EUROPE DISCOVERY SUMMIT

Yield and Quality Issue Solving by Correlating Optical Inspection Step Results With Electrical Tests

Corinne Bergès, Shilu Zhang, Jim Bird, Chris Smith

NXP Semiconductors

Abstract

- In the semiconductor manufacturing industry for automotive, parts are tested at each manufacturing step to screen likely-to-fail parts. The further upstream the weak parts are scrapped, the lower the scrapping cost will be. But testing has a cost as well.
- A recent project at NXP sought to avoid a manual defect classification of the defects observed at the wafer inspection level. Defects are now classified as killer or not-killer from a training image dataset, and a failure probability is assessed for each die.
- JMP allows a further step in correlating this failure probability to electrical tests with three types of analysis.
  - The first analysis assessed a failure probability threshold to limit the number of parts tested to limit test cost.
  - The second analysis highlighted the tests most correlated with failure probability.
  - The final analysis used the list of highlighted tests to adjust test limits to screen the parts with failure probability outliers.
- The analyses limit test cost while increasing quality.

Die-criticality definition

- In the NXP semiconductor manufacturing industry for automotive, 100% of the parts are inspected by inspection tools, at each layer or step of their manufacturing. For each of the layers, from sampled defects observed in this inspection step and from SCAN images taken on these defects, a manual classification is performed in 3 classes:
  - Killer defects as they are most likely-to-lead to a failure for the parts on which this type of defects was observed;
  - Not-killer defects with no failure probability for the parts with this type of defects;
  - Lastly, nuisances when it appears that the observations are not a defect.
- Which is done on a sample of defects, may be extended to all the defects, with an automated classification that is going to use these first sampled defects and their manual classification to build a classification model that will classify automatically all the defects observed on all the dies in these 3 classes.
- Then, a failure probability per die, also called die-criticality, is computed from the count of these 3 types of defects per layer, assuming more or less impact on the failure probability from a killer defect (weight equal to 10 in the total failure probability), a not-killer defect (weight equal to 3) or a nuisance (null weight). Furthermore, the defects observed on one layer may be more impactful than other ones on another layer: that is to say, a weight can be applied individually for each layer.

Die-criticality formula

$$Failure\_probability\ per\ die(or\ Die\_criticality) = \sum_{layer\_1}^{layer\_n} \{weight\_layer_i * \{[count\_of\_the\_nuisances\_layer_i * 0] + [count\_of\_the\_not\_killer\_defects\_layer_i * 3] + [count\_of\_the\_killer\_defects\_layer_i * 10]\}\}$$

Formula 1: Die_Failure_Probability or Die_Criticality or IPAT (Inline Part AverageTest): 3 potential different names for a same definition

EUROPE
DISCOVERY SUMMIT

Yield and Quality Issue Solving by Correlating Optical Inspection Step Results With Electrical Tests

Upstream and downstream screening

- So after inspection and thanks to the classification model, 100% of the dies have an estimated failure probability. At this step, if a threshold is set on the failure probability value, the dies can be scrapped if their failure probability is higher than the failure probability threshold. This upstream screening may avoid the next test that is the electrical test at die level (unit probing or UP test), for the dies for which the failure probability is beyond the threshold: that means a cost and time saving in electrical UP test step.

- But, if the threshold is incorrectly set, this screening can become a mess in term of yield loss if it is set too low (a too high yield loss but potentially a more powerful screening of the likely-to-fail dies) or in term of quality if it is set too high (more dies pass even if their failure probability is high, which is fitting with a low yield loss, too). So, threshold setting needs to meet the best typical compromise between the lesser yield loss versus the best quality level.

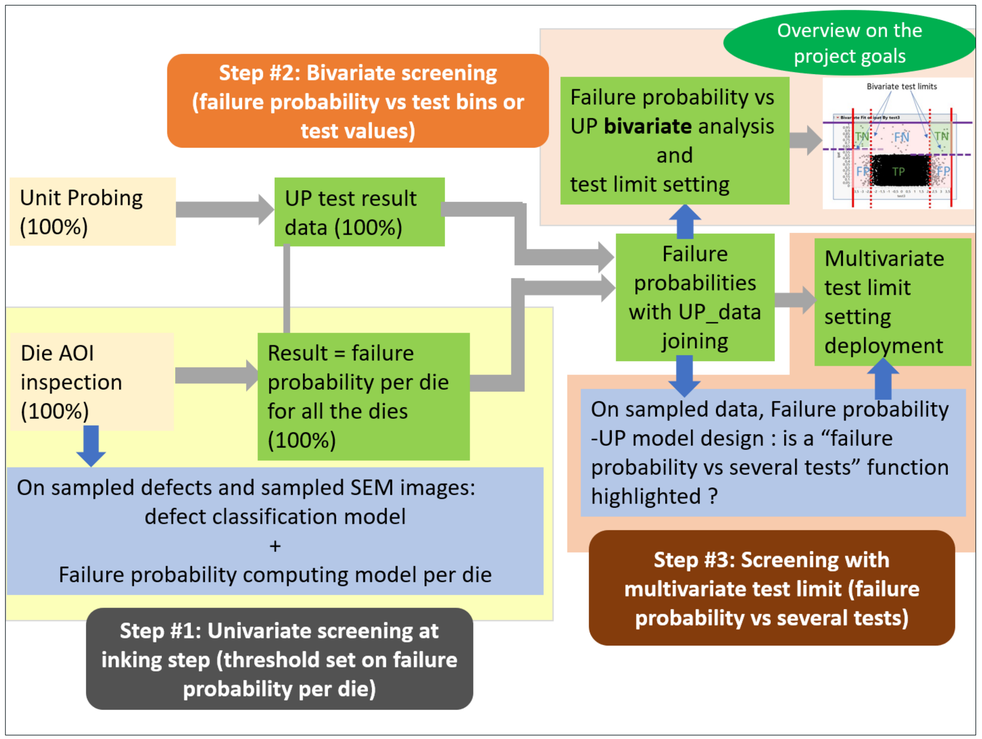


Fig 1: Scheme of the full project: data flow and analysis

Key question

- So, the dies that need to be screened out after inspection (upstream screening), are the dies which there is a high insurance that they will fail at UP test (downstream screening). A confusion matrix can be computed on the answer of the key question: can the failure probability predict the result of the UP test ? Prediction, accuracy and F1_score metrics are also computed and the searching for the best threshold will correspond to the searching for the best value of the chosen metric among these typical 3 ones.

- But UP test is not only dedicated to the screening of the defects, but it is also designed for detecting of every process shift. This may affect a little the setting of the failure probability threshold.

The three different analysis

- Type 1 analysis / Univariate analysis
  - The key question about the capability to predict the UP test result from the failure probability, is fitting from the first analysis: this analysis will use a training dataset of dies for which failure probability and UP test result is known. And the goal is to set the best threshold to meet the screening efficiency vs additional yield loss compromise. The dies with a failure probability beyond this threshold will be inked (scrapped) before their UP test.

- Type 2 analysis / Bivariate analysis
  - If the dies passed the inking step, the goal is still to improve the screening at UP test. This is possible from a failure probability vs UP bins, because a bin is fitting with one test failing in the bin number group of tests, or better, failure probability vs UP Test data correlation result. Test limits may be adjusted from this bivariate analysis result.

- Type 3 analysis / Multivariate analysis
  - Now, by designing a multivariate analysis between failure probability and all the UP tests, a multivariate limit may be implemented, in the extent that this capability can be implemented on the UP testers. Anyway, this type of analysis may highlight the key tests for which the limits may be strongly beneficially adjusted: in these terms, type 3 analysis appears as a preliminary analysis to perform before type 2 analysis, highlighting the key tests to use in the analysis 2, instead of doing it on all the tests.
  - The data analytics methods in multivariate, between die failure probability and test values, available in JMP, are the correlation analysis to highlight linear correlation, or machine learning algorithms, as decision trees, bootstrap forest or boosted trees for a good interpretability of the results, or neural networks or support vector machines when a model is looked for, with only a weak understanding of which tests are contributing the most in the model. Since a screening improvement is aimed by adjusting the limits of the tests the most correlated with die failure probability, only the methods providing a good interpretability will be used (decision trees, bootstrap forest or boosted trees).

EUROPE DISCOVERY SUMMIT

Yield and Quality Issue Solving by Correlating Optical Inspection Step Results With Electrical Tests

JMP usage and results _ Type 1 analysis / Univariate analysis

Analysis #1 is performed in three steps:

Step #1: Failure probability distribution and cumulative distribution:

- The training dataset is fitting with 9 lots of dies, tested at room temperature at unit probing: **'RoomTestData.jmp'**.
- The cumulative distribution of Failure Probabilities or Die Criticalities estimate, upper 95% and lower 95% values, is obtained as following:
  - Die criticality quantiles computed from the **'RoomTestData.jmp'** table by step of 0.001% (estimate, lower and upper at 95% confidence level): **'DieCriticality_Quantiles.jmp'**
  - From the **'DieCriticality_Quantiles.jmp'** table, plotting of the Cumulative distribution of Failure Probabilities estimate, upper 95% and lower 95% values.

Step #2: Confusion Matrix

- The training dataset is fitting with the same 9 lots of dies, tested at room and hot temperatures at unit probing: **'ConfusionMatrixTable.jmp'**
- Beyond the confusion matrix metrics, net-loss and gross-loss are computed to help in failure-probability threshold setting:
  - Net-loss is fitting with the additional yield due to the screening on failure-probability threshold, beyond the typical UP yield loss.
  - Gross-loss is the total yield loss due to the screening on failure-probability threshold
- Precision vs Recall: an improvement is expected on UP test, so the focus is not on the parts that were rejected (FN), but on the parts that passed (FP): so, Precision is a key metric, more important than Recall. A weighting F1_score could be used, with Beta = ½, to express that Precision should be twice more important than Recall.

Step #3: Failure probability threshold selection

- Precision is a key metrics. Unfortunately, a high precision can be linked to a high net-loss, which is not always possible: a compromise needs to be found between precision and net-loss.

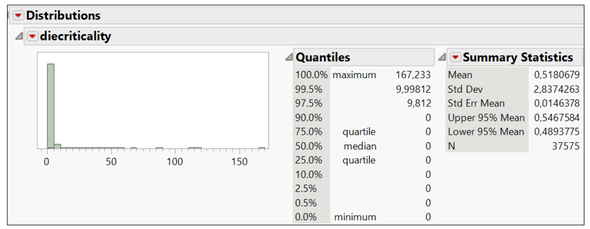


Fig 2: Die_Criticality distribution

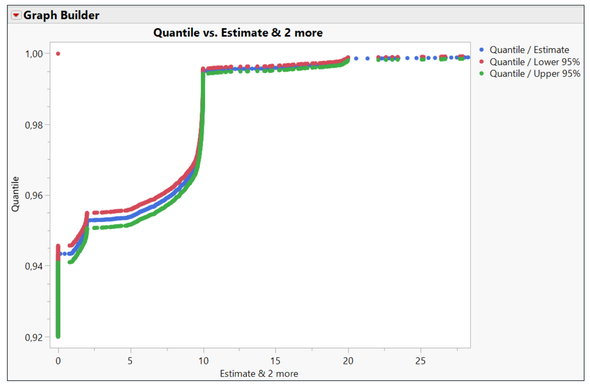


Fig 3: Cumulative distribution of Die_Criticality estimate, upper 95% and lower 95%

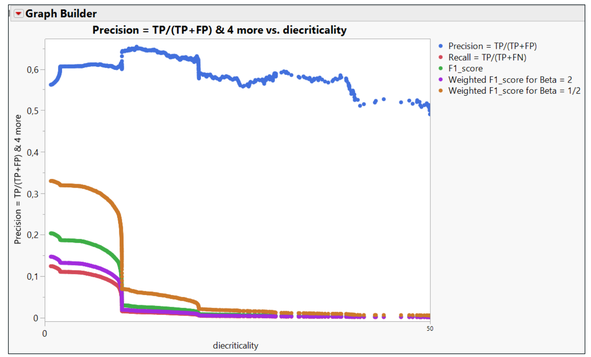


Fig 4: Confusion matrix: plotting of the typical metrics

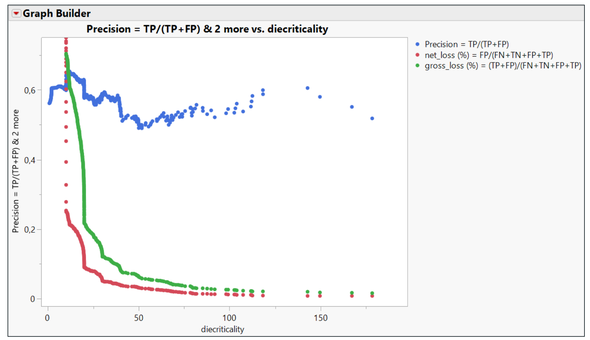


Fig 5: Precision vs yield loss (net or gross losses)

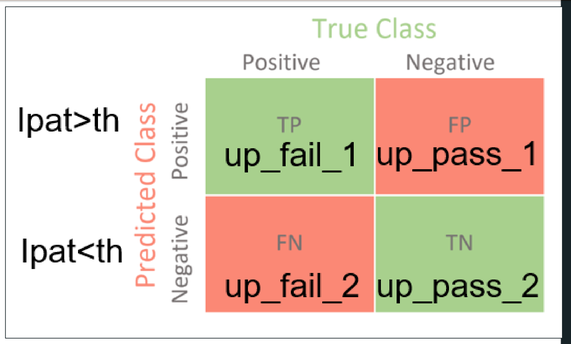


Fig 6: Confusion matrix: is diecriticality a good predictor of UP test results ?

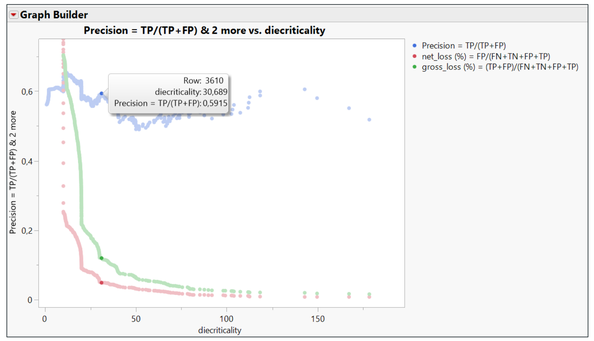


Fig 7: Precision, net_loss and gross_loss vs diecriticality

		TRUE CLASS	
		Positive	Negative
PREDICTED CLASS	Positive (Ipat>th)	126	87
	Negative (Ipat<th)	53016	124734

Fig 8: Confusion matrix (number of dies) estimated for a specific threshold choice where net_loss and gross_loss values are accepted by the business (respectively 0.05% and 0.1%)

failure probability threshold	net_loss (%)	gross_loss (%)
30,689	0,049	0,12

Fig 9: Gross and net loss values for the threshold chosen

Yield and Quality Issue Solving by Correlating Optical Inspection Step Results With Electrical Tests

Reminder: Confusion Matrix _ The metrics

- Precision

Out of all the positive predicted, what percentage is truly positive.
The precision value lies between 0 and 1.

- Recall

Out of the total positive, what percentage are predicted positive. It is the same as TPR (true positive rate).

- F1 Score

It is the harmonic mean of precision and recall. It takes both false positive and false negatives into account. Therefore, it performs well on an unbalanced dataset.
F1 score gives the same weightage to recall and precision.

- Weighted F1 Score

There is a weighted F1 score in which we can give different weightage to recall and precision. As discussed in the previous section, different problems give different weightage to recall and precision.
Beta represents how many times recall is more important than precision. If the recall is twice as important as precision, the value of Beta is 2.

TP: True Positive
FP: False Positive
TN: True Negative
FN: False Negative

$$Precision = \frac{TP}{TP + FP}$$

$$Recall = \frac{TP}{TP + FN}$$

$$F1\ score = \frac{2}{\frac{1}{Precision} + \frac{1}{Recall}} = \frac{2 * (Precision * Recall)}{(Precision + Recall)}$$

$$F_\beta = (1 + \beta^2) * \frac{(Precision * Recall)}{(\beta^2 * Precision) + Recall}$$

Formulas 2: Confusion matrix metrics

Additional definitions used in this project: gross_loss and net_loss

Net-loss is fitting with the additional yield due to the screening on failure-probability threshold, beyond the typical UP yield loss.

Gross-loss is the total yield loss due to the screening on failure-probability threshold.

$$Gross\_loss = ipat\_fail / total\_number\_of\_not\_inked\_parts = (TP+FP)/(FN+TN)$$
$$Net\_loss = up\_pass1 / total\_number\_of\_not\_inked\_parts = FP / (FN + TN)$$

Formulas 3: Gross loss and Net loss

EUROPE DISCOVERY SUMMIT

Yield and Quality Issue Solving by Correlating Optical Inspection Step Results With Electrical Tests

JMP usage and results _ Type 2_bivariate and type 3_multivariate analysis on simulated data

In order to design the analyzis and to be sure to cover any type of diecriticality vs UP test relationship, test and diecriticaity values are simulated and used in a first step.

Simulation method 1:
'curves_distinct_variability_percent.jmp'
Around twenty curves of test data are mathematically simulated from incremented failure-probability values to see if JMP succeeds to highlight the tests for which test limit adjustment would be more beneficial according to their correlation with die failure probability. As a conclusion, the study highlights the need to implement a transformation on the test data to focus only on the upper and lower test values and their correlation with die_failure probability.

Simulation method 2:
'curves_random_diecriticality.jmp'
Another simulation method is used, to be closer to the failure-probability shape.
A similar conclusion than with method 1 is obtained about the need for transformations. However, test 3 seems correctly highlighted by Bootstrap Forest and Boosted Tree analysis.

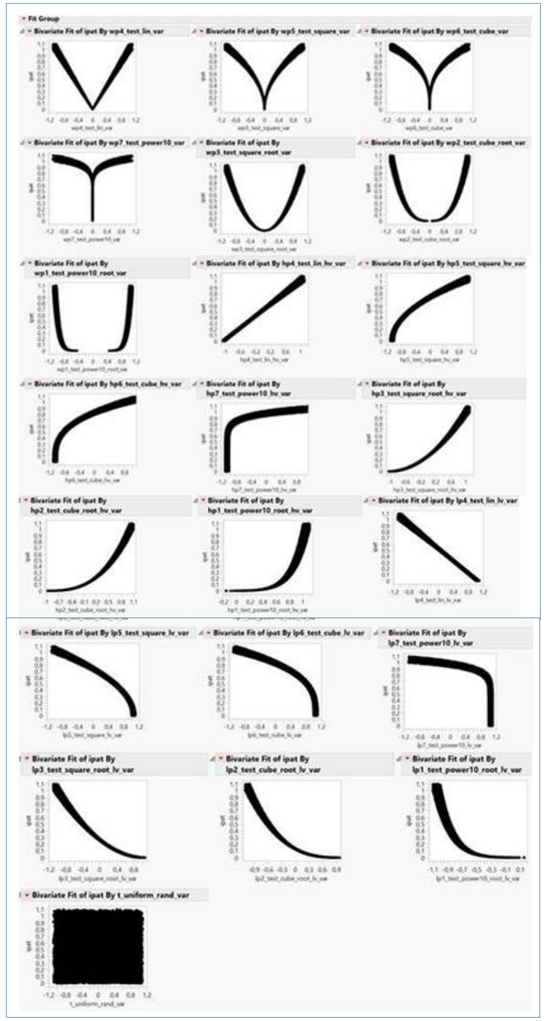


Fig 10: Around twenty diecriticality vs test data relations are mathematically simulated: which ones will JMP succeed to highlight as the most beneficial to have the test limits adjusted ?

Simulation method 1

Model Comparison

Predictors

Measures of Fit for ipat

Predictor	Creator	,2 ,4 ,6 ,8	RSquare	RASE	AAE	Freq
Pred Formula ipat	Fit Least Squares		0,9967	0,0174	0,0135	20000
ipat Predictor	Partition		0,7481	0,1524	0,1316	20000
ipat Predictor_1	Bootstrap Forest		0,9980	0,0137	0,0102	20000
ipat Predictor_2	Boosted Tree		0,9980	0,0136	0,0103	20000

Fig 11: Fit Least Squares, Partition, Bootstrap Forest and Boosted Tree analysis are the 4 platforms used in the type 3 analysis.

Source	LogWorth	PValue
lp4_test_lin_lv_var	191,627	0,00000
hp4_test_lin_hv_var	177,272	0,00000
hp5_test_square_hv_var	29,626	0,00000
lp5_test_square_lv_var	24,992	0,00000
lp3_test_square_root_lv_var	21,143	0,00000
hp3_test_square_root_hv_var	20,646	0,00000
hp6_test_cube_hv_var	8,040	0,00000
hp1_test_power10_root_hv_var	7,952	0,00000
lp6_test_cube_lv_var	6,693	0,00000
lp1_test_power10_root_lv_var	4,205	0,00006
wp6_test_cube_var	2,360	0,00436
wp3_test_square_root_var	2,195	0,00638
wp2_test_cube_root_var	2,031	0,00930
lp2_test_cube_root_lv_var	1,700	0,01996

Fig 12: Tests selected by Fit Least Squares jmp modeling for which a test limit adjustment according to ipat correlation, will be the most beneficial: actually, most of them are fitting only with upper or lower test values.

Details about simulations and curve names in **'curves_distinct_variability_percent.jmp'**:
1 random distribution
+
21 curves with names that describe their generating:
-mathematical functions: linear, square, cube, power10, square-root, cube-root, power10_root
-wp: whole part / lp: lower part of the curve / hp: upper part
-var: variability added on the mathematically generated data

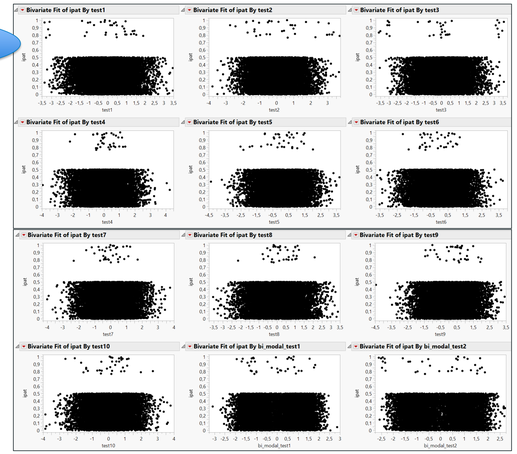


Fig 13: Diecriticality and test values randomly generated

Simulation method 2

Model Comparison

Predictors

Measures of Fit for ipat

Predictor	Creator	,2 ,4 ,6 ,8	RSquare	RASE	AAE	Freq
Pred Formula ipat	Fit Least Squares		0,0006	0,1490	0,1266	10000
ipat Predictor	Partition		0,0111	0,1482	0,1263	10000
ipat Predictor_1	Bootstrap Forest		0,2896	0,1256	0,1069	10000
ipat Predictor_2	Boosted Tree		0,0471	0,1455	0,1253	10000

Fig 14: Comparison between the 4 models used in the type 3 analysis.

Bootstrap Forest for ipat

Column Contributions

Term	Number of Splits	SS	Portion
test3	4165	4,01422185	0,1113
bi_modal_test2	4007	3,36754259	0,0933
test2	4108	3,16037764	0,0876
test1	4067	3,07462246	0,0852
test7	4210	2,99320994	0,0830
test9	4133	2,85744707	0,0792
test6	4145	2,83173531	0,0785
test5	3983	2,8066752	0,0778
test4	4028	2,76385541	0,0766
test8	4102	2,75970365	0,0765
bi_modal_test1	3997	2,73515678	0,0758
test10	3962	2,71202539	0,0752

Fig 15: Test3 is correctly selected to have its limits adjusted according to diecriticality threshold

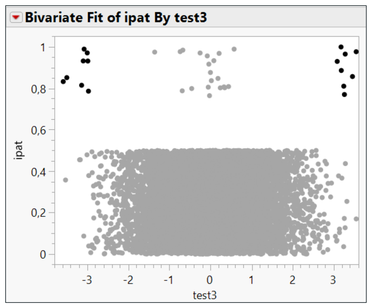
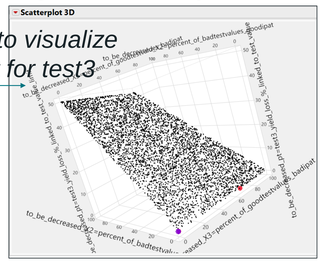


Fig 16: Bivariate analysis on test3 (type 2 analysis)

Fig 17: 3D scatterplot to visualize the best high-side limit for test3



Test limit adjustment aims to minimize the following three ratios (scatterplot 3D used in **'only_test3_valSUPtomedian220322.jmp'**):

- Yield loss generated by this new test limit (%)
- Percentage of not-rejected parts with higher IPAT scores
- Percentage of rejected parts with lower IPAT scores

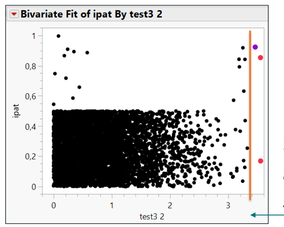


Fig 18: new test limit for high side test3

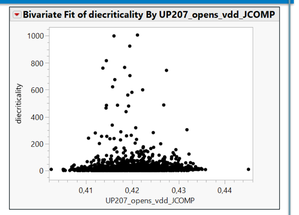
EUROPE DISCOVERY SUMMIT

Yield and Quality Issue Solving by Correlating Optical Inspection Step Results With Electrical Tests

JMP usage and results _ Type 2_bivariate analysis on real data

A ‘Failure probability vs UP test’ bivariate analysis is performed, for each test, in order to see if an adjustment of test limits may be beneficial:

- Example on real data: on ‘Diecriticality_vs_1test.jmp’, Fit Y by X of diecriticality by UP207_opens_vdd_JCOMP to be able to visualize how the test limits could be adjusted (search for an optimal method by a scatter 3D plot previously shown on simulated data: test3).



JMP usage and results _ Type 3_multivariate analysis on real data

Two multivariate analysis are interesting: the response is always diecriticality, but the factors may be:

- The hard bins as a first step: a bin is fitting with a group of UP tests per specific function; a bin highlighted by a model will indicate a group of tests highly correlated with diecriticality
- Or all the UP tests: the model will be able to detect the most correlated tests, not only a group of tests.

‘Failure probability vs UP bins’ analysis:

On **‘RoomTestData.jmp’** file, two analysis are performed to highlight the bins that show the highest values of diecriticality:

- Oneway Analysis of diecriticality By hbin_num
- Mean(diecriticality) vs hbin_num

‘Failure probability vs UP tests’ analysis:

The training dataset contains values for 28 tests (**‘Diecriticality_vs_28tests.jmp’**).

By a Fit Model analysis, 7 tests among the 28 ones are highlighted as the most contributing ones for Die_Failure_Probability, or as significant factors in the modeling of Die_Failure_Probability: a bivariate analysis between each of these tests and Die_Failure_Probability should be run, if it has not been already conducted in step 2 analysis, in order to adjust the test limits (Type 2 analysis rerun on the analysis 3 result: ‘Fit Y by X of diecriticality’ script in **‘Diecriticality_vs_28tests.jmp’**).

As previously shown on simulated data, beyond this typical ‘Fit Model’ analysis, other machine learning algorithms are available in JMP. In particular, the algorithms based on decision trees provide a good interpretability on the results:

-partition analysis (one decision tree)

-bootstrap forest or boosted trees (ensemble methods based on a forest of trees).

The key result of these analysis is the contribution from all the individual tests on die criticality, or a list of the tests the most correlated with the diecriticality, and each of these tests will be analyzed in a bivariate analysis with diecriticality to look for better test limits to take this correlation into account. In particular, a boosted tree analysis highlights 4 tests that explain 80% data variability (**‘boosted_trees_contribution.jmp’**: a bivariate ‘diecriticality vs test data’ analysis could be launched on these main tests, at least as a first step.

Support Vector Machine and Neural Network algorithms are also available in JMP to look for a model between diecriticality and test values, but the results will not be useful to highlight the tests for which an adjustment of the limits will be beneficial.

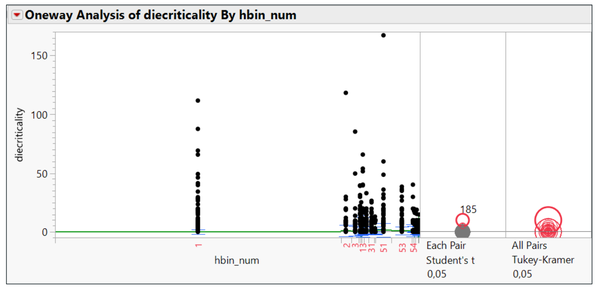


Fig 17: The bin #185 shows a different die_failure_probability in mean than the other bins (Each Pair Student’s t test)

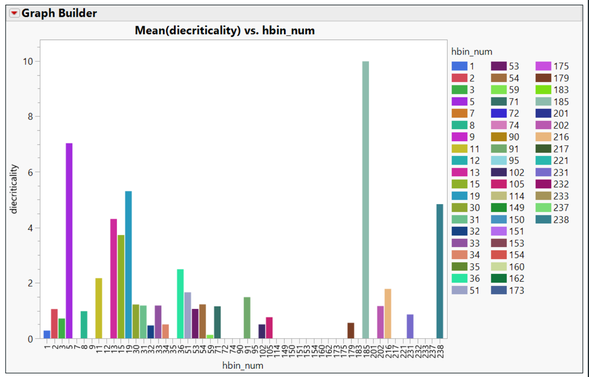
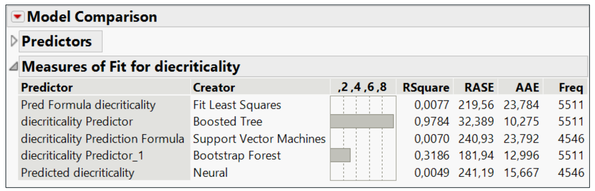
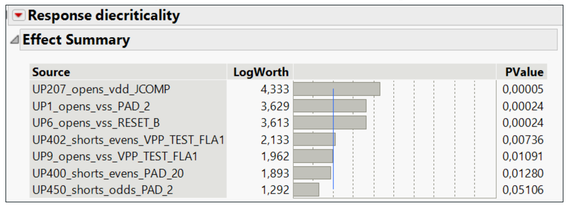


Fig 18: Die_Failure_Probability mean is higher for the bin #185 → A study should be launched on the UP tests fitting with this bin number



Predictor	Creator	,2 ,4 ,6 ,8	RSquare	RASE	AAE	Freq
Pred Formula diecriticality	Fit Least Squares		0,0077	219,56	23,784	5511
diecriticality Predictor	Boosted Tree		0,9784	32,389	10,275	5511
diecriticality Prediction Formula	Support Vector Machines		0,0070	240,93	23,792	4546
diecriticality Predictor_1	Bootstrap Forest		0,3186	181,94	12,996	5511
Predicted diecriticality	Neural		0,0049	241,19	15,667	4546

Fig 19: Model comparison on ‘Diecriticality_vs_28tests.jmp’



Source	LogWorth	PValue
UP207_opens_vdd_JCOMP	4,333	0,00005
UP1_opens_vss_PAD_2	3,629	0,00024
UP6_opens_vss_RESET_B	3,613	0,00024
UP402_shorts_evens_VPP_TEST_FLA1	2,133	0,00736
UP9_opens_vss_VPP_TEST_FLA1	1,962	0,01091
UP400_shorts_evens_PAD_20	1,893	0,01280
UP450_shorts_odds_PAD_2	1,292	0,05106

Fig 20: Tests highlighted by the Fit Model platform

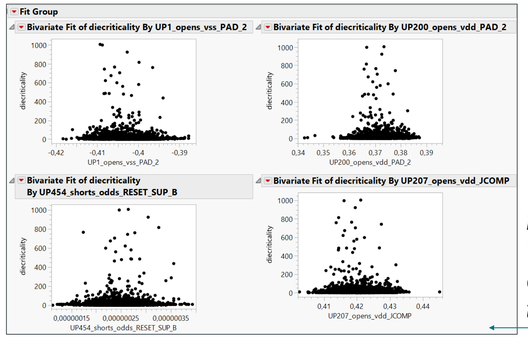
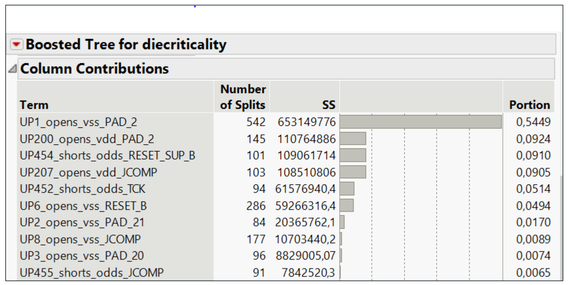


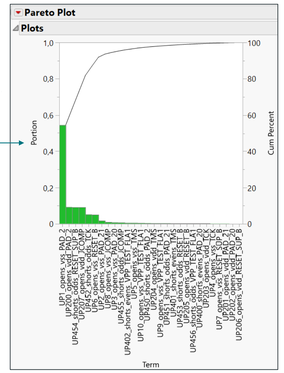
Fig 23: Bivariate ‘diecriticality vs test data’ analysis for 4 tests



Term	Number of Splits	SS	Portion
UP1_opens_vss_PAD_2	542	653149776	0,5449
UP200_opens_vdd_PAD_2	145	110764886	0,0924
UP454_shorts_odds_RESET_SUP_B	101	109061714	0,0910
UP207_opens_vdd_JCOMP	103	108510806	0,0905
UP452_shorts_odds_TCK	94	61576940,4	0,0514
UP6_opens_vss_RESET_B	286	59266316,4	0,0494
UP2_opens_vss_PAD_21	84	20365762,1	0,0170
UP8_opens_vss_JCOMP	177	10703440,2	0,0089
UP3_opens_vss_PAD_20	96	8829005,07	0,0074
UP455_shorts_odds_JCOMP	91	7842520,3	0,0065

Fig 21: Key contributors listed by Boosted Tree model

Fig 22: Pareto of the contributions by test (Boosted Tree model)



EUROPE DISCOVERY SUMMIT

Yield and Quality Issue Solving by Correlating Optical Inspection Step Results With Electrical Tests

Conclusion

A project between NXP and KLA is aiming to improve screening of the dies at wafer-inspection and UP steps, in order to increase quality without adding a too high yield loss.
The first component of this project is constituted by an image classification model, designed by KLA, that is going to classify the defect images as killer-defect, not-killer-defect and not-a-defect. Then, from this classification and from a weight that is attributed per class (10 for a killer defect, 3 for a not-killer one, and 0 for a nuisance), it is possible to compute a failure probability per die, that takes into account, for each die, the quantity of defects per class and per inspected layer.
The second component of the NXP-KLA project starts from this failure probability per die, that is also called die-criticality, and looks for a correlation between die-criticality and UP test results: in the extent that die-criticality may correctly predict the UP results, UP test step is skipped for the dies with the highest die-criticality values which are the most-likely-to-fail at UP. The key question is the adjustment of the die-criticality threshold between a too high yield loss and a too low quality level. More downstream, for dies with die-criticality smaller than the threshold, a correlation between die-criticality and UP test values can be revealed, and test limits may be consequently adjusted.

JMP is the main tool used to set the die-criticality threshold and to adjust test limits for the tests which an adjustment is beneficial for. This poster presented the key concerns faced in these analysis and how JMP helped to solve them. Other points could have been presented, too, as missing data management, test collinearity, variability in the matching between hard bins and the test groups,...

A key subject on this project is also the interface that needs to be provided to the project-users. Indeed, these analysis are not performed only once, but are re-run as soon as KLA updates its image classification models and as soon as new dies are inspected and classified. An engineer may be continuously led to adjust die-criticality threshold or test limits. The analysis need to be automated and piloted through a friendly interface.

A complex connecting to the NXP inspection and UP databases, very large data volumes, need for automation and interfaces, is conducting NXP to use many different platforms in parallel of JMP (Dataiku, Python, RStudio, RShiny, H2O, …), but JMP stays more interesting to quickly design an analysis, validate assumptions, estimate a first threshold value on samples, ….

Reference

Corinne Bergès, Jim Bird, Mehul D. Shroff, Edwin Lumanauw, Sreerag Raghunathan, Chris Smith, , *'Inspection methodologies and machine-learning approaches for defectivity data in semiconductor industry for automotive applications: case study for field-failure prevention'*, IPFA2022

Acknowledgements

The authors thank the leader, Onder Anilturk, and all the team members of this key project for NXP, for involving them in their activities and sharing their knowledge, thus allowing the development of machine learning in a new domain, i.e., automated optical inspection.
Many thanks also to project team members from KLA company for their expertise in image classification and their explanations about their models, which strongly helps NXP to build the analysis that use their model results.

Thank you

Thank you for your attention

Questions / Contact

Contact: Corinne Bergès corinne.berges@nxp.com