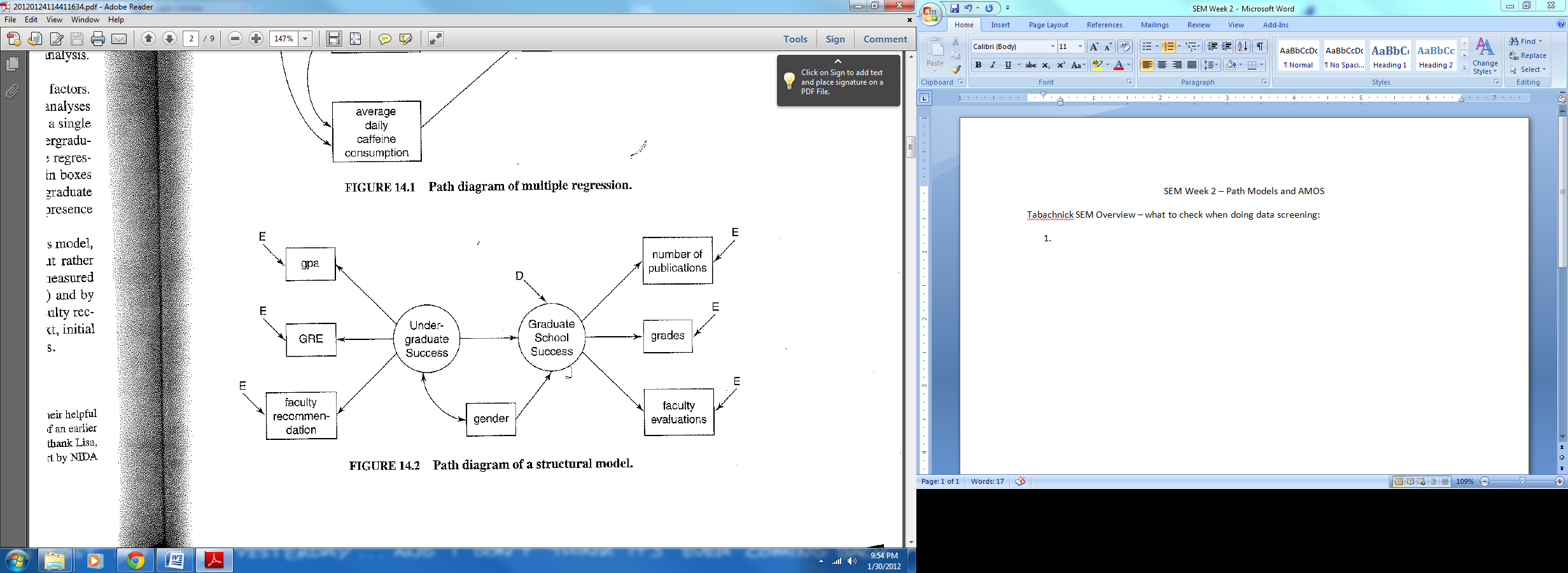
SEM Week 2 – Path Models and AMOS

Tabachnick SEM Overview – what to check when doing data screening:



1. First part explains the basic variables we discussed last week (see week 1 notes about latent and measured variables).
2. Remember:
   1. Measurement model is the relationship between the latents and observed variables
   2. Structural model is relationship between latents
3. Questions you can answer with SEM:
   1. Adequacy – are the fit statistics good? You want the relationship between the covariance matrix you supplied and the one the model calculated to be close.
   2. Testing theory – does the model converge? Or are other paths that you didn’t hypothesize better?
   3. Reliability – do the questions load on the factors (latents) like you expect?
   4. Parameter estimates – you can test certain arrows and see what to expect in the future.
   5. Intervening variables – aka mediators and moderators.
   6. Group differences – are the paths, diagrams and estimates the same for all the subgroups in your data?
   7. Latent growth modeling – looking at changes over time.
   8. Multilevel modeling – use this especially for nested data, meaning that variables are the same (students are in classes that are in schools, etc.).
4. Things to think about:
   1. Sample size – the rule is about 100 – the more the better.
   2. Missing data – you *cannot* have it during SEM, AMOS will flip.
   3. Multivariate normality
      1. You can check this in the data set with a fake regression – all the variables must be tested together.
   4. Multivariate outliers
      1. Check for with Mahalanobis.
   5. Linearity – you can check bivariate scatterplots, but easier to check with your fake regression.
   6. Multicollinearity and Singularity
      1. You do not want correlations between the variables over r >.9 and especially not r>.95. Check with bivariate correlation chart.
   7. Homoscedasticity – you cannot have unequal residuals = megaphone shape on a residuals plot. Check with your fake regression.

PATH analysis with Kline

1. Discussion of correlation versus causation
   1. Often path is used to imply an order, therefore a variable that is X -> Y.
   2. Even if the model fits well, you cannot just say that x causes y because it’s usually still correlational data (some people argue endlessly about this point because it’s the same problem in regression).
      1. X must precede Y in time
      2. X causes Y but not the reverse so much
      3. X still causes Y when you control other stuff
2. Stuff you need
   1. Uh, variables, especially the ones you think are important
      1. Underestimation – you don’t have enough variables
      2. Overestimation – you have too many variables
   2. Variables that reliable, valid for your construct
3. Blocks of path
   1. Direct paths – x 🡪 y
   2. Indirect paths – x 🡪 y 🡪 y
   3. Direct feedback – x 🡪 🡨 x
   4. Indirect feedback – x 🡪x1 🡪x2🡪x
   5. Disturbance correlation – disturbance – unmeasured causes of a variable, for standardized variables
   6. Disturbance covariance – unstandardized variables
4. Directionality
   1. You must specify a direction, can’t use all feedback loops or it’s correlation
   2. If you aren’t sure, you might use a different type of statistics
5. Complexity
   1. Parameters – the number of things you estimate in your model (estimating the population based on your sample)
   2. Observations – not the number of people, but the number of variances and covariances.
      1. Variables ( variables + 1) / 2
   3. You cannot estimate more parameters than you have observations.
   4. The degrees of freedom you will get in the output will be the number of observations – the number of estimated parameters
   5. It will crash if you do this!
6. Constraints
   1. Free parameter – will be estimated by the computer – you allow it to vary
   2. Fixed parameter – set to be equal to a constant
   3. Constrained parameter – estimated by the computer with restriction (multigroup models)
      1. Equality – you are setting two paths to be equal, they can be any number, but have to be equal
      2. Cross group equality constraint – parameter must be equal across groups
   4. Proportionality, inequality, and nonlinear constraints are not used very often.
7. Models
   1. Recursive – effects are unidirectional 🡪
   2. Nonrecursive – effects include feedback loops
   3. I have never seen anyone use the bow-free, bow pattern terms for nonrecursive models
8. Identification
   1. Models are identified if they are theoretically possible (note that some theoretical models may not converge, but could be real…)
   2. Rules
      1. You must have df >= 0 as many observations as free parameters
      2. Latent variables – must be assigned a scale
   3. Underidentified – more parameters than observations
   4. Recursive models are usually identified
   5. Empirical underidentification – caused by multicollinearity since they are the same variable it reduces the number of observations you can estimate
   6. Just identified – equal numbers of parameters and observations
   7. Overidentified – more observations than parameters
9. Sample size
   1. See above – more the merrier.
10. Maximum Likelihood Estimation
    1. The math is set to maximize the likelihood that the data was drawn from the population
    2. Multivariate normality is assumed – so they assume the population is normal when estimating these parameters (so it takes the highest probability value).
    3. Full information method – all estimates are calculated in parallel.
       1. If not – partial or limited information method
    4. Fitting or discrepancy function – similar to least squares (your predict covariance matrix – the real covariance matrix = residual fit).
    5. Iterative estimation – works through several versions until it gets the lowest residuals
    6. Heywood cases – correlations over one or negative variances = bad!
11. Interpreting parameters
    1. Think about them the same way you would interpret B in a regression analysis (coefficients)
       1. As X goes up one point, Y goes up estimate number of points

