***PAPALAMBROS AND WILDE***

# *Principles of Optimal Design Solutions Manual*

Third Edition

## Chapter 2

## 2.1

Derive general expressions for the coefficients of linear, quadratic, and cubic approximations, when the sampling points are equally spaced along the $x$-axis.

### Solution

i) Linear approximation with 2 data points $(x_0, y_0)$ and $(x_1, y_1)$

Let $\Delta x = x_1 - x_0$

Approximating the solution as $y = a_0 + a_1 x$, then we obtain

$$\begin{array}{l} y_0 = a_0 + a_1 x_0 \\ y_1 = a_0 + a_1 x_1 \end{array} \Rightarrow \begin{pmatrix} 1 & x_0 \\ 1 & x_1 \end{pmatrix} \begin{pmatrix} a_0 \\ a_1 \end{pmatrix} = \begin{pmatrix} y_0 \\ y_1 \end{pmatrix} \Rightarrow \begin{pmatrix} a_0 \\ a_1 \end{pmatrix} = \begin{pmatrix} y_0 - \frac{x_0}{\Delta x}(y_1 - y_0) \\ \frac{y_1 - y_0}{\Delta x} \end{pmatrix}$$

Determining $\Delta y_1 = y_1 - y_0$, we get

$a_1 = \frac{\Delta y_1}{\Delta x}$ and $a_0 = y_0 - a_1 x_0$.

ii) quadratic approximation with 3 data points $(x_0, y_0)$, $(x_1, y_1)$ and $(x_2, y_2)$

Let $\Delta x = x_1 - x_0 = x_2 - x_1$, $\Delta y_1 = y_1 - y_0$ and $\Delta y_2 = y_2 - y_1$

Approximating the solution as $y = a_0 + a_1 x + a_2 x^2$, then we obtain

$$\begin{array}{l} y_0 = a_0 + a_1 x_0 + a_2 x_0^2 \\ y_1 = a_0 + a_1 x_1 + a_2 x_1^2 \\ y_2 = a_0 + a_1 x_2 + a_2 x_2^2 \end{array} \Rightarrow \begin{pmatrix} 1 & x_0 & x_0^2 \\ 1 & x_1 & x_1^2 \\ 1 & x_2 & x_2^2 \end{pmatrix} \begin{pmatrix} a_0 \\ a_1 \\ a_2 \end{pmatrix} = \begin{pmatrix} y_0 \\ y_1 \\ y_2 \end{pmatrix}$$

$$\Rightarrow \begin{pmatrix} 1 & x_0 & x_0^2 \\ 0 & (x_1 - x_0) & (x_1 - x_0)(x_1 + x_0) \\ 0 & (x_2 - x_0) & (x_2 - x_0)(x_2 + x_0) \end{pmatrix} \begin{pmatrix} a_0 \\ a_1 \\ a_2 \end{pmatrix} = \begin{pmatrix} y_0 \\ y_1 - y_0 \\ y_2 - y_0 \end{pmatrix}$$

$$\Rightarrow \begin{pmatrix} 1 & x_0 & x_0^2 \\ 0 & \Delta x & \Delta x(x_1 + x_0) \\ 0 & 0 & \Delta x(2x_2 - 2x_1) \end{pmatrix} \begin{pmatrix} a_0 \\ a_1 \\ a_2 \end{pmatrix} = \begin{pmatrix} y_0 \\ \Delta y_1 \\ \Delta y_2 - 2\Delta y_1 \end{pmatrix}$$

$$\Rightarrow \begin{pmatrix} 1 & x_0 & x_0^2 \\ 0 & \Delta x & \Delta x(x_1 + x_0) \\ 0 & 0 & 2\Delta x^2 \end{pmatrix} \begin{pmatrix} a_0 \\ a_1 \\ a_2 \end{pmatrix} = \begin{pmatrix} y_0 \\ \Delta y_1 \\ \Delta y_2 - 2\Delta y_1 \end{pmatrix}$$

$$\text{Therefore,} \begin{array}{l} a_2 = (\Delta y_2 - 2\Delta y_1)/2\Delta x^2 \\ a_1 = \frac{\Delta y_1}{\Delta x} - (2x_0 + \Delta x) a_2 \ . \\ a_0 = y_0 - a_1 x_0 - a_2 x_0^2 \end{array}$$

iii) cubic approximation with 4 data points $(x_0, y_0)$, $(x_1, y_1)$, $(x_2, y_2)$ and $(x_3, y_3)$

Let $\Delta x = x_1 - x_0 = x_2 - x_1 = x_3 - x_2$, $\Delta y_1 = y_1 - y_0$, $\Delta y_2 = y_2 - y_1$ and $\Delta y_3 = y_3 - y_2$

Approximating the solution as $y = a_0 + a_1 x + a_2 x^2 + a_3 x^3$, then we obtain

$$\begin{array}{l} y_0 = a_0 + a_1 x_0 + a_2 x_0^2 + a_3 x_0^2 \\ y_1 = a_0 + a_1 x_1 + a_2 x_1^2 + a_3 x_1^2 \\ y_2 = a_0 + a_1 x_2 + a_2 x_2^2 + a_3 x_2^2 \\ y_3 = a_0 + a_1 x_3 + a_2 x_3^2 + a_3 x_3^2 \end{array} \Rightarrow \begin{pmatrix} 1 & x_0 & x_0^2 & x_0^3 \\ 1 & x_1 & x_1^2 & x_1^3 \\ 1 & x_2 & x_2^2 & x_2^3 \\ 1 & x_3 & x_3^2 & x_3^3 \end{pmatrix} \begin{pmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \end{pmatrix} = \begin{pmatrix} y_0 \\ y_1 \\ y_2 \\ y_3 \end{pmatrix}$$

$$\Rightarrow \begin{pmatrix} 1 & x_0 & x_0^2 & x_0^3 \\ 0 & \Delta x & \Delta x(x_1 + x_0) & \Delta x(x_1^2 + x_1 x_0 + x_0^2) \\ 0 & 2\Delta x & 2\Delta x(x_2 + x_0) & 2\Delta x(x_2^2 + x_2 x_0 + x_0^2) \\ 0 & 3\Delta x & 3\Delta x(x_3 + x_0) & 3\Delta x(x_3^2 + x_3 x_0 + x_0^2) \end{pmatrix} \begin{pmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \end{pmatrix} = \begin{pmatrix} y_0 \\ \Delta y_1 \\ \Delta y_2 \\ \Delta y_3 \end{pmatrix}$$

$$\Rightarrow \begin{pmatrix} 1 & x_0 & x_0^2 & x_0^3 \\ 0 & 1 & (x_1+x_0) & (x_1^2+x_1x_0+x_0^2) \\ 0 & 1 & (x_2+x_0) & (x_2^2+x_2x_0+x_0^2) \\ 0 & 1 & (x_3+x_0) & (x_3^2+x_3x_0+x_0^2) \end{pmatrix} \begin{pmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \end{pmatrix} = \begin{pmatrix} y_0 \\ \Delta y_1/\Delta x \\ \Delta y_2/2\Delta x \\ \Delta y_3/3\Delta x \end{pmatrix}$$

$$\Rightarrow \begin{pmatrix} 1 & x_0 & x_0^2 & x_0^3 \\ 0 & 1 & (x_1+x_0) & (x_1^2+x_1x_0+x_0^2) \\ 0 & 0 & \Delta x & \Delta x(x_2+x_1+x_0) \\ 0 & 0 & 2\Delta x & 2\Delta x(x_3+x_1+x_0) \end{pmatrix} \begin{pmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \end{pmatrix} = \begin{pmatrix} y_0 \\ \Delta y_1/\Delta x \\ (\Delta y_2/2-\Delta y_1)/\Delta x \\ (\Delta y_3/3-\Delta y_1)/\Delta x \end{pmatrix}$$

$$\Rightarrow \begin{pmatrix} 1 & x_0 & x_0^2 & x_0^3 \\ 0 & 1 & (x_1+x_0) & (x_1^2+x_1x_0+x_0^2) \\ 0 & 0 & 1 & (x_2+x_1+x_0) \\ 0 & 0 & 0 & \Delta x \end{pmatrix} \begin{pmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \end{pmatrix} = \begin{pmatrix} y_0 \\ \Delta y_1/\Delta x \\ (\Delta y_2/2-\Delta y_1)/\Delta x \\ (\Delta y_3/6-\Delta y_2/2+\Delta y_1/2)/\Delta x^2 \end{pmatrix}$$

Therefore, $\begin{array}{l} a_3 = (3\Delta y_1 - 3\Delta y_2 + \Delta y_3)/6\Delta x^3 \\ a_2 = (\Delta y_2 - 2\Delta y_1)/2\Delta x^2 - 3(x_0+\Delta x)a_3 \\ a_1 = \frac{\Delta y_1}{\Delta x} - (2x_0+\Delta x)a_2 - (3x_0^2 + 3\Delta x x_0 + \Delta x^2)a_3 \\ a_0 = y_0 - a_1x_0 - a_2x_0^2 - a_3x_0^3 \end{array}.$

**2.2**

Consider an electric motor series cost model with the data given below (Stoecker 1971):

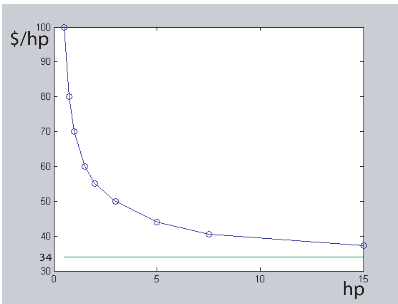
| **hp** | **Cost/$** | **$/hp** |
|---|---|---|
| 0.50 | 50 | 100.00 |
| 0.75 | 60 | 80.00 |
| 1.00 | 70 | 70.00 |
| 1.50 | 90 | 60.00 |
| 2.00 | 110 | 55.00 |
| 3.00 | 150 | 50.00 |
| 5.00 | 220 | 44.00 |
| 7.50 | 305 | 40.50 |
| 15.00 | 560 | 37.30 |

Derive the curve-fitting equation $\$/\text{hp} = 34.5 + 36(hp)^{-0.865}$.
*Hint*: Draw the curve using the table values and estimate a value for the constant term. For the steep part of the curve, draw its representation on a log–log plot to get values for the coefficient of the second term. Iterate as necessary.

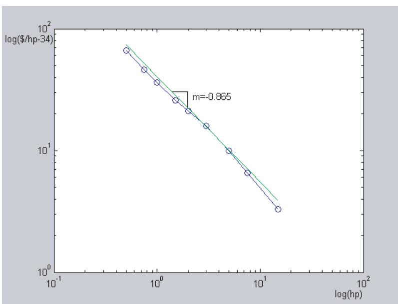
**Solution**
Assume that the form of approximated equation for data is

$$\$/hp = a + b(hp)^m \tag{2.1}$$

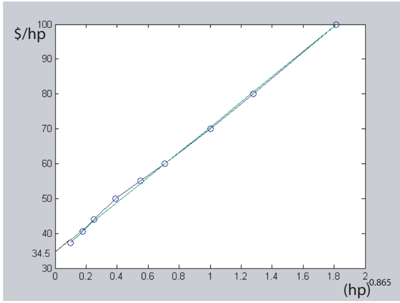
Following the hint, plot the data and estimate that \$ /hp levels out at a value of about 34.




Then plot the value of (\$/hp-34) versus hp in log-log scale for the steepest point of the curve, which is between the 6th and 7th datum. From the slope of the line, m is found to be -0.865.



Next step, the \$/hp values are plotted against $(hp)^{-0.865}$. The plot is shown below. The intersection of the line with \$ /hp axis, a, is estimated as 34.5



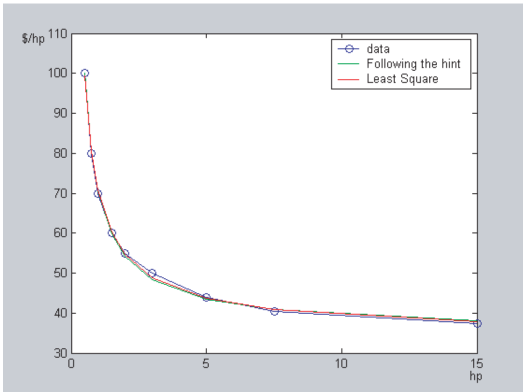

Therefore, the final approximated equation is

$$\$/hp = 34.5 + 36(hp)^{-0.865} \tag{2.2}$$

If one wants more precise coefficients, the leat square curve fitting is the most convenient and popular method. The 'lsqcurve fit' in the Matlab will give a more precise equation. And the result is following.

$$\$/hp = 34.0754 + 36.6451(hp)^{-0.8344} \tag{2.3}$$

```
% Example code for Exercise 2.2
hp=[0.5 0.75 1 1.5 2 3 5 7.5 15];
dphp=[100 80 70 60 55 50 44 40.5 37.3];
fun = inline('x(1)+x(2).*(hp).^(x(3))','x','hp');
x = lsqcurvefit(fun,[34 36.5 -0.83], hp, dphp)
```



## 2.3

Helical compression spring design is an often-used example of optimization formulation because of its simplicity. Formulate such a model with spring index and wire diameter as the two design variables. Choose an objective function (e.g., weight) and create as many constraints as you can think of. Typically, these include surging, buckling, stress, clash allowance, geometric limitations, and minimum number of coils. Select parameter values and find the solution graphically.

**Solution**
Objective Function:
Minimize the inverse of the safety factor for yielding $\overline{SF_y}$ or Fatigue $\overline{SF_f}$, ie Maximize "Reliability".

Spring Material : Music Wire (ASTM A228)
Experimental data shows that the ultimate strength of spring material is a function of the wire diameter.

$S_u = A^1 d^{A_1}$ : Ultimate Strength
$S_{us} = 0.8 S_u$ : Ultimate Shear Strength
$S_y = 0.75 S_u$ : Yield Strength Strength
$S_y s = 0.577 S_y$ : Shear Yield
$S_{NS} = C_1 d^{A_1} (\overline{NC})^{B_1}$ : Fatigue Strength as Expressed by the S-N curve

where $A^1 = 200,000$, $A_1 = -0.14$, $C_1 = 630,500$, $B^1 = -0.2137$ and NC : Number of Cycle to Failure ($10^6$)

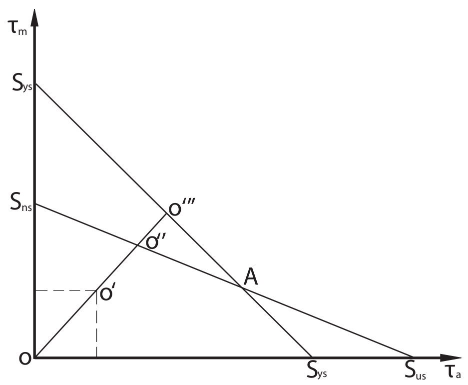



Figure 2.1. Fatigue Diagram for spring Design

From Fatigue Diagram(Figure 2.1), we have

$\overline{SF_f} = \frac{\overline{OO''}}{\overline{OO'}} = \frac{1}{(\tau_m/S_{ns}+\tau_m/S_{us})}$

$\overline{SF_y} = \frac{S_{ys}}{\tau_a+\tau_m}$

and if $\frac{\tau_a}{\tau_m} \geq \frac{S_{ns}(S_{ys}-S_{us})}{S_{us}(S_{ns}-S_{ys})}$

then fatigue will be critical, else yielding will be critical.

$$
\begin{aligned}
\tau_a &= \tfrac{8F_aDK_w}{\pi d^3} \\
\tau_m &= \tfrac{8F_mDK_w}{\pi d^3} \\
\text{where } K_w &= \tfrac{4C-1}{4C+4} + \tfrac{0.615}{C} \doteq \tfrac{1.6}{C^{0.14}} \text{, and} \\
F_a &= \tfrac{F_U-F_L}{2} \\
F_m &= \tfrac{F_U+F_L}{2}
\end{aligned}
$$

| | |
|---|---|
| $F_a$ | :Alternative Force |
| $F_m$ | :Mean Force |
| $F_U$ | :Max Force(30lb) |
| $F_L$ | :Min Force(18lb) |
| D | :Average Coil Diameter |
| d | :wire diameter |
| C | :Spring Index |

Objective function $Obj^F = \frac{1}{SF_f} = \frac{\tau_a}{S_n s} + \frac{\tau_m}{S_u s}$

$Obj^Y = \frac{1}{SF_y} = \frac{\tau_a+\tau_m}{S_y s}$

i.e. min Obj $= K_0C^0.86d^{-(2+A_1)}$

where $K_0^F = 2.04(\frac{F_U-F_L}{C_1(NC)_1^B} + \frac{F_U+F_L}{C_2})$ $C_2$ =160,000

$K_0^Y = \frac{4.07(F_U)}{C_3}$ $C_3$ = 86,550

Constraints:

Surging : For Constrained end spring, the lowest material frequency is $f_n = \frac{1}{2}(Kg_c/w)^{1/2}$

where k = spring rate = $\frac{Gd^4}{8D^3N} = \frac{F_U-F_L}{\Delta}$

W = weight of active coils = $\frac{\pi^2 d^2 DN\rho}{4}$

For Steel, $f_n = \frac{112800(F_U-F_L)}{G\Delta}Cd^{-2}$ and $\rho$: density ($0.28lb/in^3$)

G : Shear modulus ($11.5 \times 10^6 psi$)

$\Delta$ : defelction (0.25in)

The allowable nature frequency of the spring is taken as 13 times the frequency of the applied force, $f_{na}$(Hz)

$$
\begin{aligned}
&f_n \geq 13f_{na} = 500(Hz) \\
Thus,\quad &g_1 : K_1C^{-1}D^2 \leq 1 \quad K_1 = \tfrac{13Gf_{na}\Delta}{112800(F_U-F_L)}
\end{aligned} \tag{2.4}
$$

Buckling : The critical length of spring is a function of boundary conditions. An approximated equation to avoid buckling, conservatively, assuming both ends hinged is $L = Nd(1+a) \leq 11.5k(\frac{D}{2})^2/F_U$

where A= $\frac{a}{d}$ =0.004 , a: clearance under maximum force

$N = \frac{Gd^4}{8D^3k}$

$$
Thus, g_2 : K_2C^{-5} \leq 1 \; K_2 = \tfrac{GF_U(1+A)}{22.3k^2} \tag{2.5}
$$

Maximum Number of Coils : $N_min = 3$

$$
\begin{aligned}
&N \geq N_min \\
Thus,\; &g_3 : K_3C^3D^{-1} \leq 1 \; K_3 = \tfrac{8kN_min}{G}
\end{aligned} \tag{2.6}
$$

Spring Index : $C = \frac{D}{d}$

$$
\begin{aligned}
&4 \leq C \leq 20 \\
Thus,\quad &g_4 : \quad K_4C \leq 1 \quad K_4 = \tfrac{1}{20} \\
&g_5 : \quad K_5C^{-1} \leq 1 \; K_5 = 4
\end{aligned} \tag{2.7}
$$

Pocket Length : When space is limited, the length of spring must be below an upper limit.

$L + Qd \leq L_m = 1.25(in)$ Q:number of inactive coil (2)
L=Nd(1+A)

$$Thus,\, g_6 : K_6 d^2 C^{-3} + L_6 d \leq 1 \; K_6 = \frac{G(1+A)}{8kL_m}, L_6 = \frac{Q}{L_m} \tag{2.8}$$

Upper and Lower Limit on Coil Diameter:

$D + d \leq \overline{OD}$ $\overline{OD}$ : Outside diameter (1.5in)
$D - d \geq \overline{ID}$ $\overline{ID}$ : Inside diameter (0.7in)

$$\begin{aligned} Thus,\, g_7 &: K_7(C+1)d \leq 1 \qquad K_7 = \overline{OD}^{-1} \\ g_8 &: C^{-1} + K_8 C^{-1} d^{-1} \leq 1 \; K_8 = \overline{ID} \end{aligned} \tag{2.9}$$

Upper and Lower Limits on Wire Diameter :

$$\begin{aligned} & 0.004 \leq d \leq 0.25 \\ Thus,\, g_9 &: K_9 d^{-1} \leq 1 \quad K_9 = 0.004 \\ g_{10} &: K_{10} d \leq 1 \quad K_{10} = \frac{1}{0.25} \end{aligned} \tag{2.10}$$

Clash Allowance : The usual recommendation of clash allowance is approximately 10 % of the total spring deflection at the maximum force.

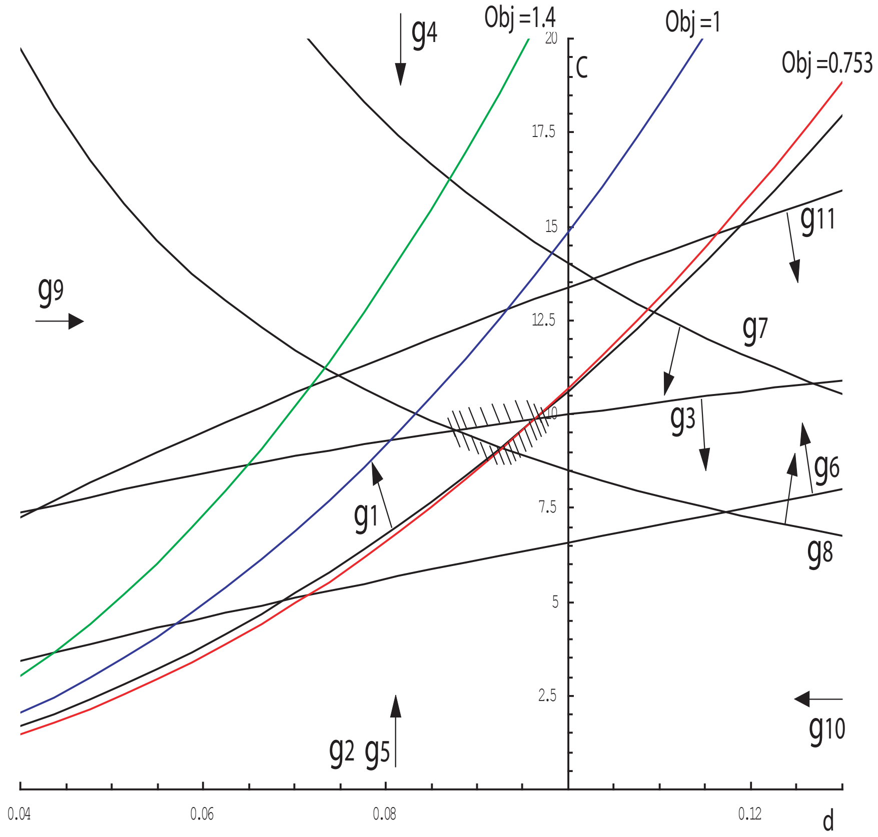
$L_2 = [N(1+A)+Q]d$ : Spring Length at Max Force
$L_S = (N+Q)d$ : Solid Length
$L_2 - L_s = NAd \leq 0.1\Delta$

$$Thus,\, g_{11} : K_{11} d^{-2} C^3 \leq 1 \; K_{11} = \frac{0.8(F_U - F_L)}{AG} \tag{2.11}$$

**Optimization Model**:

| | | | |
|---|---|---|---|
| min | Obj = | $K_0 C^0.86 d^{-(2+A_1)}$ | |
| | $g_1$ : | $K_1 d^2 C^{-1} \leq 1$ | Surging |
| | $g_2$ : | $K_2 C^{-5} \leq 1$ | Buckling |
| | $g_3$ : | $K_3 C^3 d^{-1} \leq 1$ | Min. Coils |
| | $g_4$ : | $K_4 C \leq 1$ | Max. Index |
| | $g_5$ : | $K_5 C^{-1} \leq 1$ | Min. Index |
| | $g_6$ : | $K_6 d^2 C^{-3} + L_6 d \leq 1$ | Pocket Length |
| | $g_7$ : | $K_7(C+1)d \leq 1$ | Outside Diameter |
| | $g_8$ : | $C^{-1} + K_8 C^{-1} d^{-1} \leq 1$ | Inside Diameter |
| | $g_9$ : | $K_9 d^{-1} \leq 1$ | Lower Limit on d |
| | $g_{10}$ : | $K_{10} d \leq 1$ | Upper Limit on d |
| | $g_{11}$ : | $K_{11} C^3 d^{-2} \leq 1$ | Clash Allowance |

Using the Values given for each parameter, we can graphically find the solution.
$C_* = 10$, $d_* = 0.093$(in) and $\overline{SF_f}_* = \frac{1}{0.78} = 1.28$


Obj = 1.4
Obj = 1
Obj = 0.753
C
d
g4
g9
g11
g7
g3
g6
g1
g8
g2 g5
g10
20
17.5
15
12.5
10
7.5
5
2.5
0.04
0.06
0.08
0.12

**2.4**

Sometimes the rate of flow of viscous substances can be estimated by measuring the rate that vortices are shed from an obstacle in the flow. This is the principle behind a vortex meter. A sensor gives a pulse every time a vortex passes and the volumetric rate of flow can be estimated by measuring the pulse rate. The (fictional) data in the table were taken to calibrate such a meter.

Fictional Data Representing the Pulse Rate of a Vortex Meter as a Function of the Velocity of the Fluid Passing the Meter

| **Flow Rate** $V$ | **Pulse Rate** $\rho$ | **Flow Rate** $V$ | **Pulse Rate** $\rho$ |
|---|---|---|---|
| 1.18 | 1.28 | 27.8 | 15.6 |
| 1.45 | 1.65 | 44.0 | 20.0 |
| 1.83 | 2.12 | 72.1 | 25.7 |
| 2.36 | 2.72 | 123.0 | 33.1 |
| 3.14 | 3.49 | 218.8 | 42.5 |
| 4.26 | 4.48 | 407.8 | 54.5 |
| 5.91 | 5.75 | 798.3 | 70.1 |
| 8.39 | 7.38 | 1645.2 | 90.0 |
| 12.1 | 9.49 | 3573.9 | 115.5 |
| 18.1 | 12.1 | 8186.7 | 148.4 |

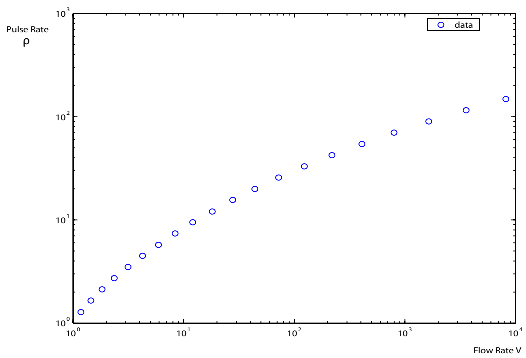
(a) Plot the data on a log–log scale. (b) Fit the data to the equation $\rho = aV^b$. (c) This fit can be improved; specifically, using the relation from (b) employ a neural net as a correction factor, namely, train a small neural net to fit the equation $\rho = \varphi(V)aV^b$ or, more appropriately, find a correction factor that is a function of $V$:

$$\varphi(V) = \frac{\rho}{aV^b}.$$

(d) Using the same log–log graph from part (a), plot the relations from parts (b) and (c), namely, plot $V$ versus $\varphi(V)aV^b$.

**Solution**

a)




b) Fit the data ro the equation $\rho = aV^b \Rightarrow \log\,\rho = \log\,a + b\log\,V$

Applying the least square method,

$$\begin{pmatrix} m & \Sigma\log\,V \\ \Sigma\log\,V & \Sigma(\log\,V)^2 \end{pmatrix}\begin{pmatrix} \log\,a \\ b \end{pmatrix} = \begin{pmatrix} \Sigma\log\,\rho \\ \Sigma(\log\,\rho\cdot\log\,V) \end{pmatrix}$$

$$\Rightarrow \begin{pmatrix} 20 & 72.7645 \\ 72.7645 & 406.5764 \end{pmatrix} \begin{pmatrix} \log a \\ b \end{pmatrix} = \begin{pmatrix} 52.4763 \\ 266.3990 \end{pmatrix}$$

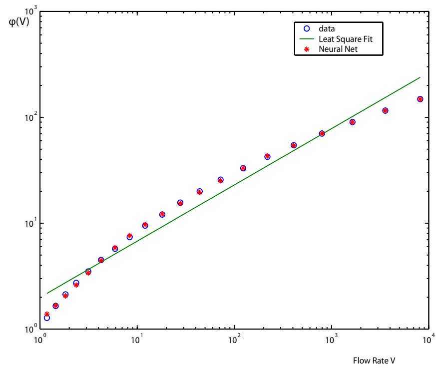
and $\begin{pmatrix} a \\ b \end{pmatrix} = \begin{pmatrix} 1.9894 \\ 0.5321 \end{pmatrix}$

Therefore, the approximated equation by the least square method is

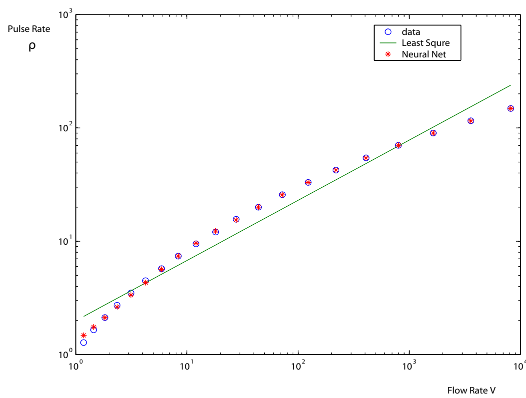
$$\rho = 1.9894 V^{0.5321} \tag{2.12}$$

c)Applying 'train' command in the Matlab, which uses the Levenberg-Marquardt method, to find a correction factor $\varphi(V) = \frac{\rho}{aV^b}$, we obtain

```
%Example Code for Exercise 2.4(c)
p = [1.18 1.45 1.83 ... 3573.9 8186.7];
t = [1.28 1.65 2.12 ... 115.5 148.4];
G=1.98935*p.^0.53128;
tt = t./G
net = newff([0 10000],[20 1],{'tansig' 'purelin'},'trainlm');
net.trainParam.epochs = 60;
net.trainParam.goal = 0.0001;
net = train(net,p,tt);
y2 = sim(net,p)
y3 = y2.*G
loglog(p,t,'o',p,G,p,y3,'*')
```




d) From $\varphi(V) = \frac{\rho}{aV^b}$, we can get a more accurate plot



## 2.5

Consider the case where there is no correlation between any of the data points. (a) If a constant term were used for $f(x)$, what would the kriging model degenerate to? (b) Consider the opposite extreme where there is perfect correlation between data points, say, as in a straight line. What happens to the kriging system?

**Solution**

The kriging metamodel in multidimensional form is $Y(\mathbf{x}) = f(\mathbf{x}) + Z(\mathbf{x})$, and is comprised of two parts: a polynomial $f(\mathbf{x})$ (assuming a polynomial kernel), and a functional departure from that polynomial $Z(\mathbf{x})$. This can be written as,

$$Y(\mathbf{x}) = f(\mathbf{x}) + Z(\mathbf{x}) = \sum_{j=1}^{k} \beta_j f_j(\mathbf{x}) + Z(\mathbf{x}) \tag{2.13}$$

where $f_j(\mathbf{x})$ are the basis functions (i.e. the polynomial terms), $\beta_j$ are the corresponding coefficients, and $Z(\mathbf{x})$ is a stochastic Gaussian process.
Revisiting equation (2.13), the polynomial term of the model comprises of the k×1 vector of regression functions

$$\mathbf{f}(\mathbf{x}) = [f_1(\mathbf{x}), f_2(\mathbf{x}), \cdots, f_k(\mathbf{x})]^T$$

and the associated k×1 vector of constants

$$\beta = [\beta_1, \beta_2, \cdots, \beta_k]^T.$$

We next define the n×k expanded design matrix

$$\mathbf{F} = \begin{pmatrix} \mathbf{f}^t(x_1) \\ \mathbf{f}^t(x_2) \\ \vdots \\ \mathbf{f}^t(x_n) \end{pmatrix}$$

If we notates the stochastic process as

$$\mathbf{z} = [Z(x_1), Z(x_2), \cdots, Z(x_n),]^T$$

then, for the output of the sampling data $\mathbf{y} = [y_1, y_2, \cdots, y_n]^T$, one can rewrite equation (2.13) as

$$\mathbf{y} = \mathbf{F}\beta + \mathbf{Z} \tag{2.14}$$

$$where \quad \{F\}_{ij} = f_j(x_i), \{\beta\}_{\mathbf{j}} = \beta_{\mathbf{j}} \tag{2.15}$$

Using maximum likelihood estimation, we estimate the coefficient $\beta$ as

$$\hat{\beta} = (\mathbf{F}^T\mathbf{R}^{-1}\mathbf{F})^{-1}\mathbf{F}^T\mathbf{R}^{-1}\mathbf{y} \tag{2.16}$$

and therefore the new point of interest $\mathbf{x}$ has an estimate value

$$\hat{y}(x) = \mathbf{f}^T\beta + \mathbf{r}^T(x)\mathbf{R}^{-1}(\mathbf{y} - \mathbf{F}\hat{\beta}) \tag{2.17}$$

If a constant term was used for $\mathbf{f}(\mathbf{x})$ (e.g. f(x)=$\beta$) and $\mathbf{F} = \begin{pmatrix} 1 \\ \vdots \\ 1 \end{pmatrix} = \mathbf{1}$, the equation (2.17) reduces to equation (2.28) in the text.

$$\hat{y}(x) = \beta + \mathbf{r}^T(x)\mathbf{R}^{-1}(\mathbf{y} - \beta\mathbf{1}) \tag{2.18}$$

where

$$\beta = (\mathbf{1}^T\mathbf{R}^{-1}\mathbf{1})^{-1}\mathbf{1}^T\mathbf{R}^{-1}\mathbf{y} \tag{2.19}$$

Because of no data correlation, $\mathbf{R}$ is equal to $\mathbf{I}$. Then from equation (2.19), $\beta = \mathbf{1}^T\mathbf{y}$ and $\hat{\mathbf{y}}(x) = \beta$. That means, the model degenerates to the polynomial coefficient estimate.

b) If all data points are perfectly correlated, the matrix $\mathbf{R}$ has linearly dependent columns and is, therefore, singular, i.e. cannot be inverted. Thus the kriging "system" cannot be solved.

## 2.6

Derive the expressions for the midpoint and slope of the midpoint given in Figure 2.6.

**Solution**
We note that Figure 2.6 is the logistic function $f(x) = \frac{1}{1+e^{-(wx-b)}}$, where $w$ is a slope parameter and $b$ is a bias/offset parameter. The logistic function is a common function used in neural network neurons, as well as a number of other fields due to its nice analytic properties. Specifically, the derivative (or gradient in its multivariate form) include the original logistic function, i.e., $\frac{df}{dx}f(x) = f(x)(1-f(x))$.

To solve for the midpoint of the logistic function, we note that this function is symmetric and the infinite limits bound the range within $[0, 1]$, leading to a midpoint at $f(x) = 0.5$. Solving for this midpoint,

$$0.5 = \frac{1}{1+e^{-(wx-b)}} \tag{2.20}$$

$$0.5e^{-(wx-b)} = 0.5 \tag{2.21}$$

$$e^{-(wx-b)} = 1 \tag{2.22}$$

$$\log(1) = b - wx \tag{2.23}$$

$$x = \frac{b}{w} \tag{2.24}$$

To solve for the slope of the logistic function at the midpoint, we first find its derivative,

$$\begin{aligned}
\frac{df}{dx}f(x) &= \frac{df}{dx}[e^{(wx-b)}(1+e^{(wx-b)})^{-1}] \\
&= e^{(wx-b)}(1+e^{(wx-b)})^{-1} - e^{2(wx-b)}(1+e^{(wx-b)})^{-2} \\
&= \frac{e^{(wx-b)}(1+e^{(wx-b)}) - e^{2(wx-b)}}{(1+e^{(wx-b)})^2} \\
&= \frac{e^{(wx-b)}}{(1+e^{(wx-b)})^2} \\
&= f(x)(1-f(x))
\end{aligned}$$

Since the value of the logistic function at the midpoint is 0.5, we can see that the slope at the midpoint is then $0.5(1-0.5) = 0.25$.