, Apr. 2018, pp. 29–34.

[2] J.-S. Kim, S.-J. Jang, T.-W. Kim, H.-J. Lee, and J.-B. Lee, "A productivity-oriented wafer map optimization using yield model based on machine learning," *IEEE Trans. Semiconductor Manuf.*, vol. 32, no. 1, pp. 39–47, Feb. 2019.

[3] K. Nakata, R. Orihara, Y. Mizuoka, and K. Takagi, "A comprehensive Big-Data-Based monitoring system for yield enhancement in semiconductor manufacturing," *IEEE Trans. Semiconductor Manuf.*, vol. 30, no. 4, pp. 339–344, Nov. 2017.

[4] J. Wang, Z. Yang, J. Zhang, Q. Zhang, and W.-T.-K. Chien, "AdaBalGAN: An improved generative adversarial network with imbalanced learning for wafer defective pattern recognition," *IEEE Trans. Semiconductor Manuf.*, vol. 32, no. 3, pp. 310–319, Aug. 2019.

[5] T. Ishida, I. Nitta, D. Fukuda, and Y. Kanazawa, "Deep learning-based wafer-map failure pattern recognition framework," in *Proc. 20th Int. Symp. Qual. Electron. Design (ISQED)*, Mar. 2019, pp. 291–297.

[6] Y. Kong and D. Ni, "A practical yield prediction approach using inline defect metrology data for system-on-chip integrated circuits," in *Proc. 13th IEEE Conf. Autom. Sci. Eng. (CASE)*, Aug. 2017, pp. 744–749.

[7] K.-J. Kim, K.-J. Kim, C.-H. Jun, I.-G. Chong, and G.-Y. Song, "Variable selection under missing values and unlabeled data in semiconductor processes," *IEEE Trans. Semiconductor Manuf.*, vol. 32, no. 1, pp. 121–128, Feb. 2019.

[8] S. Kang, S. Cho, D. An, and J. Rim, "Using wafer map features to better predict die-level failures in final test," *IEEE Trans. Semiconductor Manuf.*, vol. 28, no. 3, pp. 431–437, Aug. 2015.

[9] C.-F. Chien, C.-W. Liu, and S.-C. Chuang, "Analysing semiconductor manufacturing big data for root cause detection of excursion for yield enhancement," *Int. J. Prod. Res.*, vol. 55, no. 17, pp. 5095–5107, Sep. 2017.

[10] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *J. Roy. Stat. Soc., B (Methodol.)*, vol. 39, no. 1, pp. 1–22, 1977.

[11] C. J. van Rijsbergen, *Information Retrieval*. London, U.K.: Butterworths, 1979.

[12] Y. Yang and X. Liu, "A re-examination of text categorization methods," in *Proc. 22nd Annu. Int. ACM SIGIR Conf. Res. Develop. Inf. Retr. (SIGIR)*, 1999, pp. 42–49.

[13] R. B. Kline, *Principles and Practice of Structural Equation Modeling*. New York, NY, USA: Guilford publications, 2015.

[14] K. P. Burnham and D. R. Anderson, "Multimodel inference: Understanding AIC and BIC in model selection," *Sociol. Methods Res.*, vol. 33, no. 2, pp. 261–304, 2004.

[15] R. Kohavi, "A study of cross-validation and bootstrap for accuracy estimation and model selection," in *Proc. IJCAI*, vol. 14. Montreal, QC, Canada, 1995, pp. 1137–1145.

[16] A. Kataria and M. Singh, "A review of data classification using k-nearest neighbour algorithm," *Int. J. Emerg. Technol. Adv. Eng.*, vol. 3, no. 6, pp. 354–360, 2013.

[17] R. M. Neal, "Monte Carlo implementation of Gaussian process models for Bayesian regression and classification," 1997, *arXiv:physics/9701026*. [Online]. Available: https://arxiv.org/abs/physics/9701026

[18] M. Seeger, "PAC-Bayesian generalisation error bounds for Gaussian process classification," *J. Mach. Learn. Res.*, vol. 3, no. 2, pp. 233–269, Feb. 2003.

[19] J. M. Hilbe, *Logistic Regression Models*. Boca Raton, FL, USA: CRC Press, 2009.

[20] P. Geurts, D. Ernst, and L. Wehenkel, "Extremely randomized trees," *Mach. Learn.*, vol. 63, no. 1, pp. 3–42, Apr. 2006.

[21] P. Bühlmann and T. Hothorn, "Boosting algorithms: Regularization, prediction and model fitting," *Stat. Sci.*, vol. 22, no. 4, pp. 477–505, Nov. 2007.

[22] J. H. Friedman, "Greedy function approximation: A gradient boosting machine," *Ann. Statist.*, vol. 29, no. 5, pp. 1189–1232, Oct. 2001.

[23] J. Friedman, T. Hastie, and R. Tibshirani, "Additive logistic regression: A statistical view of boosting (With discussion and a rejoinder by the authors)," *Ann. Statist.*, vol. 28, no. 2, pp. 337–407, Apr. 2000.

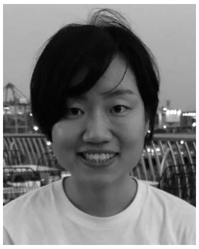
[24] T. Chen and C. Guestrin, "Xgboost: A scalable tree boosting system," in *Proc. 22nd ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2016, pp. 785–794.

[25] H. A. Abu Alfeilat, A. B. Hassanat, O. Lasassmeh, A. S. Tarawneh, M. B. Alhasanat, H. S. Eyal Salman, and V. S. Prasath, "Effects of distance measure choice on K-Nearest neighbor classifier performance: A review," *Big Data*, vol. 7, no. 4, pp. 221–248, Dec. 2019.

[26] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *J. Mach. Learn. Res.*, vol. 12, pp. 2825–2830, Oct. 2011.

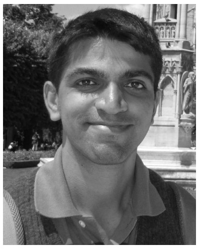
[27] L. Breiman, "Random forests," *Mach. Learn.*, vol. 45, no. 1, pp. 5–32, 2001.

[28] B. Lenz and B. Barak, "Data mining and support vector regression machine learning in semiconductor manufacturing to improve virtual metrology," in *Proc. 46th Hawaii Int. Conf. Syst. Sci.*, Jan. 2013, pp. 3447–3456.



DAN JIANG received the B.Eng. degree in electrical and electronics engineering and the M.Eng. degree in communications engineering from Nanyang Technological University (NTU), Singapore, in 2013 and 2017, respectively. She is currently pursuing the Ph.D. degree with the Engineering Product Development (EPD) Pillar, Singapore University of Technology and Design (SUTD). Since 2013, she has been working as a Product Test Engineer at Silicon Laboratories International, on semiconductor product test and data analytics tools development. Her current research interests include semiconductor manufacturing yield prediction as well as big data and artificial intelligence for semiconductor process and quality optimization.

WEIHUA LIN received the B.Eng. degree in nuclear electronics from the University of Science and Technology of China, in 1991, and the M.Sc. degree in electronics engineering from the National University of Singapore (NUS), in 2001. He is currently the Senior Product Test Engineering (PTE) Director of Silicon Laboratories International. He has 30 years of working experience in several semiconductor companies, such as National Semiconductor, Lucent Microelectronics, and Silicon Labs. He has been focusing on IC test and product engineering as well as field application and customer support. He is currently providing technical consultation role in IC and module test development, product qualification, and yield optimization to achieve good product quality at lower possible cost.



NAGARAJAN RAGHAVAN (Member, IEEE) received the Ph.D. degree in microelectronics from the Division of Microelectronics, Nanyang Technological University (NTU), Singapore, in 2012. He is currently an Assistant Professor with the Engineering Product Development (EPD) Pillar, Singapore University of Technology and Design (SUTD). Prior to this, he was a Postdoctoral Fellow with the Massachusetts Institute of Technology (MIT), Boston, and at IMEC, Belgium, in joint association with the Katholieke Universiteit Leuven (KUL). His work focuses on integrated machine learning and physics-based reliability assessment, characterization and lifetime prediction for nanoelectronic devices as well as material design for reliability, uncertainty quantification for additive manufacturing and prognostics, and health management of electronic devices and systems. He has authored/coauthored more than 190 international peer-reviewed publications and five invited book chapters. He was an Invited Member of the IEEE GOLD Committee (2012–2014). He was a recipient of the IEEE Electron Device Society (EDS) Early Career Award in 2016, an Asia–Pacific recipient for the IEEE EDS Ph.D. Student Fellowship, in 2011, and the IEEE Reliability Society Graduate Scholarship Award, in 2008. He serves as the General Chair for the IEEE IPFA 2021 at Singapore, and has consistently served on the review committee for various IEEE journals and conferences, including IRPS, IIRW, IPFA and ESREF. He is currently serving as an Associate Editor for IEEE Access.