Curve Fitting

Mohammad Hadi

mohammad.hadi@sharif.edu

@MohammadHadiDastgerdi

Spring 2022

Overview

Curve Fitting

Curve Fitting

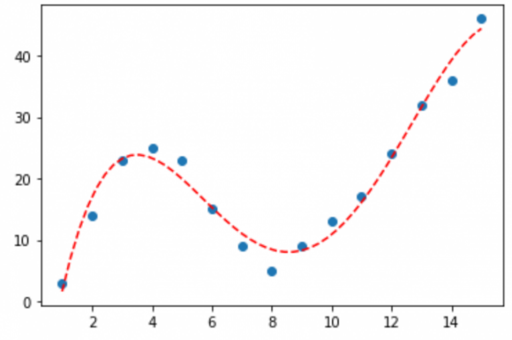

Figure: Curve fitting is the process of constructing a curve, or mathematical function, that has the best fit to a series of data points. MATLAB and Python have tools for different types of curve fitting. Curve fitting can be used to estimate the characteristic curve of different electrical elements.

Curve Fitting Types

1. Polynomial: $y = \sum_{i=0}^{n} a_n x^n$
   - Linear: $y = a_1 x + a_0$
   - Quadratic: $y = a_2 x^2 + a_1 x + a_0$
   - Cubic: $y = a_3 x^3 + a_2 x^2 + a_1 x + a_0$
2. Exponential: $y = a e^{bx} + c$
3. Power: $y = a x^b$
4. Fractional: $y = \frac{\sum_{i=0}^{n} a_n x^n}{\sum_{i=0}^{n} b_n x^n}$
5. Custom: $y = f(x)$

Polynomial Curve Fitting

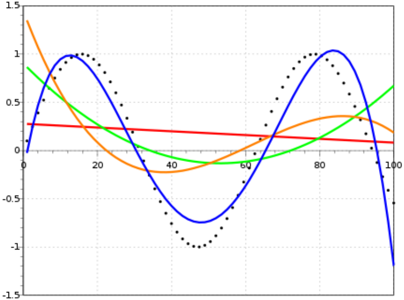

Figure: Polynomial curve fitting points generated with a sine function. The black dotted line is the "true" data, the red line is a first degree polynomial, the green line is second degree, the orange line is third degree and the blue line is fourth degree.

Linear Curve Fitting

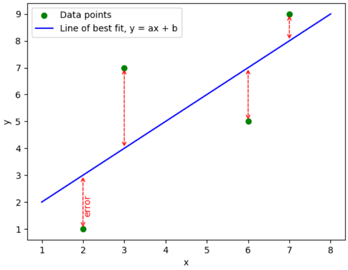

Figure: Linear curve fitting to four data points using least squares error approach.

- Linear curve fitting: $y = f(x) = ax + b$
- Least squares error: $\epsilon = \sum_{i=1}^{n}(y_i - f(x_i))^2 = \sum_{i=1}^{n}(y_i - ax_i - b)^2$
- Optimization: $\begin{cases} \frac{\partial \epsilon}{\partial a} = -2\sum_{i=1}^{n} x_i(y_i - ax_i - b) = 0 \\ \frac{\partial \epsilon}{\partial b} = -2\sum_{i=1}^{n}(y_i - ax_i - b) = 0 \end{cases}$
- Optimal coefficients: $\begin{cases} (\sum_{i=1}^{n} x_i^2)a + (\sum_{i=1}^{n} x_i)b = (\sum_{i=1}^{n} x_i y_i) \\ (\sum_{i=1}^{n} x_i)a + nb = (\sum_{i=1}^{n} y_i) \end{cases}$
- Optimal coefficients: $\begin{bmatrix} \sum_{i=1}^{n} x_i^2 & \sum_{i=1}^{n} x_i \\ \sum_{i=1}^{n} x_i & n \end{bmatrix} \begin{bmatrix} a \\ b \end{bmatrix} = \begin{bmatrix} \sum_{i=1}^{n} x_i y_i \\ \sum_{i=1}^{n} y_i \end{bmatrix}$

The End