SAW 47 Working Paper 5 (TOR 2b) – Zero-fill Regression Example

15 November 2006

Some More Thoughts on Filling Zeros in Tuning Indices: A Simple Regression Example

by
Chris Legault

Introduction

The problem of zeros in tuning indices is that a lognormal error distribution is assumed. Since the logarithm of zero is undefined these zero tuning indices must be either treated as missing data or else be replaced by a positive value. One objective method to do this is to add 1/6 of the smallest non-zero value in the series to all values. The consequences of these two approaches are considered in a simple regression example.

Methods

A 26 year population time series was simulated with each value varying uniformly between zero and 50,000 fish. A catchability coefficient of 0.0001 was applied to generate the predicted index value. Lognormal noise with exp(std dev) of 0.2 was applied to the predicted values to generate the observed indices. If an observed index was below 0.5, then it was set to zero to mimic the problem of low abundance not being detected. The constant c was determined for each realization as 1/6 of the smallest non-zero value in the observed time series. Four time series of values were created, ln(obs), ln(obs+c), ln(pred), and ln(pred+c) where ln(obs) was missing when the observed value was zero. Two slopes were computed, one for ln(obs) vs ln(pred) denoted "missing" and the other for ln(obs+c) vs ln(pred+c) deonted "add c." Since in both cases the only source of error is the lognormal error assumed around the observed values, the expectation is that both lines will have slope equal to one. Random series of populations and observation errors were drawn 10,000 times and the two slopes computed for each realization.

Results

When zero observations were treated as missing, the slope was slightly negatively biased with mean 0.983 and 90% confidence interval (0.864, 1.109). When a constant of 1/6 the smallest non-zero value was added to all observed and predicted values, the slope was highly positively biased with mean 1.261 and 90% confidence interval (1.018, 1.483). Note that the 90% confidence interval for the "add c" case does not overlap one and has a range nearly twice as large as the "missing" case.

Discussion

The reason for this large disparity between the "missing" and "add c" results can be seen by examining an extreme example of the data used in the regressions (Figure 1). There were five observations that were replaced with c=0.094 causing the five ln(obs+c) values to all be -2.364 even though the associated ln(pred+c) values ranged from -1.265 to -0.307. These points do not fall on the line that would have been fit to the remaining data and are the source of the bias in the results. More typical results followed the same pattern but with less difference between the two slopes.

The constant was added to both observed and predicted data because to ensure an appropriate comparison. In a separate simulation I did not replace values less than 0.5 with zero and found nearly identical distributions for the "missing" and "add c" slopes. This demonstrates that filling of zeros causes the problem, not the addition of a constant.

In order for the "add c" approach to be unbiased, the constant would have to be selected for each realization such that the average of the ln(pred+c) was the same as ln(obs+c) for the values when obs=0. This cannot happen because the predicted values are positive while the observed values are by definition set to zero. Thus, adding a constant to all values when a zero is in the time series will always bias the results.

Conclusion

Filling observed zeros in tuning indices causes a bias relative to the true population that is much greater than the bias introduced by treating the zeros as missing in this simple regression example.

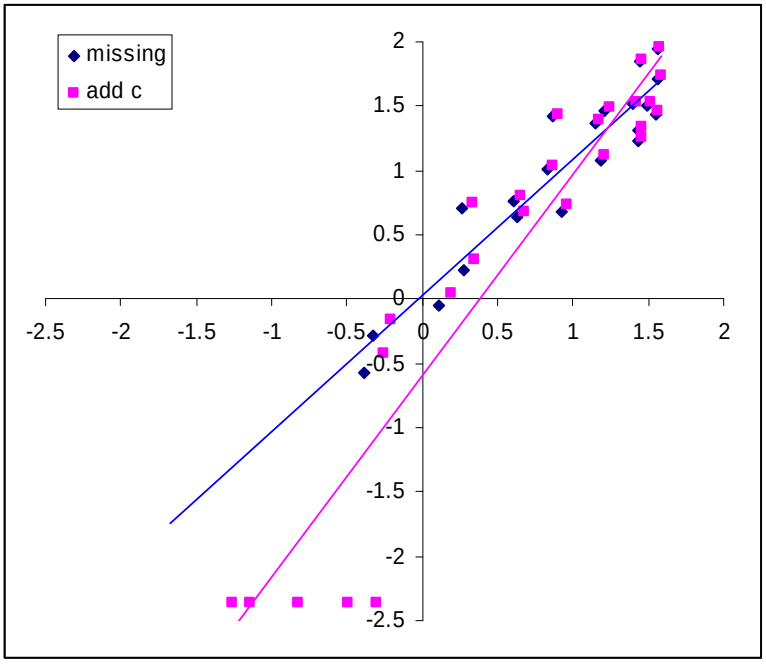

Figure 1. One realization of the "missing" and "add c" regressions. This example is extreme with "add c" slope slightly larger than the upper 90% confidence interval. The x-axis is either ln(pred) or ln(pred+c) and the y-axis is either ln(obs) or ln(obs+c).