Yield Optimization Using Advanced Statistical Correlation Methods

Jeff Tikkanen[1], Sebastian Siatkowski[1], Nik Sumikawa[2], Li-C. Wang[1] and Magdy S. Abadir
[1]University of California, Santa Barbara and [2]Freescale Semiconductor

***Abstract*—This work presents a novel yield optimization methodology based on establishing a strong correlation between a group of fails and an adjustable process parameter. The core of the methodology comprises three advanced statistical correlation methods. The first method performs multivariate correlation analysis to uncover linear correlation relationships between groups of fails and measurements of a process parameter. The second method partitions a dataset into multiple subsets and tries to maximize the average of the correlations each calculated based on one subset. The third method performs statistical independence test to evaluate the risk of adjusting a process parameter. The methodology was applied to an automotive product line to improve yield. Five process parameter changes were discovered which led to significant improvement of the yield and consequently significant reduction of the yield fluctuation.**

1. Introduction

Yield is one of the most important metrics to indicate the success of a product project. Therefore, it is not unusual that efforts to improve yield continue into the mass production stage. In this work, yield optimization specifically refers to such efforts in production stage where mass amounts of test data become available and can be utilized to improve yield.

Due to process variations, yield is not a constant across wafers and lots. For example, Fig. 1 illustrates a fluctuation of yield across wafers. The plot shows the probability density distribution of yield estimated based on 2000+ wafers. The chip is a sensor device for the automotive market, which contains a controller, sensors, analog and RF components.

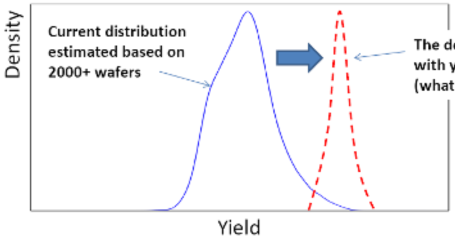


Fig. 1. **Illustration of yield fluctuation and our goal**

Given Fig. 1, it is desirable to push the yield distribution to the high end. In this sense, yield optimization can be for (1) pushing the mean of the yield to the right and (2) simultaneously reducing the variance .

Because yield is such an important metric, multiple teams are in charge of improving it. For example, yield is a function of test. Hence, it is possible to improve yield by improving test (under the constraint that quality such as customer return rate is not worsened). From this end, note that result shown in Fig. 1 was after multiple test revisions.

Yield can also be design dependent. It is noted that the result seen in Fig. 1 was also after one design revision. Therefore, additional yield improvement from Fig. 1 would represent added value to these design and test efforts.

The third way to improve yield is by adjusting the process. In order to identify which process parameter(s) to tune, evidences are required to show strong correlation relationships between process parameter(s) and certain types of fails of interest. This task is carried out by the yield analysis team.

To search for a high correlation between a process parameter and a type of fails, an intuitive methodology can be based on a flow of the following four steps: (1) Identify a type of fails to investigate. (2) Calculate the numbers of fails across N wafers as $\vec{x} = \{x_1, \ldots, x_N\}$. (3) Calculate the measured value of a selected process parameter across the N wafers, one value per wafer as $\vec{y} = \{y_1, \ldots, y_N\}$. (4) Calculate the (Pearson) correlation coefficient such as

$$Corr(\vec{x}, \vec{y}) = \frac{\sum_i^N (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_i^N (x_i - \bar{x})^2 \sum_i^N (y_i - \bar{y})^2}}$$

where $\bar{x}$ and $\bar{y}$ are the mean of $\vec{x}$ and the mean of $\vec{y}$, respectively. Then, the parameters are ranked by the correlation coefficients and the top parameters are identified.

Consider the step (1) above. In test, failing parts are organized into different test bins. Usually, similar categories of fails are grouped into the same bin. Hence, it is natural to analyze each bin independently. For example, Fig. 2 depicts the average number of fails for a list of test bins (left plot), and for the top three most failing bins (bins 26, 25, and 28), their wafer-to-wafer fail fluctuations over time (right plot).

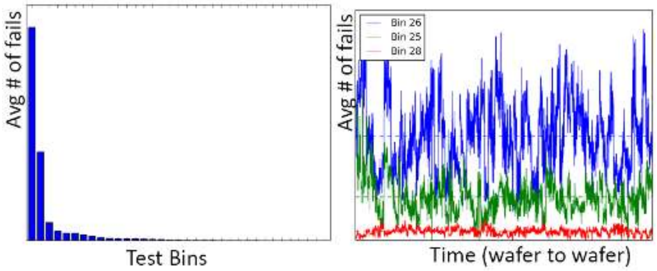


Fig. 2. **Bins of fails and their fluctuations**

Given Fig. 2, it is natural to consider bin 26 first, followed by bin 25. Suppose in step (1) we choose the bin 26. Then, in step (2) we extract the data vector $\vec{x}$ across the 2000+ wafers based on the failing dies (or ”fails”) recorded in bin 26.

Although partitioning the fails based on test bins makes sense, it is not the only way one can define a type of fails. For example, Fig. 3 shows failing statistics based on individual tests in bins 26 and 25. As we can see, tests A,B,C,D each

This work is supported in part by Semiconductor Research Corporation projects 2012-TJ-2268, 2013-TJ-2466, and by National Science Foundation Grant No. 1255818

978-1-4799-4722-5/14/$31.00 ©2014 IEEE

has a significant number of fails. Hence, the type of fails can also be defined based on each individual test.

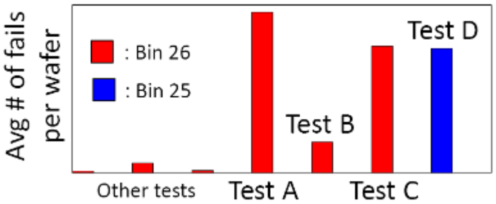


Fig. 3. **Failing statistics based on individual tests**

Now consider step (3). Suppose t process parameters are measured for each wafer (In our work, $t > 130$). In our case, a process measurement is repeated on five sites on each wafer. Fig. 4 illustrates these five sites (left plot) and the (wafer-to-wafer) fluctuation of the average measured value over the five sites for one process parameter (right plot). For each process parameter, data vector $\vec{y}$ can therefore be calculated as the vector of the average values across all wafers.

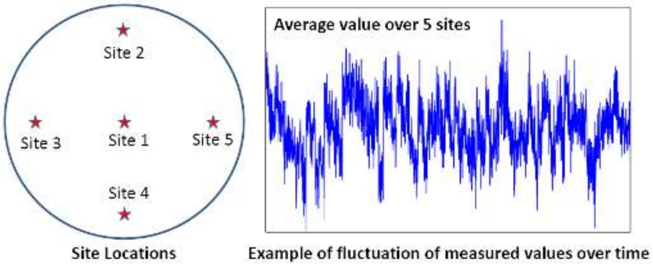


Fig. 4. **Measurement sites and a fluctuation example**

Finally, step (4) calculates the correlation coefficient $Corr(\vec{x}, \vec{y})$. Steps (3)-(4) can be applied to each process parameter to identify the one with the highest correlation. For example, the highest correlations found for bin 26, test A, test B and test C are depicted in Fig. 5.

In these plots, the x-axis is the average value of the process parameter and y-axis is the number of fails. Each dot is a wafer. Note that the measured process values in the Test B plot are more discrete than others.

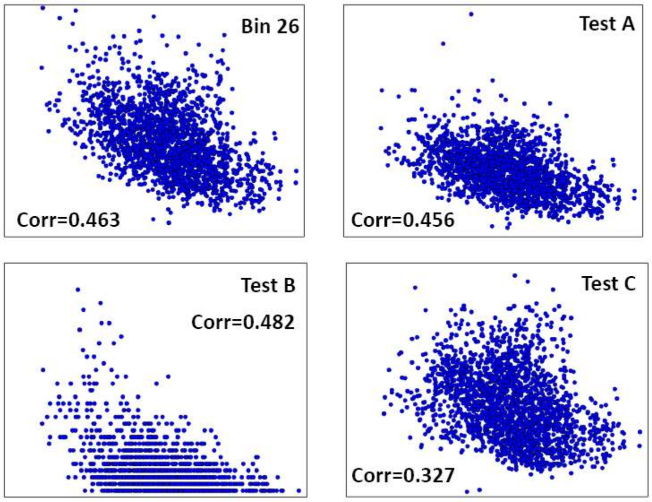


Fig. 5. **Illustrating the starting point of this work**

Fig. 5 basically shows that no strong correlation was found to support any potential process parameter adjustment. Similar results were found for other types (e.g. bin 25). Before this work, the yield analysis team had conducted extensive analysis and did not find a strong correlation. Results like Fig. 5 illustrates the starting point of this work.

The rest of the paper is organized as below. Section 2 discusses potential issues with the intuitive methodology. Section 3 explains a multivariate correlation methodology and demonstrates its usefulness in uncovering a strong correlation which could not be found before. Section 4 describes a subset discovery problem formulated to enable finding additional strong correlations. Section 5 presents a risk evaluation method based on statistical independence test to assess the risk of a process parameter adjustment. Following the uncovered process parameter changes, section 6 summarizes the silicon results with significant yield improvement. Section 7 briefly reviews prior works and comments on the statistical analysis tools used in this work. Section 8 concludes.

2. Potential issues with the intuitive methodology

An obvious concern with the intuitive methodology is the use of Pearson correlation to evaluate statistical dependence. Pearson correlation coefficient, although popular, is not a robust statistic for dependence test. For example, it is known that the correlation coefficient can be quite sensitive to strong outliers. Fig. 6 illustrates such an example in our analysis.

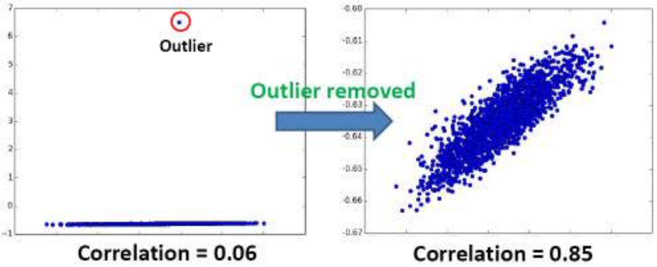


Fig. 6. **Correlation can be sensitive to outliers**

The left plot shows that an outlier is far from the majority of distribution. The correlation calculated based on this plot is 0.06. The right plot shows the result by removing the outlier (consequently, the scale of y-axis changes). The correlation becomes 0.85. The sensitivity to outliers can be one reason to use other statistics such as the rank correlation coefficients, e.g. Spearman's ρ or Kendall's τ. Alternatively, a preprocessing step can be taken to remove outliers.

2.1. *Need for multivariate analysis*

While there are many other alternatives for univariate statistical dependence test, a more fundamental issue is that the data in our analysis is inherently multivariate.

To apply a univariate analysis between $\vec{x}$ and $\vec{y}$, one has to define how to calculate $\vec{x}$ and $\vec{y}$ from the data. The correlation result can be dependent on the calculation. For example, each process parameter was measured on five sites. In the univariate analysis above, we simply take the average of the five sites.

Fig. 7 uses eight parameters P1-P8 to illustrate the variability across the five measurement sites. For each parameter, we show the correlation coefficients between measured values from all pairs of sites. Since there are five sites, there are 10 (pairwise) correlation coefficients shown for each parameter. In total, therefore there are 80 correlation coefficients (as shown in the x-axis) divided into eight blocks.

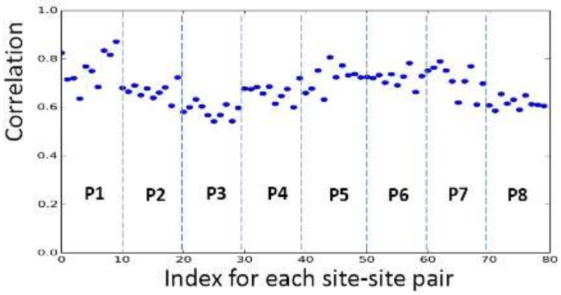


Fig. 7. **Examples of site-site pairwise correlations**

Fig. 7 shows that the correlations between two sites can range from below 0.6 to above 0.8. The bottom line: there can be significant variability across sites. If that is the case, only considering the average may not be sufficient.

Furthermore, recall from Fig. 3 that tests A-D are important due to their associated large numbers of fails. Fig. 8 depicts the distribution of test values for test A and for test D.

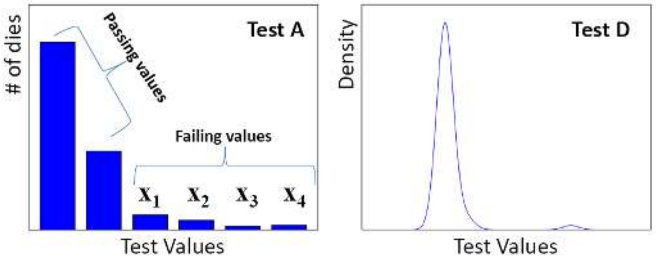


Fig. 8. **Discrete test A and continuous test D**

Test A is a discrete test. Its values can fall into 6 categories. The left plot shows a histogram of the test values across one lot of wafers. Test D is a continuous tests. The right plot shows the probability distribution of the test values estimated based on one lot of wafers.

In the intuitive methodology, $\vec{x}$ is based on the number of fails. The number of fails does not include all the information contained in a test distribution as shown in Fig. 8. For example, a process parameter may be correlated to only a subset of the fails (e.g. only having the X_4 value) or to the shape of a test distribution. For capturing these types of correlations, the correlation analysis needs to be extended to go beyond just using the number of fails as the target of the correlation.

For example, for test A we may want to correlate directly to a multivariate vector (X_1, X_2, X_3, X_4) as shown in Fig. 8 (In general, we may have X_1-X_n for a large n). For test D, we may want to correlate to some characteristics of the distribution. Both demand a multivariate correlation analysis.

Fig. 9 gives another reason to consider multivariate analysis. Fig. 9 plots two wafer heat maps based on the number of fails in a single lot. Observe that test A fails concentrate on the edge while test D fails reside more on an inner ring. This raises the question: Would it be possible that a strong correlation exists only in a certain region of a wafer?

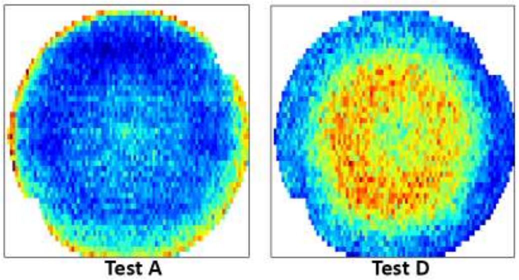


Fig. 9. **Examples of failing wafer heat map**

One way to address this question can be to partition the wafer into multiple regions. A strong correlation may exist with each region individually or with a combination of multiple regions collectively. Again, this can be formulated as a multivariate correlation analysis problem.

3. Multivariate correlation and statistical dependence

It is well known that correlation coefficient zero ($Corr(\vec{x}, \vec{y}) = 0$) is not the same as statistical independence. In other words, finding no strong correlation does not imply that the type of fails and the process parameters have no strong dependence. A zero correlation coefficient only guarantees statistical independence when the joint probability distribution $P(x, y)$ is normal. In test data analysis, this is often not the case (see, e.g., the left plot in Fig. 8 - not normal).

To go beyond correlation coefficient, one can follow the well established principles for measuring statistical dependence, proposed by Rényi who showed that one sound measure for statistical dependence is the following (see [1][2]):

$$\mathcal{Q}(P(x, y)) = \sup_{f,g} Corr(f(x), g(y)) \tag{1}$$

where f and g are Borel measurable and bounded functions. The notation "sup" denotes the least upper bound. Hence, equation (1) basically denotes the maximum correlation across *all* possible functions f, g. Rényi showed that the quantity $\mathcal{Q}(P(x, y)) = 0$ implies statistical independence. The quantity $\mathcal{Q}(P(x, y)) = 1$ implies $x = h(y)$ or $y = h(x)$ for some function h, i.e. there is a strict dependence between x and y.

In equation (1), x and y are two random variables. Replacing them with two random vectors X and Y and also replacing the "sup" with the maximum "max" we obtain the following measure of dependence for two random vectors:

$$CC(X, Y) = \max_{f,g} Corr(f(X), g(Y)) \tag{2}$$

where f and g are functions that take a vector as input and output a real value. For example, when f and g are dot-product functions with weight vectors W_x and W_y, we have

$$CC(X, Y) = \max_{W_x, W_y} Corr(\langle W_x, X\rangle, \langle W_y, Y\rangle) \tag{3}$$

where $\langle \cdot, \cdot \rangle$ denotes the dot-product of the two vectors. In this case, the correlation calculated in equation (3) is the maximum correlation across all possible linear transforms denoted by W_x, W_y. Equation (3) is the traditional Canonical Correlation Analysis (CCA) (see, e.g. [3]).

$CC(X, Y) = 0$ in equation (3) does not guarantee statistical independence because W_x and W_y are linear transforms and the functions f and g in the original equation (2) can be non-linear. To extend CCA to consider non-linear transforms, kernel CCA (KCCA) applies the so-called ”kernel trick” [4].

To apply the kernel trick, one starts with choosing a kernel function $k(X, Z)$ that measures the similarity between two vectors X and Z. A kernel $k()$ corresponds to a mapping function $\Phi()$ such that $k(X, Z) = \langle \Phi(X), \Phi(Z) \rangle$. The idea of KCCA is to apply CCA on the transformed vectors [5]:

$$KCC(X, Y) = CC(\Phi(X), \Phi(Y)) \tag{4}$$

The ”kernel trick” corresponds to calculating equation (4) without explicitly using the mapping function $\Phi()$. Instead, only the kernel function $k()$ is involved in the computation. To explore non-linear correlations we choose a kernel of which the $\Phi()$ is non-linear. In this section, we discuss how CCA can be used to find strong correlations beyond traditional correlation coefficient. In Section 5, we will discuss how KCCA can be used for risk evaluation.

3.1. Canonical Correlation Analysis (CCA)

Let $X = (x_1, \ldots, x_n)$ and $Y = (y_1, \ldots, y_m)$ be two random vectors where each x_i and each y_j are random variables. In practice, the distribution of each x_i is measured by N sample points $\vec{x}_i = (x_{1i}, \ldots, x_{Ni})$. Similarly, each y_j is measured by N sample points $\vec{y}_j = (y_{1j}, \ldots, y_{Nj})$. Hence, we have a data matrix S_x for X and a data matrix S_y for Y. This is illustrated below.

$$\begin{array}{c|cccc|} & \vec{x}_1 & \vec{x}_2 & \cdots & \vec{x}_n \\ \vec{u}_1 & x_{11} & x_{12} & \cdots & x_{1n} \\ \vec{u}_2 & x_{21} & x_{22} & \cdots & x_{2n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \vec{u}_N & x_{N1} & x_{N2} & \cdots & x_{Nn} \end{array} \times \left|\begin{array}{c} w_1^x \\ w_2^x \\ \vdots \\ w_n^x \end{array}\right| \qquad \begin{array}{c|cccc|} & \vec{y}_1 & \vec{y}_2 & \cdots & \vec{y}_m \\ \vec{v}_1 & y_{11} & y_{12} & \cdots & y_{1m} \\ \vec{v}_2 & y_{21} & y_{22} & \cdots & y_{2m} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \vec{v}_N & y_{N1} & y_{N2} & \cdots & y_{Nm} \end{array} \times \left|\begin{array}{c} w_1^y \\ w_2^y \\ \vdots \\ w_n^y \end{array}\right|$$

$$S_x \qquad\qquad w_x \qquad\qquad\qquad S_y \qquad\qquad w_y$$

Let $w_x = (w_1^x, \ldots, w_n^x)$ be a weight vector for X. In matrix multiplication ”$S_x \times w_x$,” the weight vector transforms each sample vector $\vec{u}_i$ into a canonical value $c(X)_i$ as below:

$$c(X)_i = \langle w_x, \vec{u}_i \rangle = \sum_{k=1}^{n} (x_{ik} w_k^x) \tag{5}$$

Therefore, the result of $S_x \times w_x$ is a vector $(c(X)_1, \ldots, c(X)_N)$. Similarly, $S_y \times w_y$ is a vector of N values. The correlation coefficient between the two vectors can then be calculated, denoted as $Corr(S_x w_x, S_y w_y)$. Then, the sample canonical correlation between X and Y is defined as

$$\begin{aligned} CC(X, Y) &= \max_{w_x, w_y} Corr(S_x w_x, S_y, w_y) && (6) \\ &= \max_{w_x, w_y} \frac{\langle S_x w_x, S_y w_y \rangle}{\|S_x w_x\| \|S_y w_y\|} && (7) \end{aligned}$$

Note that $S_x w_x = (w_x' S_x')'$ where $'$ denotes the matrix transpose operator. Then, we have $\langle S_x w_x, S_y w_y \rangle = w_x' S_x' S_y w_y$ for the nominator in equation (7). Similar changes can be applied to the denominator to rewrite equation (7) as:

$$\begin{aligned} CC(X, Y) &= \max_{w_x, w_y} \frac{w_x' S_x' S_y w_y}{\sqrt{w_x' S_x' S_x w_x} \sqrt{w_y' S_y' S_y w_y}} && (8) \\ &= \max_{w_x, w_y} \frac{w_x' C_{xy} w_y}{\sqrt{w_x' C_{xx} w_x w_y' C_{yy} w_y}} && (9) \end{aligned}$$

where $S_x' S_y = C_{xy}$ denotes the sample covariance matrix between X and Y, $S_x' S_x = C_{xx}$ denotes the sample covariance matrix for X and $S_y' S_y = C_{yy}$ denotes the sample covariance matrix for Y.

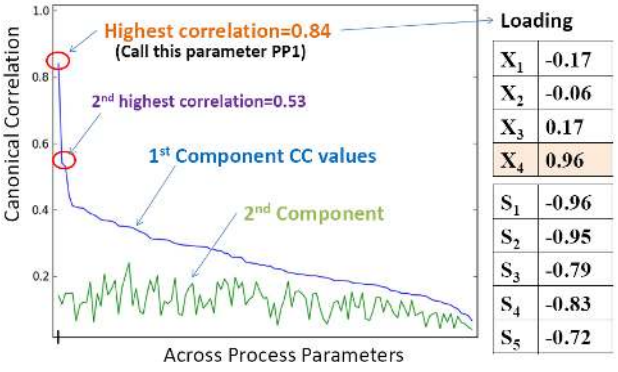
The optimization problem stated in equation (9) can be solved by applying the Lagrangian method. This leads to solving a generalized eigenproblem of the form $A w_x = \lambda B w_x$ where $A = C_{xy} C_{yy}^{-1} C_{yx}$ and $B = C_{xx}$ (see e.g. [3]).

Solving the generalized eigenproblem leads to a sequence of weight vector pairs for w_x and w_y. The number of weight vector pairs is equal to $\min(n, m)$, the smallest dimension between X and Y. The first weight vector pair gives the largest correlation. This can be called as the *1st CC component*. The second weight vector pair gives the second largest correlation (*2nd CC component*), and so on.

3.2. Analysis of test A in bin 26

Refer back to the test A plot in Fig. 8. Let $X = (X_1, X_2, X_3, X_4)$ be the random vector as shown in the plot. Each X_i is a random variable, representing on each wafer the number of dies whose test A values fall into the X_i category. As shown in Fig. 8 for test A, test values of X_1-X_4 are considered as failing. The first two are passing.

Given a process parameter P, let $Y_P = (S_1, S_2, S_3, S_4, S_5)$ be the random vector denoting the measured values on the five sites (see Fig. 4). Then, we can run CCA on (X, Y_P). This can be carried out for each process parameter P to determine which one has the highest canonical correlation to X.



X_1	-0.17
X_2	-0.06
X_3	0.17
X_4	0.96
S_1	-0.96
S_2	-0.95
S_3	-0.79
S_4	-0.83
S_5	-0.72

Fig. 10. **CCA results for Test A**

Fig. 10 plots the canonical correlations based on the first two CC components for all process parameters under consideration. Observe the correlations of the 1st CC components are consistently (much) higher than the correlations of the 2nd CC components. Also, the highest canonical correlation is 0.84, indicating a strong correlation between test A fails and the first process parameter shown in the plot (call this process parameter PP1). The second highest correlation is 0.53 which is based on the second process parameter in the plot.

The table on the right shows the *loadings* for each random variable. To understand what a loading is, suppose $\vec{x}_1$ is the column vector denoting the N sample values measured on the random variable X_1. The *loading* in CCA for X_1 is simply the regular correlation coefficient between $\vec{x}_1$ and the transformed vector $S_x w_x$, i.e. loading$(\vec{x}_1) = Corr(\vec{x}_1, S_x w_x)$.

In Fig. 10 we see that the loading for X_4 is 0.96 that is much higher than the loadings for other X's. This indicates that the canonical correlation 0.84 is contributed more from the X_4 variable than from other X variables. In other words, it is likely that the X_4 type of fails by itself have a high correlation to the PP1 parameter.

Following the same notation used above, the left plot of Fig. 11 shows how the values from $S_x w_x$ and the values from $S_y w_y$ correlate. The plot clearly shows a linear trend.

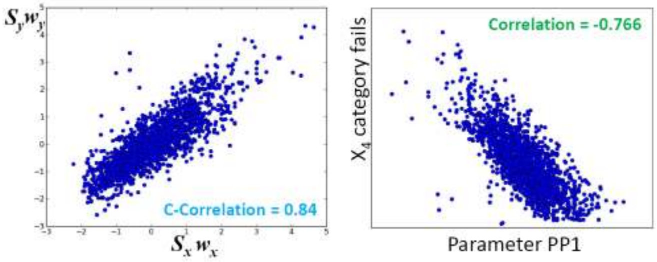


Fig. 11. **Further illustration of results shown in Fig. 10**

Then, the right plot of Fig. 11 shows how the X_4 type of fails by itself correlates to the PP1 parameter. The x-axis is the average measured values for PP1 from the five sites (each dot is a wafer). As expected, a high correlation is observed with a negative correlation coefficient -0.766 between X_4 and PP1.

Next, we removed X_4 fails from the analysis and let $X = (X_1, X_2, X_3)$. We reran the CCA on the new X. The highest canonical correlation found was around 0.52, indicating that no high correlation could be found for X_1-X_3 types of fails.

3.3. *Analysis of test D (Bin 25)*

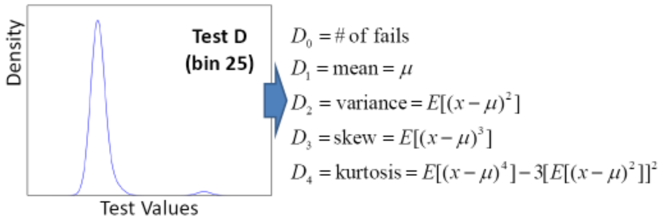

Density
Test D
(bin 25)
Test Values
D_0 = # of fails
D_1 = mean = μ
D_2 = variance = $E[(x-\mu)^2]$
D_3 = skew = $E[(x-\mu)^3]$
D_4 = kurtosis = $E[(x-\mu)^4] - 3[E[(x-\mu)^2]]^2$

Fig. 12. **Encoding a distribution into a multivariate vector**

Refer back to Fig. 8 where the distribution of test D (bin 25) is shown. Fig. 12 shows a way to encode the characteristics of the distribution into a vector of five quantities, i.e. $X = (D_0, D_1, D_2, D_3, D_4)$. Then, we ran CCA based on this X and each process parameter vector Y_P. Note that the encoding was for the entire distribution, not just for the failing distribution.

The left plot of Fig. 13 summarizes the CCA result where the highest canonical correlation found is 0.82. It is interesting to note that this highest correlation is based on the same parameter PP1 found in Fig. 10. For comparison, the right plot of Fig. 14 shows that univariate correlation between the number of test D fails (test D is the only test with bin 25) and the average PP1 value is only 0.305.

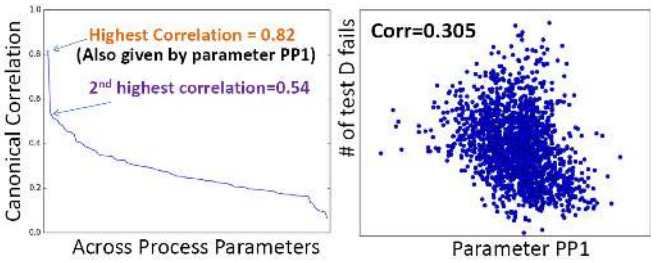


Fig. 13. **Canonical Correlation vs. Pearson Correlation**

Fig. 13 indicates that PP1 is highly correlated to some characteristics of the test D distribution, but not highly correlated to the number of test D fails.

As we examined the CCA loadings for D_0 to D_4, we found that the two highest loadings were -0.74 for D_1 (the mean) and -0.91 for D_2 (the variance). This indicated that the PP1 was mostly negatively correlated to the mean and variance of the test D distribution. To confirm, Fig. 14 shows the scatter plots for the D_1-vs-PP1 and D_2-vs-PP1 with their respective Pearson correlation coefficients. The negative correlation trends can be observed in both plots.

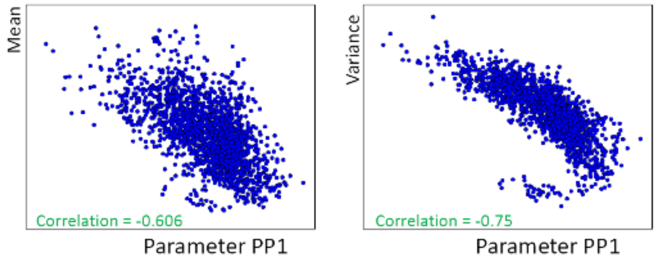


Fig. 14. **PP1 is correlated to the mean and variance of bin 25**

3.4. *Summary of the first finding - parameter PP1*

In the above CCA analyses, we use CCA to identify a high correlation scenario. Then, we analyze the loadings to identify the variable(s) contributing the most to the high correlation. Then, we use univariate correlation to confirm the findings.

Results from Fig. 11 and Fig. 14 both suggest to increase PP1 for yield improvement. With Fig. 11, the expected impact would be to reduce the number of X_4 fails. With Fig. 14, the expected impact would be to decrease the mean and variance of the test D distribution, resulting fewer fails because the test limit is set on the right of the distribution.

3.5. *Note on applying CCA in location-based analysis*

To illustrate how CCA can be applied to analyze location-based correlations, Fig. 15 gives an example of partitioning the X_4 type of fails in bin 26 into two groups, the inner group and the outer group. Let I be the number of X_4 fails in the inner group and O be the number of X_4 fails in the outer group. Let the random vector $X = (I, O)$ in CCA.

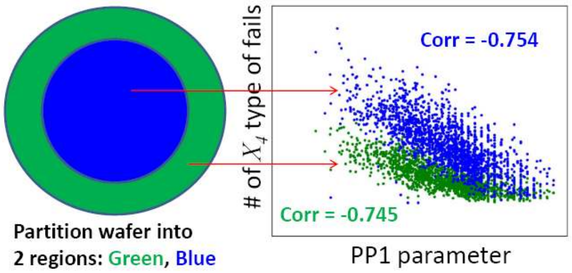


Fig. 15. **Partitioning X_4 type of fails based on their locations**

Running CCA on X and Y_{PP1} for process parameter PP1 gives canonical correlation 0.8, higher than the correlation -0.766 shown in Fig. 11. Nevertheless, Fig. 15 shows that the Pearson correlations are -0.754 and -0.745 for I-type of fails and for O-type of fails, respectively. These results are comparable to -0.766 but not higher. Hence, this particular location-based CCA did not improve the correlation result.

4. The subset discovery problem

Earlier at the end of Section 3.2 we explain that with CCA, no high correlation could be found to account for X_1-X_3 types of test A fails in bin 26. In this section, we formulate a subset discovery problem and show that solving the problem can enable us to find additional strong correlations.

Given two random vectors X, Y, suppose X, Y are measured through a dataset S of N wafers. Let $S_1, S_2, \ldots, S_k$ denote a sequences of subsets of S where each $S_i \subset S$ and for any $i \neq j$, $S_i \cap S_j = \phi$. Let $|S_i|$ denote the size of the subset S_i. Let $CC(X,Y)_{S_i}$ denote the canonical correlation of the 1st CC component based on the wafers in subset S_i, i.e. the highest canonical correlation found. Then the subset discovery problem can be stated as the following:

Subset Discovery Problem:

$$SCC(X,Y)_{\lambda,\eta} = \max_{S_1,S_2,\ldots,S_k}\left[\frac{\sum_{i=1}^{k} CC(X,Y)_{S_i}}{k}\right]$$

subject to $|\bigcup_{i=1}^{k} S_i| \geq \lambda|S|$, and

$\forall S_i, |S_i| \geq \eta|S|$,

where $0 < \lambda < 1$ and $0 < \eta < \lambda$ are given parameters.

We call $SCC(X,Y)_{\lambda,\eta}$ the *subset canonical correlation* based on the user parameters λ and η.

Suppose $\lambda = 0.5$ (50%) and $\eta = 0.1$ (10%). The constraint $|\bigcup_{i=1}^{k} S_i| \geq \lambda|S|$ basically says that the total number of samples used in the calculation of SCC has to be no less than 50% of the size of S. On the other hand, the constraint $\forall S_i, |S_i| \geq \eta|S|$ says that the size of each subset cannot be less than 10% of the total number of samples in S. It is important to note that all subsets are disjoint. We call the two constraints, the λ-*constraint* and the η-*constraint*, respectively.

4.1. Assumption for subset discovery to be useful

The subset discovery problem tries to find k disjoint subsets under the two size constraints, to maximize the average canonical correlation across the k subsets. This is based on the assumption that for some subset S_i, $CC(X,Y)_{S_i}$ is much higher than $CC(X,Y)_S$. In other words, the correlation relationship can become a lot more apparent as we focus on a particular subset (and the correlation relationship becomes blur if we use the entire dataset).

Using less data can be better because the original dataset contains noise. For example, the measurements at a certain period may be more noisy than others. The parameter λ allows user to drop a portion of the data to maximize the resulting correlation. Another reason can be because there is a drift of the correlation relationship over time. Given two subset S_i and S_j produced at different times, a strong correlation relationship can be identified based on each subset individually but not on both subsets collectively. This drift property will be discussed in detail shortly when the results are presented.

4.2. Heuristic to approach the problem

The objective function in the subset discovery problem involves calculation of canonical correlations $CC(X,Y)_{S_i}$. Further, in search for the best subsets $S_1, \ldots, S_k$, we may need to consider all possible partitions (that satisfy the constraints) of the set S. Exhaustively searching for the optimal answer can be overly expensive.

A straightforward heuristic is to incrementally find the subsets following a greedy approach. In other words, the heuristic finds a sequence of subsets $S_1, \ldots, S_k$ such that $CC(X,Y)_{S_1} > CC(X,Y)_{S_2} > \ldots > CC(X,Y)_{S_k}$. The heuristic can be described as repeating a two-step process:

1) Given S, find the subset S_i that results in maximum $CC(X,Y)_{S_i}$ where $|S_i|$ satisfies the η-*constraint*.
2) If the λ-*constraint* is not yet satisfied, let $S = S - S_i$ and repeat step (1); Otherwise, stop.

Let S_i and S_{i+1} be two consecutive subsets found by the heuristic. Let s be a wafer that $s \in S_{i+1}$. Note that it is possible to have the situation where $CC(X,Y)_{S_i} + CC(X,Y)_{S_{i+1}} < CC(X,Y)_{S_i \cup \{s\}} + CC(X,Y)_{S_{i+1} - \{s\}}$. In other words, if we move the sample s from S_{i+1} back to S_i, we improve the sum of the two correlations.

It is important to note that if the algorithm to solve the maximization problem in step (1) is ideal, then we should have $CC(X,Y)_{S_i} > CC(X,Y)_{S_i \cup \{s\}}$. However, this does not mean that s should not be included in S_i because s may decrease the correlation based on S_{i+1} more than it decreases the correlation based on S_i. Therefore, the straightforward heuristic is not optimal.

Because computing the objective function in the subset discovery problem itself can be expensive, it is preferable not to follow a process that goes beyond the linear complexity as the greedy heuristic. Hence, we modify the objective function in step (1) by introducing a regularization term based on the size of the subset.

In step (1), instead of finding a subset to maximize $CC(X,Y)_{S_i}$, we try to maximize $CC(X,Y)_{S_i} + \gamma\frac{|S_i|}{|S|}$. In other words, if adding more samples to S_i does not decrease the correlation too much, then those samples should be added. Notice that $0 < \frac{|S_i|}{|S|} \leq 1$ and $0 \leq CC(X,Y)_{S_i} \leq 1$. Hence, value ranges of the two terms are comparable. This means that the choice of γ would not be too far from 1. The optimal choice of γ can be determined experimentally.

4.3. Analysis of X_1-X_3 types of fails from test A

While CCA itself did not find high correlation for X_1-X_3 types of test A fails, as explained in Section 3.2 before, result below shows that applying subset discovery did.

TABLE I
SUBSET CANONICAL CORRELATIONS FOR FOUR PARAMETERS PP2-PP5 FOUND TO HAVE HIGH CORRELATIONS TO THE X_1-X_3 TYPES OF FAILS

	S_1	S_2	S_3	S_4	S_5	S_6	S_7	S_8	S_9
PP2	0.84	0.78	0.643	0.69	0.641	0.63	-	-	-
PP3	0.88	0.85	0.69	0.82	0.68	0.61	0.53	0.52	-
PP4	0.86	0.87	0.85	0.82	0.83	0.82	0.81	0.79	0.75
PP5	0.86	0.81	0.82	0.77	0.68	0.59	0.61	-	-

Table I summarizes the results of applying subset discovery to analyze X_1-X_3 types of fails. For λ-*constraint*, we set $\lambda = 0.5$. For η-*constraint*, we set $\eta = 0.0625$ which means using a minimum of 125 wafers in each subset for a total of 2000+ wafers. In the table, the S_i represents the ith subset found by following the greedy heuristic discussed above. Each number shown is the canonical correlation of the 1st CC component based on the particular subset.

Notice that for the same parameter, it is not always true that the correlation found with S_i is greater than or equal to that with S_{i+1}. For example, for PP2 the correlation found with S_4 (0.69) is higher than that found with S_3 (0.643). This is due to the regularization discussed above that in each step, we try to maximize the term $CC(X,Y)_{S_i} + \gamma\frac{|S_i|}{|S|}$ instead of just maximizing the canonical correlation $CC(X,Y)_{S_i}$.

4.4. Result illustration

Table I shows that the four parameters PP2-PP5 can be highly correlated to the X_1-X_3 types of fails. To illustrate why subset discovery is needed for finding these correlations, Fig. 16 and Fig. 17 show, for each parameter, a scatter plot based on selected two subsets (green and blue). The correlations shown in these plots are Pearson correlation coefficients between the number of fails in the subset and the average parameter measured value across five sites.

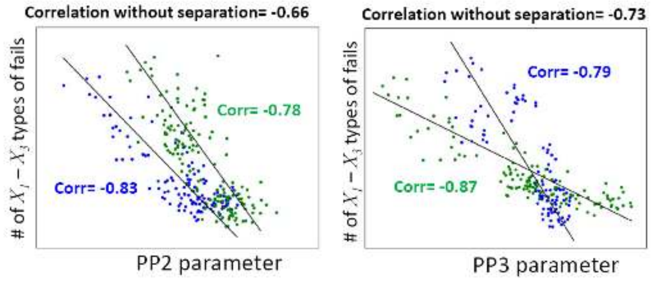


Fig. 16. **Subset discovery found two process parameters, PP2 and PP3, highly correlated to X_1-X_3 types of fails in bin 26**

Consider the first plot in Fig. 16, the two subsets individually correlate to the PP2 by -0.78 and -0.83. Collectively, the correlation drops to -0.66. The reason can be seen clearly from the plot that between the two subsets, there is a shift of the trend. Therefore, when all the data points are analyzed together, the trend becomes less apparent.

Similar shifts of trends can be observed for the PP3 plot in Fig. 16 and the two plots in Fig. 17. In all cases, the correlations based on each subset of wafers are higher than the correlations based on the two subsets combined.

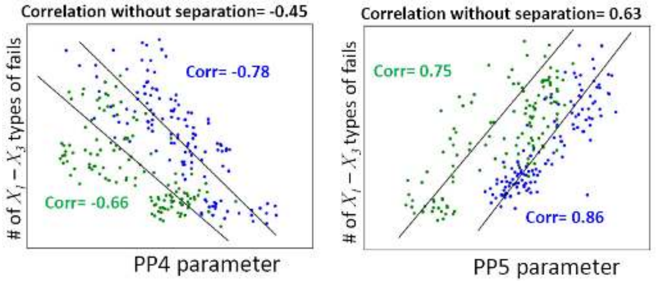


Fig. 17. **Subset discovery found two more process parameters, PP4 and PP5, correlated to X_1-X_3 types of fails in bin 26**

Fig. 16 and Fig. 17 also show that subset discovery can be applied independently of CCA to find correlations. In the two figures, all correlations are based on Pearson correlation and as we can see, high correlations can be found once the appropriate subsets are identified.

4.5. Double check X_4 types of fails from test A

Earlier with Fig. 11 we have established that the X_4 type of fails is highly correlated to the parameter PP1. This is supported by the highest canonical correlation found, 0.84, together with the Pearson correlation -0.766.

Table II shows the result of applying subset discovery to rerun the CCA analysis with $X = (X_1, X_2, X_3, X_4)$. In the subset discovery we use the same parameter setting $\lambda = 0.5$ and $\eta = 0.0625$. Table II shows that all subset canonical correlations are greater than the canonical correlation 0.84 found before. Four subsets give correlations above 0.9.

TABLE II
CONFIRMING STRONG CORRELATION BETWEEN X_4 FAILS AND PP1

	S_1	S_2	S_3	S_4	S_5	S_6	S_7	S_8
PP1	0.93	0.92	0.91	0.90	0.89	0.889	0.874	0.864

The left plot of Fig. 18 shows results for X_4 type of fails from two subsets. We see that individually the Pearson correlation coefficients are -0.91 and -0.85 which are much improved from the correlation coefficient -0.766 found before.

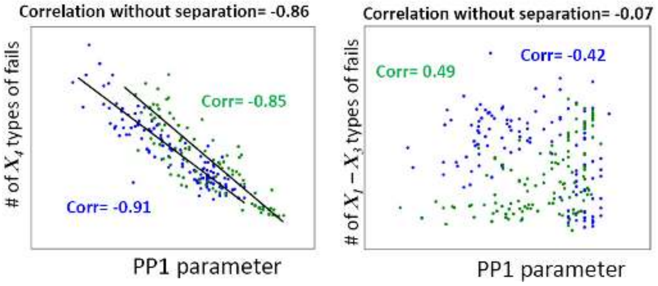


Fig. 18. **Subset discovery confirms X_4 type of fails highly correlated to PP1 while X_1-X_3 types of fails do not**

For comparison, the right plot of Fig. 18 shows results for X_1-X_3 types of fails. The result earlier shows that they do not have a high correlation to parameter PP1. The plot confirms the finding by showing two subsets with low (and opposite) correlations and with combined correlation almost zero.

4.6. Summary of findings

TABLE III
SUMMARY OF FINDINGS AND SUPPORTING EVIDENCES

Para	Fail Type	Trend	Support
PP1	X_4 type, test A, Bin 26	Negatively correlated	Figs. 11,18
PP1	Bin 25	Negatively correlated	Fig. 14
PP2	X_1-X_3 types, test A, Bin 26	Negatively correlated	Fig. 16
PP3	X_1-X_3 types, test A, Bin 26	Negatively correlated	Fig. 16
PP4	X_1-X_3 types, test A, Bin 26	Negatively correlated	Fig. 17
PP4	tests B,C, Bin 26	Negatively correlated	omitted
PP5	X_1-X_3 types, test A, Bin 26	Positively correlated	Fig. 17

Table III summarizes the correlation findings. Based on the results, the recommendation was: increasing PP1, PP2, PP3, PP4, and decreasing PP5.

5. **Risk evaluation**

Silicon experiments are expensive. Therefore, before any recommendation of process change was implemented, we had to evaluate its risk. For example, the result above shows that increasing PP1 would improve bin 25 and bin 26 yield. However, it might also simultaneously increase the failing rates of other bins. To make sure that this was unlikely to happen, we needed to assess the statistical dependence between PP1 and other bins. In other words, for risk evaluation, it was desirable to demonstrate that PP1 and other bins were likely to be statistically independent.

The CCA and subset CCA methods above could be used as a basis for risk evaluation. However, not finding a high correlation using those methods does not mean that the process parameter and the type of fails are statistically independent. As discussed in Section 3, this is because CCA only looks for linear correlations. Hence, to take the evaluation one step further, we need to consider *non-linear correlations*.

5.1. Kernel CCA (KCCA) looks for non-linear correlations

For non-linear CCA, we can employ the idea of *kernel CCA* as stated in equation (4) in Section 3 before. Fig. 19 illustrates the basic principle of kernel CCA.

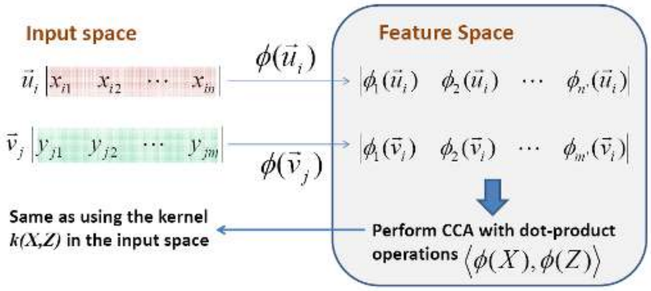

Input space
$\vec{u}_i$ $|x_{i1}\ \ x_{i2}\ \cdots\ x_{in}|$
$\vec{v}_j$ $|y_{j1}\ \ y_{j2}\ \cdots\ y_{jm}|$
$\phi(\vec{u}_i)$
$\phi(\vec{v}_j)$
Feature Space
$|\phi_1(\vec{u}_i)\ \ \phi_2(\vec{u}_i)\ \cdots\ \phi_{n'}(\vec{u}_i)|$
$|\phi_1(\vec{v}_i)\ \ \phi_2(\vec{v}_i)\ \cdots\ \phi_{m'}(\vec{v}_i)|$
Perform CCA with dot-product operations $\langle\phi(X),\phi(Z)\rangle$
Same as using the kernel $k(X,Z)$ in the input space

Fig. 19. **Illustration of kernel CCA**

Let $k()$ be a kernel function and $\Phi()$ be corresponding mapping function where $k(X,Z) = \langle\Phi(X),\Phi(Z)\rangle$. Essentially, $\Phi()$ takes an input sample vector and maps it into another vector in the *feature space*. In Fig. 19, for example, $\vec{u}_i$ is an n-dimensional sample vector (see also the matrix illustration in Section 3). $\Phi(\vec{u}_i)$ maps it to an n'-dimensional feature vector $|\Phi_1(\vec{u}_i), \ldots \Phi_{n'}(\vec{u}_i)|$ in the feature space.

Common kernels include the Gaussian kernel: $k(X,Z) = e^{-g\|X-Z\|^2}$ and polynomial kernel of degree d: $k(X,Z) = (\langle X,Z\rangle + R)^d$ for some constant R. For a Gaussian kernel, the dimensionality in the feature space is infinity ($n' = \infty$). For a polynomial kernel of degree d, the dimensionality $n' = \binom{n+d}{d}$ where n is the input dimension. (see, e.g. [4]).

Kernel CCA is equivalent to performing the regular CCA in the feature space based on the mapped vectors. Refer back to equation (7) earlier for CCA formulation. The trick is to recognize that CCA is based on the dot-product operations between two vectors, i.e. like $\langle X,Z\rangle$. Because $\langle\Phi(X),\Phi(Z)\rangle$ in the feature space is the same as $k(X,Z)$ in the input space, to perform CCA in the feature space, one can simply use the kernel operations $k(X,Z)$ in the input space to achieve the same purpose as illustrated in Fig. 19. Hence, the mapping $\Phi()$ is never explicitly involved in kernel CCA. Rather, the computation is carried out using $k(X,Z)$ in the input space.

Let S_x be the data matrix for X containing N sample vectors $(\vec{u}_1, \ldots, \vec{u}_N)$. The kernel matrix K_x is an $N \times N$ matrix $|k(\vec{u}_i, \vec{u}_j)|_{\forall i,j}$. Also let K_y denote the kernel matrix for Y. Notice that K_y is also an $N \times N$ matrix because there are N samples (wafers).

With the kernel trick, the kernel CCA can be stated as the following (α, β are N-dimensional vectors) [5]:

$$KCC(X,Y) = \max_{\alpha,\beta} \frac{\alpha' K_x K_y \beta}{\sqrt{\alpha' K_x^2 \alpha}\sqrt{\beta' K_y^2 \beta}} \qquad (10)$$

If we compare equation (10) to the original CCA formulation equation (9), we see that the only things we change are replacing S_x with K_x and S_y with K_y (see, e.g. [3]).

Given a kernel with a non-linear mapping $\Phi()$, performing CCA in the feature space is therefore equivalent to maximizing the non-linear correlation in the input space.

5.2. Kernel CCA as a statistical independence test

It turns out that the formulation of equation (10) is not very useful in practice. This is because with a powerful enough kernel, the KCC is almost guaranteed to be 1. For example, with a universal kernel (e.g. a Gaussian kernel mentioned above is a universal kernel) as defined in [6], the authors in [7] show that the KCC result is always 1, independent of the dataset. In other words, one can always find a mapping function $\Phi()$ complex enough to *overfit* the data so that the resulting correlation is 1.

To resolve the overfitting issue, the most popular way is by *regularization* [7] - In equation (10) the objective function is changed by replacing $\sqrt{\alpha' K_x^2 \alpha}$ with $\sqrt{\alpha' K_x^2 \alpha + \gamma \alpha' K_x \alpha}$ and $\sqrt{\beta' K_y^2 \beta}$ with $\sqrt{\beta' K_y^2 \beta + \gamma \beta' K_y \beta}$. The user-input parameter γ controls the "complexity" of the linear transform functions used by CCA in the feature space. A small γ allows higher complexity and vice versa. The work [7] proves that with regularization and universal kernels, $KCC(X, Y) = 0$ if and only if X and Y are independent.

5.3. Practical implementation of kernel CCA

We experimented with the regularized kernel CCA and found that in practice, it was hard to interpret the results. For example, depending on the choice of γ, the kernel CCA may give a lower correlation than regular CCA - a property that is not desirable because for risk evaluation we desire kernel CCA to be always more powerful than CCA, i.e. always gives an equal or higher correlation. We therefore chose a different implementation based on the idea proposed in [8].

The idea is to approximate kernel CCA by (1) running kernel Principal Component Analysis (KPCA) [9] to extract the first C principal components in the feature space and (2) running regular CCA based on the transformed dataset by the C principal components. In other words, in Fig. 20 the kernel trick is applied to perform PCA in the feature space (kernel PCA), and the CCA is then applied *directly* in the feature space by selecting only the first C kernel PCA components.

To illustrate the use of kernel CCA for dependence test, Fig. 20 shows results based on parameter PP1. From the CCA based analyses, we know that PP1 is highly correlated to test A and test D. We also knew that PP1 was not highly correlated to the most-frequent failing tests in bins 20, 28 and 30. Fig. 20 shows how kernel CCA differentiates these two groups.

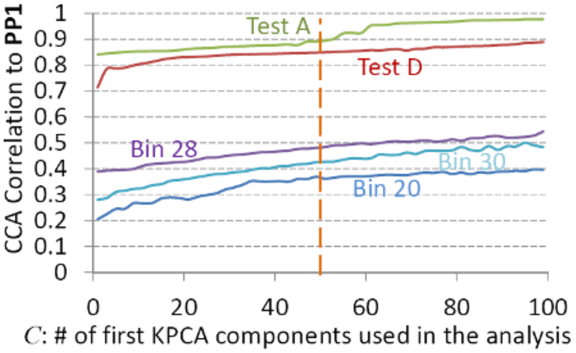


Fig. 20. **Kernel CCA risk evaluation on known results**

The x-axis shows the number C where the first C KPCA components are selected. Suppose X is an n dimensional vector $(X_1, \ldots, X_n)$. In the analysis, we expand X to X' that is an $n+C$ dimensional vector $(X_1, \ldots, X_n, PC_1, \ldots, PC_C)$ where each PC_i is a KPCA component. Hence, for $C = 0$, it is the same as the regular CCA. As we see in Fig. 20, as more KPCA components are used, the correlation become higher.

In Fig. 20, the separation between the correlated cases and uncorrelated cases is clear across all selections of C. We selected $C = 50$ to apply the kernel CCA to check if there is a dependence between all other types of fails and PP1.

Fig. 21 shows an example result of risk evaluation. In this example, we tried to evaluate the risk of adjusting PP1 by assessing the dependence between the result of a test and PP1. A test bin may comprise multiple tests. The figure shows the highest KCCA correlation found in each test bin. In each case, it shows the correlation based on CCA and then additional correlation based on the kernel CCA (with $C = 50$).

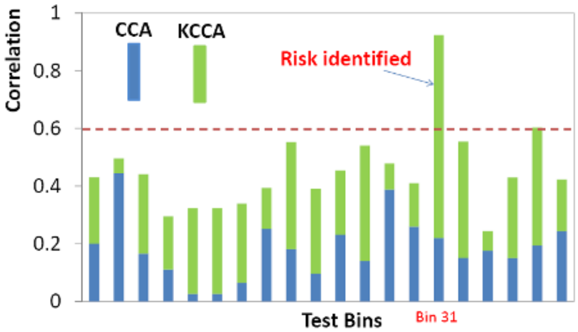


Fig. 21. **Risk evaluation with respect to adjusting parameter PP1**

For all cases, the CCA correlations are low. For all but bin 31, the KCCA correlations are also not high. However, for bin 31, its CCA correlation is very low but KCCA correlation is very high - indicating a strong non-linear dependence between this test in bin 31 and the process parameter PP1.

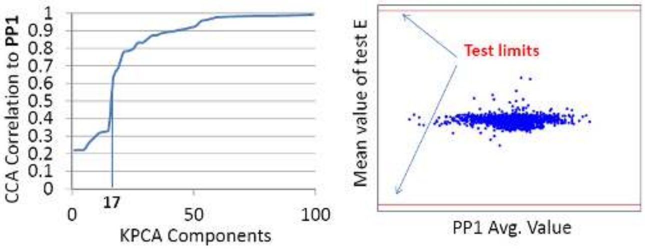


Fig. 22. **Detailed analysis of the test in bin 31 vs. PP1**

Fig. 22 provides more information on the test from bin 31. Call this test in bin 31 test E. The left shows a similar plot as that shown in Fig. 20. Observe that the KCCA correlation to PP1 increases significantly when the 17th KPCA component is included. The KCCA correlation increases to almost 1 as more components are added. This clearly indicates a strong non-linear correlation.

Since the dependence is non-linear, it is hard to visualize it. To contain the risk, the right plot shows a scatter plot where the y-axis is based on the average value of test E across each wafer. The plot shows that the distribution is not close to the test limits. Hence, even though adjusting PP1 may somehow affect test E result, the risk might not be high.

The risk with test E was presented to the product team for further evaluation. It was determined that the association between PP1 and the devices tested by test E was not high. In this case, the benefit of adjusting PP1 out-weighted the risk and hence, the adjustment was kept.

We applied the kernel CCA to assess the risk of adjusting other parameters PP2-PP5. A few other risky tests were found like that shown in Fig. 21. However, all risky tests were contained either by showing a large margin of the distribution to the test limits and/or by domain knowledge from the product team. Although risk evaluation did not invalidate any of the recommended changes, it was an essential step to sign-off the silicon experiment.

6. **Yield improvement based on silicon results**

After the risk evaluation, findings from Table III were all accepted to implement a split-lot run with multiple experimental lots. The five parameter changes were implemented as three process changes, one for PP5 (call it ADJ #1), another for PP2-PP4 (call it ADJ #2), and the third for PP1. In the split lot experiment, change for PP1 was applied across the board. A first set of wafers was based on applying only ADJ #1 (and PP1 adjustment). A second set of wafers was based on applying only ADJ #2 (and PP1 adjustment). A third set of wafers was based on applying both ADJ #1 and ADJ #2 (and PP1 adjustment). Of course, lots manufactured previously without any of the changes were used for comparison.

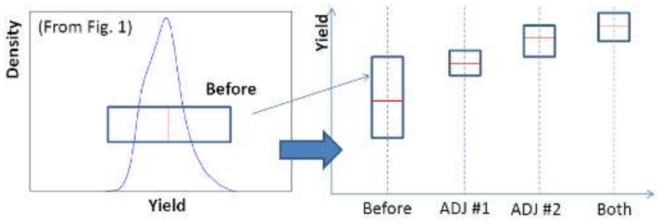


Fig. 23. **Silicon split-lot results show yield improvement**

Fig. 23 summarizes the result from the split-lot experiment. It can be clearly seen that ADJ #1 and ADJ #2 each uniquely contribute to the yield improvement and together achieve the best yield result. After the split-lot experiment and confirmation of the yield improvement, the process changes were accepted and applied in production.

7. **A brief review of prior work**

For production yield improvement, existing efforts may include approaches based on yield modeling, volume diagnosis, and/or root-causing. For example, earlier work such as [10] tried to identify the top parametric parameters that were most sensitive to yield and model their impact using multivariate regression. The work [11] used K-Means to cluster wafers into two groups, one with good yield and one with poor yield. Kruskall-Wallis and decision trees were applied to identify process parameters that most likely explained the discrepancy between the two groups.

Volume diagnosis is an effective approach for yield improvement [12][13][14]. For example, the work in [12] applied a novel statistical learning algorithm to produce accurate feature failure probabilities to better understand yield limiters. The work in [13] incorporated logic diagnosis data along with information on physical features in the layout to identify dominant defect mechanisms among failing dies.

For lithographic induced systematic issues, the work in [15] proposed methods to extract features and cluster layout snippets to identify possible defect hotspots.

Most of the tests analyzed in this work were parametric tests where root causing the failures could be difficult. Hence, the correlation approach was applied as an alternative to the root-causing effort. The goal of finding relevant process parameters to improve yield is similar to [11]. However, our analysis is much more detailed than that proposed in [11].

As discussed above, CCA and kernel CCA are well known statistical methods prior to this work. For CCA analyses, we used the Python CCA tool from scikit-learn [16]. Our subset discovery tool was built on top of the CCA tool. Our kernel CCA tool involved new script that wraps the CCA tool and kernel PCA tool [16]. Hence, CCA and kernel CCA themselves are not the novel aspects of this work. Instead, the novel aspects are in the overall methodology, including the subset canonical correlation, the implementation of kernel CCA with kernel PCA and in its application for risk evaluation.

8. **Conclusion**

This work presents a novel production yield optimization methodology based on three advanced statistical correlation methods: CCA, subset CCA and kernel CCA. We applied the methodology to optimize the production yield for an automotive product line. Silicon split-lot experiment confirms the effectiveness of our findings by showing significant yield improvement and significant reduction of the yield fluctuation. The silicon result demonstrates the added value by the proposed methodology to the existing yield optimization efforts carried out by the test, design and yield analysis teams.

REFERENCES

[1] A. Rényi. On measures of dependence. Acta Mathematica Acaddemiae Scientiarum Hungarica, Vol 10, Issue 3-4, pp. 441-451, 1959.

[2] J. Jacod and P. Protter. *Probability Essentials*. Springer, 2000.

[3] David R. Hardoon, Sandor Szedmak, John Shawe-Taylor. Canonical correlation analysis; An overview with application to learning methods *Neural Computation*, 16 (12), pp. 2639-2664, 2004.

[4] J. Shawe-Taylor, N. Cristianini, Kernel Methods for Pattern Analysis. *Cambridge University Press* 2004.

[5] F. Bach and M. Jordan. Kernel independent component analysis. *Journal of Machine Learning Research*, 3, pp. 1-48, 2002.

[6] Ingo Steinwart. On the Influence of the Kernel on the Consistency of Support Vector Machines. em Journal of Machine Learning Research, 2, pp. 67-93, 2001.

[7] A. Gretton, et al., Kernel Methods for Measuring Independence *Journal of Machine Learning Research*, 6, pp. 2075-2129, 2005.

[8] Malte Kuss and Thore Graepel. The Geometry Of Kernel Canonical Correlation Analysis. Max Planck Institute for Biological Cybernetics, Technical Report 108, May 2003.

[9] Schölkopf, B., A. Smola, and K.-R. Müller. Nonlinear component analysis as a kernel eigenvalue problem. *Neural Computation* 10, 1299-1319, 1998.

[10] Wong, A.Y A statistical parametric and probe yield analysis methodology *Defect and Fault Tolerance in VLSI Systems* pp.131-139, Nov. 1996.

[11] Chien, Chen-Fu, et al., Data Mining for Yield Enhancement in Semiconductor Manufacturing and an Empirical Study *Expert Syst. Appl.* 33, pp. 192-198, 2007.

[12] H. Tang, et al. Analyzing Volume Diagnosis Results with Statistical Learning for Yield Improvement. *IEEE ETS*, 2007

[13] Sharma, M, et al., Efficiently Performing Yield Enhancements by Identifying Dominant Physical Root Cause from Test Fail Data International Test Conference 2008.

[14] M. Sharma, et al. Determination of Dominant- Yield-Loss Mechanism with Volume Diagnosis IEEE D& T, 27 (3), 2010.

[15] Tam, Wing-Chiu, et al., Systematic defect identification through layout snippet clustering International Test Conference 2010

[16] http://scikit-learn.org/stable/