The effects of variance-covariance misspecification in randomised controlled trials with high frequency data collection. A simulation study in sport science.

Paul, A. Swinton

Contact details

Dr. Paul Swinton

School of Health Sciences, Robert Gordon University

Garthdee Road

Aberdeen, UK,

AB10 7QG

p.swinton@rgu.ac.uk, +44 (0) 1224 262 3361

Twitter Handle:
@PaulSwinton9

Abstract

Due to problems of low statistical power in sport science with pre-post randomised controlled trials (RCTs), future studies may seek to conduct high frequency data collection to achieve adequate statistical power with feasible sample sizes. Increasing frequency of data collection requires more sophisticated statistical analyses that may have to account for increased complexity in variance-covariance structures. The purpose of this simulation study was to investigate the effects of misspecifying variance-covariance structures when fitting linear mixed models (LMMs) to realistic data generated across a range of intervention durations and measurement frequencies.

Simulated data were created to compare two interventions across either eight or sixteen weeks, with measurements made either once, twice or three times per week. Data were generated according to four models with different variance-covariance structures and fixed effects held constant. The data generating models included: 1) random intercept only; 2) correlated random intercepts and slopes; 3)&4) correlated random intercepts and slopes with autoregressive level one residuals ($\varphi=0.3$, or $\varphi=0.6$). The effects of both underspecifying and overspecifying variance-covariance structures with fitted LMMs were investigated focusing on type I errors of the average treatment effect when set to zero, and statistical power of the average treatment effect when set to a small but non-zero value. Statistical power of likelihood ratio tests to distinguish variance-covariance structures was also estimated. Monte Carlo simulations were performed across the different conditions with 5000 iterations included in each analysis.

The results showed that underspecifying the variance-covariance structure by fitting a random intercept LMM when more complex structures existed created a high percentage of type I errors that ranged from ~7 to 40%. Percentages of type I errors increased with greater amounts of data from both a longer intervention and increased number of measurements per week. Results also demonstrated that statistical power decreased when the true data generating mechanism included autoregressive level one residuals compared to just correlated random intercepts and slopes. The decline in statistical power was influenced by both the amount of data available and the degree of autocorrelation. Statistical power dropped to its lowest (~0.55) when combining the least amount of data (e.g. eight weeks with one measurement per week) and highest autocorrelation ($\varphi=0.6$). Statistical power of likelihood ratio tests to correctly identify the existence of random slopes was influenced by intervention duration, with low statistical power (0.22 to 0.44) for a short intervention, and high statistical power (>0.99) for a long intervention.

In conclusion, correct specification of the variance-covariance structure is likely to be important for future RCTs in sport science conducted with high frequency measurements. Complex variance-covariance structures particularly with correlated level one residuals may require substantive increases in the number of participants recruited to obtain high statistical power. Researchers should be cautious of omitting random slopes where required given high percentages of type I errors this may cause. Caution is also required when testing for random slopes particularly with relatively small amounts of data that may be achieved with short interventions.

Introduction

Attempts to identify the most effective training interventions represents a prominent area of research in sport science. Given the volume of studies and the trial-and-error practices of athletes, researchers are often aware of several interventions that are highly effective. Randomised controlled trials (RCTs) are considered the gold standard for investigating dose-responses and teasing out the most effective interventions.[1] A major challenge with this approach, however, is that the average treatment effect (ATE) between two successful interventions is likely to be small.[2] For pre-post RCTs where a single measurement is made at baseline and post-intervention, a small ATE requires large sample sizes to reliably identify differences between groups.[3] Other factors including the length of an intervention, treatment response heterogeneity, and measurement error can also strongly influence the number of participants required.[3] A recent large meta-analysis in applied sport science identified that 90% of RCTs included sample sizes of 20 or less per group.[2] Multiple other studies have reported similar descriptions of extremely small sample sizes across different areas of research in sport science.[4,5]

In a recent simulation study generating realistic data for exercise interventions, I demonstrated the effect of plausible changes to intervention duration, interindividual heterogeneity and measurement error on sample size requirements.[3] In the worst-case scenario to obtain statistical power of 0.8 for a pre-post RCT with small ATE, short intervention, large interindividual heterogeneity and large measurement error, over 1500 participants per group may be required.[3] The limitation of pre-post RCTs to investigate small but potentially meaningful effects has been highlighted in sport science and other disciplines.[3,6] One potential strategy to increase statistical power with realistic sample sizes (e.g. 20 to 50 per group) is to collect high-frequency outcome data.[7] In sport science there are situations where outcomes such as body mass, or vertical jump height could be measured with high frequency and not interact with the intervention.[7] Similalrly, there are situations where the training intervention itself could be used to generate high-frequency outcome data potentially through prediction methods. Whilst predictions can result in increased measurement error, collection of high-frequency data has the potential to make up for the additional noise and ultimately reduce the number of participants required to obtain the desired statistical power.[7]

When conducting RCTs with high frequency data, appropriate statistical methods are required. One option that is well suited and has been recommended by some authors to be the default statistical approach to analyse experimental data is linear mixed models (LMMs)[8]. LMMs include fixed effects and random effects, the former represent population-level (i.e.,

average) trends that should be persistent across experiments, and the latter represent the extent to which these trends vary across some grouping factor such as participants.[9] LMMs are linear in their parameters but can include non-linear variables such as time from baseline allowing curvilinear changes over an intervention to be modelled.[10] LMMs can also include a range of parameters to model complex structures that can emerge within high frequency data. Conceptually, we can consider some features of an LMM to model changes in the mean and others the variance-covariance of the data generated.[11] For changes in the mean, we are primarily interested in the group mean slopes representing the expected magnitude of change expressed within a suitable timescale (e.g. per week) and whether these slopes are different. For many interventions we may also expect that there is interindividual variability such that some individuals improve more than others. This variability can be captured by including a random effect term for slope. Typically, this is described by a normal distribution with the zero value representing the group mean change, and amount of variability described by a standard deviation (σ_{b1}). To model variance-covariance we consider two main features including variance between participants and covariance of model residuals from the same participant. [11] Typically, a random intercept for each participant is included in the LMM to describe consistent variation between participants that will exist throughout an intervention. That is, participants that enter an intervention with a high outcome score will typically leave the intervention with a similar relative high outcome score. As with the random effect for slope, we typically model random intercepts with a normal distribution with zero mean and standard deviation (σ_{b0}) describing the variation between participants (e.g. the baseline variation). Within an LMM we can model random intercepts and slopes from a multivariate normal distribution and allow the effects to correlate (ρ). In many interventions we may expect a negative correlation between random intercepts and slopes such that those with higher baseline values experience lower magnitude change due to ceiling effects or simply being further along their potential progression.[12]

The addition of random effect terms combines with level one residuals (e.g. σ_e) and creates assumptions around structure of the variance-covariance in the data. With addition of only a random intercept, a LMM assumes compound symmetry (see Appendix) such that the variance is constant across time points and correlation between time points is the same regardless of the lag between measurements.[13] The addition of both random intercepts and slopes creates a more complex variance-covariance structure (see Appendix). This latter model assumes that total variance will change in a quadratic form due to the contribution of time, and correlations are likely to change between different sets of time points. The inclusion of these random effects describes between-person variation in the data and does not explain outcome variation, but partitions variation into between- and within-person categories. As a result, inclusion of random effects may eliminate within-

participant dependencies that would have existed in residuals when failing to model constant offsets (intercept) or differences in magnitude of change (slopes). Whilst the inclusion of random slopes enables correlations between participants to change across time, like the inclusion of a random intercept, the model assumes that the remaining within-person variation described by the level one residuals are not correlated across time.[11] With longitudinal measurements, particularly with high-frequency data this assumption is unlikely to be correct and we may expect residuals to be correlated such that the magnitude decays with increasing time lag.[14] Models to account for remaining dependencies in level one residuals after including random effects terms are commonly referred to as alternative covariance structures.[11] The most common structure used for longitudinal data is the first order autoregressive structure (AR1) that includes two parameters, ζ_e that assumes constant standard deviation in residuals across time points, and φ that sets the correlation between two adjacent time points, such that φ^n describes the correlation of residuals for time points separated by n intervals. A range of more complex alternative covariance structures can be included that allow for heterogeneous variance in residuals and greater control over decay in correlation across time periods of different length.[15]

When conducting RCTs, the primary interest is in interpretation of fixed effects and in particular, group differences in change. Failure to account for variance-covariance structure in the data may have little effect on estimates of the fixed effects but can greatly influence standard errors that may lead to invalid inferences.[16] Where the covariance structure specified by the LMM and the actual covariance structure are different, we refer to this situation as misspecification. Kwok et al.[17] identified three types of misspecification including underspecified, overspecified, and general-misspecification. Under- or overspecified cases occur when the variance-covariance matrix fitted is simpler or more complex than the true matrix but are nested within each other, respectively.[17] Specifying an AR1 structure when the true structure is compound symmetry would be an example of overspecification. General-misspecification occurs when the specified and true covariance matrix are not nested within each other. As researchers in sport science consider spending additional resources to collect high-frequency data to address problems with low statistical power, it is important to identify the effects of potential issues such as misspecification of variance-covariance structures on inferences made. The current simulation study was conducted to build on recent research and explore analysis of high-frequency data in the context of applied sport science. Simulating different data structures and applying likely LMMs of increasing complexity, the effects of the different structures and potential misspecification on required samples sizes and errors in inference were investigated.

Methods

Approach to the problem

This study builds upon previous simulation work investigating the use of high-frequency data to obtain sufficient statistical power with realistic sample sizes in RCTs of exercise interventions.[7] The purpose of this study was to generate realistic data with different plausible variance-covariance structures and assess the effects of under- and over-specification on statistical power and inferential errors. As with the previous simulation work,[7] a case-based approach generating realistic data informed by the findings from recent large scale meta-analyses was conducted. The data were developed to recreate a RCT investigating the ATE between a "reference" resistance training intervention known to be successful and a new "testing" resistance training intervention hypothesised to be superior in improving maximum strength. The primary outcome measure was the 1RM bench press, which could be predicted using individualised load-velocity relationships.[18-20] Simulations were organised into three broad categories where data were generated with 1) random intercepts; 2) correlated random intercepts and slopes; and 3) correlated random intercepts and slopes with autocorrelation of level one residuals (AR(1) with either $\varphi=0.3$ or $\varphi=0.6$). Percentage of Type I errors (zero ATE) and statistical power (small ATE) were calculated for interventions of eight or sixteen weeks, and for data collection comprising one, two, or three measurements per week.

Data generating and statistical model

In statistical notation, data for the random outcome variable Y were generated for $2n$ participants (n per group) and across the given time points t according to the model

$Y_{ti} = \beta_0 + \beta_1 Group_i + (\beta_2 + \beta_3 Group_i + b_{1i})t + b_{0i} + \epsilon_{ti}$, where $Group_i$ is an indicator variable set to 0 for standard intervention, and 1 for the test intervention, β_0 is the mean of the standard intervention group at baseline ($t = 0$), β_1 is the offset of the test intervention group at baseline and was set to zero, β_2 is the mean change in the standard intervention group per week, and β_3 represents the ATE. All simulations included random intercepts b_{0i}. When generating data with random intercepts and slopes, b_{0i} and b_{1i} were simulated from a multivariate normal distribution with standard deviations σ_{b0} and σ_{b1}, and correlation ρ=-0.2. These two data generating mechanisms included independent normally distributed level one residuals with standard deviation σ_ϵ. Finally, data were generated from the same random intercept and slope model but with level one residuals expressed as $\epsilon_{ti} = \varphi\epsilon_{t-1i} + \zeta_i$, with φ set to either 0.3 or 0.6, mean of ζ_i set to zero and variance set dependent on φ to ensure the total variance of level one residuals was the same across all models.

Data were simulated for a total of forty-eight conditions (4*2*2*3) comprising: 1) four different data generating models; 2) two levels of ATE (zero and small effect); 3) two intervention lengths (eight and sixteen weeks); and 4) three data collection frequencies (one, two, and three times per week). When data were generated according to the random intercept only model, proportion of Type I errors and statistical power were investigated from the overspecified cases fitting LMMs with random intercepts and slopes, either with or without autoregressive level one residuals. Data generated with correlated random intercepts and slopes model were investigated from both underspecified (random intercept) and overspecified cases (inclusion of AR1 level one residuals). Finally, data generated from random intercepts and slopes with autoregressive level one residuals were investigated from underspecified cases.

Data generation with correlated random intercepts and slopes with independent level one residuals was considered the base model for the simulations. As such, the number of participants simulated in each condition was set to achieve statistical power of 0.8 within this base model. Simulating the same number of participants within each condition across data generating models provided a means of easily comparing the effects of model misspecification on proportion of Type I errors and statistical power. For each simulation, point estimates for all fitted parameters were collected, and likelihood ratio tests performed to assess significant differences in fit of the three different classes of LMM. Simulations were performed in R,[21] with the doParallel package[22] used to conduct parallel computation comprising 5000 iterations per condition. Parameter values and group sample sizes for simulations are presented in Table 1 and R code presented in the appendix.

Table 1: Parameter values and group sample sizes used for simulations

|	Fixed effects	Random effects
Parameters		
	β_0=100, β_1=0, β_2=5/12, β_3=0.3	σ_{b0}=15, σ_{b1}=0.25, ρ=-0.2, σ_ϵ=5, φ=0.3 or 0.6, $sd(\zeta)=\sqrt{\sigma_\epsilon^2(1-\varphi^2)}$
Sample size		
Measurement frequency	Intervention length: 8 Weeks	Intervention length: 16 Weeks
1 time per week	n=84	n=21
2 times per week	n=54	n=17
3 times per week	n=41	n=15

Results

Proportion of Type I errors

Proportion of Type I errors obtained for ATE across the different conditions are presented in Figure 1. Results show that overspecification of the variance-covariance structure had minimal influence on percentage of Type I errors, which remained close to five percent. In contrast, underspecifcation with regards to including only random intercepts substantially increased percentage of Type I errors that ranged from ~7 to 40%. Percentages increased with greater amounts of data obtained from both longer intervention and increased number of measurements per week (Figure 1). Percentages also increased with the magnitude of underspecification, such that the greatest percentage of Type I errors were obtained when fitting a random intercept only model when data were generated with correlated intercepts and slopes, and strongly correlated ($\varphi=0.6$) autoregressive level one residuals (Figure 1). Underspecification by including correlated random intercepts and slopes but not including autoregression in level one residuals had minimal effect on percentage of Type I errors.
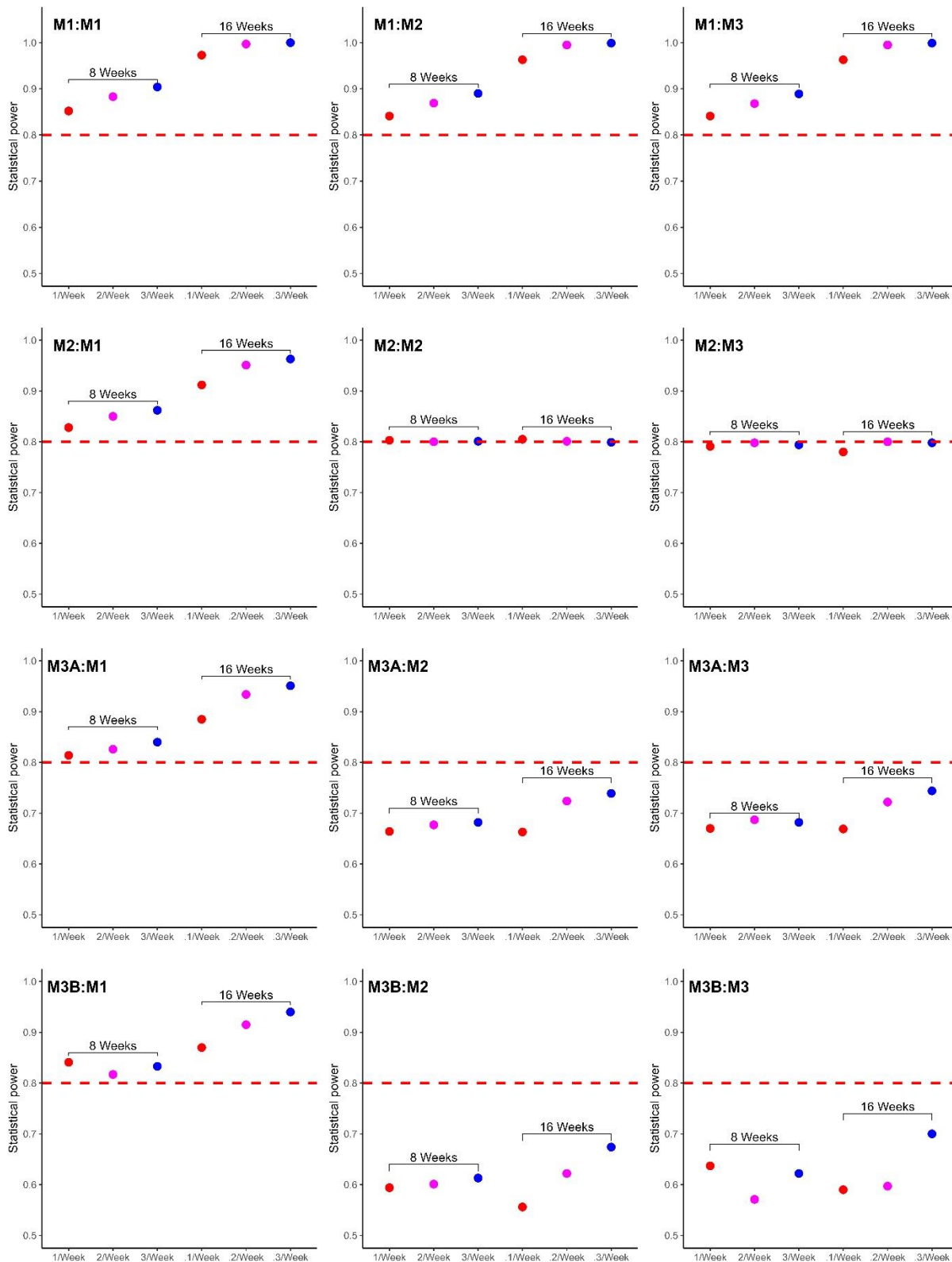
Statistical Power

Statistical power was elevated beyond 0.8 for the correctly specified random intercept only model due to powering sample sizes based on data generated from the more complex correlated random intercepts and slopes structure. Statistical power increased with both longer intervention and greater measurement frequency, with values ranging between ~0.85 to 1.0 (Figure 2). Statistical power remained elevated beyond 0.8 with the same patterns for intervention duration and measurement frequency when underspecifying with the intercept only model for more complex variance-covariance structures (Figure 2). In contrast, statistical power reduced when data were generated with autoregressive level one residuals. The magnitude of the decrease in statistical power was not influenced by whether the variance-covariance structure fitted was correctly specified or underspecified by omitting autoregressive level one residuals (Figure 2). The reduction in statistical power was influenced by measurement frequency and magnitude of the autocorrelation, with the greatest reductions obtained with only one measurement per week and for larger autocorrelations (Figure 2).

Figure 1: Percentage of Type I errors for average treatment effect across the different simulated conditions (rows) and analysed with different linear mixed models (columns).
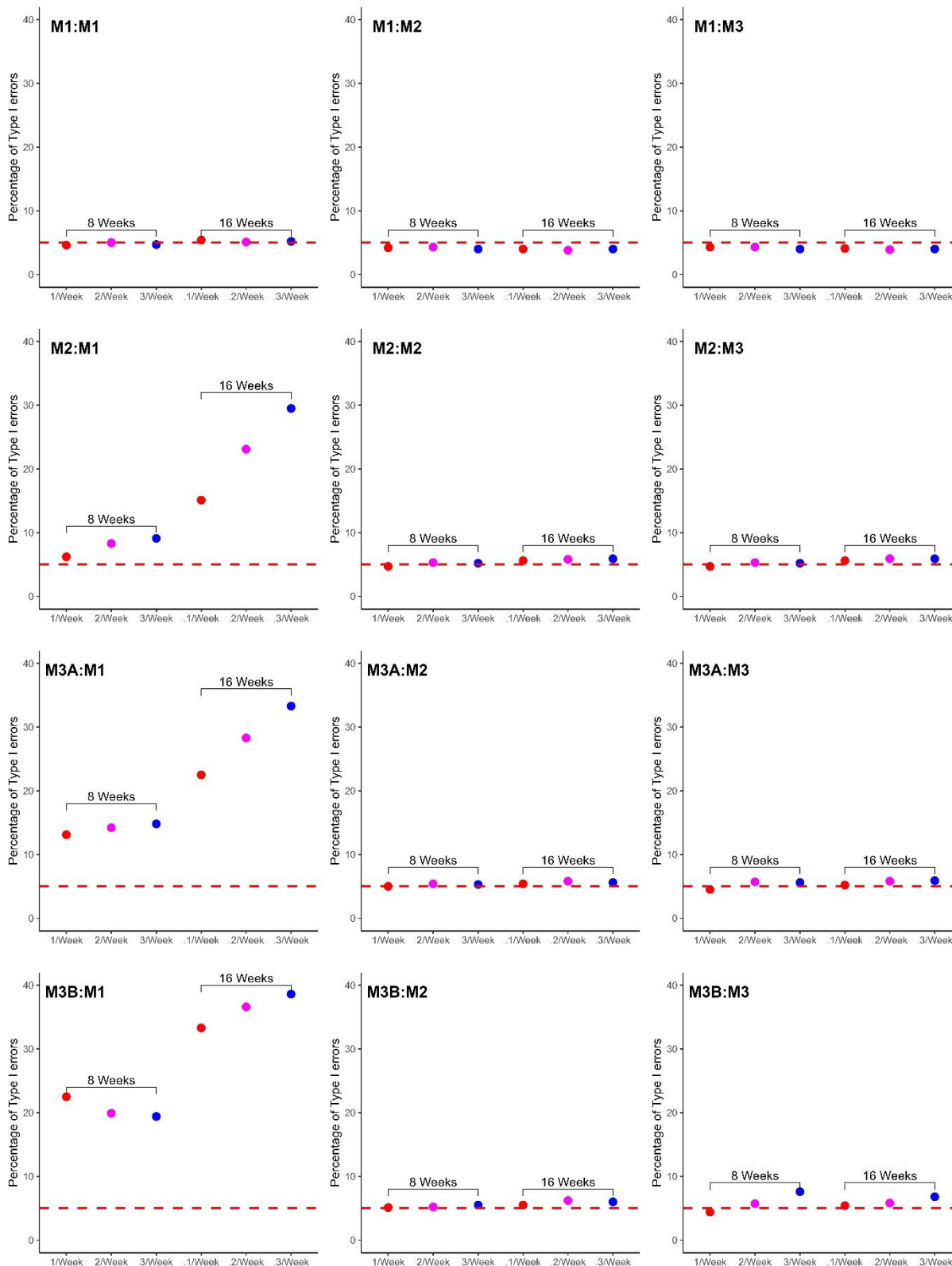
Each row represents data generated from a different model (M1: random intercepts only; M2: correlated random intercepts and slopes; M3A: correlated random intercepts and slopes with autoregressive level one residuals ($\varphi=0.3$); M3B: correlated random intercepts and slopes with autoregressive level one residuals ($\varphi=0.6$). Each column represents fitting of the different models. Hence, M3A:M2 shows percentage of Type I errors for data generated from correlated random intercepts and slopes with autoregressive level one residuals ($\varphi=0.3$) but analysed with LMM with only correlated random intercepts and slopes. Red lines denote standard percentage of Type I errors targeted in studies.

Figure 2: Statistical power for average treatment effect across the different simulated conditions (rows) and analysed with different linear mixed models (columns).

Each row represents data generated from a different model (M1: random intercepts only; M2: correlated random intercepts and slopes; M3A: correlated random intercepts and slopes with autoregressive level one residuals (φ=0.3); M3B: correlated random intercepts and slopes with autoregressive level one residuals (φ=0.6]). Each column represents fitting of the different models. Hence, M3A:M2 shows percentage of Type I errors for data generated from correlated random intercepts and slopes with autoregressive level one residuals (φ=0.3) but analysed with LMM with only correlated random intercepts and slopes. Red lines denote standard statistical power targeted in studies.

Use of likelihood ratio test to determine data generating model

The percentage of Type I errors obtained from likelihood ratio tests for overspecification when data were generated with random intercepts only, ranged from 2.0 to 2.7% when fitting correlated random intercepts and slopes, and 2.3 to 3.0% when fitting correlated random intercepts and slopes with autoregressive level one residuals. A slight increase in percentage of Type I errors ranging from 4.8 to 5.5% were obtained from likelihood ratio tests for overspecification when data were generated from correlated random intercepts and slopes but fitting models that included autoregressive level one residuals. In contrast, large variation in statistical power was seen when conducting the likelihood ratio test with underspecification for data generated with correlated random intercepts and slopes. With measurements made over an eight-week intervention, statistical power ranged from 0.22 (once per week) to 0.44 (three per week). Much greater statistical power was obtained with measurements made over a sixteen-week intervention with values ranging from 0.80 (once per week) to 0.99 (three per week). For data generated with the inclusion of autoregressive errors statistical power was greater than 0.99 for all conditions and comparisons.

Estimation of variance-covariance parameters

Analysis of point estimates of variance-covariance parameters was conducted under correctly specified LMMs and are presented in Figure 3. In general, results show greater accuracy and reduced uncertainty when increasing the amount of data available, with the greatest improvements obtained when moving from eight to a sixteen-week intervention. The correlation between random intercepts and slopes showed the greatest uncertainty that was likely to be meaningful in practice.

Figure 3: Quartile plots illustrating estimation of variance-covariance parameters using correctly specified models across different conditions.
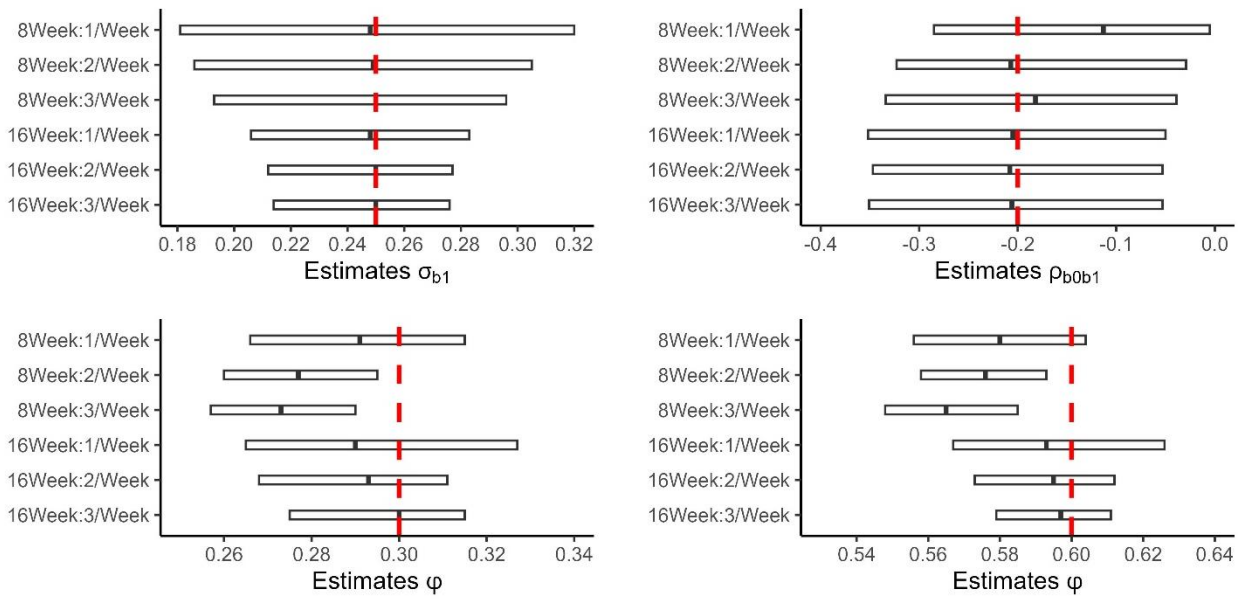
Top row: Estimates of standard deviation of random slopes and correlation between random intercepts and slopes. Bottom row: Estimates of correlation between autoregression of level one residuals. Red lines denote true parameter values.

Discussion

The results from this study show that misspecification of the variance-covariance structure when analysing high frequency RCT data with LMMs can have substantial effects on inferences made. The most serious problems with inferences occurred when underspecifying the variance-covariance structure. Most importantly, failure to include random slopes where required increased the percentage of Type I errors to unacceptable levels. Additionally, the results show that underspecifying the variance-covariance structure when determining sample size can reduce statistical power limiting the ability to correctly identify potentially meaningful ATEs.

The results from the simulations are consistent with previous research showing that excluding random effects structure such as random slopes inflates the risk of Type I errors.[23,24] The cause of this inflation is overestimating degrees of freedom, sometimes referred to as pseudoreplication, where observations are treated as independent such that residuals are falsely assumed to be independent.[25] The extent to which pseudoreplication increases Type I errors has been shown to be influenced by the statistical test applied.[25] Different approximations to degrees of freedom are available that are influenced by unbalanced data, which may be extensive in RCTs where high frequency data is sought. The extent to which missing data influences statistical power and validity of inferences is an important area for future research of RCTs comprising high frequency data.

Previous research also highlights multiple contrasting perspectives to specify the variance-covariance structure of high frequency data. One common perspective contrasts theoretical with data-driven approaches.[24,25] Another common perspective contrasts maximal versus parsimonious models.[24,25] In the simulations conducted in the present study, limitations of the data-driven approach to determine variance-covariance structure were highlighted. Statistical power to identify random slopes varied considerably depending on the amount of data available. When only one measurement per week was conducted over eight-weeks, the statistical power to return a significant difference in the likelihood ratio test of random intercepts, versus random intercepts and slopes was 0.22. The statistical power increased to 0.99 when three measurements were made per week over sixteen-weeks. In contrast, the statistical power to accurately determine the existence of an autoregressive parameter was greater than 0.99 for all conditions and correlation values. Further research is required to determine the statistical power for autoregression with a smaller correlation, and more complex alternative covariance structures with greater numbers of parameters.

Given the importance of including random slopes in the data investigated, and the potential low statistical power of the likelihood ratio test, a more theory driven approach at least for inclusion of random effects may be appropriate. It is common in sport science to assume that individuals respond differently to interventions.[26] This difference can be expressed by including random slopes. Over longer duration interventions researchers may choose to model changes in an outcome with non-linear methods including entering time as a polynomial. Random slopes can then be added to the different polynomial time coefficients, further increasing the complexity of the model.[10] Regardless of the method used, it is important to note the differences between heterogeneity in observed changes versus the recent interest in interindividual response heterogeneity.[27,28] With the former, no consideration is made to the cause of the heterogeneity with factors both external and internal to the training stimulus combined. External factors potentially include nutrition, sleep, and additional exercise performed outside the intervention, whereas internal factors describe interactions of the exercise intervention with for example training experience and genetic variants. Recent studies have attempted to parse these sources of variation by comparing changes across an intervention between an exercise group and a non-exercise control.[28] Many of the studies conducted including meta-analyses have been unable to identify additional variability of the exercise group beyond the control.[29-31] The results of these studies have been interpreted as showing a lack of interindividual response variability (e.g., additional variation in change score in the intervention group caused by interactions of participants characteristics and the exercise stimulus). However, when multiple exercise groups are compared in an RCT, the inclusion of random slopes does not distinguish between sources of variability and it should be expected that variability in change occurs and as such the effect included in the LMM.

In contrast to theory leading the inclusion of random effects, the ability to theorise which is the best alternative covariance structure may be less clear. The most appropriate structure for level one residuals after inclusion of random effects may be influenced by a range of factors including the frequency of the data and the outcome selected. The higher the frequency of the data the more likely complex alternative covariance structures with many parameters will be required. Similarly, some tests that may be considered feasible for high-frequency data collection could exhibit more or less complex variance-covariance structures depending on the outcome selected. For example, the vertical jump is frequently performed by athletes' multiple times per week to monitor both fatigue and improvements in physical capacity.[32,33] However, research shows that the same vertical jump height that is caused by the product of force and velocity can be achieved by a range of different jumping strategies potentially masking fluctuations caused by factors such as fatigue.[34] In contrast, force-time variables such as time to peak force may be sensitive to acute fluctuations [35] such that residuals from a statistical model

display a more complex structure. Given the range of factors that may influence relationships between residuals in longitudinal data collected in sport science, it seems that some use of data driven approaches is required for the alterative covariance structure with a view to achieving a degree of parsimony. Research employing timeseries approaches to investigate covariance structures within participants is required to better understand the complexity of structures that are likely to exist within the data and the most influential factors. This research will be important as the present study highlights that more complex structures modelled within longitudinal data will require greater sample sizes to achieve appropriate statistical power for ATEs. Given the context of RCTs with high frequency data as a means of achieving appropriate statistical power with manageable resources, overspecifying covariance structures may inflate sample size calculations and deter studies that may ultimately be feasible.

In addition to estimating the ATE, other parameters of an LMM can be estimated to address research questions and test specific hypotheses. For example, variability in response may be of interest to both researchers and practitioners. Interventions that produce relatively homogenous changes such that most participants experience similar meaningful improvements may be more desirable than interventions where heterogenous changes occur such that some experience large improvements, but many do not experience any improvement. The variability of random slopes provides a measure of heterogeneity and should be reported in studies that include an LMM. In the present study the results from the simulations identified that the standard deviation of the random slopes could be estimated appropriately. Precision, however, was influenced strongly by the amount of data, with study duration the most influential factor (Figure 3). Researchers and practitioners have also shown interest in quantifying potential relationships between baseline value and magnitude of change.[12] The research designs and statistical approaches commonly used, however, have been shown to be limited and influenced by regression to the mean effects. Whilst adjustment estimates are available,[36] the use of LMMs and the correlation between random intercepts and slopes has been identified as a more sophisticated approach to address the question and provide the opportunity to control for other variables where appropriate.[37] The results from this simulation study, however, highlight that uncertainty in the correlation parameter may be quite large and remained consistent across different study lengths and measurement frequencies, such that estimates for relatively small values as that used here (ρ=-0.2) do not provide clear interpretations. This finding highlights the limited advantage of additional longitudinal data and the need for greater numbers of participants to obtain precise estimates for some parameters.

In conclusion, the results of the present study show that correct specification of the variance-covariance structure is likely to be important for future RCTs in sport science conducted with high frequency measurements. For these designs it is recommended that random slopes are included in models based on a theoretical perspective rather than a data-driven approach. It is important for researchers to be aware that complex variance-covariance structures particularly with correlated level one residuals may require substantive increases in the number of participants to obtain high statistical power. Future research should apply time series analyses to longitudinal data to assess which variance-covariance structures are likely to exist within high frequency data in sport science and investigate how structures are influenced by measurement frequency and other factors including the specific outcome.

References

1. Hecksteden A, Faude O, Meyer T, Donath L. How to construct, conduct and analyze an exercise training study? Frontiers in physiology. 2018;9:389083. https://doi.org/10.3389/fphys.2018.01007.

2. Swinton, PA. Murphy, A. Comparative effect size distributions in strength and conditioning and implications for future research: A meta-analysis. 2022. Pre-print available from SportRχiv. https://doi.org/10.51224/SRXIV.202.

3. Swinton, PA. Is it time to rethink pre-post randomised controlled trials in strength and conditioning? A review of statistical approaches with derivations and simulations. 2023. Pre-print available from SportRχiv. https://doi.org/10.51224/SRXIV.363.

4. Abt G, Boreham C, Davison G, Jackson R, Nevill A, Wallace E, Williams M. Power, precision, and sample size estimation in sport and exercise science research. Journal of Sports Sciences. 2020;38(17):1933-5. https://doi.org/10.1080/02640414.2020.1776002.

5. Bernards JR, Sato K, Haff GG, Bazyler CD. Current research and statistical practices in sport science and a need for change. Sports. 2017;5(4):87. https://doi.org/10.3390/sports5040087

6. Götz FM, Gosling SD, Rentfrow PJ. Small effects: The indispensable foundation for a cumulative psychological science. Perspectives on psychological science. 2022;17(1):205-15. https://doi.org/10.1177/1745691620984483.

7. Swinton, PA. Adequate statistical power in strength and conditioning may be achieved through longer interventions and high frequency outcome measurement. 2023. Pre-print available from SportRχiv. https://doi.org/10.51224/SRXIV.364.

8. McElreath R. Statistical rethinking: A Bayesian course with examples in R and Stan. Chapman and Hall/CRC; 2018. https://doi.org/10.1201/9781315372495.

9. Brown VA. An introduction to linear mixed-effects modeling in R. Advances in Methods and Practices in Psychological Science. 2021;4(1):2515245920960351. https://doi.org/10.1177/2515245920960351.

10. Mirman D. Growth curve analysis and visualization using R. CRC press; 2017. https://doi.org/10.1201/9781315373218.

11. Hoffman L. Longitudinal analysis: Modeling within-person fluctuation and change. Routledge; 2015. https://doi.org/10.4324/9781315744094.

12. Swinton, PA. The influence of baseline capability on intervention effects in strength and conditioning: A review of concepts and methods with meta-analysis. Pre-print available from SportRχiv. https://doi.org/10.51224/SRXIV.285.

13. Kutner MH, Nachtsheim CJ, Neter J, Li W. Applied linear statistical models. McGraw-hill; 2005.

14. Browne W, Goldstein H. MCMC sampling for a multilevel model with nonindependent residuals within and between cluster units. Journal of Educational and Behavioral Statistics. 2010;35(4):453-73. https://doi.org/10.3102/1076998609359.

15. Blozis SA, Craft M. Alternative covariance structures in mixed-effects models: Addressing intra-and inter-individual heterogeneity. Behavior Research Methods. 2023:1-20. https://doi.org/10.3758/s13428-023-02133-1.

16. Fitzmaurice GM, Laird NM, Ware JH. Applied longitudinal analysis. John Wiley & Sons; 2012.

17. Kwok OM, West SG, Green SB. The impact of misspecifying the within-subject covariance structure in multiwave longitudinal multilevel models: A Monte Carlo study. Multivariate Behavioral Research. 2007;42(3):557-92. https://doi.org/10.1080/00273170701540537.

18. Balsalobre-Fernández C, Kipp K. Use of machine-learning and load–velocity profiling to estimate 1-repetition maximums for two variations of the bench-press exercise. Sports. 2021;9(3):39. https://doi.org/10.3390/sports9030039

19. Fernandes JF, Dingley AF, Garcia-Ramos A, Perez-Castilla A, Tufano JJ, Twist C. Prediction of one repetition maximum using reference minimum velocity threshold values in young and middle-aged resistance-trained males. Behavioral Sciences. 2021;11(5):71. https://doi.org/10.3390/bs11050071.

20. Pérez-Castilla A, Fernandes JF, Garcia-Ramos A. Validity of the bench press one-repetition maximum test predicted through individualized load-velocity relationship using different repetition criteria and minimal velocity thresholds. Isokinetics and Exercise Science. 2021;29(4):369-77. https://doi.org/10.3233/IES-202247.

21. R Core Team (2023). R: A Language and Environment for Statistical Computing. R Foundation for Statistical Computing, Vienna, Austria.

22. Weston S, Analytics R. doParallel: Foreach parallel adaptor for the parallel package. R package version. 2019;1:15.

23. Barr DJ, Levy R, Scheepers C, Tily HJ. Random effects structure for confirmatory hypothesis testing: Keep it maximal. Journal of memory and language. 2013;68(3):255-78. https://doi.org/10.1016/j.jml.2012.11.001.

24. Barr DJ. Random effects structure for testing interactions in linear mixed-effects models. Frontiers in psychology. 2013;4:54057. https://doi.org/10.3389/fpsyg.2013.00328.

25. Scandola M, Tidoni E. Reliability and feasibility of Linear Mixed Models in fully crossed experimental designs. Advances in Methods and Practices in Psychological Science. 2024;7(1):25152459231214454. https://doi.org/10.1177/251524592312144.

26. Swinton, PA. Assessing individual response to training in sport and exercise. A conceptual and statistical review. 2023. Pre-print available from SportRχiv. https://doi.org/10.51224/SRXIV.288.

27. Hrubeniuk TJ, Bonafiglia JT, Bouchard DR, Gurd BJ, Sénéchal M. Directions for exercise treatment response heterogeneity and individual response research. International Journal of Sports Medicine. 2021:11-22. https://doi.org/10.1055/a-1548-7026.

28. Ross R, Goodpaster BH, Koch LG, Sarzynski MA, Kohrt WM, Johannsen NM, Skinner JS, Castro A, Irving BA, Noland RC, Sparks LM. Precision exercise medicine: understanding exercise response variability. British journal of sports medicine. 2019;53(18):1141-53. https://doi.org/10.1136/bjsports-2018-100328.

29. Steele J, Fisher JP, Smith D, Schoenfeld BJ, Yang Y, Nakagawa S. Meta-analysis of variation in sport and exercise science: examples of application within resistance training research. Journal of Sports Sciences. 2023;41(17):1617-34. https://doi.org/10.1080/02640414.2023.2286748.

30. Kelley GA, Kelley KS, Stauffer BL. Effects of resistance training on body weight and body composition in older adults: An inter-individual response difference meta-analysis of randomized controlled trials. Science Progress. 2023;106(2):00368504231179062. https://doi.org/10.1177/00368504231179062.

31. Bonafiglia JT, Swinton PA, Ross R, Johannsen NM, Martin CK, Church TS, Slentz CA, Ross LM, Kraus WE, Walsh JJ, Kenny GP. Interindividual differences in trainability and moderators of cardiorespiratory fitness, waist circumference, and body mass responses: a large-scale individual participant data meta-analysis. Sports Medicine. 2022;52(12):2837-51. https://doi.org/10.1007/s40279-022-01725-9.

32. Carlson K, Magnusen M, Walters P. Effect of various training modalities on vertical jump. Research in sports medicine. 2009;17(2):84-94. https://doi.org/10.1080/15438620902900351.

33. Claudino JG, Cronin J, Mezêncio B, McMaster DT, McGuigan M, Tricoli V, Amadio AC, Serrão JC. The countermovement jump to monitor neuromuscular status: A meta-analysis. Journal of science and medicine in sport. 2017;20(4):397-402. https://doi.org/10.1016/j.jsams.2016.08.011.

34. Ruddock AD, Winter EM. Jumping depends on impulse not power. Journal of sports sciences. 2016;34(6):584-5. https://doi.org/10.1080/02640414.2015.1064157

35. Hughes S, Warmenhoven J, Haff GG, Chapman DW, Nimphius S. Countermovement jump and squat jump force-time curve analysis in control and fatigue conditions. The Journal of Strength & Conditioning Research. 2022;36(10):2752-61. https://doi.org/10.1519/JSC.0000000000003955.

36. Oldham PD. A note on the analysis of repeated measurements of the same subjects. Journal of chronic diseases. 1962;15(10):969-77. https://doi.org/10.1016/0021-9681(62)90116-9.

37. Blance A, Tu YK, Gilthorpe MS. A multilevel modelling solution to mathematical coupling. Statistical Methods in Medical Research. 2005;14(6):553-65. https://doi.org/10.1191/0962280205sm418oa.

The effects of variance-covariance misspecification in randomised controlled trials with high frequency data collection. A simulation study in sport science.

Paul, A. Swinton

Appendix

Linear mixed model overview

A typical longitudinal linear mixed model (LMM) can be expressed as

$$Y_{ti} = X_{ti}\beta + Z_i b_i + \epsilon_{ti},$$

where Y_{ti} is the response variable at time t nested within the i-th individual. X_{ti} is an n_i by p design matrix of covariates for the fixed effects, where n_i is the total number of observations for the i-th individual and p is the number of covariates included in the model. Z_{ti} is the design matrix for the random effects and is commonly formed from a subset of columns from X_{ti}. The b_i are the random effects and are unique for each individual but constant across time for each individual. Finally, ϵ_{ti} are the level one residuals and represent deviations from the individual growth curves.

We assume that the random effects b_i are independent across individuals, arise from a multivariate normal distribution with mean $\mathbf{0}$, and covariance matrix $\mathbf{G}$. Often, the level one residuals are assumed to be independent. However, more complex level 1 residuals can be included and sometimes it is useful to have $\epsilon_i = \epsilon_{(1)i} + \epsilon_{(2)i}$, where $\epsilon_{(1)i}$ represents measurement error and $\epsilon_{(2)i}$ represents serial correlation, which usually describes a random process within an individual that decreases as the time lag increases.

In matrix form we have

$$\mathbf{Y}_i = \mathbf{X}_i\boldsymbol{\beta} + \mathbf{Z}_i\mathbf{b}_i + \boldsymbol{\epsilon},$$

and so

$$\text{Var}(\mathbf{Y}_i) = \mathbf{Z}_i\text{Var}(\mathbf{b}_i)\mathbf{Z}_i^T + \sigma_\epsilon^2 \mathbf{I}_{n_i} + \tau^2 \mathbf{H}_i,$$

where $\mathbf{Var}(\mathbf{b}_i) = \mathbf{G}$, $\mathbf{I}_{n_i}$ is the identity matrix with n_i rows and columns, $\tau^2 \mathbf{H}_i$ expresses the serial correlation such that $\mathbf{H}_i$ is a $n_i \times n_i$ matrix where the (j, k) element is the correlations between the two time points within an individual. The variance can be considered easily through simulation (see below in R code). Simply simulate n data points according to the model formula with random effects held constant and serial correlations between time points (e.g. $\mathbf{H}_i$) held constant for that iteration. Then in a loop simulate $(m - 1) * n$ more data points allowing the random effects to change across iterations according to $\mathbf{G}$.

The first model that we consider is the random intercepts model

$$Y_{ti} = \beta_0 + \beta_1 Group_i + (\beta_2 + \beta_3 Group_i)t + b_{0i} + \epsilon_{ti}, \; b_{0i} \sim N(0, \sigma_{b0}^2), \; \epsilon_{ti} \sim N(0, \sigma_\epsilon^2),$$

where $Group_i$ is an indiciator variable which takes the value 0 for standard intervention, and 1 for the test intervention. This means that β_0 represents the mean of the standard intervention group at baseline $(t = 0)$, β_1 is the offset of the test intervention group at baseline, β_2 is the change in standard intervention group mean per unit of time (e.g. per week), and β_3 is any difference in the change in mean of the test intervention group relative to the standard intervention group per unit of time.

Here we have $\boldsymbol{G} = \sigma_{b0}^2$, $\boldsymbol{Z}_i = \boldsymbol{1}_{n_i}$, and so if $n_i = 4$, then we have

$$\text{Var}(\boldsymbol{Y}_i) = \begin{bmatrix} \sigma_{b0}^2 + \sigma_\epsilon^2 & \sigma_{b0}^2 & \sigma_{b0}^2 & \sigma_{b0}^2 \\ \sigma_{b0}^2 & \sigma_{b0}^2 + \sigma_\epsilon^2 & \sigma_{b0}^2 & \sigma_{b0}^2 \\ \sigma_{b0}^2 & \sigma_{b0}^2 & \sigma_{b0}^2 + \sigma_\epsilon^2 & \sigma_{b0}^2 \\ \sigma_{b0}^2 & \sigma_{b0}^2 & \sigma_{b0}^2 & \sigma_{b0}^2 + \sigma_\epsilon^2 \end{bmatrix}$$

The correlation between two time points is then $\frac{\sigma_{b0}^2}{\sigma_{b0}^2 + \sigma_\epsilon^2}$.

The second model that we consider is the random intercepts and slopes model

$$Y_{ti} = \beta_0 + \beta_1 Group_i + (\beta_2 + \beta_3 Group_i + b_{1i})t + b_{0i} + \epsilon_{ti}, \; \begin{matrix} b_{0i} \\ b_{1i} \end{matrix} \sim N\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} \sigma_{b0}^2 & \rho\sigma_{b0}\sigma_{b1} \\ \rho\sigma_{b0}\sigma_{b1} & \sigma_{b1}^2 \end{bmatrix}\right), \epsilon_{ti} \sim N(0, \sigma_\epsilon^2).$$

Here we have $\boldsymbol{Z}_i = c(\boldsymbol{1}_{n_i}, \boldsymbol{t}_{n_i})$, $\boldsymbol{G} = \begin{bmatrix} \sigma_{b0}^2 & \rho\sigma_{b0}\sigma_{b1} \\ \rho\sigma_{b0}\sigma_{b1} & \sigma_{b1}^2 \end{bmatrix}$, so if $n_i = 4$, then we have

$$\text{Var}(\boldsymbol{Y}_i) = \begin{bmatrix} 1 & t_0 \\ 1 & t_1 \\ 1 & t_2 \\ 1 & t_3 \end{bmatrix} \begin{bmatrix} \sigma_{b0}^2 & \rho\sigma_{b0}\sigma_{b1} \\ \rho\sigma_{b0}\sigma_{b1} & \sigma_{b1}^2 \end{bmatrix} \begin{bmatrix} 1 & 1 & 1 & 1 \\ t_0 & t_1 & t_2 & t_3 \end{bmatrix} + \begin{bmatrix} \sigma_\epsilon^2 & 0 & 0 & 0 \\ 0 & \sigma_\epsilon^2 & 0 & 0 \\ 0 & 0 & \sigma_\epsilon^2 & 0 \\ 0 & 0 & 0 & \sigma_\epsilon^2 \end{bmatrix} =$$

$$\begin{bmatrix} \sigma_{b0}^2 + 2t_0\rho\sigma_{b0}\sigma_{b1} + t_0^2\sigma_{b1}^2 + \sigma_\epsilon^2 & \sigma_{b0}^2 + (t_0 + t_1)\rho\sigma_{b0}\sigma_{b1} + t_0 t_1\sigma_{b1}^2 & \sigma_{b0}^2 + (t_0 + t_2)\rho\sigma_{b0}\sigma_{b1} + t_0 t_2\sigma_{b1}^2 & \sigma_{b0}^2 + (t_0 + t_3)\rho\sigma_{b0}\sigma_{b1} + t_0 t_3\sigma_{b1}^2 \\ \sigma_{b0}^2 + (t_0 + t_1)\rho\sigma_{b0}\sigma_{b1} + t_0 t_1\sigma_{b1}^2 & \sigma_{b0}^2 + 2t_1\rho\sigma_{b0}\sigma_{b1} + t_1^2\sigma_{b1}^2 + \sigma_\epsilon^2 & 0\sigma_{b0}^2 + (t_1 + t_2)\rho\sigma_{b0}\sigma_{b1} + t_1 t_2\sigma_{b1}^2 & \sigma_{b0}^2 + (t_1 + t_3)\rho\sigma_{b0}\sigma_{b1} + t_1 t_3\sigma_{b1}^2 \\ \sigma_{b0}^2 + (t_0 + t_2)\rho\sigma_{b0}\sigma_{b1} + t_0 t_2\sigma_{b1}^2 & \sigma_{b0}^2 + (t_1 + t_2)\rho\sigma_{b0}\sigma_{b1} + t_1 t_2\sigma_{b1}^2 & \sigma_{b0}^2 + 2t_2\rho\sigma_{b0}\sigma_{b1} + t_2^2\sigma_{b1}^2 + \sigma_\epsilon^2 & \sigma_{b0}^2 + (t_2 + t_3)\rho\sigma_{b0}\sigma_{b1} + t_2 t_3\sigma_{b1}^2 \\ \sigma_{b0}^2 + (t_0 + t_3)\rho\sigma_{b0}\sigma_{b1} + t_0 t_3\sigma_{b1}^2 & \sigma_{b0}^2 + (t_1 + t_3)\rho\sigma_{b0}\sigma_{b1} + t_1 t_3\sigma_{b1}^2 & \sigma_{b0}^2 + (t_2 + t_3)\rho\sigma_{b0}\sigma_{b1} + t_2 t_3\sigma_{b1}^2 & \sigma_{b0}^2 + 2t_3\rho\sigma_{b0}\sigma_{b1} + t_3^2\sigma_{b1}^2 + \sigma_\epsilon^2 \end{bmatrix}.$$

The correlation between two time points $(t, t + m)$ is then $\dfrac{\sigma_{b0}^2 + (t + (t+m))\rho\sigma_{b0}\sigma_{b1} + t(t+m)\sigma_{b1}^2}{\sqrt{\sigma_{b0}^2 + 2t\rho\sigma_{b0}\sigma_{b1} + t^2\sigma_{b1}^2 + \sigma_\epsilon^2}\sqrt{\sigma_{b0}^2 + 2(t+m)\rho\sigma_{b0}\sigma_{b1} + (t+m)^2\sigma_{b1}^2 + \sigma_\epsilon^2}}$.

The third model that we consider is the random intercepts and slopes model with first order autoregressive level one residuals

$$Y_{ti} = \beta_0 + \beta_1 Group_i + (\beta_2 + \beta_3 Group_i + b_{1i})t + b_{0i} + \epsilon_{ti}, \; \begin{matrix} b_{0i} \\ b_{1i} \end{matrix} \sim N\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} \sigma_{b0}^2 & \rho\sigma_{b0}\sigma_{b1} \\ \rho\sigma_{b0}\sigma_{b1} & \sigma_{b1}^2 \end{bmatrix}\right),$$

$\epsilon_{ti} = \varphi\epsilon_{t-1i} + \zeta_i$, $\zeta_i \sim N(0, \sigma_\zeta^2)$. As previously we have $Z_i = c(1_{n_i}, t_{n_i})$, $G = \begin{bmatrix} \sigma_{b0}^2 & \rho\sigma_{b0}\sigma_{b1} \\ \rho\sigma_{b0}\sigma_{b1} & \sigma_{b1}^2 \end{bmatrix}$.

To derive the properties of the level one residuals we express ϵ_t as an infinite moving average time series

$\epsilon_t = \varphi\epsilon_{t-1} + \zeta_t = \varphi(\varphi\epsilon_{t-2} + \zeta_{t-1}) + \zeta_t = \varphi^t\epsilon_0 + \sum_{i=0}^{t-1} \varphi^i \zeta_{t-i} \to \sum_{i=0}^{\infty} \varphi^i \zeta_{t-i}$ as $t \to \infty$ if $|\varphi| <$

1 and ϵ_0 is finite. From $\epsilon_t = \sum_{i=0}^{\infty} \varphi^i \zeta_{t-i}$ we have $E(\epsilon_t) = 0$. $\text{Var}(\epsilon_t) = \sum_{i=0}^{\infty} \varphi^{2i} \sigma_\zeta^2$ which from infinite

geometric series $\sum_{i=0}^{\infty} ar^k = \frac{a}{1-r}$ for $|r| < 1$ and for series containing only even powers of r, $\sum_{i=0}^{\infty} ar^{2k} = \frac{a}{1-r^2}$, thus

$\text{Var}(\epsilon_t) = \sum_{i=0}^{\infty} \varphi^{2i} \sigma_\zeta^2 = \frac{\sigma_\zeta^2}{1-\varphi^2}$, $|\varphi| < 1$.

For the covariance and correlation of ϵ_t and ϵ_{t+m}, We have

$\text{Cov}(\epsilon_t, \epsilon_{t+m}) = E(\epsilon_t, \epsilon_{t+m}) - 0 = E\left(\sum_{i=0}^{\infty} \varphi^i\zeta_{t-i} \sum_{j=0}^{\infty} \varphi^j\zeta_{t+m-j}\right) = E\left(\sum_{i=0}^{\infty} \sum_{j=0}^{\infty} \varphi^i\zeta_{t-i} \varphi^j\zeta_{t+m-j}\right) =$

$E\left(\sum_{i,j=0}^{\infty} \varphi^i\varphi^j \zeta_{t-i}\zeta_{t+m-j}\right) = \sum_{i,j=0}^{\infty} \varphi^i\varphi^j E\left(\zeta_{t-i}\zeta_{t+m-j}\right)$.

ζ_{t-i} and ζ_{t+m-j} are independent except when $-i = m - j$, that is $j = i + m$. In cases when $j \neq i + m$, then

$E(\zeta_{t-i})E(\zeta_{t+m-j}) = 0$. However, when $j = i + m$ $E(\zeta_{t-i}\zeta_{t+m-j}) = \sigma_\zeta^2$. Therefore, we can rewrite the sum under

the constraint that $j = i + m$ as

$\sum_{i=0}^{\infty} \varphi^i\varphi^{i+m} E(\zeta_{t-i}^2) = \sum_{i=0}^{\infty} \varphi^i\varphi^{i+m}\sigma_\zeta^2 = \sigma_\zeta^2\varphi^m \sum_{i=0}^{\infty} \varphi^{2i}$, where again we use the geometric series to give

$\text{Cov}(\epsilon_t, \epsilon_{t+m}) = \frac{\varphi^m\sigma_\zeta^2}{1-\varphi^2}$, $|\varphi| < 1$.

The correlation is then

$\frac{\varphi^m\sigma_\zeta^2}{1-\varphi^2} \frac{1-\varphi^2}{\sigma_\zeta^2} = \varphi^{|m|}, m = 0, \pm, \dots$.

From $\text{Var}(Y_i) = Z_i\text{Var}(b_i)Z_i^T + \sigma_\epsilon^2 I_{n_i} + \tau^2 H_i$, we have that $\sigma_\epsilon^2 I_{n_i} + \tau^2 H_i = \frac{\sigma_\zeta^2}{1-\varphi^2}\begin{bmatrix} 1 & \varphi & \varphi^2 & \varphi^3 \\ \varphi & 1 & \varphi & \varphi^2 \\ \varphi^2 & \varphi & 1 & \varphi \\ \varphi^3 & \varphi^2 & \varphi & 1 \end{bmatrix}$,

The variance at time point t is then $\sigma_{b0}^2 + 2t\rho\sigma_{b0}\sigma_{b1} + t^2\sigma_{b1}^2 + \frac{\sigma_\zeta^2}{1-\varphi^2}$ and the correlation between two time points $(t, t+m)$ is then

$$\frac{\sigma_{b0}^2 + \left(t+(t+m)\right)\rho\sigma_{b0}\sigma_{b1} + t(t+m)\sigma_{b1}^2 + \frac{\varphi^m\sigma_\zeta^2}{1-\varphi^2}}{\sqrt{\sigma_{b0}^2 + 2t\rho\sigma_{b0}\sigma_{b1} + t^2\sigma_{b1}^2 + \frac{\sigma_\zeta^2}{1-\varphi^2}}\sqrt{\sigma_{b0}^2 + 2(t+m)\rho\sigma_{b0}\sigma_{b1} + (t+m)^2\sigma_{b1}^2 + \frac{\sigma_\zeta^2}{1-\varphi^2}}}.$$

R Code

First we confirm the relations expressed above

```r
library(MASS)

# Check variance and covariance of model 1, four time periods

# beta_0 = 100, beta_1 = 0, beta_2 = 2, beta_3 = 0.5,
# sigma_b0 = 15, sigma_e = 5, t =0,1,2,3

set.seed(1)
Model1Collect = matrix(NA, ncol = 1000000,nrow=4)
for(i in 1:1000000){
  Model1Collect[,i] = 100 + rnorm(1,0,15) +
    2*c(0,1,2,3) + rnorm(4,0,5)
}

Model1Var = matrix(c(round(var(Model1Collect[1,]),0),
                   round(cov(Model1Collect[1,],Model1Collect[2,]),0),
                   round(cov(Model1Collect[1,],Model1Collect[3,]),0),
                   round(cov(Model1Collect[1,],Model1Collect[4,]),0),
                   NA,
                   round(var(Model1Collect[2,]),0),
                   round(cov(Model1Collect[2,],Model1Collect[3,]),0),
                   round(cov(Model1Collect[2,],Model1Collect[4,]),0),
                   NA,NA,
                   round(var(Model1Collect[3,]),0),
                   round(cov(Model1Collect[3,],Model1Collect[4,]),0),
                   NA,NA,NA,
                   round(var(Model1Collect[4,]),0)),ncol=4)

15^2
15^2+5^2
Model1Var

# Agrees

### Check correlation
Model1Cor = matrix(c(round(var(Model1Collect[1,]),0),
                   round(cor(Model1Collect[1,],Model1Collect[2,]),2),
                   round(cor(Model1Collect[1,],Model1Collect[3,]),2),
                   round(cor(Model1Collect[1,],Model1Collect[4,]),2),
                   NA,
                   round(var(Model1Collect[2,]),0),
                   round(cor(Model1Collect[2,],Model1Collect[3,]),2),
                   round(cor(Model1Collect[2,],Model1Collect[4,]),2),
                   NA,NA,
                   round(var(Model1Collect[3,]),0),
                   round(cor(Model1Collect[3,],Model1Collect[4,]),2),
                   NA,NA,NA,
                   round(var(Model1Collect[4,]),0)),ncol=4)

(15^2)/(15^2+5^2)
Model1Cor
# Agrees


# Check variance and covariance of model 2, four time periods

# beta_0 = 100, beta_1 = 0, beta_2 = 2, beta_3 = 0.5,
# sigma_b0 = 15, sigma_e = 5, t =0,1,2,3, sigma_b1 = 0.25,rho = -0.7


set.seed(123)
Model2Collect = matrix(NA, ncol = 1000000,nrow=4)
for(i in 1:1000000){
  RE = mvrnorm(n = 1, c(0,0),
               matrix(c(15^2,-0.7*15*0.25,-0.7*15*0.25,0.25^2),ncol=2))
  Model2Collect[,i] = 100 + RE[1] +
    (2+RE[2])*c(0,1,2,3) + rnorm(4,0,5)
}

Model2Var = matrix(c(round(var(Model2Collect[1,]),0),
                   round(cov(Model2Collect[1,],Model2Collect[2,]),0),
```

```
                        round(cov(Model2Collect[1,],Model2Collect[3,]),0),
                        round(cov(Model2Collect[1,],Model2Collect[4,]),0),
                        NA,
                        round(var(Model2Collect[2,]),0),
                        round(cov(Model2Collect[2,],Model2Collect[3,]),0),
                        round(cov(Model2Collect[2,],Model2Collect[4,]),0),
                        NA,NA,
                        round(var(Model2Collect[3,]),0),
                        round(cov(Model2Collect[3,],Model2Collect[4,]),0),
                        NA,NA,NA,
                        round(var(Model2Collect[4,]),0)),ncol=4)

# VAR
15^2 + 5^2
15^2 -2*1*0.7*15*0.25 + 0.25^2 + 5^2
15^2 -2*2*0.7*15*0.25 + (2^2)*0.25^2 + 5^2
15^2 -2*3*0.7*15*0.25 + (3^2)*0.25^2 + 5^2

# COV
15^2 -1*0.7*15*0.25
15^2 -2*0.7*15*0.25
15^2 -3*0.7*15*0.25

15^2 -(1+2)*0.7*15*0.25 + 1*2*(0.25^2)
15^2 -(1+3)*0.7*15*0.25 + 1*3*(0.25^2)

15^2 -(2+3)*0.7*15*0.25 + 2*3*(0.25^2)

Model2Var

# Agrees

# Check correlation
Model2Cor = matrix(c(round(var(Model2Collect[1,]),0),
                     round(cor(Model2Collect[1,],Model2Collect[2,]),3),
                     round(cor(Model2Collect[1,],Model2Collect[3,]),3),
                     round(cor(Model2Collect[1,],Model2Collect[4,]),3),
                     NA,
                     round(var(Model2Collect[2,]),0),
                     round(cor(Model2Collect[2,],Model2Collect[3,]),3),
                     round(cor(Model2Collect[2,],Model2Collect[4,]),3),
                     NA,NA,
                     round(var(Model2Collect[3,]),0),
                     round(cor(Model2Collect[3,],Model2Collect[4,]),3),
                     NA,NA,NA,
                     round(var(Model2Collect[4,]),0)),ncol=4)

(15^2 -1*0.7*15*0.25)/(sqrt(15^2 + 5^2)*sqrt(15^2 -2*1*0.7*15*0.25 + 0.25^2 + 5^2))
(15^2 -2*0.7*15*0.25)/(sqrt(15^2 + 5^2)*sqrt(15^2 -2*2*0.7*15*0.25 + (2^2)*0.25^2 + 5^2))
(15^2 -3*0.7*15*0.25)/(sqrt(15^2 + 5^2)*sqrt(15^2 -2*3*0.7*15*0.25 + (3^2)*0.25^2 + 5^2))

(15^2 -(1+2)*0.7*15*0.25 + 1*2*(0.25^2))/
  (sqrt(15^2 -2*1*0.7*15*0.25 + 0.25^2 + 5^2)*sqrt(15^2 -2*2*0.7*15*0.25 + (2^2)*0.25^2 + 5^2))

(15^2 -(1+3)*0.7*15*0.25 + 1*3*(0.25^2))/
  (sqrt(15^2 -2*1*0.7*15*0.25 + 0.25^2 + 5^2)*sqrt(15^2 -2*3*0.7*15*0.25 + (3^2)*0.25^2 + 5^2))

(15^2 -(2+3)*0.7*15*0.25 + 2*3*(0.25^2))/
  (sqrt(15^2 -2*2*0.7*15*0.25 + (2^2)*0.25^2 + 5^2)*sqrt(15^2 -2*3*0.7*15*0.25 + (3^2)*0.25^2 + 5^2))


Model2Cor

# Agrees


# Check variance and covariance of model 3, four time periods

# beta_0 = 100, beta_1 = 0, beta_2 = 2, beta_3 = 0.5,
# sigma_b0 = 15, t =0,1,2,3, sigma_b1 = 0.25, rho = -0.7,
# phi = 0.5, zeta_ = 3.5

set.seed(123)
Model3Collect = matrix(NA, ncol = 1000000,nrow=4)
for(i in 1:1000000){
  RE = mvrnorm(n = 1, c(0,0),
               matrix(c(15^2,-0.7*15*0.25,-0.7*15*0.25,0.25^2),ncol=2))
```

```
  Model3 = 100 + RE[1] +
    (2+RE[2])*c(0,1,2,3)

  Model3AR10 = 0.5*rnorm(1,0,3.5) + rnorm(1,0,3.5)
  Model3AR11 = 0.5*Model3AR10 + rnorm(1,0,3.5)
  Model3AR12 = 0.5*Model3AR11 + rnorm(1,0,3.5)
  Model3AR13 = 0.5*Model3AR12 + rnorm(1,0,3.5)
  Model3Collect[,i] = Model3 + c(Model3AR10,Model3AR11,
                                  Model3AR12,Model3AR13)}


Model3Var = matrix(c(round(var(Model3Collect[1,]),0),
                    round(cov(Model3Collect[1,],Model3Collect[2,]),0),
                    round(cov(Model3Collect[1,],Model3Collect[3,]),0),
                    round(cov(Model3Collect[1,],Model3Collect[4,]),0),
                    NA,
                    round(var(Model3Collect[2,]),0),
                    round(cov(Model3Collect[2,],Model3Collect[3,]),0),
                    round(cov(Model3Collect[2,],Model3Collect[4,]),0),
                    NA,NA,
                    round(var(Model3Collect[3,]),0),
                    round(cov(Model3Collect[3,],Model3Collect[4,]),0),
                    NA,NA,NA,
                    round(var(Model3Collect[4,]),0)),ncol=4)


# VAR
15^2 + (3.5^2)/(1-0.25)
15^2 -2*1*0.7*15*0.25 + 0.25^2 + (3.5^2)/(1-0.25)
15^2 -2*2*0.7*15*0.25 + (2^2)*0.25^2 + (3.5^2)/(1-0.25)
15^2 -2*3*0.7*15*0.25 + (3^2)*0.25^2 + (3.5^2)/(1-0.25)

# COV
15^2 -1*0.7*15*0.25+0.5*((3.5^2)/(1-0.25))
15^2 -2*0.7*15*0.25+0.5*0.5*((3.5^2)/(1-0.25))
15^2 -3*0.7*15*0.25+0.5*0.5*0.5*((3.5^2)/(1-0.25))

15^2 -(1+2)*0.7*15*0.25 + 1*2*(0.25^2) +0.5*((3.5^2)/(1-0.25))
15^2 -(1+3)*0.7*15*0.25 + 1*3*(0.25^2) +0.5*0.5*((3.5^2)/(1-0.25))

15^2 -(2+3)*0.7*15*0.25 + 2*3*(0.25^2)+0.5*((3.5^2)/(1-0.25))

Model3Var

# Agreed

Model3Cor = matrix(c(round(var(Model3Collect[1,]),0),
                    round(cor(Model3Collect[1,],Model3Collect[2,]),3),
                    round(cor(Model3Collect[1,],Model3Collect[3,]),3),
                    round(cor(Model3Collect[1,],Model3Collect[4,]),3),
                    NA,
                    round(var(Model3Collect[2,]),0),
                    round(cor(Model3Collect[2,],Model3Collect[3,]),3),
                    round(cor(Model3Collect[2,],Model3Collect[4,]),3),
                    NA,NA,
                    round(var(Model3Collect[3,]),0),
                    round(cor(Model3Collect[3,],Model3Collect[4,]),3),
                    NA,NA,NA,
                    round(var(Model3Collect[4,]),0)),ncol=4)

(15^2 -1*0.7*15*0.25+0.5*((3.5^2)/(1-0.25)))/
  (sqrt(15^2 + (3.5^2)/(1-0.25))*sqrt(15^2 -2*1*0.7*15*0.25 + 0.25^2 + (3.5^2)/(1-0.25)))
(15^2 -2*0.7*15*0.25+0.5*0.5*((3.5^2)/(1-0.25)))/
  (sqrt(15^2 + (3.5^2)/(1-0.25))*sqrt(15^2 -2*2*0.7*15*0.25 + (2^2)*0.25^2 + (3.5^2)/(1-0.25)))
(15^2 -3*0.7*15*0.25+0.5*0.5*0.5*((3.5^2)/(1-0.25)))/
  (sqrt(15^2 + (3.5^2)/(1-0.25))*sqrt(15^2 -2*3*0.7*15*0.25 + (3^2)*0.25^2 + (3.5^2)/(1-0.25)))


(15^2 -(1+2)*0.7*15*0.25 + 1*2*(0.25^2) +0.5*((3.5^2)/(1-0.25)))/
  (sqrt(15^2 -2*1*0.7*15*0.25 + 0.25^2 + (3.5^2)/(1-0.25))*sqrt(15^2 -2*2*0.7*15*0.25 + (2^2)*0.25^2 + (3.5^2)/(1-0.25)))
(15^2 -(1+3)*0.7*15*0.25 + 1*3*(0.25^2) +0.5*0.5*((3.5^2)/(1-0.25)))/
  (sqrt(15^2 -2*1*0.7*15*0.25 + 0.25^2 + (3.5^2)/(1-0.25))*sqrt(15^2 -2*3*0.7*15*0.25 + (3^2)*0.25^2 + (3.5^2)/(1-0.25)))


(15^2 -(2+3)*0.7*15*0.25 + 2*3*(0.25^2)+0.5*((3.5^2)/(1-0.25)))/
  (sqrt(15^2 -2*2*0.7*15*0.25 + (2^2)*0.25^2 + (3.5^2)/(1-0.25))*sqrt(15^2 -2*3*0.7*15*0.25 + (3^2)*0.25^2 + (3.5^2)/(1-
0.25)))
```

Model3Cor

Agreed

Code for the simulations

```r
library(MASS)
library(ggplot2)
library(tidyr)
library(dplyr)
library(scales)
library(nlme)
library(foreach)
library(doParallel)
library(Hmisc)
library(cowplot)
# https://www.java.com/en/download/
library(rJava)
library(xlsx)
library("readxl")
library(ggpubr)

BaseSimulationTime = function(n,sigmab0,sigmab1,beta0,beta2,beta3,rho,t,Iter){

  Datalmm = array(NA, c(2*n,length(t),Iter))
  G = c(rep(0,n),rep(1,n))
  for(i in 1:Iter){
    U = mvrnorm(2*n,c(0,0),
                matrix(c(sigmab0^2,rho*sigmab0*sigmab1,rho*sigmab0*sigmab1,sigmab1^2),ncol=2))
    for(j in 1:length(t)){
      Datalmm[,j,i] = beta0 + U[,1] + t[j]*(beta2 + G*beta3 + U[,2])}}
  return(Datalmm)}

BaseSimulationError = function(Data,sigma,ar){
  rown = length(Data[,1,1])
  coln = length(Data[1,,1])
  slicen = length(Data[1,1,])
  DatalmmError = array(NA, c(rown,coln,slicen))
  zeta = sqrt((sigma^2)*(1-ar^2))
  for(i in 1:slicen){
    errorsigma = matrix(rnorm(rown*coln,0,zeta),nrow=rown,ncol=coln)
    DatalmmError[,1,i] = errorsigma[,1]
    for(j in 2:coln){
      DatalmmError[,j,i]  = ar*DatalmmError[,(j-1),i] + errorsigma[,j]
    }}
  DataE = Data + DatalmmError
  return(DataE)}

PowerAnalysis = function(Data,n1,n2,n3,n4,n,t){
  Iter = length(Data[1,1,])
  nt = length(Data[1,,1])
  Collection = matrix(NA,nrow=Iter,ncol=5)

  n1FullIDs = c(seq(1,n1,1),seq(n+1,(n+n1),1))
  n2FullIDs = c(seq(1,n2,1),seq(n+1,(n+n2),1))
  n3FullIDs = c(seq(1,n3,1),seq(n+1,(n+n3),1))
  n4FullIDs = c(seq(1,n4,1),seq(n+1,(n+n4),1))

  for(i in 1:Iter){
    DEDF = as.data.frame(Data[,,i])
    DEDFL = as.data.frame(pivot_longer(DEDF, cols = everything()))
    DEDFL$ID = rep(1:(2*n),each=nt)
    DEDFL$Group = c(rep("Standard",n*nt),rep("Test",n*nt))
    DEDFL$Time = rep(t,2*n)
    DEDFL$ID = factor(DEDFL$ID)
    DEDFL$Group = factor(DEDFL$Group)

    # All participants once per week
    n1Mod = lme(value~Group+Group*Time,random=~Time|ID,
              control = lmeControl(opt = 'optim'),
              data=DEDFL[DEDFL$ID %in% n1FullIDs,])
```

```
        n2Mod = lme(value~Group+Group*Time,random=~Time|ID,
                    control = lmeControl(opt = 'optim'),
                    data=DEDFL[DEDFL$ID %in% n2FullIDs,])

        n3Mod = lme(value~Group+Group*Time,random=~Time|ID,
                    control = lmeControl(opt = 'optim'),
                    data=DEDFL[DEDFL$ID %in% n3FullIDs,])

        n4Mod = lme(value~Group+Group*Time,random=~Time|ID,
                    control = lmeControl(opt = 'optim'),
                    data=DEDFL[DEDFL$ID %in% n4FullIDs,])

        nMod = lme(value~Group+Group*Time,random=~Time|ID,
                   control = lmeControl(opt = 'optim'),
                   data=DEDFL)

      n1ModS = summary(n1Mod)
      n2ModS = summary(n2Mod)
      n3ModS = summary(n3Mod)
      n4ModS = summary(n4Mod)
      nModS = summary(nMod)

      Collection[i,1] =as.numeric(n1ModS$tTable[4,5])
      Collection[i,2] =as.numeric(n2ModS$tTable[4,5])
      Collection[i,3] =as.numeric(n3ModS$tTable[4,5])
      Collection[i,4] =as.numeric(n4ModS$tTable[4,5])
      Collection[i,5] =as.numeric(nModS$tTable[4,5])}
    return(Collection)}

PowerAnalysisP = function(Data,n1,n2,n3,n4,n,t){
  nt = length(Data[1,])
  Collection = c(NULL)

  n1FullIDs = c(seq(1,n1,1),seq(n+1,(n+n1),1))
  n2FullIDs = c(seq(1,n2,1),seq(n+1,(n+n2),1))
  n3FullIDs = c(seq(1,n3,1),seq(n+1,(n+n3),1))
  n4FullIDs = c(seq(1,n4,1),seq(n+1,(n+n4),1))

  DEDF = as.data.frame(Data)
  DEDFL = as.data.frame(pivot_longer(DEDF, cols = everything()))
  DEDFL$ID = rep(1:(2*n),each=nt)
  DEDFL$Group = c(rep("Standard",n*nt),rep("Test",n*nt))
  DEDFL$Time = rep(t,2*n)
  DEDFL$ID = factor(DEDFL$ID)
  DEDFL$Group = factor(DEDFL$Group)

  # All participants once per week
  n1Mod = lme(value~Group+Group*Time,random=~Time|ID,
              control = lmeControl(opt = 'optim'),
              data=DEDFL[DEDFL$ID %in% n1FullIDs,])


  n2Mod = lme(value~Group+Group*Time,random=~Time|ID,
              control = lmeControl(opt = 'optim'),
              data=DEDFL[DEDFL$ID %in% n2FullIDs,])

  n3Mod = lme(value~Group+Group*Time,random=~Time|ID,
              control = lmeControl(opt = 'optim'),
              data=DEDFL[DEDFL$ID %in% n3FullIDs,])

  n4Mod = lme(value~Group+Group*Time,random=~Time|ID,
              control = lmeControl(opt = 'optim'),
              data=DEDFL[DEDFL$ID %in% n4FullIDs,])

  nMod = lme(value~Group+Group*Time,random=~Time|ID,
             control = lmeControl(opt = 'optim'),
             data=DEDFL)

  n1ModS = summary(n1Mod)
  n2ModS = summary(n2Mod)
  n3ModS = summary(n3Mod)
  n4ModS = summary(n4Mod)
  nModS = summary(nMod)

  Collection[1] =as.numeric(n1ModS$tTable[4,5])
  Collection[2] =as.numeric(n2ModS$tTable[4,5])
```

```
  Collection[3] =as.numeric(n3ModS$tTable[4,5])
  Collection[4] =as.numeric(n4ModS$tTable[4,5])
  Collection[5] =as.numeric(nModS$tTable[4,5])
  return(Collection)}

PowerOUt = function(DF){
  DF1=t(as.data.frame(DF))
  rownames(DF1)=c(NULL)
  Power = c(mean(DF1[,1]<0.05),mean(DF1[,2]<0.05),
            mean(DF1[,3]<0.05),mean(DF1[,4]<0.05),
            mean(DF1[,5]<0.05))
  return(Power)
}

n.cores = parallel::detectCores() - 2
my.cluster = parallel::makeCluster(
  n.cores,
  type = "PSOCK"
)
doParallel::registerDoParallel(cl = my.cluster)
# parallel::stopCluster(cl = my.cluster)

SimulationAnalysisP = function(Data,nfull,nhalf,Weeks){
  np = length(Data[,1])/2
  nt = length(Data[1,])
  Collection = c(NULL)
  n = nhalf[1]
  # IDs of participants in analysis across whole intervention
  n1FullIDs = c(seq(1,nfull[1],1),seq(n+1,(n+nfull[1]),1))
  n2FullIDs = c(seq(1,nfull[2],1),seq(n+1,(n+nfull[2]),1))
  n3FullIDs = c(seq(1,nfull[3],1),seq(n+1,(n+nfull[3]),1))


  # IDs of participants in analysis across half intervention
  n1HalfIDs = c(seq(1,nhalf[1],1),seq(n+1,(n+nhalf[1]),1))
  n2HalfIDs = c(seq(1,nhalf[2],1),seq(n+1,(n+nhalf[2]),1))
  n3HalfIDs = c(seq(1,nhalf[3],1),seq(n+1,(n+nhalf[3]),1))

  # Full-Intervention
  T1Full = round(seq(0,Weeks,1),2)
  T2Full = round(c(seq(0,Weeks,1),seq(3/7,Weeks,1)),2)
  T3Full = round(c(seq(0,Weeks,1),seq(2/7,Weeks,1),
                   seq(4/7,Weeks,1)),2)
  # Time to match the data created
  TFull = round(c(seq(0,Weeks,1),seq(2/7,Weeks,1),seq(3/7,Weeks,1),
                  seq(4/7,Weeks,1)),2)

  # Half-Intervention
  T1Half = round(seq(0,Weeks/2,1),2)
  T2Half = round(c(seq(0,Weeks/2,1),seq(3/7,Weeks/2,1)),2)
  T3Half = round(c(seq(0,Weeks/2,1),seq(2/7,Weeks/2,1),
                   seq(4/7,Weeks/2,1)),2)

  DEDF = as.data.frame(Data)
  DEDFL = as.data.frame(pivot_longer(DEDF, cols = everything()))
  DEDFL$ID = rep(1:(2*np),each=nt)
  DEDFL$Group = c(rep("Standard",np*nt),rep("Test",np*nt))
  DEDFL$Time = rep(TFull,2*np)
  DEDFL$ID = factor(DEDFL$ID)
  DEDFL$Group = factor(DEDFL$Group)

  #### Half-Intervention Models


  # All participants once per week
  Mod1Half1 = tryCatch({ lme(value~Group+Group*Time,random=~1|ID,
                             control = lmeControl(opt = 'optim'),
                             data=DEDFL[(DEDFL$Time %in% T1Half)&(DEDFL$ID %in% n1HalfIDs),])},
                        error=function(e) 3)
  Mod2Half1 = tryCatch({ lme(value~Group+Group*Time,random=~Time|ID,
                             control = lmeControl(opt = 'optim'),
                             data=DEDFL[(DEDFL$Time %in% T1Half)&(DEDFL$ID %in% n1HalfIDs),])},
                        error=function(e) 3)
  Mod3Half1 = tryCatch({ lme(value~Group+Group*Time,random=~Time|ID,correlation=corAR1(),
                             control = lmeControl(opt = 'optim'),
                             data=DEDFL[(DEDFL$Time %in% T1Half)&(DEDFL$ID %in% n1HalfIDs),])},
                        error=function(e) 3)
```

```
if(is.list(Mod1Half1)==TRUE) {Mod1Half1S =summary(Mod1Half1)}
if(is.list(Mod1Half1)==FALSE) {Mod1Half1S =NA}

if(is.list(Mod2Half1)==TRUE) {Mod2Half1S =summary(Mod2Half1)}
if(is.list(Mod2Half1)==FALSE) {Mod2Half1S =NA}

if(is.list(Mod3Half1)==TRUE) {Mod3Half1S =summary(Mod3Half1)}
if(is.list(Mod3Half1)==FALSE) {Mod3Half1S =NA}


# n2half participants twice per week
Mod1Half2 = tryCatch({ lme(value~Group+Group*Time,random=~1|ID,
                            control = lmeControl(opt = 'optim'),
                            data=DEDFL[(DEDFL$Time %in% T2Half)&(DEDFL$ID %in% n2HalfIDs),])},
                     error=function(e) 3)
Mod2Half2 = tryCatch({ lme(value~Group+Group*Time,random=~Time|ID,
                            control = lmeControl(opt = 'optim'),
                            data=DEDFL[(DEDFL$Time %in% T2Half)&(DEDFL$ID %in% n2HalfIDs),])},
                     error=function(e) 3)
Mod3Half2 = tryCatch({ lme(value~Group+Group*Time,random=~Time|ID,correlation=corAR1(),
                            control = lmeControl(opt = 'optim'),
                            data=DEDFL[(DEDFL$Time %in% T2Half)&(DEDFL$ID %in% n2HalfIDs),])},
                     error=function(e) 3)

if(is.list(Mod1Half2)==TRUE) {Mod1Half2S =summary(Mod1Half2)}
if(is.list(Mod1Half2)==FALSE) {Mod1Half2S =NA}

if(is.list(Mod2Half2)==TRUE) {Mod2Half2S =summary(Mod2Half2)}
if(is.list(Mod2Half2)==FALSE) {Mod2Half2S =NA}

if(is.list(Mod3Half2)==TRUE) {Mod3Half2S =summary(Mod3Half2)}
if(is.list(Mod3Half2)==FALSE) {Mod3Half2S =NA}


# n3half participants three times per week
Mod1Half3 = tryCatch({ lme(value~Group+Group*Time,random=~1|ID,
                            control = lmeControl(opt = 'optim'),
                            data=DEDFL[(DEDFL$Time %in% T3Half)&(DEDFL$ID %in% n3HalfIDs),])},
                     error=function(e) 3)
Mod2Half3 = tryCatch({ lme(value~Group+Group*Time,random=~Time|ID,
                            control = lmeControl(opt = 'optim'),
                            data=DEDFL[(DEDFL$Time %in% T3Half)&(DEDFL$ID %in% n3HalfIDs),])},
                     error=function(e) 3)
Mod3Half3 = tryCatch({ lme(value~Group+Group*Time,random=~Time|ID,correlation=corAR1(),
                            control = lmeControl(opt = 'optim'),
                            data=DEDFL[(DEDFL$Time %in% T3Half)&(DEDFL$ID %in% n3HalfIDs),])},
                     error=function(e) 3)

if(is.list(Mod1Half3)==TRUE) {Mod1Half3S =summary(Mod1Half3)}
if(is.list(Mod1Half3)==FALSE) {Mod1Half3S =NA}

if(is.list(Mod2Half3)==TRUE) {Mod2Half3S =summary(Mod2Half3)}
if(is.list(Mod2Half3)==FALSE) {Mod2Half3S =NA}

if(is.list(Mod3Half3)==TRUE) {Mod3Half3S =summary(Mod3Half3)}
if(is.list(Mod3Half3)==FALSE) {Mod3Half3S =NA}


#### Full-Intervention Models

# n1full participants once per week

Mod1Full1 = tryCatch({ lme(value~Group+Group*Time,random=~1|ID,
                            control = lmeControl(opt = 'optim'),
                            data=DEDFL[(DEDFL$Time %in% T1Full)&(DEDFL$ID %in% n1FullIDs),])},
                     error=function(e) 3)
Mod2Full1 = tryCatch({ lme(value~Group+Group*Time,random=~Time|ID,
                            control = lmeControl(opt = 'optim'),
                            data=DEDFL[(DEDFL$Time %in% T1Full)&(DEDFL$ID %in% n1FullIDs),])},
                     error=function(e) 3)
Mod3Full1 = tryCatch({ lme(value~Group+Group*Time,random=~Time|ID,correlation=corAR1(),
                            control = lmeControl(opt = 'optim'),
                            data=DEDFL[(DEDFL$Time %in% T1Full)&(DEDFL$ID %in% n1FullIDs),])},
                     error=function(e) 3)

if(is.list(Mod1Full1)==TRUE) {Mod1Full1S =summary(Mod1Full1)}
```

```
if(is.list(Mod1Full1)==FALSE) {Mod1Full1S =NA}

if(is.list(Mod2Full1)==TRUE) {Mod2Full1S =summary(Mod2Full1)}
if(is.list(Mod2Full1)==FALSE) {Mod2Full1S =NA}

if(is.list(Mod3Full1)==TRUE) {Mod3Full1S =summary(Mod3Full1)}
if(is.list(Mod3Full1)==FALSE) {Mod3Full1S =NA}

# n2full participants twice per week
Mod1Full2 = tryCatch({ lme(value~Group+Group*Time,random=~1|ID,
                        control = lmeControl(opt = 'optim'),
                        data=DEDFL[(DEDFL$Time %in% T2Full)&(DEDFL$ID %in% n2FullIDs),])},
                    error=function(e) 3)
Mod2Full2 = tryCatch({ lme(value~Group+Group*Time,random=~Time|ID,
                        control = lmeControl(opt = 'optim'),
                        data=DEDFL[(DEDFL$Time %in% T2Full)&(DEDFL$ID %in% n2FullIDs),])},
                    error=function(e) 3)
Mod3Full2 = tryCatch({ lme(value~Group+Group*Time,random=~Time|ID,correlation=corAR1(),
                        control = lmeControl(opt = 'optim'),
                        data=DEDFL[(DEDFL$Time %in% T2Full)&(DEDFL$ID %in% n2FullIDs),])},
                    error=function(e) 3)

if(is.list(Mod1Full2)==TRUE) {Mod1Full2S =summary(Mod1Full2)}
if(is.list(Mod1Full2)==FALSE) {Mod1Full2S =NA}

if(is.list(Mod2Full2)==TRUE) {Mod2Full2S =summary(Mod2Full2)}
if(is.list(Mod2Full2)==FALSE) {Mod2Full2S =NA}

if(is.list(Mod3Full2)==TRUE) {Mod3Full2S =summary(Mod3Full2)}
if(is.list(Mod3Full2)==FALSE) {Mod3Full2S =NA}

# n3full participants four times per week
Mod1Full3 = tryCatch({ lme(value~Group+Group*Time,random=~1|ID,
                        control = lmeControl(opt = 'optim'),
                        data=DEDFL[(DEDFL$Time %in% T3Full)&(DEDFL$ID %in% n3FullIDs),])},
                    error=function(e) 3)
Mod2Full3 = tryCatch({ lme(value~Group+Group*Time,random=~Time|ID,
                        control = lmeControl(opt = 'optim'),
                        data=DEDFL[(DEDFL$Time %in% T3Full)&(DEDFL$ID %in% n3FullIDs),])},
                    error=function(e) 3)
Mod3Full3 = tryCatch({ lme(value~Group+Group*Time,random=~Time|ID,correlation=corAR1(),
                        control = lmeControl(opt = 'optim'),
                        data=DEDFL[(DEDFL$Time %in% T3Full)&(DEDFL$ID %in% n3FullIDs),])},
                    error=function(e) 3)

if(is.list(Mod1Full3)==TRUE) {Mod1Full3S =summary(Mod1Full3)}
if(is.list(Mod1Full3)==FALSE) {Mod1Full3S =NA}

if(is.list(Mod2Full3)==TRUE) {Mod2Full3S =summary(Mod2Full3)}
if(is.list(Mod2Full3)==FALSE) {Mod2Full3S =NA}

if(is.list(Mod3Full3)==TRUE) {Mod3Full3S =summary(Mod3Full3)}
if(is.list(Mod3Full3)==FALSE) {Mod3Full3 =NA}


Collection = c(

  c(if(!is.na(Mod1Half1S[[1]][1])==TRUE)
  {c(as.numeric(Mod1Half1S$tTable[1:4,1]),as.numeric(Mod1Half1S$tTable[1:4,2]),
    as.numeric(Mod1Half1S$tTable[1:4,5]), as.numeric(VarCorr(Mod1Half1S)[2,2]),
    as.numeric(VarCorr(Mod1Half1S)[1,2]),Mod1Half1S$AIC,Mod1Half1S$BIC,Mod1Half1S$logLik)},

  if(is.na(Mod1Half1S[[1]][1])==TRUE) {rep(NA,17)}),


  c(if(!is.na(Mod2Half1S[[1]][1])==TRUE)
  {c(as.numeric(Mod2Half1S$tTable[1:4,1]),as.numeric(Mod2Half1S$tTable[1:4,2]),
    as.numeric(Mod2Half1S$tTable[1:4,5]),
    as.numeric(VarCorr(Mod2Half1S)[3,2]),as.numeric(VarCorr(Mod2Half1S)[1,2]),
    as.numeric(VarCorr(Mod2Half1S)[2,2]),as.numeric(VarCorr(Mod2Half1S)[2,3]),
    Mod2Half1S$AIC,Mod2Half1S$BIC,Mod2Half1S$logLik)},

  if(is.na(Mod2Half1S[[1]][1])==TRUE) {rep(NA,19)}),

  c(if(!is.na(Mod3Half1S[[1]][1])==TRUE)
  {c(as.numeric(Mod3Half1S$tTable[1:4,1]),as.numeric(Mod3Half1S$tTable[1:4,2]),
    as.numeric(Mod3Half1S$tTable[1:4,5]),
```

```
      as.numeric(VarCorr(Mod3Half1S)[3,2]),as.numeric(VarCorr(Mod3Half1S)[1,2]),
      as.numeric(VarCorr(Mod3Half1S)[2,2]),as.numeric(VarCorr(Mod3Half1S)[2,3]),
      as.numeric(coef(Mod3Half1$modelStruct$corStruct, unconstrained = FALSE)),
      Mod3Half1S$AIC,Mod3Half1S$BIC,Mod3Half1S$logLik)},

  if(is.na(Mod3Half1S[[1]][1])==TRUE) {rep(NA,20)}),

  c(if(!is.na(Mod1Half1S[[1]][1])==TRUE&!is.na(Mod2Half1S[[1]][1])==TRUE)
  {anova(Mod1Half1,Mod2Half1)$`p-value`[2]},
  if(is.na(Mod1Half1S[[1]][1])==TRUE|is.na(Mod2Half1S[[1]][1])==TRUE)
  {NA}),

  c(if(!is.na(Mod1Half1S[[1]][1])==TRUE&!is.na(Mod3Half1S[[1]][1])==TRUE)
  {anova(Mod1Half1,Mod3Half1)$`p-value`[2]},
  if(is.na(Mod1Half1S[[1]][1])==TRUE|is.na(Mod3Half1S[[1]][1])==TRUE)
  {NA}),

  c(if(!is.na(Mod2Half1S[[1]][1])==TRUE&!is.na(Mod3Half1S[[1]][1])==TRUE)
  {anova(Mod2Half1,Mod3Half1)$`p-value`[2]},
  if(is.na(Mod2Half1S[[1]][1])==TRUE|is.na(Mod3Half1S[[1]][1])==TRUE)
  {NA}),

  c(if(!is.na(Mod1Half2S[[1]][1])==TRUE)
  {c(as.numeric(Mod1Half2S$tTable[1:4,1]),as.numeric(Mod1Half2S$tTable[1:4,2]),
     as.numeric(Mod1Half2S$tTable[1:4,5]), as.numeric(VarCorr(Mod1Half2S)[2,2]),
     as.numeric(VarCorr(Mod1Half2S)[1,2]),Mod1Half2S$AIC,Mod1Half2S$BIC,Mod1Half2S$logLik)},

  if(is.na(Mod1Half2S[[1]][1])==TRUE) {rep(NA,17)}),


  c(if(!is.na(Mod2Half2S[[1]][1])==TRUE)
  {c(as.numeric(Mod2Half2S$tTable[1:4,1]),as.numeric(Mod2Half2S$tTable[1:4,2]),
     as.numeric(Mod2Half2S$tTable[1:4,5]),
     as.numeric(VarCorr(Mod2Half2S)[3,2]),as.numeric(VarCorr(Mod2Half2S)[1,2]),
     as.numeric(VarCorr(Mod2Half2S)[2,2]),as.numeric(VarCorr(Mod2Half2S)[2,3]),
     Mod2Half2S$AIC,Mod2Half2S$BIC,Mod2Half2S$logLik)},

  if(is.na(Mod2Half2S[[1]][1])==TRUE) {rep(NA,19)}),

  c(if(!is.na(Mod3Half2S[[1]][1])==TRUE)
  {c(as.numeric(Mod3Half2S$tTable[1:4,1]),as.numeric(Mod3Half2S$tTable[1:4,2]),
     as.numeric(Mod3Half2S$tTable[1:4,5]),
     as.numeric(VarCorr(Mod3Half2S)[3,2]),as.numeric(VarCorr(Mod3Half2S)[1,2]),
     as.numeric(VarCorr(Mod3Half2S)[2,2]),as.numeric(VarCorr(Mod3Half2S)[2,3]),
     as.numeric(coef(Mod3Half2$modelStruct$corStruct, unconstrained = FALSE)),
     Mod3Half2S$AIC,Mod3Half2S$BIC,Mod3Half2S$logLik)},

  if(is.na(Mod3Half2S[[1]][1])==TRUE) {rep(NA,20)}),

  c(if(!is.na(Mod1Half2S[[1]][1])==TRUE&!is.na(Mod2Half2S[[1]][1])==TRUE)
  {anova(Mod1Half2,Mod2Half2)$`p-value`[2]},
  if(is.na(Mod1Half2S[[1]][1])==TRUE|is.na(Mod2Half2S[[1]][1])==TRUE)
  {NA}),

  c(if(!is.na(Mod1Half2S[[1]][1])==TRUE&!is.na(Mod3Half2S[[1]][1])==TRUE)
  {anova(Mod1Half2,Mod3Half2)$`p-value`[2]},
  if(is.na(Mod1Half2S[[1]][1])==TRUE|is.na(Mod3Half2S[[1]][1])==TRUE)
  {NA}),

  c(if(!is.na(Mod2Half2S[[1]][1])==TRUE&!is.na(Mod3Half2S[[1]][1])==TRUE)
  {anova(Mod2Half2,Mod3Half2)$`p-value`[2]},
  if(is.na(Mod2Half2S[[1]][1])==TRUE|is.na(Mod3Half2S[[1]][1])==TRUE)
  {NA}),

  c(if(!is.na(Mod1Half3S[[1]][1])==TRUE)
  {c(as.numeric(Mod1Half3S$tTable[1:4,1]),as.numeric(Mod1Half3S$tTable[1:4,2]),
     as.numeric(Mod1Half3S$tTable[1:4,5]), as.numeric(VarCorr(Mod1Half3S)[2,2]),
     as.numeric(VarCorr(Mod1Half3S)[1,2]),Mod1Half3S$AIC,Mod1Half3S$BIC,Mod1Half3S$logLik)},

  if(is.na(Mod1Half3S[[1]][1])==TRUE) {rep(NA,17)}),


  c(if(!is.na(Mod2Half3S[[1]][1])==TRUE)
  {c(as.numeric(Mod2Half3S$tTable[1:4,1]),as.numeric(Mod2Half3S$tTable[1:4,2]),
     as.numeric(Mod2Half3S$tTable[1:4,5]),
     as.numeric(VarCorr(Mod2Half3S)[3,2]),as.numeric(VarCorr(Mod2Half3S)[1,2]),
     as.numeric(VarCorr(Mod2Half3S)[2,2]),as.numeric(VarCorr(Mod2Half3S)[2,3]),
```

```
        Mod2Half3S$AIC,Mod2Half3S$BIC,Mod2Half3S$logLik)},

    if(is.na(Mod2Half3S[[1]][1])==TRUE) {rep(NA,19)}),

    c(if(!is.na(Mod3Half3S[[1]][1])==TRUE)
    {c(as.numeric(Mod3Half3S$tTable[1:4,1]),as.numeric(Mod3Half3S$tTable[1:4,2]),
        as.numeric(Mod3Half3S$tTable[1:4,5]),
        as.numeric(VarCorr(Mod3Half3S)[3,2]),as.numeric(VarCorr(Mod3Half3S)[1,2]),
        as.numeric(VarCorr(Mod3Half3S)[2,2]),as.numeric(VarCorr(Mod3Half3S)[2,3]),
        as.numeric(coef(Mod3Half3$modelStruct$corStruct, unconstrained = FALSE)),
        Mod3Half3S$AIC,Mod3Half3S$BIC,Mod3Half3S$logLik)},

    if(is.na(Mod3Half3S[[1]][1])==TRUE) {rep(NA,20)}),

    c(if(!is.na(Mod1Half3S[[1]][1])==TRUE&!is.na(Mod2Half3S[[1]][1])==TRUE)
    {anova(Mod1Half3,Mod2Half3)$`p-value`[2]},
    if(is.na(Mod1Half3S[[1]][1])==TRUE|is.na(Mod2Half3S[[1]][1])==TRUE)
    {NA}),

    c(if(!is.na(Mod1Half3S[[1]][1])==TRUE&!is.na(Mod3Half3S[[1]][1])==TRUE)
    {anova(Mod1Half3,Mod3Half3)$`p-value`[2]},
    if(is.na(Mod1Half3S[[1]][1])==TRUE|is.na(Mod3Half3S[[1]][1])==TRUE)
    {NA}),

    c(if(!is.na(Mod2Half3S[[1]][1])==TRUE&!is.na(Mod3Half3S[[1]][1])==TRUE)
    {anova(Mod2Half3,Mod3Half3)$`p-value`[2]},
    if(is.na(Mod2Half3S[[1]][1])==TRUE|is.na(Mod3Half3S[[1]][1])==TRUE)
    {NA}),

    c(if(!is.na(Mod1Full1S[[1]][1])==TRUE)
    {c(as.numeric(Mod1Full1S$tTable[1:4,1]),as.numeric(Mod1Full1S$tTable[1:4,2]),
        as.numeric(Mod1Full1S$tTable[1:4,5]), as.numeric(VarCorr(Mod1Full1S)[2,2]),
        as.numeric(VarCorr(Mod1Full1S)[1,2]),Mod1Full1S$AIC,Mod1Full1S$BIC,Mod1Full1S$logLik)},

    if(is.na(Mod1Full1S[[1]][1])==TRUE) {rep(NA,17)}),


    c(if(!is.na(Mod2Full1S[[1]][1])==TRUE)
    {c(as.numeric(Mod2Full1S$tTable[1:4,1]),as.numeric(Mod2Full1S$tTable[1:4,2]),
        as.numeric(Mod2Full1S$tTable[1:4,5]),
        as.numeric(VarCorr(Mod2Full1S)[3,2]),as.numeric(VarCorr(Mod2Full1S)[1,2]),
        as.numeric(VarCorr(Mod2Full1S)[2,2]),as.numeric(VarCorr(Mod2Full1S)[2,3]),
        Mod2Full1S$AIC,Mod2Full1S$BIC,Mod2Full1S$logLik)},

    if(is.na(Mod2Full1S[[1]][1])==TRUE) {rep(NA,19)}),

    c(if(!is.na(Mod3Full1S[[1]][1])==TRUE)
    {c(as.numeric(Mod3Full1S$tTable[1:4,1]),as.numeric(Mod3Full1S$tTable[1:4,2]),
        as.numeric(Mod3Full1S$tTable[1:4,5]),
        as.numeric(VarCorr(Mod3Full1S)[3,2]),as.numeric(VarCorr(Mod3Full1S)[1,2]),
        as.numeric(VarCorr(Mod3Full1S)[2,2]),as.numeric(VarCorr(Mod3Full1S)[2,3]),
        as.numeric(coef(Mod3Full1$modelStruct$corStruct, unconstrained = FALSE)),
        Mod3Full1S$AIC,Mod3Full1S$BIC,Mod3Full1S$logLik)},

    if(is.na(Mod3Full1S[[1]][1])==TRUE) {rep(NA,20)}),

    c(if(!is.na(Mod1Full1S[[1]][1])==TRUE&!is.na(Mod2Full1S[[1]][1])==TRUE)
    {anova(Mod1Full1,Mod2Full1)$`p-value`[2]},
    if(is.na(Mod1Full1S[[1]][1])==TRUE|is.na(Mod2Full1S[[1]][1])==TRUE)
    {NA}),

    c(if(!is.na(Mod1Full1S[[1]][1])==TRUE&!is.na(Mod3Full1S[[1]][1])==TRUE)
    {anova(Mod1Full1,Mod3Full1)$`p-value`[2]},
    if(is.na(Mod1Full1S[[1]][1])==TRUE|is.na(Mod3Full1S[[1]][1])==TRUE)
    {NA}),

    c(if(!is.na(Mod2Full1S[[1]][1])==TRUE&!is.na(Mod3Full1S[[1]][1])==TRUE)
    {anova(Mod2Full1,Mod3Full1)$`p-value`[2]},
    if(is.na(Mod2Full1S[[1]][1])==TRUE|is.na(Mod3Full1S[[1]][1])==TRUE)
    {NA}),

    c(if(!is.na(Mod1Full2S[[1]][1])==TRUE)
    {c(as.numeric(Mod1Full2S$tTable[1:4,1]),as.numeric(Mod1Full2S$tTable[1:4,2]),
        as.numeric(Mod1Full2S$tTable[1:4,5]), as.numeric(VarCorr(Mod1Full2S)[2,2]),
        as.numeric(VarCorr(Mod1Full2S)[1,2]),Mod1Full2S$AIC,Mod1Full2S$BIC,Mod1Full2S$logLik)},

    if(is.na(Mod1Full2S[[1]][1])==TRUE) {rep(NA,17)}),
```

```
c(if(!is.na(Mod2Full2S[[1]][1])==TRUE)
{c(as.numeric(Mod2Full2S$tTable[1:4,1]),as.numeric(Mod2Full2S$tTable[1:4,2]),
   as.numeric(Mod2Full2S$tTable[1:4,5]),
   as.numeric(VarCorr(Mod2Full2S)[3,2]),as.numeric(VarCorr(Mod2Full2S)[1,2]),
   as.numeric(VarCorr(Mod2Full2S)[2,2]),as.numeric(VarCorr(Mod2Full2S)[2,3]),
   Mod2Full2S$AIC,Mod2Full2S$BIC,Mod2Full2S$logLik)},

if(is.na(Mod2Full2S[[1]][1])==TRUE) {rep(NA,19)}),

c(if(!is.na(Mod3Full2S[[1]][1])==TRUE)
{c(as.numeric(Mod3Full2S$tTable[1:4,1]),as.numeric(Mod3Full2S$tTable[1:4,2]),
   as.numeric(Mod3Full2S$tTable[1:4,5]),
   as.numeric(VarCorr(Mod3Full2S)[3,2]),as.numeric(VarCorr(Mod3Full2S)[1,2]),
   as.numeric(VarCorr(Mod3Full2S)[2,2]),as.numeric(VarCorr(Mod3Full2S)[2,3]),
   as.numeric(coef(Mod3Full2$modelStruct$corStruct, unconstrained = FALSE)),
   Mod3Full2S$AIC,Mod3Full2S$BIC,Mod3Full2S$logLik)},

if(is.na(Mod3Full2S[[1]][1])==TRUE) {rep(NA,20)}),

c(if(!is.na(Mod1Full2S[[1]][1])==TRUE&!is.na(Mod2Full2S[[1]][1])==TRUE)
{anova(Mod1Full2,Mod2Full2)$`p-value`[2]},
if(is.na(Mod1Full2S[[1]][1])==TRUE|is.na(Mod2Full2S[[1]][1])==TRUE)
{NA}),

c(if(!is.na(Mod1Full2S[[1]][1])==TRUE&!is.na(Mod3Full2S[[1]][1])==TRUE)
{anova(Mod1Full2,Mod3Full2)$`p-value`[2]},
if(is.na(Mod1Full2S[[1]][1])==TRUE|is.na(Mod3Full2S[[1]][1])==TRUE)
{NA}),

c(if(!is.na(Mod2Full2S[[1]][1])==TRUE&!is.na(Mod3Full2S[[1]][1])==TRUE)
{anova(Mod2Full2,Mod3Full2)$`p-value`[2]},
if(is.na(Mod2Full2S[[1]][1])==TRUE|is.na(Mod3Full2S[[1]][1])==TRUE)
{NA}),

c(if(!is.na(Mod1Full3S[[1]][1])==TRUE)
{c(as.numeric(Mod1Full3S$tTable[1:4,1]),as.numeric(Mod1Full3S$tTable[1:4,2]),
   as.numeric(Mod1Full3S$tTable[1:4,5]), as.numeric(VarCorr(Mod1Full3S)[2,2]),
   as.numeric(VarCorr(Mod1Full3S)[1,2]),Mod1Full3S$AIC,Mod1Full3S$BIC,Mod1Full3S$logLik)},

if(is.na(Mod1Full3S[[1]][1])==TRUE) {rep(NA,17)}),


c(if(!is.na(Mod2Full3S[[1]][1])==TRUE)
{c(as.numeric(Mod2Full3S$tTable[1:4,1]),as.numeric(Mod2Full3S$tTable[1:4,2]),
   as.numeric(Mod2Full3S$tTable[1:4,5]),
   as.numeric(VarCorr(Mod2Full3S)[3,2]),as.numeric(VarCorr(Mod2Full3S)[1,2]),
   as.numeric(VarCorr(Mod2Full3S)[2,2]),as.numeric(VarCorr(Mod2Full3S)[2,3]),
   Mod2Full3S$AIC,Mod2Full3S$BIC,Mod2Full3S$logLik)},

if(is.na(Mod2Full3S[[1]][1])==TRUE) {rep(NA,19)}),

c(if(!is.na(Mod3Full3S[[1]][1])==TRUE)
{c(as.numeric(Mod3Full3S$tTable[1:4,1]),as.numeric(Mod3Full3S$tTable[1:4,2]),
   as.numeric(Mod3Full3S$tTable[1:4,5]),
   as.numeric(VarCorr(Mod3Full3S)[3,2]),as.numeric(VarCorr(Mod3Full3S)[1,2]),
   as.numeric(VarCorr(Mod3Full3S)[2,2]),as.numeric(VarCorr(Mod3Full3S)[2,3]),
   as.numeric(coef(Mod3Full3$modelStruct$corStruct, unconstrained = FALSE)),
   Mod3Full3S$AIC,Mod3Full3S$BIC,Mod3Full3S$logLik)},

if(is.na(Mod3Full3S[[1]][1])==TRUE) {rep(NA,20)}),

c(if(!is.na(Mod1Full3S[[1]][1])==TRUE&!is.na(Mod2Full3S[[1]][1])==TRUE)
{anova(Mod1Full3,Mod2Full3)$`p-value`[2]},
if(is.na(Mod1Full3S[[1]][1])==TRUE|is.na(Mod2Full3S[[1]][1])==TRUE)
{NA}),

c(if(!is.na(Mod1Full3S[[1]][1])==TRUE&!is.na(Mod3Full3S[[1]][1])==TRUE)
{anova(Mod1Full3,Mod3Full3)$`p-value`[2]},
if(is.na(Mod1Full3S[[1]][1])==TRUE|is.na(Mod3Full3S[[1]][1])==TRUE)
{NA}),

c(if(!is.na(Mod2Full3S[[1]][1])==TRUE&!is.na(Mod3Full3S[[1]][1])==TRUE)
{anova(Mod2Full3,Mod3Full3)$`p-value`[2]},
if(is.na(Mod2Full3S[[1]][1])==TRUE|is.na(Mod3Full3S[[1]][1])==TRUE)
{NA}))
```

```
names(Collection) =
  c("Mod1Half1Inter","Mod1Half1GroupBase","Mod1Half1Standard","Mod1Half1Effect",
    "Mod1Half1InterSE","Mod1Half1GroupBaseSE","Mod1Half1StandardSE","Mod1Half1EffectSE",
    "Mod1Half1InterP","Mod1Half1GroupBaseP","Mod1Half1StandardP","Mod1Half1EffectP",
    "Mod1Half1Resid", "Mod1Half1b0",
    "Mod1Half1AIC", "Mod1Half1BIC","Mod1Half1LogLik",

    "Mod2Half1Inter","Mod2Half1GroupBase","Mod2Half1Standard","Mod2Half1Effect",
    "Mod2Half1InterSE","Mod2Half1GroupBaseSE","Mod2Half1StandardSE","Mod2Half1EffectSE",
    "Mod2Half1InterP","Mod2Half1GroupBaseP","Mod2Half1StandardP","Mod2Half1EffectP",
    "Mod2Half1Resid", "Mod2Half1b0","Mod2Half1b1","Mod2Half1rhob0b1",
    "Mod2Half1AIC", "Mod2Half1BIC","Mod2Half1LogLik",

    "Mod3Half1Inter","Mod3Half1GroupBase","Mod3Half1Standard","Mod3Half1Effect",
    "Mod3Half1InterSE","Mod3Half1GroupBaseSE","Mod3Half1StandardSE","Mod3Half1EffectSE",
    "Mod3Half1InterP","Mod3Half1GroupBaseP","Mod3Half1StandardP","Mod3Half1EffectP",
    "Mod3Half1Resid", "Mod3Half1b0","Mod3Half1b1","Mod3Half1rhob0b1","Mod3Half1Phi",
    "Mod3Half1AIC", "Mod3Half1BIC","Mod3Half1LogLik",

    "Mod1Half1Mod2Half1P","Mod1Half1Mod3Half1P","Mod2Half1Mod3Half1P",

    "Mod1Half2Inter","Mod1Half2GroupBase","Mod1Half2Standard","Mod1Half2Effect",
    "Mod1Half2InterSE","Mod1Half2GroupBaseSE","Mod1Half2StandardSE","Mod1Half2EffectSE",
    "Mod1Half2InterP","Mod1Half2GroupBaseP","Mod1Half2StandardP","Mod1Half2EffectP",
    "Mod1Half2Resid", "Mod1Half2b0",
    "Mod1Half2AIC", "Mod1Half2BIC","Mod1Half2LogLik",

    "Mod2Half2Inter","Mod2Half2GroupBase","Mod2Half2Standard","Mod2Half2Effect",
    "Mod2Half2InterSE","Mod2Half2GroupBaseSE","Mod2Half2StandardSE","Mod2Half2EffectSE",
    "Mod2Half2InterP","Mod2Half2GroupBaseP","Mod2Half2StandardP","Mod2Half2EffectP",
    "Mod2Half2Resid", "Mod2Half2b0","Mod2Half2b1","Mod2Half2rhob0b1",
    "Mod2Half2AIC", "Mod2Half2BIC","Mod2Half2LogLik",

    "Mod3Half2Inter","Mod3Half2GroupBase","Mod3Half2Standard","Mod3Half2Effect",
    "Mod3Half2InterSE","Mod3Half2GroupBaseSE","Mod3Half2StandardSE","Mod3Half2EffectSE",
    "Mod3Half2InterP","Mod3Half2GroupBaseP","Mod3Half2StandardP","Mod3Half2EffectP",
    "Mod3Half2Resid", "Mod3Half2b0","Mod3Half2b1","Mod3Half2rhob0b1","Mod3Half2Phi",
    "Mod3Half2AIC", "Mod3Half2BIC","Mod3Half2LogLik",

    "Mod1Half2Mod2Half2P","Mod1Half2Mod3Half2P","Mod2Half2Mod3Half2P",

    "Mod1Half3Inter","Mod1Half3GroupBase","Mod1Half3Standard","Mod1Half3Effect",
    "Mod1Half3InterSE","Mod1Half3GroupBaseSE","Mod1Half3StandardSE","Mod1Half3EffectSE",
    "Mod1Half3InterP","Mod1Half3GroupBaseP","Mod1Half3StandardP","Mod1Half3EffectP",
    "Mod1Half3Resid", "Mod1Half3b0",
    "Mod1Half3AIC", "Mod1Half3BIC","Mod1Half3LogLik",

    "Mod2Half3Inter","Mod2Half3GroupBase","Mod2Half3Standard","Mod2Half3Effect",
    "Mod2Half3InterSE","Mod2Half3GroupBaseSE","Mod2Half3StandardSE","Mod2Half3EffectSE",
    "Mod2Half3InterP","Mod2Half3GroupBaseP","Mod2Half3StandardP","Mod2Half3EffectP",
    "Mod2Half3Resid", "Mod2Half3b0","Mod2Half3b1","Mod2Half3rhob0b1",
    "Mod2Half3AIC", "Mod2Half3BIC","Mod2Half3LogLik",

    "Mod3Half3Inter","Mod3Half3GroupBase","Mod3Half3Standard","Mod3Half3Effect",
    "Mod3Half3InterSE","Mod3Half3GroupBaseSE","Mod3Half3StandardSE","Mod3Half3EffectSE",
    "Mod3Half3InterP","Mod3Half3GroupBaseP","Mod3Half3StandardP","Mod3Half3EffectP",
    "Mod3Half3Resid", "Mod3Half3b0","Mod3Half3b1","Mod3Half3rhob0b1","Mod3Half3Phi",
    "Mod3Half3AIC", "Mod3Half3BIC","Mod3Half3LogLik",

    "Mod1Half3Mod2Half3P","Mod1Half3Mod3Half3P","Mod2Half3Mod3Half3P",


    "Mod1Full1Inter","Mod1Full1GroupBase","Mod1Full1Standard","Mod1Full1Effect",
    "Mod1Full1InterSE","Mod1Full1GroupBaseSE","Mod1Full1StandardSE","Mod1Full1EffectSE",
    "Mod1Full1InterP","Mod1Full1GroupBaseP","Mod1Full1StandardP","Mod1Full1EffectP",
    "Mod1Full1Resid", "Mod1Full1b0",
    "Mod1Full1AIC", "Mod1Full1BIC","Mod1Full1LogLik",

    "Mod2Full1Inter","Mod2Full1GroupBase","Mod2Full1Standard","Mod2Full1Effect",
    "Mod2Full1InterSE","Mod2Full1GroupBaseSE","Mod2Full1StandardSE","Mod2Full1EffectSE",
    "Mod2Full1InterP","Mod2Full1GroupBaseP","Mod2Full1StandardP","Mod2Full1EffectP",
    "Mod2Full1Resid", "Mod2Full1b0","Mod2Full1b1","Mod2Full1rhob0b1",
    "Mod2Full1AIC", "Mod2Full1BIC","Mod2Full1LogLik",

    "Mod3Full1Inter","Mod3Full1GroupBase","Mod3Full1Standard","Mod3Full1Effect",
```

```
        "Mod3Full1InterSE","Mod3Full1GroupBaseSE","Mod3Full1StandardSE","Mod3Full1EffectSE",
        "Mod3Full1InterP","Mod3Full1GroupBaseP","Mod3Full1StandardP","Mod3Full1EffectP",
        "Mod3Full1Resid", "Mod3Full1b0","Mod3Full1b1","Mod3Full1rhob0b1","Mod3Full1Phi",
        "Mod3Full1AIC", "Mod3Full1BIC","Mod3Full1LogLik",

        "Mod1Full1Mod2Full1P","Mod1Full1Mod3Full1P","Mod2Full1Mod3Full1P",

        "Mod1Full2Inter","Mod1Full2GroupBase","Mod1Full2Standard","Mod1Full2Effect",
        "Mod1Full2InterSE","Mod1Full2GroupBaseSE","Mod1Full2StandardSE","Mod1Full2EffectSE",
        "Mod1Full2InterP","Mod1Full2GroupBaseP","Mod1Full2StandardP","Mod1Full2EffectP",
        "Mod1Full2Resid", "Mod1Full2b0",
        "Mod1Full2AIC", "Mod1Full2BIC","Mod1Full2LogLik",

        "Mod2Full2Inter","Mod2Full2GroupBase","Mod2Full2Standard","Mod2Full2Effect",
        "Mod2Full2InterSE","Mod2Full2GroupBaseSE","Mod2Full2StandardSE","Mod2Full2EffectSE",
        "Mod2Full2InterP","Mod2Full2GroupBaseP","Mod2Full2StandardP","Mod2Full2EffectP",
        "Mod2Full2Resid", "Mod2Full2b0","Mod2Full2b1","Mod2Full2rhob0b1",
        "Mod2Full2AIC", "Mod2Full2BIC","Mod2Full2LogLik",

        "Mod3Full2Inter","Mod3Full2GroupBase","Mod3Full2Standard","Mod3Full2Effect",
        "Mod3Full2InterSE","Mod3Full2GroupBaseSE","Mod3Full2StandardSE","Mod3Full2EffectSE",
        "Mod3Full2InterP","Mod3Full2GroupBaseP","Mod3Full2StandardP","Mod3Full2EffectP",
        "Mod3Full2Resid", "Mod3Full2b0","Mod3Full2b1","Mod3Full2rhob0b1","Mod3Full2Phi",
        "Mod3Full2AIC", "Mod3Full2BIC","Mod3Full2LogLik",

        "Mod1Full2Mod2Full2P","Mod1Full2Mod3Full2P","Mod2Full2Mod3Full2P",

        "Mod1Full3Inter","Mod1Full3GroupBase","Mod1Full3Standard","Mod1Full3Effect",
        "Mod1Full3InterSE","Mod1Full3GroupBaseSE","Mod1Full3StandardSE","Mod1Full3EffectSE",
        "Mod1Full3InterP","Mod1Full3GroupBaseP","Mod1Full3StandardP","Mod1Full3EffectP",
        "Mod1Full3Resid", "Mod1Full3b0",
        "Mod1Full3AIC", "Mod1Full3BIC","Mod1Full3LogLik",

        "Mod2Full3Inter","Mod2Full3GroupBase","Mod2Full3Standard","Mod2Full3Effect",
        "Mod2Full3InterSE","Mod2Full3GroupBaseSE","Mod2Full3StandardSE","Mod2Full3EffectSE",
        "Mod2Full3InterP","Mod2Full3GroupBaseP","Mod2Full3StandardP","Mod2Full3EffectP",
        "Mod2Full3Resid", "Mod2Full3b0","Mod2Full3b1","Mod2Full3rhob0b1",
        "Mod2Full3AIC", "Mod2Full3BIC","Mod2Full3LogLik",

        "Mod3Full3Inter","Mod3Full3GroupBase","Mod3Full3Standard","Mod3Full3Effect",
        "Mod3Full3InterSE","Mod3Full3GroupBaseSE","Mod3Full3StandardSE","Mod3Full3EffectSE",
        "Mod3Full3InterP","Mod3Full3GroupBaseP","Mod3Full3StandardP","Mod3Full3EffectP",
        "Mod3Full3Resid", "Mod3Full3b0","Mod3Full3b1","Mod3Full3rhob0b1","Mod3Full3Phi",
        "Mod3Full3AIC", "Mod3Full3BIC","Mod3Full3LogLik",

        "Mod1Full3Mod2Full3P","Mod1Full3Mod3Full3P","Mod2Full3Mod3Full3P")

  return(Collection)}

AnalysisOUT = function(Data,Iter){
  DataDF = t(as.data.frame(Data))
  rownames(DataDF)=c(NULL)
  DFL = pivot_longer(as.data.frame(DataDF), cols = everything())

  DFL$Types =
    rep(rep(c(rep("Estimate",4),rep("SE",4),rep("Pvalue",4),rep("RE",2),rep("Model",3),
            rep("Estimate",4),rep("SE",4),rep("Pvalue",4),rep("RE",4),rep("Model",3),
            rep("Estimate",4),rep("SE",4),rep("Pvalue",4),rep("RE",5),rep("Model",3),
            rep("Model",3)),6),Iter)

  DFL$Types2 = rep(rep(c(rep(c("Intercept","GroupBase","Standard","Effect"),3),"Residual","Sigmab0",
                        "AIC","BIC","LogLik",
                        rep(c("Intercept","GroupBase","Standard","Effect"),3),"Residual","Sigmab0",
                        "Sigmab1","rhob0b1","AIC","BIC","LogLik",
                        rep(c("Intercept","GroupBase","Standard","Effect"),3),"Residual","Sigmab0",
                        "Sigmab1","rhob0b1","Phi","AIC","BIC","LogLik",
                        "Model1Model2P","Model1Model3P","Model2Model3P"),6),Iter)

  DFL$Length = rep(c(rep(rep("Half",59),3),rep(rep("Full",59),3)),Iter)
  DFL$Model =rep(rep(c(rep("Model1",17),rep("Model2",19),rep("Model3",20),rep("ModelComparison",3)),6),Iter)
  DFL$Frequency = rep(rep(c(rep("OnceWeek",59),rep("TwiceWeek",59),rep("ThreeWeek",59)),2),Iter)
  return(DFL)}

# Output with filter and group
OutputGeneral = function(Data, Groupargs,...){

  byG = syms(Groupargs)
```

```
  Data %>%
    filter(...) %>%
    group_by(!!!byG) %>%
    summarise(mean = round(mean(value,na.rm=T),3),sd = round(sd(value,na.rm=T),3),
              Q25 = round(quantile(value, 0.25,na.rm=T),3),Median = round(median(value,na.rm=T),3),
              Q75 = round(quantile(value, 0.75,na.rm=T),3))
}

Pvalue1 = function(Data, Groupargs,...){
  byG = syms(Groupargs)
  Data %>%
    filter(...) %>%
    group_by(!!!byG) %>%
    summarise(P005 = round(100*(mean(value<0.05,na.rm=T)),1))
}

SUMNA = function(Data, Groupargs,...){
  byG = syms(Groupargs)
  Data %>%
    filter(...) %>%
    group_by(!!!byG) %>%
    summarise(SumNA = sum(is.na(value)),
              length = length(value),
              Percentage = 100*(SumNA/length))
}
```