Polynomial RET Modeling: Additional Considerations

In this document, I address additional issues beyond those I discussed in the main text for analyzing RET data using SEM-based polynomial regression models. I mainly focus on addressing measurement error through the use of latent variable modeling but I also address other relevant topics including model assumptions, sensitivity analyses, higher order polynomial modeling, curvature detection, and the analysis of binary outcomes.

LATENT VARIABLE QUADRATIC SEM MODELING

There are three approaches to quadratic modeling using latent variables in an SEM context, (1) a double mean centering strategy that was originally advocated by Lin et al. (2010) for latent variable interaction models, (2) a latent moderated structural equations (LMS) method proposed by Klein and Moosbrugger (2000) that has been adopted as a default option in Mplus, and (c) a Bayesian approach for interaction analysis that is tied to the LMS method in the sense that the estimates of the two approaches are asymptotically equivalent (see Asparouhov & Muthén, 2021). I apply the double mean centering strategy in full to an example RET but I only conduct partial RET analyses for the LMS and Bayesian strategies because their application to an RET should be straightforward in light of my coverage of the double mean centering strategy.

Figure 1 presents the basic structure of the RET example. The outcome Y is thought to be determined by three mediators, M1, M2, and M3. Higher scores on all the variables are deemed desirable by program designers. The outcome is modeled as a latent variable with three interchangeable indicators, as is M1. The figure contains a quadratic term which traditionally is conceptualized as a product term that multiplies LM by itself for the target mediator, in this case M1. In the double centering strategy, LM also has three interchangeable indicators (not shown) but in the LMS and Bayesian approaches, its latent status is inferred from other model facets. Hence there are no observed indicators for LM in these frameworks. The presence of the quadratic term is thought of not so much as a formal explanatory variable but more as a convenience variable that indicates and allows the LM→LY link to be non-linear and quadratic in form rather than linear. The treatment variable on the left has two conditions, 0 = control and 1 = intervention. Typically, LM and its product term LM*LM are either assumed to be uncorrelated or are allowed to correlate but this is not shown in the figure. I discuss this issue in more depth below.

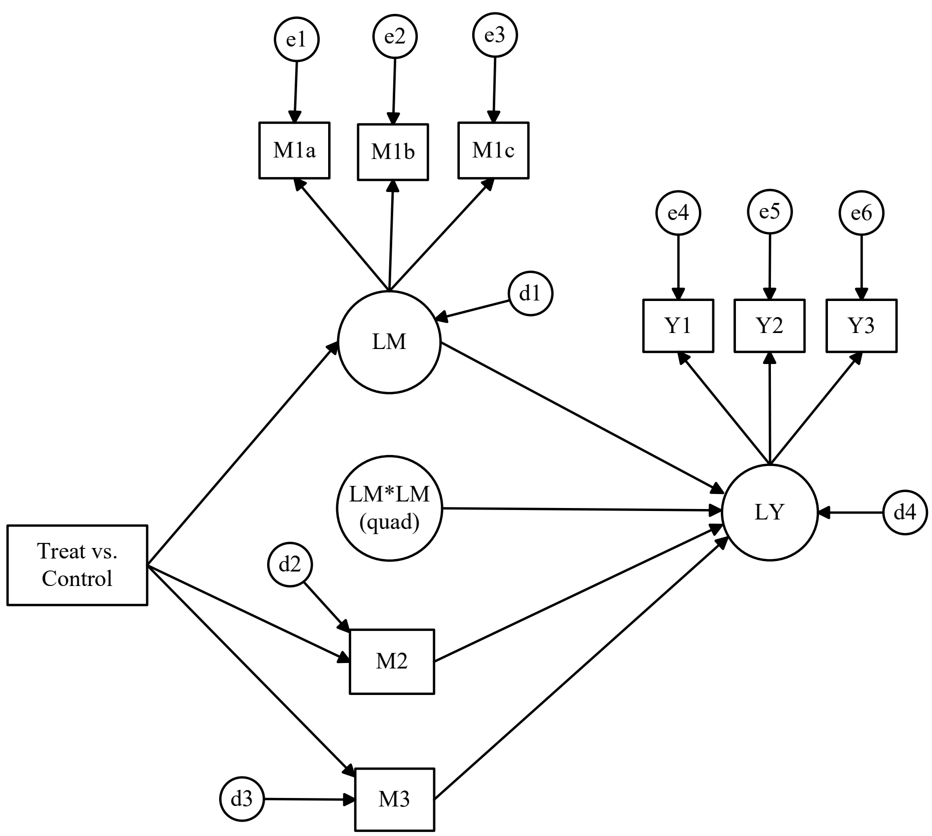


FIGURE 1. RET example

The model does not include a direct effect of the treatment condition on the outcome independent of the mediators because the investigator believes the effect of the intervention on LY are completely captured through the three mediators that the intervention targets, namely M1, M2, and M3. I do not include correlated disturbances between the mediators in the diagram to reduce clutter but I include them in the Mplus program. Omitting them makes the assumption that they are independent of one another net the common cause of the treatment condition. Sometimes such an assumption is viable, sometimes not. In the current case, the research team felt it theoretically appropriate to include the correlated disturbances. All of the variables are continuous except the treatment condition dummy variable. To provide a sense of the metrics of all the variables, they all have standard deviations near 1.0 except, of course, the treatment dummy variable.

Most applications of latent quadratic modeling focus on the case where the latent variables are exogenous. By contrast, I focus on the case where the latent variable takes the form of a mediator and is endogenous by virtue of having causal determinants (the treatment condition) in the system. The strategies one uses in such cases require adaptations of the traditional latent quadratic methods. I adapt them accordingly.

The Double Mean Centering Strategy

The double mean centering strategy was developed by Lin et al. (2010) and is based on the prior work of Marsh, Wen and Hau (2004). Their work primarily focused on the analysis of interaction effects with latent variables based on product terms, but the principles apply readily to quadratic modeling since it also focuses on latent variable product terms, namely a latent variable multiplied by itself. Indeed, one of the earliest articles on latent variable product terms to address interaction effects also formally developed extensions to quadratic modeling (Kenny & Judd, 1984).

The challenge of modeling latent variable product terms with indicators is that there are complex mathematical regularities that need to be respected for different parameters. For example, for the model in Figure 1, suppose that the quadratic latent variable is structured so as to be reflected by three indicators, $M1a^2$, $M1b^2$, and $M1c^2$. In this case, the factor loadings for the product indicators must equal the square of the constituent loadings, e.g., the loading LM*LM→$M1b^2$ must equal the square of the loading LM→ M1b; the loading LM*LM→$M1c^2$ must equal the square of the loading LM→ M1c. Similarly, the variance of the latent quadratic variable is mathematically linked to the variance of the component latent variable used to create the latent quadratic term. When estimating these parameters, programming constraints need to be implemented so as to respect the mathematical relationships (see Kenny & Judd, 1984, for elaboration of the needed constraints; for the case of latent interactions, see Marsh et al., 2004).

The constraints to impose are numerous and rather complex for everyday researchers to navigate. As well, the implied constraints are valid under the somewhat unrealistic assumption of multivariate normality among variables. Marsh, Wen and Hau (2004) developed a technique called the **unconstrained approach** that allows researchers to ignore many of the constraints, that is more user-friendly, and that is relatively robust to the presence of non-normality. The method, among other things, requires researchers to first mean center the indicators of the constituent LM variable (in our example, mean center M1a, M1b, and M1c) and then use these centered variables to form squared indicators of the quadratic latent variable via multiplication ($M1a^2$, $M1b^2$, and $M1c^2$). Lin et al. (2010) extended the Marsh et al. method to expand its ease and applicability by imposing what they called **double mean centering**. This strategy requires that after creating the values of $M1a^2$, $M1b^2$, and $M1c^2$ from the centered M1a, M1b, and M1c terms, researchers then mean center the $M1a^2$, $M1b^2$, and $M1c^2$ indicators before embarking on the final analysis. In short, mean centering occurs twice, hence the term double mean centering. For the underlying logic of their approach, see Lin et al., (2010).

It is best if I explain additional facets of the double mean centering strategy in the context of the Mplus syntax for the method for the model in Figure 1. The relevant syntax appears in Table 1.[1] I will add more syntax to the program using the `MODEL CONSTRAINT` command in Mplus but I do not do so yet because I want to request modification indices on the `OUTPUT` line to help evaluate model fit at the localized level. Mplus does not allow for modification indices in the presence of the `MODEL CONSTRAINT` command. I introduce the `MODEL CONSTRAINT` commands and state their purpose later. Also note that I ignore almost all of the mathematical linkages referred to above that dictate constraints. This is the spirit of this approach; one does not need to concern oneself with them.

Table 1: Mplus Syntax for Double Mean Centering

```
TITLE: DOUBLE MEAN CENTERING PROGRAM ;
DATA: FILE IS quadraticM.dat ;
DEFINE:
  CENTER m1a m1b m1c treat (grandmean) ; !mean center indicators
  pow1 = m1a*m1a ; !create squared indicators after mean centering
  pow2 = m1b*m1b ;
  pow3 = m1c*m1c ;
  CENTER pow1 pow2 pow3 (grandmean) ; !double mean center the above
VARIABLE:
  NAMES ARE id treat x1 m2 m3 x2 x3 m1a m1b m1c pow1 pow2 pow3
```

[1] Double mean centering also is often used for latent variable interaction analysis but it takes a somewhat different form than its use for latent quadratic analyses.

```
11.   y1 y2 y3 x4;
12.   USEVARIABLES ARE y1 y2 y3 m1a m1b m1c m2 m3 pow1 pow2 pow3 treat ;
13.   MISSING ARE ALL (-9999) ; !no missing data
14. ANALYSIS: ESTIMATOR = MLR ;
15. MODEL:
16.   lm BY m1a m1b m1c ;  !specify latent variables
17.   ly BY y1 y2 y3 ;
18.   quad BY pow1 pow2 pow3 ;
19.   m1a m1b m1c PWITH pow1 pow2 pow3 ;  !paired correlated errors
20.   [m1a@0]; [m1b@0]; [m1c@0] ;  !fix measurement intercepts to zero
21.   [pow1@0]; [pow2@0]; [pow3@0] ;
22.   [m2] (am2) ; [m3] (am3) ;
23.   ly ON lm quad m2 m3 (p1-p4) ; !regress ly onto mediators
24.   lm ON treat (p5) ;            !regress mediators onto treat
25.   m2 ON treat (p6) ;
26.   m3 ON treat (p7) ;
27.   lm quad m2 m3 WITH lm quad m2 m3 ;   !correlate these variables no pwith
28.   quad WITH treat ; !correlate quad with treat  ;
29. OUTPUT: SAMP RESIDUAL STAND(STDY) MOD(4) CINTERVAL TECH4 ;
```

Most of the syntax should be familiar. On Line 4, within the `DEFINE` command, I mean center the indicators of LM, namely M1a, M1b, and M1c. I also mean center the treatment dummy variable. Normally I would not do the latter but it is recommended when LM is endogenous. When LM is endogenous all the presumed causes of it that are in the model should be mean centered to improve interpretability of the coefficient for the lower order component part of the product term. If I had included a baseline covariate in the model influencing LM, then I would mean center that covariate by including it in Line 4. The reasons for this are technical but doing so ensures that the mean of LM is 0 which ultimately improves interpretability as illustrated below.

Mplus executes the lines under the `DEFINE` command in sequence. In Lines 5-7, I create the squared indicators for the latent quadratic term (`POW1`, `POW2`, and `POW3`) using the prior mean centered variables from Line 4. Then in Line 8, I mean center `POW1`, `POW2`, and `POW3` to yield the double mean centering. Thus, Lines 4-8 perform the required centering.

On Lines 16 to 18, I define the three latent variables. The Mplus default is to use the first variable listed to the right of `BY` as the reference indicator, so its loading is fixed to 1.0. The loadings for the remaining indicators are estimated. On Line 19, I use the Mplus `PWITH` shortcut command to define three paired correlations. There are two sets of variables for this command, one set on the left side of `PWITH` and the other set on the right side of `PWITH`. Mplus pairs them one-to-one based on their order. So, on Line 19, the error variance for `m1a` is correlated with the error variance for `pow1`, the error variance for `m1b` is correlated with the error variance for `pow2`, and the error variance for `m1c` is correlated with the error variance for `pow3`. I estimate these correlations in recognition of the fact that the indicator error variance for a component part of the quadratic product term is part of the error variance of the square of that indicator for the latent quadratic variable.

Mplus by default fixes the means and/or intercepts of latent variables to zero and estimates the measurement intercepts of the indicators. I allow these defaults to operate for the Y latent variable by not mentioning these parameters in the syntax. Fixing latent variable means and intercepts to zero is fairly standard practice in SEM. Because latent factors are unobserved, they have no intrinsic mean or scale. For the model to be identified, we need to establish scale and location. By fixing the reference indicator to 1.0 as we have done throughout this book, we pass its (error adjusted) metric or scale to the latent variable. By fixing the mean or intercept at zero we define an arbitrary location for the factor but that location generally does not impact model fit because for most models, the focus is on reproducing observed variances and covariances. However, for latent quadratic models the mean structure can matter. When we mean center all of the various indicators by the process of double mean centering, the underlying mathematics are such that meaningful parameter estimates emerge when the factor means and intercepts are fixed at zero in conjunction with the measurement intercepts being fixed at zero (see Lin et al. 2010 for details). Lines 20 and 21 in conjunction with the Mplus defaults accomplish this. As a general practice for the double mean centering approach, you will want to fix the measurement intercepts of the indicators of the latent product term and the indicators of its latent components to zero.

Line 22 estimates the intercept of the two single indicator mediators and assigns them labels for later use in the `MODEL CONSTRAINT` commands. The intercepts for LY and LM in the model are fixed to zero via Mplus defaults because they are latent variables. Finally, lines 23 to 27 define the key linear equations that are of primary interest and assign labels to the coefficients in the equations for later use in the `MODEL CONSTRAINT` commands.

Lines 28 tell Mplus to estimate the correlations between the disturbance terms for the mediators per my prior discussion. Note also they I also allow for a correlation between the latent quadratic term and the disturbances even though these correlations are not of theoretical interest and are ancillary to the model. I need to do so to mimic traditional polynomial regression in which the correlations between the predictors of the outcome (in this case LY) are perfectly captured in order for the mathematical properties we make use of (and that I illustrate below) to hold. I also allow the treatment condition to be correlated with the latent product term on Line 28 because both of the variables are exogenous and because one of them is latent, the Mplus default is to restrict the correlation to zero, which is not reasonable in this case. I need to override that default.

When I executed the syntax, the model fit indices suggested good model fit. The chi square test of perfect model fit in the population was statistically non-significant (chi square = 33.91 with 46 degrees of freedom, p < 0.91), which is consistent with the data being in accord with the model. The RMSEA was <0.001. The upper limit of the 90% confidence interval for it is 0.009. The p value for close fit was not statistically significant (p < 1.00). The CFI is 1.00 and the standardized RMR was 0.012. For localized fit, there were no theoretically meaningful modification indices greater than 4 and no meaningful residuals between the predicted and observed covariances on a cell-by-cell basis.

I now turn to the three questions for an RET, (1) is there a meaningful effect of the intervention on the outcome, (2) is there a meaningful effect of the outcome on the mediators, and (3) is there a meaningful effect of the mediators on the outcome.

Total Effect of the Program on the Outcome

The first substantive question is whether the program meaningfully affected the outcome. In full information SEM (FISEM), one estimates the overall outcome mean difference between the treatment and control groups using a model based approach, namely (a) one assumes the tested causal model is correct, and then (b) one combines the separate model parameter estimates into a model-implied total effect per my discussion of multiplicative rules in Chapters 5 and 7. This approach has challenges when there is non-linearity involved (see Chapter 15 for elaboration of these challenges). A simpler approach is to shift to limited information SEM for purposes of answering this first question by simply comparing outcome means for the treatment and control groups. This is accomplished in Mplus by executing syntax for a latent linear regression that regresses the latent outcome variable onto a dummy variable for the treatment condition (1 = intervention group, 0 = control group). Here is the syntax that accomplished this:

```
TITLE: Total effect analysis
DATA: FILE IS quadraticM.dat ;
VARIABLE:
   NAMES ARE id treat x1 m2 m3 x2 x3 m1a m1b m1c pow1 pow2 pow3
   y1 y2 y3 x4;
   USEVARIABLES ARE y1 y2 y3 treat ;
   MISSING ARE ALL (-9999) ; !no missing data
ANALYSIS:
  ESTIMATOR = MLR ;
MODEL:
ly BY y1 y2 y3 ;
ly ON treat ;
OUTPUT: SAMP RESIDUAL STAND(STDY) MOD(4) CINTERVAL TECH4 ;
```

The model yielded global fit indices and localized fit indices that were consistent with a reasonable fitting model. Here is the output for the treatment effect on the latent Y:

```
MODEL RESULTS

                                                    Two-Tailed
                    Estimate       S.E.  Est./S.E.    P-Value

 LY       ON
    TREAT              0.423      0.067      6.307      0.000
```

The estimated mean difference between the intervention and control groups for the latent Y was 0.42 ±0.13, which was statistically significant (Critical ratio (CR) = 6.31, $p < 0.05$). The analysis used Y1 as the reference indicator, so the difference of 0.423 is on the (error adjusted) metric of Y1. However, the location parameter is fixed to zero and is arbitrary. The mean difference applies to all locations on the LY metric so I can make the difference concrete and more intuitive by noting the mean value of Y1 for the control group, which I calculated external to Mplus using the program on my website called *Grouped means*. The control group mean equaled 7.81. If I add the estimated difference of 0.423 to it, I obtain an estimate of the treatment group mean = 8.23 and use these in my research report, accordingly.

Suppose that discussions with staff and other relevant constituencies established that a meaningful program effect is a mean difference of 0.20 or greater. The confidence interval for the treatment control difference was 0.29 to 0.55. Because the lower limit of the 95% confidence interval for the observed mean difference exceeded this value, I conclude that the program had a meaningful total effect on Y.

Effect of the Program on the Mediators

The second question I address is whether the program meaningfully affects the mediators. With dummy coding of the treatment variable this analysis is straightforward as shown in examples in the main text of Chapter 15 and prior chapters. However, because I mean centered the treatment variable, it no longer has 0-1 coding properties. The coefficient for T→M still equals the estimated mean difference between the intervention and control groups and hence it is meaningful. However, for M2 and M3, the intercept of their equations no longer equals the control group mean and the calculation of the estimated mean for the intervention group is less straightforward. For M1, the matter is further complicated because M1is represented by a latent variable that has a location property fixed at zero. Here is the (edited) output that reports the estimated mean difference between the intervention and control groups for the three mediators:

```
MODEL RESULTS

                                                    Two-Tailed
                    Estimate       S.E.  Est./S.E.    P-Value

LM ON TREAT            0.507      0.069      7.306      0.000
M2 ON TREAT            0.555      0.063      8.787      0.000
M3 ON TREAT            0.494      0.064      7.667      0.000
```

All of the coefficients are statistically significant. The estimated group mean difference for LM is 0.51 ±0.14, for M2 it is 0.55 ±0.13, and for M3 it is 0.49 ±0.13. Suppose prior to the study the researchers set a difference meaningfulness standard of 0.20 for all three mediators. In each case, the lower limit of the confidence interval for the respective mean difference exceeded this standard; we conclude the effects are meaningful.

To obtain the separate estimated group means for M2 and M3, I can use the `MODEL CONSTRAINT` commands. Let P = the proportion of individuals in the intervention group (originally scored 1). In the output for the equation M2 = a2 + b2 $T_{CENTERED}$ I can locate the values for a2 and b2, where $T_{CENTERED}$ is the centered treatment condition dummy variable. I will spare you the derivation but it can be shown in this case that

Mean for the intervention group = a2 + (b2)((1-P)

Mean for the control group = a2 - (b2)((P)

To obtain these estimates, their standard errors and their confidence intervals, I apply the above formulae using the labels for a2 and b2 that I assigned in Table 1 to a newly added `MODEL CONSTRAINT` section to the input syntax after calculating in an external program the value of P:

```
MODEL CONSTRAINT:
NEW ( m2t m2c diff2 m3t m3c diff3 ) ;
m2t = am2 + p6*(1-.529) ;
m2c = am2 - p6*(.529) ;
m3t = am3 + p7*(1-.529) ;
m3c = am3 - p7*(-.529);
diff2 = m2t-m2c ;
diff3 = m3t-m3c ;
```

I include the last two lines as a check on my calculations because their results should equal what I observed in the main Mplus output shown above to examine intervention-control group differences. The above commands are placed just before the `OUTPUT` line.

Before doing so, I add three more lines to obtain intervention and control group mean estimates for LM, the latent representation of M1. Actually, it is not so much the means that I seek (because the location parameter for LM is arbitrary). Instead, I am interested in their estimated standard errors so I can calculate meaningful margins of errors for them later when I alter the location parameter to better reflect the control group and intervention means in the RET. Recall that the intercept in the equation $LM = a1 + b1\ T_{CENTERED}$ is fixed to 0. Using the relevant labels from the original syntax yields the following commands:

```
m1t = 0 + p5*(1-.529) ;
m1c = 0 – p5*(.529) ;
diff1 = m1t-m1c ;
```

and when integrated with the prior commands, I get

```
MODEL CONSTRAINT:
NEW ( lmt lmc diff1 m2t m2c diff2 m3t m3c diff3 ) ;
lmt = 0 + p5*(1-.529) ;
lmc = 0 – p5*(.529) ;
m2t = am2 + p6*(1-.529) ;
m2c = am2 - p6*(.529) ;
m3t = am3 + p7*(1-.529) ;
m3c = am3 - p7*(-.529);
diff1 = lmt-lmc ;
diff2 = m2t-m2c ;
diff3 = m3t-m3c ;
```

After adding these lines to the syntax in Table 1 just before the `OUTPUT` line (and also eliminating the request for modification indices), I re-run the syntax, I obtain the following:

```
MODEL RESULTS

                                                    Two-Tailed
                    Estimate       S.E.  Est./S.E.    P-Value
New/Additional Parameters
    LMT                0.239      0.033      7.306      0.000
    LMC               -0.268      0.037     -7.306      0.000
    DIFF1              0.507      0.069      7.306      0.000
    M2T                0.302      0.043      6.965      0.000
    M2C               -0.253      0.046     -5.530      0.000
    DIFF2              0.555      0.063      8.787      0.000
    M3T                0.257      0.042      6.078      0.000
    M3C               -0.237      0.048     -4.893      0.000
    DIFF3              0.494      0.064      7.667      0.000
```

When I double check the values for the mean differences with the previous entries on the output, they all coincide. I use double the standard errors as an estimate of the margins of error for each mean. For M2, the estimated intervention group mean is 0.30 ±0.09 and the control group mean is -0.25 ±0.09. For M3, the estimated intervention group mean is 0.26 ±0.08 and the control group mean is -0.24 ±0.10. To better map the M1 means onto the (error adjusted) metric of the M1a reference indicator, I set the latent variable mean for the control group to the observed mean for the control group om M1a (which was 8.755) and add to it .507 to it (from `diff1`) to benchmark the intervention group mean, which is 8.755 + 0.507 = 9.262. I use the standard errors for the two groups (0.033 and 0.037) from the output to calculate the margins of errors. Thus, the benchmarked mean for the control group is 8.76 ±0.07 and for the intervention group it is 9.26 ±0.07.

Effect of the Mediators on the Outcome

The final question addresses whether the presumed mediators meaningfully impact the outcome. Here is the relevant output from the initial Mplus run:

```
MODEL RESULTS

                                                        Two-Tailed
                    Estimate       S.E.  Est./S.E.    P-Value

LY       ON

    LM                 0.382      0.030     12.584      0.000
    QUAD              -0.139      0.021     -6.650      0.000
    M2                 0.232      0.030      7.819      0.000
    M3                 0.198      0.029      6.943      0.000
```

For M2, for every one unit that M2 increases, the (error adjusted) mean on the Y outcome on the metric of the Y1 reference indicator is predicted to increase by 0.23 ±0.06 units (CR = 7.82, $p < 0.05$). Suppose prior to the study the research team set the value of 0.15 as the meaningfulness standard for this coefficient. The 95% confidence interval for the M2 coefficient was 0.17 to 0.29. Because the lower limit of the interval exceeds the meaningfulness standard, M2 is said to be meaningfully related to the outcome.

For M3, for every one unit that M3 increases, the (error adjusted) mean on the Y outcome on the metric of the Y1 reference indicator is predicted to increase by 0.20 ±0.06 units (CR = 6.94, $p < 0.05$). Suppose the research team set the value of 0.15 as the meaningfulness standard for this coefficient. The 95% confidence interval for the M3 coefficient was 0.14 to 0.26. Because the meaningfulness standard falls within this interval, I cannot conclude with confidence that the effect is meaningful. I can confidently conclude that the coefficient is non-zero because of its statistical significance but when I take into account sampling error, I cannot unambiguously conclude the effect is meaningful.

The analysis of M1 is more complicated because its effect on the outcome is non-linear as signified by the statistically significant coefficient for the latent quadratic term, hereafter referred to as QUAD. I refer to the component part of this product term as LM and to the latent Y variable as LY. My first goal is to obtain a sense of the nature of the non-linearity between the first mediator and the outcome. I can do so by adding plot commands to Mplus as discussed in the main text of Chapter 15. In a model with double mean centering, the mean of LM is scaled to equal 0. A score of 1 on LM is one unit above this mean on the (error adjusted) metric of M1a; a score of -1 on LM is one unit below this mean on the (error adjusted) metric of M1a; a score of 2 on LM is two units above this mean and a score -2 is two units below this mean. The variance of LM, taken from the `TECHNICAL 4` output is 1.047 and the standard deviation is the square root of 1.047 which equals 1.02. Given this, the bulk of scores on LM should fall between -1.5 and 1.5. Given this, the PLOT commands I use (but without the line numbers) are:

```
1. MODEL CONSTRAINT:
2. LOOP (lmval, -1.5, 1.5, .10);  !LM values to plot on the X axis
3.  PLOT(predly);             !temporary name of Y variable to plot
4.  predly = 0 + lmval*p1 + lmval*lmval*p2 + .041*p3 + .025*p4 ;  !generate y
5. PLOT:
6.  TYPE=PLOT2 ;
```

Line 1 invokes the `MODEL CONSTRAINT` commands. Line 2 generates scores called `lmval` between -1.5 and +1.5 in increments of 0.10 and these scores will be on the X axis of the plot. Line 3 tells Mplus I will generate Y scores with an arbitrary location parameter for LY that will be called `predly`. The `predly` scores are generated by Line 4 using the linear function to the right of the equal sign. The value 0 is the intercept and p1 through p4 are labels I gave to path coefficients in the main syntax for LM→LY, QUAD→LY, M2→LY, and M3→LY, respectively. The right hand side of the equation multiplies the generated `lmva` value by p1, the squared value of `lmva` value by p2, the mean of M2 by p3, and the mean of M3 by p4 and then adds these products together. M2 and M3 are treated as covariates that are held constant at their mean values, 0.041 and 0.025, respectively. Note that these three lines should be considered as part of the `MODEL CONSTRAINT` commands. Line 5 is a separate command line that invokes the `PLOT` features of Mplus to generate a plot and Line 6 tells Mplus to use an internal plot type defined by the number 2. I add these commands to the Table 1 syntax just before the `OUTPUT` line (but after the `MODEL CONSTRAINT` commands if I include those in the same run). After the syntax executes, I click on the PLOT menu option at the top of the Mplus interface and then click on "View plots" and "Loop plots."

Figure 2 presents the plot. The red line is the predicted outcome values and the blue lines reflect the 95% confidence limits about the predicted LY values. The curve shape is such that starting with low values of LM, increases in LM lead to increases in LY but this effect dampens as LM increases and eventually fattens out.

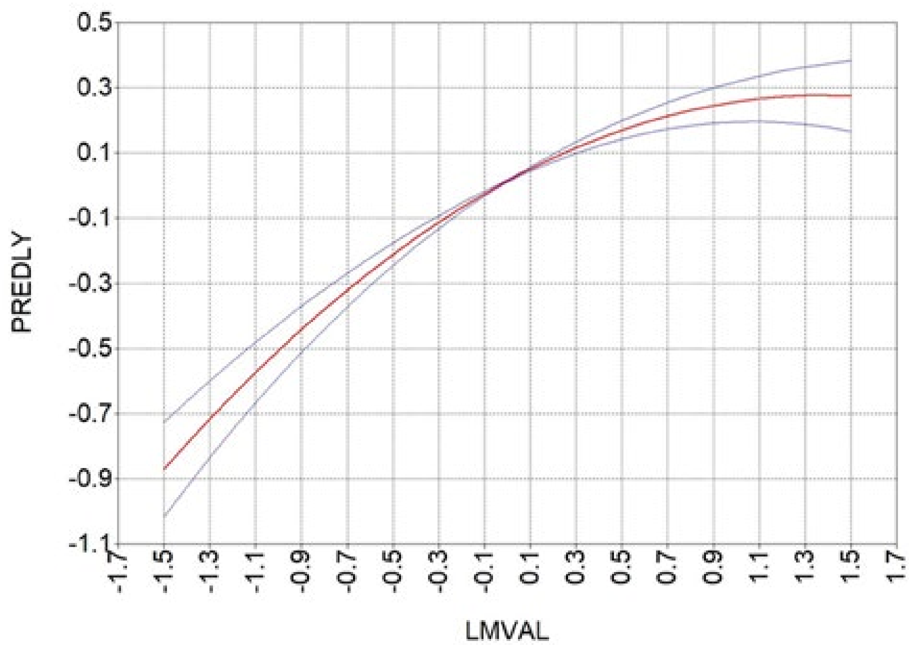


FIGURE 2. Quadratic plot from Mplus

I can quantify the above dynamic using the equation from Chapter 15 that calculates the slope for the instantaneous effect or slope of LM on LY at a given value of LM, which, using the labels from the main syntax of Table 1 is

$$p \text{ at } LM = p_1 + (2)(p_2)(LM)$$

I add the following lines to the `MODEL CONSTRAINT` command that calculate the instantaneous slope at scores of -1.5, -1.0, -0.5, 0, 0.5, 1.0 and 1.5:

```
NEW ( lin1 lin2 lin3 lin4 lin5 lin6 lin7 ) ;
lin1 = p1 + 2*p2*(-1.5) ;
lin2 = p1 + 2*p2*(-1) ;
lin3 = p1 + 2*p2*(-.5) ;
lin4 = p1 + 2*p2*(0) ;
lin5 = p1 + 2*p2*(0.5) ;
lin6 = p1 + 2*p2*(1) ;
lin7 = p1 + 2*p2*(1.5) ;
```

Here is the output for it:

```
MODEL RESULTS

                                                           Two-Tailed
                    Estimate       S.E.  Est./S.E.    P-Value
New/Additional Parameters
    LIN1               0.799      0.077     10.356      0.000
    LIN2               0.660      0.058     11.315      0.000
    LIN3               0.521      0.042     12.545      0.000
    LIN4               0.382      0.030     12.584      0.000
    LIN5               0.243      0.031      7.716      0.000
    LIN6               0.104      0.044      2.363      0.018
    LIN7              -0.035      0.061     -0.572      0.567
```

Note that the estimated effect of LM on Y steadily decreases as the value of LM increases and turns to statistical non-significance for a LM score that is 1.5 units above its mean (slope = -0.035 ±0.12, CR = .57, *ns*). The effect is strongest of the LM scores evaluated when LM is 1.5 units below its mean (slope = -0.799 ±0.15, CR = 10.36, $p < 0.05$). For individual who are relatively high on LM at baseline, the intervention will not be very effective at increasing LY. For individuals who are relatively low on LM at baseline, the intervention likely will be relatively effective at increasing LM. LM moderates the effect of LM on LY!

In Chapter 15, I used average marginal effects to document the overall effect of a predictor that was quadratically related to an outcome. The program on my webpage called *Average marginal effects* does so bit only for single indicator models. that are related quadratically. In Appendix A of Chapter 15 I provide code for calculating average effects in Mplus, but not for latent variables. As a crude approach in the present example, I used the reference indicator for LM (M1a) and the reference indicator for LY (Y1) to calculate the average marginal effect for M1A on Y taking into account the non-linearity. Here is the output:

```
Average marginal effects
 factor    AME     SE       z      p  lower  upper
    m1a 0.2761 0.0346 7.9715 0.0000 0.2082 0.3440
     m2 0.2444 0.0395 6.1948 0.0000 0.1671 0.3218
     m3 0.2026 0.0391 5.1826 0.0000 0.1260 0.2792
```

For every one unit that the first mediator increased the outcome increased 0.28 ±0.07 units taking the non-linearity of M1a→Y1a into account, controlling for M1 and M3. This result was statistically significant (CR = 7.98, $p < 0.05$). Using the meaningfulness standard of 0.15, the lower limit of the effect was greater than the meaningfulness standard suggesting the M1a→Y1a effect is meaningful.

Considered as a whole, the results suggest that all three mediators statistically significantly are associated with the outcome, but only M1 and M2 do so meaningfully. As well, given the non-linear effect of LM on YM, the effect of the M1 mediator on the outcome varies depending on LM at baseline; For individuals who are relatively high on M1 to begin with, increases in M1 likely will have little effect on the outcome.

Parenthetically, although I presented the different `MODEL CONSTRAINT` commands in segments, they all can be included in a single run. Here is the complete code I would use after the initial analysis that permits the generation of modification indices:

```
TITLE: DOUBLE MEAN CENTERING PROGRAM ;
DATA: FILE IS quadraticM.dat ;
DEFINE:
  CENTER m1a m1b m1c treat (grandmean) ; !mean center indicators
  pow1 = m1a*m1a ; !create squared indicators after mean centering
  pow2 = m1b*m1b ;
  pow3 = m1c*m1c ;
  CENTER pow1 pow2 pow3 (grandmean) ; !double mean center the above
VARIABLE:
  NAMES ARE id treat x1 m2 m3 x2 x3 m1a m1b m1c pow1 pow2 pow3
  y1 y2 y3 x4;
  USEVARIABLES ARE y1 y2 y3 m1a m1b m1c m2 m3 pow1 pow2 pow3 treat ;
  MISSING ARE ALL (-9999) ;
ANALYSIS: ESTIMATOR = MLR ;
MODEL:
  lm BY m1a m1b m1c ;  !specify latent variables
  ly BY y1 y2 y3 ;
  quad BY pow1 pow2 pow3 ;
  m1a m1b m1c PWITH pow1 pow2 pow3 ;  !paired correlated errors
  [m1a@0]; [m1b@0]; [m1c@0] ;  !fix measurement intercepts to zero
  [pow1@0]; [pow2@0]; [pow3@0] ;
  [m2] (am2) ; [m3] (am3) ;
  ly ON lm quad m2 m3 (p1-p4) ; !regress ly onto mediators
  lm ON treat (p5) ;            !regress mediators onto treat
  m2 ON treat (p6) ;
  m3 ON treat (p7) ;
  lm quad m2 m3 WITH lm quad m2 m3 ;   !correlate these variables no pwith
  quad WITH treat ; !correlate quad with treat  ;
MODEL CONSTRAINT:
  NEW (lmt lmc diff1 m2t m2c diff2 m3t m3c diff3 lin1 lin2 lin3
  lin4 lin5 lin6 lin7) ;
  lmt = 0 + p5*(1-.529) ;
  lmc = 0 - p5*(.529);
  m2t = am2 + p6*(1-.529) ;   !.529 is prop of people in intervention grp
  m2c = am2 - p6*(.529) ;
  m3t = am3 + p7*(1-.529) ;
  m3c = am3 - p7*(.529);
  diff1 = lmt-lmc ;
```

```
diff2 = m2t-m2c ;
diff3 = m3t-m3c ;
lin1 = p1 + 2*p2*(-1.5) ;
lin2 = p1 + 2*p2*(-1) ;
lin3 = p1 + 2*p2*(-.5) ;
lin4 = p1 + 2*p2*(0) ;
lin5 = p1 + 2*p2*(0.5) ;
lin6 = p1 + 2*p2*(1) ;
lin7 = p1 + 2*p2*(1.5) ;
LOOP (lmval, -1.5, 1.5, .10);  !variable to plot on x axis
PLOT(predly);
predly = 0 + lmval*p1 + lmval*lmval*p2 + .041*p3 + .025*p4 ;
PLOT:
TYPE=PLOT2 ;
```

The LMS Method with Maximum Likelihood

An alternative to the double mean centered strategy for quadratic analysis is the latent moderated structural equations (LMS) approach proposed by Klein and Moosbrugger (2000). LMS was originally intended for latent variable interaction analysis but it also is applicable to latent quadratic modeling. Note that LMS can only be invoked in Mplus for quadratic analysis for latent variables, although I show later how to "trick" Mplus into analyzing single indicator terms as well. A feature of the LMS framework that sets it apart from the double mean centering approach is that it does not require or work with product indicators of the latent quadratic term. Instead, the logic of LMS is based on "modeling the pattern of non-normality" in the data asking what the strength of the quadratic effect, if it exists at all, would have to be to generate the non-normal patterns observed in the data.

Here is the core logic: When you multiply a normally distributed variable by itself, the product term is not normally distributed. This means that although the latent variable LM may be normally distributed, the product of LM times itself will not be. The LMS approach works with the expected form of the non-normality that results from multiplying a normally distributed latent variable by itself and how this, in turn, translates into the distribution of LY and its indicators. LMS conceptualizes the distribution of the observed indicators of the latent outcome as a mixture of normal distributions dictated by the quadratic influence of LM on LY. In doing so, it partitions LM into a set of points or nodes representing both linear and non-linear effects of LM on LY via Gaussian quadrature and then computes the likelihood of the model across these points. Classic EM (expectation-maximization) algorithms are invoked to maximize the likelihood of the mixture distribution. The mechanics of LMS assume that LM is normally distributed and that its indicators are multivariately normally distributed. It further assumes that any deviation from normality in the endogenous LY is due solely to the quadratic effect of LM on LY which, in turn, translates into non-normality of the indicators of LY. Given these assumptions and the decomposition process used by LMS, product indicators are not needed to model quadratic effects of LM on LY. Instead, inferences are made about the quadratic model based on the observed non-normality in the data given the aforementioned assumptions. For mathematical details, see Klein and Moosbrugger (2000).

In this section, I apply the LMS method as implemented in Mplus to the example described in the prior section on double mean centering. I will not consider all facets of the RET analysis because they follow directly from my prior discussion of the double mean centering method. Rather, I show how to program the LMS method and I highlight comparisons with the double mean centering approach.

Although it is not necessary, some researchers mean center the indicators of the latent LM variable involved in the quadratic relationship in order to avoid estimation difficulties that might otherwise arise. I do not do so here but it is something to consider. With double mean centering, I performed an analysis that allowed for modification indices to evaluate model fit and then a second analysis that used the `MODEL CONSTRAINT` commands that disallow modification indices. In LMS, a single analysis can be performed because modification indices are not generated by the Mplus implementation of LMS. Table 2 presents the Mplus syntax.

Table 2: Mplus Syntax for LMS Method

```
TITLE: LMS QUADRATIC PROGRAM ;
DATA: FILE IS quadraticM.dat ;
DEFINE:
  CENTER treat (grandmean) ;
VARIABLE:
  NAMES ARE id treat x1 m2 m3 x2 x3 m1a m1b m1c pow1 pow2
  pow3 y1 y2 y3 x4;
  USEVARIABLES ARE y1 y2 y3 m1a m1b m1c m2 m3 treat ;
  MISSING ARE ALL (-9999) ;
ANALYSIS:
  ESTIMATOR=MLR ; TYPE=RANDOM ; ALGORITHM=INTEGRATION ;
MODEL:
  lm BY m1a m1b m1c ;
  ly BY y1 y2 y3 ;
  quad | lm XWITH lm ;
  [y1]; [y2]; [y3];
  [m2] (am2) ; [m3] (am3) ;
  ly ON lm quad m2 m3 (p1-p4) ;
  lm ON treat (p5) ;
  m2 ON treat (p6) ;
  m3 ON treat (p7) ;
  lm m2 m3 WITH lm m2 m3 ;
```

```
24. MODEL CONSTRAINT:
25.   NEW (lmt lmc diff1 m2t m2c diff2 m3t m3c diff3 lin1 lin2 lin3
26.   lin4 lin5 lin6 lin7) ;
27.   lmt = 0 + p5*(1-.529) ;
28.   lmc = 0 - p5*(.529);
29.   m2t = am2 + p6*(1-.529) ;   !.529 is prop of people in intervention grp
30.   m2c = am2 - p6*(.529) ;
31.   m3t = am3 + p7*(1-.529) ;
32.   m3c = am3 - p7*(.529);
33.   diff1 = lmt-lmc ;
34.   diff2 = m2t-m2c ;
35.   diff3 = m3t-m3c ;
36.   lin1 = p1 + 2*p2*(-1.5) ;
37.   lin2 = p1 + 2*p2*(-1) ;
38.   lin3 = p1 + 2*p2*(-.5) ;
39.   lin4 = p1 + 2*p2*(0) ;
40.   lin5 = p1 + 2*p2*(0.5) ;
41.   lin6 = p1 + 2*p2*(1) ;
42.   lin7 = p1 + 2*p2*(1.5) ;
43.   LOOP (lmval, -1.5, 1.5, .10);  !variable to plot on x axis
44.    PLOT(predly);
45.    predly = 0 + lmval*p1 + lmval*lmval*p2 + .041*p3 + .025*p4 ;
46. PLOT:
47.   TYPE=PLOT2 ;
48.   OUTPUT: SAMP RESIDUAL STAND(STDY) CINTERVAL TECH4 ;
```

The syntax closely follows that for double mean centering. In fact, the statements contained within the `MODEL CONSTRAINT` commands are identical for the two strategies. There are no product term indicators and no `BY` statement for the quadratic latent variable so that syntax is not omitted. Line 15 specifies the quadratic latent variable using the Mplus `XWITH` command. To the left of the pipe symbol is the name of your choice to assign to the latent quadratic term; in this case I chose `quad`. To the right of the pipe is the `XWITH` command that specifies the name of the latent variable to the left of `XWITH` that is to be multiplied by the name of the latent variable to the right of `XWITH`. Line 11 adds the options `TYPE=RANDOM ; ALGORITHM=INTEGRATION ;` which were not included in the double mean centering program. These new options are necessary for the LMS strategy to be used.

The structure of the output is much the same as that for double mean centering. The results for the various parameter estimates were close in value to those observed with double mean centering. One notable difference between the two programs is the reported fit indices. The double mean centering analysis has all of the traditional fit indices. The LMS analysis at the global level reports only the following statistics:

```
Number of Free Parameters                      32
```

```
Loglikelihood

          H0 Value                        -9615.910
          H0 Scaling Correction Factor       0.9768
            for MLR

Information Criteria

          Akaike (AIC)                    19295.820
          Bayesian (BIC)                  19449.497
          Sample-Size Adjusted BIC        19347.870
            (n* = (n + 2) / 24)
```

With no modification indices nor standardized z tests of cell-by-cell predicted versus observed disparities, there are few fit indices to work with. I used the program on my website called *CovToCor* to convert the predicted (also called *implied*) covariance matrix as reported on the Mplus output to a predicted correlation matrix and to also convert the observed covariance matrix reported on the Mplus to a correlation matrix. The program does both conversion operations in a single execution as shown in the video associated with the program and then differences the two correlation matrices. The resulting difference matrix should be dominated by near-zeros values and is interpretable using correlation units rather than metric-conflated covariance residuals. Here is the correlation difference matrix for the eight endogenous variables as reported by the program:

```
         Y1      Y2      Y3     M1A   M1B
Y1    0.000
Y2   -0.027  0.000
Y3    0.012  0.006  0.000
M1A   0.021 -0.006 -0.015  0.000
M1B   0.029  0.001 -0.016  0.002 0.000
M1C  -0.006  0.001  0.006  0.000 0.000
M2   -0.005  0.019 -0.020 -0.018 0.009
M3   -0.005 -0.015 -0.014 -0.022 0.012

      M1C M2  M3
M1C 0.000
M2  0.011  0
M3  0.010  0  0
```

The correlation differences all seem trivial.

I will not review the Mplus output for the parameter estimates of the LMS analysis because, as noted, it is quite similar to the output for double mean centering which I reviewed above.

Bayesian Analysis using LMS

The LMS approach with maximum likelihood can be computationally intense and requires numerical integration which is a drawback of the analytic strategy. Using the approach with Bayesian estimation avoids these issues. Asparouhov and Muthén (2021) outline the underpinnings and mechanics of the Bayesian strategy and I do not delve into them here. Consult that reference for the details. Chapter 8 describes the basics of Bayesian SEM for use in RETs. I assume here that you are familiar with the material from that chapter. The syntax is identical to the syntax in Table 2 except for two statements. Line 11 should be replaced to read

```
ESTIMATOR = BAYES ; !BITERATIONS=100000 (50000); BCONVERGENCE =.01;
```

This change tells Mplus to conduct a Bayes-based SEM using the Mplus defaults to define the (uninformative) priors. Note the commented out options to the right of the exclamation point. These override Mplus defaults to ensure Mplus works extra hard to find model convergence. Usually these options are not necessary but if you encounter estimation problems, you may want to uncomment the options and try executing the program again. They slow down the analysis considerably so you will need to be patient.

The second change is Line 48 for the program output; It should read as follows:

```
OUTPUT: RESIDUAL STAND(STDY) CINTERVAL(HPD) TECH4 TECH8 ;
```

This line invokes the HPD option for the credible intervals and requests information in `TECH8` about model convergence. In the `TECHNICAL 8` section of the output, we can check the PSR statistic to ensure it is less than 1.05, per Chapter 8. This was the case in the current example. In addition to the plots we obtain for the double mean centering and the LMS maximum likelihood strategies, after executing the Bayes-based program we obtain the following plots from the plot menu in the main Mplus interface:

```
Bayesian posterior parameter distributions
Bayesian posterior parameter trace plots
Bayesian autocorrelation plots
Bayesian prior parameter distributions
```

These plots allow for extensive exploration of the Bayesian prior and posterior distributions and how matters proceeded during the iterative process for defining the posterior distributions.

The results of this analysis were again very close to the other two programs so I will not belabor them here. The model fit information is even sparser than the LMS maximum likelihood analysis because all we are provided in the `MODEL FIT` output section is the number of free parameters in the model. The output does not include an observed correlation matrix among the variables, only a model estimated correlation matrix. This is, in part, because Bayesian estimated correlations are not the same as Pearson correlations. Purists argue that you can't compare them. In my experience, usually the Bayesian and Pearson-based covariance estimates will be close to the Bayesian estimates and this also is true of the correlation estimates, but such is not always the case. I often informally inspect the (Bayesian) predicted and observed (Pearson) correlations to ensure they are at least in the ball park of one another but I do not pursue them in any formal way. Again, they are distinct constructs.

As discussed in Chapter 8, some Bayesian SEM models provide variants of the global fit statistics and often they provide Bayesian based chi square fit statistics. However, such is not the case for the LMS approach. As Asparouhov and Muthén (2021) state "*Model fit evaluation remains a challenge for latent variable interaction models. The challenge lies in constructing an unstructured/unrestricted model that is general enough that a structural model can be compared to it, but is also well identifiable and easy to estimate…Undoubtedly, this topic requires further methodological development.*" (p. 326).

All of the results for the parameter estimates are presented in a format that emphasizes the parameter estimate, the posterior standard deviation (which is analogous to a standard error) and the 95% credible intervals (which are analogous to confidence intervals). As an example, here is output for the link between LY and LM, QUAD, M2 and M3 where I eliminate some columns from the output to conserve space and highlight key results:

```
                            Posterior           95% C.I.
               Estimate       S.D.      Lower 2.5%  Upper 2.5%

LY          ON
    LM            0.383        0.033          0.320        0.450
    QUAD         -0.138        0.022         -0.182       -0.096
    M2            0.232        0.029          0.176        0.291
    M3            0.201        0.028          0.146        0.256
```

None of the credible intervals contain the value of zero suggesting they are "statistically significant" but technically null hypothesis takes on a secondary role, if any, in Bayesian modeling. Here are the comparable results from the prior maximum likelihood analysis:

```
                                                   Two-Tailed
                    Estimate       S.E.  Est./S.E.    P-Value

LY          ON
    LM                 0.382      0.031     12.203      0.000
    QUAD              -0.137      0.020     -6.966      0.000
    M2                 0.232      0.030      7.860      0.000
    M3                 0.200      0.028      7.057      0.000
```

and for the double mean centering analysis they are:

```
                                                  Two-Tailed
                    Estimate       S.E.  Est./S.E.    P-Value

LY        ON
    LM                 0.382      0.030     12.584      0.000
    QUAD              -0.139      0.021     -6.650      0.000
    M2                 0.232      0.030      7.819      0.000
    M3                 0.198      0.029      6.943      0.000
```

All of the results are reasonably close to one another and none of the major conclusions change as a function of the method of analysis.

Concluding Comments on the Three Strategies

Each of the analytic strategies discussed has strengths and weaknesses. When the statistical assumptions of the methods are fully met, the LMS maximum likelihood method tends to have more statistical power and lower standard errors than the double mean centering approach. LMS is easy to implement from within Mplus but it is more limited in terms of the global fit indices it yields, although not to the point that it compromises the approach. Simulation studies tend to favor the LMS method even in the presence of non-normal latent predictors as long as the non-normality is modest. When non-normality is more severe, the double mean centering approached can be preferable given large sample sizes (see Cham et al., 2012; Feng et al. 2020). The Bayesian approach has strengths, especially the ability to incorporate prior distributions into the analysis. However, its lack of model fit indices is a non-trivial weakness.

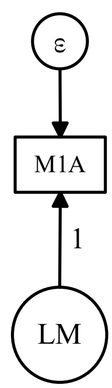
ADJUSTING FOR MEASUREMENT ERROR WITH SINGLE INDICATORS

In many research contexts one does not have multiple indicators of a construct and must use single indicators. When doing so, we implicitly assume a given measure is perfectly reliable or so reliable that the unreliability of it does not matter. On my web page on the

Resources tab for Chapter 3, I provide a document associated with the link *measurement error for single indicator SEM models* that describes strategies for sensitivity analyses to help one appreciate the consequences of measurement error for the data being analyzed. The method used in conjunction with the LMS maximum likelihood strategy can be applied to quadratic modeling. An approach I often find useful when I only have single indicators is to first analyze the data thoroughly using the methods outlined in Chapter 15. I then repeat the analyses introducing *a priori* specified amounts of measurement error in the data to see if my substantive conclusions then change. Theses constitute sensitivity analysis.

I now describe how to set up such analyses in Mplus. I assume you have read or are familiar with the aforementioned document on the *Resources* tab of my web page for Chapter 3. For my example, I use the RET study from the previous sections but do so as if I have only a single indicator of the first mediator, M1, and the outcome, Y, namely the measures M1A and Y1A that were used as reference indicators in the prior examples. Here is the basic logic of the error adjusted approach using the M1A measure.

From a measurement perspective, I can draw an influence diagram for M1A as an index of a latent variable LM, as follows:



The observed measure is assumed to be influence by (a) the unobserved latent construct that M1A is presumed to measure, and (b) random noise in the form of unreliability or measurement error represented by the term ε. The latent variable does not have a natural metric so the factor loading from LM→M1A is fixed to 1.0, per standard SEM protocol. The variance of M1A, which I compute from the data, is 1.283. Suppose the percent of measurement error in M1A is 20% or, stated another way, that M1A has a reliability coefficient of 0.80 and an unreliability coefficient of 1-.80 = .20, per Chapter 3. This means that of the 1.283 units of observed variability in M1A, 20% of it or (1.283)(0.20) = 0.2566 units is error variance. In traditional SEM, we cannot estimate the error variance for single indicator models because attempting to do so creates an under-identified model. We generally need multiple indicators to estimate the error variances. However, as I show shortly, I can fix the error variance for M1A to equal 0.2566 so that the variance of LM will then be adjusted in the analysis to represent the true variance of LM corrected for this particular level of measurement error. I am not estimating the error variance. Instead I am *a priori* fixing it at a given value that I think represents its reliability so as to introduce measurement error into the analysis.

When using single indicator models in Mplus or other SEM software, each indicator is implicitly treated as reflecting a latent variable that has a factor loading of 1.0 for its underlying latent construct and an error variance value of 0, i.e., it assumes perfect reliability. The strategy I use to incorporate measurement error is to explicitly define the measures of M1A and Y1 as single indicators of latent constructs in Mplus. I fix their factor loadings at 1.0 to give their respective latent variables the metric of the "reference indicator" (since there is only one indicator, the single observed measure must be the reference indicator), and then I fix the values of the error variance of M1A to 0.2566 and of Y1 to (1.599)(0.20) = 0.3198, namely the hypothesized error variances. I then apply the LMS method to the data. Note that in theory I could also make measurement error adjustments for M2 and M3, but to keep matters simple, I will not do so. I assume those measures are indeed measured without measurement error. Table 3 presents the syntax I use to adjust for measurement error in M1A and Y1 which is very similar to the syntax in Table 2 but with the changed syntax shown in red.

Table 3: Mplus Syntax for LMS Method

```
TITLE: LMS QUADRATIC PROGRAM WITH MEASUREMEENT ERROR ;
DATA: FILE IS quadraticM.dat ;
DEFINE:
  CENTER treat (grandmean) ;
VARIABLE:
  NAMES ARE id treat x1 m2 m3 x2 x3 m1a m1b m1c pow1 pow2
  pow3 y1 y2 y3 x4;
  USEVARIABLES ARE y1 m1a m2 m3 treat ;
  MISSING ARE ALL (-9999) ;
ANALYSIS:
  ESTIMATOR=MLR ; TYPE=RANDOM ; ALGORITHM=INTEGRATION ;
MODEL:
  lm BY m1a@1 ; m1a@0.2566 ;
  ly BY y1@1 ; y1@0.3198 ;
  quad | lm XWITH lm ;
  [y1];
  [m2] (am2) ; [m3] (am3) ;
  ly ON lm quad m2 m3 (p1-p4) ;
  lm ON treat (p5) ;
  m2 ON treat (p6) ;
  m3 ON treat (p7) ;
```

```
  lm m2 m3 WITH lm m2 m3 ;
MODEL CONSTRAINT:
  NEW (lmt lmc diff1 m2t m2c diff2 m3t m3c diff3 lin1 lin2 lin3
  lin4 lin5 lin6 lin7) ;
  lmt = 0 + p5*(1-.529) ;
  lmc = 0 - p5*(.529);
  m2t = am2 + p6*(1-.529) ;   !.529 is prop of people in intervention grp
  m2c = am2 - p6*(.529) ;
  m3t = am3 + p7*(1-.529) ;
  m3c = am3 - p7*(.529);
  diff1 = lmt-lmc ;
  diff2 = m2t-m2c ;
  diff3 = m3t-m3c ;
  lin1 = p1 + 2*p2*(-1.5) ;
  lin2 = p1 + 2*p2*(-1) ;
  lin3 = p1 + 2*p2*(-.5) ;
  lin4 = p1 + 2*p2*(0) ;
  lin5 = p1 + 2*p2*(0.5) ;
  lin6 = p1 + 2*p2*(1) ;
  lin7 = p1 + 2*p2*(1.5) ;
  LOOP (lmval, -1.5, 1.5, .10);  !variable to plot on x axis
   PLOT(predly);
   predly = 0 + lmval*p1 + lmval*lmval*p2 + .041*p3 + .025*p4 ;
PLOT:
  TYPE=PLOT2 ;
  OUTPUT: SAMP RESIDUAL STAND(STDY) CINTERVAL TECH4 ;
```

Before executing this syntax, I show you the results I obtained for the quadratic portion of the analysis when I analyzed the single indicator data assuming no measurement error per Chapter 15, i.e., I programmed Mplus using M1A, Y1, M2 and M3 as single indicators (but with M1A mean centered) and formed a product term called QUAD for M1A*M1A.

	Estimate	S.E.	Est./S.E.	P-Value
Y1 ON				
M1A	0.276	0.033	8.365	0.000
QUAD	-0.103	0.022	-4.694	0.000
M2	0.244	0.039	6.319	0.000
M3	0.203	0.039	5.149	0.000

The results are somewhat different but reasonably close to the results from the LMS analysis with multiple indicators from the prior sections. The disparities occur because the latter analysis took into account and empirically estimated measurement error and the information from all of the various measurement indicators to derive parameter estimates. To remind you, here is the corresponding table from the multiple indicator LMS maximum likelihood analysis from the prior section:

```
                                                Two-Tailed
                    Estimate       S.E.  Est./S.E.    P-Value
LY          ON
    LM                 0.348      0.042      8.341      0.000
    QUAD              -0.155      0.030     -5.213      0.000
    M2                 0.230      0.039      5.876      0.000
    M3                 0.191      0.040      4.809      0.000
```

Assuming the model is correctly specified, these estimates should be less biased than the single indicator model I ran assuming no measurement error. Here are the results for when I used all single indicators but created the pseudo-latent variables with fixed error variances corresponding M1A and Y1 reliabilities to 0.80 and used LMS per Table 3:

```
                                                Two-Tailed
                    Estimate       S.E.  Est./S.E.    P-Value

LY          ON
    LM                 0.382      0.031     12.203      0.000
    QUAD              -0.137      0.020     -6.966      0.000
    M2                 0.232      0.030      7.860      0.000
    M3                 0.200      0.028      7.057      0.000
```

Note that these values are closer to the multiple indicator analysis where measurement error was estimated and taken into account. This is because I also am taking measurement error into account in the Table 3 analyses but not as elegantly as for the Table 2 analyses with multiple indicators.

Despite the differences in the three analyses, note that the major substantive conclusions I make from the data do not change. To be sure, the magnitude of the effects differ, but the more general characterizations of the results do not.

There are non-trivial issues to consider in the fixed error variance approach to modeling and I describe those in the aforementioned document associated with Chapter 3. My main point here is that Mplus makes it relatively easy to conduct sensitivity analyses for measurement error with the Mplus LMS option.

STATISTICAL ASSUMPTIONS

Regression based quadratic models make the same general assumptions as regression modeling more generally that have been described throughout this book. As well, SEM-based latent quadratic modeling makes the same standard assumptions of SEM modeling.

An assumption that has received considerable attention for latent interaction and quadratic modeling has been the assumption that the latent variable for the component part of the latent quadratic term is normally distributed. This is particularly important for LMS latent quadratic models that do not include product indicators because it is a distribution based approach that infers values for model parameters based on the distributional properties of the data.

Lonati, Rönkkö and Antonakis (2025) have proposed diagnostics to help discern if the assumption of latent variable normality for the component part of a quadratic term is viable. They focus on two diagnostic tools, (a) multivariate normality tests, and (b) a specification test that compares the LMS estimate to an alternative estimator that has the statistical property of consistency under more relaxed assumptions. They refer to the latter test as a Hausman test based on the work of Hausman (1978). In the ensuing discussion and to be consistent with the terminology in prior sections, I refer to the latent component part of the quadratic term as LM, the latent quadratic term itself as QUAD, and the latent outcome as LY.

Tests of indicator multivariate normality in LMS latent quadratic analyses only make sense when applied to the indicators of LM. This is because there are no indicators of QUAD and the indicators of LY are expected to be non-normally distributed because of the quadratic effect of LM on LY. If a test of multivariate non-normality suggests non-normality is present in the indicators of LM, then this could either be because the latent LM variable is not normally distributed or because the error terms of the indicators are not normally distributed. In this sense, indicator multivariate normality tests are ambiguous about the extent to which LM is non-normal. Also, tests of multivariate normality often lack statistical power, rarely provide indices that give one a sense of the nature and degree of non-normality, and often make assumptions in their own right that can be violated and that undermine the tests. Having said that, I provide a program on my website called *Multivariate non-normality* that allows you to choose among three different multivariate normality tests to explore indicator non-normality. The tests are by Mardia (1970), Henze-Zirkler (1990), and Royston (1992). Each test has a reasonable track record statistically, but they also have shortcomings.

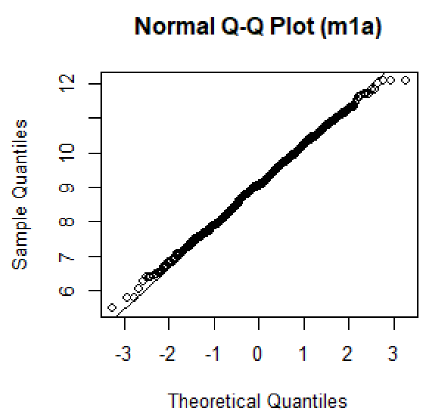
Mardia's test has two versions one that is focused primarily on skewness but it is less optimal for detecting non-normality for symmetric distributions, such as distributions with heavy tails. The other test is for kurtosis which is more focused on peakedness and tail heaviness. The Henze-Zirkler test is sensitive to most forms of non-normality but its results do not flag which aspect of non-normality is present. It tends to have less statistical power than the other two tests. Royston's test is an extension of the classic Shapiro-Wilk univariate test of non-normality and is most sensitive to deviations in the tails of a distribution and in general shape. My preference usually is to consider all of the tests in the spirit of a sensitivity framework and to treat results more as the raising of a red flag than being definitive.

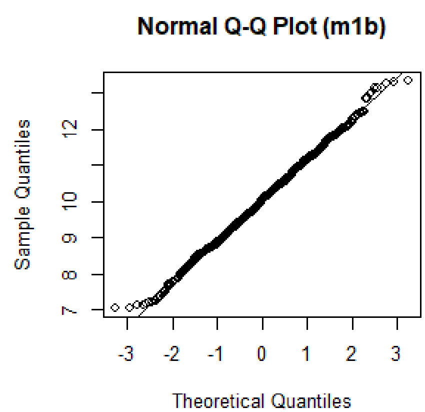
Here are the results of the Mardia test from my program as applied to the LM indicators M1A, M1B and M1C:

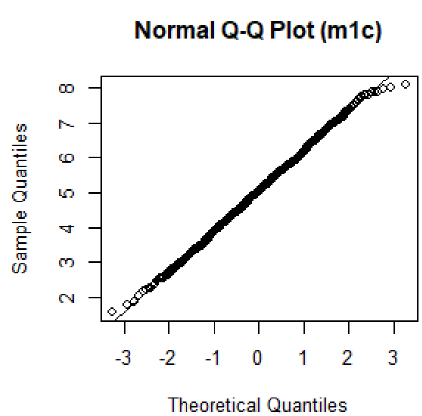
```
Multivariate Normality
             Test          Statistic           p value Result
1 Mardia Skewness     6.3328813344653 0.786563746731078    YES
2 Mardia Kurtosis -0.0326266168620937  0.97397234393959    YES

Univariate Normality
          Test  Variable Statistic   p value Normality
1 Shapiro-Wilk    m1a        0.9983    0.5098    YES
2 Shapiro-Wilk    m1b        0.9978    0.2987    YES
3 Shapiro-Wilk    m1c        0.9983    0.5243    YES
```

Here are the univariate normality Q-Q plots produced by the program for each indicator:







Blatant non-normality is not evident and this also was true for the other normality tests when applied to the data.

A program on my website called *Hausman test for LMS* applies the second test for a non-normal LM recommended by Lonati et al. (2025). This test compares the LMS estimate of the quadratic coefficient associated with QUAD against a second model's estimate of that same coefficient, such as the double mean centering approach or specialized instrumental variable approaches. The alternative model is chosen based on the idea that its estimate has or reasonably approximates the statistical property of being a consistent estimator of an effect (in a strict statistical sense of the term) but with fewer assumptions than the LMS test. For large sample sizes, the difference between two consistent estimators should be small and this is tested using the Hausman strategy. The Hausman test uses a standard hypothesis testing framework to address the null hypothesis that the LMS coefficient is consistent. The test statistic is the ratio of the squared difference between the effect estimates of the two models to the squared standard error of the difference. The significance test takes the form of a chi square statistic with a single degree of freedom. A standardized effect size of estimate difference is a z statistic that is the square root of the chi square value. Lonati et al. (2025) propose a robust version of the Hausman test and this version is implemented in the program on my website.

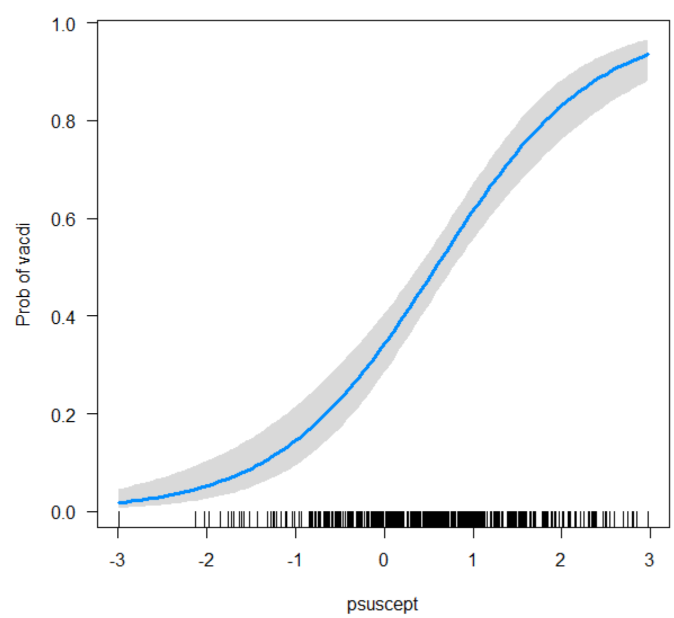
I applied the test to the LMS maximum likelihood coefficient for the prior RET example using the double mean centering model as the alternative model. The Hausman chi square was 0.01 with a p value of 0.92. The square root of the chi square statistics was 0.10. The test suggests reasonable properties on the part of LM but, of course, the quality of the test depends on the quality of the chosen alternative model and reasonably large sample sizes, usually greater than 250 or so.

Taken together, the results are suggestive that the LMS method is viable. As I discuss in Chapter 5, I am reluctant to rely on the strict use of preliminary tests to make analytic decisions but I also advocate exploring one's data prior to formal modeling to gain an appreciation of possible problems. The diagnostics suggested by Lonati et al. (2025) are helpful and I recommend examining them before embarking on LMS-based analyses, if possible. Having said that, the LMS test does have a certain degree of robustness to LM non-normality. Probably the best way to determine if there is a potential problem is to conduct a localized simulation using the collected data discussed in Chapter 28.

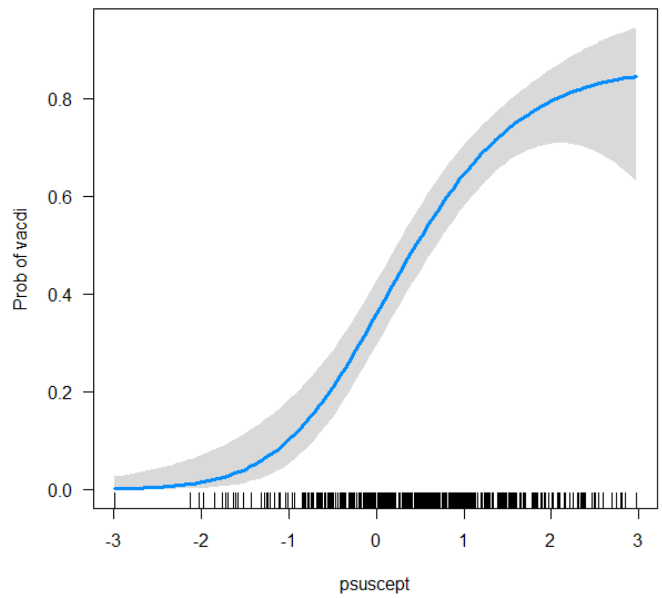
BINARY OUTCOMES

It is possible to include quadratic terms for a predictor in models with binary outcomes. However, sometimes it is difficult to gauge the effects of doing so. For logistic regression, the addition of a squared quadratic predictor to an equation in addition to its component part implies the log odds of the outcome is a non-linear function of the target predictor. I have argued that for most RETs with binary outcomes, we generally are interested in outcome probabilities as opposed to outcome odds or outcome log odds (see Chapters 5 and 12). In that sense, one needs to be cognizant of how the addition of the quadratic term changes the curve between the outcome *probabilities* and the values of the predictor. This can be difficult to intuit because the response curve without the quadratic term is inherently non-linear to begin with, namely it takes the shape of a sigmoid function. By adding a quadratic term, we essentially alter a non-linear curve to take on a different form of non-linearity but it can be difficult to grasp what that new form of non-linearity looks like.

I include a program on my website called *Response plot for a predictor* that allows you to explore the implications for probability curves of adding a quadratic term to a logit or probit model. To illustrate, consider an example where I predict whether individuals obtain a vaccination for a disease (0 = person does not, 1 = person does so) from (a) their perceptions of how susceptible they think they are to contracting the disease if they do not obtain the vaccination (scored from -3 to +3 with higher scores indicating higher perceived susceptibility), (b) the perceived severity of the disease if they contract it (also ranging from -3 to +3 with higher scores indicating greater severity), and (c) a covariate, age. I submitted the data to the program to conduct a traditional logistic regression analysis. I requested the program plot the predicted probabilities for obtaining the vaccination against the perceived susceptibility scores holding age and the perceived severity predictors at their median values. Here is the resulting plot generated by the program:

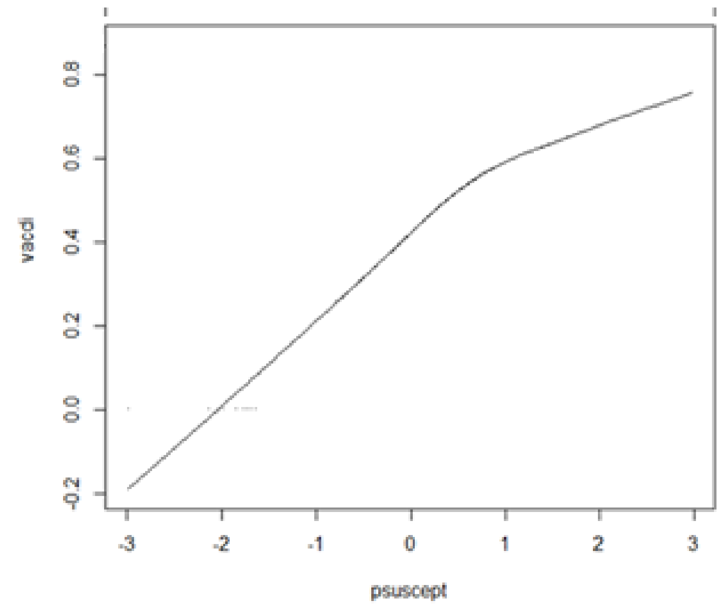


The predicted probabilities are the blue line and the gray shaded area represent 95% confidence limits about the predicted values. The "rug" at the bottom of the graph shows the frequency with which different values on the susceptibility measure occurred; the more dense the rug, the greater the concentration of scores. The sigmoid S-shape is apparent but it is a bit muted of the lower and upper ends of the susceptibility scores. Here is the plot when I added the quadratic term for perceived susceptibility to the equation, which was statistically significant:

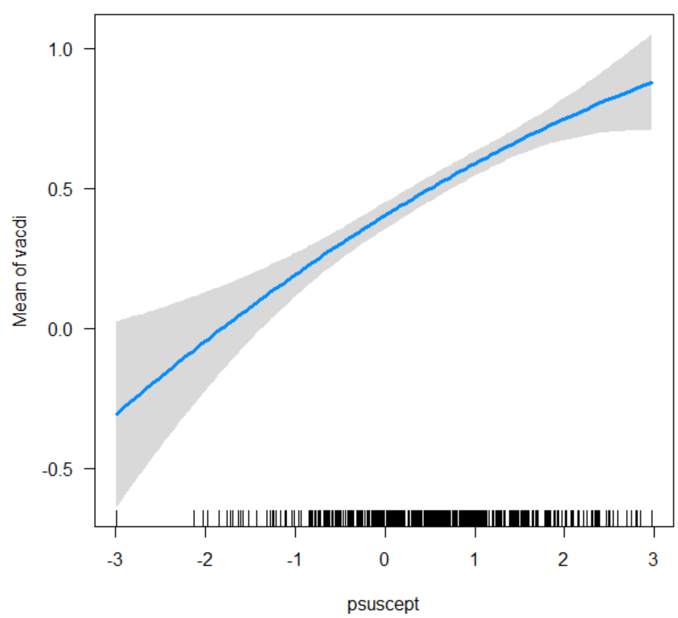


Now, the S shape of the curve has become more pronounced at the lower and upper ends of the curve. This will not always happen but it does in this instance.

The above plot illustrates the best attempt of the fitted logistic model to account for the input data. The question remains, however, if the above curves are data-consistent even though they are associated with statistically significant coefficients for the susceptibility predictor? I can gain a sense of the marginal effect of perceived susceptibility on vaccination probabilities using the *Running interval smoother* program on my website, which includes an option to fit a model-free smooth to a binary outcome and a continuous predictor. Here is the smooth yielded by that program:



Note that there is by and large a linear function but with a slight flattening at around a perceived susceptibility score of 1.0. Here is the plot the *Response plot for a predictor* program generated when I fit a modified linear probability model with a quadratic term:



This curve seems to better approximate the smoother results, but still not optimally. Perhaps, however, it is close enough for all practical purposes. Of course, the logistic model does not focus on marginal effects *per se* and its modeling dynamics are more complex than the modified linear probability model because of the properties I discuss in Chapter 12. The key point to be stressed here is that when you add predictors to modify a non-linear curve that is already non-linear, as is the case for logistic regression, one must be careful about the implied dynamics on a theoretical level. The *Response plot for a predictor* and the *Running interval smoother* programs provide you with exploratory tools in this respect.

MORE COMPLEX POLYNOMIALS

I have emphasized quadratic polynomial modeling in Chapter 15 and in the current document. Of course, one can pursue modeling with other polynomial forms, such as cubic regression, quartic regression, or quintic regression. Although cubic-based latent modeling to adjust for measurement error has been discussed in the literature and there are straightforward ways to incorporate such models into Mplus using LMS, the method for latent-based cubic applications remain relatively unexplored, especially in RETs). It is this difficult for me to recommend using it. In Mplus, to do so requires two `XWITH` commands, the first one specifying `QUAD | LM XWITH LM` followed by a second command `CUBIC| QUAD XWITH LM`. Of course, you can use whatever labels you want to the left of the pipe symbol as long as they are limited to 8 characters. My own inclination is to move to different forms of non-linear analysis discussed in Chapter 15 rather than pursue cubic or higher polynomials using LMS.

One interesting possibility for single indicator models is to use a form of polynomial modeling called **fractional polynomial regression** also known as the **multivariable fractional polynomial (MFP) regression**. It is implemented in the R package called *mfp* and is considered to be an exploratory method that takes into account curvature between predictors and an outcome. The spirit of the approach is to predict Y from M raised to the power lambda, where lambda can take on different *a priori* specified values. The default values that are automatically explored by the program are -2, -1, -0.5, 0, 0.5, 1, 2, 3, which reflect well-known established power transformations. For example, a lambda of 2 is a square transformation (M^2), a value of -1 is an inverse transformation (1/M), a value of 0.5 is a square root transformation ($M^{0.5}$), a value of 3 is a cubed transformation (M^3), a value of zero leads the program to invoke a natural log transformation, and a value of 1 is no transformation. The *mfp* program extends the transformation strategy to multi-predictor scenarios. When using *mfp* each predictor must have non-zero positive values due to the use of logarithms and other powers in the underlying algorithm. The program checks for this property and shifts the data, if necessary, for a given predictor such that its minimum value is positive (see Royston & Sauerbrei, 2008, for details). It then re-adjusts the values in the analysis so that the predicted Y are linked to the original metric of the predictors.

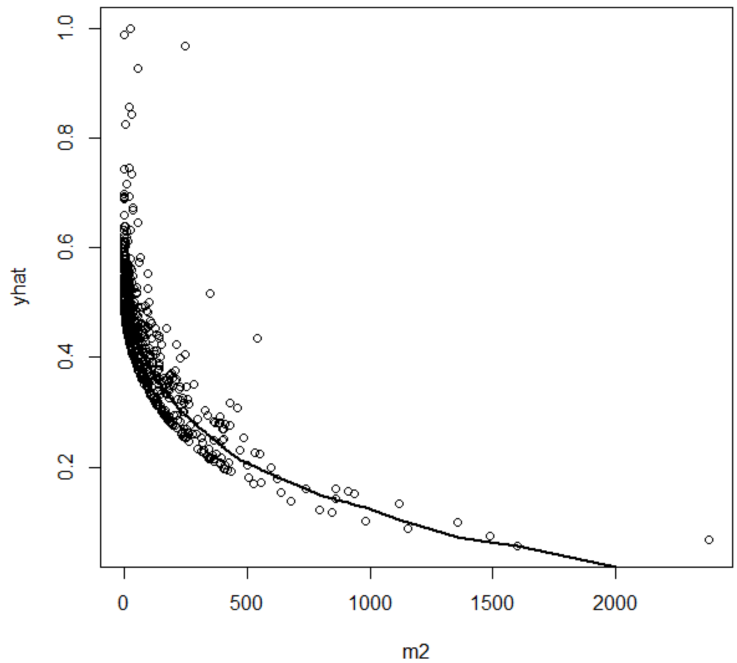
The estimation algorithm in *mfp* processes the predictors sequentially. Initially, the program orders the predictors based on their p values from lowest to highest based on linearity assumptions. The idea is to model relatively important variables before unimportant ones and this portion of the algorithm represents an approximate method to doing so. This step of the analysis also helps to reduce model-fitting difficulties that can otherwise arise. The best fitting fractional polynomial is determined for the most important predictor with all other variables assumed to be linear. The functional form (but not the regression coefficients) for the first predictor is kept and the process is then repeated for the next most important predictor and so on sequentially through the remaining predictors.

The algorithm uses the concept of degrees of freedom for a given predictor to signify the complexity of the functional form allowed for it. The degrees of freedom represent the number of regression parameters or coefficients used to model the curve for a given predictor. The most complex structure has four degrees of freedom in which the curve is modeled using two pairs of power transformations $b_1 M^{p1} + b_2 M^{p2}$, where b_1 and b_2 are different regression coefficients and p1 and p2 are different power terms. This scenario gives the model flexibility to amend the curve to a wide range of shapes, although overfitting can, of course, become an issue. The next most complex structure has two degrees of freedom and allows for more simple non-linear relationships of the form $b_1 M^{p1}$. Finally, when the degrees of freedom equals 1, a linear relationship is assumed. The underlying algorithm uses a closed testing method to determine the optimal degrees of freedom for each variable. For details, see Royston and Sauerbrei (1999, 2008). The final solution is then reported in the form of a regression equation with relevant coefficients.

I provide a program on my website called *Fractional polynomials* that implements *mfp* analyses. The program allows the user to *a priori* impose linear relationships and to include nominal predictors that are not subjected to polynomial transformations. As an example, consider a model where I have a binary outcome, Y, that is predicted from two continuous predictors, M1 and M2, using logistic regression. My interest is in documenting how the probability of Y varies a function of M1 and M2 considered multivariately. As noted, a logistic model inherently assumes an S shaped function when predicting probabilities from a continuous predictor. Here is the core output from my program:

```
Coefficients:
                              Estimate Std. Error z value Pr(>|z|)
(Intercept)                     2.7389     0.8191   3.344 0.000826 ***
I(((m2 + 1)/100)^0.5)          -0.7147     0.1334  -5.359 8.37e-08 ***
I((m1/100)^-2)                 -1.4407     0.4533  -3.179 0.001480 **
I((m1/100)^-2 * log((m1/100)))  -1.1203     0.3461  -3.237 0.001209 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

In R, the letter I stands for *implicit* or *input* and signifies the term inside the parentheses should be evaluated as an algebraic expression. Both predictors yielded statistically significant results. However, M2 is transformed by adding 1 to M2, dividing that result by 100 and then calculating the square root of the second result. The program plots the predicted probabilities of the outcome as a function of M2 ignoring M1 in a marginal sense. The plot has a smooth added to it and is shown below. The resulting curve does not have the expected S shape.



he program also includes a profile analysis option that allows you to compare predicted probabilities for different predictor profiles. For example, the median of M1 is 53. Suppose I compare the profile where M2 = 50 and M1 = 53 to a profile where M2 = 150 and M1=53. This comparison gives me a sense of the implications of a 100 point change in M2 near the lower end of the M2 scale, holding M1 constant at its median value. Here is the output with standard errors and confidence intervals calculated via percentile bootstrapping:

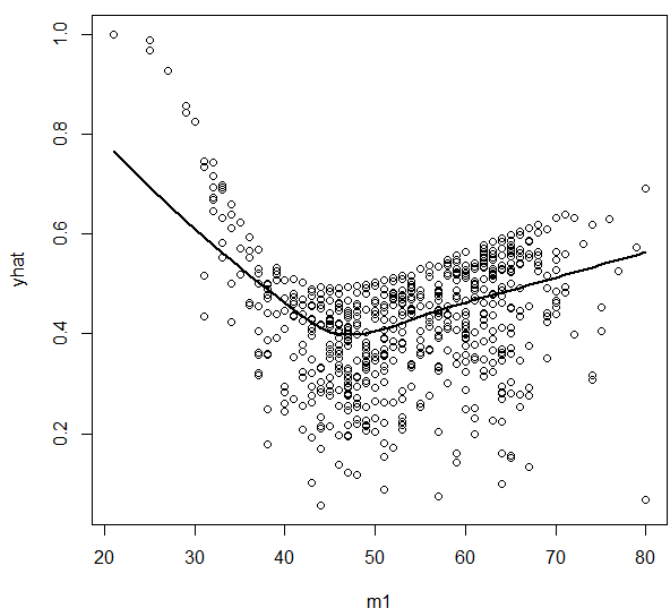
```
PROFILE ANALYSIS FOR PROFILE 1: m1=53,m2=50
                  Results
Probability    0.40892091
Standard Error 0.03822123
Lower CI       0.32705353
Upper CI       0.47018432
```

```
PROFILE ANALYSIS FOR PROFILE 2: m1=53,m2=150
                  Results
Probability    0.32382036
Standard Error 0.03673544
Lower CI       0.26507264
Upper CI       0.40049387

PROFILE 1-2 DIFFERENCE:
                          Results
Profile 1-2 Difference 0.08510055
Standard Error         0.02047349
Lower CI               0.03896771
Upper CI               0.11325418
```

The predicted probability for the first profile is 0.41 and for the second profile it is 0.32. The difference between them is 0.09, which is statistically significant because the 95% confidence interval does not contain the value of zero. I can also elaborate the implications of holding M1 constant at different values when comparing profiles.

For M1, the program divides it by 100 and then an inverse square transformation is imposed. An additional level of complexity is taken by adding a product term between this variable and the natural log of M1/100. Figure 4 presents the program generated plot.



I also can conduct profile analyses that illuminate the effects of M1 on Y holding M2 constant at different values and varying the values of M1.

The *mfp* package has been updated to a new version called `mfp2` to incorporate additional features but it is not widely available in R as of this writing. The above example makes evident the flexibility the model affords. There is, of course, the danger of overfitting but validation analyses and theory can help prevent this. Multivariable factorial polynomials are a useful tool to have in your statistical toolbox. Watch the video for the program on my webpage and consult the book by Royston and Sauerbrei (2008).

CONCLUDING COMMENTS

The current document is a useful supplement to the material on polynomial modeling for RETs. The polynomial method is well suited to limited information SEM as applied to the analysis of the effects of mediators on outcomes. My discussion has not addressed how it can be integrated with analyses of moderation, but I do so later in the book.

REFERENCES

Asparouhov, T. & Muthén , B. (2021). Bayesian estimation of single and multilevel models with latent variable interactions, *Structural Equation Modeling*, 28, 314-328.

Cham, H., West, S., Ma, Y., & Aiken, L. (2012). Estimating latent variable interactions with nonnormal observed data: A comparison of four approaches. *Multivariate Behavioral Research, 47*, 840–876.

Feng, Q., Song, Q., Zhang, L., Zheng, S., & Pan, J. (2020). Integration of moderation and mediation in a latent variable framework: A comparison of estimation approaches for the second-stage moderated mediation model. *Frontiers in Psychology, 11,* Article 2167.

Hausman, J. A. (1978). Specification tests in econometrics. *Econometrica*, 46, 1251–1271

Henze, N., & Zirkler, B. (1990). A class of invariant consistent tests for multivariate normality. *Communications in Statistics—Theory and Methods*, 19, 3595–3617,

Kenny, D. A., & Judd, C. M. (1984). Estimating the nonlinear and interactive effects of latent variables. *Psychological Bulletin*, 96, 201–210.

Klein, A. & Moosbrugger, H. (2000). Maximum likelihood estimation of latent interaction effects with the LMS method. *Psychometrika*, 65, 457–474.

Lin, G., Wen, Z., Marsh, H. & Lin, H. (2010). Structural equation models of latent interactions: Clarification of orthogonalizing and double-mean-centering strategies. *Structural Equation Modeling*, 17, 374—391.

Lonati, S., Rönkkö, M., & Antonakis, J. (2025). Normality assumption in latent interaction models. *Psychological Methods, 30*, 1242–1262.

Mardia, K. (1970). Measures of multivariate skewness and kurtosis with applications. *Biometrika,* 57, 519–530.

Marsh, H., Wen, Z. & Hau, K. (2004). Structural equation models of latent interactions: Evaluation of alternative estimation strategies and indicator construction. *Psychological Methods*, 9, 275–300.

Royston, P. (1992). Approximating the Shapiro-Wilk W-test for nonnormality. *Statistics* *and Computing*, 2, 117–119

Royston, P. & Sauerbrei, A. (1999). The use of fractional polynomials to model continuous risk variables in epidemiology. *International Journal of Epidemiology*, 28, 964-974.

Royston, P., & Sauerbrei, W. (2008). *Multivariable model-building: A pragmatic approach to regression analysis based on fractional polynomials for modelling continuous variables*. Wiley.