Income nonresponse in a opinion survey

An example of `rmarkdown`

Gonzalo Rivero

October, 16 2017

Abstract

In this report we discuss several modeling strategies to analyze item nonresponse in an opinion survey. We try parametric and nonparametric models and we find that the number of contacts is the most useful variable to predict whether someone will provide their income to the interviewer. This result has consequences for the explotation of paradata information in the development of weights.

Contents

1 Introduction **1**

2 The dataset **1**

3 Analysis **3**

3.1 The basic model 3

3.2 A more flexible model 4

3.3 Comparing the three models 5

3.4 Adding external information 6

4 Conclusions **7**

Bibliography **7**

1 Introduction

In this report we perform a analysis of the item-nonresponse to the income question in an opinion survey. The data comes from the survey organization ABC, a polling institute in Europe, and contains information about the demographic attributes and political attitudes of 1500 individuals that were interviewed using quota sampling. In this analysis, we will only use information about the age, gender and habitat of the respondant to better understand who does not provide information about income in a face-to-face survey. Our results are very relevant to the study of sensitive questions.

2 The dataset

Let's start by exploring the dataset. For instance, we could take a look at the first rows to see the structure of the data:

```
head(ceo)
```

```
## # A tibble: 6 x 5
##     age gender habitat income                                id
##   <int>  <int>   <int>  <int>                              <chr>
## 1    18      1       3     10 4b5630ee914e848e8d07221556b0a2fb
## 2    39      1       4      4 c01f179e4b57ab8bd9de309e6d576c48
## 3    39      1       5     99 11946e7a3ed5e1776e81c0f0ecd383d0
```

```
## 4     67       1        4      99 234a2a5581872457b9fe1187d1616b13
## 5     42       1        6      12 dd4ad37ee474732a009111e3456e7ed7
## 6     63       1        2      99 25e6a154090e35101d7678d6f034353a
```

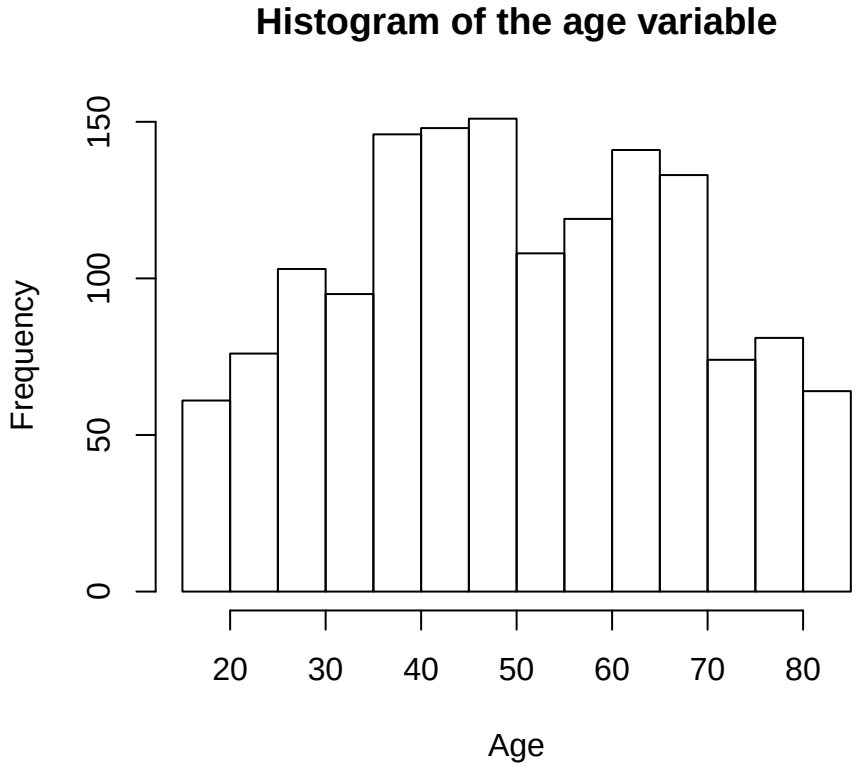
We can also get some summary statistics to get a clearer picture of the data we are trying to analyze. We could use a customized function to get more detailed information but the default `summary` in `R` is sufficient:

```
summary(ceo)
```

```
##       age           gender        habitat         income
##  Min.   :18.0   Min.   :1.00   Min.   :1.00   Min.   : 1.0
##  1st Qu.:37.0   1st Qu.:1.00   1st Qu.:3.00   1st Qu.: 6.0
##  Median :49.0   Median :2.00   Median :4.00   Median : 9.0
##  Mean   :50.5   Mean   :1.52   Mean   :3.86   Mean   :32.4
##  3rd Qu.:65.0   3rd Qu.:2.00   3rd Qu.:5.00   3rd Qu.:98.0
##  Max.   :85.0   Max.   :2.00   Max.   :6.00   Max.   :99.0
##       id
##  Length:1500
##  Class :character
##  Mode  :character
##
##
##
```

We could go to the codebook to check the specific cutoffs of the habitat variable, but suffice to notice that smaller values correspond to smaller populations and that the highest category (category 6) refers to cities with populations above 1 million people.

The age variable goes up to 85. It may be worth recoding the variable to ensure that our modeling is not driven by a few outliers. We could take a look at a histogram to evaluate how many cases are in the higher tail of the distribution:



It makes sense to trim the highest values to, for instance, 85.

```
ceo$age[ceo$age >= 85] <- 85
```

Notice that the income variable is still coded such that the "No answer/Don't Know" categories appear as values 98 and 99. We first start by cleaning the dataset and collapsing these two categories into missing values. Our new variable `R`, following the convention in the literature, will be an indicator that takes value `TRUE` whenever the respondent *has not answered* the income question.

There are several ways of accomplishing this but the easiest way is to encode as a `logical` and then coerce the `logical` to a `factor`. We will print another summary of the data to ensure that it looks correct now.

```
ceo$R <- factor(ceo$income >= 98)
summary(ceo)
```

```
##       age           gender        habitat         income
##  Min.   :18.0   Min.   :1.00   Min.   :1.00   Min.   : 1.0
##  1st Qu.:37.0   1st Qu.:1.00   1st Qu.:3.00   1st Qu.: 6.0
##  Median :49.0   Median :2.00   Median :4.00   Median : 9.0
##  Mean   :50.5   Mean   :1.52   Mean   :3.86   Mean   :32.4
##  3rd Qu.:65.0   3rd Qu.:2.00   3rd Qu.:5.00   3rd Qu.:98.0
##  Max.   :85.0   Max.   :2.00   Max.   :6.00   Max.   :99.0
##       id                R
##  Length:1500        FALSE:1091
##  Class :character   TRUE : 409
##  Mode  :character
##
##
##
```

3 Analysis

3.1 The basic model

Income is known to be a sensitive question in the literature. Tourangeau and Yan (2007) and Yan, Curtin, and Jans (2010) review the literature by survey methodologists on reporting errors in surveys on sensitive topics, noting parallels and differences from the psychological literature on social desirability. As Tourangeau and Yan (2007) put it

> The extent of misreporting depends on whether the respondent has anything embarrassing to report and on design features of the survey. The survey evidence also indicates that misreporting on sensitive topics is a more or less motivated process in which respondents edit the information they report to avoid embarrassing themselves in the presence of an interviewer or to avoid repercussions from third parties.

For our analysis, we will assume that the probability of responding to the survey can be characterized by the following model:

$$y = \begin{cases} 0 & \text{if } \alpha + \beta x + \varepsilon > 0 \\ 1 & \text{otherwise} \end{cases}$$

where $\varepsilon \sim \text{logistic}(0, 1)$. We can estimate this structure using the `glm` function with a `binomial` family (defaults to a `logit` link). Based on previous research, we decided to transform the age variable using a second-order polynomial.

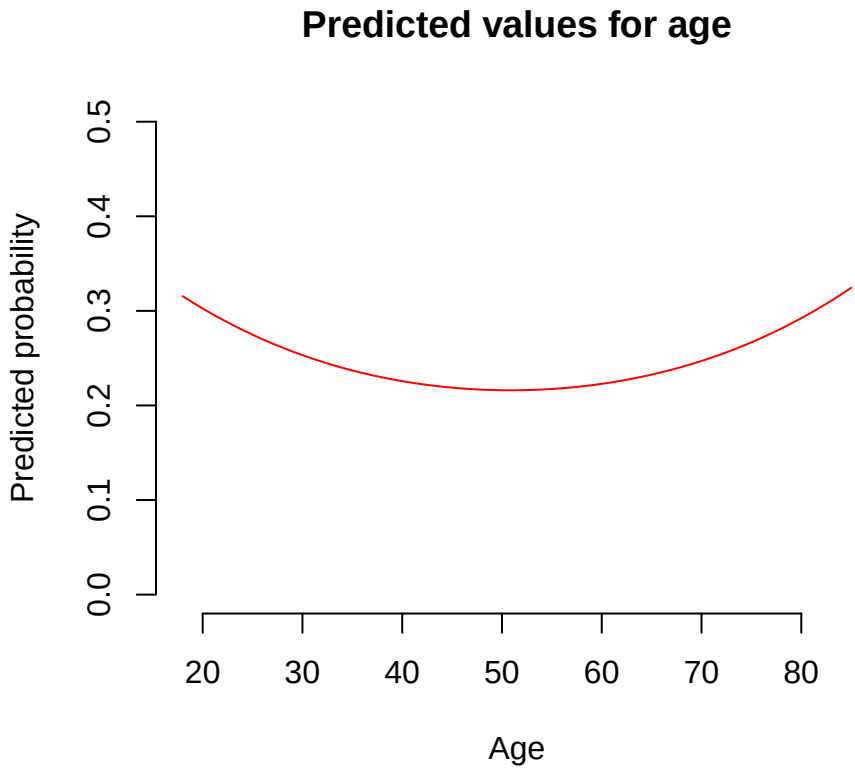
```
pmodel <- glm(R ~ age + I(age^2) + factor(gender) + factor(habitat),
              data=ceo,
              family=binomial)
```

The result of the estimation is shown below:

	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	0.2561	0.5195	0.49	0.6221
age	-0.0485	0.0185	-2.62	0.0088
I(age^2)	0.0005	0.0002	2.67	0.0076
factor(gender)2	0.1760	0.1180	1.49	0.1358
factor(habitat)2	-0.3724	0.2984	-1.25	0.2121
factor(habitat)3	-0.6374	0.2830	-2.25	0.0243
factor(habitat)4	-0.3129	0.2853	-1.10	0.2727
factor(habitat)5	0.1073	0.2976	0.36	0.7184
factor(habitat)6	0.0171	0.2797	0.06	0.9512

In the model above, we see that age, using a quadratic transformation, is statistically significative as well as a couple of categories of the habitat variable. However, it sometimes is hard to see the effect of a variable by looking only at the coefficients. In consequence, we can try to see how the predicted values look like for individuals between 18 and 85 years old.

We can now show a plot to have a clearer perspective. In this plot, we show the predicted variables in the interval $[0, 0.5]$ to put some perspective on the curvature of the function. As it can be seen the function is pretty flat and the difference across age groups that is suggested by the fit is small.



3.2 A more flexible model

The model above makes very strong assumptions about the functional form of the age variable. It seems a good idea to be slightly more flexible and use instead a semi-parametric approach that adjusts a spline to age and keeps the rest of the model *as is*.

```
npmodel <- gam(R ~ s(age) + factor(gender) + factor(habitat),
               data=ceo,
```

```
                family=binomial,
                select=TRUE)
```

The interesting part of the model corresponds to the ANOVA of the nonparametric terms, which we show below in a nicely formatted table.

	Npar Df	Npar Chisq	P(Chi)
(Intercept)			
s(age)	3	22.25	0.00
factor(gender)			
factor(habitat)			

Another approach would be to use a flexible model like a classification tree, which has several advantages for us:

1. It performs variable selection in a very natural way by simply not picking a variable.
2. It is very easy to interpret as a series of decision rules.
3. It approximates transformations of the input variables as well as interactions.

There are many different types of trees, but in this case, we have chosen the one implemented in `partykit`[1] which corresponds to the Conditional Inference tree, a variety that performs hypothesis testing in each of the splits and gets around some of the well known issues with older algorithms like CART. We can run it using the same interface as the `glm` above by simply using the default control parameters.

```
tmodel <- ctree(R ~ age + factor(gender) + factor(habitat),
                data=ceo)
print(tmodel)
```

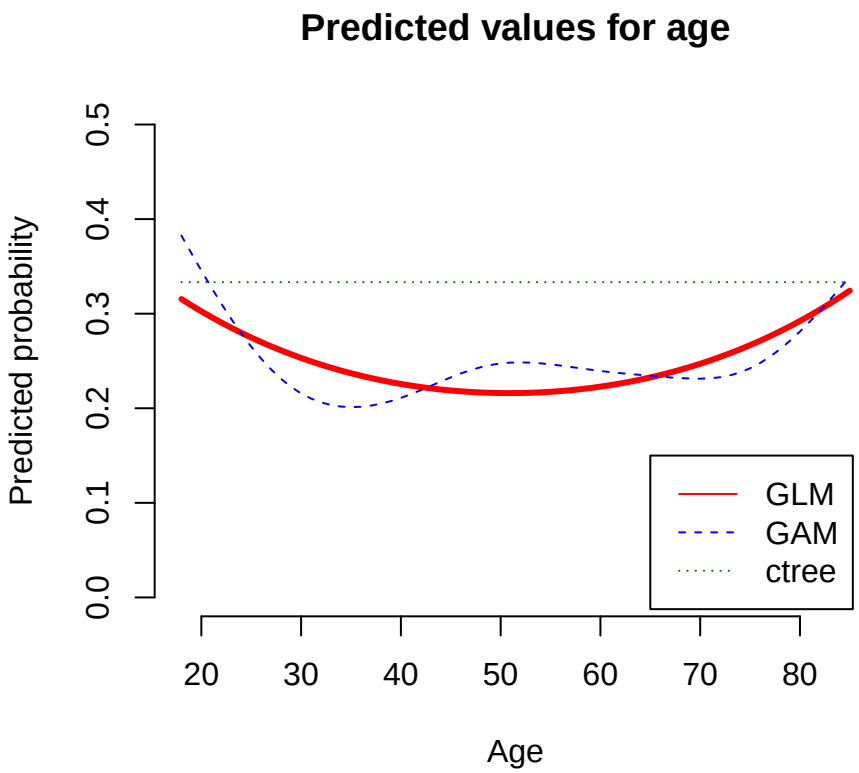
```
##
## Model formula:
## R ~ age + factor(gender) + factor(habitat)
##
## Fitted party:
## [1] root
## |   [2] factor(habitat) in 1, 5, 6: FALSE (n = 582, err = 33%)
## |   [3] factor(habitat) in 2, 3, 4: FALSE (n = 918, err = 23%)
##
## Number of inner nodes:    1
## Number of terminal nodes: 2
```

The function `print` allows us to see the tree in a plain text format, which may not be the best one, but conveys enough information about the decision rules.

3.3 Comparing the three models

It is probably more interesting to see how the three models compare together by plotting their predictions in the same figure:

[1] More information about `partykit` can be found here.



The main interpretation here is that the differences between the three models are relatively small and it seems that the tree model *more aggresively* discounts the potential effect of age.

3.4 Adding external information

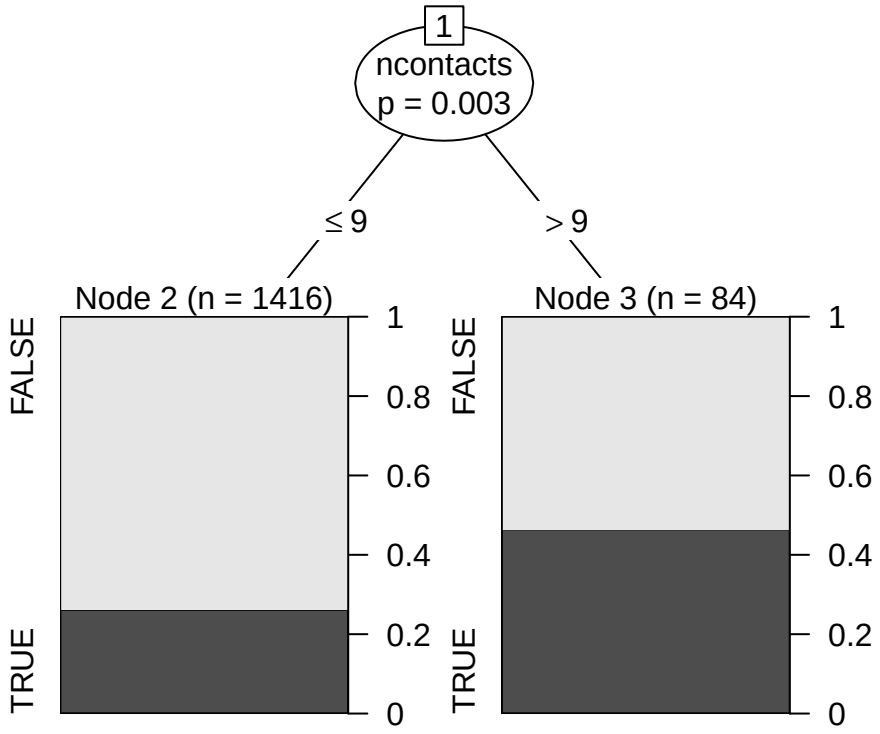
Say now that we have access to a paradata database that contains records of the number of contact attempts made for each of the ID. One could potentially argue that the degreee to which the respondent was cooperative should predict how likely the respondent is to provide information about the income question. In order to test this theory, we first need to pull the information from the SQL database and then merge the resulting information with our data using the individual ID as key.

The query that will pull the information is interesting in this case and so we included it in this report:

```
select * from paradata;
```

If we merge the survey dataset and the paradata information we could then rerun the same models as above to see whether the number of contacts has an effect on the probability of responding. To simplify matters, we will keep here only the tree model.

The resulting tree captures the idea that the response to the question is mostly driven by the number of contact attempts and as a matter of fact none of the other variables get selected.



We could have explored other models using `stan` or `python` but the model is good enough for our purposes. We could even have included `C++` code by leveraging the `Rcpp` library

4 Conclusions

We have seen here how age is not a strong predictor of the likelihood of responding to the income question in the ABC survey. Although a GLM and a semiparametric model show a nonlinear effect of age, our tree model does not even pick the age variable. It seems that people report their income based exclusively on the number of contact attempts.

Bibliography

Tourangeau, Roger, and Ting Yan. 2007. "Sensitive Questions in Surveys." *Psychological Bulletin* 133 (5). American Psychological Association: 859.

Yan, Ting, Richard Curtin, and Matthew Jans. 2010. "Trends in Income Nonresponse over Two Decades." *Journal of Official Statistics* 26 (1): 145.