



Figure 7: Regression fit for square wave using $M = 10$ terms and $\sigma = 0$

10 terms are used in the approximation. When 4 terms are used, the contribution is only due to the $\sin x$ (just 1 peak) term. Hence in Figure 6, a distinct sine-like curve is observed.

Computation of Message Length

Message Length can be thought of as the length of the encryption a transmitter sends across. There is a receiver at the other end of the transmission channel and he decodes the encrypted message. Both the transmitter and the receiver adhere to a codebook which contains some basic things that both agree on. We are interested in lossless transmission and if the message is sent as per a Gaussian distribution, it is referred as a Gaussian channel.

In a MML framework, for a given hypothesis, the message length is computed as the number of bits used in encoding the hypothesis and the data given the hypothesis. A message has two components. In this experiment, the hypothesis refers to the model in consideration. The model is defined by the number of terms and the weights. This is the first component. The second part of the message encodes the data given a particular model.

To transmit a set of observations assuming that they are drawn from a Gaussian distribution, one needs to encode the Gaussian parameters (μ and σ), and then ~~need to~~ encode the observations using the ~~5~~ distribution. This two part message (when computed ~~sums out to be~~ has the following expression using the Wallace Freeman approach.

$$\frac{1}{2} \log \frac{N\sigma^2}{N-1} + \frac{1}{2} (N-1) - \frac{1}{2} N \log \frac{2\pi}{\sigma^2} + \frac{1}{2} (2N^2) + \log(R_\mu R_\sigma) + 1 + \log(K_2) \quad (10)$$

Stuart slower. So the context, I mean talk about what HML is: an information-theoretic approach. That states that very least to the best explanation of observed data is the shortest. Thus, it can be used to compare different fits by ...

Make it into a new section IV and start with a brief intro about what you want to use HML for.

where N - number of observations as before.
 ϵ - accuracy of measurement
 R_μ - range of mean of normal distribution
 R_σ - range of $\log(\sigma)$ of normal distribution
 K_2 - lattice constant \rightarrow give an arbitrary

R_μ and R_σ are dependent on the type of problem one is dealing with. They are usually chosen based on the domain knowledge. As an example, if one is interested in encoding the heights of individuals, one can assume that average height to be between 5 and 6 feet. R_μ would then be $6 - 5 = 1$. One could similarly ~~come up with~~ ~~reasonable~~ estimate for R_σ . In the current experiment, I have shown how to estimate these parameters in (11) and (15).

- Encoding weights: The transmitter needs to send the weights across to the receiver. It is assumed that the weights correspond to values sampled from a normal distribution and hence the message length used to encode these weights is computed as per (10). To calculate the message length, one needs to estimate the parameters of the distribution and also compute R_μ and R_σ to be used in (10).

From the infinite Fourier series representation of sawtooth and square waveforms shown in (2) and (3), one can see that the coefficients (weights) of the sine and cosine terms are of the form $\pm \frac{1}{n}$. This means the magnitude of the weights is always less than 1. This helps in determining the bounds for each weight w_i . Hence $|w_i| < 1$ and $R_\mu = 2$.

To determine R_σ , one needs to estimate the bounds for σ . Now σ cannot be zero as it would mean the deviation from mean is zero and all observations are the same as the mean. Hence σ is bounded from below by ϵ , the accuracy of measurement. The lower bound of σ is set to be a constant times ϵ . The magnitude of the weights decreases as the number of terms increases in the Fourier series representation. The upper bound of σ is taken to be 1.

- Encoding x : In the experiment, the x values are sampled from a pre-defined range $[a, b]$. Further, they are sorted in increasing order. The first term of this sorted sequence is made 0 and this part of the codebook. The other x values are scaled accordingly.

Instead of sending the x 's, what is sent is the difference Δx between consecutive x values. This ~~will~~ enables the receiver to construct the current x value using the previous x value received and the difference Δx . Sending Δx results in a compact message ~~well~~. Hence, information is sent in an efficient manner. Δx 's are sent over a Gaussian channel. The overall message length to encode Δx 's is computed as per (10). However, we still need to estimate R_μ and R_σ .

Not really, you are estimating the entire message ... The entire message ...

points. No need for bullet what data? you have already talked about it. I need to link it to the distribution. The distribution is the value.

Are you going to explain this? You need at the very least to provide an intuition for every term, and for what it corresponds to (a scale number, the entire message ...)

what constant? what you said is wrong? case x before?

ed message. what transmission channel? you have not introduced this before!

(increase the number of terms, and the weights w)

this that the clearly, since this

you need to explain this: not being able to find does not immediately mean being bounded by 2

mean being bounded by 2

I mean, why I found that it decreases?

Explain!

which means it does not need to be transmitted.