Modeling Missing Response

We have a significant portion of observations with no response. But we do have all the predictors for every observation. So just as we built a model to predict the rating, can we build a model to predict whether a rider will give a rating? Going further, could we combine our rating model and nonresponse model into one model, such that our predictions for ratings will take into account biases in nonresponse?

We attempt to address these two questions in this chapter. The methods we present in this chapter could improve inference greatly on this data, but we do have concerns about the current quality of the data for which there are no reponses. As mentioned in Chapter 1, one big problem with data collection is that rides are often misclassified as bike rides when they are actually car rides or rides on public transit. We suspect that many of the unrated rides are rides that were misclassified as bike rides, and thus were not rated by the rider. (We assume that riders don't often go through the effort of correcting the classification of rides and know not to rate rides that weren't bike rides.) That being said, this issue could potentially be fixed and these methods used again in the future.

What could possibly go wrong?

We focus on the situation we have, where our response variable y_i has missing values. Define the vector $R = (r_1, r_2, \ldots, r_n)$ such that

$$r_i = \begin{cases} 1, & \text{if } y_i \text{ is missing;} \\ 0, & \text{if } y_i \text{ is observed;} \end{cases} \tag{1}$$

Rubin classifies missing data into three situations[1]:

1. **Missing Completely at Random (MCAR)**, where R is independent of Y and the predictors X. *i.e.*, $\mathbb{P}(R=1|Y,X) = \mathbb{P}(R=1.)$
2. **Missing at Random (MAR)**, where R is independent of Y, but may depend on X, *i.e.* $\mathbb{P}(R = 1|Y,X) = \mathbb{P}(R=1|Y,)$
3. **Nonignorable, or not MCAR nor MAR**, where R is dependent on Y.

We believe that rider ratings may be correlated with nonresponse. One explination: riders who have a unpleasant experience on the road, might feel more incentive to report their experience on the app than those who had a ride that was uneventful. This motivates our exploration of missing data models, though, of course, is by no means evidence of nonignorable missing data.

If missing data is nonignorable, what could go wrong with our models? Let look at a toy example. Define the data set of n observations with $X \in \mathbb{R}^n$, $Y \in \{0,1\}^n$, and R defined as before, where

$$x_i \sim \text{Normal}(0,1),$$

$$y_i \sim \text{Binomial}(\text{logit}^{-1}(4x_i)),$$

$$r_i \sim \text{Binomial}(0.3 + 0.4y_i),$$

for $i = 1, \ldots, n$.

If we attempt to fit a logistic regression model to this data, our slopes will be unbiased but our estimate of the intercept will be very biased. **??** shows the results of a simulation, computing the slope and intercepts for 1,000 different patterns of missing data for the same generated data set generated from our toy data model.

[1] @little1987 (page 14)

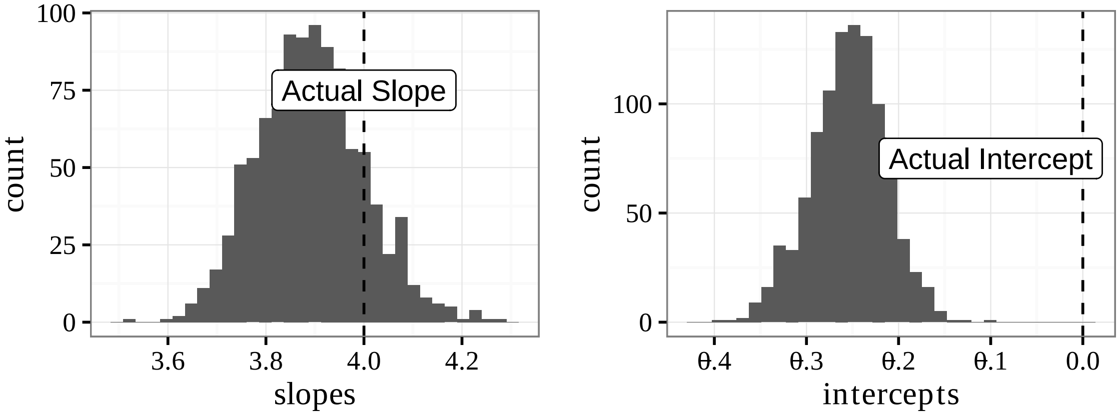

Figure 1: Simulated example of logistic regression fits to a model with nonignorable missing response. One data set of size $n = 10^4$ was computed from the toy data model. We then recomputed R 1,000 times, each time fitting a simple logistic regression model to Y and X.

It makes sense that we are underestimating our intercept. The intercept can be interpretted as the base rate, and if value of $y_i = 1$ are more likely to be missing, the overall rate we observe will be lower.

Clearly, if we have nonignorable missing response, we are in a bad situation. Having missingness depend on Y leads to biased estimates of our intercepts when we fit models. But we do have all of our predictors of y, with no missingness. Could we leverage our understanding of how X predicts Y to understand the patterns of missing response?

Using Expectation Maximization

We can try to perform the expectation maximization algorithm here, using the weighting method proposed by Ibrahim and Lipsitz[2]. Let y_i be our binary response and x_i be our predictors. With these we have our complete data logistic regression model $f(y_i \mid x_i, \beta)$, where β is a vector of parameters in the complete data model.

We then specify a logistic regression model for missingness (the r_i's): $f(r_i \mid x_i, y_i, \alpha)$, where α is the vector of parameters in the missingness model.

We begin the algorithm by getting our first estimates of α and β. We obtain $\beta^{(1)}$ by estimating β with only the non-missing data. We can then estimate the y_i for the missing data using $\beta^{(1)}$, and then use those estimates to compute $\alpha^{(1)}$.

For the E-step, we compute weights for each observation with missing response

$$w_{i\,y_i}^{(t)} = f(y_i \mid r_i, x_i, \alpha^{(t)}, \beta^{(t)}) = \frac{f(y_i \mid x_i, \beta^{(t)}) f(r_i \mid x_i, y_i, \alpha^{(t)})}{\sum_{y_i=0}^{1} f(y_i \mid x_i, \beta^{(t)}) f(r_i \mid x_i, y_i, \alpha^{(t)})}. \tag{2}$$

(For observed responses, $w_{i\,y_i}^{(t)} = 1$.) We can compute $f(y_i \mid x_i, \beta^{(t)})$ and $f(r_i \mid x_i, y_i, \alpha^{(t)})$ by making use of predictions from regression models. So in R, we can fit models and use the `predict()` function to get our probabilities from each of these models.

For the M-step, we find our next estimates of the parameters, $\alpha^{(t+1)}$ and $\beta^{(t+1)}$, by maximizing

[2] @ibrahim1996

$$Q(\alpha, \beta \mid \alpha^{(t)}, \beta^{(t)}) = \sum_{i=1}^{n} \sum_{y_i \in \{0,1\}} w_{iy_i}^{(t)} l(\alpha, \beta | x_i, y_i, r_i). \tag{3}$$

We do this by first by estimating $\beta^{(t+1)}$ using weighted maximum likelihood for the complete data model, and then estimating $\alpha^{(t+1)}$ using the same method. To maximize $l(\alpha, \beta | x_i, y_i, r_i)$, we maximize the product of their likelihoods,

$$l(\alpha, \beta | x_i, y_i, r_i) = l(\beta | x_i, y_i) l(\alpha | r_i, x_i, y_i),$$

which we can maximize by maximize each of the likelihoods separately because our estimates of α and β are independent. This allows us to use any package that can fit models by maximum likelihood estimation using weights for the observations, which includes all of the model fitting packages we used in Chapter 3.

In order to create the data to fit these models, we create an augmented data set where each observation missing the response is recorded as two rows, which represent the two possible values of the response, and also contain the weights computed in the E-step.

Figure 2: Creation of augmented data set for the weighted method of the EM algorithm for missing response data.

Original Data				**Augmented Data**			
y_i	x_i	r_i		y_i	x_i	r_i	w_i
1	2.4	0		1	2.4	0	1
0	1.3	0	→	0	1.3	0	1
NA	-0.4	0		1	-0.4	0	0.2
				0	-0.4	0	0.8

We repeat the E and M step until the values of α and β converge.

As an example, we simulated a dataset from the same model we presented earlier of size 10^4. Of those observations, 6,252 were missing. As shown in Table 1, the estimate for the intercept in the model that only considers the complete data is way off, but the model resulting from the EM algorithm are nearly as accurate as the model computed fit to the full data (with missing values filled in from the original data model.) The missing data model is also able to get accurate estimates of the parameters that define the missing data mechanism, but the estimates are quite uncertain.

Table 1: Coefficients for models fit to simulated data set (± twice the standard error.)

Model	$\hat{\beta}_0$	$2 \cdot SE_{\hat{\beta}_0}$	$\hat{\beta}_X$	$2 \cdot SE_{\hat{\beta}_X}$
Actual	0	—	4	—
Full Data Model	-0.009	0.065	3.881	0.080
Complete Data Model	-0.278	0.106	3.819	0.259
EM Final Model	0.042	0.065	3.814	0.157

Table 2: Estimates for missing data mechanism.

Model	$\hat{\alpha}_0$	$2 \cdot SE_{\hat{\alpha}_0}$	$\hat{\alpha}_Y$	$2 \cdot SE_{\hat{\alpha}_Y}$
Actual	0.3	—	0.4	—
EM Missing Data Model	0.263	0.132	0.530	0.268

Creating a Missing Data Model

Before