

# Input Attribution for Statistical Model Checking

## How do I increase the probability of my mission's success?

### Input Attribution – The “Why” of SMC

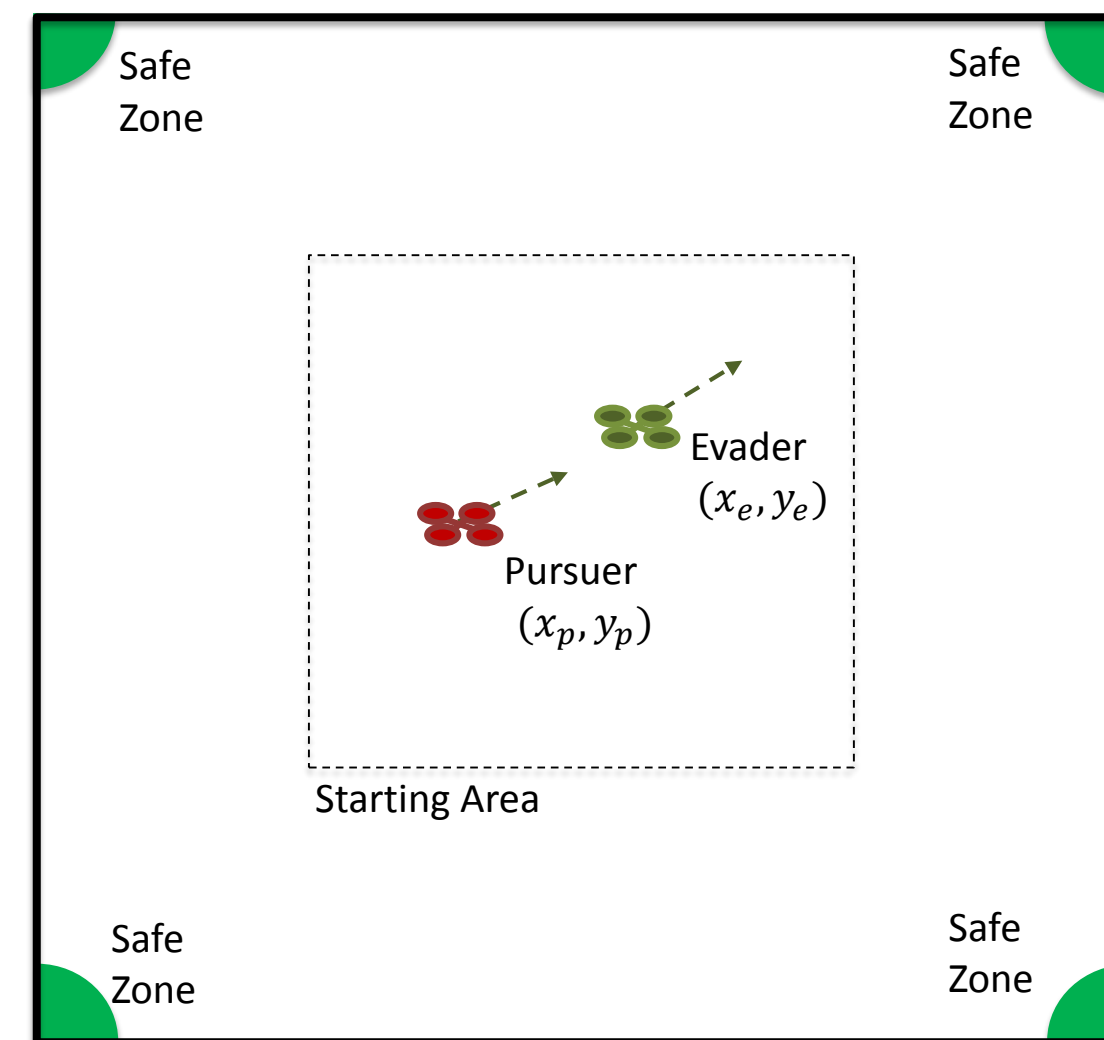
Statistical Model Checking (SMC) provides an estimate on the probability  $P[\mathcal{M} \models \Phi]$  that a predicate  $\Phi$  in a model  $\mathcal{M}$  is satisfied, but does not address why a particular result was obtained. The goal of Input Attribution (IA) is to use machine learning techniques to synthesize an explanation for an SMC result in terms of the inputs. IA for SMC can be thought of as analogous to the counter-example in traditional model checking.

A good Input Attribution has the following properties:

1. Describes relationship that actually exists in data
2. Is presented in a way that is quantitative and understandable
3. Gives investigator new insights
4. Is resilient to randomness in the system

### Example Scenario – Pursuer-Evader

Let  $(x_p, y_p)$  and  $(x_e, y_e)$  be random initial positions for a pursuer and an evader, respectively. The goal of the evader is to make it to one of several designated safe zones before it is caught by the pursuer. The SMC problem is to calculate the probability that the evader will escape. Intuitively, the probability of escape for the evader will depend on the initial distance between the pursuer and the evader, but can we synthesize this relationship purely from the SMC trials?



Example Scenario – Pursuer/Evader

### Approach – Logistic Regression

Logistic Regression (LR) is a regression model with a Boolean response variable based on the logistic function. A model  $L: \vec{x} \rightarrow \mathcal{R}$  is generated from a set of input vectors  $\vec{x}_i$  and corresponding Boolean responses  $\phi_i$ .  $L(\vec{x}_i)$  represents the log of the “odds” that the response  $\phi_i$  is 1. The logistic function maps the log odds to a probability. The LR model is a linear function of the input variables with the form:

$$L: \vec{x} \rightarrow \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \dots + \beta_N x_N$$

Each coefficient  $\beta_j$  represents the factor by which the logit (log odds) of  $\mathcal{M} \models \Phi$  increases for each unit increase of  $x_j$ . However, not all input variables may be statistically significant. When calculating each coefficient  $\beta_j$ , a standard error  $se(\beta_j)$  that can be used to calculate a “p-value” indicating the significance of each coefficient is also produced. P-values greater than about 0.05 indicate that a particular input variable is not significant. The generated input attribution is formed from the  $\beta_j$  terms that are considered statistically significant.

### Non-Linear Input Attribution

By expanding the Logistic model to include second order polynomial terms as:

$$L: \{\forall j: x_j, x_j^2\} \cup \{\forall j, k: x_j x_k\} \rightarrow \mathcal{R}$$

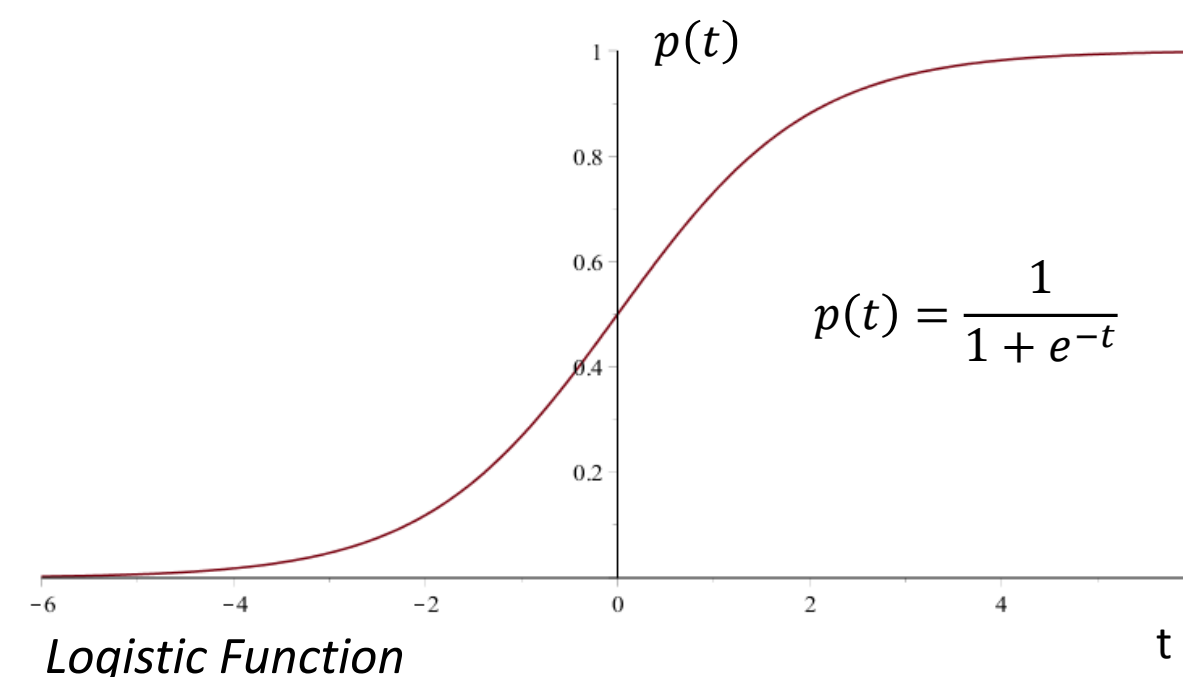
It is possible to discover more complex relationships among the input variables. After filtering terms that are not statistically significant, approximate factoring can be applied to pairs of terms to present the result in more human-readable form.

### Validation

Even though LR analysis may indicate statistical significance on one or more variables, the overall model must have a good fit to the data before an input attribution can be accepted. We use the AUC (Area Under Curve) of an ROC (Radar Operating Characteristic) analysis as a metric. Five-fold cross validation is performed and the average AUC is used. AUC represents the probability  $P[L(x_{SAT}) > L(x_{UNSAT})]$  where  $x_{SAT}$  is an arbitrary satisfying input ( $\phi = 1$ ) and  $x_{UNSAT}$  is an arbitrary unsatisfying input ( $\phi = 0$ ). An AUC of 0.5 indicates the model is no better than guessing, while an AUC of 1.0 is a perfect model.

### Experimental Results

We conducted SMC trials of the pursuer/evader scenario shown above using the V-REP simulation environment. Trials were conducted on a set of six 20-core blade servers. A target relative error of 0.01 was used which resulted in 39,960 trials. The resulting “mission success” probability for the evader was 0.214. LR analysis and input attribution was conducted using the R system and resulted in the expression shown to the right and resulted in an AUC of 0.77.



Logistic Function

$$t = \dots + 1.01x_p^2 - 2.03x_e x_p + 1.02x_e^2 + \dots$$

↓ Look for approximate factorings

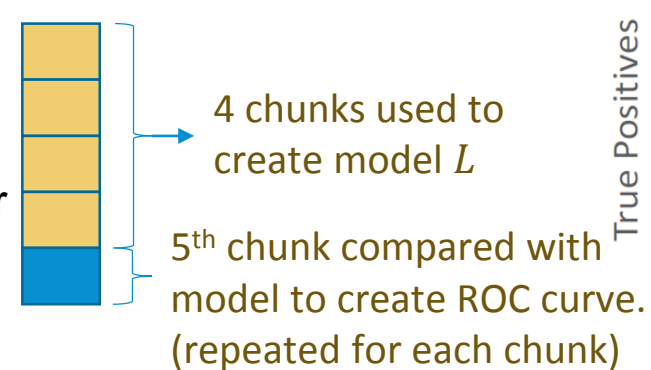
$$t = \dots + 1.01(x_p - 1.01x_e)^2 + \dots$$

↓ Accept approximation if error is small.

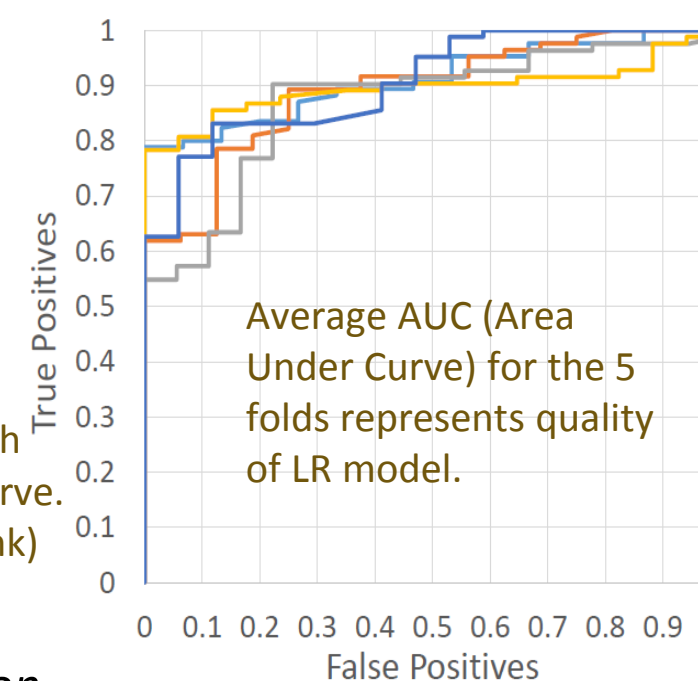
$$t = \dots + 1.01x_p^2 - 2.04x_e x_p + 1.03x_e^2 + \dots$$

Approximate Factoring

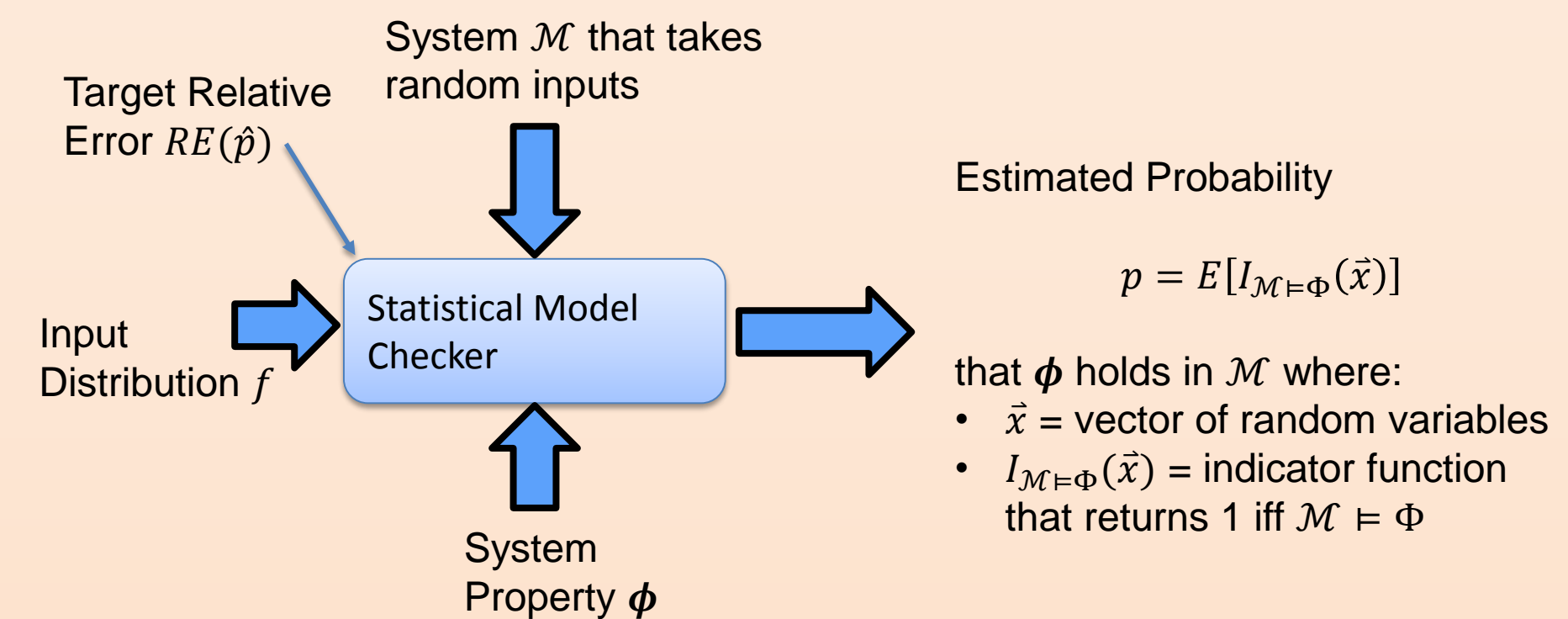
Simulation data with input and predicate results  $x_i, \phi_i$  partitioned into 5 chunks



5-Fold Cross Validation



### Statistical Model Checking (SMC) Basics



### Relative Error

Measure of accuracy for a prediction

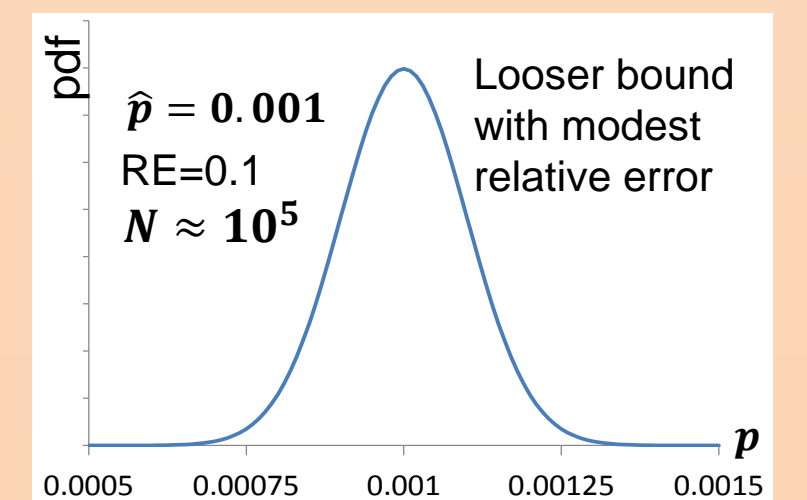
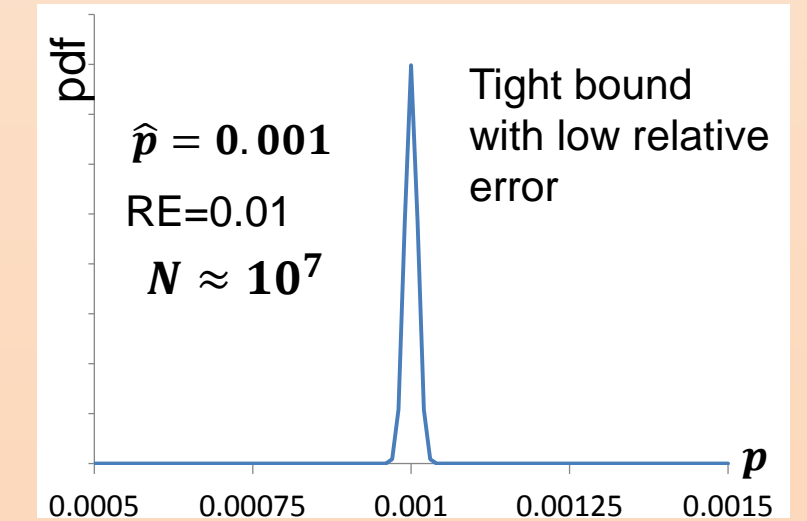
Defined as ratio of standard deviation to mean. For a probability estimate, the estimated relative error is:

$$\widehat{RE} = \frac{\hat{\sigma}}{\hat{p}}$$

Number of samples to achieve a target relative error increases

- as target relative error decreases, or
- as estimated probability decreases

$$N \approx \frac{1}{p(RE)^2}$$



Name	$\beta$	se( $\beta$ )	p-Value
$x_e x_p$	-0.124	0.0027	$< 10^{-4}$
$y_e y_p$	-0.122	0.0027	$< 10^{-4}$
$x_e^2$	0.060	0.0031	$< 10^{-4}$
$y_e^2$	0.056	0.0031	$< 10^{-4}$
$x_p^2$	0.056	0.0031	$< 10^{-4}$
$y_p^2$	0.056	0.0031	$< 10^{-4}$

Input Attribution Results

$$0.0602(x_e - 1.03x_p)^2 + 0.0561(y_e - 1.09y_p)^2$$

Factored Input Attribution

### Conclusion

We applied SMC with Input Attribution to a pursuer/evader scenario. Intuitively we expected an Input Attribution indicating that increased initial distance between pursuer and evader should be correlated with improved chance of escape for the evader. Using our LR-based approach, we synthesized a factored input attribution expression very similar to the Euclidian distance between the pursuer and evader coordinates, purely from the SMC trial data.

Copyright 2016 Carnegie Mellon University

This material is based upon work funded and supported by the Department of Defense under Contract No. FA8721-05-C-0003 with Carnegie Mellon University for the operation of the Software Engineering Institute, a federally funded research and development center.

NO WARRANTY. THIS CARNEGIE MELLON UNIVERSITY AND SOFTWARE ENGINEERING INSTITUTE MATERIAL IS FURNISHED ON AN "AS-IS" BASIS. CARNEGIE MELLON UNIVERSITY MAKES NO WARRANTIES OF ANY KIND, EITHER EXPRESSED OR IMPLIED, AS TO ANY MATTER INCLUDING, BUT NOT LIMITED TO, WARRANTY OF FITNESS FOR PURPOSE OR MERCHANTABILITY, EXCLUSIVITY, OR RESULTS OBTAINED FROM USE OF THE MATERIAL. CARNEGIE MELLON UNIVERSITY DOES NOT MAKE ANY WARRANTY OF ANY KIND WITH RESPECT TO FREEDOM FROM PATENT, TRADEMARK, OR COPYRIGHT INFRINGEMENT.

[Distribution Statement A] This material has been approved for public release and unlimited distribution. Please see Copyright notice for non-US Government use and distribution.

DM-0003832