Supplement to
***dynamic*: An R Package for Deriving Dynamic Fit Index Cutoffs for Factor Analysis**

This document is accurate as of June 2022. The methods implemented by the `dynamic` package continue to be researched and developed further, so it is possible that this information could change in the future. Updates to this document will be posted on the link above and to www.dynamicfit.app

Table of Contents

Question 1: How does `dynamic` derive cutoffs?

In the main text, we wrote that cutoffs are determined using the same decision rule as Hu and Bentler (1999). Here, we provide more information on what that means so readers have a better understanding from where the numbers in the `dynamic` output are coming.

As a broad overview, the general approach of Hu and Bentler (1999) was to generate data from a particular model and then to fit either (a) the same, correct model or (b) a misspecified, incorrect model. One of their primary motivations was to find a fit index cutoff value that could correctly classify the misspecified models as fitting poorly while also classifying the correct models as fitting well. They did this by generating a few hundred datasets so they could use an entire distribution of fit index values rather than values from a single model.

If replicating their method (i.e., using the same models, conditions, etc.), the result for CFI would be a histogram like the one shown in Figure S1 (this can also be done for other fit indices). The dark grey distribution shows CFI values from models that were known to be correct and the light grey distribution shows CFI values from models that were known to be misspecified.

To determine which value of CFI will serve as a reasonable cutoff, some criteria need to be selected. In Hu and Bentler (1999), a general (but not universal) set of criteria were that at least 95% of misspecified models should be rejected while rejecting no more than 5% of correct models. In Figure S1, the value of CFI that would reject 95% of models from light grey distribution is 0.962 (keeping in mind that higher values of CFI indicate better fit). Similarly, 0% of models from the dark grey distribution would be rejected when using a cutoff of 0.962. Therefore, 0.962 would satisfy these criteria. Indeed, this essentially matches the 0.96 CFI cutoff suggested in Hu and Bentler (1999).

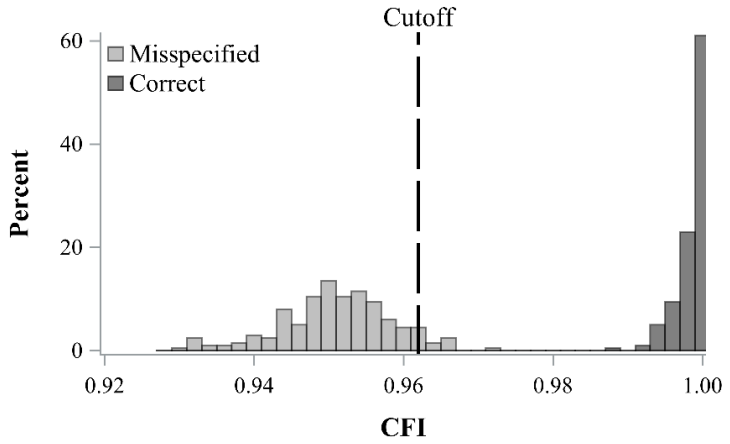

Figure S1. Histogram comparing CFI distributions for models studied in Hu and Bentler (1999). The dark grey distribution represents CFI values when the correct model was fit, the light grey distribution represents CFI values when a misspecified model was fit.

`dynamic` generally follows the same logic as Hu and Bentler (1999) such that two distributions are created: one assuming the fitted model is true and one assuming the fitted model is hypothetically misspecified. A cutoff is then selected based on a value that distinguishes between the two distributions.

The difference between `dynamic` and Hu and Bentler (1999) is the models upon which the distributions are based. Hu and Bentler (1999) consider a small number of models, but their process of selecting a cutoff is model-specific and does not generalize for models with a different number of items, different number of factors, or magnitude of factor loadings (which are all aspects that were not manipulated in Hu and Bentler's study). Consequently, when the model changes, the location and shape of the distributions change. `dynamic` uses simulation to adapt the logic of Hu and Bentler to the user's fitted model to improve the accuracy of fit assessment, essentially answering the question "what would Hu and Bentler's cutoffs have been if they used your model in their simulation"?

Question 2: What do "95/5" and "90/10" mean in the `dynamic` output?

The `dynamic` output shows two rows for each level of misspecification tested. One is labeled "95/5" and the other is labeled "90/10". As mentioned in Question 1, selecting a cutoff is dependent on satisfying some decision criteria. The approach in Hu and Bentler was to select a cutoff such that some minimum percentage of misspecified models were rejected while no more than some maximum percentage of correct models were rejected. The labels for the rows of the output correspond these percentages.

The first number is the minimum acceptable percentage of misspecified models rejected and the second number is the maximum percentage of correct models rejected. So the "95/5" row provides cutoffs that were able to reject at least 95% of misspecified models in the simulation while rejecting no more than 5% of correct models in the simulation. Similarly, the "90/10" row provides cutoffs that were able to reject at least 90% of misspecified models in the simulation while rejecting no more than 10% of correct models in the simulation.

Question 3: What do "--" and "NONE" mean in the `dynamic` output?

The 95/5 criteria are stricter and will reduce classification errors because the acceptable error rates are smaller. Therefore, if there is a cutoff value that meets the 95/5 criteria, this is reported in the output. In such cases, the 90/10 is not needed because a cutoff with lower error rates is available. If this common scenario occurs, users will see "--" in the 90/10 row of the output to indicate that the cutoffs satisfying the 90/10 criteria are suppressed because cutoffs with lower error rates exist and are shown in the 95/5 row. "--" will only appear for the 90/10 row and will never appear in the 95/5 row.

There is no guarantee that a cutoff satisfying particular criteria will exist. In such cases, "NONE" is included in the output to indicate that no cutoff can satisfy the specified criteria. Consider the plot in Figure S2 below. The dark grey distribution represents SRMR values assuming the fitted model were correct, the light grey distribution represents SRMR values assuming the fitted model were misspecified. As noted above, `dynamic` will use these distributions to locate a cutoff that can differentiate between these two distributions with minimal classification errors. It will start by searching for a value of SRMR that would reject 95% of the models from the light grey distribution while rejecting no more than 5% of the models from the dark grey distribution. In Figure S2, there is no such value that can simultaneously satisfy both requirements because the distributions overlap too much – any SRMR value that rejects at least 95% of the models that form the light grey distribution will reject more than 5% of the models forming the dark grey distribution. When the distributions overlap too much, the `dynamic` output will report "NONE" to indicate that no cutoff can satisfy the specified criteria.

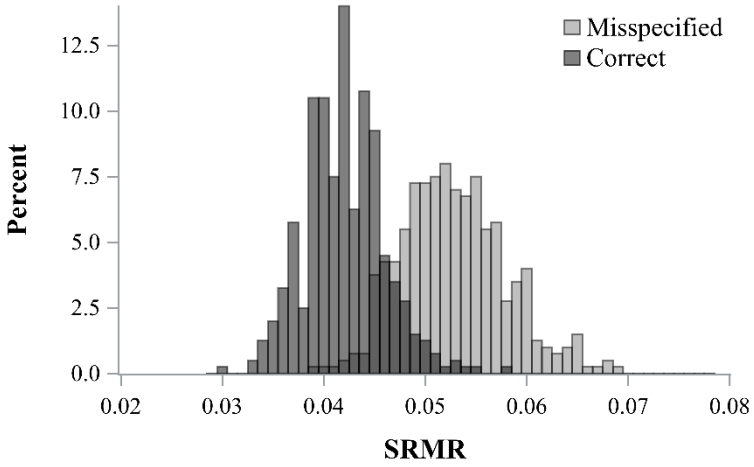

Figure S2. Histogram showing an example where the correct and misspecified distributions have too much overlap to satisfy the 95/5 criteria

"NONE" can appear in the output for either for 95/5 criteria (meaning that no cutoff exist that can reject 95% of misspecified models while rejecting no more than 5% of correct models) or the 90/10 criteria (meaning that no cutoff exist that can reject 90% of misspecified models while rejecting no more than 10% of correct models). If "NONE" appears in the 95/5 row but not in the 90/10 row, users can use the cutoffs in the 90/10 row with the added caution that the error rates will be slightly higher. If "NONE" appears in both the 95/5 and 90/10 rows of the output, the

data may be too noisy to reliability differentiate between correct and misspecified models. If this occurs, the `exactFit` function in `dynamic` may be useful because it will also be able to return cutoffs because it only uses one distribution to determine cutoffs and is not affected by overlap in distributions.

"NONE" results can occur when there is too much sampling variability to reliably distinguish between correct and misspecified models, which is more common when sample sizes are smaller and factor loadings are very weak (see McNeish & Wolf, 2022 for a simulation assessing conditions where this is more likely to occur). It is possible to receive "NONE" for some levels of misspecification but not for other. "NONE" is most common for Level-1 misspecification because the misfit is smallest and distributional overlap is most likely. Simulation results from McNeish and Wolf (2022) show that the reported cutoffs are still safe even if "NONE" appears for some indices but not for others (i.e., if a cutoff exists for RMSEA but not for CFI, the RMSEA cutoff is still trustworthy). This simulation also showed that CFI was most susceptible to "NONE" results and SRMR was least susceptible.

Question 4: How does `dynamic` determine hypothetical misspecifications to test?

In Questions 1-3, we have alluded to the fact that `dynamic` is using a distribution of fit index values from a model with some hypothetical misspecification, so a natural question is "what is the misspecification being tested?".

The complete details and rationales for choices are a little involved and are covered in full in McNeish and Wolf (2021, 2022). As a general overview, `dynamic` starts with the user's fitted model and then considers additional paths that could have been included but were not. Hypothetical misspecifications for one-factor models in the `cfaOne` function are based on adding residual correlations between items. The general rationale for this was that a main concern in many one-factor models is whether the items are measuring a single construct, and residual correlations can test for misspecifications due to local dependence or to incorrect factor structure. Different misspecification levels in the `cfaOne` function are based on the number of residual correlations that are added where higher levels have more residual correlations added. Specifically, the levels tested by `cfaOne` are:

> Level 1: The fitted model plus a 0.30 residual correlation between one-third of the items.
> Level 2: The fitted model plus a 0.30 residual correlation between two-thirds of the items.
> Level 3: The fitted model plus a 0.30 residual correlation between all of the items.

For the `cfaHB` function for multi-factor models, we focus on potentially omitted cross-loadings rather than residual correlations. This was chosen to (a) reflect different concerns with model fit for multi-factor models compared to one-factor models and (b) to maintain consistency with Hu and Bentler (who also used omitted cross-loadings) so that `cfaHB` will return the traditional Hu and Bentler cutoffs if the model from their simulation were assessed with the function. Continuity with Hu and Bentler (1999) was considered to be important to better integrate `dynamic` into the existing landscape of fit index cutoffs rather than having `dynamic` be completely distinct from current practice – we want to emphasize that `dynamic` is not

drastically changes the logic behind current practice, it is simply generalizing the logic of current practice so that cutoffs are more precise and provide more accurate assessments of model fit.

Similar to `cfaOne`, additional levels of misspecification are defined by increasing the number of cross-loadings hypothetically omitted from the model (i.e., Level-1 omits one cross-loading, Level-2 omits two cross-loadings, and so on). The number of misspecification levels tested by `cfaHB` is based on the number of factors rather than being set to a fixed number. The magnitude of the cross-loadings is based on the values standardized cross-loadings in the fitted model rather than fixed at a constant value.

Question 5: What does `dynamic` do "under the hood" after the code is run?

As mentioned in Question 4, applying `dynamic` to a one-factor model will result in four data generation conditions:

1. Data are generated exactly from the fitted model (assuming multivariate normality)

2. Data are generated from the fitted model plus additional 0.30 residual correlation for one-third of the items (Level-1)

3. Data are generated from the fitted model plus additional 0.30 residual correlation for two-thirds of the items (Level-2)

4. Data are generated from the fitted model plus additional 0.30 residual correlation for all items (Level-3)

The fitted model is then applied to all the generated datasets. The fitted model will be correct for the Condition 1 (which will form the dark grey distribution from Figure S1) but the fitted model will be misspecified for Conditions 2 through 4 (which will form three separate light grey distributions). Behind the scenes, `dynamic` will conduct all the simulations and compare the resultant distribution from Condition 1 to resultant distributions Conditions 2-4, one at a time.

For the example in the main text, the equivalent plots are shown on the next page in Figure S3 where the distribution of the fit indices assuming the fitted model were correct (in blue), the distribution of fit indices assuming the fitted model were misspecified to a particular degree (in red), the location of the dynamic cutoff (dashed lined), and the traditional Hu and Bentler (1999) cutoff as a reference point (dotted line). Users can produce these plots by including the `plot=TRUE` option in any of the functions in `dynamic`

The process is the same for a multi-factor model except that the misspecifications in Condition 2 and beyond will be based on potentially omitted cross-loadings. The number of conditions used for multifactor models will also depend on the number of factors in the model rather than being fixed to four for all models as in the one-factor context.

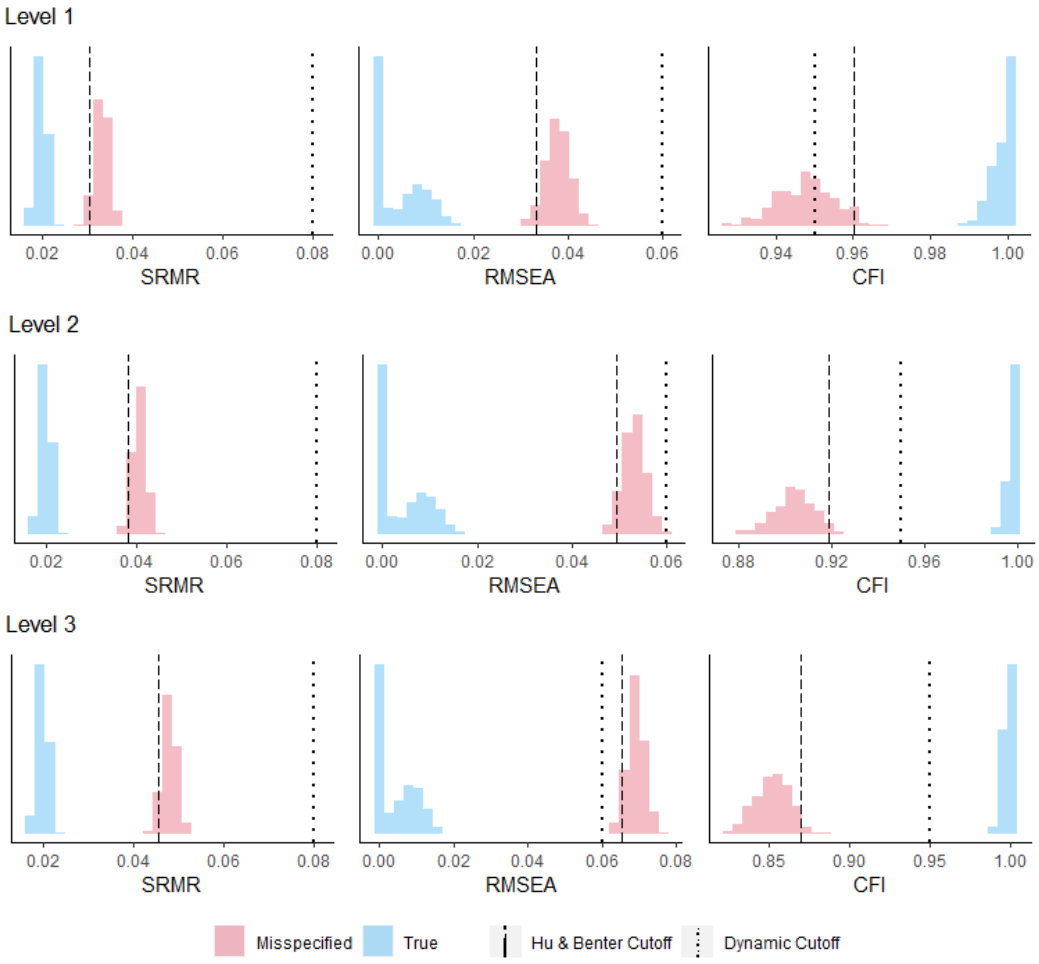

Figure S3. Plots output by `dynamic` showing distributions of fit index cutoffs when the model is correct (in blue) or when the model is hypothetically misspecified (in red) for three different magnitudes of hypothetical misspecification.

Question 6: How do I read the `dynamic` output in R?

The `dynamic` output from the R console summarizes the results of comparing distributions from Question 5. For the example analysis in the main text, the output for the R console is as follows (these were the values that were used to create Table 1 in the main text)

```
Your DFI cutoffs:
                SRMR RMSEA  CFI
Level 1: 95/5  .031  .033  .96
Level 1: 90/10   --    --   --
Level 2: 95/5  .038  .049 .919
Level 2: 90/10   --    --   --
Level 3: 95/5  .046  .065 .869
Level 3: 90/10   --    --   --

Empirical fit indices:
 Chi-Square  df p-value   SRMR   RMSEA    CFI
    969.539 189       0  0.048   0.063  0.868
```

In the output, we see "--" for the 90/10 row and numbers for the 95/5 row for all three levels, meaning that the cutoffs satisfying stricter 95/5 criteria existed for all three levels of misspecification. This follows from the plots in Question 5 because the distributions did not overlap, so the 95/5 criteria could be satisfied easily. The bottom of the output shows the fit indices from the fitted model. To assess fit with the DFI cutoffs, the empirical fit indices from the bottom of the output are compared to the different cutoffs at the top of the output.

Using RMSEA as an example, the empirical value is 0.063. This value is quantifying the degree of misspecification when applying the fitted model to these data, but it is uncertain whether this 0.063 value represents a big or small misspecification given the fitted model's characteristics. The intention of DFI cutoffs is there to determine the size of RMSEA (or other fit indices) for some hypothetical misspecifications with the same model characteristics. The DFI method runs some simulations using the aforementioned hypothetical misspecifications as benchmarks and those values are reported in the table.

To determine how to classify the degree of misspecification, we would want to compare the empirical value to the cutoffs. For lower-is-better indices like SRMR and RMSEA, the empirical value needs to be below the cutoff. The empirical value for RMSEA is above the Level-1 (0.033) and Level-2 (0.049) cutoffs, so the fitted model appears to have more misspecification than the hypothetical misspecification used for those models. The empirical value for RMSEA is slightly below the Level-3 cutoff (0.065), so it appears that the misfit in the fitted model is consistent with omitted 0.30 residual correlations for all the items in the model. For higher-is-better indices like CFI, empirical value would need to be higher than the cutoff to appropriately classify the degree of misfit.

To be clear, this does not mean that the fitted model actually *did* omit residual correlations. The misspecifications tested in Level-1, Level-2, and Level-3 are hypothetical and are used as benchmarks to gauge the sensitivity of fit indices to misfit when the model characteristics look like the fitted model (e.g., same sample size, number of items, magnitude of factor loadings). No part of `dynamic` is trying to actively identify misspecifications that are present in the model.

Instead, the interpretation is that the cumulative misspecification in the fitted model (as captured by the fit indices) is on par with omitting a 0.30 residual correlation for 10 items pairs (i.e., each of the 21 items had a residual correlation with one other item) for these model characteristics. The actual location and nature of the misspecifications in the fitted model may be different, but the different levels help determine the magnitude of these misspecification(s) and how the fit index values behavior for idiosyncratic permutations of model conditions.

Question 7: Why are there separate functions in `dynamic` for one-factor and multi-factor models? Are there major differences between the `cfaOne` and `cfaHB` functions?

The general architecture of the `cfaOne` function for one-factor models and the `cfaHB` function for multifactor models is the same – the format of the output and the decision rules for deriving cutoffs is identical in both functions. In fact, we initially combined them into one function that would internally parse the model statement as a first step to determine how many factors were present in the model. However, during initial testing, it was reported that users were less clear about what misspecifications were being tested when there was one function that made the choice internally compared to two separate functions from which the user selects. Therefore, the two functions exist so that each function only has one process for creating hypothetical misspecifications to help users interpret the results from `dynamic` more easily.

With that said, the two functions differ in two ways. First, as outlined in Question 4, the process for creating hypothetical misspecifications differs by function because different types of misspecification are more or less relevant for different types of models. For instance, omitted cross-loadings may be relevant in multifactor models but can't exist in one-factor models because there is no second factor on which an indicator can cross-loading. Second, the default number of misspecification levels differs for each function. The `cfaOne` function tests 3 levels of misspecification for any one-factor model; the `cfaHB` function bases the number of misspecification levels on the number of factor minus 1.

Question 8: What type of models does `dynamic` support?

As of this writing in June 2022, `dynamic` supports confirmatory factor analysis with continuous indicators only and the simulation component of the software assumes multivariate normality. It takes some time to work out generalizations for other types of models because a general method for identifying relevant misspecifications must be done model by model. For instance, potential misspecification that are relevant to latent growth models would likely be very different than a confirmatory factor analysis because latent growth models are typically interested in aspects like the function form of growth being correct or whether the correlation between repeated measures is reasonable rather than things like omitted cross-loadings that are relevant to confirmatory factor analysis.

Our current work is focused on extending the method to categorical indicators, non-normality, missing data, and measurement invariance; so we expect those or related extensions will be the next to be added to `dynamic`.

Question 9: How long does `dynamic` take to run?

This can depend on several aspects and it is hard to provide a specific estimate. `dynamic` incorporates a simulation-based method, so it generates and analyzes a few thousand datasets. If the sample sizes are large, models are large, or if the estimation is difficult and converges slowly, each individual dataset can take more time and the overall run times will increase accordingly.

The specifications of a user's machine can also affect run times because different processors perform analyses at different speeds, so the run times may not necessarily be the same for the same analysis done on different machines. As a rough guideline, the example analysis in the main text took about 3 minutes to run on the authors' computers and users should expect run times of at least 1 minute using the default settings. If users are deciding between the `dynamic` R package and the www.dynamicfit.app Shiny web application, the R package will be more expedient because it runs locally rather than through a server.

In the `dynamic` R package, users can change the number of replications used in the simulation within all the functions with the `reps` option. The default is 500 replications but researchers with large models or large sample sizes may wish to lower this number to 100 or 200 replications for exploratory purposes to decrease run times if multiple models are being considered. However, we strongly recommend 500 replications when reporting final results to obtain sufficient precision. Also note that the Shiny application uses 500 replications and does not provide an option to change the number of replications, so changing the `reps` option in `dynamic` can lead to different results if users are comparing the results from the R package to the Shiny application.