An MLAB Example: A Missing Data Imputation Procedure

Zhiping You, Ph.D.
Civilized Software Inc.
12109 Heritage Park Circle
Silver Spring, MD 20906 USA
Phone: (301) 962-3711
Email: csi@civilized.com
URL: www.civilized.com

The general problem of handling missing data in the presence of observed covariates arises in many situations. For example, data on the incubation period of the HIV virus in *AIDS* patients are often censored. One procedure for handling this missing data has been proposed recently by *Gang Chen* and *Grace Yang* [1]. An example of this procedure is given here using the mathematical and statistical modeling software package *MLAB*.

Suppose we have $m + n$ HIV-positive individuals. For n of these individuals, we know both the amount of time that they have incubated the $AIDS$ virus and their current sero-index value ($T4$-cell count). Thus we have the data $(y_1, t_1), \ldots, (y_n, t_n)$ where t_i is the incubation time and y_i is the sero-index value for the ith individual. For the remaining m individuals, we know their current sero-index values $z_1, \ldots, z_m$, but we do not know the associated incubation times $s_1, \ldots, s_m$; these times are left-censored.

Our goal is to estimate the missing data values $s_1, \ldots, s_m$ associated with the covariate values $z_1, \ldots, z_m$. The procedure we use is based on resampling and has utility in a wide range of circumstances.

Let $t_1, \ldots, t_n$ and $s_1, \ldots, s_m$ be samples of the random variables $T_1, \ldots, T_n$ and $S_1, \ldots, S_m$. Let $y_1, \ldots, y_n$ and $z_1, \ldots, z_m$ be samples of the random variables $Y_1, \ldots, Y_n$ and $Z_1, \ldots, Z_m$. Note that, generally, T_i and Y_i are correlated, as are S_j and Z_j, in some unknown manner.

We want to estimate values $s_1, \ldots, s_m$ so that s_i will be a plausible and useful sample of S_i. (The notions of plausible and useful are somewhat dependent upon how the data is to be used.)

In this example, we will always choose s_i from $\{t_1, \ldots, t_n\}$, so no interpolation is involved. Also note we could use partial information on s_i, such as $s_i > h_i$ by merely taking $s_i = h_i$ whenever the generated value turns-out to be less than h_i.

The basic idea is: for any given z_i, we want to choose the corresponding value s_i from $\{t_1, \ldots, t_n\}$ according to the probability $P(S_i = t_j \mid z_i, (y,t))$. We will use the following formula for this conditional probability:

$$P(S_i = t_j \mid z_i, (y,t)) = \frac{K((z_i - y_j)/a_j)}{\sum_{k=1}^{n} K((z_i - y_k)/a_k)}$$

where K is a suitable kernel function and $a_1, a_2, \ldots, a_n$ are kernel-width parameters. Often K is the tent function, *i.e.* $K(x) =$ if $|x| > 1$ then 0 else $1 - abs(x)$. Another useful choice is the *Epanechnikov* kernel function $K(x) = 0.75 \cdot \max(1 - x^2, 0)$

Note that $P(S_i = t_j \mid z_i, (y,t))$ only depends on the values $t_1, \ldots, t_n$ through the index j.

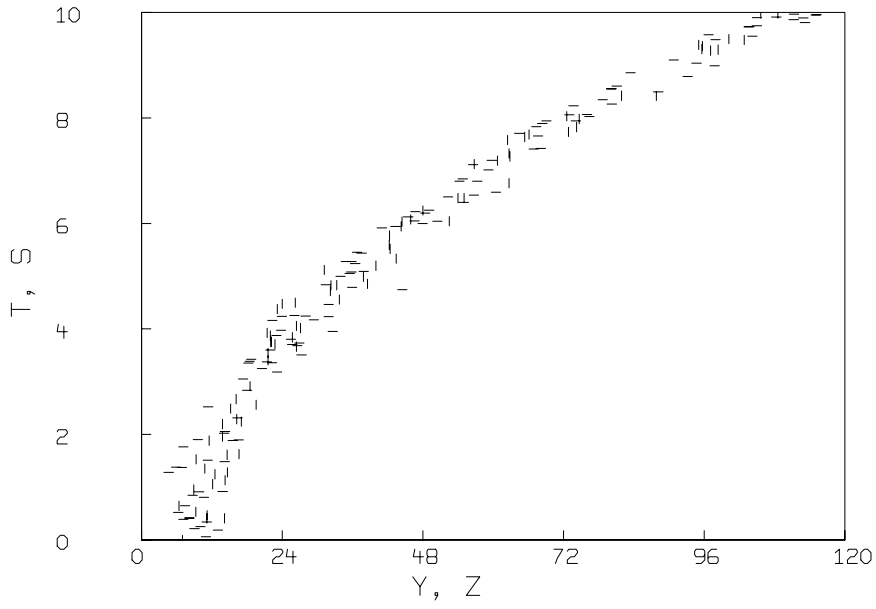
Let $p_{ij} := P(S_i = t_j \mid z_i, (y,t))$, and let $q_{ij} := \sum_{k=1}^{j} p_{ik}$, We can partition the unit interval $[0,1]$ into the intervals $[0, q_{i1}), [q_{i1}, q_{i2}), \ldots, [q_{i,n-1}, q_{in}]$ where $q_{in} = 1$. We now generate a uniform random number v in $(0,1)$ and see which subinterval, $[q_{i,j-1}, q_{ij})$, v falls into, and then choose s_i to be t_j.

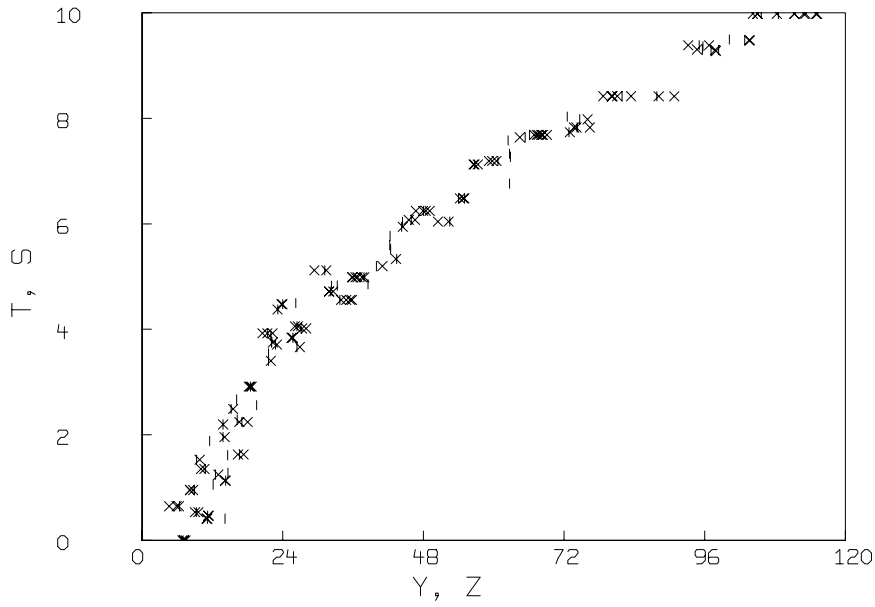
These choices for $s_1, \ldots, s_m$ "repair" the original data and thus solve our problem. Here is an example of this procedure given as an *MLAB do-file*. Note we use varying kernel-widths which are functions of the spacing of the y-observations.

```
/* read-in the Y, T, Z and S1 observed values */
y = read(yfile);
t = read(tfile);
z = read(zfile);
s1 = read(sfile); /* we will see how well we predict these values */
n = nrows(y);
m = nrows(z);

/* compute the varying kernel-width vector av */
v = sort(y &' t); t = v col 2; y = v col 1;
yd = y - rotate(y,1); yd[1] = yd[2];
av = 2*mmean(yd,floor(n/10)) *' mstddev(t,floor(n/8))

fct maxf(a,b) = if (a > b) then a else b
fct k(x) = if abs(x) > 1 then 0 else 1 - abs(x)
```

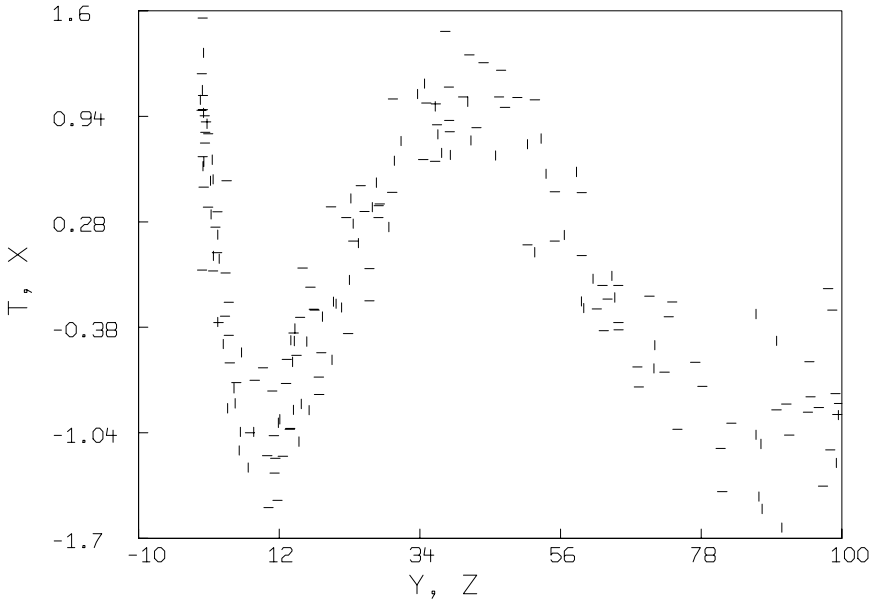
```
/* now, impute s[1:m] */
for i = m:1 do {\
        zv = z[i]; \
        mav = maxf on (av &' minv(abs on (zv-y))); \
        p = k on (zv-y)/'mav; /* generate conditional probabilities p */ \
        p = psum(p) &' (1:n); r = ran(0,0,p[n]); \
        s[i] = t[ceiling(lookup(p,r))]; /*select s[i] according to p*/ \
}
/* now s[1:m] holds the imputed values associated with z[1:m] */

draw y &' t lt none pt hbar ptsize .002 color green; /* fully specified data */
draw z &' s1 lt none pt vbar ptsize .01 color red; /* missing data */
view
delete w
draw y &' t lt none pt hbar ptsize .01 color green; /* fully specified data */
draw z &' s lt none pt xpt ptsize .01 color blue; /* imputed data */
view
```

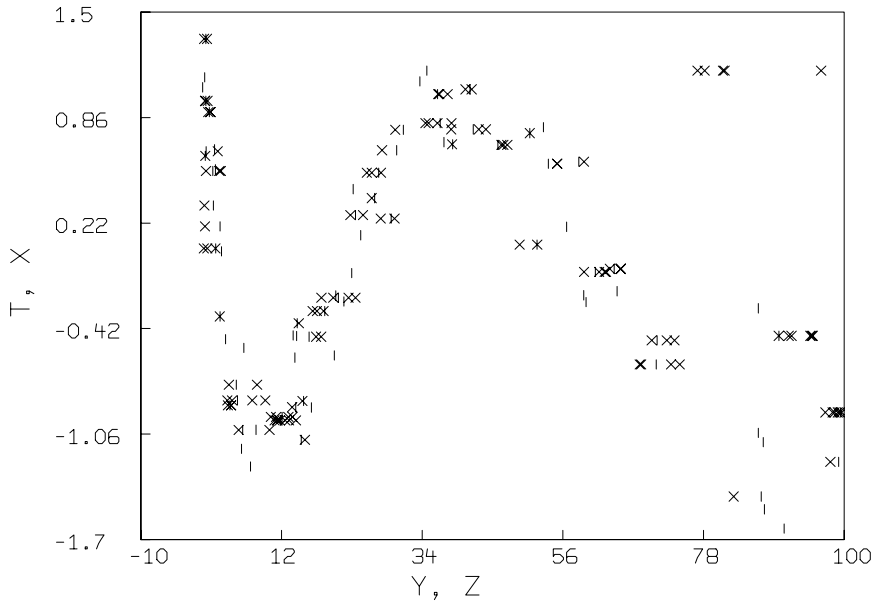
Here are some examples showing the application of this procedure for imputing missing data.





Model function $f(x) = sqrt(x - POISRAN(0,10))$. Left: vbar for original data, hbar for missing data. Right: vbar for original data, cross for imputed data.

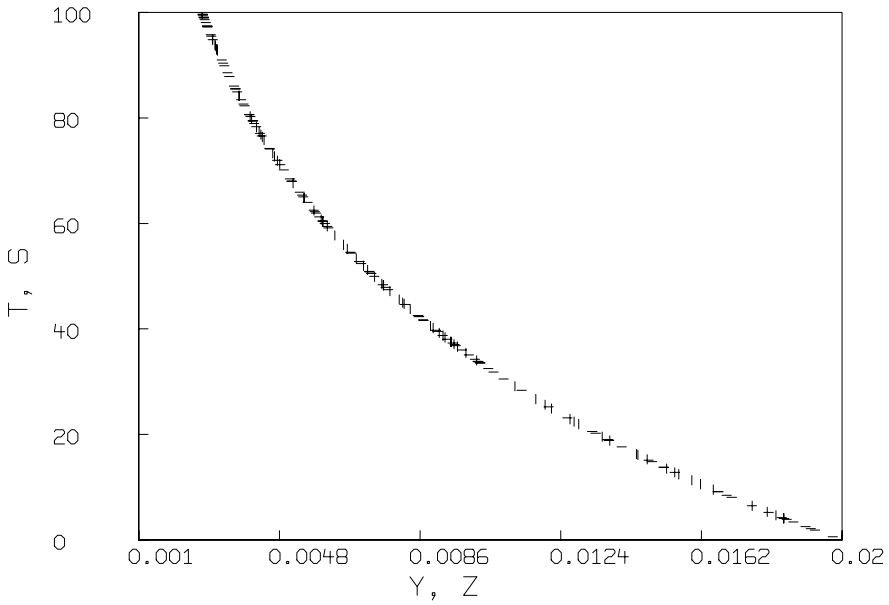


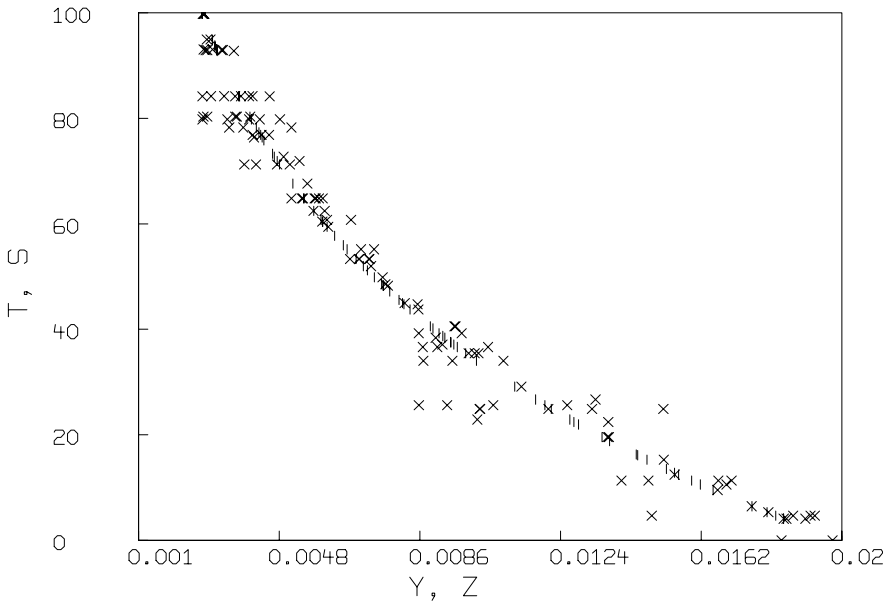


Model function $x(t) = t^2 + 0.1 \cdot NORMRAN(0), y(t) = cos\ \ (t) + 0.3 \cdot NORMRAN(0)$ *Left: vbar for original data, hbar for missing data. Right:*

vbar for original data, cross for imputed data.

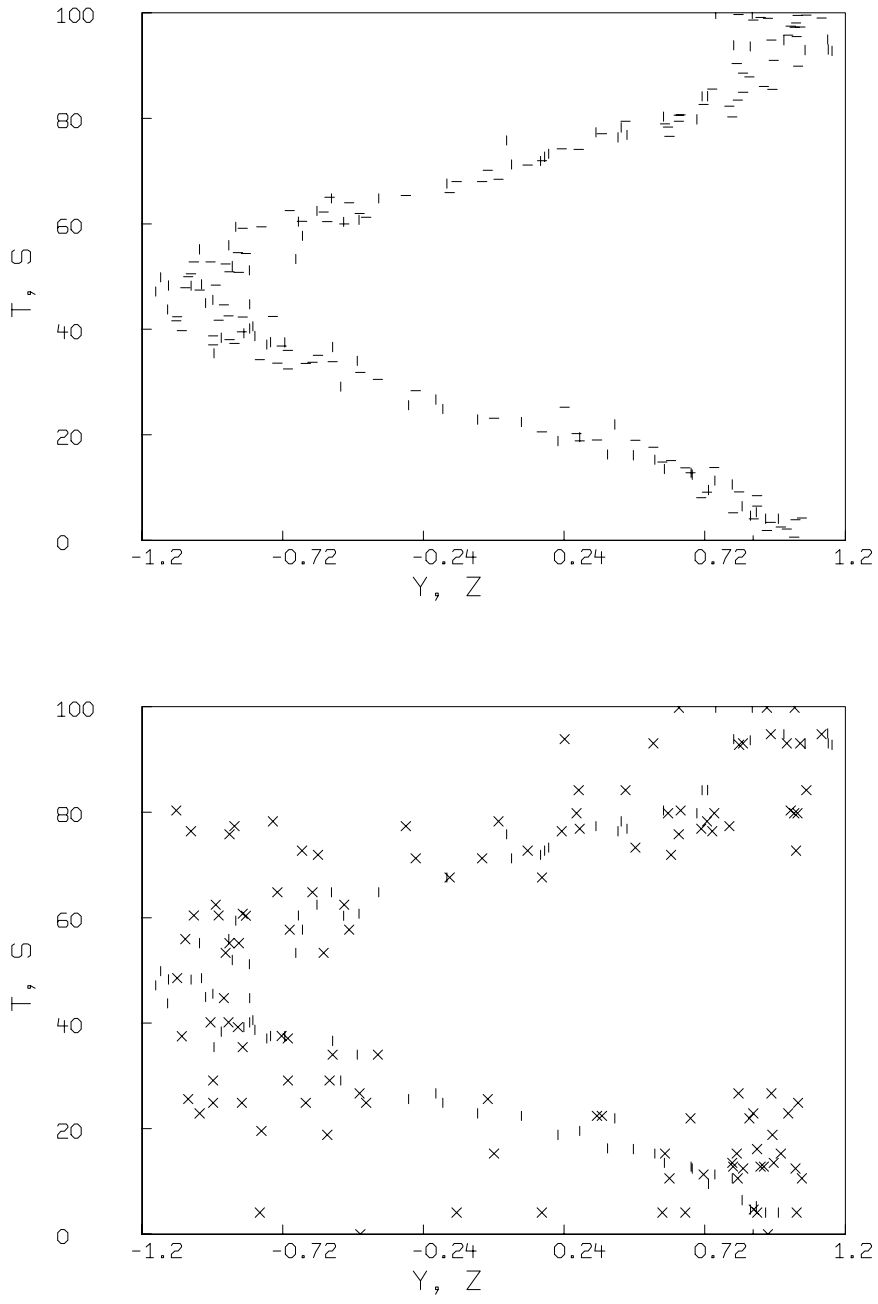
Here is an example of data without error.





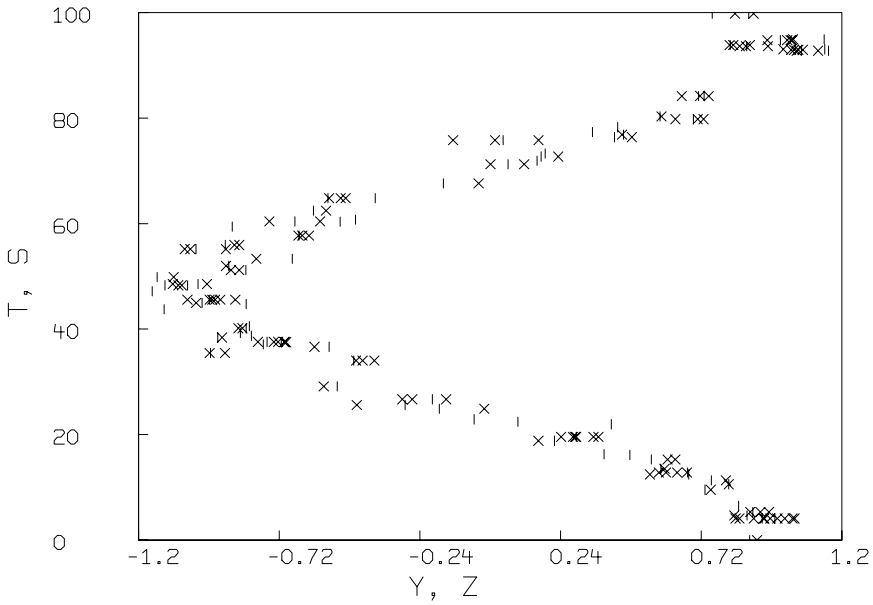
Model function $f(x) = -(log\ x - log\ b)/b$ *where* $b = 0.02$. *Left: vbar for original data, hbar for missing data. Right: vbar for original data, cross for imputed data.*

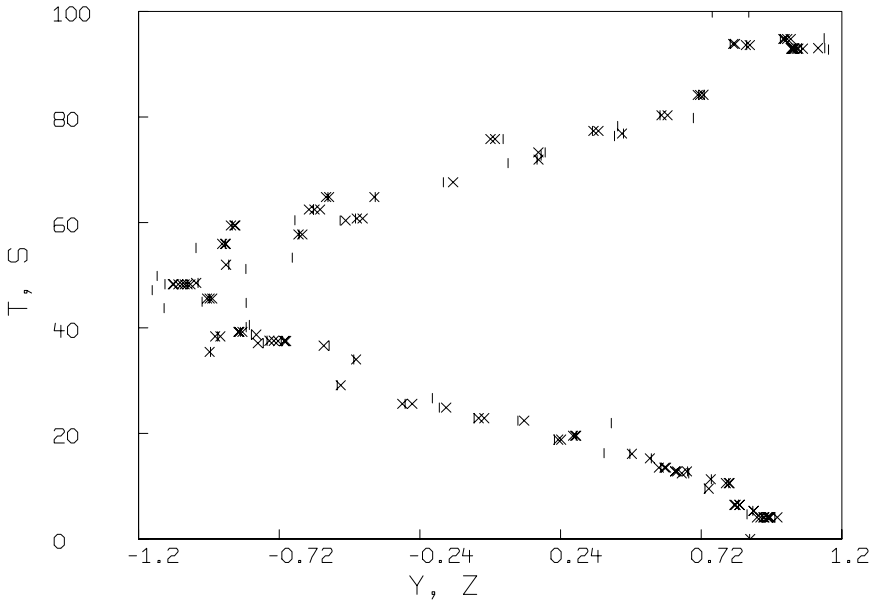
Here is an example based on a non-single-valued function where the conditional distribution of $S_i \mid \{z_i, (y,t)\}$ becomes increasingly bi-modal. In this case the kernel-width estimation method used above fails. The second pair of the following pictures show some results for modified choices of kernel-widths.



Model function $f(x) = 15 \cdot arccos(x - 0.1 \cdot NORMRAN(0))$. *Left: vbar for original data, hbar for missing data. Right: vbar for original data, cross for imputed data. Kernel-widths depend on moving standard deviations of the*

data.





Model function $f(x) = 15 \cdot arccos(x - 0.1 \cdot NORMRAN(0))$. *Kernel-widths*

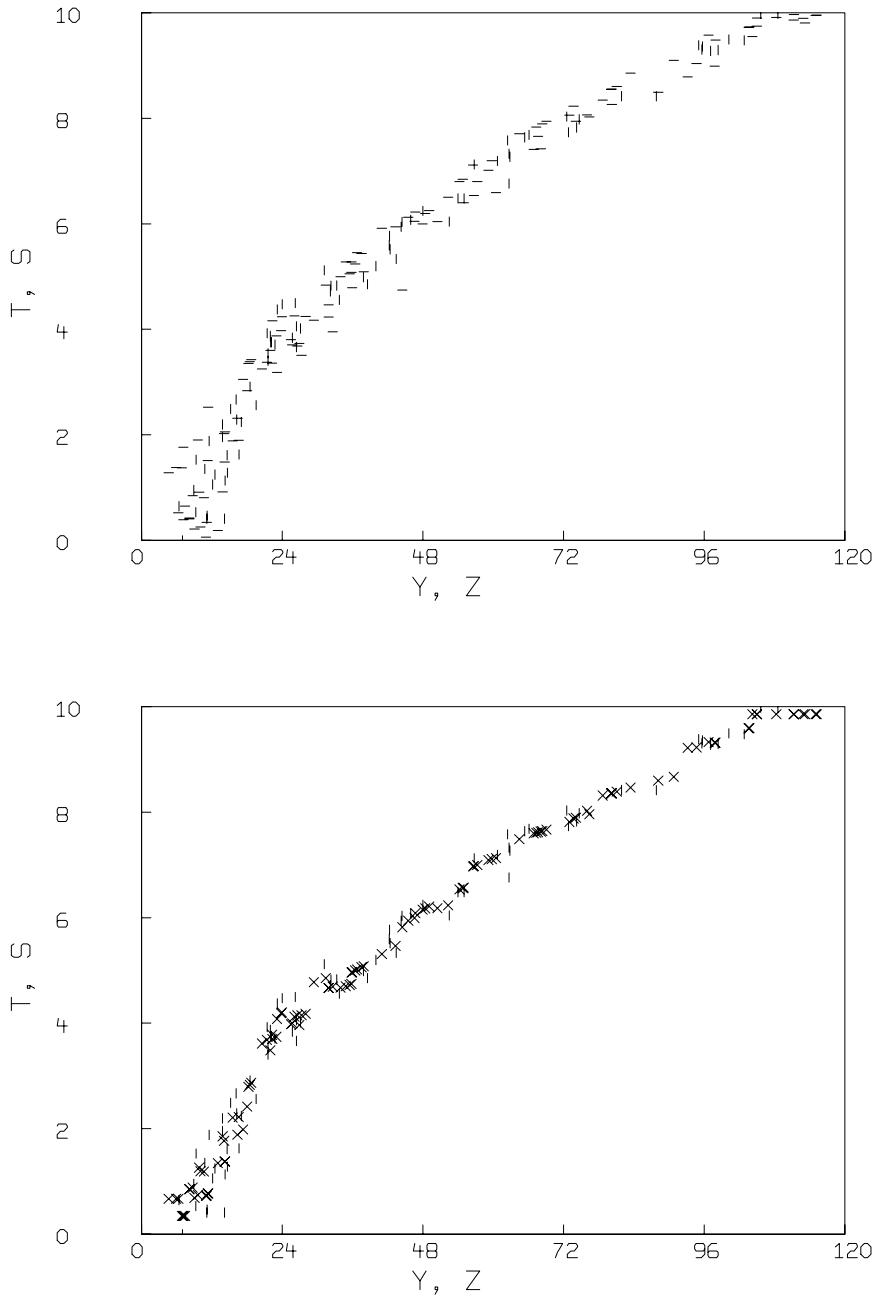
do not depend on the standard deviation of the data. Left: kernel-widths = 5 times moving means. Right: kernel-widths = moving means.

Note that we can produce imputed data values for missing data in an indexed time-series by taking y and z to be the integers $1, 2, \ldots, m+n$, where z specifies those integer times at which data was not recorded.

The noise in the imputed data, viewed as a stochastic process, is dependent upon the widths used in the kernel function K and upon the known values $\{t_1, \ldots, t_n\}$ from which the imputed values are drawn. Often we would like the noise in the imputed data to be similar to the noise seen in the complete observed data. In the limiting case, where the data points are drawn without error from the graph of a smooth single-valued function, imputed values should be obtained by means of some suitable interpolation scheme, IS, such that $IS(z_i)$ produces a value for S_i by interpolating with a smooth function specified by the complete data points $(y_1, t_1), \ldots, (y_n, t_n)$. In general, we can define a smoothing interpolation function specified by the complete data points as the result of some weighted-average procedure. One such choice is to compute $IS(z_i) = E(S_i \mid z_i, (y, t))$, based on our kernel estimate of the conditional distribution for S_i.

In general, we could then choose an imputed data value for S_i as $\alpha C(z_i) + (1-\alpha)\ IS(z_i)$ where $C(z_i)$ denotes the *Chen-Yang* procedure applied to select an imputed value associated with the covariate value z_i. The mixing parameter α can be chosen as a function of the noise percieved in the complete observed data; when no noise is present, $\alpha = 0$, and when no trend is apparent, $\alpha = 1$.

An example of this mixing computation for the first data set given above is shown below where $\alpha = .5$.



Model function $f(x) = -(\log x - \log b)/b$ *where* $b = 0.02$. *Left: vbar for original data, hbar for missing data. Right: vbar for original data, cross for imputed data.*

[1] *A conditional bootstrap procedure for reconstruction of incubation period of aids.* Mathematical Biosciences V117 p253-269