In the current version, factor correlation is positive, all loadings are positive.

For an arbitrary indicator, e.g., X1 in Model B in Figure 1,

X1=ML\*F1+CL\*F2+E1

Var(X1) = var(ML\*F1+CL\*F2) + var(E1)

We want to impose the constraint: var(l1\*F1+l2\*F2)<=1, so that var(E1) is at its lowest 0. Factor variances are set to 1, so we have:

var(ML\*F1+CL\*F2) = ML2 + CL2 + 2\* φ\* ML \*CL <= 1

CL2 + 2\* φ\* ML \*CL + ML2 <= 1

CL2 + 2\* φ\* ML \*CL + φ2\* ML2 - φ2\* ML2 +ML2 <= 1

(CL + φ\* ML)2 + ML2(1- φ2) <= 1

(CL + φ\* ML)2 <= 1 - ML2(1- φ2)

Taking square-root, we have that it must be that:

- sqrt(1 - ML2(1- φ2)) <= CL + φ\* ML <= sqrt(1 - ML2(1- φ2))

Because currently CL and ML are positive right side is what is currently implemented in the Shiny App for **the average crossloading (CL):**

max = sqrt(1-(input$aveloading2)^2+(input$fcor2\*input$aveloading2)^2))-input$fcor2\*input$aveloading2

However, if we allow CL < 0 and/or phi<0, we should implement the minimum as the left side of the equation as well:

- sqrt(1 - ML2(1- φ2)) <= CL + φ\* ML

- sqrt(1 - ML2(1- φ2)) - φ\* ML <= CL

In the output$slidemax\_cross2, min=0 has been replaced with:

min = -sqrt(1-(input$aveloading2)^2+(input$fcor2\*input$aveloading2)^2))-input$fcor2\*input$aveloading2

The only change from the max value is the minus sign. As well, because with negative factor correlation, the max can exceed zero, the Shiny app now sets max= min(1,(sqrt(1-(input$aveloading2)^2+(input$fcor2\*input$aveloading2)^2)-input$fcor2\*input$aveloading2)), so that it is not permitted to exceed 1.

**The range (CR)** is currently set in the Shiny App to be:

min = 0,

max = round(min(1,2\*input$aveloading\_cross2,

2\*(sqrt(1-(input$aveloading2)^2+(input$fcor2\*input$aveloading2)^2)-input$fcor2\*input$aveloading2-input$aveloading\_cross2))

value = min(0,input$input$aveloading\_cross2, (1-input$input$aveloading\_cross2))

The **maximum** value is **smallest** of the following: 1, 2\*CL, 2\*(sqrt[1 - ML2(1- φ2)] - φ\* ML – CL)

In other words, the maximum range is either double the average crossloading if it’s permitted, or double the difference between the average crossloading and the maximum crossloading (also used in the prior slider), because sqrt[1 - ML2(1- φ2)] - φ\* ML – CL = Max\_CL – CL.

For example, if CL=.18 with the slider from 0 to .59, the crossloadings range options are 0-.36. This is because we choose between .18\*2 = .36, and 2\*(.5-.18) = 2\*.32 = .64

If CL=.45 with the slider from 0 to .59, the crossloadings range are 0-.28. This is because we choose between .45\*2 = .90, and 2\*(.59-.45) = 2\*.14 = .28.

How to generalize this to **negative cross-loadings**? The minimum is still zero, as range is always positive.

In the **maximum**, we change the first value from 1 to 2 (because range can double due to negative crossloadings being permitted).

The first restriction, CL\*2, no longer is relevant, because loadings can be negative. Instead, we have to add 2\*abs(Min\_CL - CL), where Min\_CL is the new minimum from output$slidemax\_cross2 defined above. When Min-CL is zero, this automatically reduces to 2\*CL. Otherwise, it’s wider.

Therefore, the **maximum** value is **smallest** of the following: 2 (double because this is the true maximum when loadings can be negative), 2\*abs(-sqrt[1 - ML2(1- φ2)] - φ\* ML – CL), 2\*(sqrt[1 - ML2(1- φ2)] - φ\* ML – CL)

For example, if CL=.18 with the slider from Min\_CL=-.87 (you get this from defaults for main loadings) to Max\_CL=.59, the CR is 0 to .82. This is because we choose the smallest absolute value of the following values: 2, 2\*(-.87 - .18) =- 2.1, and 2\*(.59-.18) = .82.

For another example, if CL=-.6 with the slider from Min\_CL=-.87 (you get this from defaults for main loadings) to Max\_CL=.59, the crossloadings range options are 0 to .82. This is because we choose the smallest absolute value of the following values: 2, 2\*(-.87 + .6) =- .54, and 2\*(.59 + .6) =2.38

Update: Changing the first option in the list from 1 to 2 means that even in the old scenarios where everything is positive, the CR limit is higher. Solution, if CR > CL print something like, **“Warning: You’re allowing negative crossloadings!” Or just put this below that slider.**