Applications of machine learning and data mining methods for advanced equipment control and process control

Yi Fan WANG, Yuri ISHIZAKI, Kazuho SAKURAI, Luyi LI, Takuya NAKAGAWA, Sumika ARIMA
s1520513, arima@sk.tsukuba.ac.jp
University of Tsukuba
1-1-1 Tennodai, Tsukuba city, Ibaraki pref., Japan
Phone: +81 -298-535-578 Fax: +81-298-535-558

Keyword: Machine Learning (ML), Data mining (DM), discrete process, virtual metrology (VM)

I. INTRODUCTION -Semiconductor manufacturing is characterized by a sequence of sophisticated manufacturing processes, often exceeding several hundred production steps. Such processes possess both aspects of continuous and discrete operations. In this paper, we suggest applications of DM/ML techniques responding to two issues of AEC/APC; Fault detection and classification (FDC) of a discrete process, and highly accurate virtual metrology.

1) FDC of a discrete process -Many literatures have aimed at adaptive FD modeling and prediction of continuous process machines mainly. On the other hand, there is few progress of FD modeling for the discrete process such as a final test process, though the final test process is sometimes a bottleneck of a supply chain of semiconductor products.

2) Highly accurate virtual metrology -There is no doubt that an accurate virtual metrology improves the productivity of semiconductor manufacturing because the quality test processes occupy almost a half of a whole manufacturing process. The test is also increased in leading-edge process. The problem is the accuracy of a VM model is limited because most of FD/VM models are based on a multivariate statistical analysis which assumes the data variance along a Normal distribution. The practical variance under a product-mix and step-mix manufacturing conditions is sometimes out of the assumption. Therefore, recent literatures applied ML for accurate VM modeling.

3) Literature of FD and VM -Recently, basic or extended SVM method is applied for semiconductor manufacturing and is confirmed accurate than Neural Networks (NN) (Lee & Kim, 2015) though NN have been applied more. Appendix will give more details. To response the issues above, we will introduce applications of a machine learning tool and a data mining technique for semiconductor manufacturing.

II. DISCUSSION and III. CONCLUSION

A) Ability of ML –Support Vector Machine (SVM)

1) Basics of the method - The basic idea of SVM is to map the data into a higher dimensional space called feature space and to find the optimal hyperplane in the feature space that maximizes the margin between classes as shown in Fig.1. Support vectors are a subset of the training data used to define the boundary between the two classes. For more details, appendixes will help readers.

2) Application target -CVD process in a real mass production factory is a target of experiments. Quality is measured by nine points on a wafer after the CVD process. Quality was categorized by nine classes of 2factors (dimensions), uniformity and design conformity as shown in Table.1.

3) Data analysis and Result -Two cases are evaluated, case #1 of two classes (Table.3, Fig.2), and case #2 of the nine quality classes in 2 dimensions (Fig.3). For Case #1, basic methods of single variable(2 or 3σSPC), multivariable (Hotelling-T^2 , Linear class discrimination), and ML (kernel-SVM) are comparably evaluated. Table.2 and Fig.2 shows the accuracy of each method. For Case #2, ML (kernel-SVM) and multivariable analyses (linear and non-linear class discriminations) are evaluated.

4) Conclusion -SVM was applied to construct an accurate VM model that provided multi-class quality prediction of the product. The VM model predicted with 100% accuracy the quality of the product after a CVD process. The accuracy depends on the set of input variables (Table. 2), and the best here is a case variables of all subunits are included (Figs. 2 and 3).

B) Ability of DM –Association analysis (AA)

1) Basics of the method –AA is a data-mining method to discover cause-and-effect of events and is known as a simultaneous purchasing discovery of a diaper and the beer for instance. AA's versatility (association rule A->B) leads wide application fields.

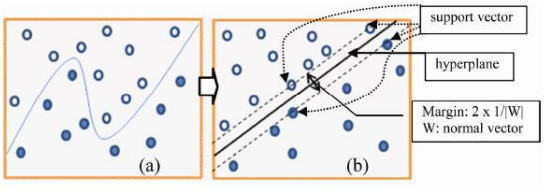
2) Application target - A typical discrete event process is a final test process after a semiconductor assembly fabrication, and its problem is minor -stoppages which occupy 25% of production period. The machine of the final test has sensors (Fig. 4), and there are 10 main types of alarms (Table.4).

3) Data analysis and Result – AA is extended to consider time-series changes (Fig.5) of the alarms. 3 month data constitute the learning data, and following 2 month data is used for testing. Five rules are selected by examining 44 rules, detected by learning data, against test data (Table. 5) as effective (Table.6).

4) Conclusion – The rules include known (consistent with operator's experience) and unknown rules, and the rule based on data supports ability of a learning organization like an example of Table.5 (right side).

Fig.1 Kernel SVM: (a) case needed non-linear discrimination and (b) mapping from original space to feature space by a kernel function:

$$\Phi : x_i \in X^p \rightarrow z_i = F(x_i) \in Z^q (p < q), \quad \langle z_i, z_j \rangle = K(x_i, x_j)$$



Note: *Solid line*: the optimal separating hyperplane $\langle w, z \rangle + b = 0$; *Dashed line*: $\langle w, z \rangle + b = 1, -1$.

Table.1 Multi-dementional quality class definition (nine classes for 2D)

		Uniformity by wafer[a]			
		a	*b*	*c*	*All (abc)*
Design-conformity[b]	*A*	46	18	6	70
	B	14	2	8	24
	C	2	0	4	6
	All (*ABC*)	62	20	18	100

[a] *a*, *b* and *c* are uniformity levels in better rank order.

[b] *A*, *B* and *C* are conformity levels in better rank order. *A*, *B* and *C* are conformity levels in better rank order.

Note: Numbers in table are probability of the sample data, colours means difference indication/actions to do).

Table.2 Equipment variables for experimentation

Parameter set #	*Average*	*SD*	*Parameter set #*	*Average*	*SD*
1	All (18)	All (18)	8	Subunit# 3(7)	Subunit#
2	All (18)	–	9	Subunit# 3(7)	–
3	–	All (18)	10	–	Subunit#
4	Subunit# 1(6)	Subunit# 1(6)	11	Subunit# 4(4)	Subunit#
5	Subunit# 1(6)	–	12	Subunit# 4(4)	–
6	–	Subunit# 1(6)	13	–	Subunit#
7	Subunit# 2(1)	Subunit# 2(1)			

Table.3 Case#1- Accuracy of MVA(T^2) and SPC(k σ)

	PCA + statics T^2 and Q	*3σ*	*2σ*
Fault detection rate [%]	67	33	67
False alarm	1	1	28

Fig.2 Case#1- VM accuracy -two classes for 1D (x={1,2,…13} is # of Eq. variables set in Table X)

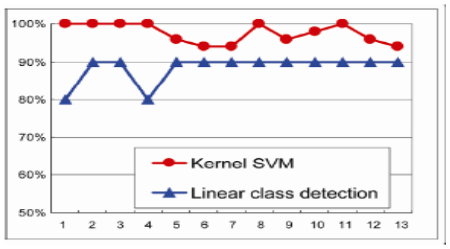


Fig. 3 Case#2 -VM accuracy -nine classes for 2D

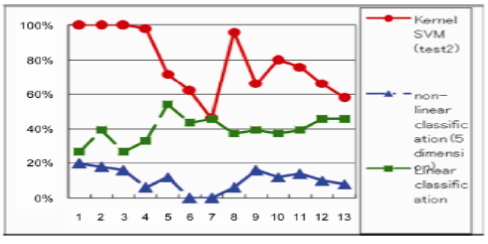


Fig.4 Structure of final test machine (top view).

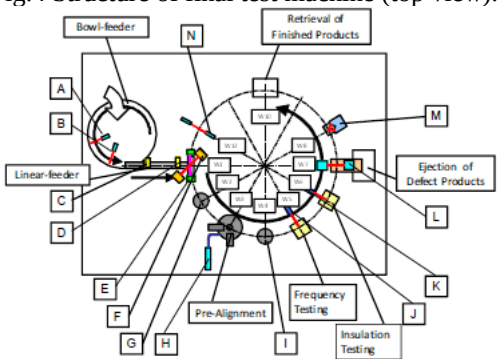


Table.4 The ten minor-stoppage types.

Name		Position	Sensor
PF-Stuck Al	Parts Feeder Stuck	Linear-feeder	D,E,F
WC Al	Work Completion	Linear-feeder	E,F
WS-F Al	Work Supply Failure	W1	F
PASM Al	Pre-Alignment Sucking Miss	W3,W4	G,H,I
WL-at-W5 Al	Work Left at W5	W5	J
WL-at-W6 Al	Work Left at W6	W6	K
SM Al	Sucking Miss	W7	F,L
FDD Al	Failure to Detect a Defect	W8	M
YF Al	Yield of Finished product	W10	-
FRF Al	Failure to Retrieve a Finished product	W12	N

Fig.5 AA extended with time-series (*k-th* window)

# of stoppages of type *i* occurred in the *k-th* window,

$$\Delta \underline{x}^T(k) = \underline{x}^T(k) - \underline{x}^T(k-1), \quad k = 2, \ldots, K_L, \qquad (1)$$

Then $\Delta x(k,i)$ can be standardized as

$$\underline{z}(k) = [z(k,1), \ldots, z(k,N)] ;$$

$$z(k,i) = \frac{\Delta x(k,i) - \mu_i}{\sigma_i} . \qquad (4)$$

$$I(k,i) = \begin{cases} -1 & if \ z(k,i) \le -\alpha \\ 0 & if \ z(k,i) \in (-\alpha, \alpha) \\ 1 & if \ z(k,i) \ge \alpha \end{cases} . \qquad (5)$$

$$M(k,i,y) = \delta_{\{I(k-1,i)=y\}} + \delta_{\{I(k,i)=y\}} . \qquad (6)$$

A typical association rule $\mathcal{R}$ would consist of the condition part expressed in terms of a set of $M(k,i,y)$'s for $i \in \mathcal{N}$ and $y \in \{-1,1\}$, and the conclusion part written as $I(k+1,r) = 1$ for some $r \in \mathcal{N}$.

Table.5 Effective ARs With $\alpha = 0.5$, $\beta = 0.02$ and $\gamma = 0.30$

No	LHS		RHS
1	$M(k,\text{WC},1) = 1$	$\Rightarrow$	$I(k+1,\text{WS-F}) = 1$
2	$M(k,\text{FRF},1) = 1$ $M(k,\text{FRF},-1) = 1$	$\Rightarrow$	$I(k+1,\text{WS-F}) = 1$
3	$M(k,\text{FRF},1) = 1$ $M(k,\text{FRF},-1) = 1$	$\Rightarrow$	$I(k+1,\text{PF-Stuck}) = 1$
4	$M(k,\text{WC},-1) = 1$ $M(k,\text{WS-F},-1) = 1$	$\Rightarrow$	$I(k+1,\text{WS-F}) = 1$
5	$M(k,\text{WL-at-W6},1) = 1$ $M(k,\text{PF-Stuck},-1) = 1$	$\Rightarrow$	$I(k+1,\text{PF-Stuck}) = 1$

PM policies:
1) 1,2,4 then Head cleaning
2) 3,5 then Feeder cleaning

Table.6 Accuracy of selected association rules

No	Data	*SUPP*	*CONF*	*LIFT*
1	Learning	0.066	0.306	1.691
	Test	0.060	0.313	1.382
2	Learning	0.030	0.385	2.128
	Test	0.048	0.364	1.608
3	Learning	0.024	0.308	1.502
	Test	0.048	0.364	2.036
4	Learning	0.030	0.333	1.844
	Test	0.036	0.333	1.474
5	Learning	0.030	0.500	2.441
	Test	0.024	0.667	3.733

Given an association rule $\mathcal{R}$, let $VAL(\mathcal{R})$ be the set of units for which $\mathcal{R}$ is VALid. Similarly, we define $COND(\mathcal{R})$ and $CONC(\mathcal{R})$ to be the set of units meeting the CONDition of $\mathcal{R}$ and that satisfying the CONClusion of $\mathcal{R}$ respectively. It should be noted that $VAL(\mathcal{R}) = COND(\mathcal{R}) \cap CONC(\mathcal{R})$. The three measures *SUPP*, *CONF* and *LIFT* are then defined as

$$SUPP(\mathcal{R}) = \frac{|VAL(\mathcal{R})|}{K_L - 3} , \qquad (7)$$

$$CONF(\mathcal{R}) = \frac{|VAL(\mathcal{R})|}{|COND(\mathcal{R})|} \qquad (8)$$

and

$$LIFT(\mathcal{R}) = \frac{|CONF(\mathcal{R})|}{|CONC(\mathcal{R})|/(K_L - 3)} , \qquad (9)$$

APPENDIXES

I-3) Literature of FD and VM

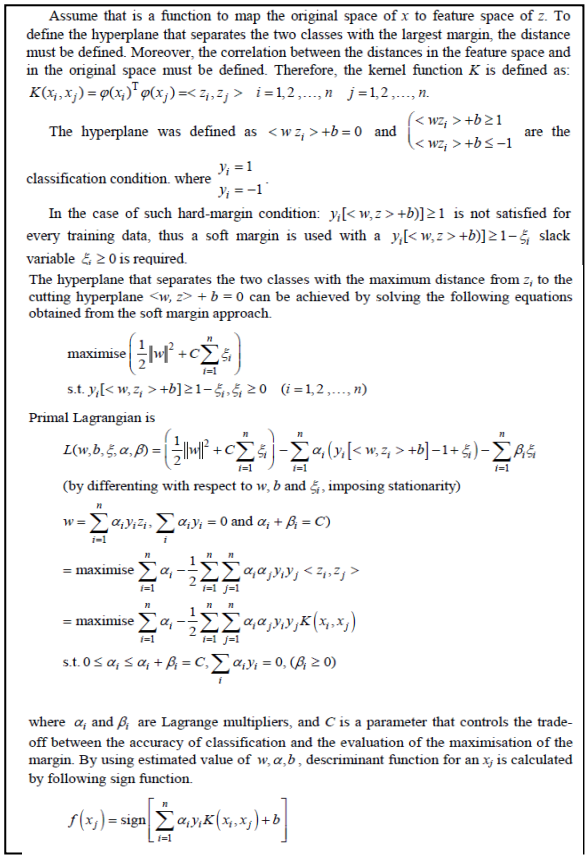
Below table summarizes methods applied in literatures of "Metrology" and "Process control" in IEEE Trans. on Semiconductor Manufacturing (2011-2015). This table will be updated more info.

Keyword	集計
–	1
BA	1
BPA	1
DMW	1
DT	1
DWT	1
EWMA	3
GMV	1
Gompertz	1
Hotelling-T2+Q	1
KDD	1
kNN	3
Logistic Regression	1
LP	1
Mean	1
MPCA	1
NN	7
No future selection	1
PCA	2
PICC	1
PLS	2
Random Projections	1
SF	1
SVM	2
SVM-RFE	1
test structures	1
(空白)	31
総計	70

A) Support Vector Machine (SVM)

SVM was originally introduced to address the Vapnik's structural risk minimization principle (1995). The basic idea of SVM is to map the data into a higher dimensional space called feature space and to find the optimal hyperplane in the feature space that maximizes the margin between classes as shown in Fig.1. Kernel functions, such as Polynomial kernel, Gaussian kernel, or Sigmoid kernel are used to map the original data to feature space. The simplest SVM deals with a two-class classification problem—in which the data is separated by a hyperplane defined by a number of support vectors. Support vectors are a subset of the training data used to define the boundary between the two classes. For more details, please refer to Cristianini and Shawe-Taylor (2006).

SVMs were originally designed for binary classifications. However, many real-world problems have more than two classes. Multi-class classification using SVM is still an on-going research issue (Hsu et al. (2002)). Most researchers view multi-class SVMs as an extension of the binary SVM classification problem as summarized by Wong and Hsu (2006). Two approaches, one-against-all and one-against-one methods, are commonly used. The one-against-all method separates each class from all others and constructs a combined classifier. The one-against-one method separates all classes pairwise and constructs a combined classifier using voting schemes. The one-against-all method is probably the earliest used implementation for multi-class classification. It constructs k SVM models where k is the number of classes. The t-th SVM is trained with all of examples in the t-th class with positive labels, and all other examples with negative labels. The decision function chooses the class of a sample that corresponds to the maximum value of k binary decision functions specified by the furthest positive hyperplane. This approach is computationally expensive because we need to solve k quadratic programming optimization problems with sample size n. The one-against-one method involves binary SVM classifier construction for all pairs of classes. This method construct k(k-1)/2 classifiers where each one is trained on data from two classes. This number is usually larger than the number of one-against-all classifiers. Although this suggests large training times, the individual problems are significantly smaller because in average, each QP problem has about 2n/k variables. The decision function assigns an instance to a class which has the largest number of votes. The voting approach is called "Max Wins" strategy.



Assume that is a function to map the original space of x to feature space of z. To define the hyperplane that separates the two classes with the largest margin, the distance must be defined. Moreover, the correlation between the distances in the feature space and in the original space must be defined. Therefore, the kernel function K is defined as: $K(x_i, x_j) = \varphi(x_i)^T \varphi(x_j) = < z_i, z_j > \quad i = 1,2,\ldots,n \quad j = 1,2,\ldots,n.$

The hyperplane was defined as $< w z_i > +b = 0$ and $\begin{cases} < w z_i > +b \geq 1 \\ < w z_i > +b \leq -1 \end{cases}$ are the classification condition. where $\begin{matrix} y_i = 1 \\ y_i = -1 \end{matrix}$.

In the case of such hard-margin condition: $y_i[< w, z > +b)] \geq 1$ is not satisfied for every training data, thus a soft margin is used with a $y_i[< w, z > +b)] \geq 1 - \xi_i$ slack variable $\xi_i \geq 0$ is required.

The hyperplane that separates the two classes with the maximum distance from z_i to the cutting hyperplane $<w, z> + b = 0$ can be achieved by solving the following equations obtained from the soft margin approach.

$$\text{maximise} \left(\frac{1}{2} \|w\|^2 + C \sum_{i=1}^{n} \xi_i \right)$$

$$\text{s.t. } y_i[< w, z_i > +b] \geq 1 - \xi_i, \xi_i \geq 0 \quad (i = 1,2,\ldots,n)$$

Primal Lagrangian is

$$L(w, b, \xi, \alpha, \beta) = \left(\frac{1}{2} \|w\|^2 + C \sum_{i=1}^{n} \xi_i \right) - \sum_{i=1}^{n} \alpha_i \left(y_i [< w, z_i > +b] - 1 + \xi_i \right) - \sum_{i=1}^{n} \beta_i \xi_i$$

(by differenting with respect to w, b and ξ_i, imposing stationarity)

$$w = \sum_{i=1}^{n} \alpha_i y_i z_i, \sum_i \alpha_i y_i = 0 \text{ and } \alpha_i + \beta_i = C)$$

$$= \text{maximise} \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j < z_i, z_j >$$

$$= \text{maximise} \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j K(x_i, x_j)$$

$$\text{s.t. } 0 \leq \alpha_i \leq \alpha_i + \beta_i = C, \sum_i \alpha_i y_i = 0, (\beta_i \geq 0)$$

where α_i and β_i are Lagrange multipliers, and C is a parameter that controls the trade-off between the accuracy of classification and the evaluation of the maximisation of the margin. By using estimated value of w, α, b, descriminant function for an x_j is calculated by following sign function.

$$f(x_j) = \text{sign} \left[\sum_{i=1}^{n} \alpha_i y_i K(x_i, x_j) + b \right]$$

Fig-A1. Hyperplane separating two classes

B) Association analysis – basic and beyond

<Basic>

The problem of how to mine association rules from a large-scale data set has been addressed by many researchers, represented by Agrawal et al.(1993) and Agrawal and Srikant (1994). "apriori algorithm" has been proposed for speedy detection of the rules.

STEP.1

Select item set of the larger support (SUPP) value than the threshold value (lower bound)

STEP.2

Create rules of larger confidence (CONF) than the threshold value (lower bound) (CONF_MIN).

(1) select one of item sets of size(# of elements in the set) more than 2.

(2) Make and evaluate rules

(a) Set an element of the selected set to right side of rule {A}->{B}. And set other elements are set in left side of the rule({A}). Calculate CONF of the rule, and the rule becomes valid when the CONF value is larger than CONF_MIN.

(b) Repeat (a) for all possible rules in which one of elements in the item set is settled to part {B}.

(c) Repeat (a) and (b) for all selected item sets.

Fig-B1. Procedure of "apriori algorithm"

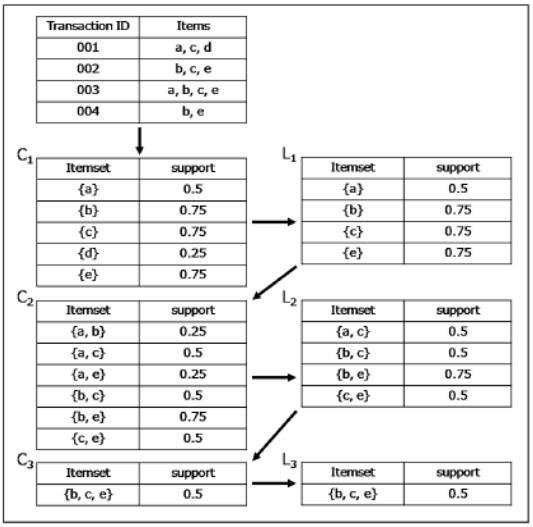


Fig-B2. Example of STEP.1

<Beyond basic>

When the association rule approach is applied to sequential data for prediction, some peculiar issues arise, as discussed in Agrawal and Srikant (1995), Lu et al.(1998), Jiang and Gruenwald (2006) and Qin and Shi (2006) to name a few. In this paper, this line of research is followed for developing preventive maintenance policies to control the minor-stoppages in semiconductor manufacturing.

Real data have been collected from a semiconductor factory. The data set consisting of K_L windows would be used as the learning data and a set of association rules would be established tentatively by following the procedure. The next K_T windows would be then used as the testing data, where a tentative association rule is chosen to be a formal rule if the accuracy of the association rule over the testing data exceeds a pre-specified level. For each of such formal rules, an action plan is devised so as to reduce the minor-stoppages by implementing the action plan whenever the condition(s) of the rule could be observed.

In practice, the learning data may be collected for 3 months, while the testing data may consist of the windows over the subsequent 2 months. The resulting selected association rules would be applied to real data for 1 month following the testing period so as to reduce minor-stoppages. This learning-testing procedure would be repeated monthly on a rolling horizon basis for updating the selected association rules.

REFERENCES

Vapnik, V.N. (1995) *The Nature of Statistical Learning Theory*. New York: Springer-Verlag.

Cristianini, N. and Shawe-Taylor, J.S. (2006) *An introduction to Support Vector Machines* (10th printing). New York: Cambridge University Press.

Hsu, C.W. and Lin, C.J. (2002) 'A comparison of methods for multi-class support vector machines', *IEEE Transactions on Neural Networks*, Vol. 13, No. 2, pp.415–425.

Wong, W. and Hsu, S. (2006) 'Application of SVM and ANN for image retrieval', *European Journal of Operations Research*, Vol. 173, pp.938–950.

Agrawal, R., Imielinski, T., and Swami, A. (1993). Mining association rules between sets of items in large databases. *Proceedings of the 1993 ACM SIGMOD International Conference on Management of Data*, pages 207–216.

Agrawal, R. and Srikant, R. (1994). Fast algoritms for mining association rules. *Proceedings of the 20th Conference on Very Large Data Bases*, pages 478–499.

Agrawal, R. and Srikant, R. (1995). Mining sequential patterns.*Proceedings of the International Conference on Data Engineering*.

Lu, H., Han, J., and Feng, L. (1998). Stock movement Prediction and n-dimensional inter-transaction association rules. *Proceedings of the 1998 ACM SIGMOD Workshop on Research Issues on Data Mining and Knowledge Discovery*, pages 12:1–12:7.

Qin, L. and Shi, Z. (2006). Efficiently mining association rules from time series. *International Journal of Information Technology*, 12(4):30–38.