Mediation and Moderation Analyses with R

Stephen D. Short

Saturday, February 28, 2015

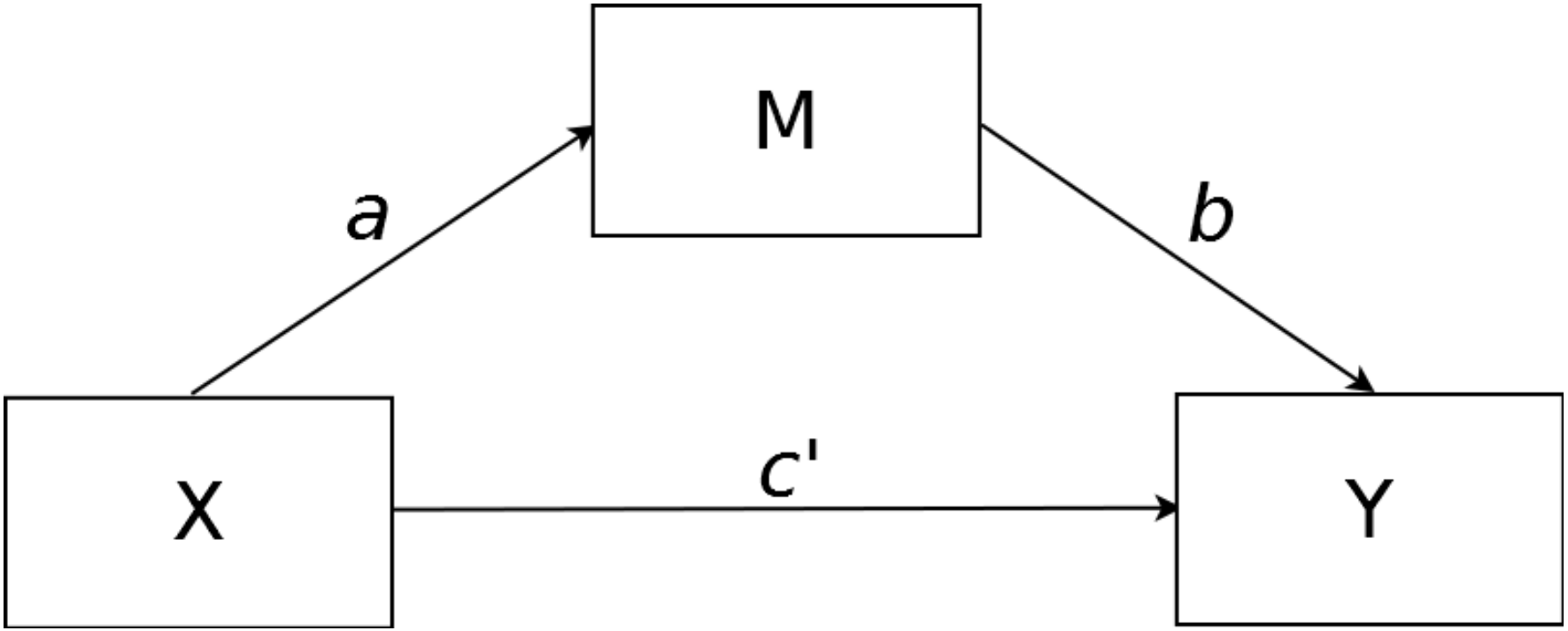
Overview

- Mediation analysis in R
  - Simple mediation model example
  - Multiple mediator model example
- Moderation analysis in R
  - Continuous moderator model example
  - Simple slope figures
- Tips

For slides and code please visit `http://stephendshort.wix.com/psyc`

Mediation

- Occurs when the effect of one variable (X) on another variable (Y) "passes through" a third variable (M)



$$M = a_0 + aX + e_M$$

$$Y = b_0 + bM + c'X + e_y$$

- The indirect effect is quantified as ab

Notable Mediation Packages Available in R

- R packages for mediation analyses
  - *BayesMed* (Nuijten, Wetzels, Matzke, Dolan, & Wagenmakers, 20015)
  - *bmem* (Zhang & Wang, 2011)
  - *mediation* (Tingley, Yamamoto, Hirose, Keele, & Imai, 2014)
  - *powerMediation* (Qui, 2015)
  - *RMediation* (Tofighi & MacKinnon, 2010)
- Functions within other packages
  - mediate () in *psych* package (Revelle, 2012)
  - mediation () in *MBESS* package (Kelley & Lai, 2012)

Note. This is not a complete list, but merely suggestions for social science researchers

Example 1: Data

- From Pollack, VanEpps, & Hayes (2012)
  - Also example data in Hayes (2013) mediation text
- Does economic stress (X) lead to a desire to withdraw from small business (Y), as a result of negative affect (M)?
- $N = 262$ small business owners
  - X = estress (1-7 Likert scale)
  - M = affect (1-5 Likert scale)
  - Y = withdraw (1-7 Likert scale)
- Example data available from www.afhayes.com

Example 1: Simple Mediation with *MBESS*

- First, install *MBESS*

```
# Note. # sign used to comment code
install.packages("MBESS")
# You only need to install the package once on your computer
```

- Load *MBESS*. You'll need to do this each new R session

```
library(MBESS)
```

- Import your data

```
estressData <- read.table("estress.csv",sep=",",header=T)
```

- *MBESS* can also analyze summary statistics (e.g., covariance matrix)

Example 1: Simple Mediation with *MBESS*

- Some of the mediation() arguments

```
mediation(x, mediator, dv, conf.level = 0.95,
          bootstrap = FALSE, B = 1000,
          which.boot= "both")
```

- Tip: First run with a small amount of replications to check your code.

```
results1 <- mediation(estressData$estress,
                      estressData$affect,
                      estressData$withdraw,
                     bootstrap = TRUE, B = 10000,
                     which.boot = "BCa")
```

```
[1] "Bootstrap resampling has begun. This process may take a
```

Example 1: *MBESS* Bootstrapped Indirect Effect

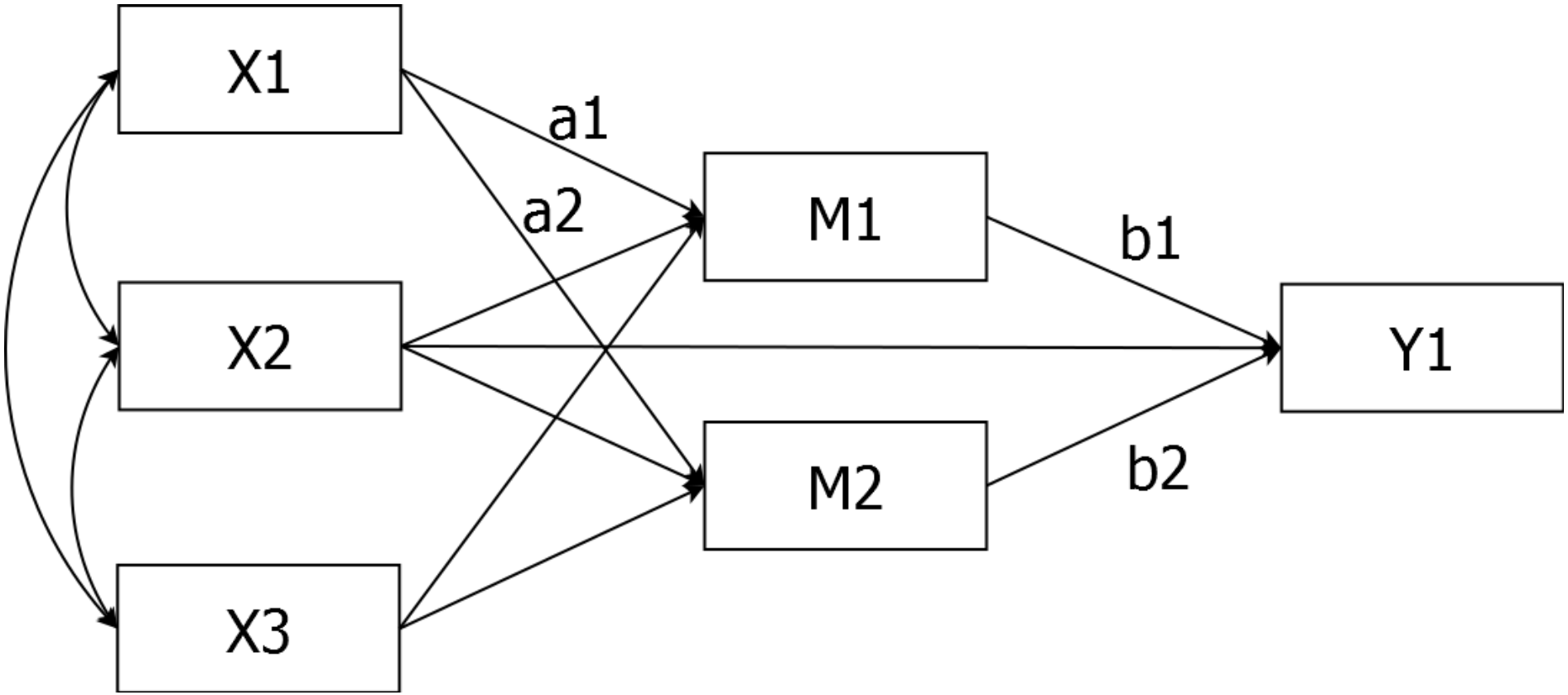
- The object 'results1' contains a lot of information
  - M on X regression table and model fit
  - Y on X and M regression table and model fit
  - Bootstrap results
  - Measures of indirect effect
  - Residual based indices
- Example 95% confidence interval for indirect effect

```
#Ask for first row of the Bootstrap.Results
# which contains estimate of indirect effect
results1$Bootstrap.Results[1,]
```

```
  Estimate CI.Lower_BCa CI.Upper_BCa
0.13296411   0.07615661   0.20867695
```

Example 2: Multiple Mediators Model

- What if you are interested in more than one predictor (X), mediator(M), or outcome (Y)?



Multiple Mediator Models in R

- *MBESS* only allows simple mediation
- *mediation* can support multiple mediators, but use caution
  - mediations() was not designed to correct for issues of multiple testing
- *psych* mediate() allows for simple or parallel mediation
  - Currently, function is beta version
- Recommendation: analyze with a multivariate modeling package
  - *lavaan* (Rossel, 2012)
  - *open-mx* (Boker et al., 2011)
  - *sem* (Fox, Nie, & Byrnes, 2014)

Example 2: Multiple Mediators Model via *lavaan*

- Specify the model

```
example2 <- '  ## regressions
          m1 ~ a1*x1 + x2 + x3
          m2 ~ a2*x1 + x2 + x3
          y1 ~ b1*m1 + b2*m2 + x2
          ## define indirect effects
          ind1 := a1 * b1
          ind2 := a2 * b2
          totalind := ind1 + ind2
          ## correlated residual variances
          m1 ~~ m2 '
```

Example 2: Multiple Mediators Model via *lavaan*

- Use sem() to analyze the model

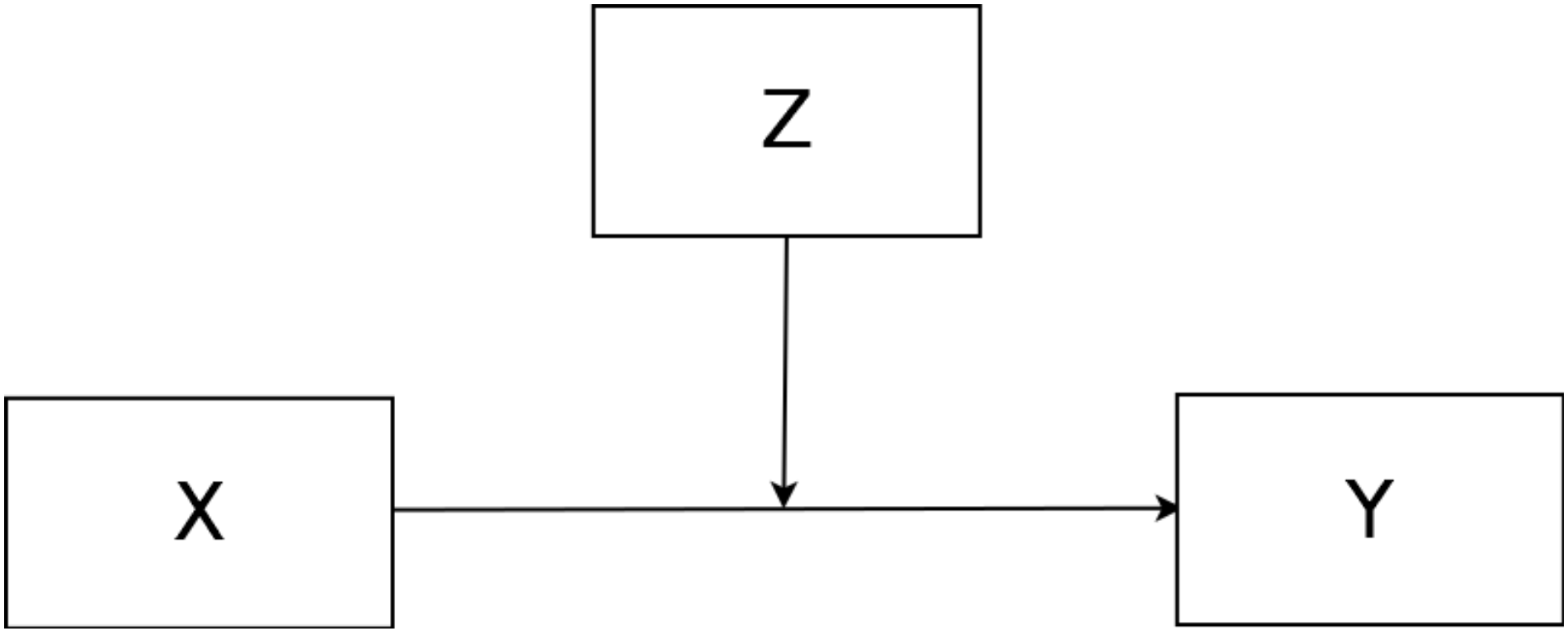
```
results2 <- sem(example2, data = ex2, meanstructure = TRUE,
                se = "boot", bootstrap = 10000)
```

- Use summary() to view results

```
summary(results2, fit.measures=TRUE, standardized=TRUE)
```

Moderation

- Moderation occurs when the effect the predictor (X) on the outcome (Y) depends on the moderator (Z)
  - Depending on Z, X to Y changes in strength



$$Y = b_0 + b_1 X + b_2 Z + b_3 XZ + e_y$$

Moderation Analysis in R

- R is preloaded with several important functions
  - lm() is used to fit linear models in R
  - aov() can also be used for ANOVA designs
- *rockchalk* (Johnson, 2015) offers a variety of helpful functions
  - Simple slopes plots
  - Test simple slopes
  - Generate regression results tables

Example 3: Continous x Continous Interaction

- The following example data (epi.bfi) comes from the *psych* package (Revelle, 2012)
- $N = 231$ undergraduate students from a Midwest school
  - bdi = Beck Depression Inventory
  - epiNeur = Neuroticism from Eysenck Personlaity Inventory
  - stateanx = state anxiety
- Can depression be predicted by one's prevalence of state-based axniety and neuroticism?
- Does relation between neuroticism (X) and depression (Y) depend on state-anxiety (Z)?

Continous x Continous Variable Interaction

- First, load the *psych* package and retrieve our example data.

```
#make sure psych package is installed, then load package
library(psych)
#the epi.bfi dataset is now present and we could check it by c
#head() displays first six lines of dataset
head(epi.bfi)
```

```
  epiE epiS epiImp epilie epiNeur bfagree bfcon bfext bfneur b
1   18   10      7      3       9     138    96   141      51
2   16    8      5      1      12     101    99   107     116
3    6    1      3      2       5     143   118    38      68
4   12    6      4      3      15     104   106    64     114
5   14    6      5      3       2     115   102   103      86
6    6    4      2      5      15     110   113    61      54
  traitanx stateanx
1       24       22
```

Continous x Continous Variable Interaction

- Estimate the model with an interaction between state anxiety and neuroticism

```
#lm() fuction format is y ~ x*z
#Using '*' between variables tells R to estimate
#main effects and interaction
example3 <- lm(bdi ~ stateanx*epiNeur, data=epi.bfi)
```

Contionous x Continous Variable Interaction

- summary(example3) displays model results
  - Subset of summary(example3)

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	0.064	2.1856	0.029	0.9768
stateanx	0.038	0.0606	0.619	0.5368
epiNeur	-0.148	0.1887	-0.782	0.4347
stateanx:epiNeur	0.015	0.0047	3.279	0.0012

- We have a significant Neuroticism X State Anxiety interaction
- We can create a simple slopes plot to examine this closer

Simple Slopes Plot with *rockchalk*

- Specify what values of the moderator state anxiety (modX) you would like to use to examine the relationship between neuroticism (X) and depression (Y)

```
library(rockchalk)
#To plot simple slopes, we can use the plotSlopes() command.
plotSlopes(example3, plotx="epiNeur",
           modx="stateanx", modxVals="std.dev.")
```

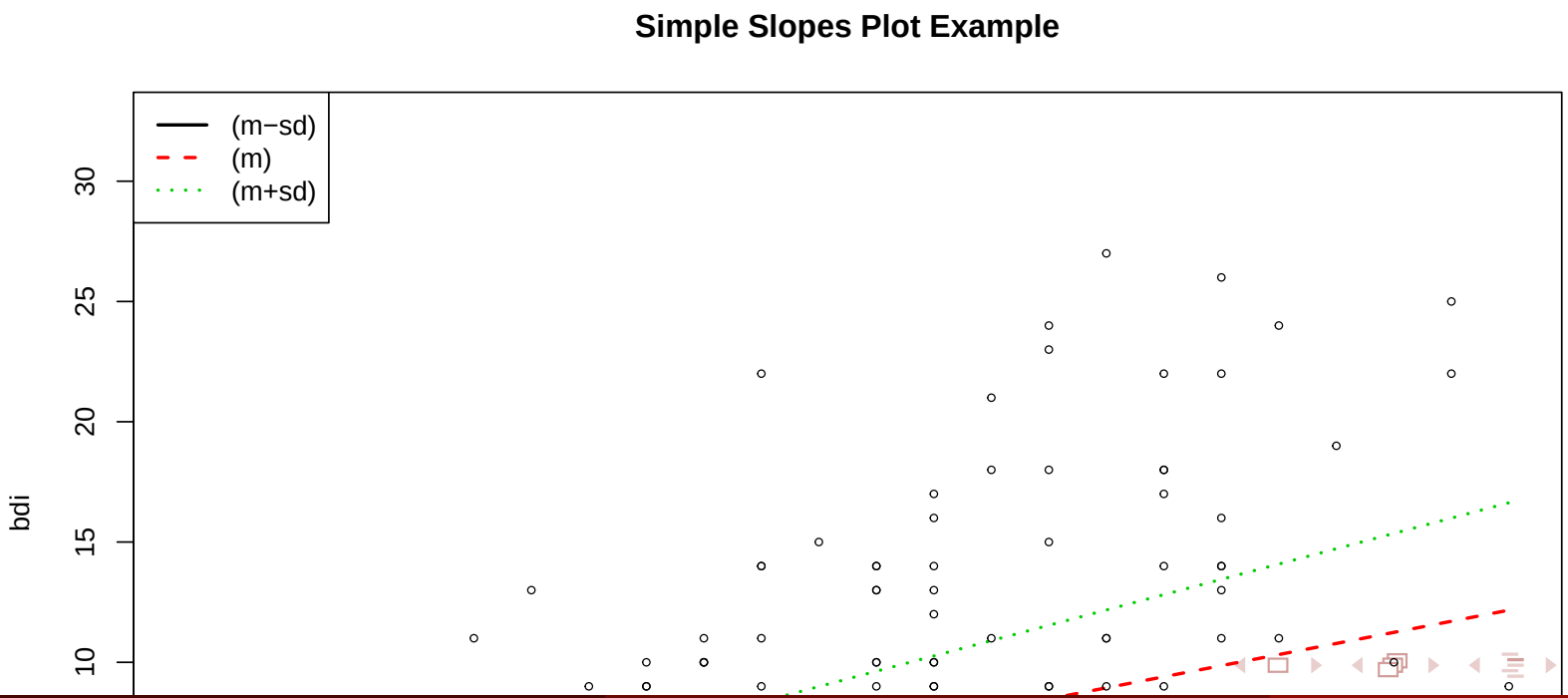
- Alternatives:
  - modxVals = "quantile"
  - modxVals = c(##, ##, ##). Replace ## with your desired values

Simple Slopes Plot with *rockchalk*

```
Attaching package: 'rockchalk'

The following object is masked from 'package:MASS':

    mvrnorm
```

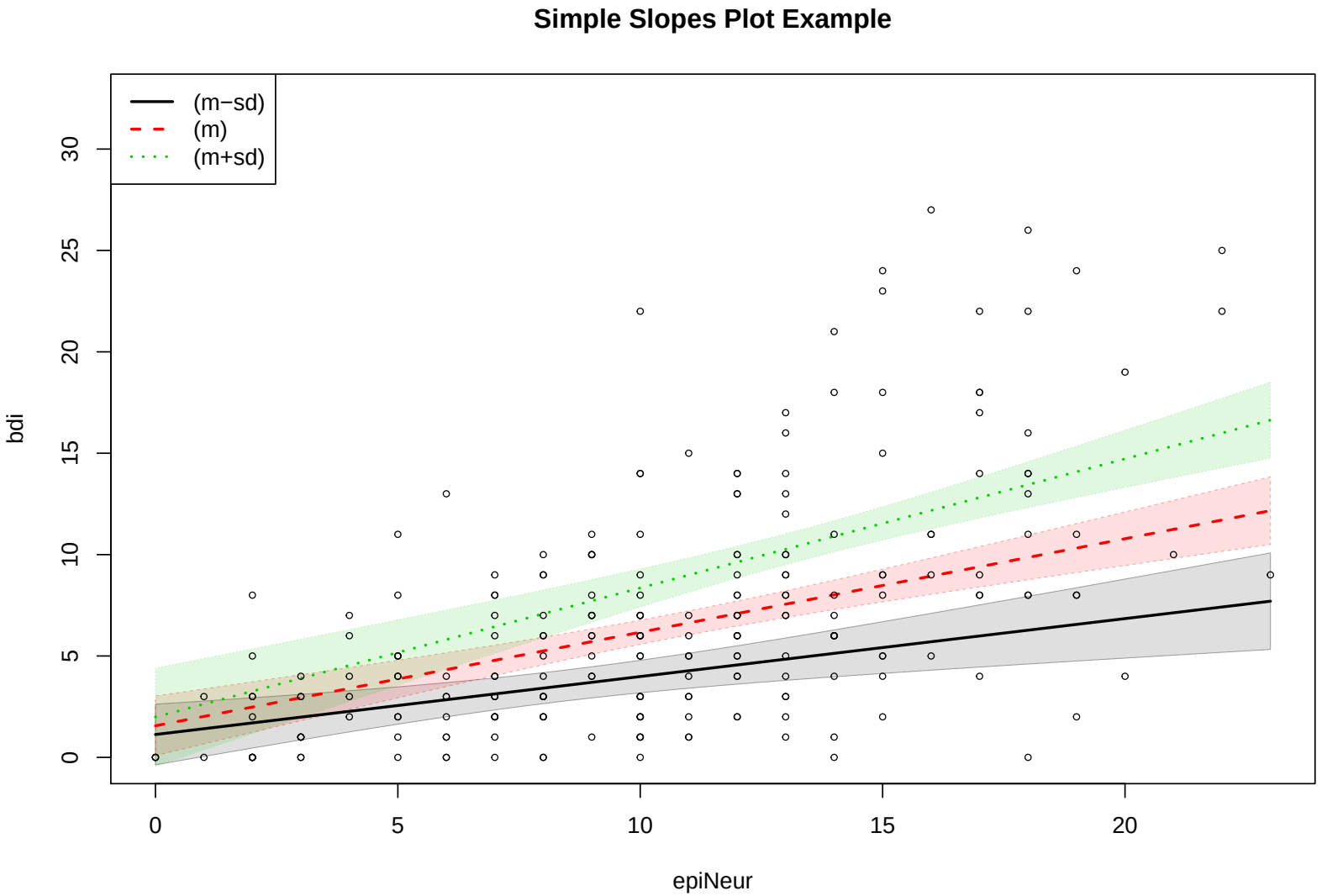


Simple Slopes Plot with *rockchalk*

- We could add a 95% confidence interval around our simple slopes!
- Use plotCurves()

```
plotCurves(example3, plotx="epiNeur",
           modx="stateanx", modxVals="std.dev.",
           interval="confidence", main = "Simple Slopes
           Plot Example")
```

Simple Slopes Plot with *rockchalk*



Final Tips

- Loading multiple packages in one R session may "mask" certain functions.
  - Ex: *psych* package loaded, then you load *mediation* package

"The following object is masked from 'package:psych' :

mediate"

- Use a "::" to use mediate() function from *psych*

```
psych::mediate(y, x, m, data)
```

- Load packages of most interest last
- Use the search() to see order R will search for objects

Final Tips

- Try out new packages on example data first
  - Replicate previous results
- Check for package updates

```
update.packages(c("MBESS","rockchalk"))
```

- New packages are always becoming available
  - or you can create your own!

Thank You!

Stephen D. Short, Ph.D.

Department of Psychology

College of Charleston

shortsd@cofc.edu

For slides and code please visit `http://stephendshort.wix.com/psyc`