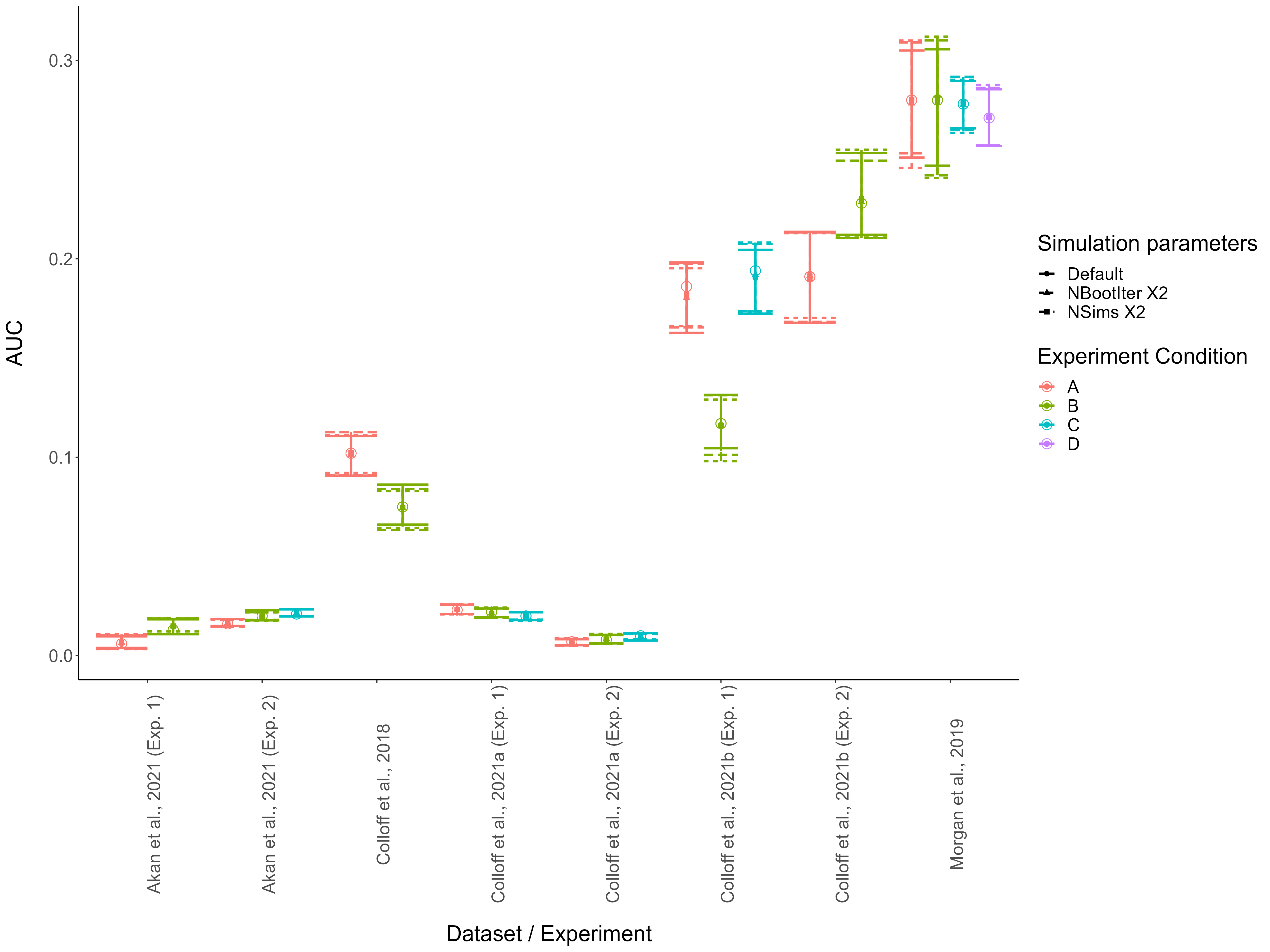
powe(R)OC Testing Results

Eric Mah

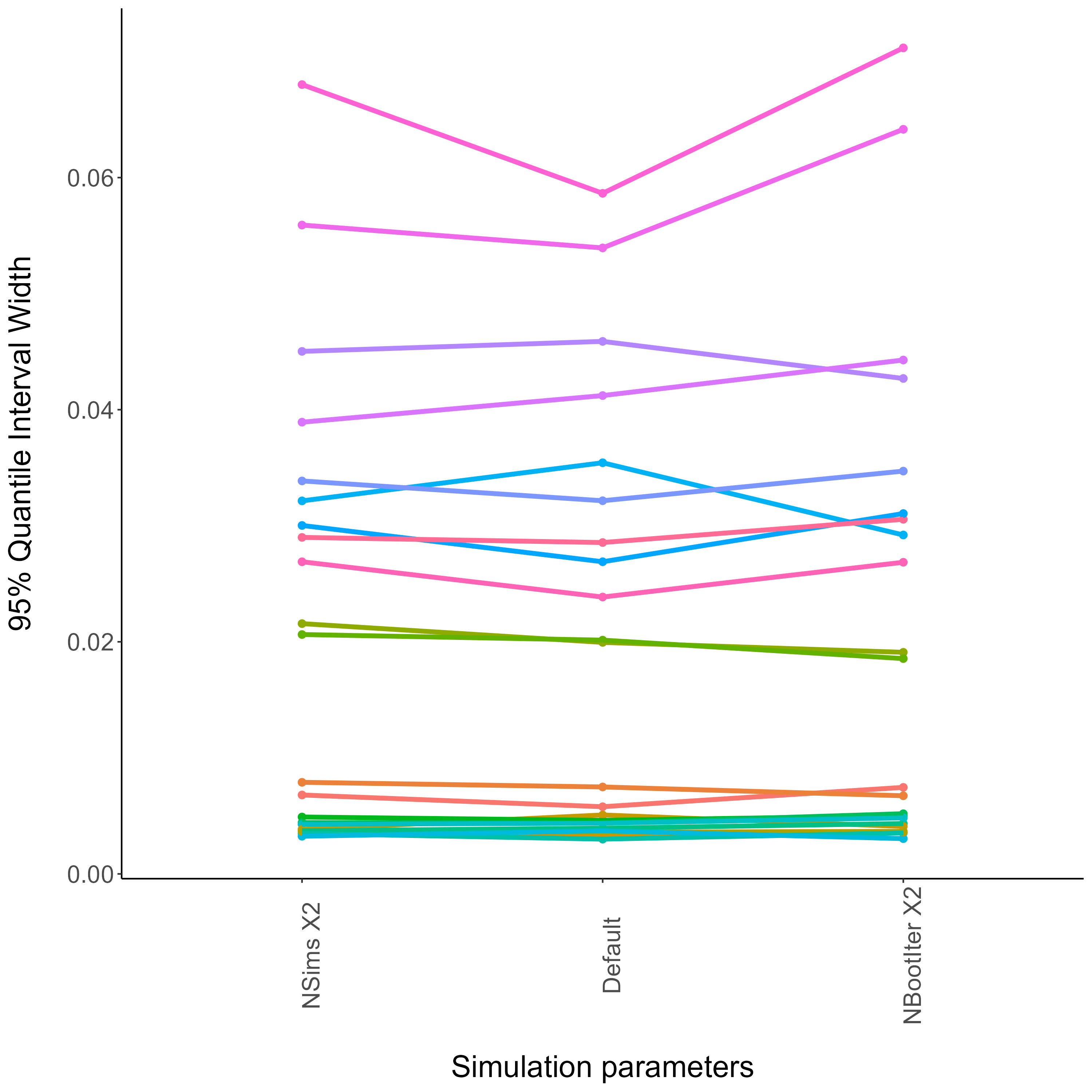
This document details several tests of simulation validity/performance conducted using real datasets. Data/articles used for testing can all be found in <https://github.com/E-Y-M/poweROC/tree/main/Dataset%20testing%20and%20reports>, and were obtained from the Open Science Framework. If you have ROC data (along with analysis parameters) you are willing to share for the purposes of simulation testing, feel free to email me at [ericmah@uvic.ca](mailto:ericmah@uvic.ca). Issues/comments on the app or simulation testing results can be posted on GitHub at <https://github.com/E-Y-M/poweROC/issues>.

## AUC Recovery

At a basic level, simulation validity depends on the ability of the simulations to recover AUC values close to those in the original dataset. The figure below depicts original AUC estimates from various papers with open data (5 papers, 8 experiments, 22 ROC curves computed using the same N’s/pAUC cutoffs in the original papers), along with simulated estimates and intervals: 

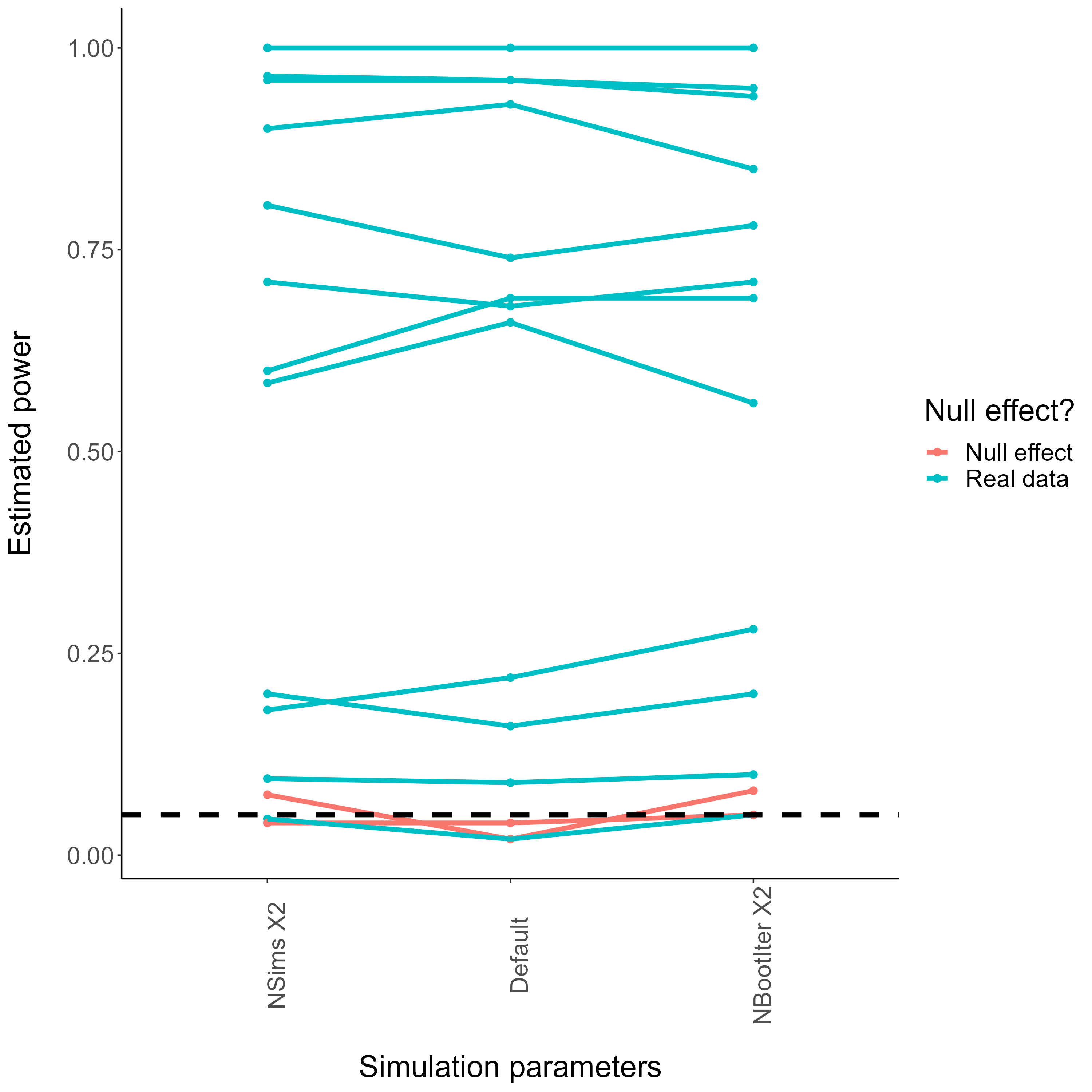
Testing the ability of the simulation to recover AUC values from experiments. Open circles represent original AUC values, all other points represent simulation estimates under various conditions (“NSims” = Number of simulated datasets per effect size/N, “NBootIter” = Number of bootstrap iterations per AUC comparison). Error bars = 95% quantiles on the mean estimated AUC for the simulations. Overall, simulations demonstrate excellent ability to recover original AUC values, even under default settings (NSims = 100, NBootIter = 1000).

## Simulation precision under different conditions

Still, the question remains as to whether increasing the number of simulations or bootstrap iterations increases power. The figure below shows the width of the 95% quantile intervals for the AUC estimates above, as a function of the simulation conditions. 

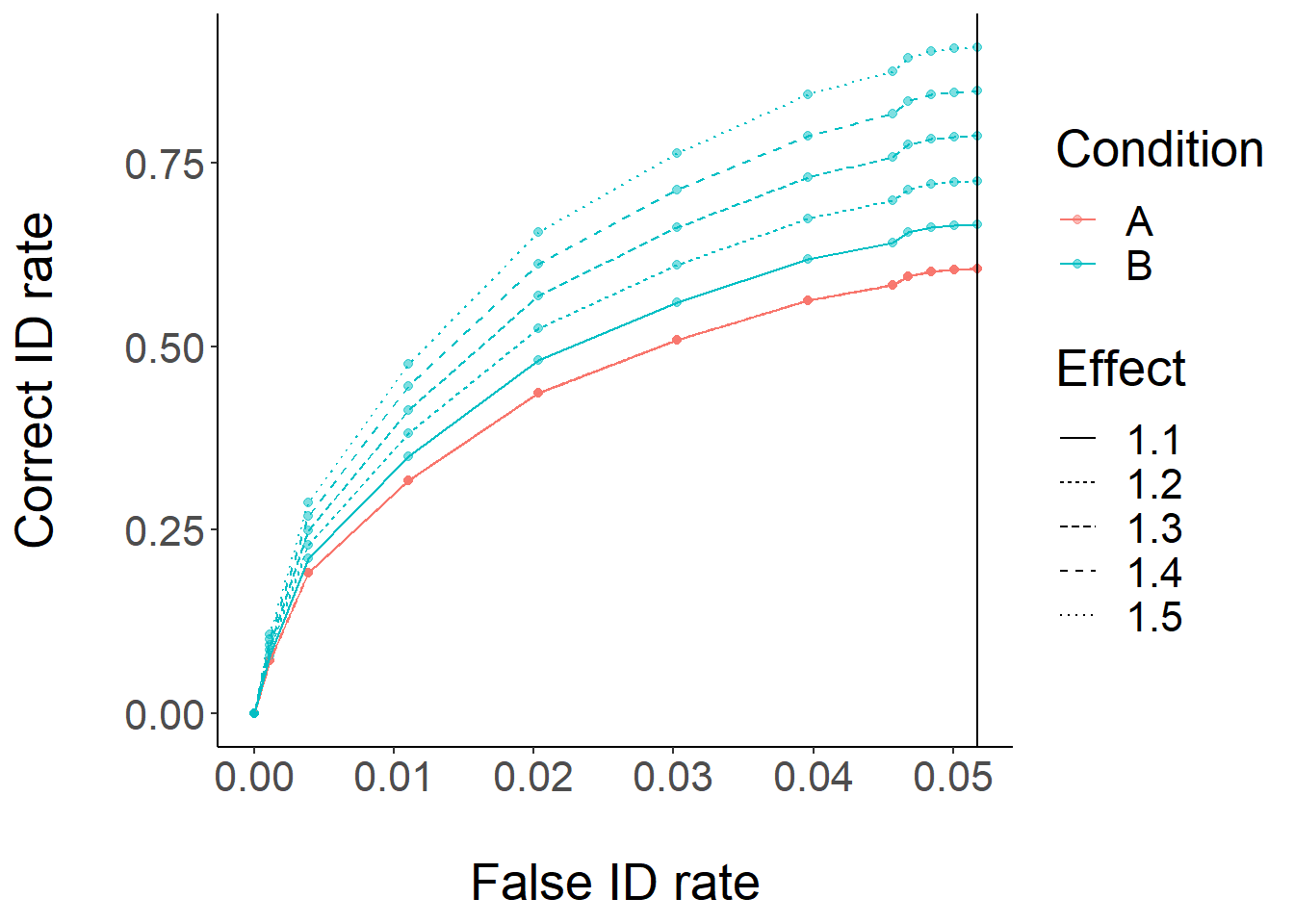
Based on these simulations, it does not seem that increasing the number of simulations or bootstrap iterations necessarily or substantially increases the precision of the simulation beyond the default settings, suggesting that the default settings will result in reasonable estimates.

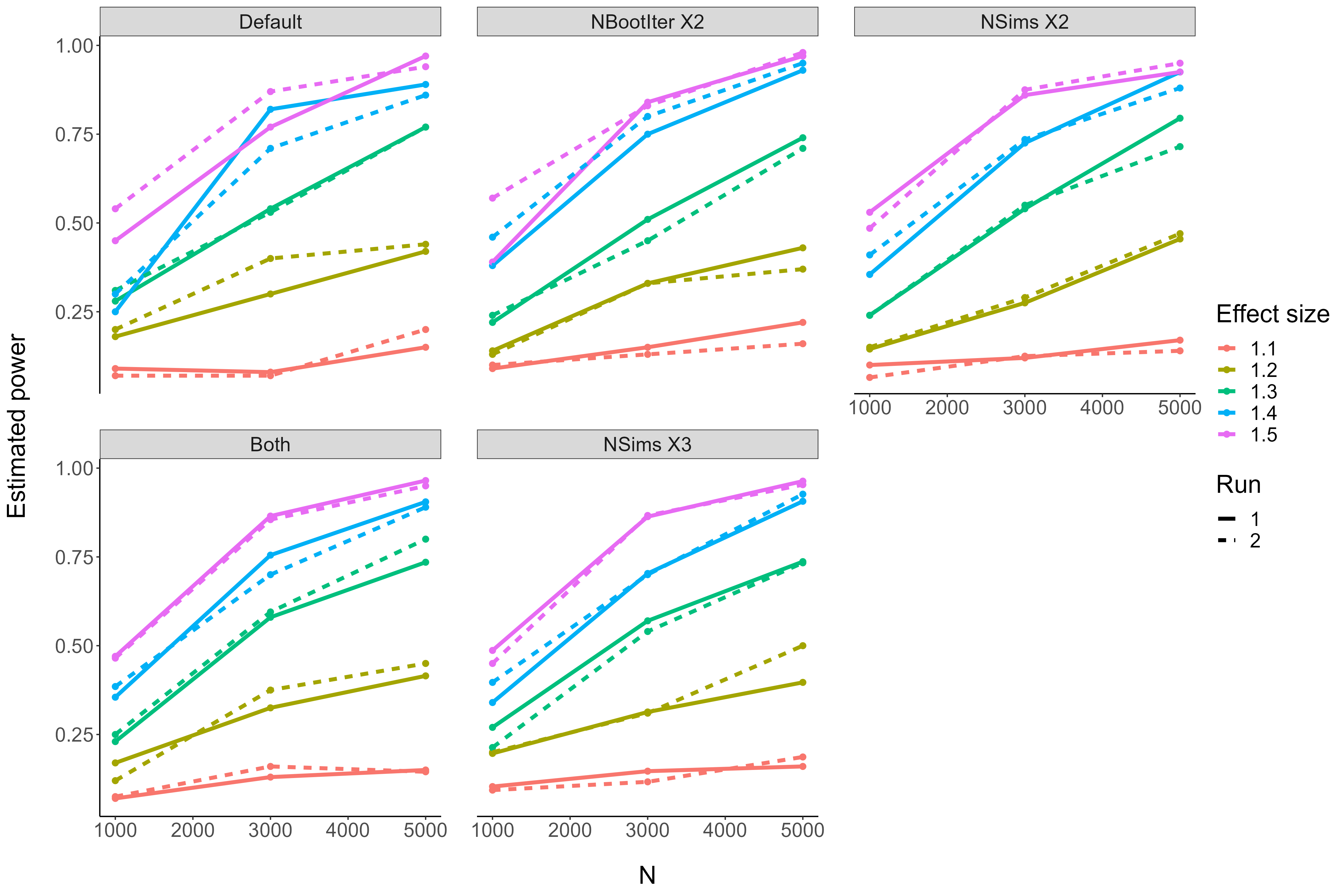
## Power estimates under different conditions

It is not clear whether the default simulation settings result in the most accurate power estimates. I simulated power for 13 ROC comparisons from the papers above. I also conducted two simulation runs of a dataset with a prespecified null effect (using the “Medium Similarity” condition from Colloff et al., 2021 as a base) to compare with the normative Type I Error Rate of .05, all under the three different simulation conditions. These power estimates are plotted below: 

Power estimates differed slightly across the different simulation conditions, but no clear patterns emerged. In these examples, the maximum range of estimated power was .10. Importantly, power estimates in the two null effect simulations were close to the nominal Type I Error Rate of .05 (though the non-default settings resulted in slightly higher estimates).

## Power curves in a full simulation example

Finally, I examined the behaviour of the different simulation settings for a full simulation example (i.e., involving multiple N’s/sample sizes). I simulated power for 5 effect sizes and 3 sample sizes, again using the “Medium Similarity” condition data from Colloff et al. (2021) as a base. For each simulation setting I ran two simulations to get a basic idea of run-to-run consistency. First, the hypothetical ROCs that were tested for this analysis: 

Next, power curves for these simulations: 

These simulations result in the same general expected pattern, but there are a few things worth noting. First, it appears that the default settings (though the fastest for simulation) result in a good amount of run-to-run variability, and a violation of power simulation expectations (i.e., higher power for a smaller effect size with the same sample size). Between increasing the bootstrap iterations and increasing the # of sims, increasing the # of sims seems to result in more run-to-run consistency while maintaining the expected patterns of results, at the cost of increasing the required simulation time (Default = ~50 minutes, NSims X2 = ~100 minutes). At least in these examples, upping the default values of both NSims and NIter did not seem to offer substantial benefit over increasing NSims, and increasing NSims beyond 200 did not seem to result in a substantial gain.

## Recommendations for users

In light of these testing results, I recommend that users: a) Use the default simulation parameters if analysis time is a concern, but to up the # of simulations per sample/effect size to 200 if time is not a concern, b) select only a few effect sizes/sample sizes for simulation (e.g., based on prior results documented in the app), c) set final planned sample size slightly higher than their target power (e.g., + .05-.10), and d) conduct a couple simulation runs (e.g., one with default settings to get a general idea of the sample size, then a finer-grained simulation including only a few sample sizes and using more simulations & bootstrap iterations).