Regression Analysis

3 Aug 2012

(I) Problem

We have a set of N data points $\mathcal{D} = \{(x_1, y_1), (x_2, y_2), \ldots, (x_N, y_N)\}$. The idea is to come up with a best approximation fucntion that would predict the value of y for a given value of x. The nature of the underlying function that defines the data or the details of how the data was generated is usually unknown. These data points might be totally random or may be related by a characteristic function.

(II) Theory

In the absence of the underlying function, the data is treated to be a linear combination of some set of functions.

$$y = \sum_{i=1}^{M} w_i \phi_i(x) \tag{1}$$

$$= \bar{w}^T \bar{\phi}(x) \tag{2}$$

where M is the number of functions, $\bar{w} = [w_1 w_2 \ldots w_M]^T$, and $\bar{\phi}(x) = [\phi_1(x)\phi_2(x)\ldots\phi_M(x)]^T$. As a particular case, the set of functions $\{\phi_(x)\}$ belong to a set of orthogonal basis collection of functions. Any two functions in this orthogonal basis set satisy the following two criteria:-

$$\int_a^b \phi_i(x)\phi_j(x)dx = \begin{cases} 1, & \text{if } i = j, \\ 0, & \text{otherwise.} \end{cases}$$

(1) is a *linear model* for regression. Approximating the output values y_n using the linear model results in an error. Minimizing this error forms the backbone of this approach. The error in approximation is given by $(y_n - \hat{y_n})^2$ where $\hat{y_n}$ is the estimated y value. The combined error for all N data points can then be written as

$$\mathcal{E} = \sum_{n=1}^{N} (y_n - \hat{y_n})^2 \tag{3}$$

The error given a M and $\bar{w}$ is

$$\mathcal{E}(\bar{w}) = \sum_{n=1}^{N} (y_n - \sum_{i=1}^{M} w_i \phi_i(x_n))^2 \tag{4}$$

The optimal set of weights are those that minimize $\mathcal{E}(\bar{w})$

$$\bar{w}^* = \underset{\bar{w}}{\operatorname{argmin}}\, \mathcal{E}(\bar{w})$$

Differentiating $\mathcal{E}(\bar{w})$ with respect to $\bar{w}$, we have

$$\begin{aligned}\frac{d}{d\bar{w}}\mathcal{E}(\bar{w}) &= \frac{d}{d\bar{w}}\sum_{n=1}^{N}(y_n - \bar{w}^T\bar{\phi}(x_n))^2 \\ &= 2\sum_{n=1}^{N}(y_n - \bar{\phi}(x_n)^T\bar{w})\bar{\phi}(x_n)\end{aligned}$$

Now,

$$\frac{d}{d\bar{w}}\mathcal{E}(\bar{w}) = 0 \Rightarrow \sum_{n=1}^{N}(y_n - \bar{\phi}(x_n)^T\bar{w})\bar{\phi}(x_n) = 0$$

$$\therefore \sum_{n=1}^{N} y_n\bar{\phi}(x_n) = \sum_{n=1}^{N}\bar{\phi}(x_n)^T\bar{w}\bar{\phi}(x_n) \tag{5}$$

If $\bar{y} = \begin{bmatrix} y_1 \\ y_2 \\ . \\ . \\ . \\ y_N \end{bmatrix}$ and $\Phi = \begin{bmatrix} \bar{\phi}(x_1)^T \\ \bar{\phi}(x_2)^T \\ . \\ . \\ . \\ \bar{\phi}(x_N)^T \end{bmatrix}_{N\times M}$, then (5) can be expressed as

$$\begin{aligned}(\Phi^T\Phi)\bar{w} &= \Phi^T\bar{y} \\ \bar{w} &= (\Phi^T\Phi)^{-1}\Phi^T\bar{y}\end{aligned} \tag{6}$$

(III) Data Generation

- *Generating X's:* The range from which the x values need to be generated is defined at runtime via command-line arguments using the parameters `low` and `high`. Number of samples (`nsamples`) is also specified at runtime. Using a random data generator, these x values are obtained.

- Corresponding to a particular function that is also specified at runtime, the function values $f(x)$ for the respective x's are computed.

- *Generating Y's:* To the previously generated $f(x)$ values, some amount of *Gaussian noise* is added to account for any errors in the actual experiment conducted.

$$y = f(x) + \epsilon \quad \text{and} \quad \epsilon \sim \mathcal{N}(\mu, \sigma)$$

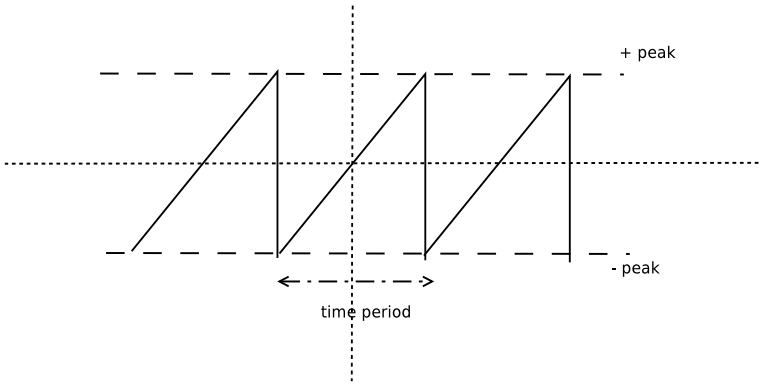

Figure 1: Sawtooth function

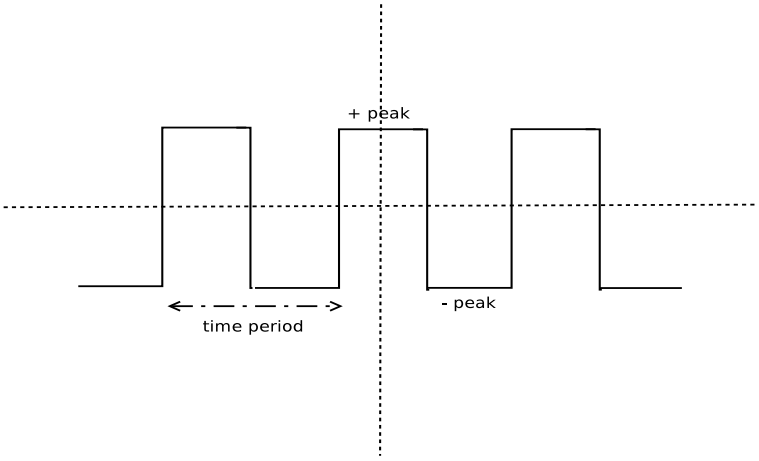

Figure 2: Square function

(IV) Experiment
Tests are done by simulating data from two functions in particular, namely, the `sawtooth` and `square` functions. Each of these functions can be represented using an infinite Fourier series representation.

As a particular test case of the program, I generated data values(x's) from a range (not randomly but sequentially). The data is sampled at fixed intervals. To this data, the corresponding function values are computed (shown in red). To this set of points, a regression fit using a certain number of terms is performed. The blue colored curves are the regression fits for the corresponding functions. From the plots, it can be seen how the linear least squares model fits the actual data. Increasing the number of terms approximates the original function in a better sense.

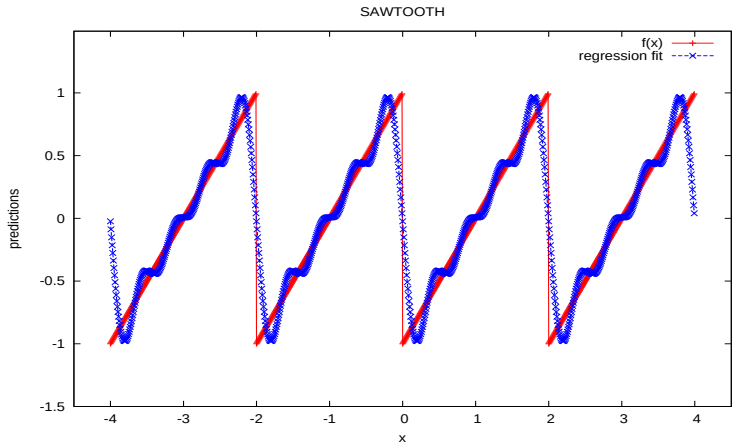

Figure 3: Regression fit for *sawtooth* wave using 7 terms and $\sigma = 0.1$

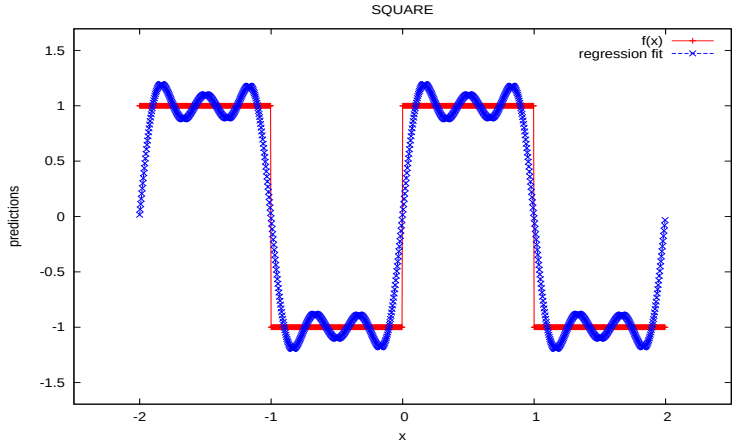

Figure 4: Regression fit for *square* wave using 9 terms and $\sigma = 0.1$

(V) Computing Message Length

The message length described using the *Wallace Freeman* approach for data sampled from a Normal distribution is given by

$$\frac{1}{2}\log\frac{N\sigma^2}{N-1}+\frac{1}{2}(N-1)-\frac{1}{2}N\log\frac{2\pi}{\epsilon^2}+\frac{1}{2}(2N^2)+\log(R_\mu R_\sigma)+1+\log(K_2) \quad (7)$$

where N – number of samples
ϵ – accuracy of measurement
R_μ– range of *mean* of normal distribution
R_σ– range of $\log(\sigma)$ of normal distribution

The Message Length has two components. The first part of the message comprises transmitting the number of terms and the weights. The second part involves sending the y values. The encoding of x values may be

included in the fist part as this does not affect the result when we are comparing two competing hypotheses.

- *Encoding* X: In the experiment, x's are drawn from a predefined range `[a,b]`. The x values are sorted in increasing order. The first term of this sorted sequence is made 0. This is done so that the first x value sent is always 0 and this is part of the code book.

  Instead of sending the x's, what is sent is the difference Δx between consecutive x values. The Δx values are sent over a Gaussian channel. So the corresponding parameters as per (7) need to be estimated to compute the message length.
  (i) To estimate $R_{\mu_{\Delta x}}$

$$
\begin{aligned}
x \in [a,b] &\Rightarrow a \leqslant x \leqslant b \\
\therefore a \leqslant x_i \leqslant b \quad &\text{and} - b \leqslant -x_j \leqslant -a \\
\text{If} \quad \Delta x = x_i - x - j, \quad & a - b \leqslant \Delta x \leqslant b - a \\
\therefore a - b \leqslant \mu_{\Delta x} &\leqslant b - a \qquad (8)
\end{aligned}
$$

  (ii) To estimate $R_{\log(\sigma_{\Delta x})}$

$$
\begin{aligned}
\sigma^2_{\Delta x} &= \sum_{i=1}^{N-1} \frac{(\Delta x_i - \mu_{\Delta x})^2}{N-1} \\
\text{Consider} \quad (\Delta x_i - \mu_{\Delta x})^2 &= \Delta x^2 + \mu^2_{\Delta x} - 2\Delta x \mu_{\Delta x} \\
a - b \leqslant \Delta x \leqslant b - a &\Rightarrow 0 \leqslant \Delta x^2 \leqslant (b-a)^2 \qquad (9) \\
\text{and} \quad a - b \leqslant \mu_{\Delta x} \leqslant b - a &\Rightarrow 0 \leqslant \mu^2_{\Delta x} \leqslant (b-a)^2 \qquad (10) \\
\text{Also} \quad -2(b-a)^2 &\leqslant -2\Delta x \mu_{\Delta x} \leqslant 2(b-a)^2 \qquad (11) \\
\text{Adding} \quad (9),(10),(11) &\Rightarrow -2(b-a)^2 \leqslant (\Delta x_i - \mu_{\Delta x})^2 \leqslant 4(b-a)^2 \\
\therefore 0 \leqslant \frac{(\Delta x_i - \mu_{\Delta x})^2}{N-1} &\leqslant \frac{4(b-a)^2}{N-1} \\
\therefore 0 \leqslant \sigma^2_{\Delta x} &\leqslant \frac{4(b-a)^2}{N-1} \\
\therefore 0 \leqslant |\sigma_{\Delta x}| &\leqslant \frac{2(b-a)}{\sqrt{N-1}} \qquad (12)
\end{aligned}
$$

  From (8), $R_{\mu_{\Delta x}} = 2(b-a)$, and
  From (12), upper bound of $\log(\sigma_{\Delta x}) = \log \frac{2(b-a)}{\sqrt{N-1}}$, and lower bound of $\log(\sigma_{\Delta x})$ is dependent on ϵ, the accuracy of measurement. Hence, the lower bound is set to 3ϵ.
  $\therefore \qquad R_{\log(\sigma_{\Delta x})} = \log \frac{2(b-a)}{\sqrt{N-1}} - \log(3\epsilon)$

  Using these values of $R_{\mu_{\Delta x}}$ and $R_{\log(\sigma_{\Delta x})}$ in (7), the message length to encode Δx is computed.

- *Encoding number of samples and number of functions*: The maximum number of data samples is N_{max} and the maximum number of terms is M_{max}. The two integers are assumed to be drawn from a

uniform distribution over their respective ranges; so $N \in [1, N_{max}]$ and $M \in [1, M_{max}]$. These two numbers together can be transmitted in $\log_2(N_{max}M_{max})$ bits. In the experiment N_{max} is set to 100000 and M_{max} is set to 100.

- *Encoding weights*: The Fourier analysis of the sawtooth and square functions determines the weights and their absolute value is always less than 1. This means $-1 < \mu < 1$, and hence $R_\mu = 2$ for the weights. Also, σ_{max} is set to 1 and σ_{min} is set to 3ϵ. The weights are then transmitted over a Gaussian distribution.

- *Encoding Y*: The second part of the message involves transmitting the y values. Since in the first part of the message the weights and x values are transmitted, the receiver can construct the estimated $\hat{y}$ values. So it is enough if thr transmitter send the deviation from the original y values. Hence, Δy's are sent over the Gaussian channel. In this case, σ_{max} is set to 2 and σ_{min} is set to 3ϵ. The Δy's are then transmitted over a Gaussian distribution.