APPENDIX: A Note on Listwise Deletion versus Multiple Imputation

February 7, 2018

Extended Simulations

In this section I introduce gradually more complicated data structures, and examine the relative performance of multiple imputation and listwise deletion using similar methods.

Probabilistic Missingness

In the simulations in the main text, missinginess is deterministic: all data for X_2 below a critical threshold are missing. A more realistic scenario would see *probabilistic* missingness in X_2 across the distribution of X_2. If $P(Missing)$ is the same across all values of X_2, of course, then missingness is completely random. To induce probabilistic but nonignorable missingness, I set $P(X_2 = Missing) = \Phi(X_2 + p)$, where $\Phi(\cdot)$ is the CDF of the standard normal. Because $X_2 \sim N(0, 1)$, $X_2 = 0$ has a 50% chance of being missing when $p = 0$, and the probability of missingness decreases as X_2 (or p) increases.

The results of these simulations, across various levels of missingness p, appear in Figure 1.

```
df <- read.csv("by_missingness_prob")
source("make_comparison_plots_coverage.R")
```

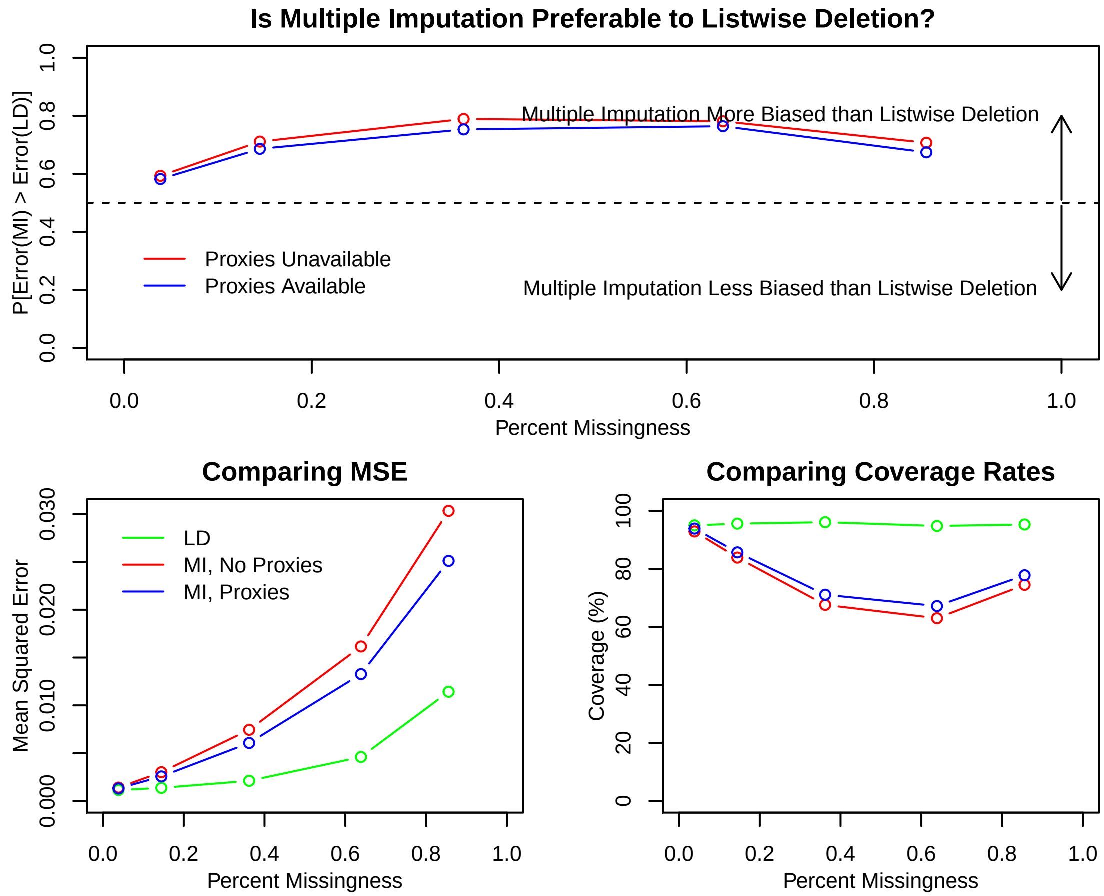

Figure 1: Simulation Results with Probabilistic Missingness

As before, listwise deletion remains superior to multiple imputation, even with proxies, because missingness is still confined to X_2 only. All methods have lower MSE than the simulations above where missingness was deterministic, which makes sense because the distribution of $P(Y|X_2 = Obs)$ is closer to that of $P(Y|X_2 = Missing)$, which is the case because there is now overlap between the distributions of the missing and observed values of X_2. However, for any particular draw of the data, MI is still more likely to generate estimates of β_2 with greater error than listwise deletion. Coverage rates for listwise deletion outperform those for multiple imputation, especially as missingness becomes more common.

Missingness in Y

Missingness in Y also threatens inferences, and analytical results from Allison (2002) and others make clear that listwise deletion will be biased under this procedure. To capture this, I extend the previous simulations by including both probabilistic missinginess in X_2 and probabilistic missingness in Y. The process generating missinginess in X_2 is the same as above. To generate missingness in Y, I set $P(Y = Missing) = 1 - \Phi(N(Y) - p))$, which induces a greater probability of missingness for larger values of Y. Note here that Y is scaled to a standard normal when calculating $P(Y = Missing)$. Note also that missingness in Y is not a function of X_2, conditional on X_2.

The results of these simulations, again across various levels of missingness p, appear in Figure 2.

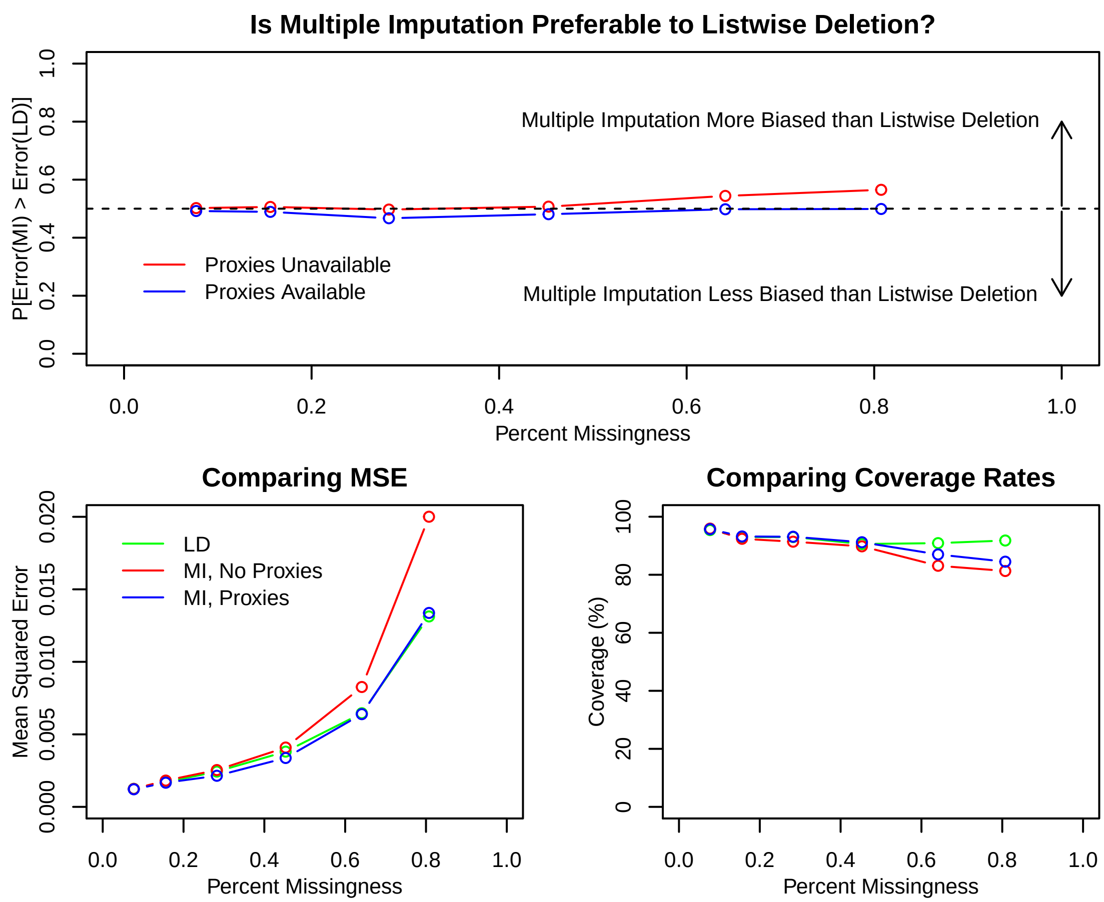

Figure 2: Simulation Results with Missingness in Y

Here we see the first evidence that multiple imputation is not strictly inferior to listwise deletion For low to moderate levels of missingness, the two perform equivalently, regardless of the presence of U_1 and U_2 as proxies. At the highest levels of missingness, the proxies become critical for the performance of multiple imputation, but MI is not superior to listwise deletion even in this case.

Missingness as a Function of a Proxy

An alternative way to generate probabilistic missingness in X_2 is to allow X_2 to be missing as a function not of its own value, but as a function of the value of one of its proxies. To show this, I keep probabilistic missingness in Y, generated by $P(Y = Missing) = \Phi(N(Y)/6)$ to ensure that roughly 1/12 of the values of Y are missing, and then induce probabilistic missingness of X_2 through deterministic missingness in U_1: $P(X_2 = Missing) = 1$ if $U_1 < p$. This kind of data generating process is particularly useful for illustrating the strengths of multiple imputation because now, *missingness itself* in X_2 is perfectly predicted by U_1.

The results by level missingness appear in Figure 3.

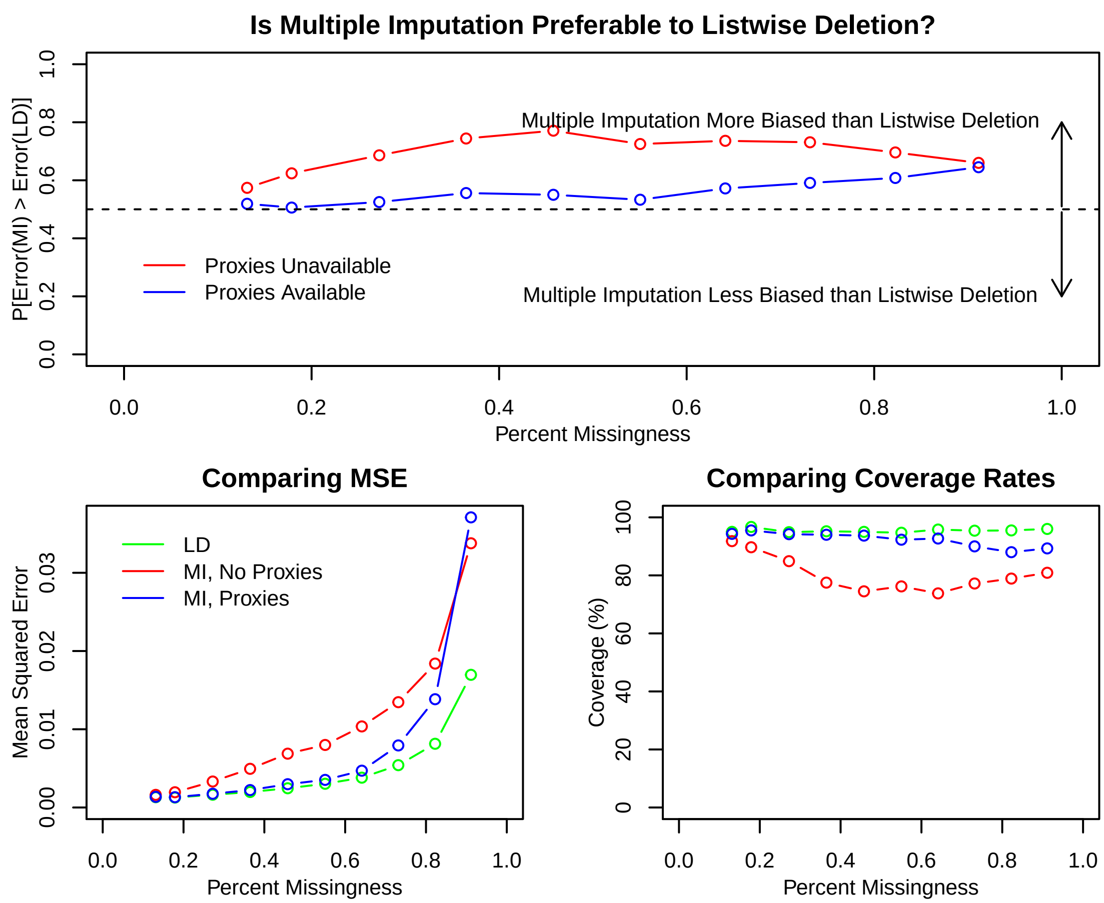

Figure 3: Simulation Results with Missingness as a Function of the Proxy

As expected, this source of missingness makes multiple imputation more dependent on the proxies than we saw in the previous example. Without proxies MI fares significantly worse than listwise deletion, with proxies MI fares worse than listwise deletion but relatively better than without them. Coverage rates diverge as missingness becomes more common. However, at the highest levels of missingness, proxies are no longer much help for multiple imputation.

Correlation between X_1 and X_2

Another extension can allow X_2 to be correlated with X_1, which until now has played no substantive role in the simulations. I do this by simulating X_1, U_1, and U_2 as draws from a multivariate normal with mean vector zero and covariance matrix

$$\mathbf{\Sigma} = \begin{bmatrix} 1 & \frac{V}{2} & \frac{V}{2} \\ \frac{V}{2} & V & 0 \\ \frac{V}{2} & 0 & V \end{bmatrix}$$

As a result, the MI may "borrow strength" in estimating missing values of X_2 and Y from X_1, which is fully observed. Figure 4 illustrates the results.

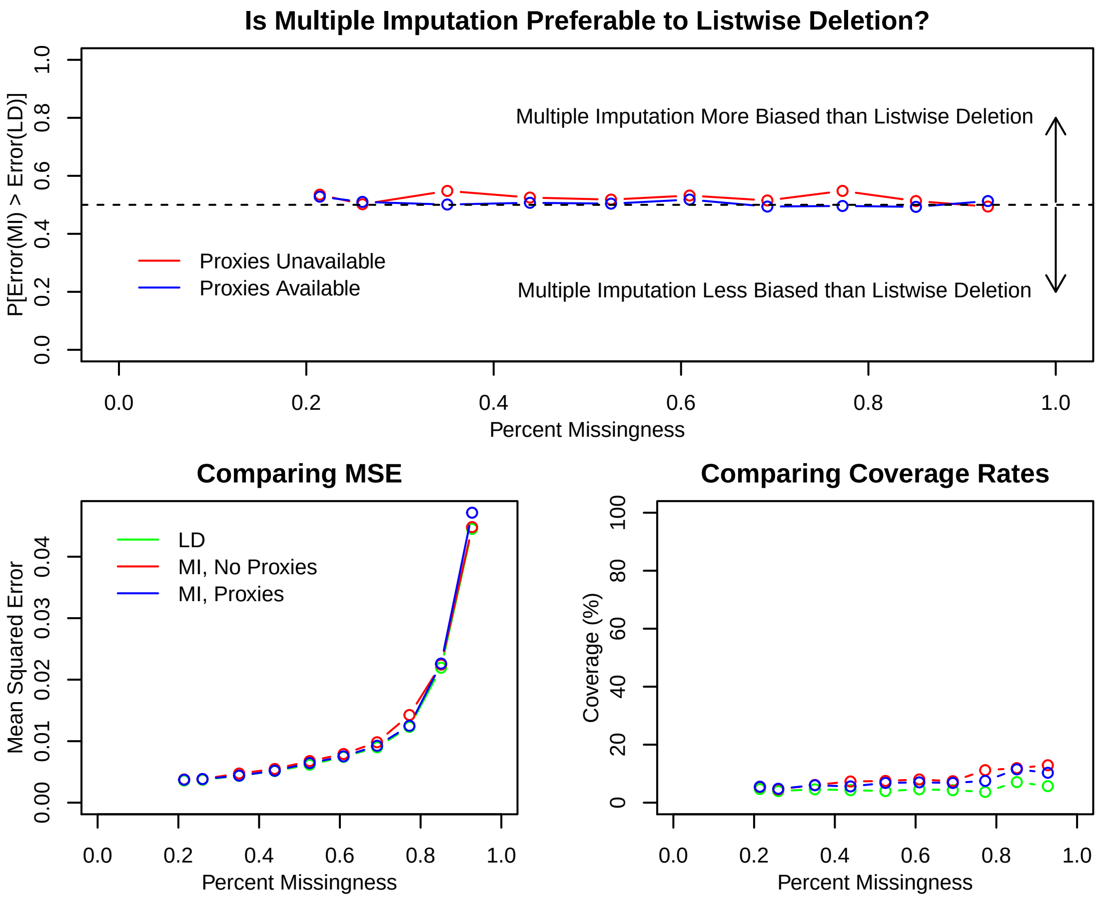

Figure 4: Simulation Results with Correlated X1 and X2

Now, multiple imputation and listwise deletion display roughly equal MSE for any level of missingness, and coverage rates are poor for both methods. Inducing a correlation between X_1 and X_2 helps MI to perform about as well as listwise deletion. The theoretical intuition behind this results is that in cases where listwise deletion is known to be biased (nonignorable missingness in both Y and X), the fact that the data can be represented as a multivariate normal distribution with a fully observed covariate X_1 allows MI to impute values for X_2 better than it otherwise could by exploiting the information about X_2 that is contained in X_1. However, neither β_1 nor β_2 are generally estimated with lower MSE under multiple imputation when compared to listwise deletion. To check how propitious these results really are, in Figure 5 I increase the value of σ_η^2 to 0.5 from 0.2, representing a scenario where the proxies U_1 and U_2 are rather less informative about X_2, and hence more of a departure from MAR data.

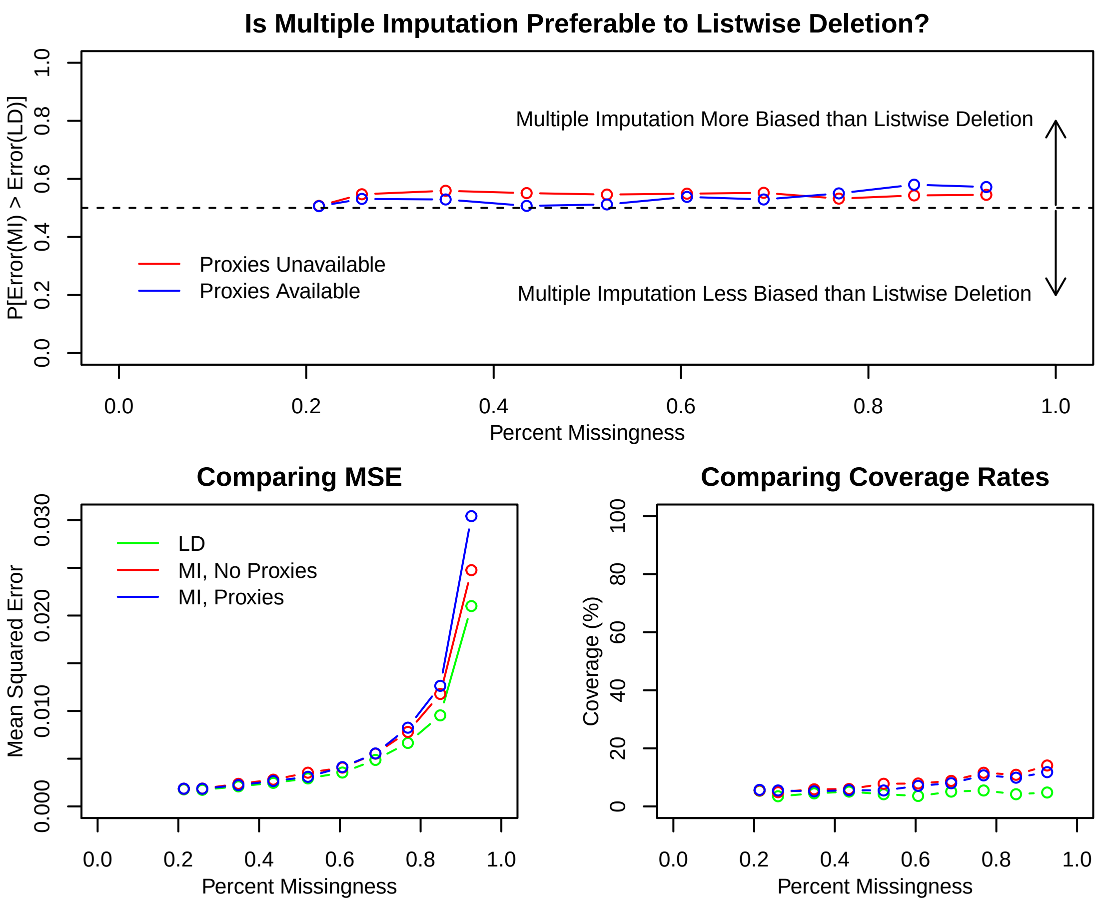

Figure 5: Simulation Results with Correlated X1 and X2 and Less Informative Proxies

In these results, MI systematically fares slightly worse than listwise deletion in terms of MSE, and slightly better than listwise deletion in terms of coverage.

An Exotic Example

Many statistical models to which multiple imputation might be applied are far more complex than the baseline simulations above, with more independent variables, more kinds of missingness, discrete predictors, non-normal distributions, and so forth. Might the messiness of these models allow multiple imputation to shine relative to listwise deletion? In this final exercise I explore one example of a more exotic data generating process to see how MI might perform when it encounters such data in the wild.

Specifically, to the example with missingness in both Y and X_2 and a correlation between X_1 and X_2 describe above, I add

- X_3^* and X_4 are drawn from a bivariate normal distribution with mean vector $0, 100$ and $\rho = .3$
  - X_3 is a dummy variable that takes the value of one when $X_3^* > .5$. It is missing based on the same simulation parameter p as described in the section on Probabilistic Missingness above, but the extent of missingness in X_3 differs between $X_3 = 0$ and $X_3 = 1$: $X_3 = 0$ is missing with probability $p - .05$, and $X_3 = 1$ is missing with probability $p + .15$.
  - X_4 enters the regression model in logarithmic form
- X_5 is drawn from a standard normal distribution. 5% of its values are missing completely randomly.
- X_6 and X_7 are drawn from a bivariate normal distribution with mean vector $100, 0$ and $\rho = .3$.

The model generating Y is $Y = 5X_1 + 5X_2 + 5X_3 + 5log(X_4) + 5X_5 + \epsilon$. Note that X_6 and X_7 are completely unrelated to Y, and the analyst knows that the correct model excludes them. However, both X_6 and X_7 are included in the multiple imputation procedure because the analyst believes that their presence might help, and couldn't hurt.

The results appear in Figure 6. When encountering this particular species of exotic missing data in the wild, MI performs about the same as listwise deletion. It is possible that as data become ever more multidimensional, with 50 or even 100 covariates, then multiple imputation will begin to outperform listwise deletion more systematically. However, there are no theoretical results indicating that researchers may appeal the complexities of exotic real world data to justify the generic superiority of multiple imputation over listwise deletion for nonignorable missing data. There are certainly specific instances of complex real world data where MI will outperform listwise deletion with nonignorable missing data, although I have not found them.

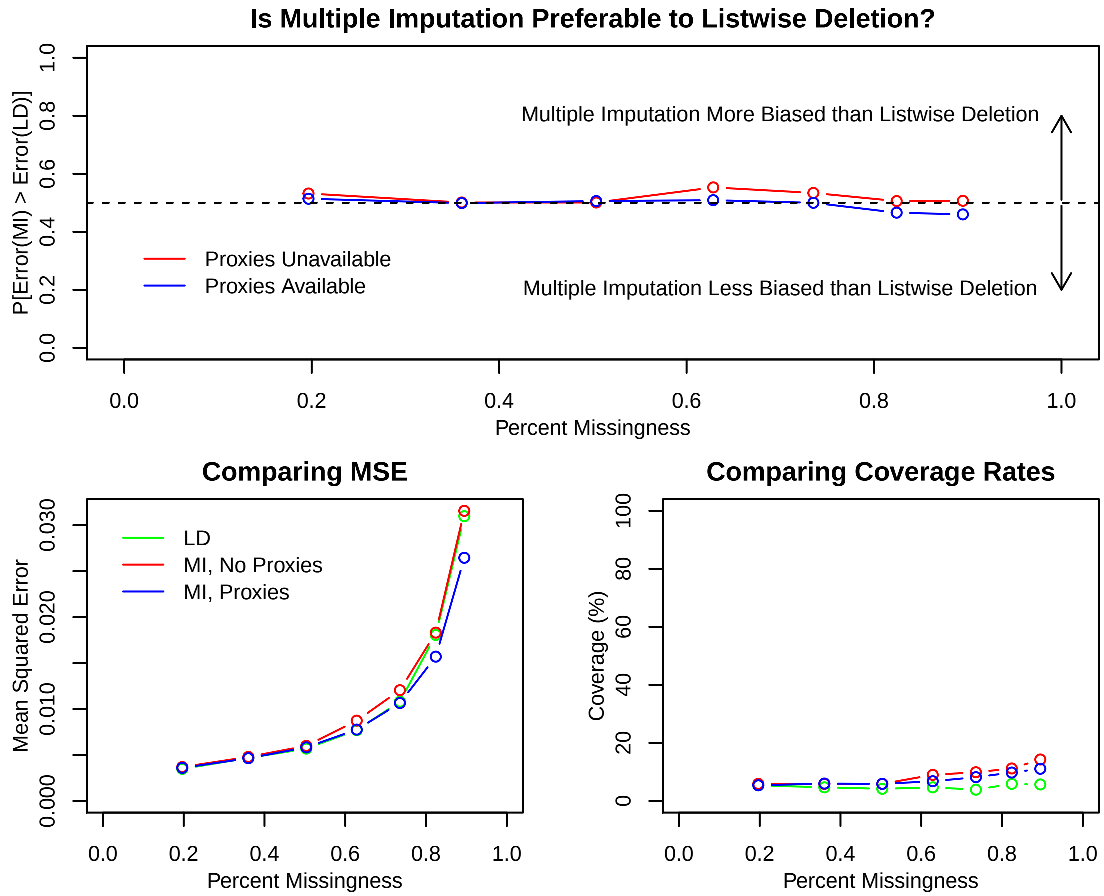

Figure 6: An Exotic Data Generating Process

References

Allison, Paul. 2002. *Missing Data*. SAGE Publications.