Experimental Design for Simulation

[Law, Ch. 12][Sanchez et al.1]

Peter J. Haas

CS 590M: Simulation
Spring Semester 2020


Overview

Goal: Understand the behavior of your simulation model

▶ Gain general understanding (today’s focus)
▶ What factors are important?
▶ What choices of controllable factors are robust to uncontrollable factors?
▶ Which choice of controllable factors optimizes some performance measure?

Overview, Continued

Challenge: Exploring the parameter space

▶ Ex: 100 parameters, each “high” or “low”
▶ Number of combinations to simulate: $2^{100} \approx 10^{30}$
▶ Say each simulation consists of one floating point operation(!)
▶ Use world’s fastest computer: Summit (148.6 petaflops)
▶ Required time for simulation: approximately \textbf{271,000 years}
Basic Concepts: Factors

Factors (simulation inputs)

- Have impact on responses (simulation outputs)
- Levels: Values of a factor used in experiments
- Factor taxonomy:
  - Quantitative vs qualitative (can encode qualitative)
  - Discrete vs continuous
  - Binary or not
  - Controllable vs uncontrollable
- Factors must be carefully defined
  - Ex: \((s, S)\)-inventory model
  - Use \((s, S)\) or \((s, S - s)\) as the factors?

<table>
<thead>
<tr>
<th>Factor type</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>quantitative (cont.)</td>
<td>Poisson arrival rate</td>
</tr>
<tr>
<td>quantitative (discr.)</td>
<td># of machines</td>
</tr>
<tr>
<td>qualitative</td>
<td>service policy (FIFO, LIFO, ...)</td>
</tr>
<tr>
<td>binary</td>
<td>(open,closed), (high,low), ...</td>
</tr>
<tr>
<td>controllable</td>
<td># of servers</td>
</tr>
<tr>
<td>uncontrollable</td>
<td>weather (sun, rain, fog)</td>
</tr>
</tbody>
</table>

Basic Concepts: Designs

Design matrix

- One column per factor
- Each row is a design point
  - Contains a level for each factor
  - Level values determined by a domain expert
  - Natural or coded design levels
- Can have multiple replications of the design
  - Especially in simulation!

Design point | Factor settings | point | x1 | x2 | x3 |
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>-1 -1 -1</td>
<td>1</td>
<td>-1</td>
<td>-1</td>
<td>-1</td>
</tr>
<tr>
<td>2</td>
<td>+1 -1 -1</td>
<td>2</td>
<td>+1</td>
<td>-1</td>
<td>-1</td>
</tr>
<tr>
<td>3</td>
<td>-1 +1 -1</td>
<td>3</td>
<td>-1</td>
<td>+1</td>
<td>-1</td>
</tr>
<tr>
<td>4</td>
<td>+1 +1 -1</td>
<td>4</td>
<td>+1</td>
<td>+1</td>
<td>-1</td>
</tr>
<tr>
<td>5</td>
<td>-1 -1 +1</td>
<td>5</td>
<td>-1</td>
<td>-1</td>
<td>+1</td>
</tr>
<tr>
<td>6</td>
<td>+1 -1 +1</td>
<td>6</td>
<td>+1</td>
<td>-1</td>
<td>+1</td>
</tr>
<tr>
<td>7</td>
<td>-1 +1 +1</td>
<td>7</td>
<td>-1</td>
<td>+1</td>
<td>+1</td>
</tr>
<tr>
<td>8</td>
<td>+1 +1 +1</td>
<td>8</td>
<td>+1</td>
<td>+1</td>
<td>+1</td>
</tr>
</tbody>
</table>

2^3 factorial design
Some Bad Designs: Capture the Flag

Confounded effects
- Claim: Speed is the most important
- Claim: Stealth is the most important
- Claim: Both are equally important
- There is no way to determine who is right without more data
- Moral: haphazardly choosing design points can use up a lot of time while not providing insight

One-factor-at-a-time (OFAT) sampling
- Claim: Neither speed nor stealth is important
- Problem: an interaction between two factors is being missed

Experimental Design for Simulation
Overview
Basic Concepts and Terminology
Pitfalls
Regression Metamodels and Classical Designs
Other Metamodels
Data Farming

Understanding Simulation Behavior: Metamodels

Simulation metamodels approximate true response
- Simplified representation for greater insight
- Allows "simulation on demand"
- Allows factor screening and optimization

Main-effects metamodel (quantitative factors)
\[ R(x) = \beta_0 + \beta_1 x_1 + \cdots + \beta_k x_k + \epsilon \]

Metamodel with second-order interaction effects
\[ R(x) = \beta_0 + \beta_1 x_1 + \cdots + \beta_k x_k + \sum_i \sum_j \beta_{ij} x_i x_j + \epsilon \]

- \( R \) = simulation model output (i.e., response)
- Factors \( x = (x_1, \ldots, x_k) \)
- \( \epsilon \) = mean-zero noise term, often assumed to be \( N(0, \sigma^2) \)

Experimental Design for Simulation
Overview
Basic Concepts and Terminology
Pitfalls
Regression Metamodels and Classical Designs
Other Metamodels
Data Farming

A Classical Design: \( 2^k \) Factorial Design

Basic setup: \( k \) factors with two levels each (−1, +1)
- Metamodel for \( k = 2 \):
  \[ R(x) = \beta_1 x_1 + \beta_2 x_2 + \beta_{12} x_1 x_2 + \epsilon \]
  \[ r(x) = E[R(x)] = \beta_1 x_1 + \beta_2 x_2 + \beta_{12} x_1 x_2 \]

Estimating “main effects”
- Avg. change in \( r \) when \( x_1 \) goes from −1 to +1 (\( x_2 \) fixed):
  \[ \frac{(n-\bar{n}) + (n-\bar{n})}{2} = -n-\bar{n}+n+n = \frac{R_3}{2} = 2\beta_1 \]
- Similarly, \( \frac{R_4}{2} = 2\beta_2 \)
- Method-of-moments estimators:
  \[ 2\hat{\beta}_1 = \frac{R x_1}{2} \] and \[ 2\hat{\beta}_2 = \frac{R x_2}{2} \]
2\textsuperscript{k} Factorial Design, Continued

**Estimating “interaction effect”**

- (Effect of ↑ \(x_1\) with \(x_2\) high minus effect with \(x_2\) low) / 2
- \[ \frac{(r_{+})-(r_{-})}{2} = r_{12} \]
- Method of moments estimator: \[ 2\hat{\beta}_{12} = \frac{R_{12}}{2} \]

**Observations:**

- Can replicate design to get (Student-t) CI’s for coefficients
- Estimating effects ⇔ estimating regression coefficients
- Above analysis generalizes to more factors, e.g.,

\[
R(x) = \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 + \beta_{12} x_1 x_2 + \beta_{13} x_1 x_3 + \beta_{23} x_2 x_3 + \beta_{123} x_1 x_2 x_3 + \epsilon
\]

<table>
<thead>
<tr>
<th>Design point</th>
<th>Factor settings</th>
<th>Observed response (R)</th>
<th>Predicted expected value (r)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>-1 -1</td>
<td>( R_1 )</td>
<td>( r_1 = -\beta_1 - \beta_2 )</td>
</tr>
<tr>
<td>2</td>
<td>+1 -1</td>
<td>( R_2 )</td>
<td>( r_2 = -\beta_1 + \beta_2 )</td>
</tr>
<tr>
<td>3</td>
<td>+1 +1</td>
<td>( R_3 )</td>
<td>( r_3 = \beta_1 - \beta_2 )</td>
</tr>
<tr>
<td>4</td>
<td>+1 -1</td>
<td>( R_4 )</td>
<td>( r_4 = \beta_1 + \beta_2 )</td>
</tr>
</tbody>
</table>

Using more than two levels gives more detail

- E.g., capture the flag with \(2^2\) versus \(11^2\) designs
  - After achieving a minimal level of stealth, speed is more important
  - Only possible for very small number of factors

\[ m^k \] Designs

**Space-Filling Designs**

**Random Latin Hypercube design**

- Based on random permutations of levels for each factor
- Good coverage of param. space w. relatively few design points
- Carefully crafted LH designs are needed in practice
Gaussian Metamodeling (Kriging)

**Ordinary kriging (deterministic simulations)**

- $Z(x)$ is a Gaussian process
- Models uncertainty due to interpolation
- $(Z(v_1), Z(v_2), \ldots, Z(v_n)) \sim N(0, R(\theta))$
- $r(v_i, v_j) = e^{-\theta(v_i - v_j)^2}$
- $\hat{Y}(x_0) = \hat{\mu} + r^T(x_0)R(\hat{\theta})^{-1}(Y - 1\hat{\mu})$
  - $\hat{\mu}$ and $\hat{\theta}$ are MLE estimates
  - $Y = (Y_1, \ldots, Y_m)$ and $1 = (1, 1, \ldots, 1)$
  - $r = (r(x_0, x_1), r(x_0, x_2), \ldots, r(x_0, x_m))$

**Stochastic kriging (stochastic simulations)**

- $\epsilon$ is $N(0, \sigma^2)$ ("the nugget")
- Captures simulation variability
- Many other variants
  - Fitted derivatives
  - Varying $\sigma^2$
  - Non-constant mean function

---

**Kriging + Trees**

Idea: Build multiple models on subsets of homogeneous data

- Recursively split data to
  - Maximize heterogeneity (e.g., Gini index)
  - Maximize goodness of fit statistic (e.g., $R^2$)
- Build model on each subset
Modern "big data" approach

- Unlike real-world experiments, easier to generate a lot of simulation data
- Most effort usually spent building model, so work it hard!
- Use analytical, graphical, and data mining techniques on generated data

Gaining insight through visualizations

- Examine the results through visualizations
- More sophisticated methods than simple regression
- Analyze flat areas (robustness)
- Other characteristics of interest

Graphical Methods

- Gaining insight through visualizations
- Analyze the results through visualizations
- More sophisticated methods than simple regression
- Analyze flat areas (robustness)
- Other characteristics of interest

Data Farming

- Modern "big data" approach
- Unlike real-world experiments, easier to generate a lot of simulation data
- Most effort usually spent building model, so work it hard!
- Use analytical, graphical, and data mining techniques on generated data

Visual analytics

- Experiments are clustered based on system performance
- Parallel-coordinate plot relates performance to factor levels
- Ex: Manufacturing model with parameters P1, P2, P3, P4

Data Mining and Visual Analytics

- Experiments are clustered based on system performance
- Parallel-coordinate plot relates performance to factor levels
- Ex: Manufacturing model with parameters P1, P2, P3, P4