Learning Surrogate Models of Processes From Experiments or Simulations

Alison Cozad¹  Nick Sahinidis¹  David Miller²

¹Department of Chemical Engineering, Carnegie Mellon University, Pittsburgh, PA
²US DOE-National Energy Technology laboratory, Morgantown, WV
OBJECTIVE: PROCESS SYNTHESIS

- Black box function (simulation, experiment, etc.)

Ideally

Derivative-based optimization

\[
\begin{align*}
\text{min} & \quad f(x) \\
\text{s.t.} & \quad g(x) = 0
\end{align*}
\]
OBJECTIVE: PROCESS SYNTHESIS

Black box function
(simulation, experiment, etc.)

Derivative-based optimization

\[
\begin{align*}
\min & \quad f(x) \\
\text{s.t.} & \quad g(x) = 0
\end{align*}
\]

- Lack of an algebraic model
- Computationally costly simulations/experiments
- Scarcity of fully robust simulations/experiments
OBJECTIVE: PROCESS SYNTHESIS

Black box function (simulation, experiment, etc.)

Derivative-based optimization

\[
\begin{align*}
\min & \quad f(x) \\
\text{s.t.} & \quad g(x) = 0
\end{align*}
\]

ALAMO
Automated Learning of Algebraic Models for Optimization

\[
\begin{align*}
z &= f(x) \\
\min & \quad f(x) \\
\text{s.t.} & \quad g(x) = 0
\end{align*}
\]

Data Models
MODELING PROBLEM STATEMENT

• Build a model of output variables $z$ as a function of input variables $x$ over a specified interval

\[
x \in \mathbb{R}^D \\
x^l \leq x \leq x^u
\]

$z = f(x)$

Independent variables:
Operating conditions, Inlet flow properties, Simulation or experimental geometry, Parameters, etc.

Dependent variables:
Efficiency, Outlet flow conditions, Conversions, Heat flow, etc.

• Desired model traits:
  ✓ Accurate
  ✓ Tractable in algebraic optimization: Simple functional forms
  ✓ Generated from a minimal data set
ALGORITHMIC FLOWSHEET

Start

Initial sampling

Build surrogate model

Update training data set

Adaptive sampling

Model converged

false

true

Stop
ALGORITHMIC FLOWSHEET

Start

Initial sampling

Build surrogate model

Adaptive sampling

Model converged

true → Stop

false → Update training data set

Update training data set

Stop
• Goal: To generate an initial set of input variables to evenly sample the problem space

\[ x = (x^1, x^2, \ldots, x^i, \ldots, x^N) \quad x^i = \begin{pmatrix} x^i_1 \\ x^i_2 \\ \vdots \\ x^i_d \\ \vdots \\ x^i_D \end{pmatrix} \]

• Design of experiments: Latin hypercube design
  – Space-filling design
• After running the design of experiments, we will evaluate the black-box function to determine each $z^i$

\[ x = \begin{pmatrix} x^1 & x^2 & \ldots & x^i & \ldots & x^N \end{pmatrix} \]

\[ z = \begin{pmatrix} z^1 & z^2 & \ldots & z^i & \ldots & z^N \end{pmatrix} \]
ALGORITHMIC FLOWSHEET

Start

Initial sampling

Build surrogate model

Update training data set

Adaptive sampling

Model converged

false

true

Stop
MODEL IDENTIFICATION

• Goal: Identify the **functional form and complexity** of the surrogate models

\[ z = f(x) \]

• Functional form:
  
  – General functional form is unknown: Our method will identify models with combinations of **simple basis functions** of the following forms

<table>
<thead>
<tr>
<th>Category</th>
<th>( X_j(x) )</th>
</tr>
</thead>
<tbody>
<tr>
<td>I.  Polynomial</td>
<td>( (x_d)^\alpha )</td>
</tr>
<tr>
<td>II. Multinomial</td>
<td>( \prod_{d=1}^{m} (x_d)^{\alpha_d} ), for ( m = 1, 2, \ldots )</td>
</tr>
<tr>
<td>III. Exponential and loga-</td>
<td>( \exp \left( \frac{x_d}{\gamma} \right)^\alpha ), ( \log \left( \frac{x_d}{\gamma} \right)^\alpha )</td>
</tr>
<tr>
<td></td>
<td>logarithmic forms</td>
</tr>
<tr>
<td>IV. Expected bases</td>
<td>From experience, simple inspection, etc.</td>
</tr>
</tbody>
</table>

  – **Surrogate model will have the form:**

\[ \hat{z} = \sum_{j \in \mathcal{B}} \beta_j X_j(x) \]
PART A: FIND A FUNCTIONAL FORM

- Let’s assume we know \textit{a priori} the complexity or number of terms in the model, $T$

- Identify the optimal $T$ subset of basis functions such that we
  - Minimize model error
  - Ensure that only $T$ terms are present in the model
  - Enforce the linear least squares condition for the chosen $T$ basis functions

Ordinary linear least squares problem:

$$\min_{\beta} \sum_{i=1}^{N} \left( z^i - \sum_{j \in \mathcal{B}} \beta_j X_{ij} \right)^2$$

where $X_{ij}$ is the basis function $j$ at the black-box data point $i = 1, 2, \ldots, N$
## SELECTING THE BEST $T$ SUBSET

### Linear least squares regression

$$
\min_{\beta} \sum_{i=1}^{N} \left( z^i - \sum_{j \in B} \beta_j X_{ij} \right)^2
$$
SELECTING THE BEST $T$ SUBSET

Linear least squares regression

Solve for a subset of bases

$$\min_{\beta} \sum_{i=1}^{N} \left( z^i - \sum_{j \in B} \beta_j X_{ij} \right)^2$$

$S_h \subset B$, such that $|S_h| = T$

$h = 1$

$$\min_{\beta} \sum_{i=1}^{N} \left( z^i - \sum_{j \in S_1} \beta_j X_{ij} \right)^2$$

$h = 2$

$$\min_{\beta} \sum_{i=1}^{N} \left( z^i - \sum_{j \in S_2} \beta_j X_{ij} \right)^2$$

$h = 3$

$$\min_{\beta} \sum_{i=1}^{N} \left( z^i - \sum_{j \in S_3} \beta_j X_{ij} \right)^2$$, ...
SELECTING THE BEST $T$ SUBSET

<table>
<thead>
<tr>
<th>Linear least squares regression</th>
</tr>
</thead>
<tbody>
<tr>
<td>$S_h \subset B$, such that $</td>
</tr>
</tbody>
</table>

**Solve for a subset of bases**

\[
\min_h \left( \min_{\beta} \sum_{i=1}^{N} \left( z^i - \sum_{j \in S_1} \beta_j X_{ij} \right)^2, \min_{\beta} \sum_{i=1}^{N} \left( z^i - \sum_{j \in S_2} \beta_j X_{ij} \right)^2, \min_{\beta} \sum_{i=1}^{N} \left( z^i - \sum_{j \in S_3} \beta_j X_{ij} \right)^2, \ldots \right)
\]
### SELECTING THE BEST $T$ SUBSET

#### Linear least squares regression

\[
\min_{\beta} \sum_{i=1}^{N} \left( z^i - \sum_{j \in B} \beta_j X_{ij} \right)^2
\]

Solve for a subset of bases

\[
\begin{align*}
\min_h & \left( \min_{\beta} \sum_{i=1}^{N} \left( z^i - \sum_{j \in S_1} \beta_j X_{ij} \right)^2, \min_{\beta} \sum_{i=1}^{N} \left( z^i - \sum_{j \in S_2} \beta_j X_{ij} \right)^2, \min_{\beta} \sum_{i=1}^{N} \left( z^i - \sum_{j \in S_3} \beta_j X_{ij} \right)^2, \ldots \right) \\
& \quad \text{such that } |S_h| = T
\end{align*}
\]

Substitute closed form solution

\[
\begin{align*}
\min_h & \sum_{i=1}^{N} \left( z^i - \sum_{j \in S_h} \beta_j X_{ij} \right)^2 \\
\text{s.t. } & \sum_{i=1}^{N} X_{ij} \left( z^i - \sum_{j \in S_1} \beta_j X_{ij} \right) = 0, \quad \sum_{i=1}^{N} X_{ij} \left( z^i - \sum_{j \in S_2} \beta_j X_{ij} \right) = 0, \quad \sum_{i=1}^{N} X_{ij} \left( z^i - \sum_{j \in S_3} \beta_j X_{ij} \right) = 0, \quad \ldots
\end{align*}
\]
SELECTING THE BEST $T$ SUBSET

\[
\begin{align*}
\text{Solve for the best subset of basis functions} & \\
\min_h & \sum_{i=1}^{N} \left( z^i - \sum_{j \in S_h} \beta_j X_{i,j} \right)^2 \\
\text{s.t.} & \sum_{i=1}^{N} X_{i,j} \left( z^i - \sum_{j \in S_h} \beta_j X_{i,j} \right) = 0, \quad j \in S_h, \quad \forall S_h \subset \mathcal{B}
\end{align*}
\]
Choose terms to minimize the model error

\[
\min \quad SE = \sum_{i=1}^{N} z_i - \sum_{j \in B} \beta_j X_{ij}
\]

s.t. \[
\sum_{j \in B} y_j = T
\]

\[
-U(1-y_j) \leq \sum_{i=1}^{N} X_{ij} \left( z^i - \sum_{j \in B} \beta_j X_{ij} \right) \leq U(1-y_j) \quad j \in B
\]

\[
\beta^l y_j \leq \beta_j \leq \beta^u y_j \quad j \in B
\]

\[
y_j = \{0, 1\} \quad j \in B
\]
### BEST $T$ SUBSET MINIMIZATION MODEL

Choose terms to minimize the model error

$$\min \quad SE = \sum_{i=1}^{N} z_i - \sum_{j \in B} \beta_j X_{ij}$$

s.t. $\sum_{j \in B} y_j = T$

$$-U(1 - y_j) \leq \sum_{i=1}^{N} X_{ij} \left( z^i - \sum_{j \in B} \beta_j X_{ij} \right) \leq U(1 - y_j) \quad j \in B$$

$$\beta^l y_j \leq \beta_j \leq \beta^u y_j \quad j \in B$$

$$y_j = \{0, 1\} \quad j \in B$$
### BEST $T$ SUBSET MINIMIZATION MODEL

**Choose terms to minimize the model error**

| min $SE = \sum_{i=1}^{N} z_i - \sum_{j \in B} \beta_j X_{ij}$ |

**Exactly $T$ terms in the model**

| s.t. $\sum_{j \in B} y_j = T$ |

**Defining the subset of basis functions used**

$$-U(1 - y_j) \leq \sum_{i=1}^{N} X_{ij} \left(z^i - \sum_{j \in B} \beta_j X_{ij}\right) \leq U(1 - y_j) \quad j \in B$$

$$\beta^l y_j \leq \beta_j \leq \beta^u y_j \quad j \in B$$

$$y_j = \{0, 1\} \quad j \in B$$

**Basis function used in the model**

- $y_j = 1$ (assuming loose bounds on $\beta_j$)
- $\beta_j$ is chosen to satisfy a least squares regression

**Basis function NOT used in the model**

- $y_j = 0$
- $\beta_j = 0$
PART B: FIND MODEL COMPLEXITY

• In reality, we don’t know $T$ a priori
• How to find $T$:
  – Corrected Akaike Information Criterion ($AIC_c$)
    • Gives an estimate of the difference between a model and the true function
      
      $$AIC_c = N \log \left( \frac{SSE}{N} \right) + 2T + \frac{2T(T+1)}{n-T-1}$$

      Accuracy + Complexity

      $T$

      $AIC_c$

      Solution
Full algorithm

Start

Initial sampling

Build surrogate model

Adaptive sampling

Update training data set

false

Model converged

true

Stop

Start

T=1

Solve for best $T$ term model

$AICc(T) \geq AICc(T-1)$

OR

$RMSE \leq tol1$

no

yes

Stop
ALGORITHMIC FLOWSHEET

Start

Initial sampling

Build surrogate model

Update training data set

Adaptive sampling

Model converged

false

ture

Stop
ADAPTIVE SAMPLING

- Goal: Search the problem space for areas of model inconsistency or model mismatch

- More succinctly, we are trying to find points that maximize the model error with respect to the independent variables

\[
\max_x \left( \frac{z(x) - \hat{z}(x)}{z(x)} \right)^2
\]

- Optimized using a black-box or derivative-free solver (SNOBFIT)
ADAPTIVE SAMPLING

• Goal: Search the problem space for areas of model inconsistency or model mismatch

• More succinctly, we are trying to find points that maximizes the model error with respect to the independent variables
ADAPITIVE SAMPLING

• Goal: Search the problem space for areas of model inconsistency or model mismatch

• More succinctly, we are trying to find points that maximizes the model error with respect to the independent variables
ADAPTIVE SAMPLING

• Goal: Search the problem space for areas of model inconsistency or **model mismatch**

• More succinctly, we are trying to find points that maximizes the **model error** with respect to the independent variables

---

- Black-box function
- Data points
- Surrogate model
Goal: Search the problem space for areas of model inconsistency or model mismatch

More succinctly, we are trying to find points that maximizes the model error with respect to the independent variables.
• Goal: Search the problem space for areas of model inconsistency or model mismatch

• More succinctly, we are trying to find points that maximizes the model error with respect to the independent variables
ADAPTIVE SAMPLING

• Goal: Search the problem space for areas of model inconsistency or model mismatch

• More succinctly, we are trying to find points that maximizes the model error with respect to the independent variables

Reducing the order of the model is not always desirable
ADAPTIVE SAMPLING FLOWSHEET

Full algorithm

Start

Initial sampling

Build surrogate model

Adaptive sampling

Model converged = true

Stop

Update training data set

false

Model converged = false

Stop

Start

Find new candidate point(s) to sample

Sample simulation at $x_{i'}$, for new data point $i'$

$\max (e_{i', k}) \leq tol2$

Check model error

estError $\leq tol3$

otherwise

Model converged = false

Model converged = true

Stop
ACCURACY VALIDATION

• Compare the method with and without adaptive sampling:

Full algorithm:

- Initial sampling
- Build surrogate model
- Adaptive sampling
- Model converged
  - true: Stop
  - false: Update training data set

Single Latin hypercube:

- Initial sampling
- Build surrogate model
- Stop

• Two sets of known equations made up of functions
  – Present in the algorithm’s basis set
  – Not present in the algorithm’s basis set
FUNCTIONS PRESENT IN THE BASIS

Known 2D equations with 2 to 10 randomly generated terms tested in triplicate

Mean Test Error

Terms in Actual Equation

With adaptive sampling feedback using $N$ points

Without feedback using $N$ points (only Latin hypercube sampling)
FUNCTIONS NOT IN THE BASIS

A. $\alpha x_1^\beta x_2^\gamma$
B. $\frac{\alpha}{\beta + x_1^\gamma + x_2^\delta}$
C. $\alpha x_i^\beta e^{\gamma x_j}$
D. $\alpha x_i^\beta \log(\gamma x_j)$

Terms in Actual Equation

Mean Test Error

With adaptive sampling feedback using $N$ points
Without feedback using $N$ points (only Latin hypercube sampling)
TEST CASE: CUMENE PRODUCTION

Generate Models:

\[ x^C(T^{sp}, P) = f_1(T^{sp}, P) \]
\[ T^{react}(T^{sp}, P) = f_2(T^{sp}, P) \]
\[ F^{rec}(T^{sp}, P) = f_3(T^{sp}, P) \]

Over the Range:

\[ 100^\circ F \leq T^{sp} \leq 250^\circ F \]
\[ 0.82 \text{ atm} \leq P \leq 1.36 \text{ atm} \]
GENERATING THE SURROGATES

Initial data set: 3 points
Final data set: 23 points

• Maximum error found at each iteration may increase
  – Due to the derivative-free solver is given more information at each iteration
FINAL MODELS

\[ x^C(T^{sp}, P) = 0.00156 T^{sp} - \left(5.08 \cdot 10^{-6}\right) (T^{sp})^2 + \left(7.12 \cdot 10^{-9}\right) (T^{sp})^3 + \frac{3.08}{PT^{sp}} + \frac{4.15 \cdot 10^{-10}}{P^3} (T^{sp})^3 - \frac{12.2 \sqrt{P}}{\sqrt{T^{sp}}} + \frac{7.06 \sqrt[3]{P}}{3\sqrt{T^{sp}}} + 0.427 \]

\[ T_{\text{react}}(T^{sp}, P) = \frac{77.1}{\sqrt{P}} + 780.0 \]

\[ F^{\text{rec}}(T^{sp}, P) = \left(5.37 \cdot 10^4\right) \sqrt{0.01 T^{sp}} \sqrt{1.22 P} - 163.0 T^{sp} - \left(6.26 \cdot 10^4\right) \left(0.01 T^{sp}\right)^{\frac{1}{3}} (1.22 P)^{\frac{1}{3}} + \frac{206.0 T^{sp}}{P} + \frac{9377.0}{\sqrt{0.01 T^{sp}}} - \frac{2055.0}{P^3} + \left(2.76 \cdot 10^{-4}\right) (T^{sp})^3 - \frac{0.452(T^{sp})^2}{P^2} - 0.022 P^2 (T^{sp})^2 + \frac{5.44 \cdot 10^{-4}}{P^3} (T^{sp})^3 - \left(1.28 \cdot 10^{-5}\right) P^3 (T^{sp})^3 \]
$T^{sp} = 250^\circ F$

$P = 0.82\text{atm}$

$x^C = 0.981$

$T^{react} = 828.1^\circ F$

$F^{rec} = 645.7 \frac{\text{lb}}{\text{hr}}$

$m^{cw} = 5.00 \cdot 10^7 \frac{\text{lb}}{\text{hr}}$

$A = 7,706 \text{ft}^2$

$\leq 5.2\% \text{error}$
PROCESS OPTIMIZATION

\[
\begin{align*}
T^{\text{sp}} &= 250^\circ F \\
\frac{P}{\text{atm}} &= 0.82 \\
x^C &= 0.981 \\
T^{\text{react}} &= 828.1^\circ F \\
\frac{F^{\text{rec}}}{\text{lb/hr}} &= 645.7 \\
\frac{m^{\text{cw}}}{\text{lb/hr}} &= 5.00 \cdot 10^7 \\
A &= 7,706 \text{ft}^2 \\
\text{max} & x^C \\
\text{min} & \text{Utilities}
\end{align*}
\]

\[
\begin{align*}
T^{\text{sp}} &= 250^\circ F \\
\frac{P}{\text{atm}} &= 1.36 \\
x^C &= 0.964 \\
T^{\text{react}} &= 848.9^\circ F \\
\frac{F^{\text{rec}}}{\text{lb/hr}} &= 373.16 \\
\frac{m^{\text{cw}}}{\text{lb/hr}} &= 4.04 \cdot 10^4 \\
A &= 2,364 \text{ft}^2 \\
\text{\leq 3.0\% error}
\end{align*}
\]
PROCESS OPTIMIZATION

\[ T^{sp} = 250^\circ F \]
\[ P = 0.82 \text{atm} \]
\[ x^C = 0.981 \]
\[ T^{react} = 828.1^\circ F \]
\[ F^{rec} = 645.7 \frac{\text{lb}}{\text{hr}} \]
\[ m^{cw} = 5.00 \cdot 10^7 \frac{\text{lb}}{\text{hr}} \]
\[ A = 7,706 \text{ft}^2 \]
\leq 5.2\% \text{error}

\[ \min \text{ Utilities} \]
\[ T^{sp} = 250^\circ F \]
\[ P = 1.36 \text{atm} \]
\[ x^C = 0.964 \]
\[ T^{react} = 848.9^\circ F \]
\[ F^{rec} = 373.16 \frac{\text{lb}}{\text{hr}} \]
\[ m^{cw} = 4.04 \cdot 10^4 \frac{\text{lb}}{\text{hr}} \]
\[ A = 2,364 \text{ft}^2 \]
\leq 3.0\% \text{error}

\[ \min \text{ CaptialCost} \]
\[ \text{s.t. } x^C \geq 0.95 \]
\[ T^{sp} = 168.1^\circ F \]
\[ P = 0.82 \text{atm} \]
\[ x^C = 0.948 \]
\[ T^{react} = 860.4^\circ F \]
\[ F^{rec} = 172.7 \frac{\text{lb}}{\text{hr}} \]
\[ m^{cw} = 4.40 \cdot 10^4 \frac{\text{lb}}{\text{hr}} \]
\[ A = 2,310 \text{ft}^2 \]
\leq 5.7\% \text{error}
The algorithm we developed is able to model black-box functions for use in optimization such that the models are:

- Accurate
- Tractable in an optimization framework (low-complexity models)
- Generated from a minimal number of function evaluations

Surrogate models can then be incorporated within an optimization framework **flexible objective functions** and additional constraints.

**CONCLUSIONS**