

CIs on mxAlgebra

Mike W.-L. Cheung

8 Jun 2019

Optimizer: NPSOL

Multiplied by 2: slight differences on both lbound and ubound

```
library(OpenMx)

## To take full advantage of multiple cores, use:
##   mxOption(key='Number of Threads', value=parallel::detectCores()) #now
##   Sys.setenv(OMP_NUM_THREADS=parallel::detectCores()) #before library(OpenMx)
mxOption(NULL, "Default optimizer", "NPSOL")

set.seed(1000)

my.df <- data.frame(x=rnorm(100, mean=0, sd=1))

mu <- mxMatrix(type="Full", nrow=1, ncol=1,
               free=TRUE, values=0, labels="mean", name="mu")
sigma <- mxMatrix(type="Symm", nrow=1, ncol=1,
                 free=TRUE, values=1, labels="variance", name="sigma")

expectation <- mxExpectationNormal(covariance="sigma", means="mu", dimnames = "x")

model2 <- mxModel("Two", mxData(my.df, type="raw"), expectation,
                 mxFitFunctionML(), mu, sigma,
                 mxAlgebra(variance*2, "two_variance"),
                 mxCI(c("variance", "two_variance")))
fit2 <- mxRun(model2, intervals = TRUE)

## Running Two with 2 parameters
summary(fit2, verbose=TRUE)

## Summary of Two
##
## data:
## $Two.data
##      x
## Min.   :-2.32249
## 1st Qu.: -0.55924
## Median : 0.04118
## Mean    : 0.01638
## 3rd Qu.: 0.57152
## Max.    : 2.67007
##
## free parameters:
##      name matrix row col Estimate Std.Error A lbound ubound
```

```

## 1      mean      mu      1      x 0.01637923 0.1001549
## 2 variance  sigma      x      x 1.00309902 0.1418596
##
## confidence intervals:
##
##                lbound estimate  ubound note
## variance                0.7693045 1.003099 1.342783
## Two.two_variance[1,1] 1.5379716 2.006198 2.688370
##
## CI details:
##
##           parameter side      value      fit diagnostic statusCode
## 1           variance lower 0.7693045 287.9513      success          OK
## 2           variance upper 1.3427833 287.9651      success          OK
## 3 Two.two_variance[1,1] lower 1.5379716 287.9639      success          OK
## 4 Two.two_variance[1,1] upper 2.6883698 287.9916      success          OK
##
##           method      mean variance
## 1 neale-miller-1997 0.01637921 0.7693045
## 2 neale-miller-1997 0.01637922 1.3427833
## 3 neale-miller-1997 0.01637922 0.7689858
## 4 neale-miller-1997 0.01637921 1.3441849
##
## Model Statistics:
##
##           | Parameters | Degrees of Freedom | Fit (-2lnL units)
##           |-----|-----|-----|-----|
## Model:           2           98           284.0971
## Saturated:       2           98           NA
## Independence:    2           98           NA
##
## Number of observations/statistics: 100/100
##
## condition number of the information matrix: 2.006196
## maximum absolute gradient: 1.50232e-05 ( variance )
## chi-square:  2 ( df=NA ) = NA, p = 1
## Information Criteria:
##
##           | df Penalty | Parameters Penalty | Sample-Size Adjusted
## AIC:           88.09714           288.0971           288.2209
## BIC:          -167.20953           293.3075           286.9910
## CFI: NA
## TLI: 1 (also known as NNFI)
## RMSEA: 0 [95% CI (NA, NA)]
## Prob(RMSEA <= 0.05): NA
## To get additional fit indices, see help(mxRefModels)
## timestamp: 2019-06-08 14:26:19
## frontend time: 0.154892 secs
## backend time: 0.002310038 secs
## independent submodels time: 7.05719e-05 secs
## cpu time: 0.1572726 secs
## Wall clock time: 0.1572726 secs
## OpenMx version number: 2.13.2
## Need help? See help(mxSummary)

```

```

CI2 <- summary(fit2)$CI
CI2.diff <- data.frame(t(CI2[, 1:3]))
CI2.diff$variance_x2 <- 2*CI2.diff[, 1]
CI2.diff$diff <- CI2.diff[, 2] - CI2.diff[, 3]
CI2.diff

```

```

##           variance Two.two_variance.1.1. variance_x2           diff

```

```
## lbound    0.7693045          1.537972    1.538609 -0.0006372986
## estimate  1.0030990          2.006198    2.006198  0.0000000000
## ubound    1.3427833          2.688370    2.685567  0.0028032572
```

Multiplied by 5: slight difference on the lbound and NA on the ubound

```
model5 <- mxModel("Five", mxData(my.df, type="raw"), expectation,
                  mxFitFunctionML(), mu, sigma,
                  mxAlgebra(variance*5, "five_variance"),
                  mxCI(c("variance", "five_variance")))
fit5 <- mxRun(model5, intervals = TRUE)
```

```
## Running Five with 2 parameters
```

```
summary(fit5, verbose=TRUE)
```

```
## Summary of Five
##
## data:
## $Five.data
##      x
## Min.   :-2.32249
## 1st Qu.: -0.55924
## Median :  0.04118
## Mean    :  0.01638
## 3rd Qu.:  0.57152
## Max.    :  2.67007
##
## free parameters:
##      name matrix row col  Estimate Std.Error A lbound ubound
## 1    mean      mu    1   x  0.01637923 0.1001549
## 2 variance  sigma   x   x  1.00309902 0.1418596
##
## confidence intervals:
##                lbound estimate  ubound note
## variance                0.7693045 1.003099 1.342783
## Five.five_variance[1,1] 3.8402109 5.015495      NA !!!
##
## CI details:
##                parameter side      value      fit      diagnostic
## 1                variance lower 0.7693045 287.9513      success
## 2                variance upper 1.3427833 287.9651      success
## 3 Five.five_variance[1,1] lower 3.8402109 288.0013      success
## 4 Five.five_variance[1,1] upper 6.7417010 288.0703 alpha level not reached
##                statusCode          method      mean variance
## 1                OK neale-miller-1997 0.01637921 0.7693045
## 2                OK neale-miller-1997 0.01637922 1.3427833
## 3                OK neale-miller-1997 0.01637922 0.7680422
## 4 infeasible non-linear constraint neale-miller-1997 0.01637922 1.3483402
##
## Model Statistics:
##                | Parameters | Degrees of Freedom | Fit (-2lnL units)
##                |-----|-----|-----|-----|
## Model:                2                98                284.0971
## Saturated:            2                98                NA
```

```

## Independence:                2                98                NA
## Number of observations/statistics: 100/100
##
## condition number of the information matrix: 2.006196
## maximum absolute gradient: 1.50232e-05 ( variance )
## chi-square:  ^2 ( df=NA ) = NA,  p = 1
## Information Criteria:
##      | df Penalty | Parameters Penalty | Sample-Size Adjusted
## AIC:      88.09714                288.0971                288.2209
## BIC:     -167.20953                293.3075                286.9910
## CFI: NA
## TLI: 1 (also known as NNFI)
## RMSEA: 0 [95% CI (NA, NA)]
## Prob(RMSEA <= 0.05): NA
## To get additional fit indices, see help(mxRefModels)
## timestamp: 2019-06-08 14:26:19
## frontend time: 0.03781748 secs
## backend time: 0.002791643 secs
## independent submodels time: 2.074242e-05 secs
## cpu time: 0.04062986 secs
## Wall clock time: 0.04062986 secs
## OpenMx version number: 2.13.2
## Need help? See help(mxSummary)

```

```

CI5 <- summary(fit5)$CI
CI5.diff <- data.frame(t(CI5[, 1:3]))
CI5.diff$variance_x5 <- 5*CI5.diff[, 1]
CI5.diff$diff <- CI5.diff[, 2] - CI5.diff[, 3]
CI5.diff

```

```

##      variance Five.five_variance.1.1. variance_x5      diff
## lbound  0.7693045                3.840211    3.846522 -0.006311446
## estimate 1.0030990                5.015495    5.015495  0.000000000
## ubound  1.3427833                NA         6.713916         NA

```

Multiplied by 10: NA in both lbound and ubound

```

model10 <- mxModel("Ten", mxData(my.df, type="raw"), expectation,
  mxFitFunctionML(), mu, sigma,
  mxAlgebra(variance*10, "ten_variance"),
  mxCI(c("variance", "ten_variance")))
fit10 <- mxRun(model10, intervals = TRUE)

```

```
## Running Ten with 2 parameters
```

```
summary(fit10, verbose=TRUE)
```

```

## Summary of Ten
##
## data:
## $Ten.data
##      x
## Min.   :-2.32249
## 1st Qu.: -0.55924
## Median : 0.04118

```

```

## Mean      : 0.01638
## 3rd Qu.  : 0.57152
## Max.     : 2.67007
##
## free parameters:
##      name matrix row col Estimate Std.Error A lbound ubound
## 1      mean      mu   1   x 0.01637923 0.1001549
## 2 variance  sigma   x   x 1.00309902 0.1418596
##
## confidence intervals:
##                lbound estimate ubound note
## variance                0.7693045 1.003099 1.342783
## Ten.ten_variance[1,1]          NA 10.030990          NA !!!
##
## CI details:
##      parameter side      value      fit      diagnostic
## 1      variance lower 0.7693045 287.9513      success
## 2      variance upper 1.3427833 287.9651      success
## 3 Ten.ten_variance[1,1] lower 7.6650868 288.0628 alpha level not reached
## 4 Ten.ten_variance[1,1] upper 13.5511086 288.1994 alpha level not reached
##      statusCode      method      mean variance
## 1              OK neale-miller-1997 0.01637921 0.7693045
## 2              OK neale-miller-1997 0.01637922 1.3427833
## 3 infeasible non-linear constraint neale-miller-1997 0.01637922 0.7665087
## 4 infeasible non-linear constraint neale-miller-1997 0.01637923 1.3551109
##
## Model Statistics:
##      | Parameters | Degrees of Freedom | Fit (-2lnL units)
##      Model:      2          98          284.0971
##      Saturated:  2          98          NA
##      Independence: 2          98          NA
## Number of observations/statistics: 100/100
##
## condition number of the information matrix: 2.006196
## maximum absolute gradient: 1.50232e-05 ( variance )
## chi-square:  2 ( df=NA ) = NA, p = 1
## Information Criteria:
##      | df Penalty | Parameters Penalty | Sample-Size Adjusted
##      AIC:      88.09714          288.0971          288.2209
##      BIC:     -167.20953          293.3075          286.9910
##      CFI: NA
##      TLI: 1 (also known as NNFI)
##      RMSEA: 0 [95% CI (NA, NA)]
##      Prob(RMSEA <= 0.05): NA
## To get additional fit indices, see help(mxRefModels)
## timestamp: 2019-06-08 14:26:19
## frontend time: 0.02559519 secs
## backend time: 0.00248909 secs
## independent submodels time: 7.390976e-06 secs
## cpu time: 0.02809167 secs
## Wall clock time: 0.02809167 secs
## OpenMx version number: 2.13.2
## Need help? See help(mxSummary)

```

```

CI10 <- summary(fit10)$CI
CI10.diff <- data.frame(t(CI10[, 1:3]))
CI10.diff$variance_x10 <- 10*CI10.diff[, 1]
CI10.diff$diff <- CI10.diff[, 2] - CI10.diff[, 3]
CI10.diff

##          variance Ten.ten_variance.1.1. variance_x10 diff
## lbound    0.7693045                    NA      7.693045  NA
## estimate  1.0030990                   10.03099    10.030990  0
## ubound    1.3427833                    NA     13.427833  NA

```

Optimizer: CSOLNP

Multiplied by 2: slight differences on both lbound and ubound

```

library(OpenMx)

mxOption(NULL, "Default optimizer", "CSOLNP")

set.seed(1000)

my.df <- data.frame(x=rnorm(100, mean=0, sd=1))

mu <- mxMatrix(type="Full", nrow=1, ncol=1,
               free=TRUE, values=0, labels="mean", name="mu")
sigma <- mxMatrix(type="Symm", nrow=1, ncol=1,
                  free=TRUE, values=1, labels="variance", name="sigma")

expectation <- mxExpectationNormal(covariance="sigma", means="mu", dimnames = "x")

model2 <- mxModel("Two", mxData(my.df, type="raw"), expectation,
                  mxFitFunctionML(), mu, sigma,
                  mxAlgebra(variance*2, "two_variance"),
                  mxCI(c("variance", "two_variance")))

fit2 <- mxRun(model2, intervals = TRUE)

```

Running Two with 2 parameters

```
summary(fit2, verbose=TRUE)
```

```

## Summary of Two
##
## data:
## $Two.data
##      x
## Min.   :-2.32249
## 1st Qu.: -0.55924
## Median : 0.04118
## Mean    : 0.01638
## 3rd Qu.: 0.57152
## Max.    : 2.67007
##
## free parameters:

```

```

##      name matrix row col  Estimate Std.Error A lbound ubound
## 1    mean      mu   1   x 0.01637919 0.1001549
## 2 variance  sigma  x   x 1.00309920 0.1418600
##
## confidence intervals:
##                lbound estimate  ubound note
## variance                0.7693044 1.003099 1.342783
## Two.two_variance[1,1] 1.5379715 2.006198 2.688370
##
## CI details:
##      parameter side      value      fit diagnostic statusCode
## 1      variance lower 0.7693044 287.9513      success      OK
## 2      variance upper 1.3427832 287.9651      success      OK
## 3 Two.two_variance[1,1] lower 1.5379715 287.9639      success      OK
## 4 Two.two_variance[1,1] upper 2.6883697 287.9916      success      OK
##      method      mean variance
## 1 neale-miller-1997 0.01637948 0.7693044
## 2 neale-miller-1997 0.01637917 1.3427832
## 3 neale-miller-1997 0.01637916 0.7689858
## 4 neale-miller-1997 0.01637917 1.3441849
##
## Model Statistics:
##      | Parameters | Degrees of Freedom | Fit (-2lnL units)
##      Model:      2          98          284.0971
##      Saturated:  2          98          NA
##      Independence: 2          98          NA
## Number of observations/statistics: 100/100
##
## condition number of the information matrix: 2.006207
## maximum absolute gradient: 5.886704e-06 ( mean )
## chi-square:  2 ( df=NA ) = NA, p = 1
## Information Criteria:
##      | df Penalty | Parameters Penalty | Sample-Size Adjusted
##      AIC:      88.09714          288.0971          288.2209
##      BIC:     -167.20953          293.3075          286.9910
##      CFI: NA
##      TLI: 1 (also known as NNFI)
##      RMSEA: 0 [95% CI (NA, NA)]
##      Prob(RMSEA <= 0.05): NA
## To get additional fit indices, see help(mxRefModels)
## timestamp: 2019-06-08 14:26:19
## frontend time: 0.04155922 secs
## backend time: 0.003650665 secs
## independent submodels time: 2.098083e-05 secs
## cpu time: 0.04523087 secs
## Wall clock time: 0.04523087 secs
## OpenMx version number: 2.13.2
## Need help? See help(mxSummary)

```

```

CI2 <- summary(fit2)$CI
CI2.diff <- data.frame(t(CI2[, 1:3]))
CI2.diff$variance_x2 <- 2*CI2.diff[, 1]
CI2.diff$diff <- CI2.diff[, 2] - CI2.diff[, 3]
CI2.diff

```

```
##          variance Two.two_variance.1.1. variance_x2          diff
## lbound   0.7693044          1.537972    1.538609 -0.0006373491
## estimate 1.0030992          2.006198    2.006198  0.0000000000
## ubound   1.3427832          2.688370    2.685566  0.0028032732
```

Multiplied by 5: slight difference on the lbound and NA on the ubound

```
model5 <- mxModel("Five", mxData(my.df, type="raw"), expectation,
                  mxFitFunctionML(), mu, sigma,
                  mxAlgebra(variance*5, "five_variance"),
                  mxCI(c("variance", "five_variance")))
fit5 <- mxRun(model5, intervals = TRUE)
```

```
## Running Five with 2 parameters
```

```
summary(fit5, verbose=TRUE)
```

```
## Summary of Five
##
## data:
## $Five.data
##      x
## Min.   :-2.32249
## 1st Qu.: -0.55924
## Median :  0.04118
## Mean    :  0.01638
## 3rd Qu.:  0.57152
## Max.    :  2.67007
##
## free parameters:
##      name matrix row col  Estimate Std.Error A lbound ubound
## 1    mean      mu    1   x 0.01637919 0.1001549
## 2 variance  sigma   x   x 1.00309920 0.1418600
##
## confidence intervals:
##
##                lbound estimate  ubound note
## variance                0.7693044 1.003099 1.342783
## Five.five_variance[1,1] 3.8402108 5.015496      NA  !!!
##
## CI details:
##
##      parameter side      value      fit      diagnostic
## 1          variance lower 0.7693044 287.9513      success
## 2          variance upper 1.3427832 287.9651      success
## 3 Five.five_variance[1,1] lower 3.8402108 288.0013      success
## 4 Five.five_variance[1,1] upper 6.7417008 288.0703 alpha level not reached
##
##      statusCode      method      mean variance
## 1          OK neale-miller-1997 0.01637948 0.7693044
## 2          OK neale-miller-1997 0.01637917 1.3427832
## 3          OK neale-miller-1997 0.01637943 0.7680422
## 4 infeasible non-linear constraint neale-miller-1997 0.01637917 1.3483402
##
## Model Statistics:
##      | Parameters | Degrees of Freedom | Fit (-2lnL units)
##      Model:          2          98          284.0971
```



```

## Saturated:                2                98                NA
## Independence:            2                98                NA
## Number of observations/statistics: 100/100
##
## condition number of the information matrix: 2.006207
## maximum absolute gradient: 5.886704e-06 ( mean )
## chi-square:  2 ( df=NA ) = NA, p = 1
## Information Criteria:
##      | df Penalty | Parameters Penalty | Sample-Size Adjusted
## AIC:      88.09714          288.0971          288.2209
## BIC:     -167.20953          293.3075          286.9910
## CFI: NA
## TLI: 1 (also known as NNFI)
## RMSEA: 0 [95% CI (NA, NA)]
## Prob(RMSEA <= 0.05): NA
## To get additional fit indices, see help(mxRefModels)
## timestamp: 2019-06-08 14:26:19
## frontend time: 0.01882124 secs
## backend time: 0.002175093 secs
## independent submodels time: 7.390976e-06 secs
## cpu time: 0.02100372 secs
## Wall clock time: 0.02100372 secs
## OpenMx version number: 2.13.2
## Need help? See help(mxSummary)

```

```

CI5 <- summary(fit5)$CI
CI5.diff <- data.frame(t(CI5[, 1:3]))
CI5.diff$variance_x5 <- 5*CI5.diff[, 1]
CI5.diff$diff <- CI5.diff[, 2] - CI5.diff[, 3]
CI5.diff

```

```

##      variance Five.five_variance.1.1. variance_x5      diff
## lbound  0.7693044          3.840211    3.846522 -0.006311445
## estimate 1.0030992          5.015496    5.015496  0.000000000
## ubound  1.3427832          NA        6.713916          NA

```

Multiplied by 10: NA in both lbound and ubound

```

model10 <- mxModel("Ten", mxData(my.df, type="raw"), expectation,
  mxFitFunctionML(), mu, sigma,
  mxAlgebra(variance*10, "ten_variance"),
  mxCI(c("variance", "ten_variance")))
fit10 <- mxRun(model10, intervals = TRUE)

```

```
## Running Ten with 2 parameters
```

```
summary(fit10, verbose=TRUE)
```

```

## Summary of Ten
##
## data:
## $Ten.data
##      x
## Min.   :-2.32249
## 1st Qu.:-0.55924

```

```

## Median : 0.04118
## Mean   : 0.01638
## 3rd Qu.: 0.57152
## Max.   : 2.67007
##
## free parameters:
##      name matrix row col Estimate Std.Error A lbound ubound
## 1   mean      mu    1   x 0.01637919 0.1001549
## 2 variance  sigma  x   x 1.00309920 0.1418600
##
## confidence intervals:
##                lbound estimate ubound note
## variance                0.7693044 1.003099 1.342783
## Ten.ten_variance[1,1]          NA 10.030992          NA !!!
##
## CI details:
##      parameter side      value      fit      diagnostic
## 1      variance lower 0.7693044 287.9513      success
## 2      variance upper 1.3427832 287.9651      success
## 3 Ten.ten_variance[1,1] lower 7.6650864 288.0628 alpha level not reached
## 4 Ten.ten_variance[1,1] upper 13.5511087 288.1994 alpha level not reached
##      statusCode      method      mean variance
## 1              OK neale-miller-1997 0.01637948 0.7693044
## 2              OK neale-miller-1997 0.01637917 1.3427832
## 3 infeasible non-linear constraint neale-miller-1997 0.01637917 0.7665086
## 4 infeasible non-linear constraint neale-miller-1997 0.01637920 1.3551109
##
## Model Statistics:
##      | Parameters | Degrees of Freedom | Fit (-2lnL units)
##      Model:      2              98              284.0971
##      Saturated:  2              98              NA
##      Independence: 2              98              NA
## Number of observations/statistics: 100/100
##
## condition number of the information matrix: 2.006207
## maximum absolute gradient: 5.886704e-06 ( mean )
## chi-square:  2 ( df=NA ) = NA, p = 1
## Information Criteria:
##      | df Penalty | Parameters Penalty | Sample-Size Adjusted
##      AIC:      88.09714              288.0971              288.2209
##      BIC:     -167.20953              293.3075              286.9910
## CFI: NA
## TLI: 1 (also known as NNFI)
## RMSEA: 0 [95% CI (NA, NA)]
## Prob(RMSEA <= 0.05): NA
## To get additional fit indices, see help(mxRefModels)
## timestamp: 2019-06-08 14:26:19
## frontend time: 0.04312468 secs
## backend time: 0.002074718 secs
## independent submodels time: 1.0252e-05 secs
## cpu time: 0.04520965 secs
## Wall clock time: 0.04520965 secs
## OpenMx version number: 2.13.2
## Need help? See help(mxSummary)

```

```

CI10 <- summary(fit10)$CI
CI10.diff <- data.frame(t(CI10[, 1:3]))
CI10.diff$variance_x10 <- 10*CI10.diff[, 1]
CI10.diff$diff <- CI10.diff[, 2] - CI10.diff[, 3]
CI10.diff

##          variance Ten.ten_variance.1.1. variance_x10 diff
## lbound    0.7693044                      NA      7.693044  NA
## estimate  1.0030992                    10.03099    10.030992  0
## ubound    1.3427832                      NA     13.427832  NA

```

Optimizer: SLSQP

Multiplied by 2: slight differences on both lbound and ubound

```

library(OpenMx)

mxOption(NULL, "Default optimizer", "SLSQP")

set.seed(1000)

my.df <- data.frame(x=rnorm(100, mean=0, sd=1))

mu <- mxMatrix(type="Full", nrow=1, ncol=1,
               free=TRUE, values=0, labels="mean", name="mu")
sigma <- mxMatrix(type="Symm", nrow=1, ncol=1,
                  free=TRUE, values=1, labels="variance", name="sigma")

expectation <- mxExpectationNormal(covariance="sigma", means="mu", dimnames = "x")

model2 <- mxModel("Two", mxData(my.df, type="raw"), expectation,
                  mxFitFunctionML(), mu, sigma,
                  mxAlgebra(variance*2, "two_variance"),
                  mxCI(c("variance", "two_variance")))

fit2 <- mxRun(model2, intervals = TRUE)

```

Running Two with 2 parameters

```
summary(fit2, verbose=TRUE)
```

```

## Summary of Two
##
## data:
## $Two.data
##      x
## Min.   :-2.32249
## 1st Qu.: -0.55924
## Median : 0.04118
## Mean    : 0.01638
## 3rd Qu.: 0.57152
## Max.    : 2.67007
##
## free parameters:

```

```

##      name matrix row col  Estimate Std.Error A lbound ubound
## 1    mean      mu   1   x 0.01637922 0.1001549
## 2 variance  sigma  x   x 1.00309917 0.1418596
##
## confidence intervals:
##                lbound estimate  ubound note
## variance                0.7696252 1.003099 1.341373
## Two.two_variance[1,1] 1.5392535 2.006198 2.682746
##
## CI details:
##           parameter side      value      fit diagnostic statusCode
## 1           variance lower 0.7696252 287.9386    success          OK
## 2           variance upper 1.3413728 287.9386    success          OK
## 3 Two.two_variance[1,1] lower 1.5392535 287.9385    success          OK
## 4 Two.two_variance[1,1] upper 2.6827456 287.9386    success          OK
##           method      mean variance
## 1 neale-miller-1997 0.01637960 0.7696252
## 2 neale-miller-1997 0.01637661 1.3413728
## 3 neale-miller-1997 0.01638253 0.7696267
## 4 neale-miller-1997 0.01633743 1.3413728
##
## Model Statistics:
##           | Parameters | Degrees of Freedom | Fit (-2lnL units)
##           |-----|-----|-----|-----|
## Model:           2                98                284.0971
## Saturated:       2                98                NA
## Independence:    2                98                NA
## Number of observations/statistics: 100/100
##
## condition number of the information matrix: 2.006196
## maximum absolute gradient: 4.24081e-08 ( variance )
## chi-square:  2 ( df=NA ) = NA, p = 1
## Information Criteria:
##           | df Penalty | Parameters Penalty | Sample-Size Adjusted
## AIC:           88.09714                288.0971                288.2209
## BIC:          -167.20953                293.3075                286.9910
## CFI: NA
## TLI: 1 (also known as NNFI)
## RMSEA: 0 [95% CI (NA, NA)]
## Prob(RMSEA <= 0.05): NA
## To get additional fit indices, see help(mxRefModels)
## timestamp: 2019-06-08 14:26:19
## frontend time: 0.03797221 secs
## backend time: 0.002266884 secs
## independent submodels time: 7.867813e-06 secs
## cpu time: 0.04024696 secs
## Wall clock time: 0.04024696 secs
## OpenMx version number: 2.13.2
## Need help? See help(mxSummary)

```

```

CI2 <- summary(fit2)$CI
CI2.diff <- data.frame(t(CI2[, 1:3]))
CI2.diff$variance_x2 <- 2*CI2.diff[, 1]
CI2.diff$diff <- CI2.diff[, 2] - CI2.diff[, 3]
CI2.diff

```

```
##          variance Two.two_variance.1.1. variance_x2          diff
## lbound   0.7696252                1.539253    1.539250 2.986838e-06
## estimate 1.0030992                2.006198    2.006198 0.000000e+00
## ubound   1.3413728                2.682746    2.682746 6.663023e-08
```

Multiplied by 5: slight difference on both lbound and ubound

```
model5 <- mxModel("Five", mxData(my.df, type="raw"), expectation,
                  mxFitFunctionML(), mu, sigma,
                  mxAlgebra(variance*5, "five_variance"),
                  mxCI(c("variance", "five_variance")))
fit5 <- mxRun(model5, intervals = TRUE)
```

```
## Running Five with 2 parameters
```

```
summary(fit5, verbose=TRUE)
```

```
## Summary of Five
##
## data:
## $Five.data
##      x
## Min.   :-2.32249
## 1st Qu.: -0.55924
## Median : 0.04118
## Mean    : 0.01638
## 3rd Qu.: 0.57152
## Max.    : 2.67007
##
## free parameters:
##      name matrix row col Estimate Std.Error A lbound ubound
## 1 mean mu 1 x 0.01637922 0.1001549
## 2 variance sigma x x 1.00309917 0.1418596
##
## confidence intervals:
##      lbound estimate ubound note
## variance          0.7696252 1.003099 1.341373
## Five.five_variance[1,1] 3.8481294 5.015496 6.706860
##
## CI details:
##      parameter side value fit diagnostic statusCode
## 1 variance lower 0.7696252 287.9386 success OK
## 2 variance upper 1.3413728 287.9386 success OK
## 3 Five.five_variance[1,1] lower 3.8481294 287.9386 success OK
## 4 Five.five_variance[1,1] upper 6.7068604 287.9386 success OK
##      method mean variance
## 1 neale-miller-1997 0.01637960 0.7696252
## 2 neale-miller-1997 0.01637661 1.3413728
## 3 neale-miller-1997 0.01618869 0.7696259
## 4 neale-miller-1997 0.01633227 1.3413721
##
## Model Statistics:
##      | Parameters | Degrees of Freedom | Fit (-2lnL units)
##      Model:          2          98          284.0971
```

```

## Saturated:                2                98                NA
## Independence:             2                98                NA
## Number of observations/statistics: 100/100
##
## condition number of the information matrix: 2.006196
## maximum absolute gradient: 4.24081e-08 ( variance )
## chi-square:  ^2 ( df=NA ) = NA, p = 1
## Information Criteria:
##      | df Penalty | Parameters Penalty | Sample-Size Adjusted
## AIC:      88.09714                288.0971                288.2209
## BIC:     -167.20953                293.3075                286.9910
## CFI: NA
## TLI: 1 (also known as NNFI)
## RMSEA: 0 [95% CI (NA, NA)]
## Prob(RMSEA <= 0.05): NA
## To get additional fit indices, see help(mxRefModels)
## timestamp: 2019-06-08 14:26:19
## frontend time: 0.04397488 secs
## backend time: 0.002664089 secs
## independent submodels time: 7.629395e-06 secs
## cpu time: 0.0466466 secs
## Wall clock time: 0.0466466 secs
## OpenMx version number: 2.13.2
## Need help? See help(mxSummary)

```

```

CI5 <- summary(fit5)$CI
CI5.diff <- data.frame(t(CI5[, 1:3]))
CI5.diff$variance_x5 <- 5*CI5.diff[, 1]
CI5.diff$diff <- CI5.diff[, 2] - CI5.diff[, 3]
CI5.diff

```

```

##      variance Five.five_variance.1.1. variance_x5      diff
## lbound  0.7696252                3.848129    3.848126  3.18977e-06
## estimate 1.0030992                5.015496    5.015496  0.00000e+00
## ubound  1.3413728                6.706860    6.706864 -3.42285e-06

```

Multiplied by 10: NA in both lbound and ubound

```

model10 <- mxModel("Ten", mxData(my.df, type="raw"), expectation,
  mxFitFunctionML(), mu, sigma,
  mxAlgebra(variance*10, "ten_variance"),
  mxCI(c("variance", "ten_variance")))
fit10 <- mxRun(model10, intervals = TRUE)

```

```
## Running Ten with 2 parameters
```

```
summary(fit10, verbose=TRUE)
```

```

## Summary of Ten
##
## data:
## $Ten.data
##      x
## Min.   :-2.32249
## 1st Qu.:-0.55924

```

```

## Median : 0.04118
## Mean   : 0.01638
## 3rd Qu.: 0.57152
## Max.   : 2.67007
##
## free parameters:
##      name matrix row col Estimate Std.Error A lbound ubound
## 1   mean      mu    1   x 0.01637922 0.1001549
## 2 variance  sigma  x   x 1.00309917 0.1418596
##
## confidence intervals:
##                lbound estimate  ubound note
## variance                0.7696252  1.003099  1.341373
## Ten.ten_variance[1,1] 7.6962679 10.030992 13.413719
##
## CI details:
##      parameter side      value      fit diagnostic statusCode
## 1      variance lower  0.7696252 287.9386      success          OK
## 2      variance upper  1.3413728 287.9386      success          OK
## 3 Ten.ten_variance[1,1] lower 7.6962679 287.9385      success          OK
## 4 Ten.ten_variance[1,1] upper 13.4137190 287.9386      success          OK
##      method      mean variance
## 1 neale-miller-1997 0.01637960 0.7696252
## 2 neale-miller-1997 0.01637661 1.3413728
## 3 neale-miller-1997 0.01623469 0.7696268
## 4 neale-miller-1997 0.01639234 1.3413719
##
## Model Statistics:
##      | Parameters | Degrees of Freedom | Fit (-2lnL units)
##      Model:           2           98           284.0971
##      Saturated:       2           98           NA
##      Independence:    2           98           NA
## Number of observations/statistics: 100/100
##
## condition number of the information matrix: 2.006196
## maximum absolute gradient: 4.24081e-08 ( variance )
## chi-square:  2 ( df=NA ) = NA, p = 1
## Information Criteria:
##      | df Penalty | Parameters Penalty | Sample-Size Adjusted
##      AIC:      88.09714           288.0971           288.2209
##      BIC:     -167.20953           293.3075           286.9910
## CFI: NA
## TLI: 1 (also known as NNFI)
## RMSEA: 0 [95% CI (NA, NA)]
## Prob(RMSEA <= 0.05): NA
## To get additional fit indices, see help(mxRefModels)
## timestamp: 2019-06-08 14:26:19
## frontend time: 0.02811646 secs
## backend time: 0.004070759 secs
## independent submodels time: 1.454353e-05 secs
## cpu time: 0.03220177 secs
## Wall clock time: 0.03220177 secs
## OpenMx version number: 2.13.2
## Need help? See help(mxSummary)

```

```
CI10 <- summary(fit10)$CI
CI10.diff <- data.frame(t(CI10[, 1:3]))
CI10.diff$variance_x10 <- 10*CI10.diff[, 1]
CI10.diff$diff <- CI10.diff[, 2] - CI10.diff[, 3]
CI10.diff
```

```
##           variance Ten.ten_variance.1.1. variance_x10           diff
## lbound    0.7696252                7.696268    7.696252  1.549558e-05
## estimate  1.0030992                10.030992   10.030992  0.000000e+00
## ubound    1.3413728                13.413719   13.413728 -8.626028e-06
```

```
sessionInfo()
```

```
## R version 3.6.0 (2019-04-26)
## Platform: x86_64-pc-linux-gnu (64-bit)
## Running under: Ubuntu 18.04.2 LTS
##
## Matrix products: default
## BLAS: /usr/lib/x86_64-linux-gnu/blas/libblas.so.3.7.1
## LAPACK: /usr/lib/x86_64-linux-gnu/lapack/liblapack.so.3.7.1
##
## locale:
## [1] LC_CTYPE=en_SG.UTF-8      LC_NUMERIC=C
## [3] LC_TIME=en_SG.UTF-8       LC_COLLATE=en_SG.UTF-8
## [5] LC_MONETARY=en_SG.UTF-8   LC_MESSAGES=en_SG.UTF-8
## [7] LC_PAPER=en_SG.UTF-8     LC_NAME=C
## [9] LC_ADDRESS=C              LC_TELEPHONE=C
## [11] LC_MEASUREMENT=en_SG.UTF-8 LC_IDENTIFICATION=C
##
## attached base packages:
## [1] stats      graphics  grDevices  utils      datasets  methods   base
##
## other attached packages:
## [1] OpenMx_2.13.2
##
## loaded via a namespace (and not attached):
## [1] Rcpp_1.0.1      lattice_0.20-38 digest_0.6.18  MASS_7.3-51.1
## [5] grid_3.6.0     magrittr_1.5   evaluate_0.13  stringi_1.4.3
## [9] Matrix_1.2-17  rmarkdown_1.12 tools_3.6.0    stringr_1.4.0
## [13] xfun_0.7       yaml_2.2.0     parallel_3.6.0 compiler_3.6.0
## [17] htmltools_0.3.6 knitr_1.22
```