Surrogate (approximations)

- Originated from experimental optimization where measurements are very noisy
- “Approximation” can be actually more accurate than data!
- Great interest now in applying these techniques to computer simulations
- Computer simulations are also subject to noise (numerical)
- However, simulations are exactly repeatable, and if noise is small may be viewed as exact.
- We will discuss polynomial response surfaces and Kriging to deal with both situations
Polynomial response surface approximations

• Data is assumed to be “contaminated” with normally distributed error of zero mean and standard deviation $\sigma$

• Response surface approximation has no bias error, and by having more points than polynomial coefficients it filters out some of the noise.

• Consequently, approximation may be more accurate than data
Fitting approximation to given data

• Noisy response model

• Data from $n_y$ experiments

• Linear approximation

• Rational approximation

• Error measures

\[ y = \hat{y}(x, \beta) + \varepsilon \]

\[ y_i = \hat{y}(x_i, \beta) + \varepsilon_i \]

\[ \hat{y} = \beta_1 + \beta_2 x \]

\[ \hat{y} = \frac{\beta_1}{x + \beta_2} \]

\[ e_{rms} = \sqrt{\frac{1}{n_y \sum_{i=1}^{n} [y_i - \hat{y}(x_i, \beta)]^2}} \]

\[ e_{av} = \frac{1}{n_y \sum_{i=1}^{n} |y_i - \hat{y}(x_i, \beta)|} \]

\[ e_{max} = \max_{x_i} |y_i - \hat{y}(x_i, \beta)| \]
Linear Regression

• Functional form
• For linear approximation
• Estimate of coefficient vector denoted as \( \mathbf{b} \)

- Rms error

• Minimize rms error
  \[ \mathbf{e}^T \mathbf{e} = (\mathbf{y} - \mathbf{X} \mathbf{b}^T)^T (\mathbf{y} - \mathbf{X} \mathbf{b}^T) \]

- Differentiate to obtain
  \[ \mathbf{X}^T \mathbf{X} \mathbf{b} = \mathbf{X}^T \mathbf{y} \]

Beware of ill-conditioning!
Example 3.1.1

• Data: \( y(0)=0, \ y(1)=1, \ y(2)=1 \)

• Fit linear polynomial \( y=b_0+b_1x \)
  
  \[ x = \begin{bmatrix} 1 & 0 \\ 1 & 1 \\ 1 & 2 \end{bmatrix}, \quad y = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}. \]

• Then

  \[ x^T x = \begin{bmatrix} 1 & 0 \\ 1 & 1 \\ 1 & 2 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 1 & 1 \\ 1 & 2 \end{bmatrix} = \begin{bmatrix} 3 & 3 \\ 3 & 5 \end{bmatrix}, \]

  \[ x^T y = \begin{bmatrix} 1 & 0 \\ 1 & 1 \\ 1 & 2 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}. \]

• Obtain \( b_0=1/3, \ b_1=0. \)

\[
\begin{align*}
y(0) &= b_0 = 0 \\
y(1) &= b_0 + b_1 = 1 \\
y(2) &= b_0 + 2b_1 = 0
\end{align*}
\]

\[
\begin{align*}
3b_0 + 3b_1 &= 1 \\
3b_0 + 5b_1 &= 1
\end{align*}
\]
Comparison with alternate fits

- Errors for regression fit
  
  \[ e_{\text{rms}} = 0.47 \quad e_{\text{av}} = 0.44 \quad e_{\text{max}} = 0.67 \]

- To minimize maximum error obviously \( y=0.5 \). Then \( e_{\text{av}} = e_{\text{rms}} = e_{\text{max}} = 0.5 \)

- To minimize average error, \( y=0 \) \( e_{\text{av}} = 1/3 \), \( e_{\text{max}} = 1 \), \( e_{\text{rms}} = 0.577 \)

- What should be the order of the progression from low to high?
Three lines

Figure 3.1: Fitting a straight line to three data points
Estimating the accuracy of the approximation (surrogate)

- From assumption that error is due to normally distributed uncorrelated random variables, get estimate to error standard deviation (called standard error)

\[ \hat{\sigma}^2 = \frac{\mathbf{e}^T \mathbf{e}}{n_y - n_\beta} \]

- Standard measure of accuracy

- Coefficient of multiple determination measures how much of variability in data is captured by approximation

\[ SS_y = \sum_{i=1}^{n_y} (y_i - \bar{y})^2 \quad SS_r = \sum_{i=1}^{n_y} (\hat{y}_i - \bar{y})^2 \quad R^2 = \frac{SS_r}{SS_y} \]

- Adjusted coefficient of multiple determination accounts for the fitting bias

\[ R_a^2 = 1 - (1 - R^2) \frac{n_y - 1}{n_y - n_\beta} \]
Cross validation

- Error estimates based on model assumptions are vulnerable
- For polynomial response surface approximations assumptions are rarely satisfied
- Cross validation divides data into $n_g$ groups
- Fit the approximation to $n_g - 1$ groups, and use last group to estimate error. Repeat for each group
- When each group consists of one point, error called PRESS (prediction error sum of squares)
- Calculate error at each point and then presenting r.m.s error
- Can be shown that
- Can be used only if not ill-conditioned

\[ e_{pi} = \frac{e_i}{1 - E_{ii}} \quad E = X(X^T X)^{-1} X^T \]
Estimating error in coefficients

• Depending on design of experiments, some coefficients are more accurately estimated than others
• Standard error in coefficient is given as

\[ se(b_i) = \hat{\sigma} \sqrt{\left(X^T X\right)_{ii}} \]

• Coefficients that are poorly estimated may need to be dropped to improve accuracy of predictions using approximation
• Unfortunately, dropping one coefficient changes t-statistics for others
• Need to iterate in dropping and adding coefficients
Example 3.2.1

- Given data
- Use Microsoft Excel to fit linear and quadratic polynomials
- Compare standard errors and t-statistics of coefficients

<table>
<thead>
<tr>
<th>X</th>
<th>-2</th>
<th>-1</th>
<th>0</th>
<th>1</th>
<th>2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Y</td>
<td>-1.5</td>
<td>-1.5</td>
<td>0</td>
<td>1.25</td>
<td>1.75</td>
</tr>
</tbody>
</table>
### SUMMARY OUTPUT

**Regression Statistics**
- Multiple R: 0.968334
- R Square: 0.937671
- Adjusted R: 0.916895
- Standard Error: 0.435412
- Observations: 5

### ANOVA

<table>
<thead>
<tr>
<th>df</th>
<th>SS</th>
<th>MS</th>
<th>F</th>
<th>Significance F</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>8.55625</td>
<td>8.55625</td>
<td>45.13187</td>
<td>0.006732</td>
</tr>
<tr>
<td>3</td>
<td>0.56875</td>
<td>0.189583</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>9.125</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Coefficients

<table>
<thead>
<tr>
<th>Coefficient</th>
<th>Standard Error</th>
<th>t Stat</th>
<th>P-value</th>
<th>Lower 95%</th>
<th>Upper 95%</th>
</tr>
</thead>
<tbody>
<tr>
<td>Intercept</td>
<td>0.194722</td>
<td>0</td>
<td>1</td>
<td>-0.61969</td>
<td>0.619693</td>
</tr>
<tr>
<td>X Variable</td>
<td>0.137689</td>
<td>6.718026</td>
<td>0.006732</td>
<td>0.486811</td>
<td>1.363189</td>
</tr>
</tbody>
</table>

### RESIDUAL OUTPUT

<table>
<thead>
<tr>
<th>Observation</th>
<th>Predicted Y</th>
<th>Residuals</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>-1.85</td>
<td>0.35</td>
</tr>
<tr>
<td></td>
<td>Intercept</td>
<td>X Variable</td>
</tr>
<tr>
<td>------------</td>
<td>-----------</td>
<td>------------</td>
</tr>
<tr>
<td>Coefficients</td>
<td>0.525</td>
<td>0.162569</td>
</tr>
<tr>
<td>Standard Error</td>
<td>0.358284</td>
<td>0.137366</td>
</tr>
<tr>
<td>t Stat</td>
<td>-1.64972</td>
<td>1.64972</td>
</tr>
<tr>
<td>P-value</td>
<td>0.10714</td>
<td>0.358284</td>
</tr>
<tr>
<td>Lower 95%</td>
<td>-2.95054</td>
<td>0.137366</td>
</tr>
<tr>
<td>Upper 95%</td>
<td>-0.29904</td>
<td>0.137366</td>
</tr>
<tr>
<td>Lower 95%</td>
<td>-1.64972</td>
<td>0.137366</td>
</tr>
<tr>
<td>Upper 95%</td>
<td>1.64972</td>
<td>0.137366</td>
</tr>
</tbody>
</table>

**ANOVA**

<table>
<thead>
<tr>
<th>df</th>
<th>SS</th>
<th>MS</th>
<th>F</th>
<th>Significance F</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>8.596429</td>
<td>4.298214</td>
<td>16.26351</td>
<td>0.057926</td>
</tr>
<tr>
<td>2</td>
<td>0.528571</td>
<td>0.264286</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>9.125</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Regression Statistics**

- Multiple R: 0.970605
- R Square: 0.942074
- Adjusted R Square: 0.884149
- Standard Error: 0.514087
- Multiple R: 0.942074
- R Square: 0.514087

**Residual Output**

<table>
<thead>
<tr>
<th>Observed Y</th>
<th>Predicted Y</th>
<th>Residual</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Quadratic fit**

<table>
<thead>
<tr>
<th>X Variable</th>
<th>Observed Y</th>
<th>Predicted Y</th>
<th>Residual</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.525</td>
<td>0.162569</td>
<td>0.51331</td>
<td>0.268614</td>
</tr>
<tr>
<td>0.162569</td>
<td>0.51331</td>
<td>0.268614</td>
<td></td>
</tr>
<tr>
<td>0.137366</td>
<td>0.137366</td>
<td>0.137366</td>
<td></td>
</tr>
<tr>
<td>0.137366</td>
<td>0.137366</td>
<td>0.137366</td>
<td></td>
</tr>
<tr>
<td>0.137366</td>
<td>0.137366</td>
<td>0.137366</td>
<td></td>
</tr>
</tbody>
</table>
Graphical comparison

Figure 3.4: Comparison of Linear and Quadratic Fit
Exercise

• Generate data for the function $y=x$ at $n$ points on the interval (-2,2) with random noise uniformly distributed in (-0.25,0.25)

• Fit quadratic polynomial for $n>3$

• Check how many points you need to get coefficient of $x$ within (0.9,1.1) with 95% confidence

• Compare standard error to PRESS error for each case

• Debug on five point data from Example 3.2.1
Design of experiments

• Selecting sampling points is known as design of experiments
• Classical designs include full-factorial design, \( n \times n \times n \times n \times n \ldots \), central composite design, which is based on 3-level full-factorial design with only face points for the intermediate level
• D-optimal design minimizes variance - volume of coefficient ellipsoid
• Latin hypercube sampling (LHSDESIGN) distributes points evenly through design space.
4.2.1 Interpolation, extrapolation and prediction variance

- Interpolation is mathematically contrasted to regression or least-squares fit.
- As important is the contrast between interpolation and extrapolation.
- Extrapolation occurs when we are outside the convex hull of the data points.

\[ x = \sum_{i=1}^{n+1} \alpha_i x_i : \quad \sum_{i=1}^{n+1} \alpha_i = 1, \quad \text{and} \quad \alpha_i \geq 0, \quad i = 1, \ldots, n + 1 \]

- For high dimensional spaces we must have extrapolation!
Prediction variance

• Linear regression model
  \[ \hat{y} = \sum_{i=1}^{n_\beta} b_i \xi_i(x) \]

• Define \( x_i^{(m)} = \xi_i(x) \) then
  \[ \hat{y} = x^{(m)T} b \]

• With some algebra
  \[ \text{Var}[\hat{y}(x)] = x^{(m)T} \Sigma_b x^{(m)} = \sigma^2 x^{(m)T} (X^T X)^{-1} x^{(m)} \]

• Standard error
  \[ s_y = \hat{\sigma} \sqrt{x^{(m)T} (X^T X)^{-1} x^{(m)}} \]
Example 4.2.1

• For a linear polynomial $y = b_1 + b_2 x_1 + b_2 x_2$ find the prediction variance in the region

$$-1 \leq x_1 \leq 1, -1 \leq x_2 \leq 1$$

• (a) For data at three vertices

$$x_1^T = [-1, -1], x_2^T = [-1, 1], x_3^T = [1, -1]$$

• (b) For data at all four vertices
Data at three vertices

- Get

\[ x^{(m)} = \begin{cases} 1 \\ x_1 \\ x_2 \end{cases}, \quad x^{(m)T}(X^TX)^{-1}x^{(m)} = 0.5(1 + x_1 + x_2 + x_1^2 + x_2^2 + x_1x_2) \]

- Actual standard error multiplied by noise amplitude

- At the vertices \[ s_y = \hat{\sigma} \]

- Minimum error

\[ s_y = \frac{1}{\sqrt{3}} \hat{\sigma} \quad \text{at} \quad x_1 = x_2 = -\frac{1}{3} \]

- At

\[ x_1 = x_2 = 1, \quad s_y = \sqrt{3}\hat{\sigma} \]
Data at four vertices

- Now
  \[ x = \begin{bmatrix} 1 & -1 & -1 \\ 1 & -1 & 1 \\ 1 & 1 & -1 \\ 1 & 1 & 1 \end{bmatrix}, \quad x^T x = 4 \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \]

- And
  \[ x^{(m)}^T (X^T X)^{-1} x^{(m)} = 0.25(1 + x_1^2 + x_2^2) \]

- Error at vertices
  \[ s_y = \frac{\sqrt{3}}{2} \hat{\sigma} \]

- At the origin minimum is
  \[ s_y = \frac{1}{2} \hat{\sigma} \]

- How can we reduce error without adding points?
Central Composite Design (CCD)

- Classical DOE for quadratic RS
- FCCCD
  - Move axial points to surface of the hypercube
Box-Benken designs

- Disturb only a small number of variables from nominal value. For these use all combinations. For example for $n=3$, with two variables changing

\[
\begin{array}{ccc}
x_1 & x_2 & x_3 \\
-1 & -1 & 0 \\
-1 & 1 & 0 \\
1 & -1 & 0 \\
1 & 1 & 0 \\
-1 & 0 & -1 \\
-1 & 0 & 1 \\
1 & 0 & -1 \\
1 & 0 & 1 \\
0 & -1 & -1 \\
0 & -1 & 1 \\
0 & 1 & -1 \\
0 & 1 & 1 \\
0 & 0 & 0
\end{array}
\]
Variance optimal designs

• A key to most optimal DOE methods is moment matrix \( M=X^TX/n_y \)
• The variance of the coefficients and the prediction variance depend on \( (X^TX)^{-1} \).
• A good design of experiments will minimize the terms in this matrix, especially the diagonal elements
• D-optimal designs maximize determinant of moment matrix
• Inversely proportional to square of volume of confidence region on coefficients
Example

- Given the model $y = b_1 x_1 + b_2 x_2$, and the two data points (0,0) and (1,0), find the optimum third data point $(p,q)$.

- We have

\[
X = \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ p & q \end{bmatrix} \quad X^T X = \begin{bmatrix} 1 + p^2 & pq \\ pq & q^2 \end{bmatrix} \quad \text{det}(X^T X) = q^2
\]

- So that the third point is $(p,1)$, for any value of $p$.

- Finding D-optimal design in higher dimension is difficult optimization problems often solved heuristically.
Other criteria

• A-optimal minimizes trace of inverse of moment matrix, minimizes the sum of the variances of the coefficients

• G-optimality minimizes the maximum of the prediction variance.
Example

• For the previous example, find the A-optimal design

\[ X = \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ p & q \end{bmatrix} \quad X^T X = \begin{bmatrix} 1 + p^2 & pq \\ pq & q^2 \end{bmatrix} \quad \text{det}(X^T X) = q^2 \]

\[ (X^T X)^{-1} = \frac{1}{q^2} \begin{bmatrix} q^2 & -pq \\ -pq & 1 + p^2 \end{bmatrix} \]

\[ tr((X^T X)^{-1}) = 1 + \frac{1 + p^2}{q^2} \]

• Minimum at (0, 1)
Latin Hypercube Sampling (LHS)

- Advantages
  - Space-filling
  - arbitrary number of design points
  - Matlab LHSDESIGN can perform optimization to maximize minimum distance between points
Review of various DOE

• Questions to ask
  – Is noise an important issue
  – What surrogate do we prefer
  – How many simulations can we afford
  – How many variables do we need to include
  – Do we want to do all the simulations at once or can we do adaptive sampling
Recommendations

• Low-dimensional spaces with much noise
  – Full factorial or ccd for box domains
  – D-optimal designs for irregular domains and with adaptive sampling

• Low-dimensional spaces without much noise
  – Minimum bias, LHS and orthogonal arrays for box-like domains
  – Monte Carlo and optimized distance designs for irregular domains
Recommendations

• High-dimensional spaces with noise
  – Box-Benken and partial factorial or small CCD for regular shaped domains
  – D-optimal designs for irregularly shaped domains

• High-dimensional spaces without much noise
  – LHS and optimized distance designs
Surrogate modeling

- There are many surrogate models
  - Polynomial response surface approximation (PRS or RSA)
  - Kriging (KRG)
  - Radial basis functions (RBF)
  - Support vector regression (SVR) etc.

- Major steps in surrogate modeling
  - Choice of sampling data points (DOE)
  - Simulations
  - Model construction
  - Model appraisal (error estimation)
Kriging

\[ \hat{y}(x) = \sum_{i=1}^{N_v} \beta_i \xi_i(x) + Z(x) \]

\[ C(Z(x), Z(s), \theta) = \prod_{i=1}^{N_v} \exp(-\theta_i (x_i - s_i)^2) \]

- Named after a South African mining engineer D. G. Krige
- Assumption: Systematic departures \( Z(x) \) are correlated
- Gaussian correlation function \( C(x, s, \theta) \) is most popular

Linear trend model
Systematic departure

Sampling data points
Linear Trend Model
Systematic Departure
Kriging
Radial basis neural networks

\[ \hat{y}(x) = \sum_{i=1}^{N_{RBF}} w_i a_i(x) \]

\[ a = \text{radbas} \left( \| w - x \| / b \right); \text{radbas} \left( n \right) = e^{-n^2} \]

- Neurons (Radial basis functions) at some of data points used to approximate response. Other points used to estimate error
- User-defined constants:
  - Spread constant: radius of influence \( 1/b \)
  - Error goal: minimum sum of square of errors
Support vector regression

Fig. 1. *The soft margin loss setting for a linear SVM (from Schölkopf and Smola, 2002)*

- We do not care about error below epsilon, but we care about magnitude of polynomial coefficients
Suggested Exercises

Exercise on page 16.
Generate D-optimal and LHS designs with Matlab for a square and compare for different number of points

Source: www.library.veryhelpful.co.uk/Page11.htm