

Mediation and Moderation Materials

Brier Gallihugh, M.S.

2024-02-23

Table of contents

Running a Moderation Analysis in R	2
Running a Mediation Analysis in R	2
Assumptions of Moderation Analyses	5
Assumptions of Mediation Analyses	6
Using Moderation and Mediation Usings Hayes PROCESS Macro (for R)	8
A Moderation Example Using Hayes PROCESS Macro	9
A Mediation Example Using Hayes PROCESS Macro	10

```
set.seed(10311993)
library(mediation)
library(psych)
library(tidyverse)

# Created Toy Data Set
# Variance Covariance
sigma <- rbind(c(1,-0.4,-0.3), c(-0.4,1, 0.7), c(-0.3,0.7,1))
# Variable Mean
mu <- c(7, 50, 7)
# Generate the Multivariate Normal Distribution
df <- as.data.frame(mvrnorm(n=100, mu=mu, Sigma=sigma))
df <- round(df,0)
colnames(df) <- c("mediator1","outcome","predictor")
df$condition <- rep(1:2,50)
```

Running a Moderation Analysis in R

```
moderation <- lm(outcome ~ condition*predictor, data = df) ①  
summary(moderation) ②
```

- ① Create a mediation object using the `lm()` function. The `condition*predictor` syntax gets you both the main effects of condition and predictor as well as the interaction effect between the two
- ② Show a summary of the moderation using the `summary()` function.

Call:

```
lm(formula = outcome ~ condition * predictor, data = df)
```

Residuals:

Min	1Q	Median	3Q	Max
-1.79555	-0.56073	-0.05061	0.55043	1.71457

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	44.85018	1.68125	26.677	< 2e-16 ***
condition	-0.01414	1.06533	-0.013	0.98943
predictor	0.76026	0.23452	3.242	0.00163 **
condition:predictor	-0.01533	0.14964	-0.102	0.91864

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.8027 on 96 degrees of freedom

Multiple R-squared: 0.5089, Adjusted R-squared: 0.4936

F-statistic: 33.16 on 3 and 96 DF, p-value: 8.49e-15

Running a Mediation Analysis in R

```
#Regress M on X  
outcomeM_fit <- lm(mediator1 ~ condition, data = df) ①  
summary(outcomeM_fit) ②  
  
#Regress Y on M and X  
outcomeY_fit <- lm(outcome ~ mediator1 + condition, data = df) ③  
summary(outcomeY_fit) ④
```

```

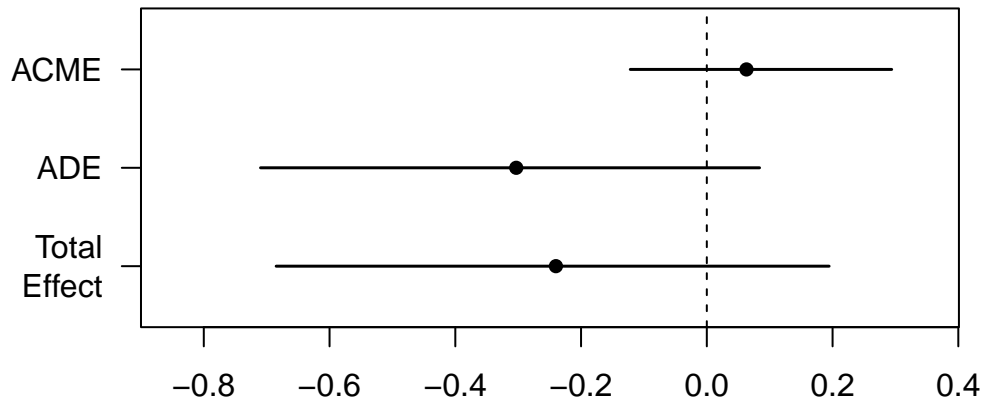
#Run Mediation with Bootstrap
outcome_fit <- mediation::mediate(outcomeM_fit,           ⑤
                                outcomeY_fit,
                                treat = "condition",
                                mediator = "mediator1",
                                boot = TRUE,
                                sims = 5000)

#Summary of Mediation
summary(outcome_fit)                                     ⑥

#Path Coefficients
plot(outcome_fit)                                       ⑦

```

- ① Run a regression of the M (mediator) on X using the `lm()` function
- ② Show output of the M on X regression using the `summary()` function
- ③ Run a regression of Y on M and X using the `lm()` function
- ④ Show output of the Y on M and X regression using the `summary()` function
- ⑤ Run a mediation using the two regressions above. `treat` is the name of your X condition. `mediator` is the name of your mediating variable. Setting `boot` to `TRUE` will ensure that your mediation is bootstrapped. Lastly, the `sims` argument tells R how many samples you wish to bootstrap from. Typically you want ~ 5000 or more.
- ⑥ For a summary of your mediation, use the `summary()` function. The indirect effect is labeled ACME
- ⑦ The `plot()` function here will give you a graphical representation of the output above with respect to the range of the confidence interval for each metric. Please note by default this is the 95% confidence interval



Call:

```
lm(formula = mediator1 ~ condition, data = df)
```

Residuals:

Min	1Q	Median	3Q	Max
-2.860	-0.755	0.140	1.140	2.280

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	7.0000	0.3412	20.515	<2e-16 ***
condition	-0.1400	0.2158	-0.649	0.518

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.079 on 98 degrees of freedom

Multiple R-squared: 0.004276, Adjusted R-squared: -0.005884

F-statistic: 0.4209 on 1 and 98 DF, p-value: 0.518

Call:

```
lm(formula = outcome ~ mediator1 + condition, data = df)
```

Residuals:

	Min	1Q	Median	3Q	Max
	-2.2245	-0.5522	-0.0769	0.4724	3.4724

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	53.53460	0.74376	71.979	< 2e-16 ***
mediator1	-0.45066	0.09569	-4.709	8.28e-06 ***
condition	-0.30309	0.20487	-1.479	0.142

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.022 on 97 degrees of freedom

Multiple R-squared: 0.1954, Adjusted R-squared: 0.1788

F-statistic: 11.78 on 2 and 97 DF, p-value: 2.634e-05

Causal Mediation Analysis

Nonparametric Bootstrap Confidence Intervals with the Percentile Method

	Estimate	95% CI Lower	95% CI Upper	p-value
ACME	0.0631	-0.1217	0.29	0.52
ADE	-0.3031	-0.7098	0.08	0.12
Total Effect	-0.2400	-0.6849	0.19	0.28
Prop. Mediated	-0.2629	-6.0955	4.66	0.76

Sample Size Used: 100

Simulations: 5000

Assumptions of Moderation Analyses

```
# Residual Normality
shapiro.test(residuals(moderation)) ①

# Multicollinearity
car::vif(moderation, type = c("predictor")) ②

# Independence of Errors
car::durbinWatsonTest(moderation) ③
```

- ① Test of the residual normality of the moderation using the `shapiro.test()` function
- ② Test of the multicollinearity of the moderation analyses using the `vif()` function in the `car` package. Because there is an interaction, you must specify an additional argument of `type = c("predictor")` to properly account for the interaction effect.
- ③ To test the independence of errors assumption, you can do so using the `durbinWatsonTest()` function from the `car` package.

Shapiro-Wilk normality test

```
data: residuals(moderation)
W = 0.98684, p-value = 0.4272
```

```

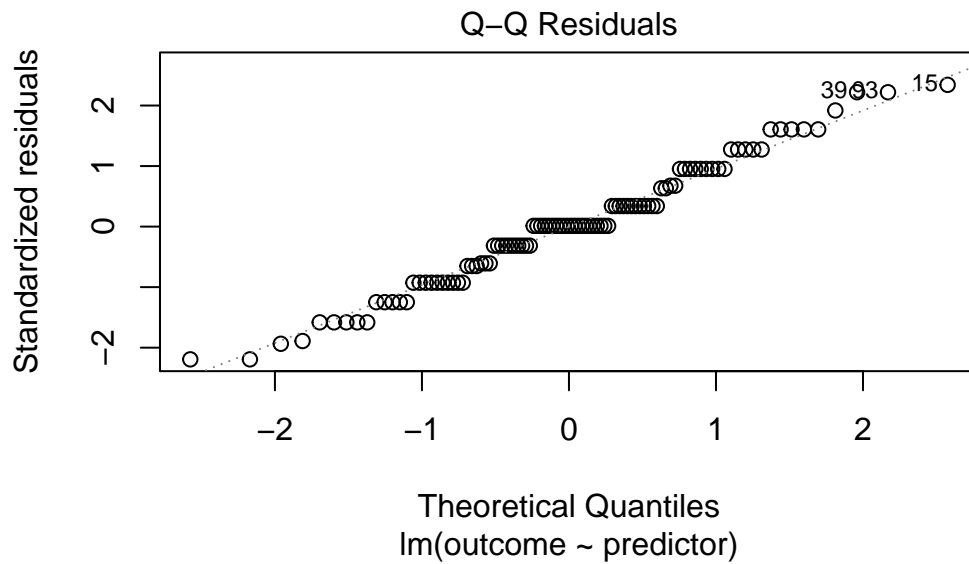
          GVIF Df GVIF^(1/(2*Df)) Interacts With Other Predictors
condition   1  3              1      predictor              --
predictor   1  3              1      condition              --
lag Autocorrelation D-W Statistic p-value
  1      -0.02268275      2.029087  0.756
Alternative hypothesis: rho != 0
```

Assumptions of Mediation Analyses

```
# Linearity
plot(lm(outcome ~ predictor, data = df), 2)
```

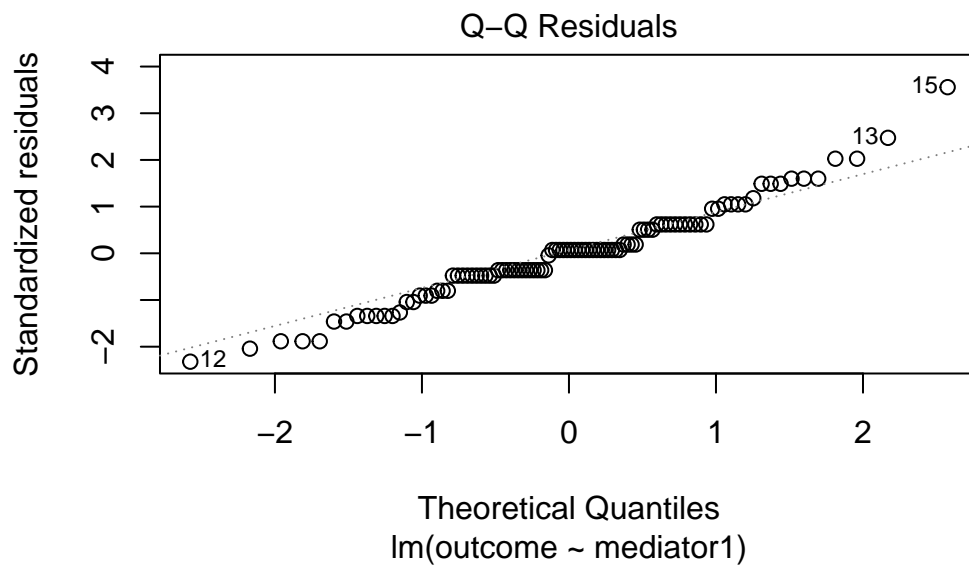
①

- ② To assess multicollinearity, the best course of action is a simple correlation matrix. You can achieve this using the `cor()` function for a correlation matrix



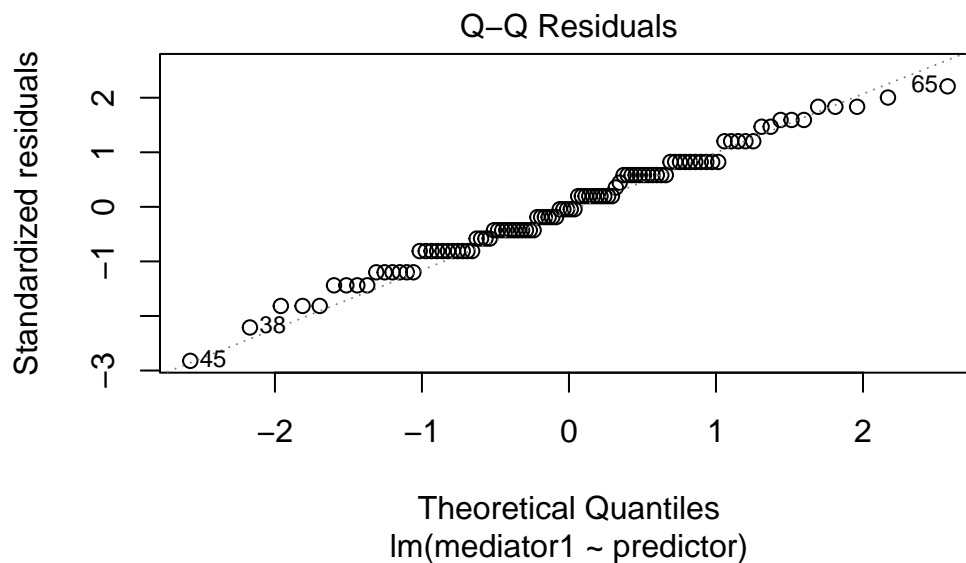
```
plot(lm(outcome ~ mediator1, data = df), 2)
```

①



```
plot(lm(mediator1 ~ predictor, data = df), 2)
```

①



```
# Multicollinearity
cor(df)
```

②

	mediator1	outcome	predictor	condition
mediator1	1.00000000	-0.4210068	-0.38328907	-0.06539201
outcome	-0.42100683	1.00000000	0.71129322	-0.10692147
predictor	-0.38328907	0.7112932	1.00000000	-0.07432941
condition	-0.06539201	-0.1069215	-0.07432941	1.00000000

Using Moderation and Mediation Usings Hayes PROCESS Macro (for R)

Click on the following [link](#) to download the R script for the PROCESS macro for R.

```
source("process.R")
```

***** PROCESS for R Version 4.3.1 *****

Written by Andrew F. Hayes, Ph.D. www.afhayes.com

Documentation available in Hayes (2022). www.guilford.com/p/hayes3

PROCESS is now ready for use.

Copyright 2020–2023 by Andrew F. Hayes ALL RIGHTS RESERVED

Workshop schedule at <http://haskayne.ucalgary.ca/CCRAM>

A Moderation Example Using Hayes PROCESS Macro

```
process(data = df,
        y = "outcome",
        x = "predictor",
        w = "mediator1",
        model = 1,
        stand = 1)
```

①
②
③
④
⑤
⑥

- ① Assign your data to the `data` argument
- ② Assign your outcome variable to the `y` argument
- ③ Assign your predictor variable to the `x` argument
- ④ Assign your moderator to the `w` argument
- ⑤ Set your `model` argument to 1 for simple moderation
- ⑥ The `stand = 1` argument standardizes your output

***** PROCESS for R Version 4.3.1 *****

Written by Andrew F. Hayes, Ph.D. www.afhayes.com

Documentation available in Hayes (2022). www.guilford.com/p/hayes3

Model : 1
Y : outcome
X : predictor
W : mediator1

Sample size: 100

Outcome Variable: outcome

Model Summary:

R	R-sq	MSE	F	df1	df2	p
0.7294	0.5320	0.6141	36.3739	3.0000	96.0000	0.0000

Model:

	coeff	se	t	p	LLCI	ULCI
constant	47.3198	3.6872	12.8336	0.0000	40.0008	54.6389
predictor	0.5567	0.5256	1.0592	0.2922	-0.4866	1.6001
mediator1	-0.2975	0.5240	-0.5676	0.5716	-1.3377	0.7427
Int_1	0.0169	0.0761	0.2222	0.8246	-0.1341	0.1679

Product terms key:

Int_1 : predictor x mediator1

Test(s) of highest order unconditional interaction(s):

	R2-chng	F	df1	df2	p
X*W	0.0002	0.0494	1.0000	96.0000	0.8246

***** ANALYSIS NOTES AND ERRORS *****

Level of confidence for all confidence intervals in output: 95

NOTE: Standardized coefficients not available for models with moderators.

Tip

The Hayes PROCESS for R requires that all data is numeric in nature. As such, ensure that any potential factor variables are numeric prior to running the analyses. A failure to do so will result in PROCESS not running.

A Mediation Example Using Hayes PROCESS Macro

```
process(data = df,
  y = "outcome",
  x = "predictor",
  m = "mediator1",
  model = 4,
```

①

②

③

④

⑤

```
stand = 1,
boot = 5000)
```

⑥
⑦

- ① Assign your data to the `data` argument
- ② Assign your outcome variable to the `y` argument
- ③ Assign your predictor variable to the `x` argument
- ④ Assign your mediator to the `m` argument
- ⑤ Set your `model` argument to 4 for simple mediation
- ⑥ The `stand = 1` argument standardizes your output
- ⑦ The `boot` argument specifies the number of samples you wish to bootstrap

***** PROCESS for R Version 4.3.1 *****

Written by Andrew F. Hayes, Ph.D. www.afhayes.com
Documentation available in Hayes (2022). www.guilford.com/p/hayes3

Model : 4
Y : outcome
X : predictor
M : mediator1

Sample size: 100

Random seed: 818206

Outcome Variable: mediator1

Model Summary:

	R	R-sq	MSE	F	df1	df2	p
	0.3833	0.1469	0.9975	16.8766	1.0000	98.0000	0.0001

Model:

	coeff	se	t	p	LLCI	ULCI
constant	9.4738	0.6609	14.3352	0.0000	8.1623	10.7852
predictor	-0.3812	0.0928	-4.1081	0.0001	-0.5654	-0.1971

Standardized coefficients:
coeff

predictor -0.3833

Outcome Variable: outcome

Model Summary:

R	R-sq	MSE	F	df1	df2	p
0.7292	0.5317	0.6081	55.0760	2.0000	97.0000	0.0000

Model:

	coeff	se	t	p	LLCI	ULCI
constant	46.5259	0.9080	51.2386	0.0000	44.7237	48.3281
predictor	0.6722	0.0784	8.5694	0.0000	0.5165	0.8279
mediator1	-0.1824	0.0789	-2.3121	0.0229	-0.3389	-0.0258

Standardized coefficients:

	coeff
predictor	0.6446
mediator1	-0.1740

Bootstrapping progress:

		0%
		1%
>		1%
>		2%
>>		2%
>>		3%
>>		4%
>>>		4%
>>>		5%
>>>		6%

>>>>	6%
>>>>	7%
>>>>>	7%
>>>>>	8%
>>>>>	9%
>>>>>>	9%
>>>>>>	10%
>>>>>>>	10%
>>>>>>>	11%
>>>>>>>	12%
>>>>>>>>	12%
>>>>>>>>	13%
>>>>>>>>	14%
>>>>>>>>>	14%
>>>>>>>>>	15%
>>>>>>>>>>	15%
>>>>>>>>>>	16%
>>>>>>>>>>	17%
>>>>>>>>>>>	17%
>>>>>>>>>>>	18%
>>>>>>>>>>>	19%

