Noname manuscript No.
(will be inserted by the editor)

Continuous Prediction of Manufacturing Performance Throughout the Production Lifecycle

Sholom M. Weiss · Amit Dhurandhar · Robert J. Baseman · Brian F. White · Ronald Logan · Jonathan K. Winslow · Daniel Poindexter

Received: date / Accepted: date

Abstract We describe methods for continual prediction of manufactured product quality prior to final testing. In our most expansive modeling approach, an estimated final characteristic of a product is updated after each manufacturing operation. Our initial application is for the manufacture of microprocessors, and we predict final microprocessor speed. Using these predictions, early corrective manufacturing actions may be taken to increase the speed of expected slow wafers (a collection of microprocessors) or reduce the speed of fast wafers. Such predictions may also be used to initiate corrective supply chain management actions. Developing statistical learning models for this task has many complicating factors: (a) a temporally unstable population (b) missing data that is a result of sparsely sampled measurements and (c) relatively few available measurements prior to corrective action opportunities. In a real manufacturing pilot application, our automated models selected 125 fast wafers in real-time. As predicted, those wafers were significantly faster than average. During manufacture, downstream corrective processing restored 25 nominally unacceptable wafers to normal operation.

Keywords manufacturing · data mining · prediction

1 Introduction

The manufacturing of chips is a complex process, taking months to produce a modern microprocessor. Starting from the initial wafer, the chips are pro-

Sholom M. Weiss, Amit Dhurandhar, Robert J. Baseman, Brian F. White
IBM Research Yorktown Heights NY 10598 USA
E-mail: sholom@us.ibm.com, adhuran@us.ibm.com, baseman@us.ibm.com, bfwhite@us.ibm.com

Ronald Logan, Jonathan K. Winslow, Daniel Poindexter
IBM Microelectronics Fishkill NY 12533 USA
E-mail: llogan@us.ibm.com, jkwinslo@us.ibm.com, poindext@us.ibm.com

duced by the application of hundreds of steps and tools. Given the complexity of these processes and the long periods needed to manufacture a microprocessor, it is not surprising that extensive efforts have been made to collect data and mine them looking for patterns that can eventually lead to improved productivity (Goodwin et al., 2004), (Harding et al., 2006), (Melzner, 2002), (Weber, 2004), (Weiss et al., 2010). Among the primary roles of data mining in semiconductor manufacturing are quality control and the detection of anomalies. When something goes wrong, such as a significant reduction in yield, the data are pulled and examined to find probable causes. From a data collection perspective, tens or even hundreds of thousands of measurements are taken and recorded to monitor results at different stages of chip production. Since, the objective is mostly to monitor quality of production, wafer measurements can be sparsely sampled, typically less than 10%.

In contrast to monitoring production for diagnostic application, in this paper we consider prediction of final chip performance. Each wafer, and its constituent chips, has an incremental history of activity and measurement accrued during its manufacture. In its purest and most ambitious form, our objective is to predict the final outcome of each wafer in terms of critical functional characteristics. Months may pass before a chip is completed, hence there is great interest in mining production data to predict its performance prior to final testing (Irani et al., 1993), (Apte et al., 1993), (Fountain et al., 2000). While many alternative testing measurements are reasonable to measure the health of a wafer, in our initial applications, we designate a proxy for microprocessor speed as the predicted outcome. Thus during manufacture, the average speed of the finished product is estimated at a time far from completion.

Using the same data that are recorded to monitor individual elements of the fab manufacturing process, the final performance of a wafer is estimated. This exercise implicitly raises, and in part addresses the question of how much power such a set of measurements, designed explicitly for the purposes of monitoring unit and integrated process performance, has for this very different prediction application.

Measures of speed are the final critical characteristics used in this paper to measure outcome. A chip running too slow is clearly a negative outcome, as is a chip running too fast, since it may consume too much power. The advantages of accurately predicting final performance are manifold. Among the actions that might be taken are as follows:

- Correct wafers with expected poor performance.
- Prioritize manufacturing times for expected best-performing wafers allocating them to high-priority customers, With an average wafer manufacturing time of many months, theoretically the highest-yielding wafers could be finished earlier than otherwise expected.
- Queue wafers based on expected performance and current demand.

Predicting final performance based on incomplete measurements is a difficult task. It implies accurate and highly predictive measurements. The benefits

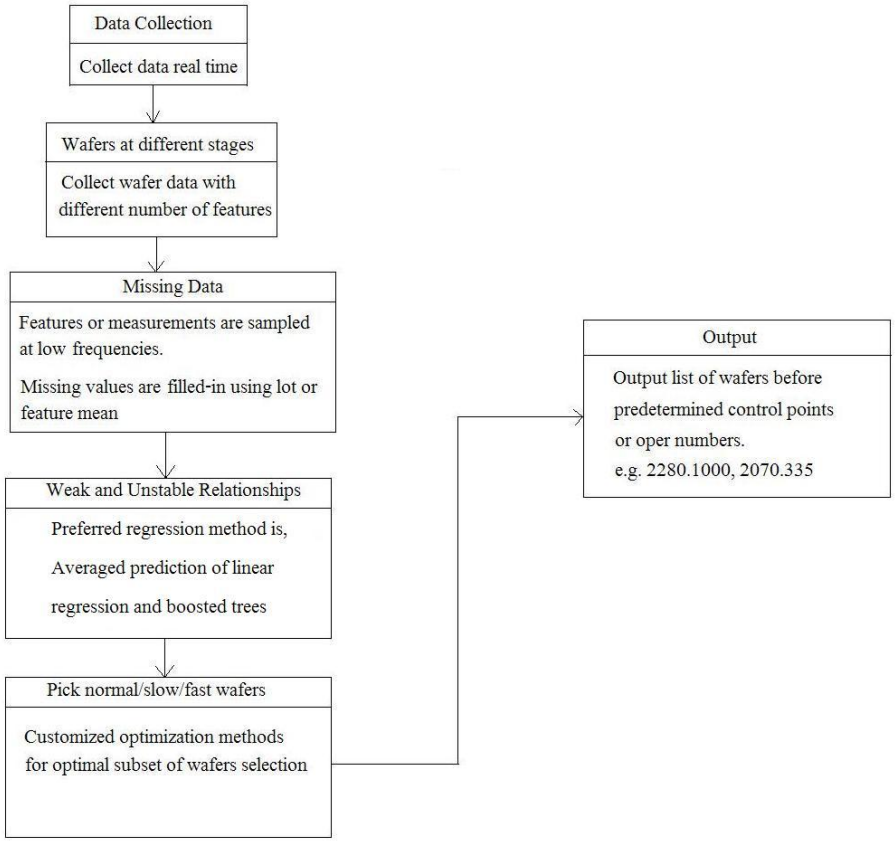

Fig. 1 Overview of the applied methodology

can potentially be great in improving manufacturing efficiency and yield and the early detection of potentially weak outcomes. From a machine learning perspective, technical difficulties abound, from time-varying populations and the inherent instabilities of massively missing data. To address these difficulties, knowledge-based methods for missing values are developed, specialized sampling techniques are employed, and combined learning methods such as linear and boosted trees are invoked. An overview of the applied methodology is shown in Figure 1.

In Section 2, we provide more domain specific details for semiconductor manufacturing. In Sections 3 and 4, we describe the development of regression models predicting microprocessor speed. In Sections 5-7, we then describe the further development of these models for real-world applications requiring the identification of normal wafers and requiring the identification of aber-

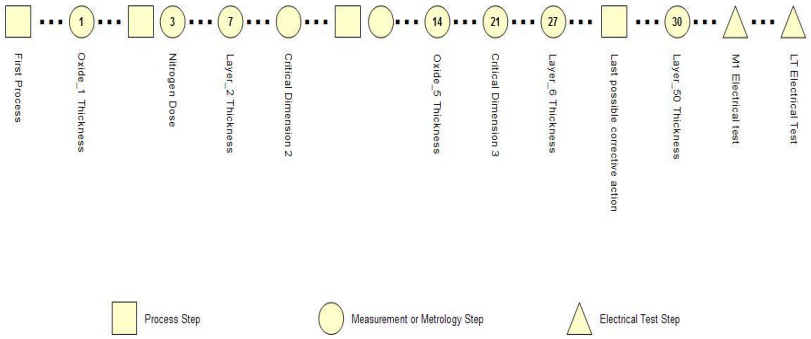

Fig. 2 Stages of wafer/chip manufacturing. A wafer moves from left to right. Circles with numbers reflect measurements used in these models

rantly fast or slow wafers. Additionally, the performance of these real-world applications is measured in terms of the benefits of suggested actions.

2 Background

It takes a few months to manufacture a microprocessor, during which a wafer undergoes incremental processing (nominally value adding) and measurement (nominally non-value adding) operations. During production, in total, thousands of different measurements are taken, and while some relatively small number of measurements are made on at least one wafer in every lot, as few as only 5 to 10% of the wafers may undergo any single measurement. Furthermore, there may be varying degrees of coordination in the selection of lots and wafers between measurements. Thus some lots and wafers may have many measurements while other lots and wafers have only a very few or no measurements beyond the relatively small set of compulsory measurements.

Figure 2 illustrates the progression of a wafer through the line for a mainframe microprocessor. Here, a wafer starts at step 1, where a Pad Oxide operation is performed, and proceeds to increasingly numbered steps. Wafers typically travel in groups of 25, called a lot. Measurement steps monitoring the quality of individual processing steps, or assessing the quality of integrated processing progress, follow many processing steps. These measurement steps may be performed on randomly selected lots, with a lot sampling frequency determined by quality control metrics, and most commonly on 2 to 4 randomly selected wafers within each sampled lot. The same wafers may not necessarily be measured on following steps, so that most wafers will have a random collection of measurements, with many of them unknown.

The target outcome for prediction is a real-valued electrical test (PSRO) serving as a proxy for the average microprocessor speed on the wafer. The higher the PSRO the slower the wafer. This test is conducted on all wafers as one of the last set of electrical tests. In an ideal implementation, we would

update a (regression) prediction of PSRO measured at final test for each wafer after each processing and measurement step.

In our initial implementations, we established a limited number of landmarks in the production process where predictions are updated. These landmark steps are selected based on knowledge of the production line. While the ideal implementation of continual prediction covers all possibilities, a reasonable alternative is to make the predictions after these critical landmark steps. This coordinates the data collection for all wafers, so that they are synchronized relative to completeness of data, and more amenable to statistical modeling. Engineering knowledge also plays an important role in defining the landmarks. From the engineering perspective, landmarks may be selected based on the potential actions that may be taken. In our case, we can continue to model and predict after each step, and predictions tend to get more accurate as more steps are completed. However, corrective processing action is only feasible during early stages of manufacture, that is, with less than 50% of steps completed. In Figure 2, we might establish landmarks at step 7 and 14, where predictions after step 14 might be useful for customer triage, but no corrective processing action can be taken.

For our primary application, the most critical prediction of final speed was made at a landmark marking the last time for corrective processing action. If a wafer's predicted speed was unacceptably high or low, its progress on the line was halted until an engineering review and response, including tailored remedial downstream processing. The basic unit for sampling is a wafer and its historical record. Depending on the application and manufacturing line operation policies, it may be necessary to predict final mean or median speed by individual wafer or by lot. In our initial implementation, we predicted mean lot speed by averaging the predictions of the individual wafers comprising those lots.

All our experiments were performed in a major production fab, not an R&D facility. This multi-billion dollar fab is used to manufacture IBM products and customer products under contract such as microprocessors for game consoles. Multiple products are manufactured on the same line, and at each step, multiple sets of tools are available to perform the same function. We have access to all stored fab data and can perform data analyses. Under special approval, we were allowed to perform a restricted set of experiments for a small set of wafers within the standard production line consistent with engineering protocols to improve wafer performance. We had absolutely no mandate or capability to alter the overall recipes of production or to manage supply chain for customers. We proceed with essentially no change to protocols in place for chip production.

3 Methods and Procedures

Our application has the following input and output characteristics:

- Input: Sparsely sampled control measurements on a wafer such as physical measurements (wafer mean film thicknesses, dopant doses), lithographic metrology (wafer mean critical dimensions and layer to layer overlays), and electrical measurements (wafer mean individual transistor to small scale macro performance measures). Defectivity measurements, having relatively little influence on PSRO were not included.
- Output: Performance indicators such as speed or power consumption measurements. In our studies, we use the electrical test (PSRO) serving as a proxy for microprocessor speed to be our target.

Using these input measurements, the objective is to predict the output measure long before it is actually measured. In the ideal application a variety of engineering and management actions may be initiated based on the continuously updated predictions of final wafer characteristics. Unwarranted corrections to the wafers or supply-chain actions may be very costly, in the worst case ruining salable products. This imposes a clear requirement that the predictions be made with high precision. Thus, depending on the expected accuracy of prediction, we restrict actions to those wafers that are predicted to be most deviant. In our application these are the estimated fastest and slowest wafers.

3.1 Collecting Data

In this work, we explore the use of preexisting control measurements for predictive applications.

The data are all real valued and can be posed in a standard vector format. For any wafer, W(i), the target speed prediction, can be made by mapping from the input vector X(i) to the output, Y(i). We collected data and made predictions using wafer mean and median values and did not explore data and predictions by individual chip or wafer region.

Figure 2 illustrates the progression of a wafer through the line for a mainframe microprocessor. Here, a wafer starts at the step labeled First Process and proceeds to the right through increasingly numbered steps. Thousands of different measurements may be defined for a given manufacturing route and are in place to assess the quality of unit processes or integrated processing progress.

To reduce cost and manufacturing cycle time, these measurements are made only on a fraction of lots, and on a fraction of the wafers within each lot. The fractions sampled are generally determined by quality control considerations. Thus, while a relatively small number of compulsory measurements are made on many wafers in every lot, as few as only 5 to 10% of the wafers may undergo any single measurement. Furthermore, there may be varying degrees of coordination in the sampling of lots and wafers between measurements. As a result, some lots and wafers may have many measurements while other lots and wafers have only a very few or no measurements beyond the relatively small set of compulsory measurements. While such measurement sampling

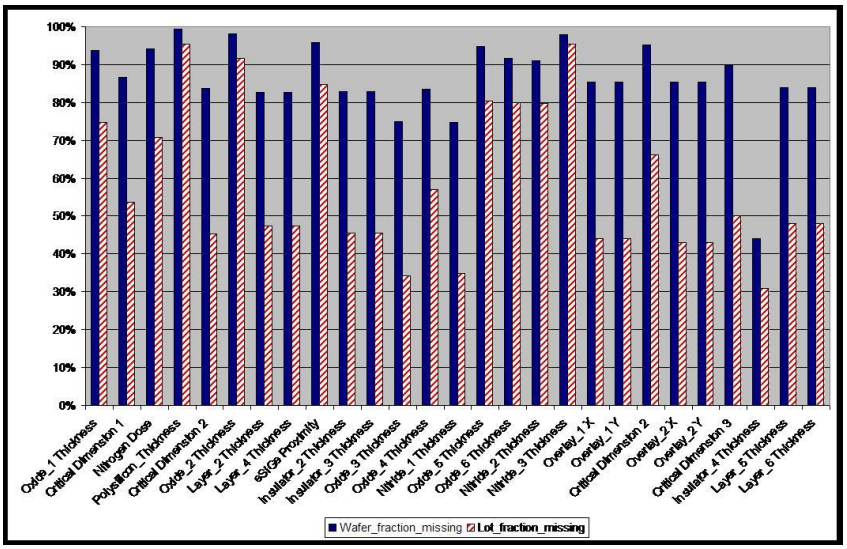

Fig. 3 Missing data characteristics.

policies are optimized for control applications, they are obviously suboptimal for predictive applications where the ideal would be all measurements on all wafers.

The complete data for wafers that have finished final test testing can be readily retrieved from a database. This data is complete only in the sense that all measurements that will be ever made on these wafers have already been made. The measurements for many lots and wafers may be missing, and the types of missing measurements are inconsistent from wafer to wafer. However, the wafers of interest, for which actionable predictions are to be made, have not completed even half of the full processing flow. Thus the input data vector for those wafers is additionally highly censored.

This results in a standard data presentation with one practical deficiency: Most of the data items are missing. Figure 3 presents the wafer and lot fractions of missing data measurements from a sample of 6435 completed wafers. Approximately 90% by wafer, and from 50% to 90% by lot of the nominally anticipated measurements are missing. The frequency of sampling varies by measurement and is determined by the engineering team based on their view of the importance of the measurement and quality control considerations. It's the measurement that is randomly sampled, not the wafer. It's not the case that a wafer has either completely sample measurements or not– each wafer will be be missing random selections of measurements. We have no capability to change that frequency, and we use the data as given.

To estimate whether unit and integrated processes are operating within specification, sampling of some measurement values is adequate to collect mean values for quality control. When the goal is modified to use these same measurements for prediction, the inadequacy of current data collection standards

is manifest. With 90% missing, prediction is not feasible. How then do we transform an intractable problem due to lack of data to a feasible application with adequate data?

Given knowledge of which measurements had the most significant predictive power, one could imagine implementing a full lot and wafer test on a limited set of measurements, as a long term strategy. Depending on the particular predictive application implemented, some other quality control measurements could be reduced in frequency, offsetting the additional cost and time associated with full test for the highly predictive measurements.

In theory, another strategy would be to replace missing measurements with predictions from a set of virtual metrology models. These models use process trace data, process consumable characteristics, and chamber state information, generally available for all wafers, as inputs to predict the results of unit processes. However the accuracy of such predictions for many processes is not yet well established, especially over tool maintenance cycles. So this must be regarded as an ambitious, risky, long term strategy (Khan et al., 2007), (He and Zhu, 2012), (Zhu and Baseman, 2012).

However, for immediate and practical action, the current data samples must be used as is. Wafers are processed and measured together as a lot, explicitly so in batch processing tools, implicitly so in single wafer tools, undergoing the same process simultaneously, in the same tools. We can take advantage of these relationships to improve estimates of missing measurements. Consider the following hierarchy of possibilities for estimating a missing measurement for a wafer.

- Full sample measurement mean
- Lot measurement mean
- Split lot measurement mean

The simplest idea is to estimate missing measurements by the global measurement mean, using the complete sample. This approach would allow machine learning to function, possibly succeeding when the most predictive measurements are more fully sampled. In our application, over 90% of measurements are missing, and this approach fails to predict accurately.

The second idea is to use the wafer's lot mean. Because the wafers with a lot are generally processed identically, this approach can improve results greatly over using a global mean.

The next idea improves somewhat over the lot mean. In the course of production, some wafers may temporarily be split from their parent lots into child lots to undergo rework processes, travel along branch routes for measurements, act as send aheads for control feedback, or test improved processes. The child lots may undergo single or multiple processes at different times and by different tools. In this case, at the expense of additional record-keeping, the individual child lot means are used for estimating each wafer's missing values, based on each wafer's lot membership at each process, rather than using the full lot means.

The variance of a measurement within a lot is usually much less than between lots because the within-lot wafers use the same tools on each step. For different lots, tools from varying generations and manufacturers are available for each step. That explains the rationale for using within lot estimates for missing values. Of the three alternatives cited here, in our application, the detailed child-lot option yields the best predictive accuracy.

It is also important to note that other machine learning methods for filling in missing values, such as expectation-maximization based methods, were tested and resulted in less accurate predictions than the suggested approach; possibly because they are agnostic. Moreover, such methods are significantly more computationally expensive, which is undesirable in the anticipated large-scale applications.

3.2 Sampling and Evaluation

In the previous section, we reviewed the sampling of measurements. This is inherent in the operation of the fab, and is something that is unlikely to be modified due to time and cost constraints.

In this application, our data set is continually growing due to the manufacture of additional chips. If the data are stable and are from the identical population, the complete sample would be used for learning. Once the manufacturing process has stabilized, the physical relationships among the measurements should also stabilize. The largest sample in a high-dimensional feature space is likely best for learning and most representative of the complete population.

Here we see competing themes for learning. Depending on the stability of the manufacturing processes, we are pulled in different directions. If the population is stationary, the standard train and test model can be applied on the full sample. However, it is not unusual for the population to be non stationary in the complex manufacturing environment for semiconductors. Yield or performance enhancing process adjustments may continue over a significant portion of a product's life cycle, while nominally stable processes may evolve within or in some cases temporarily outside of control limits. In these environments, the population acts like a time series, where the most recent data are more valuable that older historical data.

To make predictions and measure performance, a separate train and test set of prior results are essential. Clearly, lots must be completely separated, given their underlying relations among their wafers. Because results may change over time and the population is not stationary, independent time-ordered sets are advantageous over randomly sampled wafers or lots. This time-ordering corresponds to the real manufacturing environment, where we look at recently manufactured wafers to predict future wafer performance. This application has thousands of wafers to sample, and ample data are present for training and testing. If the populations from these two time periods are very similar, some reasonable percentage of the complete sample could be used for training

and testing, for example 70% training and 30% testing. However, given the nonstationary nature of data, better results can be achieved by restricting the training data to a window of k days. This reflects the usual time-series expectation – for non-periodic data– that the more recently completed wafers are most indicative of expected results for current wafers that are still progressing. In our case, we use the following constraints on data sampling:

- one year of data for complete sample of n wafers
- k wafers for training
- n - k wafers for testing

The value of k is typically much smaller than n, perhaps 3 months of data. However, the choice of k must also be verified by testing, and several possibilities are examined. The population may change, and that implies that these values and experiments may be performed periodically to verify previous choices. Yet, we know that even good performance on test cases could change over time, so it is wise to have a large test set taken over a longer timeframe that is representative of varying conditions. In particular, we have gone through periods where pessimism is more warranted in predictions, especially when changes are being made to enhance the manufacturing processes. The expectation is that updates to the manufacturing process are implemented with an eventual return to stability. Thus we adopt an emphasis on recent data for training, and more extensive historical data for testing.

Algorithm 1 illustrates the evaluation procedure that is used to estimate model predictive performance for the current wafers and to determine sample and model characteristics. In a static environment, one might simply choose those modeling characteristics that minimize error. However, the application environment is dynamic–wafers enter and leave the manufacturing line and processes and fab performance may change. Directions in fab and model performance may also change, but not on a daily basis. Therefore some overall knowledge about the trends in model performance must be applied. One reasonable strategy is to make major modeling decisions in an experimental phase, and then watch trends over time before making major revisions. However, the estimates for individual wafers are critically important for decisions made on a daily basis. Typically, only wafers with the most extreme predictions will be selected for actions. The procedure of Algorithm 1 is used for our internal estimates. Real-world decisions are made by selecting wafers for revision, and the consequences of those decisions are the ultimate evaluation of predictive performance.

4 Methods for Learning

From a machine learning perspective, the objective is to predict the eventual outcome of product testing, PSRO measured at final test. Given a set of real-valued measurements including the outcome, regression methods are applicable. We could also view the task as classification, when well defined speed

Algorithm 1 Model Evaluation

1. Collect sample S1 of wafers with known completed measurements
2. Collect independent sample. S2, for testing.
3. Learn a prediction model from S1 and evaluate on S2.
4. For step 3, any learning model learned from S1 is acceptable, subject to fair performance evaluation on S2.
5. example of a prediction model for step 3 is a linear model, where
   Given n wafers in S1, each with j measurements find the best set weights such that error is minimized as in this computation for the k-th wafer.
   wt(1)* M(1)+ ...+ wt(j)M(j) = P(k)
6. Error is estimated by MSE or MAD for difference in true value T(j) and predicted wafer target measurement P(j),

Method	Step 3	Step 7	Step 14
Boosted trees and linear	0.04	0.08	0.69
Boosted trees	-0.01	0.05	0.62
Ridge	-0.13	0.03	0.59
SVM	-0.02	0.04	0.60
HMML	-0.22	0	0.16
BTM	-0.25	-0.15	0.14

Table 1 Above is the comparison in terms of average R^2 of different state-of-the-art learning methods at different steps in the processing (Figure 2) of a wafer based on weeks of daily experimentation.

thresholds can be specified. Early exploratory experiments not described here demonstrated far better predictive value for regression analysis than classification. Predicting the continuous PSRO provides a natural ordered ranking of the wafers. The most likely candidates for correction are those with the most extreme predictions or those outside a specified normal range.

Using the procedure in Algorithm 1, different learning methods can be compared and the one with best results selected. This is a standard approach to selecting learning algorithms in a stationary population when predictive performance is the primary goal. However, the fab population is not stationary, and periods of relative stability and periods of rapid change are both anticipated.

To deal with these changes and also based on experimentation over many weeks, linear regression and forests (boosted trees) (Schapire, 1990) methods were combined and used for modeling with carefully customized train and test protocols.

The results of testing several learning methods are shown in Table 1. In this table, SVM stands for support vector machines (Vapnik, 1998). HMML stands for a hidden markov model based method with lasso regression in every state (Liu et al., 2009). BTM stands for best of the time series methods using SPSS

expert modeler. The ensemble learning method, which averages the predictions of boosted trees and linear regression performs the best overall. The reported results are R^2 values averaged over weeks of experimentation. R^2 is a standard measure in statistics used to evaluate regression algorithms. It is defined as,

$$R^2 = 1 - \frac{mse(M)}{mse(\mu_t)}$$

where $mse(M)$ denotes the mean squared error of a model M on the test set while $mse(\mu_t)$ denotes the mean squared error of the training set target mean on the test set. In our case, M would signify the regression functions learned using the different learning methods while μ_t would signify the mean PSRO computed over the training set. Hence, R^2 values closer to 1 imply that M is much superior to μ_t. Negative R^2 values imply that using M is inferior to using the simple prediction of the training set mean, and are highly suggestive of nonstationarity in the underlying input output relationships.

The classical linear model is a simplified model that assumes a fixed representation. In our experiments, it usually performed worse than the forests which are collections of decision trees generated from random subsampling of the training data. However, in nonstationary environments, i.e. fab performance is evolving, the linear method could win. The reason is likely tied to its simplified and restricted perspective that does not overfit the data and is more robust.

The forests, numbering in the hundreds of decision trees, are capable of modeling much more complex functions than the single linear regression model. When the population is stable, the forests will perform much better. When fab behavior is evolving, the results can weaken because the fit to the (stable) training data is too tight.

The predictions of these two methods can be averaged. This is an effective strategy for dealing with evolving fab dynamics. Combining two or more independent methods is known to often give better results (Bao et al., 2009), (Dzeroski and Ženko, 2004), (Bell et al., 2009). The methods can be evaluated independently and in combination. In our applications, they are retrained on the data every day, so there is ample opportunity to examine which variation is doing better. Besides the purely empirical evaluation, one may have knowledge of the overall performance of the fab. For example, just looking at the trend in mean speed over several weeks can suggest whether the fab performance is stable or not.

Algorithm 2 is a overview of a procedure for sampling, learning and evaluating the models induced from the current sample of wafer data.

5 Optimizing Predictions

The overall mission is the early identification of wafers or lots that will be unacceptably fast or slow, and the implementation of effective countermeasures. The engineering staff recognizes an acceptable range of speeds for each

Algorithm 2 Model Learning

- I. For all learning methods including trees:
  1. Collect sample S1 of wafers with known completed measurements
  2. Collect independent sample. S2, for testing.
- II. For boosted decision trees and other multiple-sample leaning methods:
  1. Randomly re-sample from sample S1 and create S3
  2. Learn a prediction model from S3
  3. Repeat steps 3 and 4, k times
  4. Average the results for all k trees For new wafer prediction, average all k predictions.
- III Customize boosted trees
  1. Determine best sample period for creating S1 and S2. For example, 90 days of wafer production.
  2. In step II-1, determine best random re-sample size. For example, randomly sample 100
  3. In step II-1, overweight most erroneously predicted wafers during re-sampling
- IV. Multiple models of different types (e.g. boosted trees and linear models
  1. Average predictions of forests and linear models on S2 sample.

product. If our predictions were completely accurate, we could simply report and act on all wafers predicted outside of that acceptable range. We can see in Table 1 that predictions are far from completely accurate using data collected prior to step 7, which is the last opportunity to implement downstream corrective processing.

Analogous to predictive sales applications where lift is plotted, i.e. plotting the gain from ranked selection of prospects versus random selection, these predictions can be ordered and ranked. Wafers in the extreme tails of the prediction distribution are usually much more likely to be out of range, and of interest in our application. The test data are used to estimate expected deviations from the mean. Given a specific threshold, for example all wafers predicted above t, overall deviation of the true values from the mean are measured. Additionally measured are deviation in the correct direction and deviation in the negative direction. A measure of accuracy is provided, where a prediction is scored as correct when it is in the same direction as the true answer, i.e. above or below the mean. The results for selected threshold, t, should surpass a minimum degree of accuracy for both direction and deviation. An effective threshold must provide highly accurate predictions and identify wafers with meaningfully large absolute deviations from the desired range.

The selected wafers will undergo corrective processing to increase or decrease their speed. In general we consider corrective processing strategies designed to adjust wafers slightly, to move wafers from outside a desired range into the range, rather than trying to move the wafers to the center of the

range. Assuming a modest increase in speed for a predicted slow wafer, a mistake in prediction could make it too fast and actually degrade the wafer yield, a costly expense. However, if the increase in speed maintains the wafer's chips within the upper bound, then the expense is minor. Thus, a more detailed analysis of thresholds for prediction is warranted to find an interval where prediction is most accurate. Algorithms 3 and 4 are procedures for optimizing the thresholds for detecting high or low values based on predictions of the model described in the Section 4.

Algorithm 3 Detecting High Values

1. Build statistical prediction model for sample of completed wafers or lots.
2. Collect a separate test sample from either earlier or later completed wafers
3. Using the model in (1) and test sample in (2), predict the target measurement for each wafer
4. For each wafer, compute the prediction error by comparing to the true target measurement.
5. For the subset X of wafers above threshold x, compute mse (mean square error) [or mad (mean absolute deviation)]
6. Compute good mse (or mad) for wafers in (5) that are above the mean of the sample.
7. Compute bad mse (or mad) for wafers in (5) that are below the mean of the sample.
8. Compute an accuracy rate:
   (number of predicted wafers actually above mean) / (number of wafers predicted above sample test mean)
9. Using these accuracy estimates, engineering staff selects high threshold for decision based on expected costs and yields.

Although we focus on the early detection and correction of aberrant wafers, other applications of our system require early detection with high accuracy of "normal" wafers, i.e. not fast, not slow. For example, some machine build designs can only use chips with relatively tight power performance specifications and customized wafer back-end processing. Any chips tailored in the back end for that design, not ultimately meeting those tight specifications, may be unusable for another build. In such a case, improving the likelihood that chips tailored for that design will meet those tight specifications can reduce yield loss.

The task of early detection of normal wafers is not merely a trivial complement to the prediction with high accuracy of aberrant wafers: The absence of a prediction of aberrant wafers does not imply a prediction of normal. While the models we have developed for detecting aberrant wafers have high accuracy, their recall is limited and the applications exploiting those models are relatively forgiving of false negatives. Thus the early detection of normal wafers is a more difficult and complex problem from detecting fast or slow alone.

Algorithm 4 Detecting low values

1. Build statistical prediction model for sample of completed wafers or lots.
2. Collect a separate test sample from either earlier or later completed wafers
3. Using the model in (1) and test sample in (2), predict the target measurement for each wafer
4. For each wafer, compute the prediction error by comparing to the true target measurement.
5. For the subset X of wafers below threshold x, compute mse (mean square error) [or mad (mean absolute deviation)]
6. Compute good mse (or mad) for wafers in (5) that are below the mean of the sample.
7. Compute bad mse (or mad) for wafers in (5) that are above the mean of the sample. .
8. Compute an accuracy rate:
   (number of predicted wafers actually below mean) / (number of wafers predicted below sample test mean)
9. Using these accuracy estimates, engineering staff selects low threshold for decision based on expected costs and yields.

One approach is to find an interval for an ordered set of wafer speed estimates, where the true normal occurrence rate is very high. Figure 5 describes a procedure for finding an interval for normal wafers. In the absence of this application, wafers would be chosen randomly for back end customization, and a base fraction of chips will fail to meet final specs. Thus for this application, we measure success in terms of the reduction of the number of customized chips falling outside the desired PSRO window.

Figure 4 summarizes characteristics of the entire test wafer population and wafers from three intervals selected by Algorithm 5.

Choosing wafers randomly leads to a fallout of 17% at final test. Prediction intervals 1, 2 and 3, respectively are intervals that include roughly 75, 50 and 15% of the total test population, and are optimal in terms of the numbers of wafers falling outside the desired PSRO window. The fraction of selected wafers falling outside the window can be reduced to as little as 2.7%. The *reduction in loss* shows the reduction in the number of failing chips in each interval from the default random selection of chips, as a fraction of the number of chips failing with random selection, which is as high as 84% for prediction interval 3. We see a clear tradeoff between fraction of the population selected for customization and the likelihood of failing to meet final specs.

In addition we show the accuracy, recall, and r2 for the entire sample population and the three prediction intervals. We note a striking variation in the R^2 across the different prediction intervals. These prediction intervals are defined, as above, in terms of an optimal reduction in wafers falling outside the desired window. In the course of these experiments optimal smaller prediction intervals were not always strict subsets of optimal larger prediction intervals.

Algorithm 5 Detecting normal wafers

1. Build statistical prediction model for sample of completed wafers or lots.
2. Collect a separate test sample from either earlier or later completed wafers
3. Using the model in (1) and test sample in (2), predict the target measurement for each wafer
4. For each wafer, compute the prediction error by comparing to the true target measurement.
5. For the subset X of wafers below threshold x, compute mse (mean square error) [or mad (mean absolute deviation)]
6. Specify a normal range (x to y), i.e. an lower and upper bound on normal wafers
7. Examine an interval of wafer predictions on the test sample. Compute an accuracy ratio:
   (number of true normal wafers within the interval)/ (number of predicted wafers within the interval
8. Examine all intervals where each upper or lower bound is considered in increments of j (example, normal range is 10 to 11 and increments are .1)
9. Choose the best accuracy such that a minimum of k wafers are covered.

In some cases, optimal smaller prediction intervals were disjoint from larger optimal prediction intervals. The large variation in R^2 reported here across the three prediction intervals reflects primarily variation in the effectiveness of the train mean as an estimate of the wafers int he prediction interval.

We anticipate the use of our normal-finding algorithms for the manufacture of relatively low volume products. Thus the rate of yield loss can be cut dramatically, by relatively modest (relative to required product volumes) reductions in the population of candidate customization wafers.

	Choosing Wafers Randomly	Choosing Wafers in Prediction Interval 1	Choosing Wafers in Prediction Interval 2	Choosing Wafers in Prediction Interval 3
Fraction of Test Wafers Included in Prediction Interval	1.00	0.75	0.49	0.15
Fraction of Wafers in Chosen Interval Falling Outside Desired PSRO Window	0.17	0.081	0.039	0.027
Reduction in Loss (Fractional, From 16.7%)	0.00	0.52	0.77	0.84
Accuracy (Fraction of Wafers in Prediction Interval Actually in Desired Window)	0.83	0.92	0.96	0.97
Recall (Fraction of Wafers in Desired Window in Prediction Interval)	1.00	0.83	0.57	0.18
r^2 for Wafers in Prediction Interval	0.59	0.35	0.07	0.33

Fig. 4 Sample results for normal wafers

	All Test Wafers	Sample 1	Sample 1 Complement	Sample 2	Sample 2 Complement	Mystery Wafers
Sample Size	5388	174	5214	51	5337	77
Number Actually Slow	2620	147	2473	43	2577	70
Accuracy (Fraction of Number in Sample Actually Slow)	0.49	0.84	0.47	0.84	0.48	0.94
Average Deviation from Train Mean	0.04	0.66	0.02	0.71	0.04	1.19
Recall (Faction of Number Slow in Sample	1.00	0.056	0.94	0.016	0.98	n/a
r^2	0.06	0.46	0.03	0.55	0.05	n/a

a)

	All Test Wafers	Sample 1	Sample 1 Complement	Sample 2	Sample 2 Complement
Average	12.24	12.86	12.22	12.91	12.24
Variance	0.35	0.33	0.34	0.32	0.35
T Stat		14.2		8.4	
P(T<=t) one tail		$< 10^{-6}$		$< 10^{-6}$	

b)

Fig. 5 Results from retrospective study

6 Results

The concepts presented here have been implemented in a fully automated system that predicts the final test PSRO proxy for final chip microprocessor speed. Data for training, testing, and prediction are extracted from the fab's data warehouse, which is updated within minutes of any newly completed measurement for a wafer. In our current implementation, samples, decision models, and estimates are updated once a day.

A simple evaluation of predictive model performance on test data sets is an inadequate characterization of overall system performance. Rather, below, we describe two comprehensive evaluations. Retrospectively using complete historical data, we performed a complete simulation of daily resampling, model building and testing. In a smaller, more expensive prospective study, we performed true real-world testing in a manner similar to evaluating the efficacy of a drug versus a placebo. In both studies, the application is for remedial action to a wafer prior to a landmark step S.

Retrospective Study: In Figure 2, the decision to hold a wafer and commit to corrective downstream processing must be made by the landmark step 7 (LS7). Thus the system will compute predictions using only those measurements collected prior to that landmark. Using data from all the wafers that were completed through final test during a two month period, we examined the daily estimation process for each wafer just prior to LS7. Twenty-four lots

By Wafer	Including all Wafers		Excluding Fastest Lot	
	BAU Processing	Predicted Fast	BAU Processing	Predicted Fast
Sample Size	290	35	290	29
Number Actually Fast	151	29	151	23
Accuracy (Fraction Fast in Sample)	0.52	0.83	0.52	0.79
Average	10.570	10.312	10.570	10.409
Variance	0.298	0.151	0.298	0.079
T Stat	3.524		2.622	
P(T<=t) one tail	0.00045		0.00572	

a)

By Lot	Including all Wafers		Excluding Fastest Lot	
	BAU Processing	Predicted Fast	BAU Processing	Predicted Fast
Sample Size	59	5	59	4
Number Actually Fast	28	4	28	3
Accuracy (Fraction Fast in Sample)	0.47	0.80	0.47	0.75
Average	10.540	10.335	10.540	10.457
Variance	0.208	0.109	0.208	0.045
T Stat	1.288		0.676	
P(T<=t) one tail	0.127		0.264	

b)

Fig. 6 Results from real time study

of approximately 25 wafers were completed during this time period. Of those 24 lots, 3 lots were predicted to be substantially fast and 3 substantially slow. All 6 of the identified lots had average speed offsets in the predicted direction which is evidence of operationally high accuracy, especially given the potential impact of downstream processes of uncertain impact and stability.

Figure 5 is a summary of statistical results from a single day's model of the line. Two independent test set samples were examined using different thresholds as described above. We see that roughly 90% of the wafers predicted to be slow in both test samples were actually slower than average, a highly operationally accurate result. Figure 5 also shows that the mean of the wafers selected in each sample is significantly different from the mean of the wafers not selected. We also note the anticipated tradeoff between the number of wafers exceeding a predicted speed threshold and the accuracy of those predictions. This model was then applied to (mystery) wafers outside of the train and test sets. The 90% accuracy of the predictions on the mystery wafers was similar to that on the test wafers. Deviations from the mean were larger for the mystery set than the test set. The extent of deviation from the mean is a critical factor in determining whether corrective processing is warranted. In this system learning and optimizing methods are tailored to identify wafers with extreme deviations, however no explicit controls are introduced to assure any minimum absolute deviations.

Real Time Study: In a second, prospective pilot study, we intervened directly in the production process to correct expected fast wafers. Many of the control measurements in place are used as part of the fab's run to run control system. This pilot study explores the advantages of adding an additional level of control to the preexisting control systems.

In the study design, a quota of 5 partial lots, about 70 wafers, was allocated for intervention. We would notify an engineer to hold a predicted fast lot prior to LS7, and then the lot would be split. Half of the lot would continue in the regular fashion, i.e. with business as usual processing, and half would be processed in a fashion to introduce a small speed reduction.

Given that the predictions are expected to be effective on an statistical basis over large sets of wafers, this strategy allows for correction of fast wafers while doing relatively little harm to those slow or normal wafers incorrectly predicted as fast.

Although predictions were made by wafer, the pilot study was conducted by selecting particular lots for split lot processing. Lot wise predictions were made by averaging across wafers in each lot. Once a lot was selected for split lot processing wafers were selected for normal processing or corrective processing without regard to the wafer level speed prediction. Figure 6 summarizes the results of the Real Time Pilot.

On a by wafer basis, the predictions were roughly 80% accurate in identifying wafers that were fast. The mean values of wafers predicted fast were significantly faster that the remainder of the population, and by several tenths of a picosecond, an operationally significant amount. One of the 5 lots predicted fast was much faster than the other 4 lots, and Figure 6 includes results from the Pilot when that one particular lot is excluded from the analyses. There is a modest reduction in accuracy, and the difference in the mean values is reduced, but the T statistic for difference in means remains statistically significant at the 0.005 level.

A purely by-wafer analysis of the results is incomplete given the processing relationships shared by wafers within a lot. Figure 6 also reports the results by lot. Here the statistical significance of the test for difference in means is reduced dramatically due to the reduction in sample size, though the accuracy in identifying lots remains high, even when the extremely fast lots are removed from the analysis.

From a macro-decision perspective, as a result of the predictions and corrective processing, the extremely fast lot is corrected into a normal range, while the other 4 lots remain in the normal range when modest corrections are applied.

7 Discussion

We have described a fully functioning system that predicts mean wafer speed prior to final testing. Speed serves as a proxy for estimating overall wafer health during manufacture. The advantages of accurate prediction are manifold

including wafer correction and prioritization for different customers. Although the current implementation does not accurately predict future performance of all wafers, we have shown promising results for identifying some outliers.

Clearly, this is a difficult prediction problem. The measurements are sampled in small quantities and the utility of these measurements is uncertain, especially when applied to individual wafer estimation. Processes may evolve over time as described above, and manufacturing tool performance may evolve over time reflecting a dynamic mix of products in a multi-purpose fab such as IBM's 300mm line.

We examined in detail an application with opportunities for corrective action. Here the prediction must be made prior to a landmark operation so that corrective action is feasible. Prediction accuracy is limited by the manufacturing steps that occur downstream of landmark, to which the learning system is oblivious. Any hope of making highly accurate predictions with such a data set relies strongly on the stability of the processes occurring downstream from that last data collection step and or an assumption that the downstream operations have relatively little influence on speed.

From a modeling perspective, the nonstationary nature of the manufacturing processes along with overwhelming missing data makes for a complex problem. Despite all these complications, we have shown that estimation significantly beyond chance is feasible and in some cases reasonable predictions can be made at the wafer and lot level.

There are many opportunities for future improvements to this system. We anticipate improvements in accuracy with applications to increasingly stable manufacturing environments.

An improvement in quality of measurement, or an increase in the sampling rate of wafer measurements, will also likely lead to better results. Thousands of different measurements are made in the course of the manufacturing line, many at low frequency. If the most predictive measurements are identified, an increase in their measurement sampling frequency should be beneficial for prediction.

Here we considered on-wafer physical measurements, on-wafer chemical measurements, on-wafer metrology, and in-line kerf electrical measurements as potential predictors. Other classes of measurements characterizing the fab and manufacturing process, included so-called process trace data and data characterizing fab facilities, environment, and consumables should be considered as additional powerful predictors.

Data input for learning, testing and prediction in these implementations was aggregated by wafer. Many unit manufacturing processes exhibit significant across wafer non-uniformities. In a related but different problem of monitoring and predicting yield, it was reported that some semiconductor yield models show improvements with spatially resolved estimates, e.g. by individual chip or by region (Krueger et al., 2011). Yield prediction has been a heavily studied problem in semiconductor literature (Stapper, 1989), (Kumar et al., 2006), (Yeh et al., 2007), (Hu, 2009), (Chien et al., 2013), (Li et al., 2006), (Holden and Serearuno, 2005), (Lee et al., 2000), (Su and Tai-Lin, 2003), where

defect data is the primary driver in estimating yield, usually of memory chips. In our case however, we had only electrical and physical measurements taken early on in the manufacturing process to estimate microprocessor speed. Moreover, we described an online system which runs daily in the fab and adapts to changing dynamics as opposed to a static yield model. There has also been work on throughput prediction (Chien et al., 2012) using machine learning models so as to reduce fab cycle times, however this problem is orthogonal to both yield prediction and our work.

We expect, a priori, that some manufacturing line measurements should reflect well known physical relationships, for instance known relationships between transistor gate dimensions and transistor speed should be reflected in relationships between manufacturing gate metrology and microprocessor speed. The empirical observations manufacturing results reveal these relationships to varying degrees. A variety of explanations for weak or evolving empirically observed relationships are possible, suggesting explicit use of known physical models in such systems.

Overall, there are many possibilities for improving fab operations by examining in detail the reasons for poor predictions and modifying fab operations. In our case, we assume no change to fab operations and experiment within the existing operations framework.

From a machine learning perspective, models could be incrementally updated as new measurements are recorded. Specialized algorithms would be needed for incremental learning because not only are new wafers incrementally observed, but also older wafers have revised information. Our current algorithms make a fresh start every day with the latest sample and complete batch learning. Those procedures are adequate when the system is not stressed by time and data constraints. Both knowledge from chip-making and possibly improved machine learning techniques could produce a new class of methods for estimating chip performance.

References

Apte, C., Weiss, S., and Grout, G. (1993). Predicting defects in disk drive manufacturing: A case study in high-dimensional classification. In *IEEE CAIA (93)*, pages 212–218.

Bao, X., Bergman, L., and Thompson, R. (2009). Stacking recommendation engines with additional meta-features. In *RecSys '09: Proceedings of the third ACM conference on Recommender systems*, pages 109–116, New York, NY, USA. ACM.

Bell, R., Bennett, J., Koren, Y., and Volinsky, C. (2009). The million dollar programming prize. *IEEE Spectrum*, pages 28–33.

Chien, C., Chang, K., and Wang, W. (2013). An empirical study of design-of-experiment data mining for yield-loss diagnosis for semiconductor manufacturing. *Journal of Intelligent Manufacturing*.

Chien, C., Hsu, C., and Hsiao, C. (2012). Manufacturing intelligence to forecast and reduce semiconductor cycle time. *Journal of Intelligent Manufacturing*, 23:2281–2294.

Dzeroski, S. and Ženko, B. (2004). Is combining classifiers with stacking better than selecting the best one? *Machine Learning*, 54(3):255–273.

Fountain, T., Dietterich, T., and Sudyka, B. (2000). Mining ic test data to optimize vlsi testing. In *Proceedings of the sixth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 18–25.

Goodwin, R., Miller, R., Tuv, E., Borisov, A., Janakiram, M., and Louchheim, S. (2004). Advancements and applications of statistical learning/data mining in semiconductor manufacturing. *Intel Technology Journal*, 8(4):325–336.

Harding, J., Shahbaz, M., Srinivas, and Kusiak, A. (2006). Data mining in manufacturing: A review. *Manufacturing Science and Engineering*, 128(4):969–976.

He, J. and Zhu, Y. (2012). Hierarchical multi-task learning with application to wafer quality prediction. In *IEEE 12th International Conference on Data Mining (ICDM)*.

Holden, T. and Serearuno, M. (2005). A hybrid artificial intelligence approach for improving yield in precious stone manufacturing. *Journal of Intelligent Manufacturing*, 16:21–38.

Hu, H. (2009). Supervised learning models in sort yield modeling. In *Adv. Semiconduct. Manuf. Conf.*, pages 133–136.

Irani, K. B., Cheng, J., Fayyad, U. M., and Qian, Z. (1993). Applying machine learning to semiconductor manufacturing. *IEEE Expert: Intelligent Systems and Their Applications*, 8(1):41–47.

Khan, A., Moyne, J., and Tilbury, D. (2007). An approach for factory-wide control utilizing virtual metrology. *IEEE Transactions on Semiconductor Manufacturing*, 20:364–375.

Krueger, D., Montgomery, D., and Mastrangelo, C. (2011). Application of generalized linear models to predict semiconductor yield using defect metrology data. *IEEE Transactions on Semiconductor Manufacturing*, 24:44–58.

Kumar, N., Kennedy, K., Gildersleeve, K., Abelson, R., Mastrangelo, C., and Montgomery, D. (2006). A review of yield modeling techniques for semiconductor manufacturing. *Int. J. Prod. Res.*, 44:5019–5036.

Lee, D. Y., Cho, H. S., and Cho, D. Y. (2000). A neural network model to determine the plate width set-up value in a hot plate mill. *Journal of Intelligent Manufacturing*, 11:547–557.

Li, T.-S., Huang, C.-L., and Wu, Z.-Y. (2006). Data mining using genetic programming for construction of a semiconductor manufacturing yield rate prediction system. *Journal of Intelligent Manufacturing*, 17:355–361.

Liu, Y., Kalagnanam, J., and Johnsen, O. (2009). Learning dynamic temporal graphs for oil-production equipment monitoring system. In *KDD*, pages 1225–1234, New York, NY, USA. ACM.

Melzner, H. (2002). Statistical modeling and analysis of wafer test fail counts. In *Advanced Semiconductor Manufacturing 2002 IEEE/SEMI Conference*

and Workshop, pages 266–271.

Schapire, R. (1990). The strength of weak learnability. *Mach. Learn.*, 5:197–227.

Stapper, C. (1989). Fact and fictions in yield modeling. *Microelectronics Journal*, 8:103–109.

Su, C.-T. and Tai-Lin (2003). Chiang optimizing the ic wire bonding process using a neural networks/genetic algorithms. *Journal of Intelligent Manufacturing*, 14:229–238.

Vapnik, V. (1998). *Statistical Learning Theory*. Wiley & Sons.

Weber, C. (2004). Yield learning and the sources of profitability in semiconductor manufacturing and process development. *IEEE Transactions on Semiconductor Manufacturing*, 17(4):590–596.

Weiss, S., Baseman, R., Tipu, F., and et al. (2010). Rule-based data mining for yield improvement in semiconductor manufacturing. *Applied Intelligence*, 3:318–329.

Yeh, C., Chen, C., and Chen, K. (2007). Validation and evaluation for defect-kill-rate and yield estimation models in semiconductor manufacturing. *Int. J. Prod. Res.*, 45:829–844.

Zhu, Y. and Baseman, R. (2012). Virtual metrology and run-to-run control in semiconductor manufacturing. In *18th ISSAT International Conference on Reliability and Quality in Design*.