Using R for Analyzing Indifference Point Data

The public domain software *R* can be downloaded for the Windows, Mac, or Linux platforms at https://cran.r-project.org. This download contains the core functionality of *R* and a wide range of "libraries" that can be loaded as needed. Furthermore, various *R* packages can be downloaded and installed that contain additional libraries with specific functionality. Both the core program and the various libraries are frequently updated, so issues can arise when a user downloads a new library that is incompatible with their currently installed version of *R*. The solution is to download a new version of *R* and to reinstall the packages that contain the libraries of interest. Although *R* contains help files for every command, these files are often rather cryptic for the average user. A more accessible general reference to learn how to perform many of the basic tasks in *R* is the *Quick-R* website, http://www.statmethods.net, as well as numerous books.

In the examples below, I make heavy use of the "nlme" library in *R* that is part of the initial download. Another library that is of significant utility for graphing multilevel data is the "lattice" library. At the time of writing, there are over 9,500 packages available for *R*, most of which are not part of the core installation and would require separate installation. In general, I commonly use only a handful of libraries that are not automatically installed with the core download.

Some users will get an obscure error when attempting to download and install a package. This usually arises either because the person has not checked the "install dependencies" box or the computer's virus protection software is blocking part of the install. Checking "install dependencies" ensures that the download will also install any libraries necessary for the functioning of the library being downloaded. If you are still having problems, you should

temporarily disable your virus protection while you install new libraries (note, this is only necessary for the initial installation, not for subsequent calls of the library).

There are two *R* packages that are important when making condition comparisons and simple effects tests, multcomp and lsmeans. I prefer multcomp despite its need for greater understanding of variable coding because of its flexibility, but lsmeans is more user-friendly when performing basic comparisons. The multcomp basics are covered in https://cran.r-project.org/web/packages/multcomp/vignettes/multcomp-examples.pdf and the lsmeans package is covered in https://cran.r-project.org/web/packages/lsmeans/vignettes/using-lsmeans.pdf. Note that these files demonstrate the range of functionality in these packages, but the typical psychologist is likely to only need a couple of the commands in these libraries (see file "Example with interactions" for an example of the use of the multcomp library).

The Logic of Nonlinear Multilevel Modeling

Here, I will outline the basic logic and syntax involved in using *R* to fit a hyperbolic function to discounting data. The basic syntax for fitting discounting curves in *R* is illustrated in the following two commands:

```
myrList<-nlsList(Value~1000/(1+exp(logk)*Delay)|Subject,
data=myd, start=c(logk=-2))

myr<-nlme(Value~1000/(1+exp(logk)*Delay), fixed=logk~1,
random=logk~1|Subject, data=myd, start=c(logk=-2))
```

The first command fits the hyperbolic to each subject individually whereas the second command uses multilevel modeling to fit the subjects as a collective. The e^{logk} hyperbolic variation (Equation 2) lies at the heart of both function calls. The value of the numerator of the hyperbolic equation (here, 1000) would vary depending on the amount immediately available. In the *nlsList* syntax, the "| Subject" simply indicates that the fit is to be done separately for each subject, and the "data=myd" phrase designates the name of the dataframe being analyzed (in this case "myd",

see the accompanying *R* command files for a full example). When doing nonlinear curve fitting, it is necessary to provide an initial estimate of the parameters in the model. Intelligent choice of these starting values is critical for the model to converge on a solution, and it is worth the time to try different values to ensure that you have the model with the best fit. While I tend to use AIC as a model selection criterion, there are published treatments of alternative model metrics that can help inform the researcher's choice (Burnham & Anderson, 2004; Pitt & Myung, 2002).

For the multilevel modeling function *nlme*, the additional arguments *fixed* and *random* are specified. The "fixed=logk~1" component indicates that there is one value of *logk* to be estimated for the entire set of data. The "random=logk~1 | Subject" component indicates that each subject is assumed to have a *logk* that varies around the fixed (or group) estimate of *logk*. These so called "random effects" do not generate statistical tests of their values; they are included to model the covariance between data values produced by the same subject. A failure to model these dependencies can create egregious errors in statistical analysis; these major problems with ignoring data dependencies prompted the development of repeated measures ANOVA. There are a range of additional options that can be passed to these functions to improve model fit and convergence, some of which are included in the Supplemental Materials; for a full treatment, see Pinheiro and Bates (2004).

The nlsList command will produce individual estimates of the *logk* values along with their standard errors for each subject for which the algorithm converged (the "summary(myrList)" command provides this information). Thus, there is no overall assessment of the model in terms of R^2, AIC, BIC, RMSE, or other fit metrics. Furthermore, the consideration of additional within- or between-subject variables must be included in a second

analysis stage that is, of course, ignorant of the varying precisions of the *logk* values revealed by the nlsList analysis.

In contrast, the nlme command will produce an estimate of a single overall *logk* for the subjects along with its standard error (and associated statistical details) as well as an overall metric of its fit to all of the subjects' data (the AIC and BIC are automatically provided and recommended for nonlinear fits). The individual *logk* estimates can be obtained from the "coef" function. Furthermore, nlme can include between- and within-subject variables as modifiers of the estimated discount rate. Estimating the effects of these variables occurs concurrently with the estimation of the discount rate.

To illustrate the inclusion of a categorical predictor of discounting rate, the following examples are based on the $500 and $10,000 conditions of Kaplan and Reed (2013). Although the design was within-subject, for the between-subject analysis shown below the data from only one condition from each subject was used. For the within-subject analysis (i.e., consistent with the actual design), the full data set was used. Before analyzing the data, each indifference point was standardized by dividing it by its condition's undiscounted value of $500 or $10,000 thus necessitating the use of 1.0 in the numerator of the hyperbolic function.

The following command illustrates the inclusion of a single two-level, between-subject categorical predictor into the nlme call:

```
myr<-nlme(Value~1/(1+exp(logk)*Delay),
fixed=logk~Condition, random=logk~1|Subject, data=myd,
start=c(logk=-2, 0))
```

In the following treatment, I will assume the default 0/1 dummy coding of condition. In this command, *logk* is allowed to have two estimates, one for the condition value that was dummy-coded as 0 (*logk.(Intercept)* in the output) and one for the difference between this value and that of the condition value that was dummy-coded as 1 (*logk.Condition* in the output). Because there

are now two parameters that must be estimated, there are two start values; the first is for the intercept (i.e., the *logk* for the first condition) and the second is for the difference in *logk* between the two conditions (by choosing the value to be 0 here, I assume no difference as a starting point). The critical portion of the output generated from a summary of the model along with the first five subjects' model coefficients is shown below.

```
> summary(myr)
Nonlinear mixed-effects model fit by maximum likelihood

      AIC         BIC       logLik
-369.1762   -349.8098     188.5881

Fixed effects: logk ~ Condition
                         Value Std.Error  DF    t-value p-value
logk.(Intercept)    -2.9808499 0.1917112 779 -15.548645  0.0000
logk.Condition10000 -0.4340097 0.2713859 779  -1.599234  0.1102

> coef(myr)
    logk.(Intercept) logk.Condition10000
1        -1.39074185          -0.4340097
2        -4.34986279          -0.4340097
3        -1.33801175          -0.4340097
4        -1.50858139          -0.4340097
5        -3.48339436          -0.4340097
```

Thus, the estimated *logk* for the $500 magnitude condition was -2.98 (and significantly different from zero) and the estimated *logk* for the $10,000 condition was -2.98 – .43 = -3.41 (the difference was not significantly different from zero, $p = .11$). The *coef* command reveals the variation in *logk* estimates across subjects. Note that the effect of the condition difference (*logk.Condition10000*) is constant across subjects because this variable was between-subject; although present for every row in the coefficients table, this adjustment will only be applied for subjects in the $10,000 group.

The situation for a within-subject variable is a bit more complicated. The following command assumes the condition variable varies within-subject:

```
myr<-nlme(Value~1/(1+exp(logk)*Delay),
fixed=logk~Condition, random= logk~Condition|Subject,
data=myd, start=c(logk=-2, 0))
```

The key difference here is the inclusion of "Condition" in the random effects specification. This indicates that the condition difference in *logk* values at the individual subject level is allowed to vary across subjects. The fixed effect of condition is the result of empirical interest to most researchers. The random effect involving condition must be included to model the within-subject dependence between conditions involving the same subject as well as to allow the effect of this within-subject variable to vary across subjects (this is called a "slope effect" in multilevel parlance). Some subjects might have produced similar discounting rates across the conditions whereas others might have produced very different rates. It is common to evaluate whether it was necessary to allow this variation by omitting condition in the random effect specification and then comparing the model that included it with the model that excluded it. Model comparison can be done by comparing AIC/BIC values or with a direct likelihood ratio test using "anova(modelname1, modelname2)". In this case, the model including condition as a random effect was much more likely to have produced the data therefore providing strong evidence that the subjects were differentially affected by the $500 versus $10,000 manipulation. It is critical to use the model including condition as a random effect if it generates a significantly better fit because such a result suggests that it is necessary to model the dependency between observations collected from the same condition (Gelman & Hill, 2006; Pinheiro & Bates, 2004).

A portion of the output from this model is shown below, first for the model summary and second for the first five subjects' model coefficients that include the random effect variation in intercept (here, estimating the *k* for the $500 condition) and slope (here, estimating the difference between the $500 and $10,000 conditions) across subjects:

```
> summary(myr)

Nonlinear mixed-effects model fit by maximum likelihood

        AIC       BIC   logLik
  -880.1351 -846.9265 446.0676

Fixed effects: logk ~ Condition
                          Value  Std.Error   DF    t-value p-value
logk.(Intercept)    -2.9468562 0.12696647 1715 -23.209719       0
logk.Condition10000 -0.4712007 0.09726704 1715  -4.844403       0

> coef(myr)
    logk.(Intercept) logk.Condition10000
1        -1.27146190        0.4206014520
2        -4.25984389        0.4988673153
3        -1.59223555       -1.5678549232
4        -1.61661335       -1.3761303983
5        -3.51849744       -0.8712103308
```

Each subject in this analysis had a different estimated adjustment for the condition difference because the size of this effect could be different for each subject and was allowed to vary in the random effect designation. For example, for the first subject the estimated *logk* for the $500 condition was -1.27 and for the $10,000 condition was -1.27 + .42 = -.85.

Model Assumptions

In the examples provided here and in the accompanying *R* command files, the residuals are assumed to be normally distributed and to satisfy the homogeneity of variance assumption common to linear regression techniques. These assumptions were satisfied for the data presented here. It is possible that an experiment including a lot of steep discounters or non-discounters might run into more ceiling (indifference points near the max) or floor (indifference points near zero) effects that can truncate one or the other tail of the distribution of the residuals. When encountered, it is possible to use an extra parameter for the nlme fit in which the residuals are differentially weighted as a function of the delay: “weights=varPower(form=~Delay).” To determine its necessity, versions of the model with and without the weighting can be run and

then compared using the AIC or running a likelihood ratio test (e.g., "anova(model1, model2)" where model1 and model2 are the variables containing the two models).

If the residuals are not normally distributed, a generalized nonlinear multilevel model could be explored. Unfortunately, the existing tools are not well tested and I cannot yet recommend their use. However, for the datasets I have analyzed, the nlme command has been sufficient.

References

Burnham, K. P., & Anderson, D. R. (2004). Multimodal inference: Understanding AIC and BIC in model selection. *Sociological Methods and Research, 33*, 261-304.

Gelman, A., & Hill, J. (2006). *Data analysis using regression and multilevel/hierarchical models*. New York: Cambridge University Press.

Kaplan, B. A., & Reed, D. D. (2013). Decision processes in choice overload: A product of delay and probability discounting? *Behavioural Processes, 97*, 21-24. doi:10.1016/j.beproc.2013.04.001

Pinheiro, J. C., & Bates, D. M. (2004). *Mixed-effects models in S and S-PLUS*. New York: Springer.

Pitt, M. A., & Myung, I. J. (2002). When a good fit can be bad. *Trends in Cognitive Sciences, 6*, 421-425.