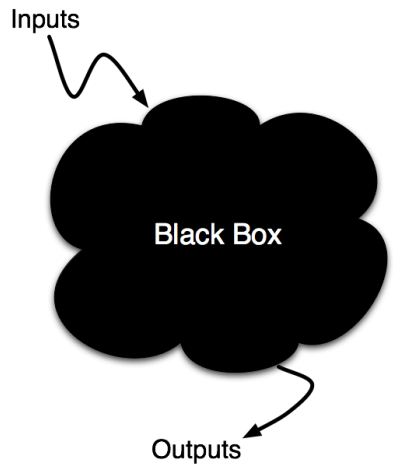
National Aeronautics and Space Administration

Model-Based Validation Testing

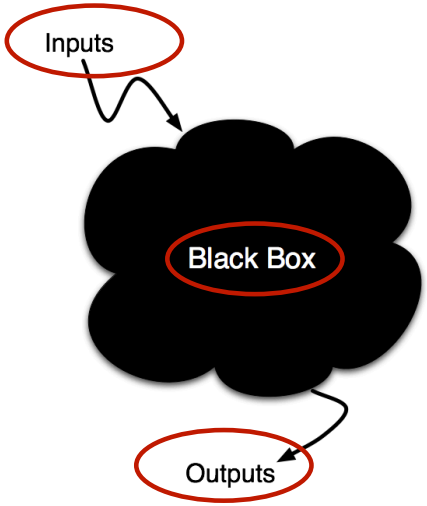
Dr. Misty D. Davies
Research Computer Engineer
NASA Ames Research Center

NASA/FAA/JPDO V&V Meeting
September 29, 2011
FAA Tech Center, NJ

www.nasa.gov



What if you have to make decisions about a "black box"?



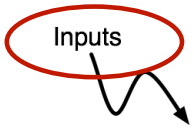
What if you have to make decisions about a "black box"?

How do we define these things?

What question would you like to answer? Some examples:

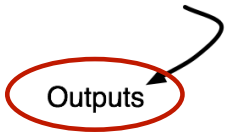
- *How robust is this CD&R algorithm to increased traffic and/or conformance uncertainty?*
- *How long does an emergency change in a scheduling algorithm affect traffic?*

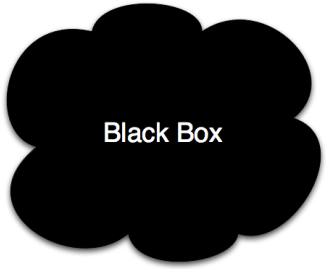
What if you have to make decisions about a "black box"?



The question can usually be phrased as: "As this input changes, how does the output change?"

•*How robust is this CD&R algorithm to increased traffic and/or conformance uncertainty?* ***Inputs:*** # of aircraft, distance from radar track hit or predicted trajectory, ***Outputs:*** LOS vs. predicted LOS (using real data), alarm rate as a function of average distance from nominal interior route

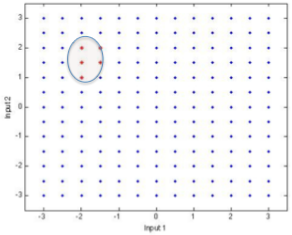




The system or systems-in-combination you need in order to answer your question.

Some examples:

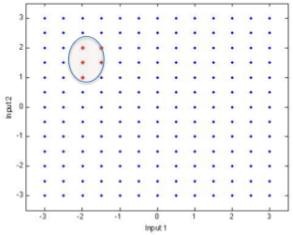
- *Trajectory synthesizers*
- *Schedulers*
- *Conflict prediction*
- *Conflict resolution*
- *Traffic flow management*

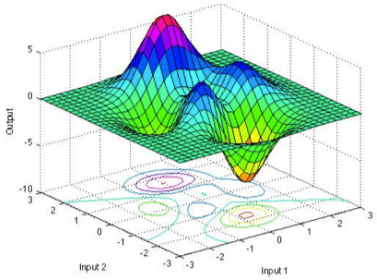


Standard testing casts a grid across the nominal input space, looking for failures.

Most of the space in traditional testing remains uncovered—you know very little about:

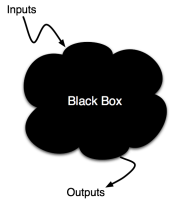
- how likely failures are between data points,
- what the space might look like beyond the nominal input spaces,
- if there is a correlation between your failures and the values of a particular set of variables.





Instead, we build models between the input and output spaces. This means that we can make predictions about where to test in order to find failures, and it can give us a probability for whether or not there are failures in our untested space. It also lets us find patterns in high-dimensional, complex spaces.

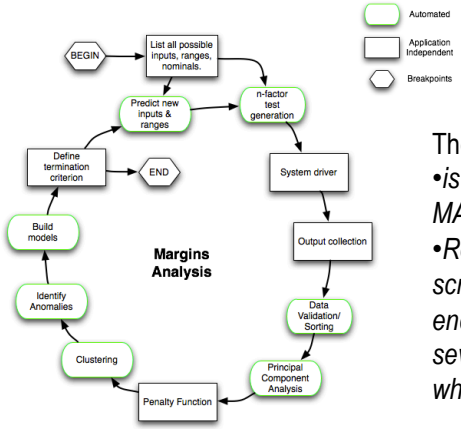
Use n-factor combinatorial testing, replication, and data understanding to create the test suite.



Treat testing as an experiment.

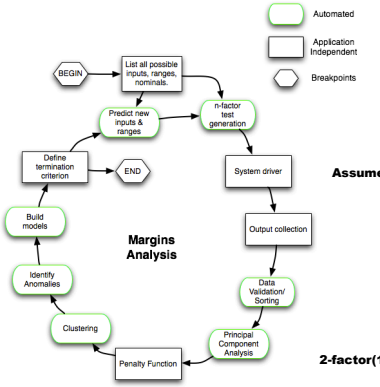
Build a model of the black box, and use statistical techniques to compare the model to the actual system. Use the model to predict system behavior. Refine the model.

Use component analysis techniques to find dependencies in the outputs. Seek anomalies in addition to failures in meeting objectives.



The **Margins Analysis**:

- *is a set of well-documented MATLAB tools for validation testing*
- *Requires application-specific scripts. (We've written these for end-users, and can also supply several examples for researchers who want to write their own.)*



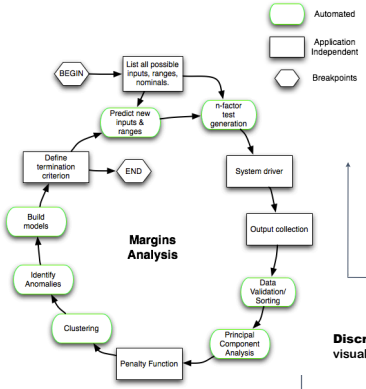
n-factor test generation:

Assume 20 binary values (about 1 million tests)

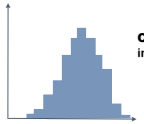
1	1	0	1	0	0	0	1	0	0	0	1	1	1	0	1	1	1	1	0
0	1	1	0	1	1	1	0	1	1	1	0	0	0	1	0	0	0	0	1
1	0	1	1	1	0	1	1	1	0	1	1	0	1	1	1	0	1	0	0
0	0	0	0	0	1	0	0	0	1	0	0	1	0	0	1	0	1	1	1
0	1	0	1	1	0	0	0	1	0	0	0	0	1	0	0	0	1	1	1
1	0	1	0	0	0	0	1	0	1	1	1	1	0	1	1	1	0	0	0
1	1	0	1	0	1	1	1	0	0	0	0	0	0	1	1	1	1	1	1
0	0	1	1	1	1	1	1	0	1	0	1	1	1	0	0	0	0	0	0
1	1	0	0	0	0	0	0	1	0	1	1	1	1	0	1	1	1	1	0
1	0	1	1	1	1	0	1	0	1	0	0	0	0	0	0	1	1	1	0
0	1	0	1	0	0	1	1	0	0	0	1	1	0	1	1	0	0	0	1

2-factor(11 tests)

3-factor(26 tests)

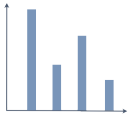


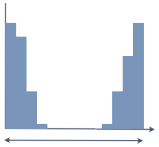
Data validation/sorting:



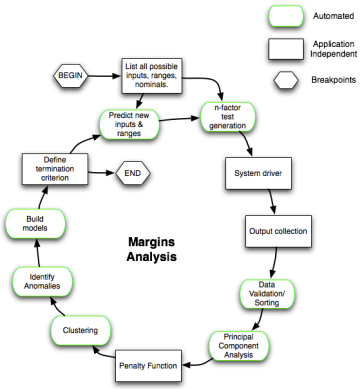
Continuous: how far is the plane from the nominal interior route?

Discrete: is the plane operating under visual or instrument flight rules?

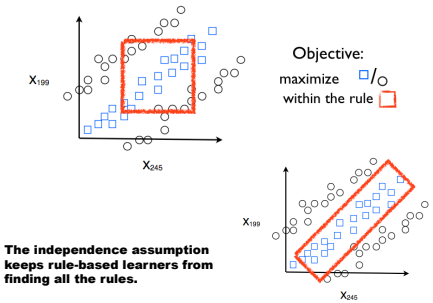




Periodic: what is the plane's heading?

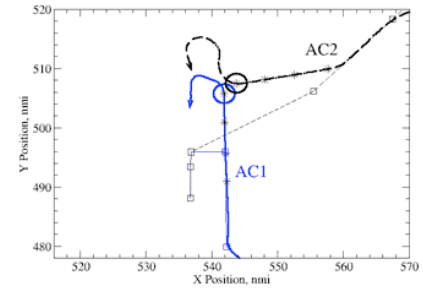


Principal Component Analysis:



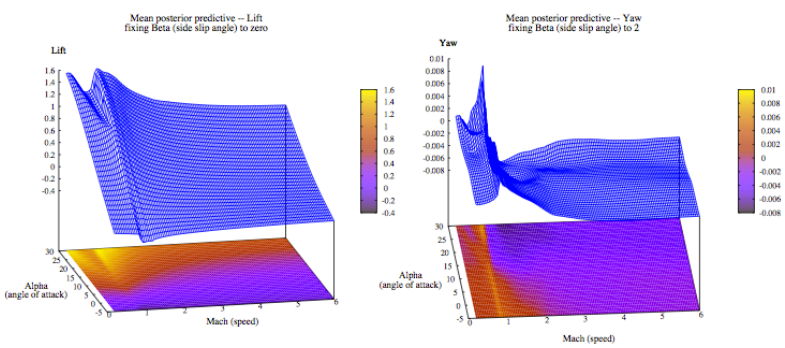
The independence assumption keeps rule-based learners from finding all the rules.

Terminal TSAFE

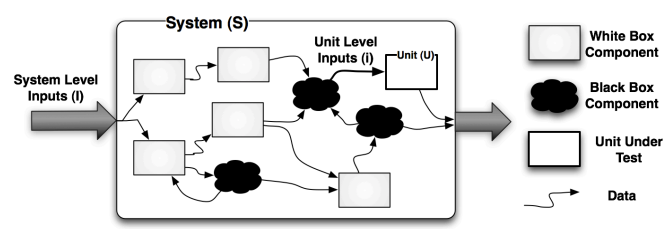


This figure represents a sample operational error. It shows several minutes of track data for two aircraft (AC1 and AC2) leading up to the loss of separation (LOS) where the two circles, which are 3 nautical miles in diameter, overlap. The asterisks are minute markers denoting three minutes before the LOS. The sharp turn to the right of AC2 after LOS is due to controller intervention.

How confident can we be in the models we are building?



Surfaces predicting the values of yaw and lift that were built by Bayesian statistical emulation for the Langley Glide-Back Booster. Note that the discontinuity in the surfaces as the booster crosses into the supersonic regime has been captured by the model. (Images courtesy Dr. Herbert Lee, UC Santa Cruz)

Idea: Combine heuristic testing and formal techniques

- Explicit techniques give (some) guarantees
- Heuristic techniques provide scalability