In [1]:
```r
# Model 2
# Assuming you have a data frame named 'data' with columns: BFI..E., BFI..A., BF
data <- read.csv("Complete_Data_Modified.csv")

# Define your variables
independent_vars <- c('BFI..E.', 'BFI..A.', 'BFI..C.', 'BFI..N.', 'BFI..O.')
dependent_var <- 'Panas..'
mediator_var <- 'PAQ'

# Initial guess for the parameters (adjust as needed)
initial_parameters <- list(a = 1, b = 1, c = 1, d = 1, e = 1, f = 1)

# Define the nonlinear model
nonlinear_model <- nls(Panas.. ~ a * BFI..E. + b * BFI..A. + c * BFI..C. + d * B
                       data = data,
                       start = initial_parameters)

# Display the model coefficients and formula
cat("Fitted Coefficients:\n")
print(coef(nonlinear_model))

cat("\nModel Formula:\n")
cat("Panas.. = a * BFI..E. + b * BFI..A. + c * BFI..C. + d * BFI..N. + e * BFI..

#residuals(nonlinear_model)  # Access the residuals

# Display the summary of the model
summary(nonlinear_model)
```

```
Fitted Coefficients:
            a             b             c             d             e
 0.1231050805  0.2256167879  0.0003251576  0.1796053621  0.3799882346
            f
-0.1064233145

Model Formula:
Panas.. = a * BFI..E. + b * BFI..A. + c * BFI..C. + d * BFI..N. + e * BFI..O. +
f * PAQ
Formula: Panas.. ~ a * BFI..E. + b * BFI..A. + c * BFI..C. + d * BFI..N. +
    e * BFI..O. + f * PAQ

Parameters:
    Estimate Std. Error t value Pr(>|t|)
a  0.1231051  0.0851504   1.446  0.14982
b  0.2256168  0.0919596   2.453  0.01501 *
c  0.0003252  0.0908500   0.004  0.99715
d  0.1796054  0.0636745   2.821  0.00528 **
e  0.3799882  0.1027501   3.698  0.00028 ***
f -0.1064233  0.0563148  -1.890  0.06023 .
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.5886 on 200 degrees of freedom

Number of iterations to convergence: 1
Achieved convergence tolerance: 2.816e-06
```

In [2]:
```r
# Model 2
# Assuming you have a data frame named 'data' with columns: BFI..E., BFI..A., BF
```

```
data <- read.csv("Complete_Data_Modified.csv")

# Define your variables
independent_vars <- c('BFI..E.', 'BFI..A.', 'BFI..C.', 'BFI..N.', 'BFI..O.')
dependent_var <- 'Panas...1'
mediator_var <- 'PAQ'

# Initial guess for the parameters (adjust as needed)
initial_parameters <- list(a = 1, b = 1, c = 1, d = 1, e = 1, f = 1)

# Define the nonlinear model
nonlinear_model <- nls(Panas...1 ~ a * BFI..E. + b * BFI..A. + c * BFI..C. + d *
                       data = data,
                       start = initial_parameters)

# Display the model coefficients and formula
cat("Fitted Coefficients:\n")
print(coef(nonlinear_model))

cat("\nModel Formula:\n")
cat("Panas...1 = a * BFI..E. + b * BFI..A. + c * BFI..C. + d * BFI..N. + e * BFI

#residuals(nonlinear_model)  # Access the residuals

# Display the summary of the model
summary(nonlinear_model)
```

```
Fitted Coefficients:
          a           b           c           d           e           f
 0.02401384  0.05685320  0.21472128  0.24344070  0.13998985 -0.06541089

Model Formula:
Panas...1 = a * BFI..E. + b * BFI..A. + c * BFI..C. + d * BFI..N. + e * BFI..O.
+ f * PAQ
Formula: Panas...1 ~ a * BFI..E. + b * BFI..A. + c * BFI..C. + d * BFI..N. +
    e * BFI..O. + f * PAQ

Parameters:
  Estimate Std. Error t value Pr(>|t|)
a  0.02401    0.06393   0.376  0.70758
b  0.05685    0.06904   0.823  0.41121
c  0.21472    0.06821   3.148  0.00189 **
d  0.24344    0.04780   5.092 8.14e-07 ***
e  0.13999    0.07714   1.815  0.07106 .
f -0.06541    0.04228  -1.547  0.12341
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.4419 on 200 degrees of freedom

Number of iterations to convergence: 1
Achieved convergence tolerance: 4.962e-08
```

In [4]:

```
# Model 2
# Assuming you have a data frame named 'data' with columns: BFI..E., BFI..A., BF
data <- read.csv("Complete_Data_Modified.csv")

# Define your variables
independent_vars <- c('BFI..E.', 'BFI..A.', 'BFI..C.', 'BFI..N.', 'BFI..O.')
dependent_var <- 'CBCL'
```

```
mediator_var <- 'PAQ'

# Initial guess for the parameters (adjust as needed)
initial_parameters <- list(a = 1, b = 1, c = 1, d = 1, e = 1, f = 1)

# Define the nonlinear model
nonlinear_model <- nls(CBCL ~ a * BFI..E. + b * BFI..A. + c * BFI..C. + d * BFI.
                       data = data,
                       start = initial_parameters)

# Display the model coefficients and formula
cat("Fitted Coefficients:\n")
print(coef(nonlinear_model))

cat("\nModel Formula:\n")
cat("CBCL = a * BFI..E. + b * BFI..A. + c * BFI..C. + d * BFI..N. + e * BFI..O.

#residuals(nonlinear_model)  # Access the residuals

# Display the summary of the model
summary(nonlinear_model)
```

```
Fitted Coefficients:
          a           b           c           d           e           f
 0.348180733 -0.091720702  0.117568530  0.567517272  0.225236921  0.007797156

Model Formula:
CBCL = a * BFI..E. + b * BFI..A. + c * BFI..C. + d * BFI..N. + e * BFI..O. + f
* PAQ
Formula: CBCL ~ a * BFI..E. + b * BFI..A. + c * BFI..C. + d * BFI..N. +
    e * BFI..O. + f * PAQ

Parameters:
   Estimate Std. Error t value Pr(>|t|)
a  0.348181   0.112178   3.104  0.00219 **
b -0.091721   0.121148  -0.757  0.44988
c  0.117569   0.119686   0.982  0.32714
d  0.567517   0.083886   6.765 1.44e-10 ***
e  0.225237   0.135364   1.664  0.09769 .
f  0.007797   0.074190   0.105  0.91640
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.7754 on 200 degrees of freedom

Number of iterations to convergence: 1
Achieved convergence tolerance: 3.211e-07
```

In [5]:

```
# Model 2
# Assuming you have a data frame named 'data' with columns: EI..Self.A., EI..Sel
data <- read.csv("Complete_Data_Modified.csv")

# Define your variables
independent_vars <- c('EI..Self.A.', 'EI..Self.M.', 'EI..Social.A.', 'EI..RM.',
dependent_var <- 'Panas..'
mediator_var <- 'PAQ'

# Initial guess for the parameters (adjust as needed)
initial_parameters <- list(a = 1, b = 1, c = 1, d = 1, e = 1, f = 1)
```

```
# Define the nonlinear model
nonlinear_model <- nls(Panas.. ~ a * EI..Self.A. + b * EI..Self.M. + c * EI..Soc
                       data = data,
                       start = initial_parameters)

# Display the model coefficients and formula
cat("Fitted Coefficients:\n")
print(coef(nonlinear_model))

cat("\nModel Formula:\n")
cat("Panas.. = a * EI..Self.A. + b * EI..Self.M. + c * EI..Social.A. + d * EI..R

#residuals(nonlinear_model)  # Access the residuals

# Display the summary of the model
summary(nonlinear_model)
```

```
Fitted Coefficients:
            a             b             c             d             e
 0.0139406756  0.0118589558  0.0091191634  0.0002821801 -0.3059018756
            f
-0.0628522582

Model Formula:
Panas.. = a * EI..Self.A. + b * EI..Self.M. + c * EI..Social.A. + d * EI..RM. +
e * Total + f * PAQ
Formula: Panas.. ~ a * EI..Self.A. + b * EI..Self.M. + c * EI..Social.A. +
    d * EI..RM. + e * Total + f * PAQ

Parameters:
    Estimate Std. Error t value Pr(>|t|)
a  0.0139407  0.0051482   2.708 0.007358 **
b  0.0118590  0.0045646   2.598 0.010074 *
c  0.0091192  0.0055811   1.634 0.103845
d  0.0002822  0.0054271   0.052 0.958585
e -0.3059019  0.0807584  -3.788 0.000201 ***
f -0.0628523  0.0550553  -1.142 0.254977
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.5982 on 200 degrees of freedom

Number of iterations to convergence: 1
Achieved convergence tolerance: 2.896e-07
```

In [6]:

```
# Model 2
# Assuming you have a data frame named 'data' with columns: EI..Self.A., EI..Sel
data <- read.csv("Complete_Data_Modified.csv")

# Define your variables
independent_vars <- c('EI..Self.A.', 'EI..Self.M.', 'EI..Social.A.', 'EI..RM.',
dependent_var <- 'Panas...1'
mediator_var <- 'PAQ'

# Initial guess for the parameters (adjust as needed)
initial_parameters <- list(a = 1, b = 1, c = 1, d = 1, e = 1, f = 1)

# Define the nonlinear model
nonlinear_model <- nls(Panas...1 ~ a * EI..Self.A. + b * EI..Self.M. + c * EI..S
                       data = data,
```

```
                        start = initial_parameters)

# Display the model coefficients and formula
cat("Fitted Coefficients:\n")
print(coef(nonlinear_model))

cat("\nModel Formula:\n")
cat("Panas...1 = a * EI..Self.A. + b * EI..Self.M. + c * EI..Social.A. + d * EI.

#residuals(nonlinear_model)  # Access the residuals

# Display the summary of the model
summary(nonlinear_model)
```

```
Fitted Coefficients:
           a            b            c            d            e            f
 0.004738517  0.005746909  0.017362128 -0.006548561 -0.113159044 -0.040845458

Model Formula:
Panas...1 = a * EI..Self.A. + b * EI..Self.M. + c * EI..Social.A. + d * EI..RM.
+ e * Total + f * PAQ
Formula: Panas...1 ~ a * EI..Self.A. + b * EI..Self.M. + c * EI..Social.A. +
    d * EI..RM. + e * Total + f * PAQ

Parameters:
   Estimate Std. Error t value Pr(>|t|)
a  0.004739   0.003534   1.341   0.1815
b  0.005747   0.003134   1.834   0.0682 .
c  0.017362   0.003832   4.531 1.01e-05 ***
d -0.006549   0.003726  -1.758   0.0803 .
e -0.113159   0.055444  -2.041   0.0426 *
f -0.040845   0.037798  -1.081   0.2812
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.4107 on 200 degrees of freedom

Number of iterations to convergence: 1
Achieved convergence tolerance: 2.919e-08
```

In [7]:

```
# Model 2
# Assuming you have a data frame named 'data' with columns: EI..Self.A., EI..Sel
data <- read.csv("Complete_Data_Modified.csv")

# Define your variables
independent_vars <- c('EI..Self.A.', 'EI..Self.M.', 'EI..Social.A.', 'EI..RM.',
dependent_var <- 'CBCL'
mediator_var <- 'PAQ'

# Initial guess for the parameters (adjust as needed)
initial_parameters <- list(a = 1, b = 1, c = 1, d = 1, e = 1, f = 1)

# Define the nonlinear model
nonlinear_model <- nls(CBCL ~ a * EI..Self.A. + b * EI..Self.M. + c * EI..Social
                       data = data,
                       start = initial_parameters)

# Display the model coefficients and formula
cat("Fitted Coefficients:\n")
print(coef(nonlinear_model))
```

```
cat("\nModel Formula:\n")
cat("CBCL = a * EI..Self.A. + b * EI..Self.M. + c * EI..Social.A. + d * EI..RM.

#residuals(nonlinear_model)  # Access the residuals

# Display the summary of the model
summary(nonlinear_model)
```

```
Fitted Coefficients:
           a            b            c            d            e            f
 0.011329674  0.013885071  0.011294803  0.003867066 -0.186026861  0.015969155

Model Formula:
CBCL = a * EI..Self.A. + b * EI..Self.M. + c * EI..Social.A. + d * EI..RM. + e
* Total + f * PAQ
Formula: CBCL ~ a * EI..Self.A. + b * EI..Self.M. + c * EI..Social.A. +
    d * EI..RM. + e * Total + f * PAQ

Parameters:
   Estimate Std. Error t value Pr(>|t|)
a  0.011330   0.006317   1.794   0.0744 .
b  0.013885   0.005601   2.479   0.0140 *
c  0.011295   0.006848   1.649   0.1006
d  0.003867   0.006659   0.581   0.5621
e -0.186027   0.099087  -1.877   0.0619 .
f  0.015969   0.067550   0.236   0.8134
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.734 on 200 degrees of freedom

Number of iterations to convergence: 1
Achieved convergence tolerance: 8.366e-08
```

In [ ]: