Improving Quality Control by Early Prediction of Manufacturing Outcomes

Sholom M. Weiss
sholom@us.ibm.com
Mathematical Sciences Dept.
IBM T.J. Watson
1101 Kitchawan Road
Yorktown Heights, USA

Amit Dhurandhar
adhuran@us.ibm.com
Mathematical Sciences Dept.
IBM T.J. Watson
1101 Kitchawan Road
Yorktown Heights, USA

Robert J. Baseman
baseman@us.ibm.com
Mathematical Sciences Dept.
IBM T.J. Watson
1101 Kitchawan Road
Yorktown Heights, USA

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.

Categories and Subject Descriptors

H.2.8 [**Database Management**]: Data Mining

General Terms

Algorithms

Keywords

manufacturing, quality control, prediction

KDD'13, August 11–14, 2013, Chicago, Illinois, USA.

1. INTRODUCTION

Modern-day instrumented manufacturing is a complex process, sometimes taking weeks to even months to produce the final product. Starting from the initial crude state, the final product is produced by the application of hundreds of steps and tools. Typical examples of where we encounter such heavily instrumented operations is the semiconductor industry, the pharmaceutical industry, the (processed) food industry. Given the complexity of these processes and the time to manufacture, 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 [6], [7], [13], [17], [18]. Among the primary roles of data mining in these domains are quality control and the detection of anomalies. When something goes wrong, such as a significant reduction in final product quality, the data are pulled and examined to find probable causes. Many of these industries are extremely sensitive to such mishaps. Even a meager (few percent) drop in quality could cost a corporation millions to even a few billions of dollars. Conversely, a few percent increase in quality can be highly lucrative. From a data collection perspective, tens or even hundreds of thousands of measurements are taken and recorded to monitor results at different stages of production. Since, the objective is mostly to monitor quality of production, 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 product quality. In particular, we focus on the semiconductor industry, where we predict the final microprocessor performance. The challenges we face and the methods we employ are largely applicable to other such domains mentioned before.

Each wafer, which is a collection of 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 the great interest in mining data prior to final testing [9], [1], [5]. Moreover, if such an endeavor were to be successful, it would greatly enhance manufacturing productivity.

While many alternative testing measurements are reasonable to measure the health of a wafer, in our initial applications, we designate a proxy for microprocessor chip speed as the predicted outcome. Thus during manufacture, the

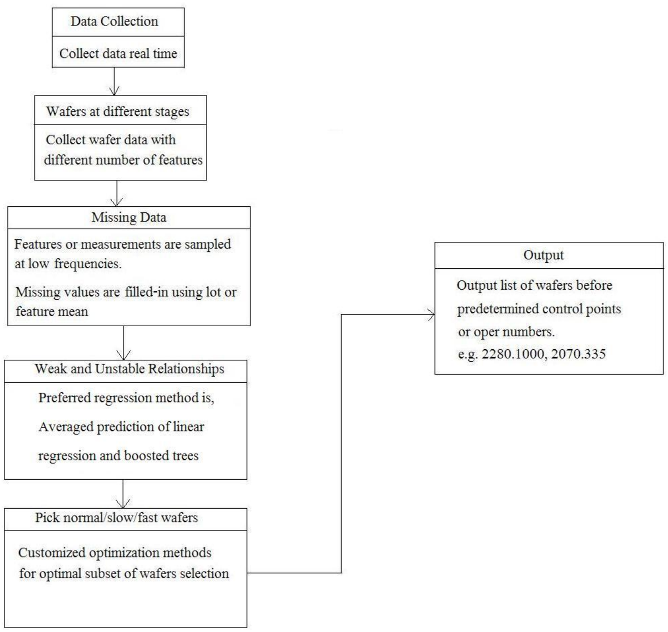

Figure 1: Overview of the applied methodology

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.
- Queue wafers for key customers.
- Queue wafers based on expected performance and current demand.

Predicting final performance based on incomplete measurements is a difficult task. It implies having accurate and highly predictive measurements. The benefits 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: with time-varying populations to inherent instabilities of massively missing data, to only a few measurements being known before critical steps. To address these difficulties, knowledge-based methods for filling in missing values are developed, specialized sampling techniques are employed, combined learning methods such as linear with boosted trees are invoked, and customized schemes to adjust and optimize the predictions obtained from the learning methods are deployed. An overview of the applied methodology is shown in Figure 1.

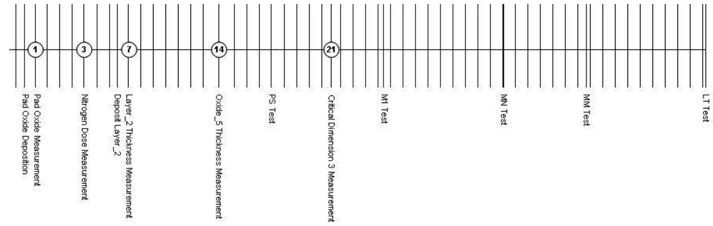

Figure 2: Stages of wafer/chip manufacturing. A wafer moves from left to right.

In real microprocessor production experiments, our automated models selected 125 predicted fast wafers (5 lots) in real-time. Wafers from these selected lots were split for post prediction processing to allow corrective processing and assessment of prediction accuracy. These selected wafers were significantly faster than average, as predicted. Of the 5 lots, one lot was fast enough that downstream corrective processing restored nominally unacceptable wafers to normal operation.

2. APPLICATION 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 an electrical test (PSRO) serving as a (inverse) proxy for microprocessor speed. The higher the PSRO the slower the wafer and vice-versa. This test is conducted on all wafers as one of the last set of electrical tests (LT) conducted on test structures built in the wafer kerfs. In an ideal implementation, we would update a prediction of PSRO measured at LT for each wafer after each processing and measurement step.

However, in these initial implementations, we established a limited number of landmarks in the production process at which to provide and update predictions. 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.

3. PROCEDURES FOR DATA PREPARATION

Our application has the following input and output characteristics:

- Input: Control measurements on a wafer such as physical measurements, lithographic metrology, and electrical measurements.
- Output: Performance indicators such as speed or power consumption measurements.

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 nominally salable products. This imposes a clear requirement that the predictions be made with high precision. Thus, depending on the expected precision, 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 Data Preprocessing

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). The complete data for wafers that have completed testing can be readily retrieved from a database.

Field	Fraction Missing
Lot ID	0%
Wafer ID	0%
Oxide_1 Thickness	94%
Critical Dimension 1	87%
Nitrogen Dose	94%
Polysilicon_Thickness	99%
Critical Dimension 2	84%
Oxide_2 Thickness	98%
Layer_2 Thickness	83%
Layer_4 Thickness	83%
eSiGe Proximity	96%
Insulator_2 Thickness	83%
Insulator_3 Thickness	83%
Oxide_3 Thickness	75%
Oxide_4 Thickness	83%
Nitride_1 Thickness	75%
Oxide_5 Thickness	95%
Oxide_6 Thickness	92%
Nitride_2 Thickness	91%
Nitride_3 Thickness	98%
Overlay_1 X	86%
Overlay_1 Y	86%
Critical Dimension 2	95%
Overlay_2 X	85%
Overlay_2 Y	85%
Critical Dimension 3	90%
Insulator_4 Thickness	45%
Layer_5 Thickness	84%
Layer_6 Thickness	84%
Wafer Median PSRO	0%

Figure 3: Input data characteristics

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 highly censored. 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, or an assumption that the downstream operations have relatively little influence on speed. This results in a standard data presentation with one practical deficiency: Most of the data items are missing. Figure 3 presents fraction of the missing data for each of the measurements from a sample of 6435 completed wafers. Approximately 90% of the nominally anticipated measurements are missing. The missing values are not consistent for a select set of measurements. Instead, the measurements are randomly sampled, not corresponding to any particular requirement for the feature. Thus the actual recorded data will vary in the number of missing values from wafer to wafer.

The missing values reflect both the measurement sampling policies as well as the censored nature of the data. 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. That has traditionally been the main goal in sampling the measurements. 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? Modifying the sampling procedures to full data collection, at least for some key measurements, is a potential long-term strategy. However, for immediate and practical action, the current data samples must be used as is. A related issue, that we do not address further here, is whether the particular physical or chemical measurement designed for optimal quality control of a given unit process is an optimal, or even adequate, measurement for the purposes of prediction.

During manufacturing, wafers in a lot are generally processed and measured together as a group, explicitly so in batch processing tools, implicitly so in single wafer tools, undergoing the same process essentially 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. 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. 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 for reasons mentioned earlier.

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 to domain specific information. 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. Under the assumption that 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 nonstationary 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 time-frame 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.

Figure 4 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 mod-

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
Linear	-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. SVM stands for support vector machines [16]. HMML stands for a hidden markov model based method with lasso regression in every state [12]. BTM stands for best of the time series methods using SPSS expert modeler.

eling 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 Figure 4 is used for our internal estimates. Actual decisions are made about selecting wafers for revision, and the consequences of those decisions are the ultimate evaluation of predictive performance.

All aspects of this automatic machine learning application are influenced by the requirement to deal effectively with a complex manufacturing environment evolving through periods of relative stability and rapid change. The nature of sampling of data and evaluation is essential to any predictive analysis. We have seen how these dynamics influence our sampling and evaluation techniques. Next, we examine our approach to learning. We see that once again, our approaches and techniques to learn from training data are influenced strongly by the need to operate in both periods of relative stability and rapid change.

4. METHODS FOR LEARNING

From a machine learning perspective, the objective is to predict the eventual outcome of product testing, PSRO measured at LT. 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 thresholds can be specified. Our early experiments 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 Figure 4, 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, two methods were combined and used for modeling:

- linear regression
- forests (boosted trees) [14]

The results of testing several learning methods are shown in table 1. 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. 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 [2], [4], [3]. 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.

Figure 5 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 product. If our predictions were completely accurate, we could

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 (mean absolute deviation) for the difference in true value T(j) and predicted wafer target measurement P(j).

Figure 4: Model Evaluation

- 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 1 and 2, 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 wafers.
  3. In step II-1, overweight most erroneously predicted wafers during resampling
- IV. Multiple models of different types (e.g. boosted trees and linear models)
  1. Average predictions of forests and linear models on S2 sample.

Figure 5: Model Learning

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, 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 use 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 put 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. In figure 6, procedures for optimizing the thresholds for detecting high or low values based on predictions of the model described in the Section 4.

Although we nominally focus on the early detection and correction of aberrant wafers, other applications of our system require early detection with high accuracy of "normal"

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/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 above/below the mean of the sample.
7. Compute an accuracy rate:
   $\frac{\text{number of predicted wafers actually above/below mean}}{\text{number of wafers predicted above/below sample test mean}}$
8. Using these accuracy estimates, engineering staff selects high/low threshold for decision based on expected costs and yields.

Figure 6: Detecting High/Low Values

wafers, i.e. not fast, not slow. For example, some machine 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 precision, 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.

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 7 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 failing to meet the spec.

Table 2 summarizes characteristics of wafers from several intervals selected by the method of Figure 7. The *reduction in loss* shows the reduction in the number of failing chips in a given interval from the default random selection of chips, as a fraction of the number of chips failing with random selection. For interval 1, it's 100% - (2.3% / 15.4%) = 79%. We see clearly the expected tradeoff between fraction of the population selected for customization and the likelihood of failing to meet final specs. We anticipate use of our system for normal finding applications to address 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	f_o	$\mathcal{L}_r$	f_i
Randomly	15.4%		
Interval 1	3.2%	79%	13%
Interval 2	6.3%	59%	31%
Interval 3	7.5%	51%	45%

Table 2: Sample results for normal wafers. f_o and f_i are fraction of wafers outside PSRO target range and included in the prediction interval respectively. $\mathcal{L}_r$ is reduction in loss relative to random.

6. RESULTS

The concepts presented here have been implemented in a fully automated system that predicts the LT 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 the landmark step.

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 LT during a two month period, we examined the daily estimation process for each wafer just prior to LS7. Twenty-four lots of approximately 25 wafers were completed during this time period. Of those 24 lots, 3 lots were predicted to be substantially fast and 3 substan-

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:
   $\frac{\text{number of true normal wafers within the interval}}{\text{number of predicted wafers within the interval}}$
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.

Figure 7: Detecting normal wafers

	Test 1	Test 2	Mystery Wafers
Predicted slow	174	51	77
Actually slow	150	46	70
Accuracy	86%	90%	94%
Mean PSRO > train mean	+0.78	+0.93	+1.33

Table 3: Results from retrospective study.

tially 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.

Table 3 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 sets were actually slower than average, a highly operationally accurate result. We also see the anticipated tradeoff between the number of wafers exceeding a predicted speed threshold and the accuracy of those predictions, although relatively large reductions in the numbers of wafers identified are required for relatively small improvements in accuracy. This model was then applied to (mystery) wafers outside of the train and test sets. The 94% 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 study, we intervened directly in the production process to correct nominally fast wafers. A quota of 5 lots, about 125 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.

From a macro-decision perspective, one of the 5 lots is clearly a too-fast lot and is saved and corrected, while the other 4 lots remain in the normal range when modest corrections are applied. At the micro-decision level, the accuracy of predictions in this pilot was less than in the retrospective study. 21 of the 32 wafers identified were faster than target. One likely explanation for the reduction in accuracy is the fact that during the prospective study there was on going active experimentation with downstream processes known to influence PSRO.

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.

From a modeling perspective, the nonstationary nature of the manufacturing processes along with overwhelming missing data makes for a complex analysis. 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.

It is important to note that the strategies employed here could be adapted to other manufacturing environments mentioned before, that share similar concepts like distinct manufacturing steps and recorded intermediate measurements. Products in these other domains also tend to move in groups through the manufacturing steps and hence, the ideas for filling in missing values could be easily applied. The sampling, learning and adjustment of predictions methodologies described in this paper to choose faulty products also naturally extend to these other domains. In fact, we have already explored such possibilities with the manufacture of consumer products, snack products and pharmaceuticals, with some initial promise.

There are many opportunities for future improvements in the performance of the system. We anticipate improvements in accuracy with applications to increasingly stable manufacturing environments, where a fab is dedicated to a particular product, rather than a potpourri of products as is the case with the IBM fab. Another direction that could lead to further enhancement is by improving the quality of measurements, or by increasing the sampling rate of wafer measurements. 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 yield, it was reported that some semiconductor yield models show improvements with spatially resolved estimates, e.g. by individual chip or by region [10]. Yield monitoring has been a heavily studied problem in semiconductor literature [15, 11, 19, 8], 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.

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 additional 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 constraints. Both knowledge from chip-making and possibly improved machine learning techniques could produce a new class of methods for estimating chip performance.

Acknowledgments

We would like to thank Ronald Logan, Jonathan K. Winslow and Daniel Poindexter from the IBM fab in east Fishkill, for their guidance in the development of the system based on their domain expertise. We would also like to thank Brian White for software support.

8. REFERENCES

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

[2] X. Bao, L. Bergman, and R. Thompson. 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, 2009. ACM.

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

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

[5] T. Fountain, T. Dietterich, and B. Sudyka. 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, 2000.

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

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

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

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

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

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

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

[13] H. Melzner. Statistical modeling and analysis of wafer test fail counts. In *Advanced Semiconductor Manufacturing 2002 IEEE/SEMI Conference and Workshop*, pages 266–271, 2002.

[14] R. Schapire. The strength of weak learnability. *Mach. Learn.*, 5:197–227, July 1990.

[15] C. Stapper. Fact and fictions in yield modeling. *Microelectronics Journal*, 8:103–109, May 1989.

[16] V. Vapnik. *Statistical Learning Theory*. Wiley & Sons, 1998.

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

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

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