Using an Augmented Covering Array to Test a New JMP Platform
Michael Crotty
JMP Division, SAS Institute, Cary, NC 27513

A covering array enables a software tester to efficiently test interactions between components of a software system. Due to external considerations, sometimes specific levels of interactions must be included in a test plan. In this poster, a test plan is developed for a new analysis platform in JMP12. The test plan starts with a design based on a 2-covering array and then incorporates additional runs that need to be tested regardless of the design. The new Covering Array platform in JMP® Pro 12 gives efficiency metrics for various levels of coverage, even when a design has been augmented manually. These metrics are presented and discussed.

Primary Objective
To test a new analytical platform in JMP®, a good test plan is needed to efficiently cover the space of possible test cases. An analysis platform in JMP has many possible inputs:
- various types of data
- specification of roles for the data columns
- model fits and fitting options
- output options
- many more

We focus on partitioning the input space of model fitting options for the Destructive Degradation platform (new in JMP 12).

Factors and Levels
To construct a covering array test design, we need to define factors and levels. For the model fitting options in Destructive Degradation, some care is needed to partition the model fitting options to avoid having too many test cases.

For this problem, the fitting options are a logical place to start to partition the input space of the analytic platform. There are four clear inputs for model fitting:
- Y (response) transformation (3 levels).
- Time transformation (3 levels).
- Response distribution (8 levels).
- Path definition (13 levels).

A two-covering design for the above factors and levels would result in a minimum of 8x13=104 runs, so we use equivalence partitioning to partition the factors with 8 and 13 levels. The 8 levels for Response distribution are partitioned into 5 equivalence classes; the 13 levels for Path definition are partitioned into 4 equivalence classes.

This reduces the minimum number of runs for a two-covering design from 104 down to 5x4=20! The Covering Array platform constructs a two-covering design for all four factors that achieves this minimum of 20 runs. Now, we have a more manageable test plan.

But, there are some specific four-factor runs that are required in the test plan. We add these incrementally to our original 20 test runs. The Covering Array module in JMP Pro allows us to check levels of coverage after any amount of design augmentation.

Specification for test runs from development:
- I would suggest covering following selected case combinations:
  1. Normal, Linear, Linear, all path
  2. Lognormal, Linear, Linear, all path
  3. Normal, Log10, Log10, all path
  4. Lognormal, Log10, Log10, all path
- I suggest covering all 13 paths for (1); paths can be selective for (2), (3), (4), after we resolve all issues regarding path discovered in (1).

Note: A complete four-covering design would result in 3x3x8x13=936 runs, so we only want to include the specified four-factor test runs. Even considering the partitioned design, a complete four-covering design would be 3x3x5x4=180 runs.

Definitions
Coverage
The ratio of the number of distinct t-factor settings that appear in the design to the total possible number of t-factor settings, expressed as a percent. A t-coverage of 100% indicates that all possible t-factor settings are covered by the design. Note that each t-factor setting can appear multiple times.

Diversity
The ratio of the number of distinct t-factor settings in the design to the total number of occurrences of t-factor settings in the design, expressed as a percentage. The t-diversity measures how well the design avoids replication. A t-diversity of 100% indicates that no t-factor settings are repeated. A t-diversity of 50% indicates that the average number of times that distinct t-factor settings appear is two.

Implementation Note
The metrics above show less than 100% two-coverage for the original 20 run design. The first 20 runs shown to the left are missing the Log10/Log10/Rate combination. The original 20 run two-covering design specified a Log10 Time transformation for run 12. It was later changed to Linear so an existing test with the settings Linear/Linear/Weibull/Rate could be reused. The change could be made while keeping 100% two-coverage after runs 23-25 were added. (Run 25 replaced the missing Log10/Log10/Rate combination.)

Secondary Objective
Before we start implementing a particular test case, we need to select the original levels from the equivalence classes for Response distribution and Path definition that we will test. For instance, a particular test case for run 5 will use either a Lognormal or Loglogistic Response distribution, and it will use one of the first two Path definitions. We would like to calculate coverage for the augmented test plan above, but in terms of the original levels of the four factors. Ultimately, we are concerned with implementing the augmented covering array test plan shown above, but we are also concerned with maximizing coverage across the original levels of the four factors. We can again use the Covering Array module in JMP Pro for this goal.

To the right, the specific implemented test runs are enumerated. Metrics for two- and three-coverage are given in the table beneath the runs. Comparing the improvement of these coverage metrics as new test cases are implemented provides a measure of progress toward the goal of maximizing coverage of the overall factor space.

For some equivalence classes, implementing all levels within the class can be done at little additional cost. In these cases, including some or all levels within the equivalence class in the design can improve coverage metrics substantially.

Note: The Covering Array module in JMP Pro will only provide meaningful metrics for this secondary objective if all levels of the original factors are represented. In other words, all 8 levels of Response distribution and all 13 levels of Path definition must appear in at least one run.

Acknowledgements
Thanks to Joseph Morgan for guidance on setting up the covering array and many discussions of test planning and equivalence partitioning of factor levels over the years. Thanks to Peng Liu for development of the destructive degradation platform and guidance in testing strategy.

Reference