FAQ for the NlsyLinks Package

William Howard Beasley (Howard Live Oak LLC, Norman)
Joseph Lee Rodgers (Vanderbilt University, Nashville)
David Bard (University of Oklahoma Health Sciences Center, OKC)
Kelly Meredith (Oklahoma City University, OKC)
Michael D. Hunter (University of Oklahoma, Norman)

May 28, 2017

Contents

1 Getting Started 1

2 Gen1 and Gen2 1

3 Ambiguous twins 2

4 Ambiguous siblings 2

5 Retaining vs. dropping the ambiguous twins and siblings 3

6 Race and Gender Variables 4

7 Grant support 5

1 Getting Started

Where is a good place to start?

Welcome to the `NlsyLinks` package, which facilitates research with the NLSY. The initial focus of the package was to assist behavior genetics, but it has been expanded to help with NLSY research in general.

fThe current FAQ and the NLSY ACE vignette provide overview, and the reference manual describe the package's syntax and functions; the documents are available on CRAN.

Literature targeting general behavior genetics is listed in the appendix of the NLSY ACE vignette. Furthermore, the articles and books involving the NLSY kinship links provide more specialized information.

2 Gen1 and Gen2

What does "Gen1" and "Gen2" mean?

This package considers both generations of the NLSY79. The first generation (*ie*, 'Gen1') refers to subjects in the original NLSY79 sample (`http://www.bls.gov/nls/nlsy79.htm`). The second generation (*ie*, 'Gen2') of subjects are the biological offspring of the original females -*i.e.*, those in the NLSY79 Children and Young Adults sample (http://www.bls.gov/nls/nlsy79ch.htm). The NLSY97 is a third dataset that can be used for behavior genetic research (http://www.bls.gov/nls/nlsy97.htm), although this vignette focuses on the two generations in the NLSY79.

Standard terminology is to refer second generation subjects as 'children' when they are younger than age 15 (NSLYC), and as 'young adults' when they are 15 and older (NLSY79-YA); though they are the same respondents, different funding mechanisms and different survey items necessitate the distinction. This cohort is sometimes abbreviated as 'NLSY79-C', 'NLSY79C', 'NLSY-C' or 'NLSYC'. This packages uses 'Gen2' to refer to subjects of this generation, regardless of their age at the time of the survey.

3 Ambiguous twins

What are "ambiguous twins"?

MZ twins share all of the genetic information (*i.e.*, $R = 1$), while DZ twins on average share half (*i.e.*, $R = 0.5$). Sometimes a sibling pair doesn't have enough information for us to classify comfortably as either MZ or DZ. We assign these "ambiguous twins" $R = .75$. Currently there are 12 ambiguous twins in the NLSY79C sample.

Of these 13 pairs, all had close birthdays and were the same gender. 12 pairs are ambiguous because the mother didn't complete an NLSY survey since 1993; the first twin items were presented in 1994 (*e.g.*, `R48257.00`, and `R48260.00`). The mother of the 13th pair (*i.e.*, subjects 864902 and 864903) simply avoided responding to the twin survey items.

Occasionally they mother of twins provided conflicting evidence. Fortunately, these mother were consistent among their most recent responses. For instance, Subjects 392401 and 392402 were indicated DZ in 1998, but MZ in 2000, 2002, and 2004. This pair was assigned $R = 1$.

Gen2 ambiguous twins can be viewed with:

```r
subset(Links79Pair, RelationshipPath=='Gen2Siblings' & R==.75)
```

```
##       ExtendedID SubjectTag_S1 SubjectTag_S2    R RelationshipPath
## 5200        1460        146001        146002 0.75     Gen2Siblings
## 20230       5658        565901        565902 0.75     Gen2Siblings
## 24698       6639        663901        663902 0.75     Gen2Siblings
## 26363       7111        711101        711102 0.75     Gen2Siblings
## 29752       7913        791406        791407 0.75     Gen2Siblings
## 36439       9596        959601        959602 0.75     Gen2Siblings
## 37729      10012       1001201       1001202 0.75     Gen2Siblings
## 39899      11191       1119103       1119104 0.75     Gen2Siblings
## 40067      11486       1148601       1148602 0.75     Gen2Siblings
## 40154      11733       1173301       1173302 0.75     Gen2Siblings
## 40158      11739       1173901       1173902 0.75     Gen2Siblings
## 42728      12574       1257402       1257403 0.75     Gen2Siblings
```

4 Ambiguous siblings

What are "ambiguous siblings"?

Similar to ambiguous twins, ambiguous siblings are sibling pairs that we cannot comfortably classify as either full-siblings ($R = .5$) or half-siblings ($R = .25$). All siblings in the NLSY79-C/YA dataset share the same biological mother, so for these pairs, the problem is reduced to determining if they share the same biological father. There are two typical reasons for classifying siblings as ambiguous: (a) the relevant items are missing responses, or (b) the existing responses conflict with each other.

For instance, there are at least 194 Gen2 pairs where one sibling explicitly reported they shared a biological father, while the other sibling explicitly reported they did not. These subjects can be viewed with:

```
dsLinks <- Links79PairExpanded
isGen2Sib <- dsLinks$RelationshipPath=='Gen2Siblings'

olderFullYoungerHalf <- (dsLinks$RExplicitOlderSibVersion==.5 &
                           dsLinks$RExplicitYoungerSibVersion==.25)

olderHalfYoungerFull <- (dsLinks$RExplicitOlderSibVersion==.25 &
                           dsLinks$RExplicitYoungerSibVersion==.5)


dsLinks[isGen2Sib & (olderFullYoungerHalf | olderHalfYoungerFull), ]
```

Another example occurs when a subject reports they are unsure or if their own responses are inconsistent over the years. These 80 Gen2 pairs can be viewed with:

```
dsLinks[ isGen2Sib & (dsLinks$RExplicitOlderSibVersion==.375 |
     dsLinks$RExplicitYoungerSibVersion==.375), ]
```

When the one perspective provided inconclusive evidence of R, we looked at other perspectives to resolve their relationship.

5 Retaining vs. dropping the ambiguous twins and siblings

I am running ACE models with sibling pairs. Do you recommend including the pairs who are classified as $R = .375$ or $R = .75$? Or should I exclude them from the analyses?

This important issue touches Behavior Genetic concepts and modeling pragmatics. However, this issue typically has an easier resolution than it used to. In the links we released 10 years ago, there were 3,053 Gen2 pairs classified as ambiguous;in our current version, this has been reduced to 610. From one perspective, we are more likely to recommend dropping the ambiguous siblings because there are fewer of them (and therefore less potential gain by including them).

Here's another perspective. Usually if they're missing the data necessary to determine the R value, they're also missing the phenotype, so they'd contribute very little to the analysis anyway. If there's only a few in an R group, it may not be worth including them. Virtually none of the ambiguous twins have phenotype values for both Gen2 siblings.

Our advice to include/exclude an R group also depends on the kind of analysis. Some analyses break up the R values into separate categories (like multiple group SEMs). While some analyses treat R like a continuous variable (like DF analysis, or SEMs with constraint/definition variables). If you're running the former, we're more likely to recommend dropping small R groups, because they're more likely to be estimated poorly (eg, the covariance matrix is more likely to misbehave). If you're running the latter, the estimation is more robust. (Though the estimation's robustness is a different issue that if that R group is a good representation).

We don't recommend blindly dropping the ambiguous twins and siblings in every analysis. For each scenario, the group sizes and phenotypic measurement issues should be considered.

We do recommend running a casual sensitivity test, at the very least. Run different models that include and exclude the small R groups. Hopefully the estimates change in expected ways (*e.g.*,, including ambiguous siblings makes only a small difference) and you don't have to dig deeper. For all analyses, inspect each *R* group's covariance matrix, especially with for the MZs, which typically is the smallest group.

6 Race and Gender Variables

Where are the race and gender variables?

Any NLSY dataset extracted using the NLS Investigator will include some mandatory fields, including race and gender. For Gen1, these are the `R02147.00` and `R02148.00` variables; for Gen2, they are the `C00053.00` and `C00054.00` variables.

The NlsyAce vignette describes how to incorporate extracts into R for manipulation and analysis (*e.g.*, "DF analysis with a univariate outcome from a Gen2 Extract"). Those vignette example focused on incorporating outcomes. The FAQ entry focuses on race and gender.

For Gen1, create a string variable that points to your extract (*e.g.*, `filePathOutcomes <- "C:/BGResearch/NlsExtracts/gen2-birth.csv"`). For the code to work on your computer, this example will reference a file it knows exists; but make sure you replace this with your unique path.

```
library(NlsyLinks)
filePathOutcomes <- file.path(path.package("NlsyLinks"), "extdata", "gen1-life-course.csv")
```

Next, pass that path to the `ReadCsvNlsy79Gen1()` function, which converts that CSV into an R `data.frame` and adds a few extra columns that will make NLSY research easier.

```
dsDemographics <- ReadCsvNlsy79Gen1(filePathOutcomes)
summary(dsDemographics)
```

```
##    SubjectTag        SubjectID       ExtendedID      Generation    R0214700
##  Min.   :    100   Min.   :    1   Min.   :    1   Min.   :1    Min.   :1.000
##  1st Qu.: 317225   1st Qu.: 3172   1st Qu.: 3171   1st Qu.:1    1st Qu.:2.000
##  Median : 634350   Median : 6344   Median : 6338   Median :1    Median :3.000
##  Mean   : 634350   Mean   : 6344   Mean   : 6337   Mean   :1    Mean   :2.434
##  3rd Qu.: 951475   3rd Qu.: 9515   3rd Qu.: 9505   3rd Qu.:1    3rd Qu.:3.000
##  Max.   :1268600   Max.   :12686   Max.   :12686   Max.   :1    Max.   :3.000
##     R0214800        R9908100         R9908200       R9908300
##  Min.   :1.000   Min.   :-998.0   Min.   :-998   Min.   :-998.0
##  1st Qu.:1.000   1st Qu.:-998.0   1st Qu.:-998   1st Qu.:-998.0
##  Median :1.000   Median :  20.0   Median :  20   Median :-996.0
##  Mean   :1.495   Mean   :-246.2   Mean   :-438   Mean   :-731.2
##  3rd Qu.:2.000   3rd Qu.:  25.0   3rd Qu.:  27   3rd Qu.:  20.0
##  Max.   :2.000   Max.   :  47.0   Max.   :  51   Max.   :  50.0
##     R9908600         R9909700         R9909800         R9910400
##  Min.   :-999.0   Min.   :-999.0   Min.   :-999.0   Min.   :-4.000
##  1st Qu.:  16.0   1st Qu.:-998.0   1st Qu.:-998.0   1st Qu.:-4.000
##  Median :  21.0   Median :   9.0   Median :   0.0   Median :-4.000
##  Mean   :-193.8   Mean   :-328.8   Mean   :-355.9   Mean   :-1.929
##  3rd Qu.:  25.0   3rd Qu.:  36.0   3rd Qu.:   1.0   3rd Qu.: 0.000
##  Max.   :  51.0   Max.   : 416.0   Max.   :   1.0   Max.   : 9.000
```

You can see which variables were added or renamed by `ReadCsvNlsy79Gen1()` (*i.e.*, the first four), and which still retain the original names from the NLS Investigator (*i.e.*, the ones that start with 'R' and are followed by numbers). The first two original variables happen to correspond to Gen1's race and gender. Rename to something more salient to you, such as

```r
dsDemographics <- RenameNlsyColumn(dsDemographics, "R0214700", "Race")
dsDemographics <- RenameNlsyColumn(dsDemographics, "R0214800", "Gender")
```

Finally, consider converting the numeric variables to factor variables so your code will be more readable.

```r
#The official documentation calls this last level "NON-BLACK, NON-HISPANIC"
dsDemographics$Race <- factor(x=dsDemographics$Race,
                              levels=1:3,
                              labels=c("Hispanic", "Black", "NBNH"))
dsDemographics$Gender <- factor(x=dsDemographics$Gender,
                              levels=1:2,
                              labels=c("Male", "Female"))
```

For Gen2, race and gender can be incorporated with a similar approach. Remember to change `filePathOutcomes` to your desired location.

```r
library(NlsyLinks)
filePathOutcomes <- file.path(path.package("NlsyLinks"), "extdata", "gen2-birth.csv")
dsDemographics <- ReadCsvNlsy79Gen2(filePathOutcomes)  #Notice this function is for Gen2.
# summary(dsDemographics) #Uncomment to see the summary

dsDemographics <- RenameNlsyColumn(dsDemographics, "C0005300", "Race")
dsDemographics <- RenameNlsyColumn(dsDemographics, "C0005400", "Gender")

dsDemographics$Race <- factor(x=dsDemographics$Race,
                              levels=1:3,
                              labels=c("Hispanic", "Black", "NBNH"))
dsDemographics$Gender <- factor(x=dsDemographics$Gender,
                              levels=1:2,
                              labels=c("Male", "Female"))
```

If you have another `data.frame` that needs to be merged with the demographic dataset, use `SubjectTag` as the key, because this variable is guaranteed to be unique when the subjects in both generations are in the same dataset. Supposed your other dataset is called `dsOutcomes`, the merging code would be

```r
dsCombined <- merge(x=dsDemographics, y=dsOutcomes, by="SubjectTag", all=TRUE)
```

Make sure that `dsOutcomes` also has the `SubjectTag` variable; this will happen automatically if it was read into R using the `ReadCsvNlsy79Gen1()` or `ReadCsvNlsy79Gen2()` functions. For more information about `SubjectTag`, please see the `Links79Pair` entry in the reference manual.

7 Grant support

This package's development was largely supported by the NIH Grant 1R01HD65865, "NLSY Kinship Links: Reliable and Valid Sibling Identification" (PI: Joe Rodgers; Vignette Construction by Will Beasley)