Improving Measurement by Accounting for Time-Varying Fluctuations: Design-Based and Model-Based Methods

Jinghui Liang
梁靖晖

Submitted in fulfilment of the requirements for the
Degree of Doctor of Philosophy

School of Psychology and Neuroscience
College of Medical, Veterinary and Life Sciences
University of Glasgow

October, 2025

Abstract

During an experimental or a survey session, participants adapt and change due to learning, fatigue, fluctuations in attention, or other physiological or environmental changes. This temporal variation affects measurement, potentially reducing research power and validity. This thesis discussed how time-varying fluctuations bias measurements and how dealing with these fluctuations can improve measurements in two typical psychological research environments: cognitive experiments and psychological measurements. Two methodological parts are presented. The first one reviews typical cognitive experimental designs, and provided methods to account for time-varying fluctuations and improve power in experimental studies. These methods are based on better randomization algorithm and advanced statistics models. The second part introduced how to control time-varying fluctuations in psychometric datasets by finite mixture models to increase validity. It also provides an online platform for better randomizing and counterbalancing surveys. Both parts found that dealing with time-varying fluctuations benefits to power and validity gains, therefore increasing the reproducibility of psychological studies.

Acknowledgements

I would like to take this chance to thank some people.

First of all, thank you to my supervisor, Dr. Dale Barr. Thank you for all your support, inspiration, patience, and encouragement since day one. Thank you for reshaping my R knowledge, teaching me statistical skills and Emacs sorcery, and helping me make every bit of magic happen. Beyond your psychological wisdom and professional guidance, I am especially grateful to you—from the bottom of my heart—for introducing me to a colorful Glaswegian life. The first greeting from Glasgow, the first dram of whisky, the first live music gig... I remember and treasure every story. More than an academic role model, I am very glad to have you as my best friend. From you, I have learned not only "how to be a good psychological researcher," but also "how to be a good person."

I also wish to thank Alistair Beith. Thank you for sharing your extensive programming knowledge and always being there to help when my machine glitched—again and again. I might never understand those Emacs configurations you provided, but my computers do. So I thank you on their behalf as well. And if Dale brought me into the world of whisky, then you were my guide through it. I (along with my computers and bottles) wish you all the best—slàinte!

I would like to thank the China Scholarship Council for funding my doctoral work. I am also grateful to the members of the Methods & Metascience Journal Club, who offered informative and sincere suggestions for my project. I thank Tuo Liu at Goethe University Frankfurt—thank you for keeping in touch after the conference, for sharing your inspiring knowledge of psychometrics, and for helping me make further progress in the field. I look forward to our future collaboration.

Thanks also to Christoph Scheepers, Valentina Gosetti, Chaona Chen, Mirza Muchammad Iqbal, Alejandro Bahena Rivera, and Xuanyi Ma. Thank you for always supporting me and taking care of my mental health. I am lucky to have such a vibrant group of people in my life. Thank you to all my lovely colleagues in the School of Psychology and Neuroscience. Don't worry if your name is not in this short acknowledgment-it is in my mind.

A special shout-out to my friends. Thank you to Zuchen Huang, who was at the University of Manchester. As a friend of over twelve years, thank you for being there through good times and bad, and for always making time to listen—even when you were busy. I'll see you soon at your PhD graduation ceremony. Thank you also to Ming Huang, Zhaorong Chen, and Jianbin Li. I treasure every happy moment we shared, and I wouldn't have remembered how to speak Cantonese without you. Thank you to all my friends in China for always checking if I am still alive. Although calculating the time difference, staying up until 3 am just for raising up a pint together with me is never a healthy idea, but you guys always do so and I appreciate it.

Thank you to Zirui Zhang, best roommate ever. Thank you for always being nice and taking care of my diet. Your chili oil noodles conquered my stomach. I truly think you should take it to Hell's Kitchen or Culinary Class Wars, although that makes a show nothing to see for your super fast easy win.

My thanks go to my family. Thank you to my father, Zhan Liang; my mother, Lixia Ma; my sister, Jiaxin Liang; and my brother-in-law, Zhiwen Huang. Thank you for your care and heartwarming support over the years. Your love has made me strong enough to face life's challenges. 阿爸阿妈，我终于毕业啦！

Finally, I wish to thank my loving partner, Li Zhang. Thank you for being in my life and for bringing me happiness and love—I wouldn't be the same person without you. Especially, thank you for your unconditional support during my writing year. Traveling back and forth between two cities was tiring, but it was worthy. Because with every second the train moved, we were getting closer to each other.

Contents

List of Figures

List of Tables

Declaration

All work in this thesis was carried out by the author unless otherwise explicitly stated.

All work in this thesis was supported by a Chinese Scholarship Council (CSC) Grant (CSC no. 202108060121) to the author. The funding source had no role in the design or execution of the research.

Chapter 1

The Problem of Time-Varying Fluctuations in Measurements

"No man ever steps in the same river twice, for it is not the same river and he is not the same man." – Heraclitus

1.1 Background

Every psychology researcher knows that participating in an experiment or filling in an questionnaire takes time. Information about how long a study or questionnaire will take is always provided in advertisements for participant recruitment, and participants are often rewarded in proportion to the study duration. Given this, it is an oddity of current research practice that the temporality of data collection is barely considered when designing experiments or when analyzing data. In their approach to design and analysis, the typical researcher tends to behave as though all the data for a given participant has been collected *simultaneously*, without any of the mind wandering, task adaptation, excitement and/or boredom that characterizes human participation in any prolonged activity. Indeed, it is doubtful that even participants who were highly motivated to provide perfectly consistent, truthful, and attentive responses would be able to avoid occasional lapses of attention or not change their response speed or accuracy as they learn about the experimental task.

In this thesis I attempt to attack the problem of the non-simultaneity of measurement head on, focusing on the two major data collection paradigms in psychology and neuroscience: cognitive experiments and survey instruments. Nearly all experimental studies and psychometrics investigations take repeated measurements of the neurological and/or behavioral responses of a single participant, either under constant or varying conditions of measurement. Often this is simply more practical, economical, and statistically efficient than taking a single measurement from each of many subjects. Or a researcher might wish to investigate whether a psychological construct is valid under certain samples, which requires applying multiple items (i.e., questions in a psychological measurement instrument) measuring the same construct to a participant. But whatever the underlying motivation, most experiments involve repeated measurements

which by their very nature form a time series, and the full dataset forms a collection of such.

In cognitive experiments, traditionally delivered in a lab but increasingly taking place online, the researcher usually measures some dependent variable (such as response time) under a variety of experimental conditions. In modern experiments, it is typical to take many repeated measurements from the same participant under the same repeated conditions within a single testing session. For instance, in the well-known Stroop task, participants are presented with color words (e.g., RED, BLUE, GREEN) and must respond with the color that each word is printed in, with stimuli appearing in congruent (the word RED in printed in red font) or incongruent (the word RED appearing in green font) conditions. The set of familiar color words is finite and so is the set of recognizable colors, and so each participant will see many stimuli (usually in the tens but sometimes in the hundreds) in each of the two conditions. At the start of the session, the participant may be keenly interested in the task but will take slightly long to respond to each stimulus because of their unfamiliarity with the experiment. Over time, they become more familiar and start responding more quickly and accurately. But over time, the participant might become bored and tired, and experience lapses of attention. All of these things are known to affect measurement, and it seems likely that they will affect it in idiosyncratic ways from one participant to the next. These human factors introduce "nuisance variation" into the measurement process, much of which is largely unpredictable.

The same processes are at play in the psychometric context, where a survey instrument is deployed with the aim of measuring some latent psychological property. To do so, researchers choose a validated psychometrics instrument. Participants answer the questions on a survey sequentially, with items usually measuring the same underlying construct. For instance, there may be items such as "I have a positive life attitude", "I am full of hope for my future". But with similar questions appearing over and over again, a participant might provide a similar answers to a similar previous question just for the sake of speed and consistency, rather than carefully thinking about the wording and assessing how truthfully it represents them. Their seriousness and attentiveness to the task might come and go over time, just as it does in an experimental study.

So for the two main measurement scenarios in behavioral studies, experiments and surveys, human factors are likely to introduce *time-varying fluctuations* into the measurement process. The presence of time-varying fluctuations reflects the fundamental sequential and repeated nature of experiments and surveys. Time-varying fluctuations are extremely common; that is because "*unlike molecules or plots of barley, human beings adapt quickly and continuously to their environment*" (Baayen et al., 2017 p.207). Such adaption applies to the experimental environment and task: participants may may switch between discrete performance strategies (Ashwood et al., 2022), experience attentional lapses (Robertson et al., 1997), fatigue (van der Linden et al., 2003), or other fluctuations in underlying physiological states. Also, in a psychometric investigation, task-takers may: adjust their response of current items based on their previous responses (Gehlbach & Barge, 2012); provide more consistent responses as related life experience is extracted and primed (Ozkok et al., 2019); learn from previous items to take

action to the following items (Chen & Wang, 2007). Given that the vast majority of studies in psychology and neuroscience involve repeated measures taken on the same participants, it is no exaggeration to point out that nearly all studies in these fields are contaminated by time-dependent noise.

1.2 Potential harms of ignoring the non-simultaneity of measurement

As noted, to ignore the time dimension when designing and analyzing psychological data is to behave as though the data has been collected simultaneously. But the assumption of simultaneity rarely, if ever, holds. What are consequences for behavioural studies? As I argue below (and show in subsequent chapters), in the experimental context, contamination of measurement by time-varying fluctuations impairs power, while in the psychometric context, it impairs validity.

Some see temporal fluctuation in measurement as a potential threat to the validity of statistical analyses of repeated-measures data. Parametric approaches to data analysis make the fundamental assumption that the residuals of the analytical model are all independent. But if we accept the above arguments that people unavoidably adapt and change over time, then this assumption will be violated everywhere except in single item surveys or experiments having only a single trial. Ignoring time variation in the model can induce *temporal autocorrelation* in the model residuals, whereby residuals taken from observations taken close in time will be more strongly correlated than residuals separated in time. Usually, this appears as a positive correlation as a function time the time lag between residuals, which violates independence assumptions (Bence, 1995). However, much of the literature warning about the harms of autocorrelation involves the analysis of just one or perhaps several time series, a common situation in economic forecasting or political polling, but very much unlike the "multi-level" context of experimental studies where there are as many time series as participants.

Recently, researchers have expressed concerns that failing to account for temporal autocorrelation in the analysis multi-level time-series may inflate false positives (Amon & Holden, 2021; Baayen et al., 2017). Indeed, some have gone as far as suggesting that that the presence of time-dependent error structure in multi-level data wholly invalidates the use of classical statistical approaches (Amon & Holden, 2021). However, this view ignores that most experimentalists, whether they know it or not, already employ a design practice that safeguards against the potential for temporal autocorrelation to induce false positives: namely, basic randomization (Thul et al., 2021).

Researchers typically randomize or counterbalance the sequence in which experimental conditions are presented to participants. As we explain in Chapter 3, when basic randomization is employed, ignoring temporal fluctuations does not induce false positives, but rather impairs statistical power. Thus, the tradition of ignoring temporal fluctuations may be a contributor to chronically low statistical power in psychology and neuroscience (Button et al., 2013; Cohen, 1962; Maxwell, 2004). Statistical power has emerged as key point of discussion in debates about replicability and reproducibility in science, and has generally been accompanied by calls

to increase sample size well beyond traditional standards.

Many analyses of experimental data work with participant means rather than the raw trial-by-trial observations, so independence is not a concern, as now we are dealing with residuals from condition means calculated over many trials rather than residuals from individual trials. Although means are more likely to be independent than the individual observations comprising them, the means will still be contaminated from the time-varying fluctuations, and so power is still compromised.

For psychometrics, when evaluating the utility of a measurement instrument by means of traditional analysis (e.g., factor analysis), classic test theory assumes that the variances of item scores are purely accounted for by the psychological construct, and no relationships remain among any item pairs after accounting for it (McNeish, 2018). Researchers sometimes find their data does not validate the measured psychological construct and, on this basis, suspect that the test instrument (usually a questionnaire) has *low validity*. However, it is possible that such low validity might arise is partly because time-varying fluctuations violate the local independence assumption by producing measurements that are related in time. In questionnaire datasets, where measurements are usually ordinal, time-varying fluctuations would not exhibit pattern that is easy to visualize as they do in continuous datasets because they affect participants' probability of selecting a response category (Myszkowski & Storme, 2024). Therefore, time-varying fluctuations are harder to discover.

1.3 Model-based and design-based approaches

In the rare cases that researchers acknowledge the need to attempt to counteract the contaminating effect of time-varying fluctuations, it is generally only in the analysis stage, after all the data have already been collected. Recent years have seen an ever-increasing number of studies acknowledging the problem and proposing and debating potential analytical solutions (Asparouhov et al., 2023; Baayen et al., 2017; Baayen et al., 2022; Myszkowski & Storme, 2024; Ozkok et al., 2019; Thul et al., 2021). One attractive approach is to use time-series modeling, a well-established framework for statistically modelling change over time (Fitzmaurice et al., 2012; Mirman, 2016). One state-of-the-art modeling framework that we will explore in more detail in Chapter 4 is Generalized Additive Mixed Modeling (GAMMs), which make it possible to flexibly model many different kinds of continuous wiggly patterns (Baayen et al., 2017; Wood, 2017). In Chapter 7, we will also consider models of autocorrelation structure within the psychometric context.

Model-based approaches can be highly effective and can generate powerful insight, but have been developed specifically for situations where the dimension of time is of theoretical interest. They can also be extremely difficult to implement especially when data is sparse, they can yield results that are hard to interpret and easy to misunderstand, and demand a high degree of technical knowledge (Thul et al., 2021). While these sophisticated approaches are critical in situations where time is of central interest, it is asking a lot of researchers to apply them

routinely across all situations, where time-varying effects are seen as more of a nuisance than a potential source of insight.

A major contribution of the current thesis is to show the potential benefits of considering time-varying fluctuations at the earliest stage of the research process: *while studies are being designed.* There is tremendous potential to improve the quality of measurement by designing studies that acknowledge the human element from start to finish. The *design-based* approach that I advocate for experimental studies shows how improving upon the basic randomization procedure can improve the quality of measurement and thus, improve the quality of power. Indeed, the power gains attainable are almost as good as those attainable using the most powerful model-based approaches. Although adapting a design-based approach to the psychometric context is possible, it is more challenging, and so this thesis is limited to laying the groundwork for future efforts in the psychometric arena that build upon the experimental results.

1.4 Overview

How can we improve measurement by accounting for time-varying fluctuations with design-based and model-based methods? This thesis is divided into two parts, with the first (and most substantial) part focusing on time-varying fluctuations in the experimental context, and the second part focusing on the psychometric context.

The first part, **Experimental Studies**, has four chapters (Chapters 2 to 5) and is the core of this thesis. This part focuses on how to deal with time-varying fluctuations in experimental datasets to boost power. To consider the theoretically maximal power gains of our approach, we start with the simplest experimental design: a one-factor design. Chapter 2 discusses how time-varying fluctuations represent an untapped source of additional power in repeated-measures experiments, and the possibility to clean up such fluctuations by designing better randomization algorithms. This chapter also contains a literature review that seeks to characterize what counts as a 'representative' experimental design. In Chapter 3, we outline a new randomization algorithm for one-factor designs, *Permuted Subblock Randomization* (PSR), and show how it helps segregate time-varying fluctuations from effects of interest, therefore improving power. Under a Monte Carlo simulation study, we found that our design-based approach can boost power by up to 45% for representative one-factor experimental designs. A tutorial about how to easily implement PSR in experiments is also detailed. Next, in Chapter 4, we demonstrate the test-independent properties, demonstrating that it yields similar benefits to power regardless of whether data are analyzed with ANOVA or linear-mixed effects models. More importantly, this chapter compares the performance of the design-based approach of PSR to a model-based approach that uses GAMMs, finding that PSR yields gains that often approach the gains of GAMMs. We also found that combining design-based and model-based approaches can optimize the power by over 50% in certain cases. The last chapter of Part I, Chapter 5, extrapolates PSR to 2x2 factorial designs. We discuss how power is allocated across different effects in the 2x2 factorial design framework, and introduced two variations of PSR that adapt it to the factorial context. We conducted a new Monte Carlo simulation study to validate the power advantages

of PSR in this chapter. Our results suggested that PSR benefits power in representative 2x2 factorial designs and can either achieve a general power improvement or increase power for specific main effects or interaction terms.

The second part of this thesis, **Psychological Measurement**, contains two chapters (Chapters 6 and 7). Building on the findings from Part I, it focuses on dealing with time-varying fluctuations in psychological measurement datasets, especially in psychometrics surveys. Because of long-term debates and the difficulties in psychometrics, it is not clear design-based approaches are useful as they are in experimental designs.

Therefore, in Chapter 6, we started with a model-based approach, Mixture Autoregressive Confirmatory Factor Analysis (MAR-CFA), to improve validity of psychometrics datasets. The MAR-CFA is a methodological extension of a recent approach (Ozkok et al., 2019), and seeks to capture time-varying fluctuations by clustering latent heterogeneity. Our simulation results showed that ignoring time-varying fluctuations in datasets could lead to a 20% overestimation of factor loadings, while MAR-CFA efficiently moderated such overestimation. However, implementing MAR-CFA imposes a considerable computational burden and requires a relatively large sample size. Therefore, in Chapter 7, we introduced a new online survey builder and data collection platform, **RandomiSur**, aiming to provide a new design-based perspective to improve validity in psychometrics. In this chapter, I firstly detail why design-based approaches are less appreciated in the field of psychometrics. Then RandomiSur is introduced as an online tool to counterbalance and randomize psychometric measurement. This is based on our success in dealing time-varying fluctuations in experimental studies. A tutorial about how to deploy RandomiSur and how to launch a randomized psychometrics survey with it is provided in this chapter in order to help future research to better design psychometric measurement instruments.

At the time of writing of this thesis, the findings presented in Chapters 2 and 3 have been published in the peer-reviewed journal *Psychological Methods*, see Liang & Barr (2024).

Part I: Experimental Studies

Chapter 2

Typical Study Designs for Cognitive Experiments

2.1 Background

As Chapter 1 outlined, time-varying fluctuations could be seen as an untapped power source. If time-dependent noise is controlled through randomization—a design-based approach—an effect of interest could be detected more easily as it is less likely to be masked. It is also possible to use statistical modelling to "clean up" temporal nuisance effects from the data (Baayen et al., 2017; Baayen et al., 2022), but doing so is challenging because the functional form of such effects is generally unknown. Moreover, it may be inconsistent across participants, and can include curvilinear, asymptotic, and/or discontinuous components (as will happen when participants take rest breaks during a session). But statistical modelling is not the only way to improve power. If we accept that humans adapt and change over an experiment, then we can-and should-design our experiments with this temporal variation in mind. As adaptions and changes are dependent to sequenced stimuli, carefully designing the trial presentation order might be helpful to eliminate such effects.

Textbook discussions of organizing the presentation order of within-participant conditions in experimental studies typically focus on counterbalancing. Furthermore, these discussions concern situations where there is only a single observation per condition per participant. In such circumstances, counterbalancing ensures each condition appears in each sequential position the same number of times across participants. For example, a common approach is to counterbalance the presentation order of conditions across participants using a Latin Square design (Kirk, 2013; Rosenthal & Rosnow, 2008; Shadish et al., 2002). As presented in Figure 2.1 (left panel), if there are four conditions labelled $ABCD$, then the first participant might get the order $ABCD$, with participants two through four receiving orders $BCDA$, $CDAB$, and $DABC$. This scheme has two nice properties: (1) each condition appears the same total number of times in the data (four, in this example); and (2) each condition appears in each sequential position equally often.

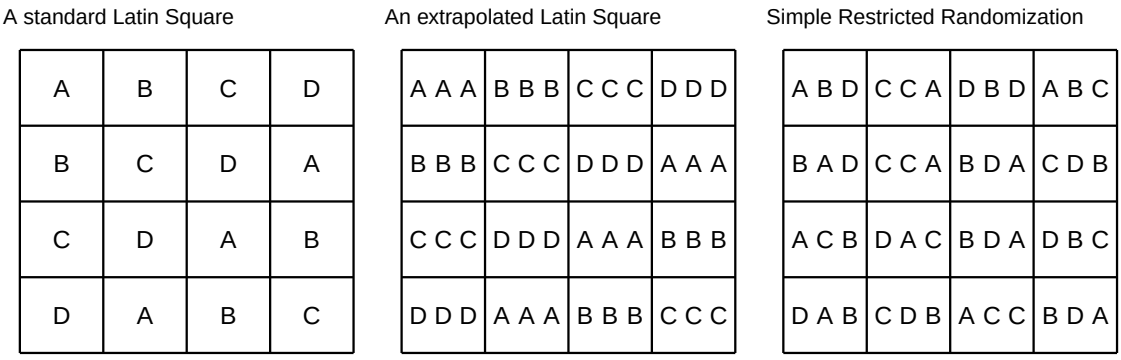

Figure 2.1: Trial sequences organized in a counterbalanced Latin Square (left panel), an extrapolated Latin Square (middle panel), and the simple restricted randomization.

However, for cognitive experiments where there are multiple observations per experimental condition, Latin Square is not a good method for counterbalancing. One could, for instance, extrapolate the Latin Square logic to repeated observations by simply organizing the conditions in each level into a block, as presented in Figure 2.1 (middle panel). If each participant gets each condition three times, then participant one gets $AAABBBCCCDDD$, participant two gets $BBBCCCDDDAAA$, and so on. However, this has three disadvantages. First, the condition of each trial is highly predictable within each block. Second, each condition is always preceded (and followed) by the same condition (e.g., B always comes before C and after A), which is also seen in the standard Latin Square. Third, Monte Carlo simulations suggest that for datasets having time-varying error structure (due to learning effects, waxing and waning of attention, fatigue, etc.), blocking the levels of an independent variable together can be catastrophic for power (Thul et al., 2021).

To overcome these disadvantages, using randomization will generally be preferable over counterbalancing using a Latin square. However, experimentalists will not find much in the literature to guide their randomization choices because much of the literature on randomization is focused on clinical trials (Berger et al., 2021), which differ from cognitive experiments in important ways. Many randomized clinical trials are between-subject designs where each participant is randomly allocated to a treatment or control arm of a study, and there is much concern with avoiding selection bias in the allocation process. In contrast, most cognitive experiments employ within-subject designs where each participant not only gets the full set of conditions (i.e., experimental conditions), but usually multiple measurements are taken at each. Many clinical trials and most cognitive experiments use a form of restricted randomization (Pocock, 1979) where the randomization process is constrained to ensure a balanced study. In the context of laboratory experiments, this entails taking the full set of balanced conditions (e.g., three trials in each of conditions A,B,C and D) and scrambling the order independently for each participant, as presented on Figure 2.1, right panel. We refer to this approach as *simple restricted randomization*.

Employing randomization allows researchers to maximally remove selection bias and position effects of stimuli or treatment conditions. But if we see controlling time-varying fluctuations

as a potential way of improving measurement, we should also see imposing further restrictions on randomization as a potentially accessible and efficient way to improve power. By further restricting randomization, we can produce a sequence where stimuli or treatment conditions are more evenly distributed over time while still remaining relatively unpredictable. However, this potential is generally overlooked by researchers when applying randomization. In fact, although nearly all laboratory within-subject design experiments involve randomization in the sequencing of stimuli or conditions, when researcher express motivations for imposing further restriction they seldom mention power, but rather aim to avoid cross-trial contamination or to obscure the purpose of an experiment from participants (van Casteren & Davis, 2006). Additionally, in analyzing experimental data—whether intentional or not—researchers often perform analyses that effectively assume *simultaneity of measurement*: that is, they assume a measurement's position in the sequence has no bearing on its value. But if we recognize the time-dependent nature of experiments, simultaneous measurement is impossible. Here, we contend that experimental researchers can benefit by considering more sophisticated approaches to randomization found in clinical trials research that consider the non-simultaneity of measurement. In the following chapters, we introduce new restricted randomization approaches that accomplish this in the experimental context.

To know how well an algorithm will perform, it is essential to understand the environment in which it will be deployed. So before introducing these algorithms, we begin with a study aiming to find the typical design structures of common cognitive experiments. What types of experimental designs are most representative of cognitive experiments? One aspect concerns the number of independent variables (factors) in the design, which determines the number of treatment conditions. For one-factor designs, which will be the focus of the next two chapters, there are three essential design parameters: (a) the number of factor levels, denoted as n_k; (b) the number of repetitions per level, denoted as n_r; and (c) the number of participants, denoted as n_p. These parameters can be collected from published research. However, before going into collecting design parameter data, some details about these parameters would be helpful to better understanding why they are crucial to randomization.

We start out with the number of factor levels, n_k. Obviously, a factor must have at least two levels $n_k \geq 2$ to be counted as a factor and to be subject to randomization. For a factor with only two levels A and B and only one trial per condition, also two random sequences are possible, AB and BA. However, even with such simple randomized sequences, we still avoid bias from a potential position effect: A would not always appear in front of B, and B would not always appear behind A. With n_k increasing, the number of possible randomized sequences expands, and we are more likely to avoid situations where one level would always appeared in front of or behind another level. There are special cases in designs with more than one factors (e.g., factorial designs), where n_k could also represent the number of conditions constructed by combining levels from all factors.

The number of repeated observations taken within each level, n_r, also determines the space of possible random sequences. As mentioned, repeated-measure designs with multiple mea-

surements per factor level is extremely common in psychology and neuroscience. With a fully within-participant design, the researcher takes $n_k \times n_r$ observations for each participant. Applying simple restricted randomization with small n_k but large n_r may not be a good idea because there will be lots of runs of trials in the same condition. For example, given a factor with only two levels A and B, a very large n_r would be unlikely to generate sequences that regularly switching conditions ($ABABAB$...) and more likely to bunch condition together ($AAABBB$...). In subsequent chapters, we will explore randomization algorithms that help more evenly spread the conditions.

Finally, every experiment has a certain number of participants, n_p. Discussions of power in the literature almost entirely focus on this one parameter, n_p. Undeniably, participant sample size is a major factor that affects power, (Marszalek et al., 2011), but it also determines the space of possible randomizations for the whole sample. In many clinical experiments and cognitive experiments that are between-subject, n_p is determined to construct a balanced design. When multiple treatment groups are determined, researchers need to randomly allocate participants to treatment groups in order to construct a balanced design across groups. In this case, n_p directly affects the randomization process. In fully within-subject experiments, attention should be also be directed at the sequencing of the $n_k \times n_r$ observations. As n_p increases, the more independently randomized sequences generated, the more possible trials would be evenly spread into all possible positions in sequences. Therefore, in fully within-subject experiments, although n_p does not take part in the essential randomization process, it indirectly affects the utility of randomization by allowing more evenly distributed stimuli or treatment conditions across the sample. However, n_p cannot be always increased due to limitations in real life (ethical issues, research expenditure, special samples, etc.). So investigating how power can be improved with fixed (and typical) values of n_p is useful.

To determine typical design parameters, we conducted a comprehensive literature review of recent cognitive experiments. Our results help ensure that we focus on typical design parameters when evaluating the new randomizations that appear in the forthcoming chapters.

2.2 Methods

We targeted all cognitive experiments reported in a recent full publication year (2023) in the *Journal of Experimental Psychology: Human Perception & Performance*[2.1]. This journal focuses on publishing experimental papers in different areas of psychology, and the design parameters retrieved from its publications are representative to common experimental settings. We extracted design parameters from 161 experiments reported in 68 articles that fulfilled our three selection criteria, which were that articles must include: (1) at least one experiment where randomization was explicitly mentioned; (2) repeated measurements of at least one within-subject independent variable; and (3) multiple observations for each within-subject factor level or for each combination of within-subject factors, if a design has more than one within-subject factors. For each experiment, we collected design parameters n_p, n_r, and n_k. During the information

[2.1] At the time this review was finished, 2023 was the last full publication year of the target journal.

extraction process, if a study contained more than one experiment, corresponding information for each experiment was independently extracted. We also noted that some experiments used a *pseudorandomization* strategy that applies randomization on top of certain limitations (e.g., applying for some treatment groups, some specific trials, or with limitations for relative treatment orders). These experiments were included with the label *pseudo*. In addition, for studies with one within-subject factor, n_k simply represented the number of factor levels; while for experiments with multi-factor designs, we calculated n_k as a product of the number of factor levels for each factor and n_r as the number of within-subject repetitions within each condition of the design. For example, given an experiment having two within-subject factors and each factor has two levels, the number of conditions, n_k, is 4. And if there are two repetitions per conditions, $n_r = 2 \times 4 = 8$.

2.3 Results and discussion

Table 2.1 listed the literature review results. Out of these 161 experiments, 30.4% had a single within-participant factor, 61.2% had $n_k = 2$, 30.6% had $n_k = 3$, and the remaining 8.2% had $n_k \geq 4$. On top of basic employment of randomization, 11 experiments (0.1%) additionally restrict the presentation order (i.e., using pseudorandomization design), in order to allow interleaved conditions (e.g., Yarrow et al., 2023) and to suppress continuous appearances of stimuli under the same condition (e.g., Severijnen et al., 2023).

Table 2.1: Literature review results

Paper	Randomization	n_p	n_k	n_r	Notes
Ma & Abrams (2023)	Y	24	2	48	Exp 1
Ma & Abrams (2023)	Y	24	2	54	Exp 2
Schirmer et al. (2023)	pseudo	61	4	50	Exp 2
Chan & Saunders (2023)	Y	16	4	45	Exp 1
Chan & Saunders (2023)	Y	16	4	45	Exp 2
Guitard & Cowan (2023)	Y	120	4	3	Exp 3
Guitard & Cowan (2023)	Y	120	4	6	Exp 3
Gibson et al. (2023)	Y	40	8	5	Exp 1
Gibson et al. (2023)	Y	80	8	5	Exp 2
Vandenberghe & Vannuscorps (2023)	Y	30	6	10	Exp 1
Vandenberghe & Vannuscorps (2023)	Y	60	6	5	Exp 2
Mainka et al. (2023)	Y	20	5	10	
Savino & Kahan (2023)	Y	32	16	5	Exp 1
Savino & Kahan (2023)	Y	32	32	3	Exp 2
Sobrinho & Souza (2023)	Y	28	4	6	Exp 1
Sobrinho & Souza (2023)	Y	112	4	6	Exp 2
Bissett et al. (2023)	Y	66	12	20	

Continued on next page

Continued from previous page

Paper	Randomization	n_p	n_k	n_r	Notes
Kinoshita et al. (2023)	Y	40	6	20	Exp 1
Kinoshita et al. (2023)	Y	49	6	20	Exp 3
Kinoshita et al. (2023)	Y	41	6	20	Exp 2
Kinoshita et al. (2023)	Y	42	6	20	Exp 4
Pedziwiatr et al. (2023)	pseudo	36	2	3	Exp 1
Pedziwiatr et al. (2023)	pseudo	18	2	3	Exp 2
Pedziwiatr et al. (2023)	pseudo	20	2	3	Exp 3
Hu et al. (2023)	Y	74	3	2	Exp 2
Overkott & Souza (2023)	Y	36	6	13	Exp 1a
Qiu et al. (2023)	Y	49	4	6	Exp 1
Qiu et al. (2023)	Y	57	8	3	Exp 2a
Qiu et al. (2023)	Y	58	8	3	Exp 2b
Scheibel & Indefrey (2023)	Y	40	4	15	Exp 1
Scheibel & Indefrey (2023)	Y	37	4	15	Exp 2
Nedergaard et al. (2023)	Y	222	8	1	
L. Chen et al. (2023)	Y	20	2	21	Exp 1a
L. Chen et al. (2023)	Y	44	2	21	Exp 1b
L. Chen et al. (2023)	Y	33	4	10	Exp 2
L. Chen et al. (2023)	Y	36	4	10	Exp 3
Durgin & Portley (2023)	Y	40	24	2	
M. S.-Y. Chen et al. (2023)	Y	40	2	8	Exp 1a
M. S.-Y. Chen et al. (2023)	Y	40	4	12	Exp 1b
M. S.-Y. Chen et al. (2023)	Y	40	2	8	Exp 2a
M. S.-Y. Chen et al. (2023)	Y	40	4	12	Exp 2b
Asaoka & Wada (2023)	Y	23	16	16	Exp 1
Asaoka & Wada (2023)	Y	23	16	16	Exp 2
Marzola & Cohen (2023)	Y	61	8	30	Exp 1
Marzola & Cohen (2023)	Y	87	8	30	Exp 2
Klassen et al. (2023)	Y	36	2	20	
Severijnen et al. (2023)	pseudo	80	32	3	
M. E. Yu et al. (2023)	Y	192	2	32	
M. E. Yu et al. (2023)	Y	50	2	32	
H. Yu et al. (2023)	Y	24	3	20	Exp 1a
H. Yu et al. (2023)	Y	48	2	60	Exp 2a
Yan et al. (2023)	Y	240	2	65	Exp 1
Yan et al. (2023)	Y	120	2	65	Exp 2
Yan et al. (2023)	Y	120	2	65	Exp 3
Fang et al. (2023)	Y	91	6	2	Exp 1

Continued on next page

Continued from previous page

Paper	Randomization	n_p	n_k	n_r	Notes
Fang et al. (2023)	Y	43	6	12	Exp 2a
Fang et al. (2023)	Y	30	6	12	Exp 2b
Fang et al. (2023)	Y	32	12	6	Exp 2c
Barnes et al. (2023)	Y	25	12	2	Exp 1
Barnes et al. (2023)	Y	48	12	2	Exp 2
Barnes et al. (2023)	Y	78	12	2	Exp 3
Barnes et al. (2023)	Y	75	16	3	Exp 4
Veldre et al. (2023)	Y	44	18	20	Exp 1a
Veldre et al. (2023)	Y	42	10	15	Exp 1b
Veldre et al. (2023)	Y	61	20	15	Exp 2a
Veldre et al. (2023)	Y	59	20	15	Exp 2b
Veldre et al. (2023)	Y	59	10	30	Exp 3a
Veldre et al. (2023)	Y	58	10	30	Exp 3b
Negen et al. (2023)	Y	12	3	28	Exp 3
Negen et al. (2023)	Y	12	3	28	Exp 4
Negen et al. (2023)	Y	12	3	23	Exp 5
Negen et al. (2023)	Y	12	3	42	Exp 6
Negen et al. (2023)	Y	12	3	43	Exp 7
Negen et al. (2023)	Y	12	4	16	Exp 8
Negen et al. (2023)	Y	12	3	27	Exp 9
Milligan et al. (2023)	Y	40	6	4	Exp 1
Milligan et al. (2023)	Y	41	6	4	Exp 2
Peker et al. (2023)	Y	20	16	2	
Babu et al. (2023)	Y	18	4	6	
Barbosa Escobar et al. (2023)	Y	300	2	10	Exp 1
Barbosa Escobar et al. (2023)	Y	300	2	10	Exp 2
Kershner & Hollingworth (2023)	Y	60	2	10	Exp 1
Kershner & Hollingworth (2023)	Y	20	2	10	Exp 2
Kershner & Hollingworth (2023)	Y	20	2	10	Exp 3
Kershner & Hollingworth (2023)	Y	20	2	8	Exp 4
Ramgir & Lamy (2023)	Y	96	4	12	Exp 1
Ramgir & Lamy (2023)	Y	48	8	12	Exp 2
Goodridge et al. (2023)	Y	12	9	300	
Lavelle et al. (2023)	Y	55	12	5	Exp 1
Lavelle et al. (2023)	Y	54	8	4	Exp 2
Gutzeit et al. (2023)	Y	40	3	12	Exp 1
Gutzeit et al. (2023)	Y	40	3	12	Exp 2
Narhi-Martinez et al. (2023)	Y	28	3	5	Exp 1

Continued on next page

Continued from previous page

Paper	Randomization	n_p	n_k	n_r	Notes
Narhi-Martinez et al. (2023)	Y	28	3	21	Exp 2
Narhi-Martinez et al. (2023)	Y	56	3	21	Exp 3
Bogon et al. (2023)	Y	25	4	16	Exp 1
Bogon et al. (2023)	Y	45	4	16	Exp 2
Ziaka & Protopapas (2023)	Y	42	8	7	
Cui et al. (2023)	Y	37	12	12	Exp 1a
Cui et al. (2023)	Y	36	12	12	Exp 1b
Cui et al. (2023)	Y	35	12	12	Exp 3b
Cui et al. (2023)	Y	44	4	37	Exp 2
Cui et al. (2023)	Y	41	4	37	Exp 3c
Sears et al. (2023)	Y	60	6	8	Exp 1
Sears et al. (2023)	Y	60	6	8	Exp 2
Chang et al. (2023)	Y	24	3	120	Exp 1
Chang et al. (2023)	Y	24	3	120	Exp 2
Nguyen & van Buren (2023)	Y	50	4	16	Exp 1,4
Nguyen & van Buren (2023)	Y	100	4	16	Exp 2,3,6
Manzone & Welsh (2023)	Y	24	4	15	Exp 1
Manzone & Welsh (2023)	Y	23	4	148	Exp 2
Garnier-Allain et al. (2023)	Y	21	8	7	Exp 1
Garnier-Allain et al. (2023)	Y	21	8	36	Exp 1
Garnier-Allain et al. (2023)	Y	21	28	10	Exp 1
Garnier-Allain et al. (2023)	Y	21	28	13	Exp 2
Garnier-Allain et al. (2023)	Y	21	4	96	Exp 2
Bollini et al. (2023)	pseudo	42	4	13	Exp 1
Bollini et al. (2023)	pseudo	42	4	13	Exp 2
Bollini et al. (2023)	Y	42	2	27	Exp 3
B. E. Wirth et al. (2023)	Y	31	4	9	Exp 1
B. E. Wirth et al. (2023)	Y	36	4	9	Exp 2
Kang & Longo (2023)	Y	20	2	18	
Siqi-Liu & Egner (2023)	pseudo	30	4	15	Exp 1
Siqi-Liu & Egner (2023)	pseudo	83	8	4	Exp 2
Siqi-Liu & Egner (2023)	pseudo	170	8	3	Exp 3
Schaaf et al. (2023)	Y	48	4	10	Exp 1
Schaaf et al. (2023)	Y	48	4	10	Exp 2
Van Geert & Wagemans (2023)	Y	283	2	27	Task 1
Van Geert & Wagemans (2023)	Y	283	2	82	Exp 2
Van Geert & Wagemans (2023)	Y	283	2	66	Exp 3
Lerebourg et al. (2023)	Y	43	4	10	Exp 1

Continued on next page

Continued from previous page

Paper	Randomization	n_p	n_k	n_r	Notes
Lerebourg et al. (2023)	Y	43	4	16	Exp 2
Lerebourg et al. (2023)	Y	80	4	10	Exp 3
Lerebourg et al. (2023)	Y	80	4	10	Exp 4
Lee & Cho (2023)	Y	32	4	24	Exp 1
Lee & Cho (2023)	Y	32	4	24	Exp 2
Schmalbrock et al. (2023)	Y	32	8	4	Exp 1a
Schmalbrock et al. (2023)	Y	22	8	4	Exp 1b
Schmalbrock et al. (2023)	Y	40	8	4	Exp 2a
Schmalbrock et al. (2023)	Y	40	8	4	Exp 2b
R. Wirth et al. (2023)	Y	24	4	40	Exp 1
Honda et al. (2023)	Y	127	10	5	Task 1
Honda et al. (2023)	Y	127	4	24	Task 3
Woźniak et al. (2023)	Y	184	18	2	
Wentura et al. (2023)	Y	58	4	16	Exp 1
Wentura et al. (2023)	Y	39	4	16	Exp 2
Wentura et al. (2023)	Y	57	4	16	Exp 3
Wentura et al. (2023)	Y	38	4	16	Exp 4
Colvett et al. (2023)	Y	78	8	18	Exp 1
Colvett et al. (2023)	Y	65	8	18	Exp 2
Colvett et al. (2023)	Y	66	8	18	Exp 3
Zhang et al. (2023)	Y	30	4	12	Exp 1
Zhang et al. (2023)	Y	28	4	12	Exp 2
Zhang et al. (2023)	Y	28	4	12	Exp 3
Zhang et al. (2023)	Y	28	4	12	Exp 4
Eggleston et al. (2023)	Y	90	2	50	Exp 1
Eggleston et al. (2023)	Y	90	4	25	Exp 2
Hoversten & Martin (2023)	Y	56	2	62	
Yarrow et al. (2023)	pseudo	20	18	7	
Cheng et al. (2023)	Y	18	2	90	Exp 1
Cheng et al. (2023)	Y	21	4	5	Exp 2

The idea of randomization has been considered in multiple types of cognitive experiments, including those about action control, bilingualism, emotional expression, and so on. Discussions about the power of studies in these papers exclusively focused on participant sample sizes.

To obtain central ranges for the design parameters, we calculated the first (25%), second (50%, or median) and third (75%) quartiles for each parameter. This analysis resulted in sample sizes of 25, 40, and 60 ($n_p \in \{25, 40, 60\}$), factor level values of 2, 4, and 8 ($n_k \in \{2, 4, 8\}$), and repetition-per-condition values of 6, 12, 24 ($n_r \in \{6, 12, 24\}$). As these reviewed studies rep-

resented typical settings of current cognitive experiments, central ranges of design parameters should cover a good part of the design parameter space. Fully crossing these parameters yielded 27 unique study designs. We refer these 27 designs as representative experimental designs.

Across studies being reviewed, almost all provided reasons for choosing sample size based on the G-power calculation results, yet none of them discussed higher power gains from their randomization strategies. In Chapter 3, we will purpose a design-based method that can improve power in these representative experimental designs.

Chapter 3

Improving Power for ANOVA in One-Factor Designs with Permuted Subblock Randomization (PSR)

3.1 Background

Chapter 2 outlined the potential of randomization to power gain in repeated-measure within-subject experiment designs, and provided representative experiment settings. In this chapter, we show how a simple improvement to the randomization procedure for experiments having within-subject factors can substantially improve power by controlling time-varying fluctuations. Since randomization happens in stimuli sequence management, let us start with further looking at the simple restricted randomization sequences, and understand why it is suboptimal to power gains.

In clinical trials where between-subject experiments are commonly implemented, randomization algorithms have been developed to ensure patients are allocated to treatment/control arms in a balanced manner as the study unfolds. It is often the case that the clinical researcher does not know in advance when the target sample size will be reached; for example, because enrollment happens in parallel across multiple testing sites as patients become available. Given the study may be completed (or terminated) unexpectedly, there is motivation to preserve balance among the number of patients across treatment arms as they are enrolled. The easiest solution would be to simply alternate allocation sequentially among the treatment arms; for instance, for a study with treatments A and B, the allocation would alternate sequentially between them (e.g., $ABABABABAB\ldots$). However, the predictable nature of this sequence would enable the experimenter to know in advance the identity of the next allocation, opening up the possibility for selection bias (Blackwell & Hodges, 1957). An algorithm in common use that reflects a compromise between balance and predictability is *Permuted-Block Randomization* (Altman & Bland, 1999; Efron, 1971; Hill, 1952; Matts & Lachin, 1988). In the simplest version, a study with n_k treatment arms is divided into n_b blocks of n_p patients, where n_p is a multiple of n_k.

For instance, if there are four treatment arms A, B, C, and D, then n_p may be four or eight or some larger number that is a multiple of four. A balanced set of treatments is allocated to each block and then a permutation of those treatments is chosen at random, independently from the sequence chosen for other blocks. For example, if $n_p = 8$, then two As, two Bs, two Cs, and two Ds are chosen and randomly ordered to form a sequence for that block, e.g., $AADBCCBD$.

The *simple restricted randomization* approach used in nearly all laboratory experiments can be seen as a special case of Permuted-Block Randomization where there is a single block spanning the full set of trials allocated to each participant. For instance, in a study with four within-subject experimental conditions A, B, C, and D, and with five observations in each condition, the researcher will create a presentation order for each participant by permuting the full set of $4 \times 5 = 20$ symbols. But as mentioned, typical motivations for randomization are not to increase power. Experimenters also often organize trials into blocks, but mainly to allow participants to have a rest break during a long experimental session.

We adapt the Permuted-Block Randomization approach from clinical trials into the experimental context and estimate the potential improvement to power across a set of representative designs from Chapter 2. Experimenters often divide trials into blocks to give participants rest breaks during a session. We introduce an algorithm, *Permuted-Subblock Randomization (PSR)*, that further subdivides these blocks into balanced subblocks and permutes the order of levels within each subblock. This approach smooths out imbalances over time that can mask effects of interest. We used Monte Carlo simulation to estimate power improvements under four hypothetical scenarios of time-varying errors. Depending on the design and the structure of the dependencies, PSR boosted power over simple restricted randomization between 4% and 45%, with a median boost of about 13%. We supply an applied example showing how to use the R package `explan` to implement PSR when planning a hypothetical experiment.

3.2 Achieving balance over time with Permuted-Subblock Randomization (PSR)

Although researchers are typically well aware of time-varying fluctuations, they rarely take them into account when ordering the sequence of levels over time. To see why it matters, let us consider an example experiment consisting of response time measurements with a single four-level within-participant factor with levels A, B, C, and D, representing four levels in the experiment. Let us also assume that we collect sixteen measurements for each participant, four for each condition.

Usually, before performing an ANOVA a researcher would calculate means for each subject in each condition over the four measurement occasions in that condition, and then submit the means rather than the raw observations to ANOVA. The sensitivity of this analysis depends on the F ratio, a ratio of mean squares, e.g., MS_{treat}/MS_{error}, where MS_{treat} represents the treatment variance (i.e., the variance introduced by the independent variable) and MS_{error} is the residual sum of squares (RSS). Both MS_{treat} and MS_{error} are scaled by their degrees of freedom. In the ANOVA framework, this ratio amounts to *signal plus noise* divided by *noise*,

such that the ratio will approach 1 for a null effect and increase beyond 1 as the signal variance increases. We could potentially increase the ratio by designing the experiment in a way that helps more clearly segregate the signal from the noise.

The errors for each participant in an ANOVA model are usually considered to be independent over time. But due to time-varying fluctuations, it is less likely the case. For example, consider three characteristic ways they can be time dependent (Figure 3.1): exponential decay, random walk, and pink noise. Exponential decay characterizes a situation where there is a "practice" or "learning" effect such that measurements decline toward some asymptotic value, a pattern often seen with response times (Newell & Rosenbloom, 1981). A random walk situation can be thought of as involving the waxing and waning of attention, such that measurements are determined by a slowly changing attention state. This results in a more undulating type of pattern. Finally "pink noise", a type of noise that is characteristic of self-organizing systems, has also been detected in the sequence of responses in behavioral experiments (Gilden et al., 1995). The cognitive processes that might give rise to such a pattern are difficult to characterize, but the pattern is said to have a "fractal" signature, meaning that the same undulating pattern is present at multiple time scales (Holden, 2005). All three of these patterns will give rise to positive *temporal autocorrelation*, the property whereby a series is positively correlated with lagged versions of itself. More simply, this may be thought of as errors being more similar to one another the closer they are taken in time.

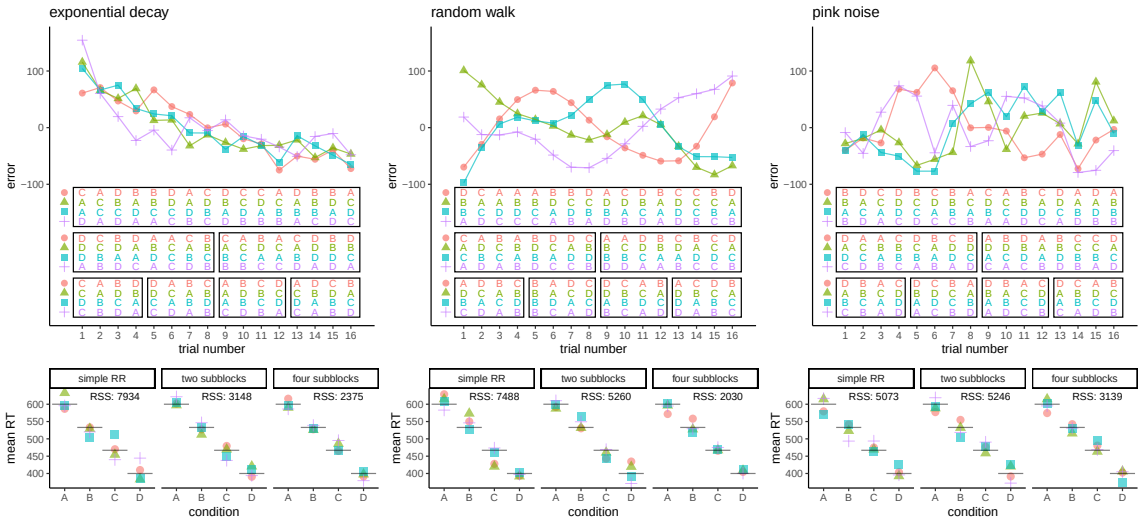

Figure 3.1: Three characteristic patterns of time-dependent errors: exponential decay, random walk, and pink noise. The top three charts show hypothetical patterns of response time errors (vertical axis) for four different participants, distinguished by shape (circle, square, triangle, plus) and color. Just below the curves are possible condition labels (A, B, C, D) for each subject and trial (horizontal axis), organized into subblocks (indicated by boxes). Three possible designs are represented: simple restricted randomization ("simple RR": top four rows of labels in a single box), two subblocks (next four rows in the two side-by-side boxes) and four subblocks (bottom four rows of labels in four boxes). The charts at the bottom show the observed mean RTs for each participant in each design plotted against the true condition means (background lines). RSS = Residual Sum of Squares.

To sequence the order of levels, researchers generally apply simple restricted randomization to

the entire sequence of balanced level labels, and do so independently and separately for each participant. Let us consider how this approach might be suboptimal for power. Assume that the true response time means across levels are known to be 600, 533, 467, and 400 milliseconds in conditions A, B, C, and D respectively. To keep our example simple, let us further assume that the participants are from a homogeneous population where there are no individual differences (e.g., by-subject random intercept and random slope variances are zero). Given this assumption, the observed level means for each participant in each scenario under simple restricted randomization (Figure 3.1, bottom "simple RR" charts) deviate from their true values due to trial-level noise alone. Under the exponential decay scenario, the mean for participant 3 (the blue square in Figure 3.1) is lower than it should be in level B (504 vs the true mean of 533) and much higher than it should be in level C (514 vs the true mean of 467). The reason for this is that the unconstrained randomization resulted in all four C trials appearing earlier in the session (positions 2, 3, 5, and 6) than all four B trials (positions 8, 12, 13, and 14) for this participant. Thus, the C trials tended to appear when RTs were artificially high and B trials when RTs were artificially low. This lopsided temporal distribution results in the mean for C being higher than the mean for B for this subject, masking a true effect going in the opposite direction.

The above analysis suggests that it is the imbalance of levels over certain intervals of the entire sequence that can potentially mask effects of interest. We can avoid this masking by striving to maintain balance through the session, so that individual levels do not bunch together where errors are momentarily high or low. We could easily achieve this by cycling through the sequence of levels trial-by-trial (e.g., $ABCDABCD\ldots$), but we must also avoid creating a sequence that would allow the participant to predict the level of the next trial. This need to preserve balance over time while avoiding predictability is the exact problem that Permuted-Block Randomization solves in the clinical context, and we can apply it to trials within an experiment just as it is applied to patients within a clinical study. We can maximize balance over time by subdividing the fully balanced block of level labels into balanced subblocks, each containing one repetition of each level. We can then avoid perfect predictability by permuting the order of conditions within each subblock and then concatenate the subblocks back together to form the final sequence. We refer to this algorithm as *Permuted-Subblock Randomization* (PSR). The PSR algorithm is identical to Permuted-Block Randomization but the application is different, in that it is applied to a sequence of trials in an experiment rather than to sequence of patients in a clinical study.[3.1]

Had we designed our experiment using PSR with four subblocks instead of simple restricted randomization, we would have avoided masking the main effect and would achieve a smaller residual sum of squares (e.g., RSS = 2375 vs 7934 for the exponential decay scenario), thus improving power. Still, this four subblock situation could be seen as too predictable for human participants, who, once familiar with the set of levels, might still be able to predict the identity

[3.1] We opted for the term 'subblock' instead of 'block' to avoid confusion, because in the experimental context the latter term is usually used to refer to a set of trials that a participant completes in a single series without pausing for a break.

of the next trial with some accuracy. But there is an intermediate design strategy between maximally constrained four subblock version and simple restricted randomization: one with two subblocks. We put two of each label in each subblock, pull labels out in a random order, and concatenate the results. Applying this strategy would, on average (but not deterministically), reduce the residual sum of squares across all three scenarios compared to simple restricted randomization, thus improving our power while yielding a sequence that to participants would still seem mostly random. In the exponential decay scenario, going from simple restricted randomization to PSR with two subblocks decreases the RSS from 7934 to 3148; however, for the pink noise scenario, it actually increased the RSS (from 5073 to 5246). This just shows that the power advantage is stochastic rather than deterministic: in the long run, using more subblocks is likely to increase power, but it may or may not do so for a particular dataset.

3.3 The Permuted-Subblock Randomization algorithm

We are now ready to more generally define our *Permuted-Subblock Randomization* (PSR) algorithm for one-factor designs with an arbitrary number of levels and repetitions per level. The algorithm consists of three steps that we refer to as *split*, *permute*, and *concatenate*. Let n_k be an integer representing the number of levels of the independent variable (with $n_k \geq 2$) and let n_r be an integer representing the number of repetitions for each level, where $n_r \geq 1$. For instance, in the above example there are four levels ($n_k = 4$) and four repetitions of each ($n_r = 4$). Each level must be denoted by a distinct label (e.g., A, B, C, D). The full set of labels to be attached to all trials in a single session of the experiment will be of size $n_k \times n_r$.

In the first *split* step, the labels are to be divided into subblocks following two rules. First, each subblock must have at least one and at most n_r of each label. Second, in a given subblock, there should be an equal number of each label (e.g., three of each, but not three As, two Bs, one C and two Ds). Normally, all subblocks would have identical frequency distributions of labels (e.g., each subblock having three of each label) but deviation from this practice may sometimes be desirable. The maximum possible number of subblocks is always n_r, where each subblock contains one label for each condition, and the minimum number of subblocks is one, where there is a single subblock with all n_r repetitions. This latter situation is equivalent to simple restricted randomization. The set of possible subblocks, S_{n_r} is defined as the set of integer divisors of n_r. For example, for an experiment with $n_r = 12$ (i.e., 12 repetitions of each level of the independent variable), the number of subblocks n_s is any of the possible values in the set S_{12}, $n_s = \{1, 2, 3, 4, 6, 12\}$. Going forward, we refer to different subblock configurations using a subscript; thus, PSR_1 is single-subblock (i.e., simple restricted) randomization, PSR_2 has two subblocks, and so forth.

The *permute* and *concatenate* steps are straightforward. In the *permute* step, labels within each subblock are randomly permuted into a new sequence without constraint. The resulting strings of labels are then concatenated into a single long string to form the level order for the experiment. In a later section of this chapter, we walk through an applied example showing how to use the accompanying R package `explan` to implement this algorithm. In the next

section, we describe the Monte Carlo simulations used to evaluate the performance of PSR.

3.4 Methods

To assess the long-run Type I error and power of the PSR algorithm, we conducted Monte Carlo simulations in which we generated data from hypothetical experiments with time-dependent errors (i.e., allowing time-varying fluctuations). Our simulations were run in R version 4.3.2 (R Core Team, 2023) with add-on packages `numbers` version 0.8.5 (Borchers, 2022a), `pracma` version 2.4.2 (Borchers, 2022b), and `tuneR` version 1.4.5 (Ligges et al., 2023). The full code for the simulation along with simulation results is available at the Open Science Framework (OSF) and can be accessed at https://osf.io/w6tej.

3.4.1 Determining representative study design parameters

To cover the *representative experiment environments*, design parameters were provided by literature review results in Chapter 2. That is, 27 unique designs by fully crossing sample sizes of 25, 40, 60 ($n_p \in \{25, 40, 60\}$), factor level values of 2, 4, and 8 ($n_k \in \{2, 4, 8\}$), and repetition-per-condition values of 6, 12, 24 ($n_r \in \{6, 12, 24\}$).

As already noted, once n_r is been determined, the set of possible subblocks S_{n_r} (of which n_s is the set size) can be calculated by determining the integer divisors of n_r. The designs above allowed for variable numbers of n_s depending on the number of repetitions per level: four for $n_r = 6, n_s \in \{1, 2, 3, 6\}$; six for $n_r = 12, n_s \in \{1, 2, 3, 4, 6, 12\}$; and eight for $n_r = 24, n_s \in \{1, 2, 3, 4, 6, 8, 12, 24\}$, yielding 162 total combinations of designs and subblocks.

3.4.2 Data-generating process

Unlike the simplified example in the above section, we used a more realistic data generating process that included individual differences (by-subject random intercepts and random slopes). To keep things simple, our focus in these simulations is on a situation where stimuli are treated as fixed rather than random. The evaluation of performance of PSR where stimuli are treated as a random factor is beyond the scope of the current investigation.

For the data-generating process, the response time Y_{ijt} for participant i in condition j on trial number t is given by Formula 3.1,

$$Y_{ijt} = \mu + S_{\mu i} + \beta_j + S_{\beta ij} + e_{it} \tag{3.1}$$

where μ is the grand mean; β_j is the main effect of condition j, $S_{\mu i}$ and $S_{\beta i}$ are the random intercept and random slope for participant i, respectively; and e_{it} is the error for participant i on trial t with variance σ^2. As in the standard ANOVA parameterization, $\sum_j \beta_j = 0$ and for any participant i, $\sum_j S_{\beta ij} = 0$.

We defined the variance components $S_{\mu i}$, $S_{\beta ij}$, and e_{it} to approximately match the empirical distribution documented in a meta-analysis of variance components in psycholinguistics (Barr et al., 2013), where the error variance was the largest component, the by-subject random intercept variance was about 35% of the error variance, and the by-subject random slope variance was about 11% of the error variance.[3.2] As a computational example, fixing the trial level variance at 1 with 24 subjects and 48 trials per subject yields an expected sum of squares for trial-level error of $SS_{trial} = 24 * 48 = 1152$. The random intercept variance was set to 35% of this size ($SS_{participant} = 403.20$) and the random slope variance to 11% ($SS_{\text{treat x participant}} = 126.72$). The intercept parameter μ was drawn from a uniform distribution from -1 to 1.

A main effect can manifest in a variety of different ways. To be agnostic about the shape of the main effect, we set the target sum of squares for the main effect, SS_{treat}, to a fixed value based on a target effect size of η^2, and let the cell means vary randomly. We define η^2 as $SS_{treat}/(SS_{treat} + SS_{error})$. We estimated power curves for each design by varying η^2 from zero to some target value in six steps (seven values).

To maximize sensitivity, we used different ranges for η for each triplet of design parameters $\langle n_p, n_k, n_r \rangle$. We determined the range for each setting using Monte Carlo simulation. For each setting, we made an initial guess about the value of η^2 that would yield 50% power, and then ran 1,000 Monte Carlo runs with that value of η^2. We adjusted the value of η^2 until the power estimate fell within the 95% Agresti-Coull confidence interval for a process with $p = .5$ and 1,000 samples. Once determined, that value of η^2 then became the median value for the range for that design. The η^2 values are presented as supplementary materials of the published version of this chapter, available at https://supp.apa.org/psycarticles/supplemental/met0000717/met0000717_supp.html.

3.4.3 Time-varying error patterns

The errors e_{it} for each participant were simulated to reflect four different time-varying fluctuation patterns found in real world data: (1) a learning effect, showing a typical "exponential decay" pattern; (2) Gaussian random walk, corresponding to the waxing and waning of attention over an experimental session; (3) $\frac{1}{f^\alpha}$ scaling, or so-called "pink noise" (Gilden et al., 1995); and (4) a mixture of patterns (1) to (3). For examples of patterns (1) to (3), see Figure 3.1. For comparison, we also included a fifth (baseline) scenario in which all errors were temporally independent. The error sequence generated for each participant was normalized to have a standard deviation of 1.

The exponential decay scenario followed the function $e^{-(\lambda + S_{\lambda i})\frac{t}{N_t}}$ where t is trial number and $N_t = N_k N_r$, the total number of trials. The fixed λ parameter determines the steepness of the slope for the population. For each dataset, this value was drawn from a uniform distribution with range $[\log_e 2, \log_e 15]$. So that the change was not identical across subjects, the fixed λ parameter was offset by a by-subject random effect for participant i, $S_{\lambda i}$, which was drawn from

[3.2] See Table 4 in the Appendix of Barr et al. (2013) at https://talklab.psy.gla.ac.uk/simgen/realdata.html

a normal distribution with a standard deviation of .8. So that the curves were not completely smooth, each was mixed with 10% Gaussian white noise.

The Gaussian random walk scenario was based on the `stat_gp()` function (Thul et al., 2021), which follows the numerical method developed by Shinozuka & Deodatis (1991). The function takes two arguments: σ, which represents the standard deviation, and γ, which represents the correlation length. In our simulations, we used values of $\sigma = 1$ and $\gamma = 2$, equivalent to Scenario 4 in Thul et al. (2021), for which technical details are provided in their Appendix.

Pink or $\frac{1}{f^\alpha}$ noise was created using the (non-exported) internal function `TK95()` of R package `tuneR` (Ligges et al., 2023). For each dataset, the fixed parameter α was drawn from an uniform distribution, $\alpha \sim U(.8, 1.2)$.

The "mixed" condition was created to reflect that all of the above processes are likely operating for individual participants but in different proportions, and was created by mixing together the previous three patterns with each pattern randomly weighted. For each participant in each dataset, three weights were randomly generated numbers from a uniform distribution between zero and one (`runif(3)` in R), which were then normalized to sum to one. The error pattern for that participant was then the weighted sum of the three patterns, and different participants in the same dataset would have different weighted sums.

3.4.4 Analysis

Combining our 162 design-subblock combinations further with our five error scenarios (independent, exponential decay, gaussian random walk, pink noise, and mixed) and seven steps of effect size (η) yielded a total of 5,670 possible parameter settings for the simulations. To simplify presentation of the results, out of the 27 possible designs we chose three as "focal" designs, representing the smallest (25 participants, $n_k = 2$, $n_r = 6$), largest (60 participants, $n_k = 8$, $n_r = 24$), and medium-sized or most typical design (40 participants, $n_k = 4$, $n_r = 12$). For the focal designs, we ran simulations at all possible subblock settings. For the remaining 24 non-focal designs, we reduced the number of simulations by only running them with one of three subblock settings: (1) one subblock (baseline); (2) the maximum number of subblocks, henceforth $\max(S_{n_r})$; and (3) the number just below the maximum number of subblocks, henceforth (S_{n_r}). For example, for $n_r = 24$, where $S_{24} = \{1, 2, 3, 4, 6, 8, 12, 24\}$, we ran simulations with only 1, 12, and 24 subblocks.

For each of these 3,150 hypothetical experiments, we estimated Type I error and power by conducting 10,000 Monte Carlo runs, with each run consisting of randomly generating a dataset based on the parameters and then analyzing it. We performed an Analysis of Variance (ANOVA) on each dataset using the built-in `aov()` function in R. We first calculated the means for each condition for each participant and then submitted the means to analysis using the model formula `Y ~ A + Error(id)` where `Y` is the DV, `A` is a factor representing the independent variable, and `id` is a factor identifying individual participants. The p-values were extracted from the `summary()` table of results, with $\alpha = .05$.

The proportion of datasets yielding a statistically significant main effect was reported as an estimate of the Type I error rate and statistical power for the PSR algorithm. The Type I error rate is estimated by the proportion of significant effects detected when η is zero (i.e., false positive rate). When $\eta > 0$, the proportions of effects determined to be significant provide an estimate of power (true positives).

3.5 Results

3.5.1 Power

To estimate power, for each of the 2,700 cases where $\eta > 0$ we calculated the proportion of runs (out of 10,000) that yielded a significant main effect. Then, for each of the 135 cases yielded by combining the 27 designs with the 5 error structures, we estimated the power advantage of PSR by calculating the percent increase in power for $\text{PSR}_{\max(S_{nr})}$ and $\text{PSR}_{(S_{nr})}$ relative to PSR_1 (simple restricted randomization) at each of the six non-zero values of η for that case.[3.3] We then extracted the maximum power advantage observed across all six values of η for $\text{PSR}_{\max(S_{nr})}$ and $\text{PSR}_{(S_{nr})}$.

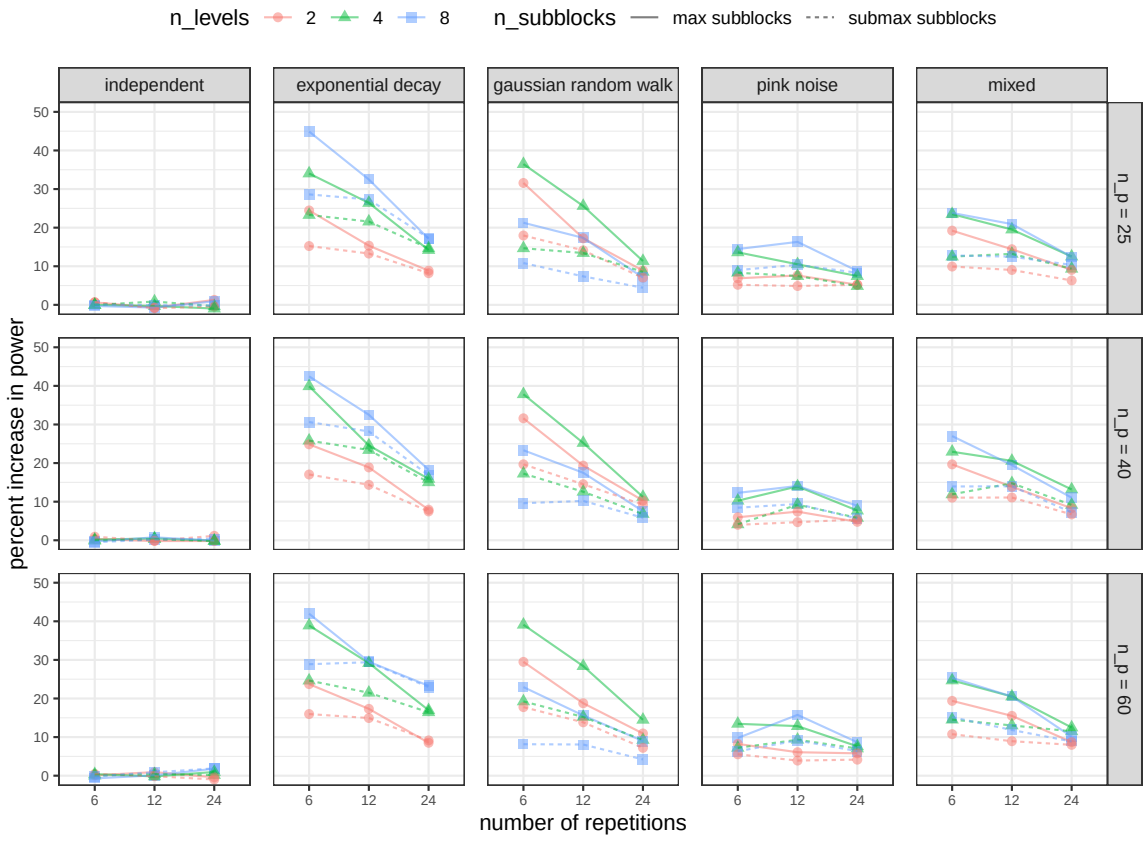

Figure 3.2: Median percent increase in power from baseline (simple restricted randomization) across all six non-zero levels of η by number of participants (n_p), error structure, number of repetitions (n_r), number of factor levels (n_k), and number of subblocks (max, submax).

Figure 3.2 shows the median power broken down by case. As expected, when the error structures

[3.3] To illustrate, if power increased from .20 to .25, this would be characterized as a 25% increase, because $100 \times \frac{(.25-.20)}{.20} = 25$.

were independent, the power advantage for using the maximum or sub-maximum number of subblocks was negligible, with estimated increases in power of just 0.1% and 0.2% respectively.

Across all four cases with non-independent errors and both numbers of subblocks, power gains ranged from 3.9% to 44.9%, with a median power gain of 13.3%. For designs using $\text{PSR}_{\max(S_{nr})}$, the greatest power gains were in the exponential decay scenario with a range from 7.9% to 44.9%, followed by Gaussian random walk (7.1% to 39.1%). Gains were more modest for the mixed scenario (8.3% to 27.0%), and smallest for pink noise (4.8% to 16.3%). For designs using $\text{PSR}_{(S_{nr})}$, gains were lower overall but ranked in the same order, with exponential decay showing largest gains (7.5% to 30.6%), followed by Gaussian random walk (4.2% to 19.7%), then mixed (6.3% to 15.2%), and finally, pink noise (3.9% to 10.4%).

Some further observations from Figure 3.2 warrant mention. First, the power gains seem to be largely independent of the number of participants, n_p, which makes sense since the subblocking mechanism that produces the power gains operates at the participant level. Next, power gains seem most impressive when there are fewer repetitions (i.e., when n_r is small). Beyond this, there seems to be a complex relationship between n_r, n_k (the number of factor levels) and the type of error structure in determining the amount of gain. For example, for the exponential decay structure, gains are largest for the design with eight factor levels, followed by four and then two; but for the Gaussian random walk structure, gains are largest for four, followed by two, then by eight. For the mixed structure, the gains for four and eight levels are roughly the same and both exceed the two level case.

Figure 3.3 presents full power curves for the PSR algorithm across all the three focal designs, where we ran all possible numbers of subblocks, not just the maximum and submaximum. (Power curves for all non-focal designs are provided in the supplementary materials at https://supp.apa.org/psycarticles/supplemental/met0000717/met0000717_supp.html). When no time-varying dependency was present in the data ("independent" error scenario), there was no discernible effect of the number of subblocks on power. Across all four time-dependent scenarios, the PSR algorithm consistently increased power in proportion to the number of subblocks.

3.5.2 Type I error rates

Although the demonstrated power gains are impressive, it is important to confirm that these gains are not achieved at the expense of the control of the false positive rate. To estimate Type I error, we calculated the proportion of simulation runs with a statistically significant main effect (with $\alpha = .05$) for each of the 450 cases where $\eta^2 = 0$. To determine whether Type I error rates were close to the nominal $\alpha = .05$ level, we checked whether observed rates fell within the 99.9% Agresti-Coull confidence interval for a binary process with probability .05 and 10,000 samples, which was [0.0433, 0.0577]. Cases that fall below this interval are deemed *conservative* (i.e., have inflated false negatives); cases that fall above this interval are deemed *anti-conservative* (i.e., have inflated false positives); and finally, cases that fall within this interval are deemed *nominal* (i.e., are calibrated to the α level).

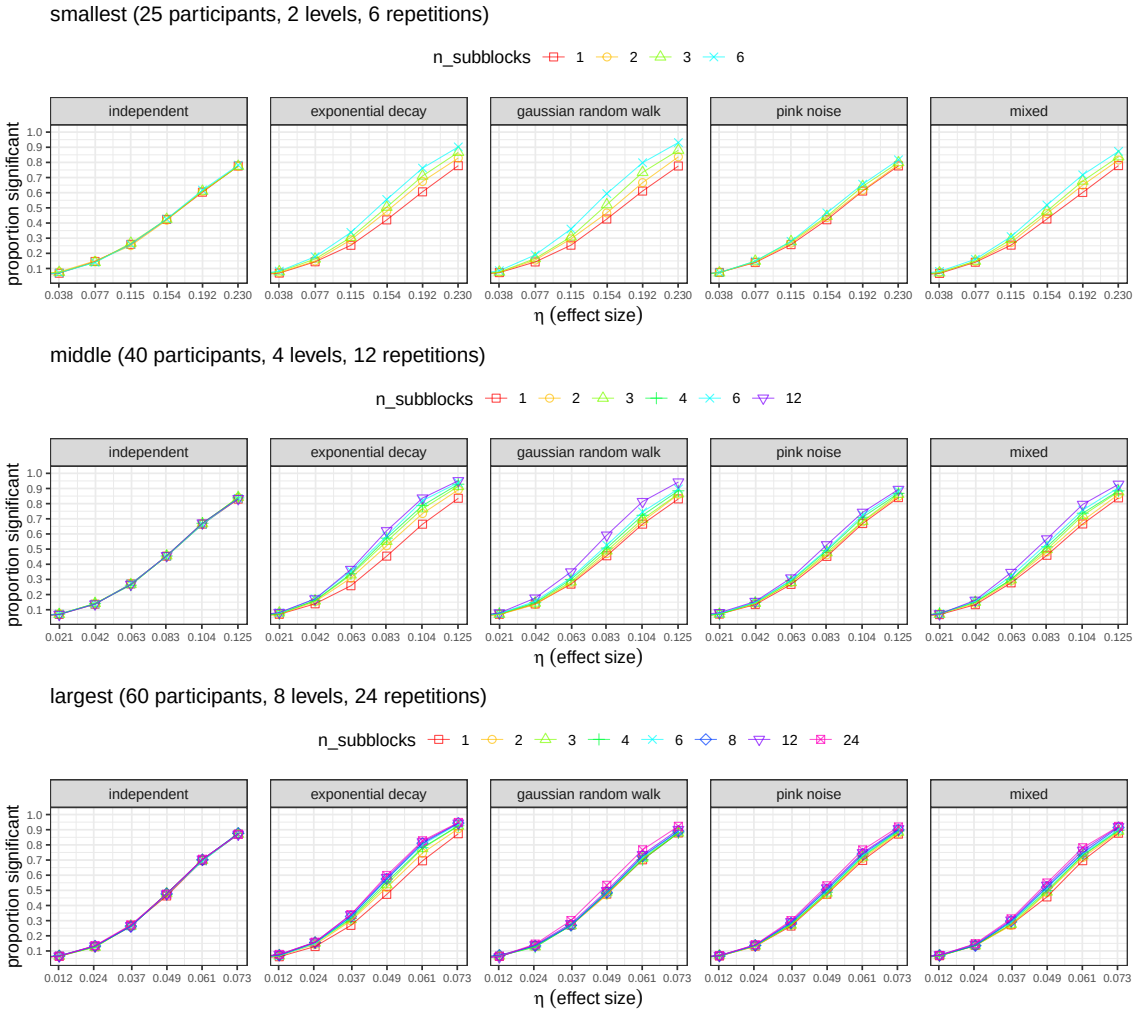

Figure 3.3: Power curves ($\alpha = .05$) for the three focal designs plotted by number of subblocks, trial-level error variance scenario, and effect size (η). Each data point is estimated from 10,000 Monte Carlo simulations.

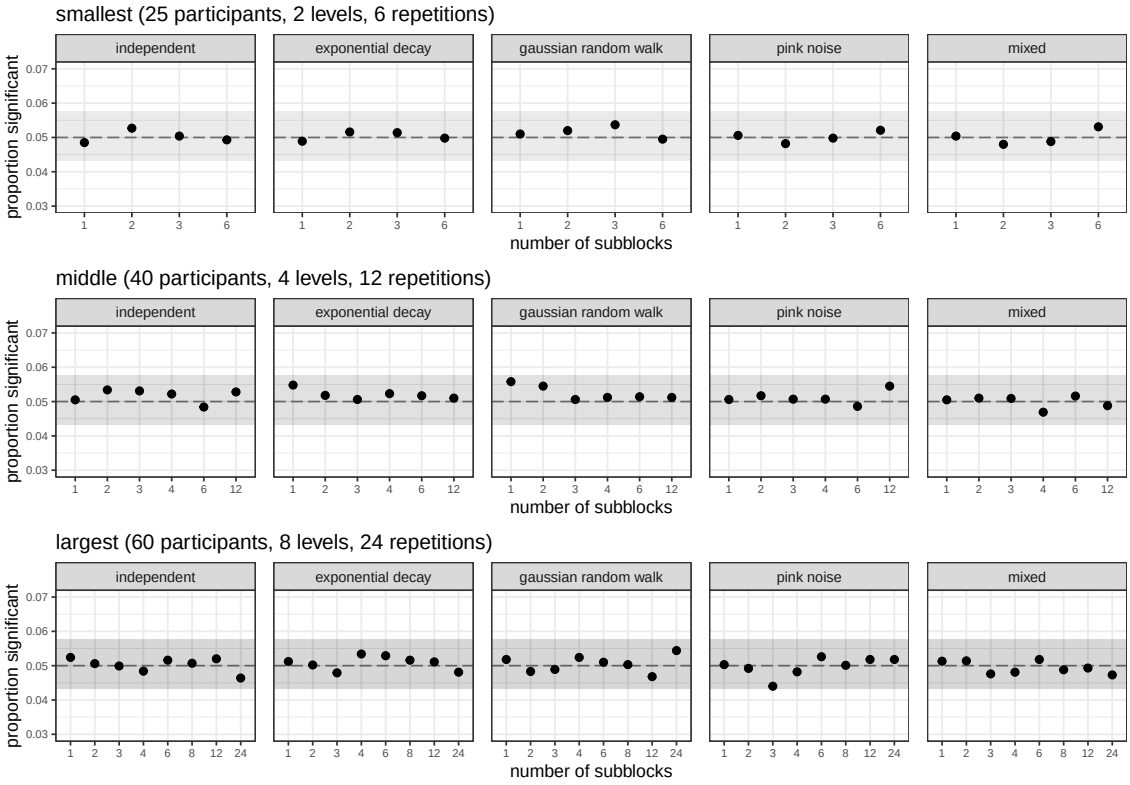

Figure 3.4: Type I error rates ($\alpha = .05$) for the three focal designs: small (top row), middle (middle row), and large (bottom row), broken down by number of subblocks, error variance scenario, and effect size (η). Each data point is estimated from 10,000 Monte Carlo simulations. The shaded region represents the 99.9% Agresti-Coull confidence interval for 10,000 random samples from a process with a "success" rate of .05.

As a baseline, we evaluated the overall performance of the field-standard of simple restricted randomization (i.e., PSR_1). As expected, for the 27 cases with no time-varying error pattern (i.e., independent structure) the range of Type I error rates was [0.0477, 0.0555], which falls within the reference range and thus reflects nominal performance. More importantly, the range of Type I error rates for the remaining 135 cases with a single subblock but where there was a time-varying error pattern, the Type I performance was also nominal, with a range of [0.0456, 0.0570]. This latter observation counters prevailing wisdom that time-varying error structure inflates Type I error rates, further supporting the claim of Thul et al. (2021) that appropriate randomization is an effective defense against autocorrelated errors.

The key question, however, concerns the Type I error performance of cases with multiple subblocks. Figure 3.4 presents results for the three focal cases; remaining cases are available in the supplementary materials. Fortunately, the PSR algorithm maintained nominal error rates across all 315 cases with multiple subblocks. For the 252 cases where errors were non-independent, the Type I error range was [0.0440, 0.0560]. For the 63 cases where errors were independent, the range was [0.0438, 0.0539], indicating that the PSR algorithm is safe to use even when there is no time-dependent error structure in the data.

3.6 Applied example: Using PSR in experiment planning

Let us consider an example of implementing PSR for a hypothetical experiment that uses our `explan` package for the R programming language (R Core Team, 2023). As our hypothetical example, we present a version of the classic Stroop paradigm (Stroop, 1935) in which each participant sees a series of four color words ("blue", "red", "green", and "yellow") presented in four different font colors. The participant's task is to respond with the font color of each presented word. The standard finding is that people take longer and produce more errors naming the font color when it mismatches the meaning of the word (*incongruent* condition) than when it matches (*congruent* condition). For example, on average people would take longer and make more mistakes responding "yellow" to the word "blue" printed in a yellow font than to respond "yellow" to the word "yellow" printed in yellow font.

In our hypothetical experiment, the color words will be presented to participants one by one, and our independent variable *congruency* has two levels ($n_k = 2$, congruent and incongruent). Half of the stimuli are to be congruent and half are to be incongruent, with 24 repetitions in each level ($n_r = 24$) for a grand total of 48 trials per participant. We will consider two different functions for applying PSR using `explan`.

The first method would be to directly use the `psr()` function from `explan`, which will create a vector containing the sequence of levels for a single participant. The `psr()` function takes three arguments: (1) `levels`, a vector containing the levels of the independent variable, which has n_k elements; (2) `n_subblocks`, the desired number of subblocks n_s; and (3) `n_reps`, the number of times each level is repeated for each participant (n_r). To decide on the value of `n_subblocks`, it is useful first to consider the set of possible subblocks using the function

`possible_subblocks()`. For $n_r = 24$, there are eight possible values for n_s.

```
## NB: first install the explan package using:
## remotes::install_github("dalejbarr/explan")
library("explan")


possible_subblocks(24)
```

```
[1]  1  2  3  4  6  8 12 24
```

We might then choose the sub-maximum value of 12 subblocks. We would then apply the `psr()` function to create a sequence for a single participant.

```
options(width = 77)
psr(c("congruent", "incongruent"), 12, 24)
```

```
 [1] "congruent"   "congruent"   "incongruent" "incongruent" "incongruent"
 [6] "incongruent" "congruent"   "congruent"   "congruent"   "incongruent"
[11] "incongruent" "congruent"   "congruent"   "congruent"   "incongruent"
[16] "incongruent" "congruent"   "incongruent" "congruent"   "incongruent"
[21] "congruent"   "congruent"   "incongruent" "incongruent" "congruent"
[26] "incongruent" "incongruent" "congruent"   "incongruent" "congruent"
[31] "congruent"   "incongruent" "congruent"   "incongruent" "incongruent"
[36] "congruent"   "congruent"   "incongruent" "congruent"   "incongruent"
[41] "incongruent" "congruent"   "incongruent" "congruent"   "incongruent"
[46] "incongruent" "congruent"   "congruent"
```

The above vector is the sequence of levels for a single participant. The code below shows how to use the base R function `lapply()` to do this for 20 participants, returning the results in list format (not shown).

```
lapply(seq_len(20),
       \(.x) psr(c("congruent", "incongruent"), 12, 24))
```

The utility of this simple approach is limited, however, in that returns a sequence of level labels, but what the researcher typically needs is a sequence of *stimuli*. The `explan` package provides a second function to randomize a table of stimuli, `psr_stimuli()`. The researcher must first create a data frame containing the full set of stimuli delivered to each participant and then pass this table as the first argument of the function. The second argument is the name of the variable (or variables) that represent the independent variable (or variables) in the study. For illustration, the `explan` package has a built-in table corresponding to the Stroop design mentioned above, named `stroop_stimuli`.

```
head(stroop_stimuli, 6)
```

```
  stimulus_id   word font_color congruency
1           1   blue       blue  congruent
```

```
2           2  green      green  congruent
3           3    red        red  congruent
4           4 yellow     yellow  congruent
5           5   blue       blue  congruent
6           6  green      green  congruent
```

Because the first argument is a full table of stimuli, `psr_stimuli()` can infer the values of n_r and n_k from the data, and so the only further argument that the researcher must specify is the desired number of subblocks (n_s), which appears as the third argument to the function. The result of the function is a table with the rows sorted in the presentation order, as determined by PSR. By default, the function sequences the stimuli for a single participant. To get 16 participants, we set `n_part` to 16.

```
set.seed(1451) # optional: to reproduce the output below exactly

stroop12 <- psr_stimuli(stroop_stimuli, IVs = "congruency",
                        n_subblocks = 12,
                        n_part = 16)

head(stroop12, 9)
```

```
  PID sb_no stimulus_id   word font_color  congruency
1   1     1          48 yellow        red incongruent
2   1     1           8 yellow     yellow   congruent
3   1     1          34  green       blue incongruent
4   1     1          15    red        red   congruent
5   1     2          41   blue        red incongruent
6   1     2          43    red       blue incongruent
7   1     2          16 yellow     yellow   congruent
8   1     2           7    red        red   congruent
9   1     3          25   blue      green incongruent
```

In the output, the rows appear in their order of presentation, and `sb_no` is an integer specifying the subblock number that each stimulus appears in. For further information and examples, see the vignette "randomization" in the `explan` package.

3.7 Discussion

In this chapter, we discussed how a restricted randomization algorithm from clinical trials could be adapted to improve power in the context of a repeat-measure within-subject laboratory experiment. Permuted-Subblock Randomization (PSR) increases power by ensuring balance among experimental levels over time, and in this manner helps prevent the masking of effects of interest by time-dependent nuisance variation engendered by time-varying fluctuations such as learning, attentional fluctuations, or other physiological or environmental factors. PSR offers

a ready-made, free, and easy-to-implement solution to help remedy the pervasive problem of low power in psychology and neuroscience. It requires no specialized technical knowledge nor any a prior expectations or assumptions about the temporal structure of dependencies in the data. It will boost power whenever the sequences of within-subject observations are temporally autocorrelated, while maintaining strict control over the false positive rate.

How much power can one expect to gain using the algorithm? First we note that power gains were observed for all designs containing multiple subblocks and across all four scenarios containing time-varying errors, with a range between 4% and 45% and median of about 13%. To give a sense for what this would mean in terms of budgetary savings for the researcher, let us consider the case of a typical design ($n_p = 40, n_k = 4, n_r = 12$). In many situations it is likely to be impractical to use the maximum number of subblocks, as the pattern would be too predictable. Using the sub-maximum number of subblocks (PSR_6) for this design with the "mixed" error scenario and $\eta^2 = 0.01089$ yielded power gains of about 13.8% over baseline (75.8% versus 66.6%), which is fairly close to the median gain of 13%. Assuming we used PSR_6 in the study, how many fewer participants would we need to run to reach the same level of power as we would attain using PSR_1? To estimate this value, we ran a set of additional simulations (see Supplementary Materials). Using PSR_6 instead of simple restricted randomization reduced the number of participants from 40 to 33, corresponding to a savings on research expenditure of 16.9%.

Our PSR algorithms makes it possible to boost power by changing the study design without requiring any change to the analysis. We consider this a benefit of our approach. However, in discussions of randomization and counterbalancing in the literature, one often finds strong recommendations to include block as a covariate in the analysis. These recommendations often are found in situations where the blocking involves groups of participants rather than groups of trials within participants, and so may not apply to within-participant experiments. When time-varying patterns are largely idiosyncratic across participants, as in our simulations, subblock effects will largely cancel (e.g., the mean measurement in subblock i will be neither high nor low across participants, despite the observations in subblock i being correlated within each subblock for each participant). Thus, including subblock as a covariate for fully within-participant designs may serve only to complicate the analysis without necessarily yielding any tangible benefit. This may not be the case for other types of designs, such as so-called 'split-plot' designs where there is at least one between-participant factor in addition to any within-participant factors.

Although using PSR does not require any alterations to analysis, there is one situation that calls for caution: using PSR in tandem with permutation tests. Permutation tests involve the construction of a null-hypothesis distribution by permuting the condition labels and recalculating the test statistic many times over. Permutation tests assume that observations are exchangeable under the null hypothesis (Good, 2013). This implies that if PSR has been used to randomization conditions in the original data, then the re-randomization algorithm used to construct the null-hypothesis distribution should respect this constraint by permuting the

labels within the same subblocks. Failing to adhere to this violates exchangeability and may inflate the false positive rate.

One danger of using PSR is increasing the number of subblocks makes the sequence of conditions more predictable than simple restricted randomization. In practice, predictability will depend on aspects of the study design in combination with the memory capacity of the participants and the perceptual distinctiveness of the conditions. The optimal balance will depend on the relative importance of avoiding false negatives versus avoiding predictable sequences. We suggest the following guidelines. First, researchers should certainly avoid highly predictable sequences where each condition is associated with a distinct response. One should generally avoid using the maximum (n_r) number of subblocks because in many instances this is likely to result in a sequence that is too predictable, although there may be exceptions where n_k is large or the conditions are difficult to distinguish. An appealing strategy is to use the second or third highest value from the set of possible subblocks (e.g., PSR_6 or PSR_4 for $n_r = 12$). When avoiding predictable sequences is of paramount importance, advantages may still be obtained by choosing the second lowest number in the set of divisors (PSR_2 for $n_r = 12$). Also, PSR will be most useful with a reasonable number of repetitions ($n_r \geq 4$), since for $N_r = 2$ and $N_r = 3$ only two options are available: simple restricted randomization (PSR_1) and PSR with the maximal number of subblocks, the latter of which is likely to yield overly predictable sequences.

Might there be hidden dangers of using PSR that were not brought to light by our simulations? Under the context of one-factor design, we cannot answer with an unqualified "no". It is possible that we have overlooked situations where applying PSR or other such randomization algorithms will introduce statistical artifacts. One area that requires further investigation is situations where there is spectral coherence between subblock size and the time-varying pattern. For example, PSR with one repetition per subblock introduces a level of recurrence that has an n_k bandwidth. It may cause problems if the time-varying noise also has a strong n_k spectral component. Because the process is still fundamentally random, we are doubtful that such coherence would increase the false positive rate. Instead, it would tend to reduce power, much like the "blocking" effect discussed in Thul et al. (2021). This part of discussion would go further in Chapter 5.

So far, we have shown that it is possible to tap into time-varying fluctuations as a source of power through PSR. But a further question is, how to extend our randomization approach to other situations? For example, it is more realistic that researchers have a design with random effects of stimuli as well as participants. To the extent stimulus variation is of interest, the estimation of such variation will be contaminated by temporal fluctuations, since stimuli are presented over time. To capture random effects, a field-standard approach is to use linear mixed effect model (LMEM). But LMEM assumes "static" random variation components that are irrelevant to time for each participants, and therefore ignore the time-dependent noise. Therefore, combining PSR and LMEM may maximally clean up nuisance noise to allow more precise estimation of stimulus effects.

Also, we note a result that is important and seems to run contrary to commonly-held statistical wisdom. Some have held that that the presence of time-dependent error structure in multi-level data invalidates the use of classical statistical approaches (Amon & Holden, 2021) or calls for sophisticated modelling of such effects to avoid false positives (Baayen et al., 2017). Our results challenge this received wisdom, further corroborating the main conclusion from Thul et al. (2021) that they can safely be ignored, provided that presentation order has independently randomized across participants. Indeed, across the three time-dependent error scenarios we considered, the Type I error rates and power curves for the field-standard of simple restricted randomization were indistinguishable from the scenario with independent errors. Our view is that these fluctuating patterns represent an untapped additional source of power that, to date, could only be reclaimed using very advanced statistical modelling such as Generalized Additive Mixed Models (GAMMs) with factor smooths. But it also reflects a possibility that power advantages could be maximally achieved by simultaneously applying design-based and model-based approaches.

Therefore, to tap into time-varying fluctuations as power source and improvement power as much as we can, in next chapter, we extend usage scenarios of PSR, combining LMEM and GAMMs to investigate if this algorithm can achieve additional power gains than it already did.

Chapter 4

Comparing PSR with Model-Based Approaches

4.1 Background

In Chapter 3 we examined PSR in repeated-measures datasets and saw its capacity in boosting power for one-factor designs analyzed using ANOVA. Because of its test-independent characteristics, PSR could also contribute to power gains in other types of analyses. In this chapter we evaluate its performance when combined with mixed-effects models, where individual differences are directly modeled as random effects. One goal is to confirm that PSR also increases power combined with standard mixed-effects models that ignore temporal fluctuations, as it does for ANOVA. A second, more important, goal is to compare the performance of the design-based PSR to the performance of Generalized Additive Mixed Models (GAMMs), a model-based approach for capturing temporal trends in data. Finally, as these approaches are not mutually exclusive, we also seek to estimate the potential of controlling for temporal fluctuations by using GAMMs and PSR together.

As a classic "recipe", ANOVA is often performed on aggregated data (e.g., subject means). When data is aggregated, treatment-by-subject interactions cannot be estimated. Treatment-by-subject interactions reflect the fact that treatment effects vary across individuals. They can only be observed when researchers look at the raw data in a repeated-measure dataset where multiple observations are collected within conditions of the design, because no variation can be calculated from a single data point. A more modern approach is to use linear mixed-effects models (LMEMs) on the raw (unaggregated) data and to include the treatment-by-subject interaction in the model as by-subject random slopes. Under the LMEM framework, researchers can more easily estimate within-subject variance components through the random effects structure.

However, similar to ANOVA, LMEMs as typically used do not allow random effects to vary over time. With essentially static random effects, autocorrelation patterns would still remain in the residual structure. Although ignoring autocorrelation patterns in residuals will not increase

the rate of false positives when the sequence of conditions have been appropriately randomized or counterbalanced (Thul et al., 2021), as we showed in the last chapter, there is power to be gained by controlling for them in the design.

But researchers have long been aware of autocorrelation patterns in repeated-measured data, and more commonly have proposed accounting for them statistically using model-based approaches. The magnitude of the impact of time-varying fluctuations would also be a function of multiple aspects in an experiment, including the strength of the fluctuations, their structure in time, and their consistency across experimental subjects. To capture such relationships, one general purpose strategy is to explicitly model them using advanced modelling approaches such as Generalized Additive Mixed Models (GAMMs) that can flexibly estimate wiggly patterns (Baayen et al., 2017; Wood, 2017) both as fixed and random effects. The fundamental way GAMMs operate is by estimating patterns as a linear combination of a set of *basic functions*. The exact mathematical details of how they do this are not important here, and have been discussed at length elsewhere (Baayen et al., 2017). For our purposes, it is sufficient to operate with the conceptual understanding that GAMMs allow for a time-varying fixed intercept, which captures common temporal trends across all participants, as well as a time-varying random intercept (here called a "factor smooth"), which allows the pattern to vary from the common standard across participants. It would also be possible for the effects of condition to vary over time, both generally as well as across participants (time-varying slopes and time-varying random slopes); but for simplicity, we follow Thul et al. (2021) in modeling them as static random effects.

GAMMs with factor smooth may be especially useful to model time-varying fluctuations in experimental datasets, as "human factors" such as fatigue and mind wandering are common and likely to introduce temporal variation that itself will vary across subjects. It is possible that applying GAMMs can achieve better power improvement beyond PSR, because of their powerful ability to model wiggly patterns. But there are a number of trade-offs to consider, including hidden dangers that may be obscured by their technical sophistication and lack of standards for model specification (Thul et al., 2021). Another consideration is that GAMMs assume that the nature of underlying change is continuous, whereas research subjects can show discontinuous shifts if an experiment contains rest breaks.

Compared to GAMMs, PSR is far easier to implement and makes fewer assumptions about the patterns of time-varying fluctuations, and could therefore make a model-based approach redundant. This raises the question as to whether PSR offers power gains that are comparable to using GAMMs alone, and whether there are circumstances where it may even outperform them. But these approaches are not mutually exclusive-perhaps there is value to using PSR in the design and GAMMs in the analysis. In what follows, we describe a set of Monte Carlo simulations to evaluate these questions.

4.2 Methods

We conducted a Monte Carlo simulation study in order to access type I error rates and power performance of PSR used in combination with linear mixed-effects models (LMEMs), generalized additive mixed models (GAMMs), and analysis of variance (ANOVA), with the latter of these analytical techniques providing a baseline. The design parameters, data generation process, and time-varying error patterns were identical as in Chapter 3. To speed up the simulations, an R package `psrsim` was developed that incorporated functions written in C++ for data generation and for the ANOVA analysis.

Because our goal was to focus on the maximum power boost attainable under PSR, we used PSR with the maximal number of subblocks. We also include simple restricted randomization (i.e., PSR with one subblock) as a baseline.

4.2.1 Analysis

Because we were using new code for data-generation written in C++, we re-ran the ANOVA analysis from the previous chapter just in case there were any minor differences arising from the C++ libraries we used for generating random numbers. We fit both LMEM and GAMMs using the `bam()` function from the `mgcv` package.

```
lmem_model <- mgcv::bam(formula = dv ~
                          ## intercept
                          1 +
                          ## main effect
                          cond +
                          ## by-subject random intercept
                          s(id, bs = "re") +
                          ## by-subject random slope
                          s(id, cond, bs = "re"),
                        data = dat)

gamms_model <- mgcv::bam(formula = dv ~
                           ## main effects
                           cond +
                           ## time-varying fixed intercept
                           s(order, bs = "tp") +
                           ## time-varying random intercept
                           s(id, order, m = 1, bs = "fs") +
                           ## by-subject random slope
                           s(id, cond, bs = "re"),
                         data = dat)
```

It is possible to fit a pure LMEM with `mgcv::bam()` by specifying only time-independent fixed

and random effects in the model syntax. Random effects are specified differently in `bam()` than in the `lme4::lmer()` function which is more commonly used for mixed-effects modeling. In `bam()`, time-independent random effects are specified using the general function `s()`, which is also used for smooth (i.e., time-varying) terms. In the function call above that creates the `lmem_model` object, the random effect term `s(id, bs = "re")` specifies by-subject random intercepts and `s(id, cond, bs = "re")` specifies by-subject random slopes. The argument `bs = "re"` is a way of requesting a static random effect structure. This model formula is notationally equivalent to the `lme4::lmer()` formula `dv ~ cond + (1 + cond || id)` where (uncorrelated) by-subject random intercepts and random slopes are specified. Turning to the model formula used to create the `gamms_model` object, the first `s()` term, `s(order, bs = "tp")`, specifies a fixed intercept than can wiggle over time (i.e., a time-varying fixed intercept term). The argument `bs = "tp"` tells `bam()` to estimate this term by default "thin plate" basic functions. The second `s()` term, `s(id, order, m = 1, bs = "fs")` specifies factor smooths by the argument `bs = "fs"` that defines a by-subject random intercept term than can wiggle over time. The argument `m = 1` adds a penalty to the first basic function and ensures the factor smooths behave as a random effect (Baayen et al., 2017). The last `s()` term `s(id, cond, bs = "re")` specifies by-subject random slopes of the dv `cond` by setting `bs = "re"`.

As in last chapter, fully crossing all design parameters yielded 5,670 settings. To compromise between comprehensive results and computational burden, we only run simulations for our three chosen "focal designs": (a) the smallest design with $n_p = 25$, $n_k = 2$, and $n_r = 6$; (b) the middle design with $n_p = 40$, $n_k = 4$, and $n_r = 12$; and (c) the middle design with $n_p = 60$, $n_k = 8$, and $n_r = 24$. Across all three designs, only PSR_1 and PSR_{max} (i.e., PSR_6 in the smallest design, PSR_{12} in the middle design, and PSR_{24} in the largest design) are considered. This reduces the number parameter settings from 5,670 to 210. For each of these 210 designs, we conducted 10,000 Monte Carlo runs to estimate Type I error rates and power, within each run including randomly generating a dataset, analyzing it with one of our three analytical models (ANOVA, LMEM, GAMMs), and extracting the p-value. For GAMMs and LMEMs we derived p-values using Wald z. Specially for LMEM and GAMMs, we also recorded whether the model is converged on each run, and excluded non-converging models in those calculations. Combining all focal design settings and analytical methods yielded 630 scenarios under which we calculated Type I error and power from 10,000 Monte Carlo runs.

4.3 Results

4.3.1 Power

To estimate power, for 540 out of the 630 cases where $\eta > 0$, we calculated the proportion of runs (out of 10,000) that yielded a significant main effect. We also estimated the power advantage of PSR_{max} by calculating the percent increase in power for PSR_{max} relative to PSR_1 (simple restricted randomization) at each of the six non-zero values of η for that case. We then extracted the median power advantage observed across all six values of eta for PSR_{max}.

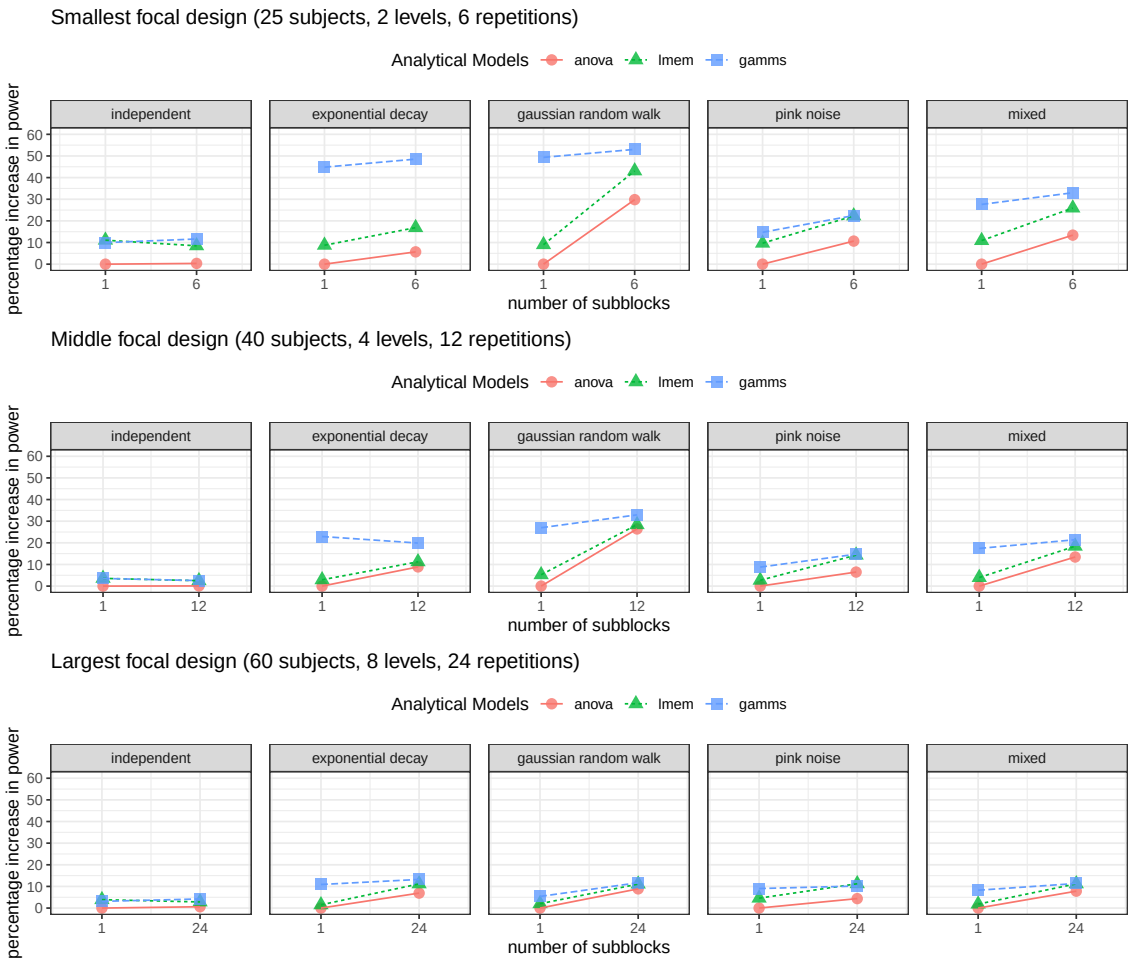

Figure 4.1: Median percent increase in power as compared to baseline (ANOVA with simple restricted randomization; i.e., 1 subblock) across all six non-zero levels of η by focal designs, error structure, and number of subblocks (1, nr).

Figure 4.1 shows the median percent increase in power with respect to baseline broken down by focal designs and collapsed over the six values of η. For the uncollapsed data, see Figure 4.2. LMEMs and GAMMs seem to have consistently higher power than ANOVA. For LMEM, such increase ranges from 8.5% to 11.1% in smallest designs, 2.5% to 3.6% in middle designs, and 2.8% to 3.9% in largest designs; for GAMMs, the increase ranges from 9.9% to 11.6% in smallest designs, 2.6% to 3.5% in middle designs, and 3.1% to 4.2% in largest designs, respectively. But the apparent superiority of LMEMs to ANOVA and part of that for GAMMs is illusory; calculations of p-values for LMEM and GAMMs are based on the t-as-z methods (Wald z statistics), and this method tends to be anti-conservative (Luke, 2017 , also see the type I error rates describe later). As the sample size increases, this anti-conservativity wanes (Barr et al., 2013).

With time-dependent error structures present and only simple restricted randomization (i.e., PSR_1), GAMMs showed consistent power advantages over ANOVA and LMEMs across all designs, ranging from 10.2% to 53.0%, with the highest increase in power observed in the smallest design under the gaussian random walk scenario. LMEM showed moderate increase in power over ANOVA in PSR_1, ranging from 11.1% to 43.1%. The highest 43.1% increase of power is observed in smallest design under the mixed error scenario. But again, it is important to note that the gain of LMEMs over ANOVA is artificial as it reflects the anti-conservativity of the former. The power gain of GAMMs over LMEMs, however, is real, and shows that a model-based approach using GAMMs can boost power by accounting for autocorrelated error structures (Barr et al., 2013; Thul et al., 2021).

Compared to our baseline of simple restricted randomization (PSR_1) with ANOVA, applying PSR_{max} (i.e., $n_s = 6$ in smallest designs, $n_s = 12$ in middle designs, and $n_s = 24$ in largest designs) increased power across all analytical methods. First, GAMMs with PSR_{max} achieved the highest power gains in all cases over other analytical approaches, ranging from 10.2% to 53.0%. Next, the power increase for the LMEM with PSR_{max} ombination ranged from 11.1% to 43.1%. And lastly, as expected, ANOVA with PSR_{max} yielded gains ranging from 4.4% to 29.8%. The greatest power improvements for all three analytical methods are observed in smallest designs under the gaussian random walk scenario, with the increase of 53.0% for GAMMs, 43.1% for LMEM, and 29.8% for ANOVA, respectively.

Where error structures were time-dependent, in most cases replacing simple restricted randomization with PSR boosted power across all analytical methods. For example, when using LMEM to analyze datasets, implementing PSR brings another 10.5% power increase on average. Furthermore, even though GAMMs already estimate time-varying components, applying PSR could increase power by 3.2% on average. However, we do observed an exception for GAMMs in the middle design under the exponential decay scenario. In this scenario, GAMMs have slightly better power under simple restricted randomization than under PSR, (22.9% versus 19.9% respectively).

One key question we set out to answer was the relative benefits of the design-based and model-

based approaches for improving power. A useful way to answer this question is to compare the power gains of PSR plus LMEM, which controls temporal variation through the design, to those of GAMMs alone with simple restricted randomization, where temporal variation is controlled statistically. Henceforth we refer PSR_{max}-LMEM as the design-based approach and PSR_1-GAMMs as the model-based approach. Surprisingly, the design-based approach achieved comparable or even better power gains as compared to GAMMs. For example, power gains were similar for both approaches in the smallest design under the mixed scenario (25.9% for the design-based versus 27.6% for the model-based approach). Furthermore, in the smallest design under the pink noise scenario, the design-based approach improved power more than the model-based approach (22.3% versus 14.7% respectively); in middle designs, the design-based exceeded the model-based approach under gaussian random walk (28.4% versus 27.0%), pink noise (14.3% versus 8.8%), and mixed (18.4% versus 17.4%); for the largest designs, the design-based approach outperformed the model-based approach in all four time-dependent error structures (11.2% versus 10.9% in exponential decay, 11.1% versus 5.4% in gaussian random walk, 11.2% versus 9.0% in pink noise, and 11.2% versus 8.2% in mixed).

Given the strong performance of the design-based approach, the question emerges as to whether there is any extra benefit to be attained by combining it with the model-based approach. To answer this question, we compared the power gains of PSR plus GAMMs, which stands for PSR_{max}-GAMMs and represented an omnibus approach, to the design approach (i.e., PSR_{max}-LMEM). According to our 12 error-design scenarios, the omnibus approach mostly brings additional power gains over the design-based approach, with the range from -1.1% to 31.7%, and a median of 2.5%. Such additional power benefit is highest in smallest design under the exponential decay error scenario: when implementing both PSR and GAMMs, an 31.7% power increase is observed over simple design-based approach. In addition, in smallest design under gaussian random walk error scenario, although implementing the design-based approach already increased power, the omnibus approach can reach another 9.9% increase on this basis (i.e., 43.1% for design-based approach and 53.0% for omnibus approach, respectively). As the sample size increases, the design-based approach had similar or even better power advantages than omnibus approach did. Indeed, we observed that in largest design under pink noise error scenario, implementing omnibus approach did not do better over the design-based approach, with the power gains of the omnibus approach being -1.1% percent lower than the design-based approach (i.e., 11.2% for design-based approach and 10.2% for omnibus approach, respectively).

4.3.2 Type I error

In the last chapter we validated that PSR does not yield anti-conservative type I error rates with ANOVA. Here we extend that analysis to LMEMs and GAMMs. To estimate Type I error under these circumstances, we calculated the proportion of simulation runs with a statistically significant main effect (with $\alpha = .05$) for each of the situations where $\eta^2 = 0$. Again, to determine whether Type I error rates were close to the nominal $\alpha = .05$ level, we checked whether observed rates fell within the 99.9% Agresti-Coull confidence interval for a binary

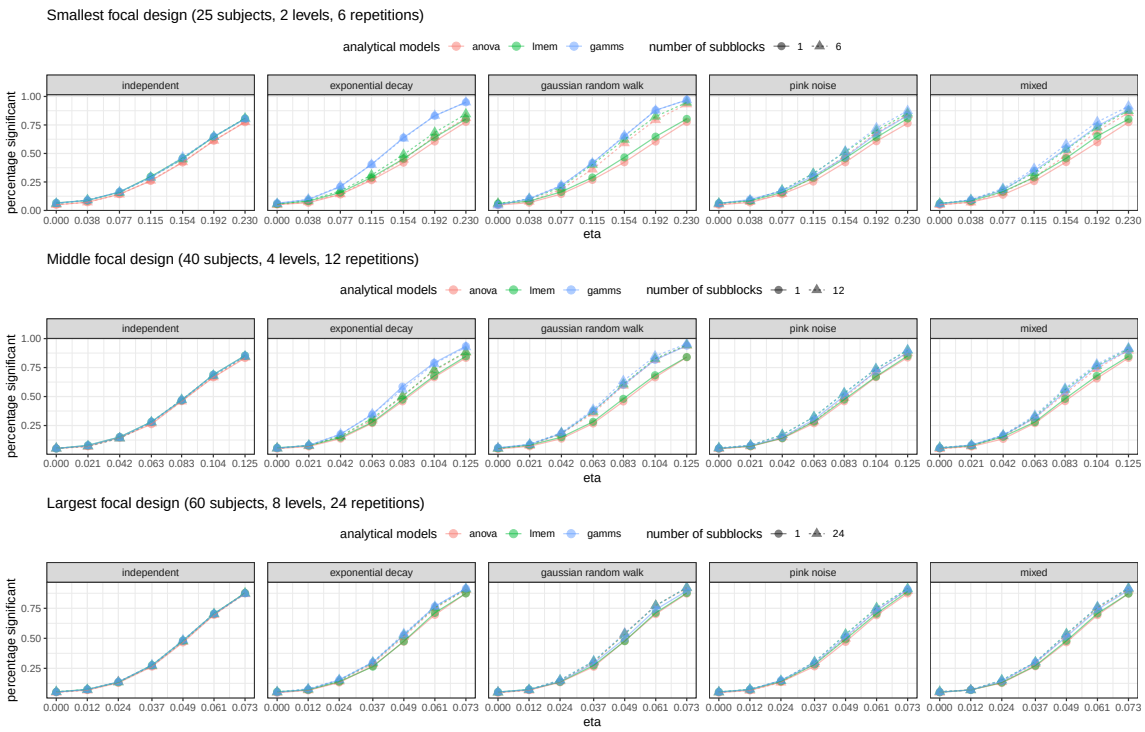

Figure 4.2: Power curves ($\alpha = .05$) for the three focal designs plotted by number of subblocks, trial-level error variance scenario, and effect size (η). Each data point is estimated from 10,000 Monte Carlo simulations.

process with probability .05 and 10,000 samples, which was [0.0447, 0.0559]. As last chapter, cases that fall below this interval are seen as conservative (i.e., have inflated false negatives); cases that fall above this interval are seen as anti-conservative (i.e., have inflated false positives); and finally, cases that fall within this interval are seen as nominal (i.e., are calibrated to the α level).

Figure 4.3 presents type I error rates for all designs. As expected, all type I error rates of ANOVA fall into the 99.9% confidence interval and are deemed nominal, ranging from 0.0467 to 0.0543. Also, anti-conservative type I error rates are observed for LMEM and GAMMs methods, especially in small samples, because of the t-as-z method used to compute p-values. Type I error rates for LMEM ranged from 0.0502 to 0.0666 and 0.0457 to 0.0653 for GAMMs. Anti-conservative type I error rates are most prominent in the smallest design with $n_s = 25$, $n_k = 2$, and $n_r = 6$; yet this situation is mitigated as the n_p increases: for all four non-independent error structures in middle focal design, anti-conservative type I error rates are acceptably closing to the upper limits of the 99.9% confidence interval, while in the largest design, only one anti-conservative type I error rate of 0.0573 was observed in mixed, for LMEM.

However, a more important question is whether using PSR with LMEMs and GAMMs adds additional anti-conservativity over simple restricted randomization. According to our results, with 5 exceptions, applying PSR did not lead to more anti-conservative type I error rates with LMEM analyses. Regarding those 12 exceptions, the increases of type I error rates are negligible, with all increase less than 0.01. For GAMMs, only one exception is observed in the smallest design under the mixed scenario, where using PSR increased the type I error

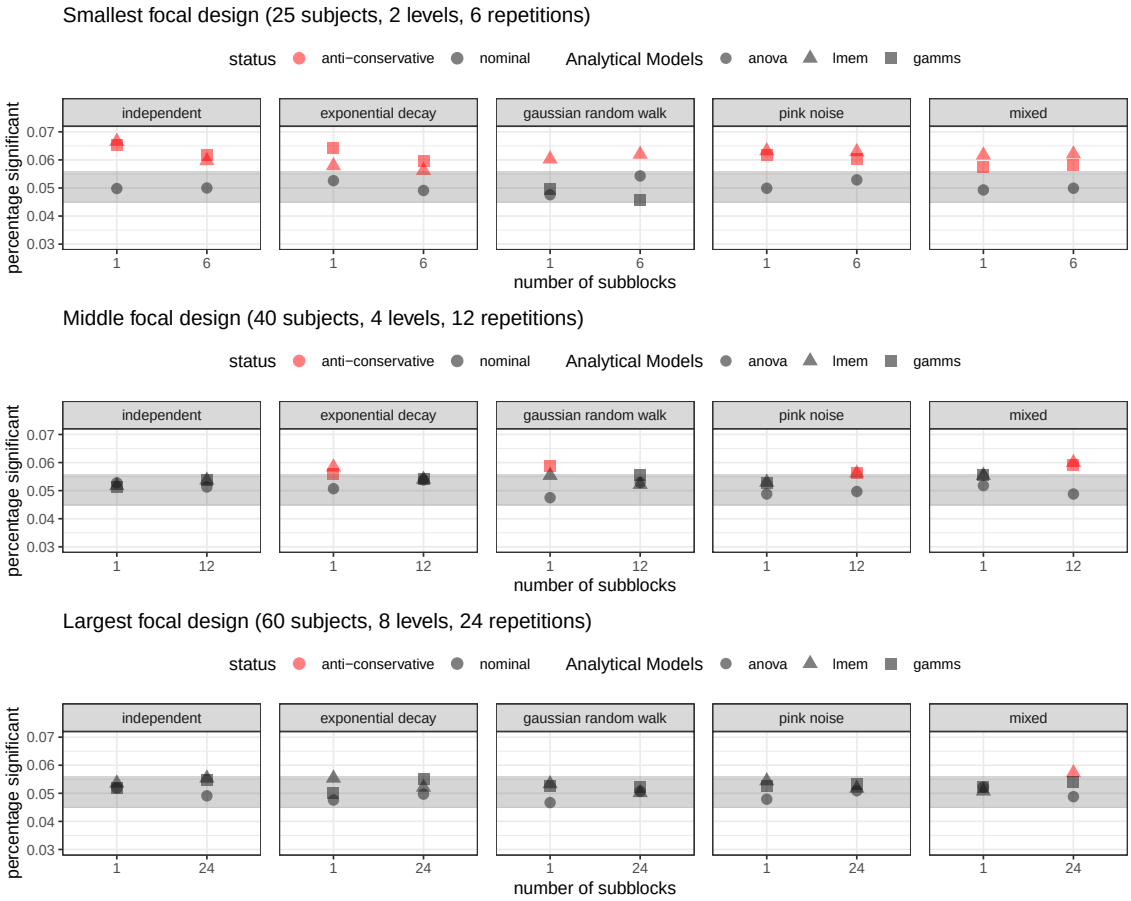

Figure 4.3: Type I error rates ($\alpha = .05$) for the three analytical approaches: ANOVA, LMEM, and GAMMs, broken down by number of subblocks, error variance scenario, and effect size (η). Each data point is estimated from 10,000 Monte Carlo simulations. The shaded region represents the 99.9% Agresti-Coull confidence interval for 10,000 random samples from a process with a "success" rate of .05.

rate from 0.0575 to 0.0581, a difference of less than 0.001. Given how small these differences are, it is unclear whether they reflect standard error from the simulations or true underlying differences. Finally, with 3 exceptions out of our 90 cases, GAMMs did not have any greater anti-conservativity than LMEM did.

4.4 Discussion

In this chapter, we extended PSR to new analytical scenarios and demonstrated its compatibility with model-based strategies to improve power. Our results suggested that a design-based approach to controlling time-dependent noise using PSR yields gains comparable to model-based approaches using GAMMs under many scenarios, and that combining both approaches can improve power even slightly more. We also established that even when advanced GAMMs are considered to model time-dependent noise, PSR is still safe to use (in terms of maintaining reasonable false positive rates).

According to our simulation results, using PSR was advantageous to power gains in most of the designs, regardless of the analytical method of choice. The power gain of PSR with traditional analytical method that ignore time-varying components (i.e., LMEM, ANOVA) is encouraging, especially in light of claims from some researchers that such models are flawed or even promote false positives when time-varying components are present (Baayen et al., 2017). Indeed, in many scenarios and designs, PSR yielded gains that were comparable or even higher than for GAMMs. And PSR has the advantage of simplicity and ease of implementation as compared to GAMMs, making power gains available to the many researchers who lack the technical knowledge to implement GAMMs or other advanced modeling strategies. That said, an important caveat is that we compared GAMMs to PSR with the maximum number of subblocks, and as noted in the previous chapter, this may result in sequences that participants find predictable. So GAMMs have the advantage that their power gains do not come at the expense of creating predictable sequences of trials.

Finally, combining PSR with LMEMs and GAMMs did not introduce any clear additional risks to anti-conservativity beyond those already associated with these approaches. If balancing type I error rates and the power is the main priority when analyzing relatively small datasets, simply using PSR with ANOVA would be the desired solution for improving measurement quality, when applicable. But once the sample size gets sufficiently large, choosing which analytical model to use and whether to combine it with PSR would depend on the expected error structure. In practice, error structures seem more likely to follow the mixed pattern as this reflect realistic processes happening at multiple time scales. In this case, applying PSR while keeping using the traditional analytical methods (LMEM or ANOVA) could have similar power performance to GAMMs. But in special cases where only a strong practice/learning effect is expected and the sample size is small, using PSR with GAMMs might optimize power.

In this chapter, the utility of PSR has validated in the framework of modelling approaches. But up to now we have only been looking at a design with a single factor. It is also useful

to determine the safety and potential benefit of using PSR in multifactor designs, such as 2x2 designs, which are extremely common in psychology and neuroscience. We are optimistic that similar benefits would be detected in a factorial context, based on the fact that the hypothesis test for a main effect in a one-factor design with four levels is equivalent to an "omnibus" test for a 2x2 factorial design. However, it is usually the case in factorial designs that certain effects (e.g., an interaction) might be more theoretically important than others (e.g., a main effect). In the next chapter, we consider these issues in more detail.

Chapter 5

Generalizing PSR to 2x2 Factorial Designs

5.1 Background

Chapters 3 and 4 demonstrated impressive power gains using the design-based approach of PSR for one-factor designs. But researchers usually use designs that are more complex. Indeed, the factorial design containing two factors is the most widely used experiment type in behavioural sciences (Kirk, 2013). A "full" factorial design should meet three requirements: (1) it should have at least two factors; (2) each factor should have at least two levels; and (3) all levels of each factor should fully combine with the levels of all other factors to form treatment conditions (or "cells", as they are often referred to). Under these requirements, the simplest full factorial design is a 2x2 (two-by-two) factorial design where there are two factors each of which has only two levels.

Let us refer to the two levels of the first factor as A and a, and the levels of the second factor as B and b. If we list A and a in rows, and list B and b in columns, we would have a 2x2 condition table, and the first factor becomes a "row factor" and the second one becomes a "column factor". The cells (or treatment conditions) in this table represent all possible level combinations of the row factor and the column factor (i.e., AB, Ab, aB, and ab, as presented in Figure 5.1, top right tables of each top row panels). In clinical and neuroscience experiments, is it very common to consider a 2x2 factorial design with one between-subject factor and one within-subject factor, with subjects randomly assigned to one level of the between-subject factor while receiving all treatment levels of the within-subject factor. But since it is more common for psycholinguistics and cognitive experiments to use within-subject designs, we will focus on the situation where both factors are within-subject, meaning that each subject would experience all four combinations of the levels of the two factors.

As in one-factor designs, in 2x2 factorial designs time-varying fluctuations would mask the true effects if stimuli sequences are not well organized. Furthermore, because the designs are more complex, the way this masking could happen would also be more complex. As for a one-way design, in a two-way ANOVA containing main effects and interactions, the sensitivity for each effect is still determined by the F ratio of mean squares, $F = MS_{treat}/MS_{error}$, and both

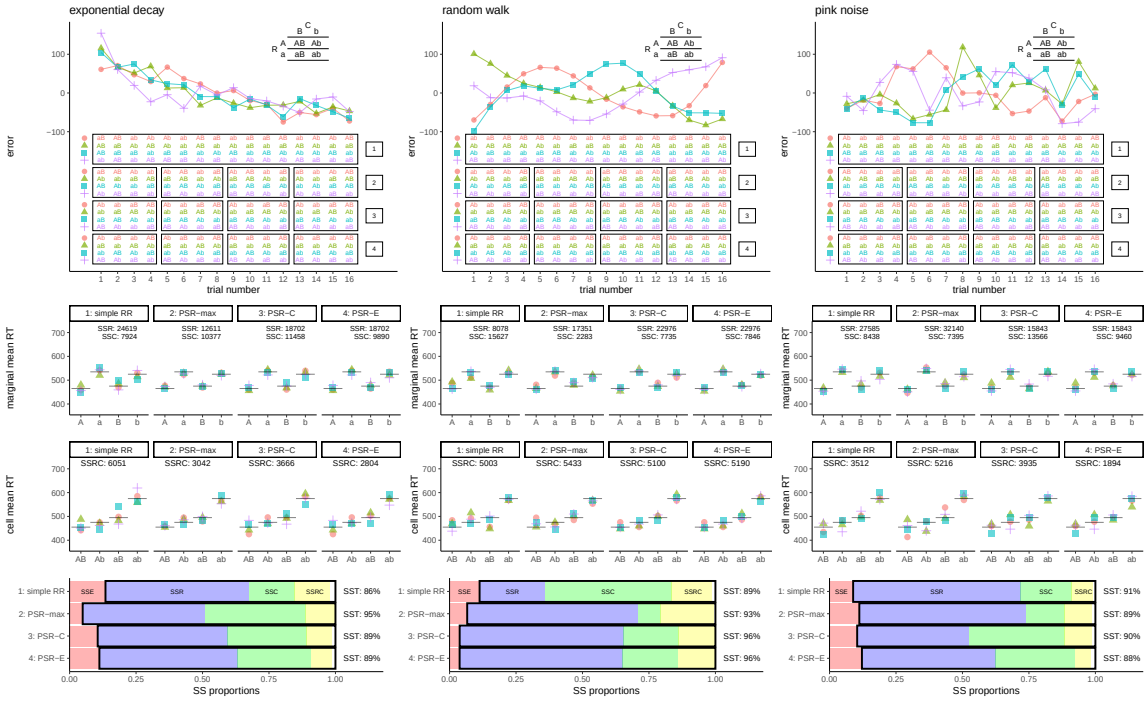

Figure 5.1: Three characteristic patterns of time-dependent errors: exponential decay, random walk, and pink noise. The top three charts show hypothetical patterns of response time errors (vertical axis) for four different participants, distinguished by shape (circle, square, triangle, plus) and color. Just below the curves are possible condition labels (AB, Ab, aB, ab) for each subject and trial (horizontal axis), organized into subblocks and "walk algorithms" (indicated by boxes and numbers). The condition tables are presented on top right of the error dependent charts. Four possible designs are represented: simple restricted randomization ('simple RR': rows in the subblock marked by number 1), PSR_{max} (rows in the four side-by-side boxes marked by number 2), PSR-C (rows in the four side-by-side boxes marked by number 3), and PSR-E (rows in the four side-by-side boxes marked by number 4). The charts on the second row show the margin mean RTs for each participant against the true marginal means. The charts on the third rows shows observed mean RTs (cell means) for each participant in each design plotted against the true condition means. All true means were shown by background lines in each chart. The charts at the bottom row shows sums of squares (SS) allocations for each variance component from the in relation to the total sum of squares, in three error structures. SSR = sum of squares of the row factor, SSC = sum of squares of the column factor, SSRC = sum of squares of interaction. SSE = sum of squares of error.

mean squares are scaled from the sum of squares, SS, by the corresponding degrees of freedom. However, in a 2x2 factorial design, SS_{treat} can be further broken down as the sum of SS_R, SS_C, and SS_{RC}, where SS_R represents the sum of squares of the row factor, SS_C represents those of the column factor, and SS_{RC} those of their interaction. As such, segregating each source of treatment variation from noise contributes to a larger SS_{treat} and would achieve a "general" overall power improvement. To take a closer look at SS_{treat}, the *main effect* for a given factor means the variance accounted for by this factor while *ignoring* the other factors in the design. Estimating the sum of squares for main effects involves computation of a mean for each level of the factor while *ignoring* the other factor (also known as "marginal means"). At the same time, a common computational formula the interaction defines this effect as what remains of SS_{treat} after the main effects are removed. Its sum of squares are calculated by summing the remaining variance in each level combinations (known as "cell means") by subtracting the main effects of the row factor and the column factor. Viewed in this way, it may be possible to discover different ways of performing restricted randomization that specifically target one of the three possible effects of interest (row, column, or interaction).

To give an intuitive example, let us extend our one-factor design example to the context of 2x2 within-subject design. Suppose the true mean RTs (cell means) for four level combinations, AB, Ab, aB, and ab are 455, 475, 495, and 575 ms, respectively. Under these true cell means, the true margin means would be (455 + 475) / 2 = 465 ms for A, (495 + 575) / 2 = 535 ms for a, (455 + 495) / 2 = 475 ms for B, and (475 + 575) / 2 = 525 ms for b, respectively. Also, since the effect of the row factor differs across the levels of the column factor (i.e., 475 - 455 = 20ms at B versus 575 - 495 = 80ms at b), there is an interaction in between row and column factors this design. To maintain the simplicity of our discussion, we further assume there are no individual differences, and all biases in individual RTs are simply due to trial-level, time-dependent noise (patterns presented in Figure 5.1), as they did in Chapter 3.

If we recognize that improving power is essentially improving the segregation of the signal variations from noise and thus increasing the proportion of SS_{treat} in the F ratio, we could try to use the PSR$_{max}$ algorithm to achieve a more uniform distribution of the repeated presentations of each conditions, as we did for the one-factor designs. In this case, we could see all four conditions as four independent levels from an "omnibus" factor and concatenate sequences by randomizing stimuli in four subblocks. Sequences resulting from this approach are presented in Figure 5.1, marked by number 2. Under PSR$_{max}$ in the exponential decay scenario, we increase the SS_{treat} proportion against simple restricted randomization from 86% to 95%, accounting for an additional 9% of the total variance (the left chart of the bottom row in Figure 5.1) . But as mentioned in Chapter 3, the PSR algorithm is not a method that deterministically improves power in every dataset, but one that does so over the long-run. On average, under PSR$_{max}$, the proportion of SS_{treat} should increase for all three time-dependent error structures, although for the chosen data set in the figure it actually decreased in the pink noise scenario (from 91% in simple restrict randomization to 89% in pink noise).

If long-run "omnibus" power improvement is indeed achieved under PSR$_{max}$, as would be

expected, it would also be useful to determine whether it does so by equally improving power for all three effects (two main effects and interaction). In the simulated data behind the figure, consistent power improvement was not found over all three error patterns: in the exponential decay scenario, PSR_{max} increased SS_C but reduced SS_R and SS_{RC}; while in the gaussian random walk and pink noise scenarios, SS_R and SS_{RC} increased under PSR_{max} but SS_C took a big hit.

5.2 "Walking" through a design table: PSR-C and PSR-E

To understand how power might be distributed among the three effects comprising SS_{treat}, let us have a closer look at the application of PSR_{max} to a single subblock containing conditions AB, Ab, aB and ab. From the perspective of omnibus power, we have four separate conditions, and so there is no particular arrangement of conditions that could possibly improve omnibus power for that subblock. But things change if we look at this from the perspective of each of the two main effects as well as from the perspective of the two simple effects that together comprise the interaction effect.

First, from the perspective of either main effect, we have two repetitions of each level within a subblock: two Bs and two bs. There are only six unique arrangements of these levels within the subblock: $BbBb$, $bBbB$, $BbbB$, $bBBb$, $BBbb$, and $bbBB$. The first two are cases where the levels alternate from trial to trial, and, following from the results in Chapter 3, this alternation should lead to higher power than cases where there are runs of the same level. But each instance of a given level B or b in the subblock is associated with a different level of the other factor, A or a. So if we want to maintain the alternating pattern $BbBb$ the only possibilities for doing this while combining with the levels of the other factor are $ABabaBAb$ or $aBAbABab$. Note that in either cases, the A and a levels get bunched together into $AaaA$ and $aAAa$; if we were to string together multiple subblocks organized in this way, we would have $AaaAAaaAAaaA\ldots$. For illustration, see Figure 5.1, top graph, row of subblocks marked 4. It is not possible to simultaneously alternate both factors—alternating one of them creates runs in the other. Given that the omnibus power is fixed, this means sacrificing some of the power for the row factor (A and a) for the sake of the column factor (B and b). The pattern where we alternate between B and b can be conceived of as a "figure-8" movement throughout the cells of the design, as shown in the right panel of Figure 5.2. Note that a second "figure-8" movement direction is also possible that would alternate A with a while creating runs of the levels of B and b. Note further that we can start this figure-8 movement at any arbitrary cell in the table. Henceforth we refer to arranging a subblock according to this figure-8 pattern as PSR-E.

What does PSR-E imply for the interaction effect? Let us consider that the interaction effect can be estimated as the difference of two simple effects: for example, the simple effect of B vs b at the level A minus the simple effect of B vs b at the level a, or: $(AB - Ab) - (aB - ab)$ which can be simplified to $AB - Ab - aB + ab$. In this equation, there are two positively-signed terms $+AB$ and $+ab$ and two negatively-signed terms $-Ab$ and $-aB$. The ideal subblock organization for improving interaction power would therefore be one in which the signs alternate within

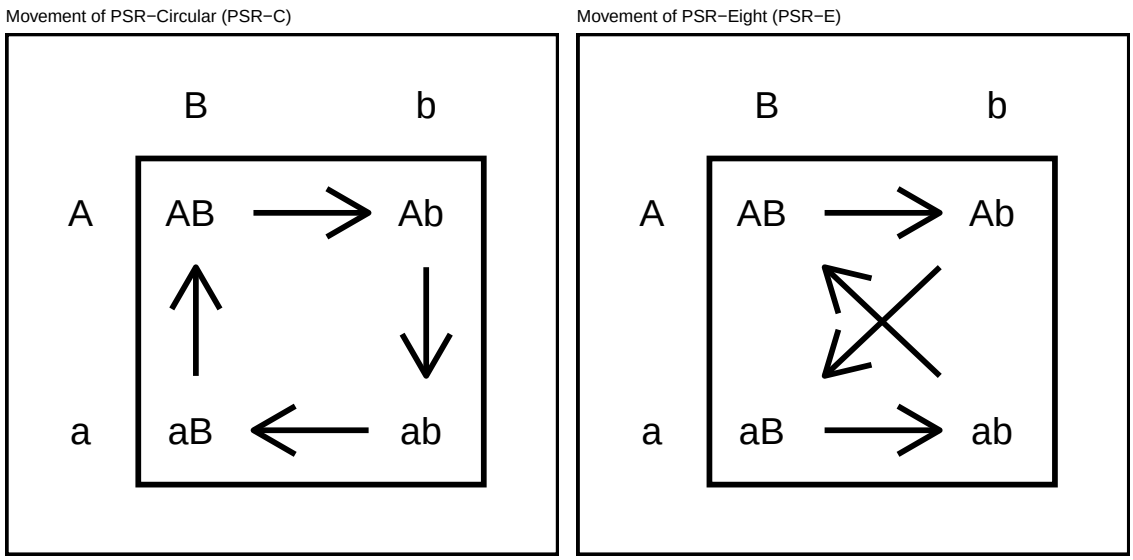

Figure 5.2: Two types of "movements" through the design table in 2x2 factorial designs, both (arbitrarily) starting from AB.

subblock, e.g., $AB - Ab + ab - aB$. But the figure-8 pattern of PSR-E creates runs of same-signed terms, e.g., $-Ab - aB + ab + AB$ (see Figure 5.1, top graph, row of subblocks marked 4), a suboptimal arrangement for interaction power.

What type of movement, then, might create the type of signed alternation that would improve interaction power? There is only one other type of within-subblock movement that could create this pattern: circular movement (Figure 5.1, left panel), henceforth PSR-C. Moving around the design table in a circle will alternate the signs in the interaction computation and thus improve power for the interaction. Returning to the above example with $+AB - Ab + ab - aB$, going clockwise produces $+AB - Ab + ab - aB$ and moving counterclockwise produces $+AB - aB + ab - Ab$. But note that clockwise movement produces runs in the levels of each main effect factor, so we hypothesize that circular movement improves interaction power at the expense of the main effects. Again, note that we can start this movement at any arbitrary random cell in the design table.

Both PSR-C and PSR-E fixed the condition sequences within a subblock while having just one repetition of a condition (cell) within the subblock. In that sense, they represent an implementation of PSR_{max}. However, unlike PSR_{max} where each subblock is randomized independently, here repetitions start at an arbitrary starting point and repeat identically over subblocks, creating a highly predictable pattern. While this repetition across subblocks is not strictly necessary, our interest here is in determining the theoretical limits of how much power can be improved without worrying about human factors. For PSR-C and PSR-E, the only thing that is truly random for each participant is the starting point of the sequence. But it is better for power to counterbalance the starting point across participants so that each cell has an equal chance of appearing first. For example, when using PSR-C, if participant one starts at AB on trial one, then participant two would start at Ab, participant three at ab, and participant four at aB. Thus if there is any overarching common pattern across participants (such as in the exponential

decay scenario), any bias will tend to cancel out.

Although developed specifically for factorial designs, we are not saying PSR-C and PSR-E should take priority over PSR_{max}. On the contrary, PSR-C and PSR-E are just two adaptations of the standard PSR algorithm that target power improvements for specific effects of interest. While we aim to investigate the maximum theoretical gains possible, we recognize that there are a wealth of possibilities for how these might be deployed in a more practical manner. For example, researchers can switch the starting points for each subblocks, alternate algorithms for each subblock, or move in different directions across subblocks to generate independent sequences. But doing so would only produce sequences that are closer to PSR_{max}. In fact, given the requirement that "each subblock only contains one repetition of each condition", each subblock is either PSR-C or PSR-E because no other pattern of "movement" through the design table is possible. Therefore, we can look at PSR_{max} as just randomly choosing starting point, movement type (PSR-C or PSR-E), and movement direction for each subblock.

To extend the PSR algorithms (PSR_{max}, PSR-C, and PSR-E) into the context of 2x2 factorial design, and to assess their long-run Type I error rates and power performance, we conducted Monte Carlo simulations in which we generated data from hypothetical experiments with time-dependent errors.

5.3 Methods

To improve simulation efficiency, all simulations were run via C++ programming language. But we also provided a wrapper R package `psrsim` for non-experts of C++.

5.3.1 Determining representative study design parameters

To mimic realistic experiments, we used different parameters to generate our datasets. First, we chose three representative sample sizes by inspecting studies involving 2x2 factorial designs from the data described in Chapter 2. As design parameters, we still use n_p to denote the number of participants, n_r to denote condition repetitions, and n_s to denote number of subblocks. Here we skipped the number of conditions, n_k, as in 2x2 factorial designs, $n_k = 4$. Based on the data from Chapter 2, we arrived at the following design parameters: $n_p = \{28, 40, 50\}$, and $n_r = \{10, 16\}$. As PSR-C and PSR-E were developed from PSR_{max}, we hereby only considered two possible numbers of subblocks, $n_s = \{1, n_r\}$. Designs with $n_s = 1$ are equivalent to simple restricted randomization and are used as a baseline. Designs with $n_s = n_r$ were those containing PSR_{max}, PSR-C and PSR-E algorithms. Similar to the one-factor design, we applied different ranges of eight different values of effect sizes η to different designs. To reduce the number of simulations, we specified identical ηs for main effects and interaction within each design. The five error structures (independent, exponential decay, gaussian random walk, pink noise, and mixed) and the procedures of determining ηs were the same as in Chapter 3.

5.3.2 Data-generating process

The value of dependent variable, `Y`, of subject i in level j of the row factor `R`, k of the column factor `C` on trial t is generated by the following formula (notice that jk combines to a condition in 2x2 factorial design).

$$Y_{ijkt} = \mu + S_{\mu i} + \beta_j + \beta_k + \beta_{jk} + S_{\beta j} + S_{\beta k} + S_{\beta jk} + e_{it} \tag{5.1}$$

In formula 5.1, μ represents the grand mean, $S_{\mu i}$ represents the random intercept of subject i; effect components β_j, β_k, and β_{jk} represent main effects of `R`, `C`, and the interaction `RC`, respectively; similarly, $S_{\beta j} + S_{\beta k} + S_{\beta jk}$ represent the random slopes of `R`, `C`, and `RC`, respectively. All random variances were specified to account for about 11% of the total variance, being around 44% of the total variance. This is aiming to maintain a typical portion of random effects in relation to total variance (about 40%, see section *Random effects in relation to residual variance* in the meta-analysis of Barr et al. (2013), Appendix). The term e_{it} stands for the error term of subject i in trial t, and its setting was identical as in Chapter 3.

5.3.3 Analysis

Fully crossing design parameters n_p and n_k yielded 6 unique designs. To simplify the presentation of our results, we chose three focal factorial designs to represent three most typical 2x2 within-subject designs. They are: the "smallest" design where $n_p = 28$, $n_r = 10$; the "medium" design where where $n_p = 40$, $n_r = 16$; and the "largest" design where $n_p = 50$, $n_r = 16$. These three designs, combining with five error structures and eight effect sizes, yielded 120 hypothetical experiments. Furthermore, since we only consider the setting as $n_s = \{1, n_r\}$ here, four PSR algorithms were considered, including (a) PSR_1 (common PSR with one subblock, equivalent to simple restricted randomization), (b) PSR_{max} (PSR with $n_s = n_r$, the maximum of all possible number of subblocks), (c) PSR-C, and (d) PSR-E. For PSR-C and PSR-E, starting conditions were counterbalanced across subjects except the largest focal design, because 50 is not a multiple of 4, leaving two sequences out from the total. This finally yielded 480 possible experiment settings. For each of these experiments, we conducted 10,000 Monte Carlo runs, within each run defined as generating and analyzing a single dataset. For the analysis, we conducted a two-way ANOVA by calculating F-ratios for two main effects and one interaction, $F_{(df_1, df_2)} = \frac{MS}{MSE}$. Here, MS and MSE are means of squares of corresponding factor and residuals, and are scaled by corresponding degrees of freedom df_1 and df_2, respectively. The p-values of `R`, `C`, `RC` were extracted with $\alpha = .05$. Identically to previous chapters, these p-values were used to estimate Type I error rates and power for all PSR variations and under different designs and error scenarios.

5.4 Results

5.4.1 Power

To estimate power, we calculated the portions of 10,000 runs that yielded significant effects for each of the two main effects and the interaction effect. We estimated power under PSR_{max}, PSR-C, PSR-E, benchmarking it against simple restricted randomization (i.e., PSR_1). Due to the large amount of data, we calculated median power increase rates for all three effects across the three focal factorial designs, presented in Figure 5.3. For temporally independent error structures, as expected, our three PSR algorithms yielded negligible power improvements against simple restricted randomization for all three effects, with median power improvements ranging from -1.3% to 1.1%. Regarding other four scenarios with autocorrelated error structures, the power improvements over simple restricted randomization ranged from -3.4% to 13.5%.

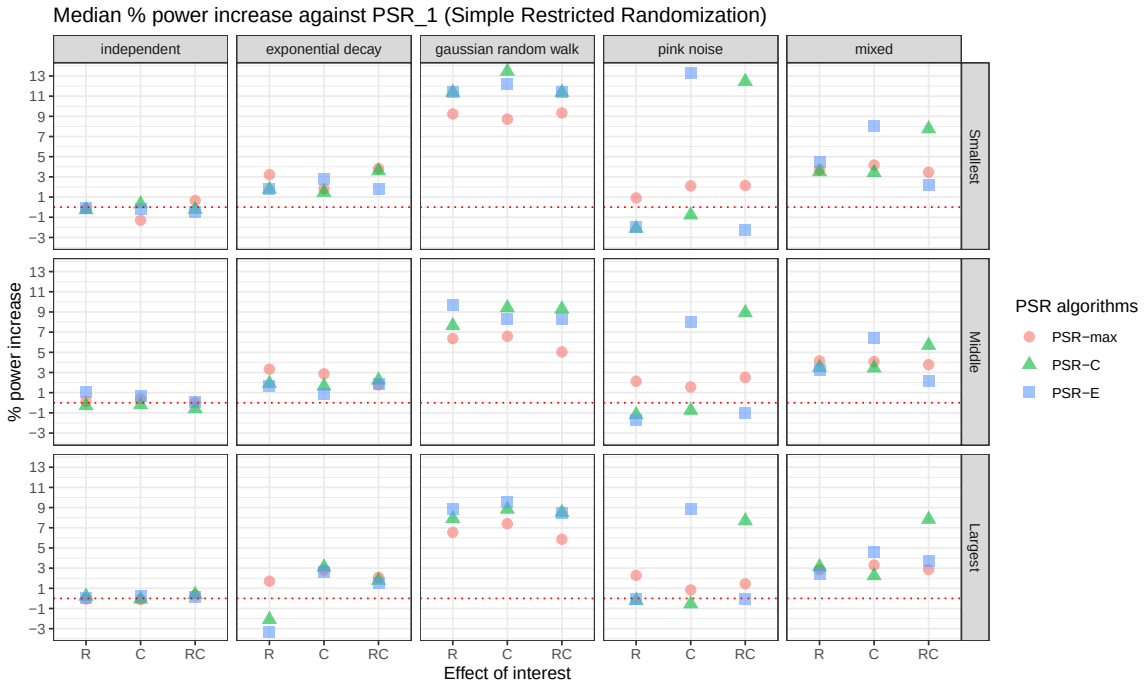

Figure 5.3: Median percent increase in power of different PSR algorithms against simple restricted randomization (PSR_1).

According to Figure 5.3, PSR_{max} turns out to be the only option among our three algorithms to guarantee power improvements for all effects, across all error structures and designs. Power gains were largest in gaussian random walk (5.0% to 9.3%), followed by mixed (2.8% to 4.2%), then exponential decay (1.7% to 3.8%), and finally pink noise (0.8% to 2.5%). As we expected, PSR_{max} distributed power roughly equally across the row, column, and interaction effects.

But compare to PSR_{max}, power gains of PSR-C and PSR-E seem to be more sensitive to the effect of interest. However, once extra power gain is achieved, PSR-C and PSR-E would show higher capacity than PSR_{max} has. For example, PSR-C boosts power for interaction *RC* in all cases, with the largest gain of 12.5% in smallest design under pink noise scenario; while using PSR_{max} yielded 2.1% power increase. In this case, PSR-C has 10.3% extra power gains

over PSR_{max}. At the same time, PSR-E can maintain consistent power improvements for the column factor C, with the largest gain of 13.3% in smallest design under pink noise scenario; while using PSR_{max} yielded 2.1% power increase. Therefore, with such settings, PSR-E has 11.2% extra power gains over PSR_{max}.

However, power impairments were also observed for PSR-C and PSR-E: in the case of largest design with exponential decay error scenario, PSR-C and PSR-E reduced the power of R by 2.1% and 3.4%, respectively. In all three designs under pink noise, PSR-C consistently impairs the power of R and C, ranging from -2.1% to -0.2%; while PSR-E consistently reduces the power of R and RC, with the range of ranging from -2.3% to -0.1%.

As expected, power gains for all three algorithms are largely determined by the types of error structures. However, unlike PSR_{max}, power gains of PSR-C and PSR-E are also determined by the number of participants n_p, and there seems to be a complex interaction between error structures and n_p. In exponential decay cases, increasing n_p from 40 to 50 (i.e., the middle focal design versus the largest focal design) harms the power for R for both PSR-C and PSR-E. However, increasing n_p would mitigate such impairment. In largest design, power losses for PSR-C and PSR-E were controlled to a negligible level, with all losses less than 1%.

5.4.2 Type I error rates

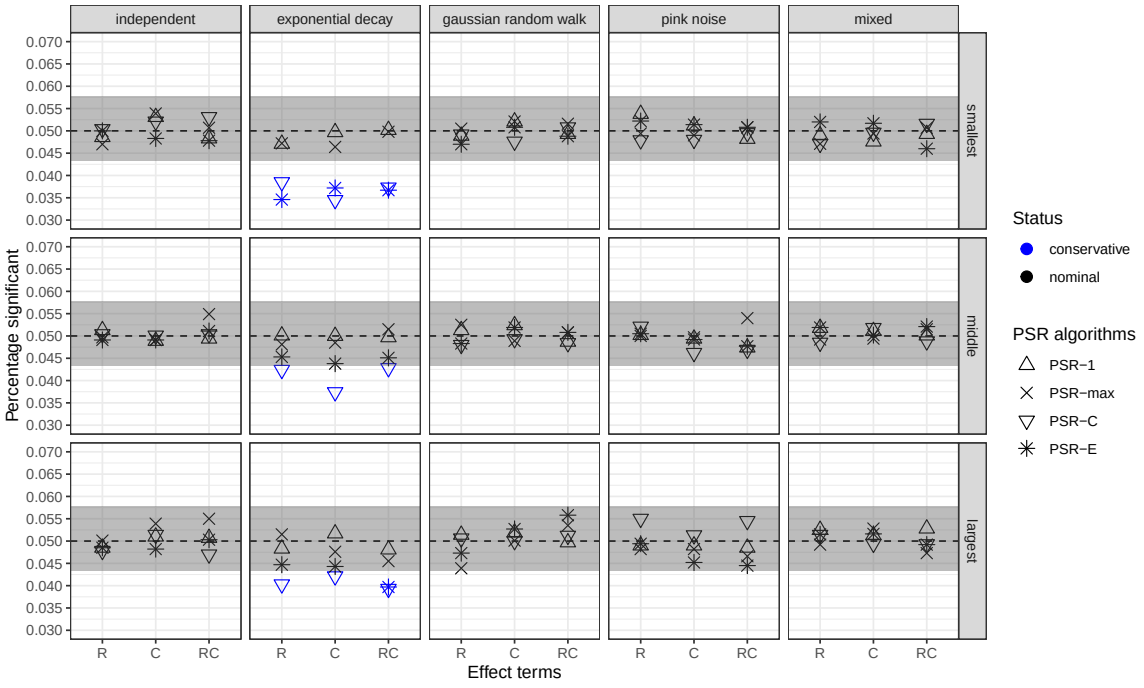

Figure 5.4: Type I error rates of PSR algorithms for effect terms, in different focal designs. The shaded regions represent the 99.9% Agresti-Coull confidence intervals for 10,000 random samples from a process with a "success" rate of .05.

We calculated the type I error rate by calculating the number of significant results in designs where $\eta^2 = 0$ (i.e., null effects). Figure 5.4 shows these results for our three focal factorial designs. We also computed the 99.9% Agresti-Coull confidence intervals with the "success rate" at the nominal level $\alpha = 0.05$, which is [0.0433, 0.0577] and marked by shaded regions.

First, in baseline scenarios where residuals were independent, the type I error rate range for simple restricted randomization (PSR_1) is [0.0483, 0.0531] across 3 designs and for all three effects of interest. This range falls in the 99.9% confidence interval, and no anti-conservative type I error rate was identified for any of effect of interest. Fortunately, none of our PSR algorithms exhibited anti-conservatism in independent cases, with type I error rate ranges of [0.0470, 0.0550] for PSR_{max}, [0.0470, 0.0531] for PSR-C, and [0.0478, 0.0511] for PSR-E, respectively. This results suggested that, in 2x2 factorial designs, even when errors were not temporally structured, it is safe to use our three algorithms.

However, since PSR_{max}, PSR-C, and PSR-E only partially remove time-dependent residuals, it is important to inspect whether the remaining autocorrelated residuals cause anti-conservative type I error rates when detecting effects of interest, especially for the latter two algorithms that involve consistently repeating sequences. Encouragingly, according to Figure 5.4, in our 12 cases where residuals were time-dependent, no algorithms indicated extra anti-conservatism over the upper limit of the 99.9% Agresti-Coull confidence interval, with type I error ranges of [0.0439, 0.0540] for PSR_{max}, [0.0345, 0.0550] for PSR-C, and [0.0346, 0.0558] for PSR-E, respectively.

However, the two new algorithms PSR-C and PSR-E showed conservativity in the exponential decay scenario for all effects of interest in all three designs. Both PSR-C and PSR-E showed conservative type I error rates for all effects in the smallest design. Increasing the number of participants ameliorated this somewhat for PSR-E except for the interaction term in the largest design. But PSR-C showed conservativity for all three effects across all sample sizes.

5.5 Applied example: Using PSR-C and PSR-E in experimental designs

As in one-factor designs, to implement PSR in 2x2 factorial designs, let us consider a hypothetical experiment with the `explan` package for R language (R Core Team, 2023). In this example, we use a factorial Stroop paradigm (Rosenbaum et al., 2017) to illustrate how to deploy PSR. Similar to the standard Stroop paradigm (e.g., Stroop, 1935), participants would see "congruent" word-color pairs where the font color matches the meaning of the word, and "incongruent" pairs where the font color mismatches the meaning of the word. But at the same time, participants need to identify the color while in one of two different positions (standing or sitting). These settings produce a standard 2x2 factorial design, with one factor being *congruency* (congruent versus incongruent), and the other being *position* (sitting versus standing). As such, combining all levels generates four conditions: *congruent-sitting*, *congruent-standing*, *incongruent-sitting*, and *incongruent-standing*. The standard finding of Rosenbaum et al. (2017) is that Stroop effect is smaller when participants are standing.

Before starting randomization, it would be useful to inspect how factors are combined and how the relative positions are determined in a sequence, as the these positions justify final PSR-C and PSR-E sequences. This inspection could be done using the `con_2x2()` function in `explan`. This function takes two arguments: (1) `facR`, a vector containing two levels of the first factor

(would be seen as the row factor); (2) `facC`, a vector containing two levels of the second factor (would be seen as the column factor).

```
## NB: first install the explan package using:
## remotes::install_github("dalejbarr/explan")
library(explan)


con_2x2(facR = c("congruent", "incongruent"),
        facC = c("sitting", "standing"))
```

```
            sitting                    standing
congruent   "congruent/sitting (1)"   "congruent/standing (2)"
incongruent "incongruent/sitting (3)" "incongruent/standing (4)"
```

According to the resulting matrix, we can see each entry contains a condition by combining one level of each factor. The numbers following determine how these conditions are originally arranged within each subblock (i.e., the original sequence without randomization).

Next, we need to randomize these conditions with our three algorithms, PSR_{max}, PSR-C, and PSR-E. For PSR_{max}, one can use `psr()` function mentioned in Chapter 3, and take all conditions as the level vector to generate independent sequences for every participant (results not shown).

```
lapply(seq_len(4),
       \(.x) psr(c("congruent/sitting", "congruent/standing",
                   "incongruent/sitting", "incongruent/standing"),
                 24, 4))
```

However, as PSR-C and PSR-E requires additional constraints to the relative positions of conditions, we developed another function, `walk_2x2()` in order to implement randomization. The `walk_2x2()` function helps user "walk" through the conditions by taking five arguments: (1) `np`, the number of desired participants; (2) `nr`, repetitions of each conditions; (3) and (4) being `facR` and `facC`, identical to those in `con_2x2()` function; and (5) `method`, a character determining the PSR algorithm, with `c` representing PSR-C and `e` being PSR-E. Based on these parameters, this function generates a list containing condition sequences for each participant under the chosen algorithm. The starting conditions are counterbalanced across participants.

```
walk_2x2(np = 4, nr = 4,
         facR = c("congruent", "incongruent"),
         facC = c("sitting", "standing"),
         method = "c")
```

```
[[1]]
 [1] "incongruent/sitting"  "congruent/standing"   "incongruent/standing"
 [4] "congruent/sitting"    "incongruent/sitting"  "congruent/standing"
 [7] "incongruent/standing" "congruent/sitting"    "incongruent/sitting"
[10] "congruent/standing"   "incongruent/standing" "congruent/sitting"
```

```
[13] "incongruent/sitting"  "congruent/standing"   "incongruent/standing"
[16] "congruent/sitting"

[[2]]
 [1] "congruent/standing"   "incongruent/standing" "congruent/sitting"
 [4] "incongruent/sitting"  "congruent/standing"   "incongruent/standing"
 [7] "congruent/sitting"    "incongruent/sitting"  "congruent/standing"
[10] "incongruent/standing" "congruent/sitting"    "incongruent/sitting"
[13] "congruent/standing"   "incongruent/standing" "congruent/sitting"
[16] "incongruent/sitting"

[[3]]
 [1] "incongruent/standing" "congruent/sitting"    "incongruent/sitting"
 [4] "congruent/standing"   "incongruent/standing" "congruent/sitting"
 [7] "incongruent/sitting"  "congruent/standing"   "incongruent/standing"
[10] "congruent/sitting"    "incongruent/sitting"  "congruent/standing"
[13] "incongruent/standing" "congruent/sitting"    "incongruent/sitting"
[16] "congruent/standing"

[[4]]
 [1] "congruent/sitting"    "incongruent/sitting"  "congruent/standing"
 [4] "incongruent/standing" "congruent/sitting"    "incongruent/sitting"
 [7] "congruent/standing"   "incongruent/standing" "congruent/sitting"
[10] "incongruent/sitting"  "congruent/standing"   "incongruent/standing"
[13] "congruent/sitting"    "incongruent/sitting"  "congruent/standing"
[16] "incongruent/standing"
```

As with a one-factor design, it is most commonly the case that researchers would like to apply PSR directly to stimuli, not just to conditions. In this case, they can turn to our factorial randomization function, `psr_2x2_stimuli()`. The `psr_2x2_stimuli()` function requires behaves quite closely to `psr_stimuli()`, and the researcher must pass a data frame containing the full set of stimuli delivered to each participant. To illustrate, we use another built-in table in `explan`, confirming to the factorial Stroop design, `stroop_stimuli_factorial`.

```
head(stroop_stimuli_factorial, 6)
```

```
  stimulus_id  word font_color congruency positions
1           1  blue       blue  congruent   sitting
2           2  blue       blue  congruent  standing
3           3 brown      brown  congruent   sitting
4           4 brown      brown  congruent  standing
5           5 green      green  congruent   sitting
6           6 green      green  congruent  standing
```

To implement PSR with the table of stimuli, four arguments are mandatory: (1) `stim_table`, the name of stimulus table; (2) `IVs`, two independent variables (factors) to be considered, with the first one being the row factor, and the second one being the column factor; (3) `n_part`, the number of desired participants; and (4) `algorithm`, the desired PSR-algorithms, with `max` representing PSR_{max}, `circular` representing PSR-E, and `eight` being PSR-E.

```
set.seed(1451) ## ensure reproducibility of the output

stroop_fac_4 <- psr_2x2_stimuli(stim_table = stroop_stimuli_factorial,
                                IVs = c("congruency", "positions"),
                                n_part = 4,
                                algorithm = "circular")

head(stroop_fac_4, 9)
```

```
  PID sb_no stimulus_id  word font_color  congruency positions
1   1     1          10  blue       blue   congruent  standing
2   1     1          72   red      green incongruent  standing
3   1     1          63 green      brown incongruent   sitting
4   1     1          27 brown      brown   congruent   sitting
5   1     2          18  blue       blue   congruent  standing
6   1     2          50  blue      brown incongruent  standing
7   1     2          49  blue      brown incongruent   sitting
8   1     2          13 green      green   congruent   sitting
9   1     3          48   red        red   congruent  standing
```

The output above has the same structure as that from `psr_sim()`. But if we take a further look at the independent variables (i.e., congruency and positions), we find that they are randomized by PSR-C, and their orders are determined by the numbers obtained from the output matrix of the `con_2x2()` function[5.1].

5.6 Discussion

In this chapter, we extended the PSR algorithm into the context of 2x2 factorial designs, and introduced two PSR variations, PSR-C and PSR-E. Compared to one-factor design, time-varying fluctuations have more complex masking effects in factorial designs, but there are also possibilities for arranging conditions to improve power for certain effects. Since 2x2 factorial designs are one of the most frequently used designs in psychology and neuroscience, adapting PSR to this context would be advantageous for improving measurement and thus increasing the reproducibility of lab experiments while keeping false positive rates at a nominal level.

As a direct generalization, PSR_{max} was the only algorithm guaranteeing the *general power*

[5.1] The function `psr_2x2_stimuli()` could still generate sequences even when `np` is not a multiple of 4. In this case, users would receive a warning message from the function, suggesting it is an unbalanced design

improvement across all effects of interest. The results of PSR_{max} would be useful wherever main effects and their interaction have the same weight of research interest. PSR-C and PSR-E showed clear power advantages in detecting specific effects of interest when error structures are participant-specific rather than general (i.e., except in the exponential decay scenario). One characteristic of PSR-C and PSR-E is, they choose a fixed starting condition and follows a given path to move across conditions, and the resulting sequence within a subblock is repeated in independent sequences for each participant. Although the starting conditions are counterbalanced across participants, the sequence is repeats identically. Thus, it would be ill-advised to use these approaches as is, with actual participants. Furthermore, in contrast to PSR_{max}, they also risk to bring in spectral alignment with time-varying patterns. As previously discussed in Chapter 3, when time-varying noise happens to have a strong spectral component, then power might be negatively affected, although the amount could be trivial. The pink noise scenario represents this situation, where the residual time-series spectrum has an overall fractal structure.

Given this possibility of spectral coherence, would within-participant counterbalancing of the subblock-level sequences have any benefit? This is doubtful, especially if we explore the space of possibilities for within-participant counterbalancing. First, as earlier explained in introduction section of this chapter, randomly choosing for each subblock whether the apply PSR-C and PSR-E, which cell to start in, and in which direction to move is effectively the same as applying PSR_{max}. Second, it is possible to generate "palindrome" PSR-C sequences (e.g., clockwise arranging conditions in one subblock then anti-clockwise arranging conditions in the next subblock) and create sequences like $AB - Ab - ab - aB - AB - aB - ab - Ab$ and so on; but such sequences lose the even distribution of conditions over time (e.g., two aBs are quite closed to each other, and Bs have been repeated for three times). We believe palindrome sequences would be no better than standard PSR-C or PSR-E sequences, as predictable condition streaks appear at the margins between subblocks. However, for PSR-C and PSR-E, complete between-subject counterbalancing in starting points and motions might be beneficial to fully clean up time-dependent variance on the sample level. But doing so requires a sample size at least double of we originally have to achieve a fully balanced design. Given that conditions are not independent as in one-factor design, how to arrange stimuli and better cancel out time-varying noise becomes more challenging. But our results have shown power advantages under all possible movements in a fully within-subject 2x2 factorial design.

In this chapter, we only considered design-based approaches for dealing with time-varying fluctuations in 2x2 factorial designs. But model-based approaches for 2x2 factorial designs are also possible, and could even be combined with design-based approaches, as we did for the one-factor design in Chapter 4. As time-dependent noise is partially removed via PSR algorithms, using statistical modeling to remove remaining time-dependent noise might be particularly useful. Future research is needed to see how this might work.

According to Chapters 3 to 5, we have seen that involving time-varying fluctuations could improve measurement by boosting power in repeated-measure within-subject experiments. One

characteristic of these experiments is that observations have temporal characteristics (e.g., reaction time) and would be obviously affected by time-dependent effects as the task unfold. But in psychology and other behavioural sciences, there is often seen another kind of dataset where observations are time-dependent: specifically, psychometric instruments implemented through questionnaire reports. In these types of datasets, observations are taken to measure psychological properties of participants (e.g., attitudes or abilities). Although observations from a questionnaire would are intended to measure these presumably static and long-term properties of participants, time-varying fluctuations could also affect measurement as the observations must be taken sequentially. In the next chapter, we focus on time-varying fluctuations in psychometrics research, another field that has generally ignored time-dependent noise. We introduce a model-based approach to improve measurement in these contexts.

Part II: Psychological Measurement

Chapter 6

Cleaning up Psychometric Measurement with Mixture Autoregression Confirmatory Factor Analysis

6.1 Background

So far, we have seen how controlling time-varying fluctuations can improve measurement by boosting power in repeated-measures experiments. The approaches we reviewed improve measurement by accounting for the *non-simultaneity of measurement*. But the non-simultaneity of measurement also affects study quality in other areas, such as in psychometrics, where an instrument, usually a questionnaire, is used to assess a psychological property of a participant. Just like in an experiment, participants repeatedly respond to items that are designed to measure one specific construct—it could be a main construct, or some facet of it. As time-varying fluctuations lead to systematic changes irrelevant to the effects of interest in experimental datasets, they would also do so in a questionnaire dataset, especially in a cross-sectional survey. As mentioned in Chapter 1, reliability requires a clear psychological construct; thus, determining the validity of a measurement is prior to an assessment of its reliability. In this chapter, I will introduce a model-based method to increase the validity of the psychometric analysis by considering time-varying fluctuations, as model-based methods are currently the most common and acceptable solutions in psychometrics[6.1]. But before going into the technical section, some general background is require to better understand the motivations of this study.

Generally, to determine the validity of a psychological measurement is to confirm whether this measurement reflects the construct it is claimed to measure. This process is completed by a variation of multivariate regression, *factor analysis*. In most behavioural research, the construct is proposed earlier than the test so that a factor analysis is often confirmatory, and it is therefore called Confirmatory Factor Analysis (CFA) thereby. Nowadays, CFA perhaps has become the most popular approach to validate a psychological measurement (Coulacoglou &

[6.1] Design-based approaches to improve measurement in psychometrics are still in debate and are difficult to implement. For further discussion about this part, see next chapter.

Saklofske, 2017).

Essentially, CFA is a multivariate linear regression model, and researchers justify the test validity by inspecting model fits and *factor loadings* (i.e., the regression coefficient of each item, represented by the arrow from the latent variable to the indicator y_i in panel a of Figure 6.1). As in other linear regression models, CFA provides Chi-square (χ^2) value to determine how well the theoretical model represents the sample. It also provides Akaike Information Criterion (AIC) and Bayesian Information Criterion (BIC) for basic model comparison. Specifically however, CFA additionally provide Comparative Fit Index (CFI), and Tucker–Lewis Index (TLI) to estimate how better the theoretical model does in explaining the dataset at hand than the baseline model where all variables are independent. Determining a questionnaire's validity requires a combination of adequate values of all these fits and parameters. Factor loadings represent the relationship between the construct and individual items. For example, if an item has a factor loading value of 0.70 on a given latent variable, then this latent variable has accounted for $0.70^2 = 49\%$ of the variance of this item.

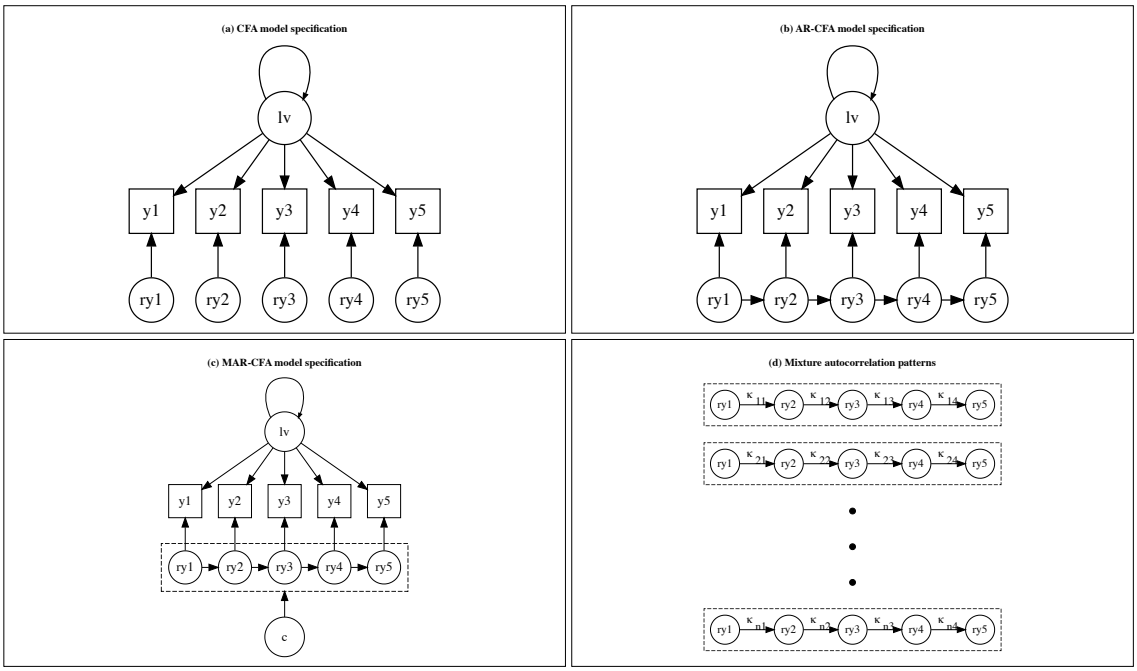

Figure 6.1: Model specifications of (a) conventional CFA, (b) AR-CFA as purposed by Ozkok et al. (2019), (c) MAR-CFA, and (d) mixture autocorrelation patterns. Circles defines latent variables that cannot be directly observed or that have to be indirectly estimated. Squares reflects observations that are obtained from the raw dataset. Arrows specified the direction of causation, pointing from predictors to indicators. Arrows starting from a latent variable and pointing to itself reflects the variance of the latent variable. Rectangles constructed by dotted lines suggested a latent cluster that is implied by MAR-CFA. lv = Latent variable. y = item (indicator). ry = residual of an item. κ = autoregression coefficient.

Despite the wide use of CFA, researchers usually find poor results from their analysis. For example, one might observe satisfying model fits but low item factor loadings from the CFA results. Usually, this situation can be accounted for by model misspecification, poor performance

of some items, or potential outliers in the datasets. At this point, test developers and analysts often see it as a positive signal and start checking the dataset, re-specifying the analytical model, and even reconsidering the theoretical construct itself, thereby refining the measurement. But for some other cases, researchers observed poor model fit and high factor loadings. This phenomenon suggests that items are measuring a hybrid of the construct but with some unexplained relationships among items. To control these potential relationships, researchers can split the measured construct into two or more components (formally stated, they add new factors) that are correlated, letting different items measure different components, and therefore yielding multi-factor models. Or they can specify some correlated items according to the model modification indices suggested by software/programming languages. In either way, researchers need to control unexplained relationships by taking them out from the original construct. But because both ways are highly data-driven, it is hard to determine if the modification really improved the validity and helped generalize the findings, or whether the improvement reflects special properties of the dataset being analyzed (Chin, 1998). Also, model modification is often an empirical endeavor and is less appreciated unless strong theoretical evidence is provided. In practice, modifications reduce confidence in the future use of the measurement being validated, as no clear factor structure or contradictory factor structures would be proposed (cf. Liang et al., 2022; Wang et al., 2020).

Modifying the original construct is not always viable, but paying attention to the sample-specific effect sources might be helpful, as we are trying to separate the homogeneity explained by measured construct from that caused by sample-specific properties. As we have already seen in experiments, time-varying fluctuations affect participants' performances. Participants still need to react to the test item by item, their responses would naturally form a time series, and such performances are unlikely to stay stable.

Historically, however, very little research has considered how dealing with time-varying fluctuations can help statistically increase the validity of CFA-based approaches. Instead, researchers behave as though responses from each participants are collected simultaneously when specifying a CFA model. But this is far from the truth, especially given that question ordering and other serial effects are a research topic in its own right: for example, long-term contextual effects and item wording effects, social desirability, and so on (Marsh, 1996; Pedregon et al., 2012). Although in another approach to investigate the relationship between measured construct and items, item response theory, researchers have introduced time-fluctuations into the analytical framework (e.g., Myszkowski & Storme, 2024), its application still requires a solid psychological construct that items within a scale measure, and the responses to an item are assumed independent of responses to other items once their contribution to the construct is taken into account (i.e., unidimensionality assumption and local independence assumption, Nguyen et al., 2014). Testing these requirements is usually completed by CFA. Therefore the item response theory analysis would be biased if the measurement validity is already biased by a CFA that ignores time-varying fluctuations.

6.2 Autoregressive CFA

Ozkok et al. (2019) noted the limitations in CFA, and they were the first to introduce the idea of time-varying fluctuations into the model, resulting in the Autoregressive CFA (AR-CFA) approach. According to Ozkok et al. (2019), the item response is described as a sequential process, and they model such process by specifying a residual structure with autoregression (i.e., the residual of an item would be predicted by the residual of its previous item, reflected by the arrow existing in residuals ry_n in panel b of Figure 6.1). Except for the residual specification, the remaining settings are identical to a conventional CFA. Under their model comparisons and a Monte Carlo simulation study, (Ozkok et al., 2019) found that conventional CFA usually overestimates the factor loadings and worsen model fit results when time-varying fluctuations are present (quantified by autoregression coefficients); while AR-CFA has satisfying performance in improving validity. Such improvement is achieved by applying "... a more theoretically rigorous approach to model specification... while also offering ways to balance the difference between highly a constrained IC-CFA vs. other less-constrained approaches such as an EFA [Exploratory Factor Analysis]." (p. 16).

The AR-CFA has two appealing advantages. First, its settings are understandable and simple—the fundamental time-varying fluctuations and response processes were only reflected in residual term specification and thus, the factor structure component stays clear. Second, compared to the data-driven modification approaches, specifying autoregressive residual structure is highly theoretically supported-researchers are trying to recover the true process happening in a test completion. In fact, Ozkok et al. (2019) even discussed the settings in multi-factor models where a construct is measured by more than one components, and items under different factors can have autoregressive relationships. Although this scenario represents the possibilities to investigate further interactions between time-varying fluctuations and the measured construct, it is beyond our topic for the model complexity and will not be discussed further.

6.3 Mixture AR-CFA: A compromise between "None" and "All"

According to the specification of AR-CFA, only one general autocorrelation pattern was included throughout all participants. But in reality, this is very unlikely to happen due to individual differences. For example, as the task goes on, task-takers might consistently increase their attention, or might lose interest; they can expend more effort to rate items or they just provide very casual, biased responses because of fatigue. But trying to capture all these changes would be very difficult, and is likely to exceed researchers' knowledge and modelling techniques. There is, however, an R package `mxsem` (Orzek, 2024) that provides algorithms to allow the specification of a Moderated Nonlinear Factor Analysis (Bauer, 2016; Kolbe et al., 2024) whose model parameters (e.g., factor loadings, regressions, covariances) can vary across participants, and thus could potentially capture all possible time-dependent noise. But the general application of the MNLFA requires a known source of individual differences (e.g., gender, age); yet there is no such reference variable in a questionnaire dataset that help determine

time-varying fluctuations, as they are invisible and embedded in the raw response series. Experienced analysts might try to specify random effect structures to capture individual differences, but they would be disappointed by the lengthy, complex, and error-prone programming and the overwhelmingly heavy computational burden (see "Moderated-Nonlinear-Factor-Analysis" vignette of `mxsem` package for the model syntax). At the simplest structure that only has three items, MNLFA needs to additionally estimate two linear mixed effect models with its minimal setup given the same amounts of information (and it is highly possible for these models to be under-identified). Any increase in item number or sample size would also slow down to the factor model estimation speed. Once it is appreciated that the return of estimating a complex model imposes high complexity and effort, the potential of MNLFA is doubtful.

But in between fully ignoring time-varying fluctuations and exhaustively estimated individual differences, we can look for a compromise to clean the time-varying fluctuations while maintaining the model as much as we can. For example, based on the AR-CFA, we can model several additional time-varying fluctuations patterns that are shared by several groups of participants, as an attempt of approaching individual differences. In order to model these patterns, we here introduce Mixture AR-CFA (MAR-CFA). Basically, MAR-CFA maintains the same factor structure specification in AR-CFA, but we apply a finite mixture model (FMM) to estimate autoregressive residual structures. As the most important difference between MAR-CFA and traditional CFA is the FMM framework, we will focus on this approach. A standard FMM can group a population into several subsets based on a heterogeneity source and allow certain parameters and/or relationships to vary across subsets (formally, the subset here is stated as *cluster*, and we will call it henceforth). Panel c of Figure 6.1 represents the model specification of MAR-CFA, whose residuals and autoregressions are seen as clusters, surrounded by rectangle with dotted edges, and the cluster membership is predicted by a new latent variable c. Panel d of Figure 6.1 represents how the autoregression coefficient (denoted by κ_{in} where i represents the ith cluster and n represents the coefficients from the nth item to $n+1$ item) vary across different clusters. At the first glance, FMM is similar to multigroup analysis where analysts can specify invariant or changing parameters across groups that are defined by a reference variable. However, in FMM, the heterogeneity source for identifying population membership is *unknown*, and the population membership is inferred from the data. That being said, no pre-defined population membership is in the original dataset.

6.4 The expectation maximization algorithm

The estimation of FMM is performed using the Expectation Maximization (EM) algorithm (Dempster et al., 1977). Basically, fitting a model is to find a mathematical solution (i.e., a parameter set) to describe our sample (the dataset at hand). However, it is hard to do so in mixture models, as the sample distribution is a combination of multiple sub-distributions. Instead, the EM algorithm attempts to provide the solution with the largest log-likelihood in a likelihood function across every possible parameter value, and the solution with the largest log-likelihood is called *global maxima*, and sample at hand would have the highest probability

to be obtained at this case, comparing to all other possible solutions. To achieve the largest log-likelihood, the EM algorithm iteratively estimates the model and updates the model input based on its previous execution until the stopping rules are met. These rules include a maximum number of iterations (how many times the EM algorithms should run) and a convergence criterion (how small the changes in log likelihood between nearby executions are small enough to ignore).

To provide an intuitive understanding behind this iterative mechanism, let us use the "mountain climber" analogy in Masyn (2013). Imagine the likelihood function is a mountain range, and the EM algorithm is a mountain climber who aims to climb to the highest peak of the range (find the global maxima). However, the climber does not know where is the peak of this mountain from the base. So the climber collects available information (dataset at hand) and chooses a reasonable starting point (the initial starting values of parameter estimates) to start climbing step by step (estimate the model). With each step, the climber would stop and consider which following step he could access (adjacent parameter sets) takes him to a higher point (larger log likelihood), and then the climber moves to that point. The climber repeats this process until he found a point that any following step of which would either take him to a lower or not apparently higher. At this point the climber thinks he reaches the peak of the mountain range and plants a flag there (the model convergence criterion is met). However, a climber cannot take this trek forever-the life supplies are limited (the maximum number of iterations). If the supplies are ran out before the climber reaches the peak (the algorithm exceeds the maximal number of iterations before the convergence criterion is met), the climber cannot continue the journey. And because there might be other possible points that take him higher, the climber would not plant the flag at the current point (model fails to converge).

However, a mountain range can have many peaks with different heights. So it is possible that the climber reaches a peak he thinks it is the highest and runs out all supplies spinning around, while the true highest peak for the whole mountain range is in somewhere else. Mapped the EM algorithm, a model can converged to a log likelihood that is the maximum of a given range, but not of the whole likelihood function. The log likelihood observed in this case is called *local maxima*. So how does a climber know the peak he reaches is the highest one of the mountain range, instead of a given area? The answer is he cannot really know with only one attempt. Instead, he can verify it by climbing with different starting point and routes again and again. Different routes might take the climber to different peaks, but if all the routes point to the same peak, then the climber has less possibility to make mistake. In practice, to avoid local maxima, researchers would use a set random start values to estimate the model at the initialization stage, as start value selection can significantly affect model estimation results (Shireman et al., 2017). And it is suggested that if the maximal log likelihood is repeated for at least twice, it can be considered as the global maxima and the parameters under this solution can be interpreted with confidence (Nylund et al., 2007). With each time the EM estimates the model, it would calculate the posterior probabilities that which individual belongs to which cluster. Therefore the final estimation results could have the record of these posterior probabilities, and researchers

could draw inference of latent heterogeneity by inspecting these probabilities.

Maybe the most well-known examples of FMM in psychology are Latent Class Analysis (LCA) where observed indicators are categorical (e.g., Atroszko et al., 2021), and Latent Profile Analysis (LPA) where the observed indicators are continuous (e.g., Shukla et al., 2018). Both LCA and LPA are useful tools to illustrate how the measured psychological construct varies in different levels on a model-implied latent predictor. Using LPA, Wang et al. (2023) illustrated the relationship between psychological flexibility and depression, anxiety and stress among Chinese college students. Results suggested that students' flexibility has obvious group heterogeneity, and students in different profiles of psychological flexibility (i.e., different levels of a model-implied psychological flexibility clusters) exhibited significantly different depression, anxiety, and stress levels.

Although sharing the similar FMM fundamental of LCA and LPA, our aim with MAR-CFA is slightly different: we would like to capture the meaningful uniqueness and homogeneity of the psychological construct as much as possible by taking off the trivial heterogeneity within clusters instead of explaining the time-dependent residual structures. Time-varying fluctuations naturally exist during a test, regardless of the sample. And for questionnaire development or validation, the major focus is the construct itself. Therefore, the idea of MAR-CFA is that the factor structure should stay stable and invariant across clusters, but the autoregressive residual structures can vary across clusters. Within each cluster, the autoregression coefficients are the same for each participant. We considered unchanged factor loadings for theoretical reasons. First, we would like to know how much the validity and measurement can be improved when time-varying fluctuations are the only known nuisance variation. Second, the effects of time-varying fluctuations are assumed to be small and do not seriously affect the construct itself-if they do, researchers should first consider the robustness of the theoretical construct that being measured or potential interactions between the measured construct and external constructs.

In this chapter, we conducted a Monte Carlo simulation study to address two questions: First, what happens when time-varying fluctuations are present but ignored? To answer this question, we compared the model performances of modelling techniques that ignore versus consider temporal characteristics (i.e., conventional CFA vs. AR-CFA vs. MAR-CFA). Second, if time-varying fluctuations cannot be ignored, in what way and to what extent can controlling them help increase the validity of measurement?

6.5 Methods

6.5.1 Determining parameters

As an early investigation into this area, we would like to maintain a simple setting. Therefore, in the current study, we considered an one-factor structure where one latent variable is measured by ten items, and the true factor loadings (λ) are identical for each item. This number of items ensures enough degrees of freedom to identify our MAR-CFA model. We included three values of true factor loadings, 0.5, 0.6, and 0.7, to mimic real measurement parameters (i.e.,

$\lambda \in \{0.5, 0.6, 0.7\}$). Regarding time-varying fluctuations, three designs were included: in design (a), all participants shared a autocorrelation pattern with a regression coefficient (κ) of 0.175. In design (b), participants have two shared autocorrelation patterns (i.e., two clusters) with different strengths $\kappa_1 = 0.05$, $\kappa_2 = 0.3$. In design (c), participants have three shared autocorrelation patterns (i.e., three clusters) ($\kappa_1 = 0.05$, $\kappa_2 = 0.175$, $\kappa_3 = 0.3$). We also included a baseline scenario where residuals are independent to time. For all designs, five sample sizes (N) were included, $N \in \{300, 600, 900, 1,200, 1,500\}$, to represent different scales of psychometric research. For designs including more than one cluster, the numbers of participants is the same across clusters. For example, in design (c) with 1,500 participants, each cluster has 500 participants. Fully crossing these parameter settings yielded 60 designs. To simplify simulations, we did not consider the interaction in between time-varying fluctuations and the latent variable. The main topic of interest is to raise the attention of time-varying fluctuations in response process, not to discuss a certain relationship in between such fluctuations and the psychological construct. This complex issue exceeds the scope of this chapter.

In terms of analytical models, we considered four models: conventional CFA model; AR-CFA model as purposed in Ozkok et al. (2019); MAR-CFA model with two clusters (referred as MAR-CFA$_2$); and MAR-CFA containing three clusters (referred as MAR-CFA$_3$). The reason for additionally considering a two-cluster CFA is that mixture models often involves a class number decision, require comparing models with different numbers of clusters. And the reason for *not* considering models with more the three clusters is simpler models are often preferable in model comparisons. When too few participants are clustered into a latent cluster, a model with fewer clusters but allocating more participants within might be preferred (Nylund et al., 2007).

6.5.2 Data generation

Directly generating datasets with mixture components is difficult, as we need to simultaneously generate multiple distributions with unknown means. In addition, it is unclear how autocorrelation strength affects latent score means, therefore we cannot determine the relative positions of simulated latent score distributions. Instead, we simulated multiple datasets with different autocorrelation strengths and combine them to form a final simulated mixture dataset that enters analysis. This manner breaks down the simulation process into separately generating multiple AR-CFA datasets, and we are now able to do so following the (modified) formulae purposed in Ozkok et al. (2019):

$$\mathbf{y_i} = \nu + \mathbf{\Lambda}_\eta \eta + \mathbf{\Lambda}_\epsilon \epsilon_\mathbf{i} \tag{6.1}$$

$$\epsilon_\mathbf{i} = \kappa \epsilon_{\mathbf{i-1}} + \mathbf{u_i} \tag{6.2}$$

Formulae 6.1 and 6.2 represents the algebraic form of the generation process, where $\mathbf{y_i}$ is an

observed array of scores of the ith item, ν is the grounded mean of the item and is fixed to zero for simplicity, $\mathbf{\Lambda}_\eta$ represents the factor loading matrix of the latent variable η, $\mathbf{\Lambda}_\epsilon$ is a symmetric matrix to capture all latent residuals. The $\mathbf{\Lambda}_\epsilon$ fixed on-diagonal elements to unity, but we also fixed off-diagonal elements to zero as no covariances across residuals were considered so that $\mathbf{\Lambda}_\epsilon$ becomes an identity matrix $\mathbf{I}$. Next, the term $\epsilon_\mathbf{i}$ represents the residual array of the observed raw scores, and it is calculated by summing its random residual component $\mathbf{u_i}$ and the AR component-the residual array of its previous item $\epsilon_{\mathbf{i-1}}$ times the autoregression array κ. As only one latent variable is considered in our design, The random residual component $\mathbf{u_i}$ follows a normal distribution, $\mathbf{u_i} \sim N(0, \mathbf{\Theta})$, and $\mathbf{\Theta}$ is a diagonal matrix of residual covariances. The covariance structure of MAR-CFA is represented by the following formula:

$$\mathbf{\Sigma} = \mathbf{\Lambda}_\eta \mathbf{\Psi} \mathbf{\Lambda}_\eta{}' + \mathbf{\Lambda}_\epsilon (\mathbf{I} - \kappa)^{-1} \mathbf{\Theta} (\mathbf{I} - \kappa')^{-1} \mathbf{\Lambda}_\epsilon{}' \tag{6.3}$$

where $\mathbf{\Sigma}$ is the model-implied covariance matrix and $\mathbf{\Psi}$ is the latent covariance matrix. As our simulation only considered one-factor structure, $\mathbf{\Psi}$ is equivalent to the variance of the latent variable η and we fixed it to 1 for CFA model identification.

Data generation was performed with the `R` (R Core Team, 2023) package `simsem` (Pornprasertmanit et al., 2021). Notice that we used the path analysis framework to generate our data by specifying `modelType = "Path"` in `simsem::model()` function. Residuals cannot be observed and thus are seen as latent variables in `simsem`. However, `simsem::model()` cannot specify regressive residual terms by general `modelType = "CFA"` and `modelType = "Sem"` arguments. The first step of data generation is create the path matrix for the MAR-CFA model. In this matrix, the regression coefficient of the latent variable towards itself is 1, and any regression coefficient from the latent variable to an exogenous variable (i.e., an item) represents the factor loading of this item, while any coefficient from the $i - 1_{th}$ item to the i_{th} item represents the AR coefficient. All other elements in this matrix were fixed to 0 to represent an independent relationship except for manual specifications. Next, we generated the residual matrix by using a wrapped `simsem::findFactorResidualVar()` function. This function calculates the (latent factor) residual variances from the path matrix and factor (residual) correlations. These two matrices would be taken by `simsem::model()` function to create `simsem` style model template and as references of `simsem::generate()`, the main function to generate data. As mentioned, when considering cluster data with different κs, we separately generated subsets and vary κs within each cluster. The changes were completed in the path matrix generation. In `simsem::generate()`, such changes yielded a multigroup template for data generation and the function would automatically generate multigroup datasets, which a new variable would be created to label cluster membership of each observation. During data generation process, a built-in R package `parallel` was used to enable parallel computation and boost the generation speed.

6.5.3 Analysis

Generated datasets were fitted by multiple models mentioned above via Mplus 8.3 (Muthen et al., 2017) as this is the most general and flexible software that fits FMM by the EM algorithm. For each hypothetical setting mentioned above, the models fitted would vary: conventional CFA models for the baseline model where no autocorrelated residual patterns were included and for the AR-CFA model where datasets were generated with a fixed AR effect term were considered across all autocorrelated residuals; a AR-CFA models where a set of autocorrelated residuals would be considered, with the regression coefficient fixed; and MAR-CFA_2. Within each cluster, autocorrelated residuals and item residuals were allowed to be freely estimated. For datasets containing two and three latent autocorrelated residual specifications, CFA, AR-CFA, MAR-CFA_2 and MAR-CFA_3 would be considered as analytical models. These settings finally yielded 210 combinations of hypothetical questionnaire designs and analytical models.

Model selection was done by aBIC fit index comparison given its satisfying performance in detecting the correct number of clusters over other indices (Tein et al., 2013). The conventional CFI and TLI indices were not considered because sample means, (co)variances are not available (not sufficient statistics) in mixture models, and the Chi-square and following statistics based on this statistic are not provided. Also the entropy value is also recorded as a measure of classification quality. The entropy ranges from 0 to 1, with higher values representing better classification results. Although not the main topic of interest, it is also helpful to inspect if MAR-CFA can deal with trivial time-dependent noise after controlling them. Besides fit indices, we also conducted the Lo-Mendell-Rubin test (LMR, Lo et al., 2001) and the bootstrapped likelihood ratio test (BLRT) to compare (nested) mixture models with $k-1$ clusters versus k clusters. For LMR and BLRT, a p-value less than 0.05 means the model with k clusters are preferable.

For each of the hypothetical designs, we ran 500 Monte Carlo runs to evaluate the average factor loading recover performance and the model selection capacity. Each run consisted of generating a dataset according to our parameter set and analyzing it with models mentioned above. To avoid local maxima, we increased the number of starting value sets by specifying `STARTS = 30 6;` in Mplus[6.2]. This command means Mplus would randomly generate 30 parameter sets for at the initialization stage, and across these 30 sets, 6 of them that yielded a model with the largest log-likelihood at the first round of iteration would be included into the final optimizations, say, subsequent iterations would be carried on these 6 sets. The number of parameter sets were smaller than the usual recommendation (e.g., `100 20`) as we have to balance the sensitivity and the heavy computational burden. As a remedy, we narrow down the convergence criteria (i.e., from the default 0.00001 to 0.000001) to ensure each iteration achieves a considerable model improvement. Instead of manually switching in between R and Mplus, we used the R package `MplusAutomation` (Hallquist & Wiley, 2018) to allow cooperation between both.

[6.2]The default for Mplus to estimate mixture models is `STARTS = 10 2;`.

6.6 Results

6.6.1 Convergence

Before presenting simulation results, it is necessary to inspect model convergence performance, since mixture models involve iteration processes and start value specifications. Across all 190 analytical models, 170 (89.5%) had a convergence rate over 85%. However, 20 (9.5%) designs combining $\lambda = 0.7$, true numbers of cluster in 2, 3, and analytical numbers of clusters in 2, 3 had very poor convergence rates, ranging from 2.2% (with the sample size of 1,500, 3 true clusters, analyzed with a 2-cluster mixture model) to 88.8% (with the sample size of 300, 3 true clusters, analyzed with a 2-cluster mixture model). Further model inspection suggested that poor convergence rates might be attributed to non-positive defined covariance matrices. To ensure the accuracy of results, data from these 20 models were excluded and will not be discussed further.

6.6.2 Factor loading recovery performances

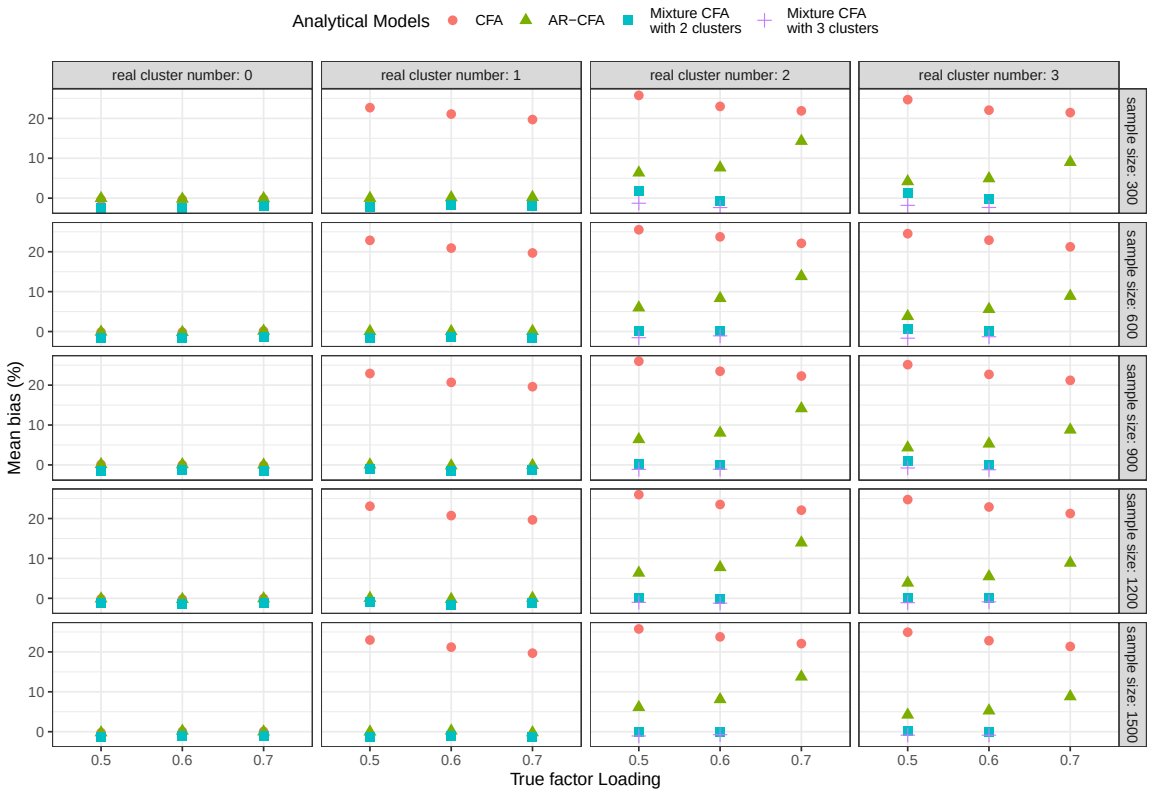

Figure 6.2: Mean bias of factor loadings for each model. A positive mean bias means the factor loading were overestimated, and vice versa. Panels in each row represents datasets generated from the same sample size. Panels in each column represents datasets generated from the same factor structure. Data from non-convergent models was omitted.

Figure 6.2 presents factor loading recovery performances for each model. In baseline scenarios where no autocorrelation patterns appear in residual structures, all analytical models performed well in recovering true factor loadings. Mean percentages of estimation bias ranged from -2.4% to 0.2%.

But for all designs with non-independent residual patterns, conventional CFA will always overestimate the factor loadings by over 20%. In the worse case of our results, where true $\lambda = 0.5$, $N = 900$, and the datasets had 2 autocorrelation patterns in the residual structure, the conventional CFA approach would, on average, overestimate the factor loadings by 26.0%. This consequence might suggest that, the portion of item score variances contributed by time-varying fluctuations have been incorrectly explained by factor loadings. Due to the strict local-independence assumption of CFA, factor loading terms absorbed the non-independent error terms in the regression process. Additionally, the principle of overestimation is independent of the sample size. This is reasonable as CFA does not specify any participant-level constrains.

Compared to conventional CFA, models allowing non-independent residual structures can efficiently reduce the mean estimation bias. However, underestimating the number of latent clusters is still likely to inflate factor loading estimations. In designs whose datasets have 2 and 3 real autocorrelation patterns, the AR-CFA consistently overestimated the factor loadings, especially when the factor structure is stable (λ =0.7). In this case, the mean bias from AR-CFA ranged from 13.8% to 14.3%. In contrast, overestimating the number of latent clusters seems to be less harmful to factor loading estimations: when there was only one autocorrelation pattern, mixture CFA with 2 clusters underestimated the true factor loadings, but such underestimation is trivial with the most severe mean bias of -2.1%; and when the datasets contain two autocorrelation patterns, misspecified models with three latent clusters would only yielded -2.1% of mean estimation bias even in the worst case. According to Figure 6.2, modelling multiple autocorrelation patterns has apparent advantages in approaching true factor loadings, as latent residual clusters extract the temporal components of item variances from the variance mixture.

6.6.3 Model selection criteria

Firgue 6.3 summarized means of entropy for mixture models. The entropy for most mixture models were small, falling under 0.7. These results suggest that the classification results cannot clearly separate individuals. Also, under the same design environments, the entropy decreases as the sample size becomes larger. These results were not surprising regarding our model specifications because we only considered two interpretive sources of variances, the effects of hypothetical construct and time-varying fluctuations, and the latter one only accounted for a vary small portion. Regarding the latent score distributions, different clusters have highly similar shapes that largely overlap. After fixing identical factor loadings across all clusters, it is difficult to distinguish clusters by controlling small (but significant) effects accounted for by temporal components. The overlapped areas of latent score distributions in between clusters become larger as sample sizes increase. Previous research has indicated that low entropy is typical for mixture models, and it even tends to decrease when the sample size is large (Fagan et al., 2013; Van Lissa et al., 2024).

Another finding from Figure 6.3 is that, in most cases, entropy increases as hypothetical cluster number becomes large, especially when true datasets have 2 clusters and analytical models estimated misspecified MAR-CFA$_3$, the entropy values are consistently larger than the true

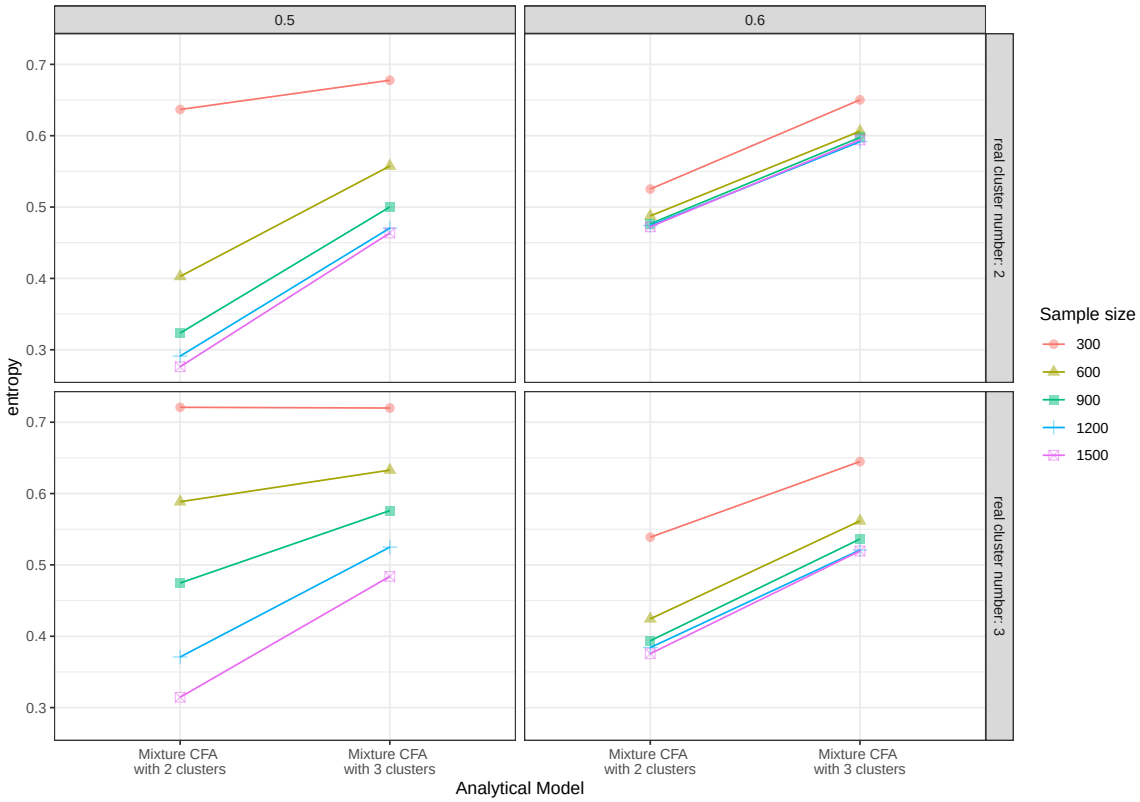

Figure 6.3: Entropy values for resulting models. Panels in each raw represents datasets generated from the same factor structure. Panels in each column represents datasets generated from the same true λ. Data from non-convergent models was omitted.

MAR-CFA$_2$. A possible inference is that mixture models are not able to clearly classify clusters that almost completely overlap. But for factor mixture models that only consider latent clusters in factor structures, entropy usually performs poorly in identifying correct cluster numbers (Lubke & Muthén, 2007), At this stage, an early conclusion is applying mixture models has limited capacity of capturing temporal heterogeneity across model-implied clusters, but it does benefit to capture the homogeneity in the sample (i.e., successfully recovering the true factor loadings).

From the entropy results, some potential for harm in model selection warrants notice. When analyzing empirical datasets, determining the number of clusters by simply reading entropy is not reliable, as true cluster numbers remain unknown. The the model selection should be a consequence of multiple fit indices. The results of the aBIC changes for model comparisons are presented in Figure 6.4. Overall the aBIC performs well in model selection. In the "baseline" scenario whose residuals are temporal-independent, all models yielded similar mean aBIC values, with the differences close to 0. When the true model has one autocorrelation pattern (i.e., a standard AR-CFA structure as in Ozkok et al., 2019), the aBIC criteria consistently lead to correct model identification results, with smaller aBIC observed in AR-CFA specification than in conventional CFA. The aBIC also prevent the overestimation of cluster numbers in this design. The mean aBIC changes higher than 0 are observed in this scenario, regardless of the true factor loadings and sample sizes, but these changes are very small in terms of the computed aBIC value ($0 < \Delta_{BIC} < 20$). Regarding datasets with more than one autocorrelation patterns,

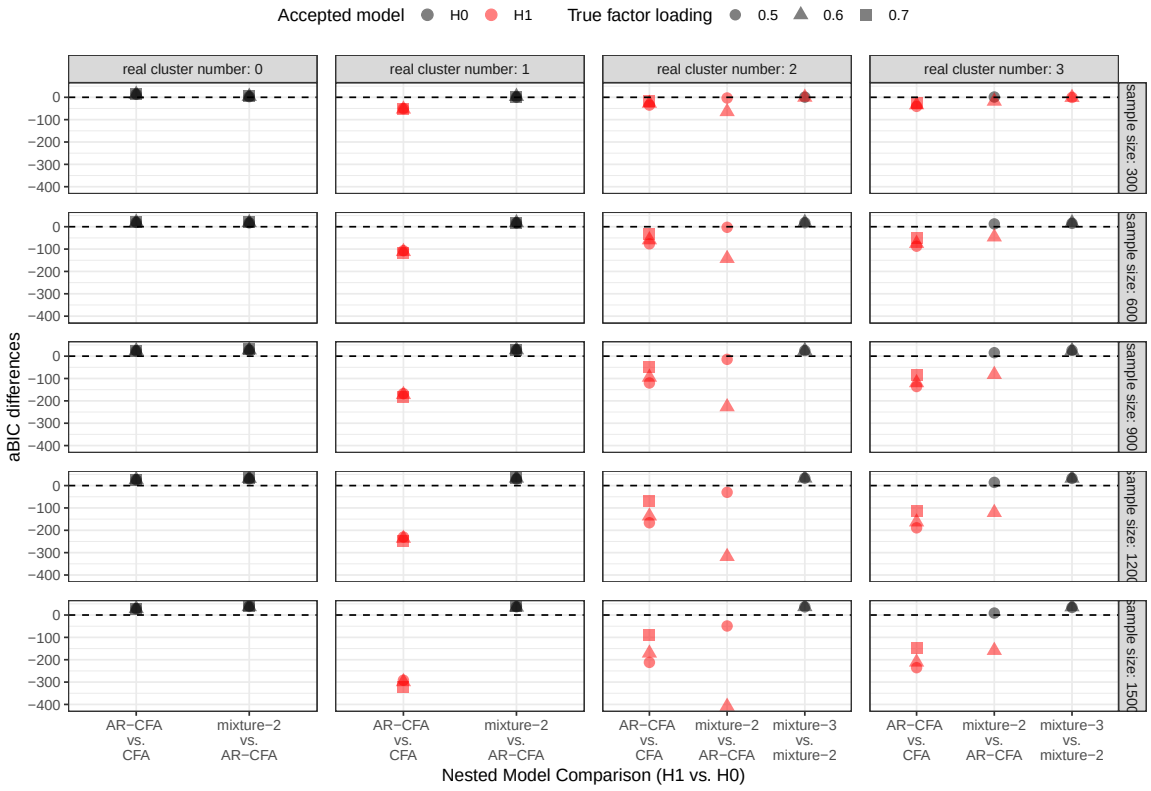

Figure 6.4: The aBIC changes in model comparison. Accepted model: H0 represents the model has one less cluster than the analytical model was preferable. Accepted model: H1 represents the current analytical model was preferable. Panels in each row represents datasets generated from the same sample size. Panels in each column represents datasets generated from the same factor structure. Data from non-convergent models was omitted.

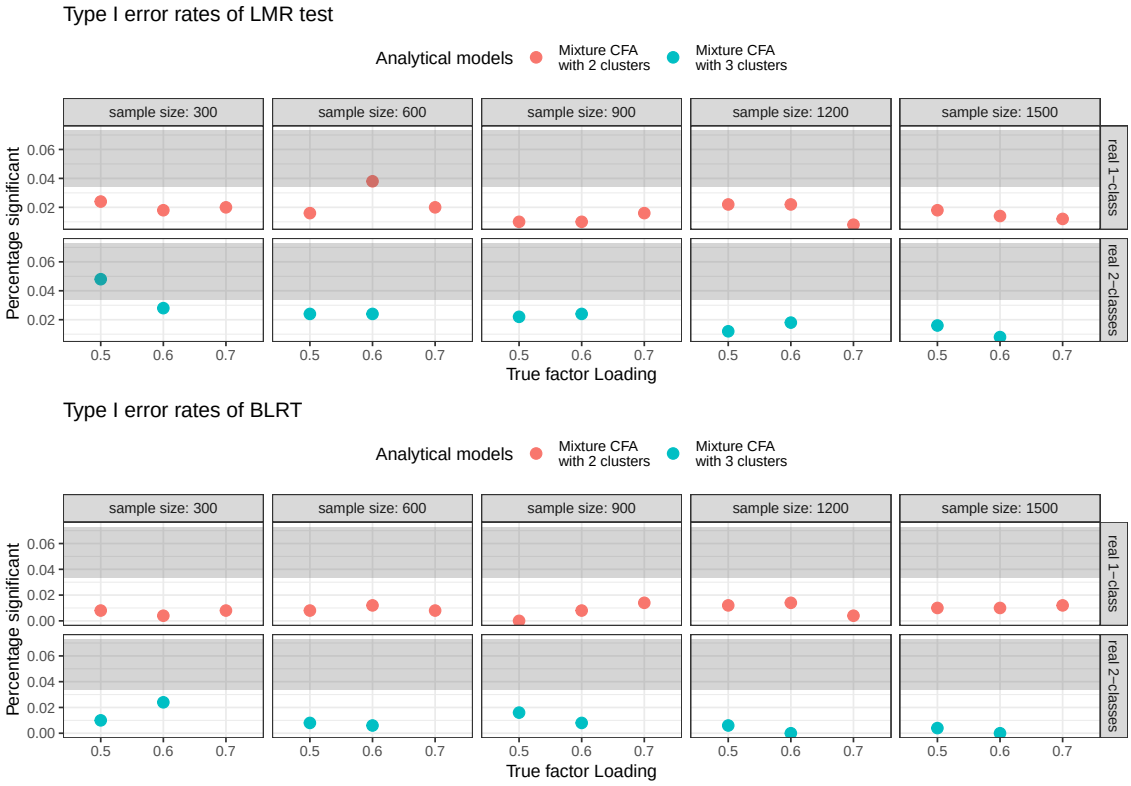

Figure 6.5: Type I error rates for LMR test and BLRT. Panels in each row represent datasets generated from the same factor structure. Panels in each column represent datasets generated from the same sample size. Data from non-convergent models was omitted.

aBIC in most cases can identify the correct mixture models with corresponding cluster sizes: all the mean aBICs can reject the AR-CFA (1-cluster model) in comparing that and 2-cluster mixture models, when the true cluster number is 2; most of the aBIC shown negative charges in 3-cluster models when the true models have 3 autocorrelation patterns. For designs with small sample sizes and an unstable factor structure (i.e., true factor loading = 0.5), model comparisons suggested very small changes in aBIC, which can be seen as results of two equally well-fitting models. But for datasets under a true 3-cluster models, small positive changes in aBIC consistently appeared. Considering the number of parameters estimated, the aBIC lost some capacities in model comparison.

The conclusion so far is: considering autocorrelations in CFA can improve measurement and yield a more realistic factor structure. But to what extent are we likely to determine a correct mixture model? To answer this question, some (nested) mixture model comparisons are further needed. The determination of latent cluster number is usually done by inspecting results from VLMR and BLRT tests. Results of these tests are presented in Figure 6.5. It is suggested that type I error rates for both MAR-CFA$_2$ and MAR-CFA$_3$ did not exceed the 95% confidence interval of the nominal alpha (grey shadowed area). These type I error rates mean in most situations, using conventional model comparison methods can correctly reject a misspecified model that overestimate the number of autocorrelation patterns.

However, extremely conservative Type I error rate raise concerns of low statistical power (i.e.,

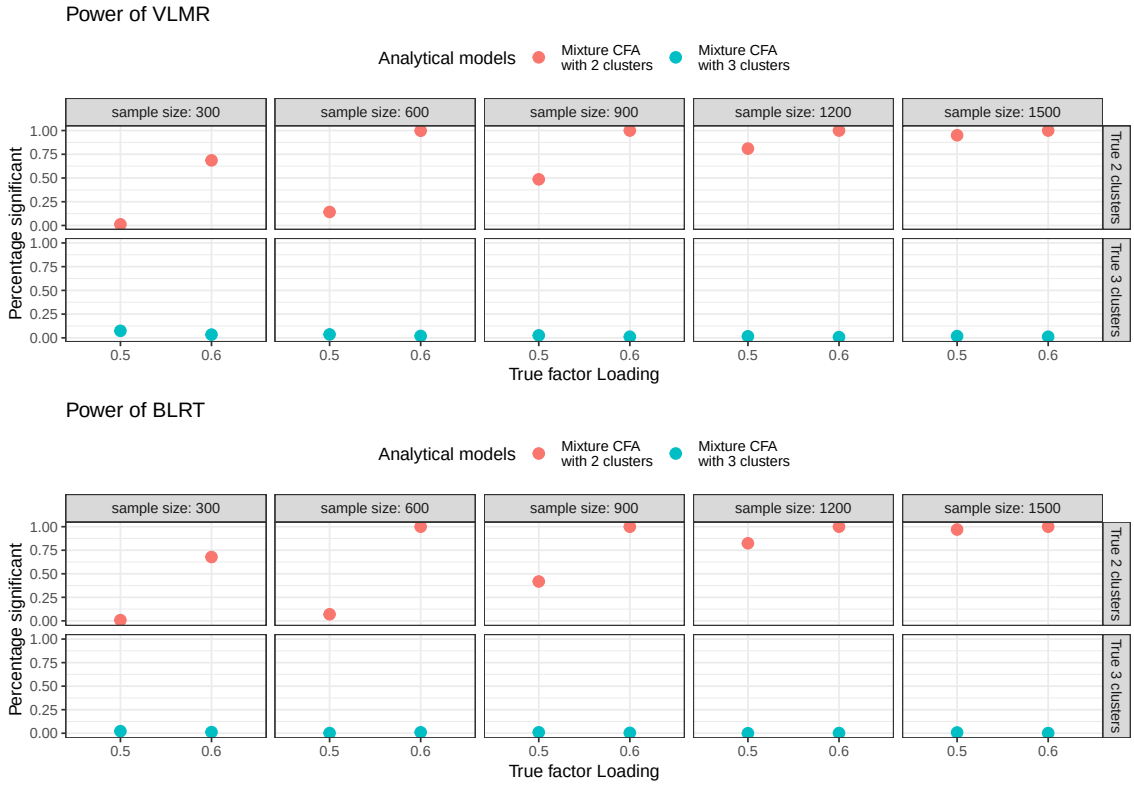

Figure 6.6: Power performance of LRT test and BLRT. Panels in each row represents datasets generated from the same factor structure. Panels in each column represents datasets generated from the same sample size. Data from non-convergent models was omitted.

inflated type II error rate). Figure 6.6 presents power performances of both model selection performances. For true 2-cluster models, the power gain is a function of sample size and factor loadings. With the "smallest" design, both LMR and BLRT achieved extremely low power; while the factor loadings increase and a stable factor structure is form, both tests had a probability over 50% to correct accept a true model. However, in 3-cluster models, both tests almost always prefer a simpler model with one less cluster. This is a symptom of homogeneity across classes in our datasets. Regarding our settings, to maximally avoid non-positive defined matrix, we hypothesized three ρs that are very close. When following a "standardized" procedure to generate data, small effects of autocorrelation would only result in distributions having different skewness. The power of VLMR and BLRT tests are hugely dependent on the sample size, number of latent clusters, and the degree of separation in between clusters. It is reported that these tests performs in a very conservative manner and tend to choose models with fewer classes when clusters have very small distance (i.e., a Cohen's d less than 0.2, Tein et al., 2013). We claim that, when applying mixture models to extract temporal components from the variances, multiple indices should be considered. Although looking at a single value might receive hints of the existence of time-varying fluctuations, it might not be informative enough to determine the correct cluster number.

To summarize our findings in model selection session, when considering time-varying fluctuations, multiple fit indices and tests should work in conjunction to detect a preferred mixture model. In most cases, using aBIC is helpful to directly compare models. All three model-

specified methods (entropy, LMR, BLRT) performed poorly in distinguishing latent clusters. But these results might be a signal that, although the effects of time-varying fluctuations are relatively small, they are still masking the true variances that are accounted for by the construct. Our results provided support to Ozkok et al. (2019), suggesting that considering time-dependent noise in psychometrics datasets could help recover the true factor loadings. Our results further indicates that, using clustering approach to divide time-dependent noise benefits to determine the factor structure.

6.7 Discussion

In this chapter we introduced a mixture model to capture time-varying fluctuations in cross-sectional psychometric and questionnaire datasets in order to improve psychological measurement. This mixture model extends the previous AR-CFA proposed by Ozkok et al. (2019) to the mixture framework, further separating time-varying fluctuation patterns within the sample. Based on our Monte Carlo simulation study, the mixture models performed well in recovering true factor loadings compared to the conventional CFA that totally ignored the temporal components in the dataset. By extracting those temporal effects from the variance components, mixture models avoided overestimating the factor loadings and yielded a factor structure that is closer to the true model. Furthermore, researchers are able to identify a model with adequate number of latent clusters by inspecting multiple model selection criteria.

To address the main issue of this chapter, it is inappropriate to ignore time-varying fluctuations in survey datasets. Regardless whether the mixture idea is applied, modelling time-varying fluctuations would considerably decrease the biases in estimating factor loadings. Under the framework of linear mixed effect regression, the effects of time-varying fluctuations are trivial, but they are magnified when researchers apply analysis techniques on top of aggregated data (i.e., mean scores and covariance matrix). Unlike behavioural or neuroscience experiments whose signals are directly measured, like reaction times of experimental trials, psychological measurement uses an indirect way to measure abstract concepts. Therefore, the psychological changes during measurement are likely to cloud measurement of the true construct in a more complex way. For example, time-varying fluctuations can take participants' responses in different directions-even though some items are less effective to measure extroversion, participants who has higher degrees of extroversion would consistently have higher scores to extroversion items and vice versa, because they adjust their responses based on the item context.

Indeed, applying conventional CFA and looking at the results, researchers are very likely to find out a portion of uniqueness to interpret the dataset, and claim the model-implied covariance matrix are able to discriminate participants with different levels of measured latent traits. But in empirical datasets, consequential uniqueness can be accounted for by the interaction between time-varying fluctuations and the measured psychological construct, or even by another construct that is not measured (Myszkowski & Storme, 2024). Incorrectly considering this uniqueness as the effect of the measured construct would exaggerate the capacity of the true construct, or say, the capacity of the questionnaire. This can be reflected by the overestimation

of inflated factor loading estimation in conventional CFA results. In terms of questionnaire development, biased factor loadings reduce the probability of yielding reproducible research, since in practice, the covariance structure included into analysis is a mixture of construct uniqueness and sample homogeneity, with the latter one very unlikely to be identical across samples. For empirical research or medical administrations, using a measurement with overestimated factor loadings would also overestimate participants' factor scores, and the locations of participants in the distribution of latent trait would be misjudged. It increases the chances of misdiagnosis in clinical settings, especially in the context of norm-references tests where factor scores would be compared to external cutoffs.

How should we extract time-varying fluctuations, especially when the true structure is completely unknown in practice? To answer this question, notice that our study is based on the assumption that time-varying fluctuations are not the effects of interest, but those needed to be controlled, as they are in experimental datasets. At this stage, our findings suggested that modelling time-varying fluctuations is advantageous to observing factor loadings that are closer to the true value, and, if possible, considering multiple common patterns across participant clusters would further eliminate the biases than simply including a "typical effects" across all participants (i.e., applying the AR-CFA). Due to individual differences, we are less likely to capture a common pattern of temporal change for each participant, but it is possible to divide them into a few subsets.

But now, a further question emerges as whether we should adapt our models after modelling time-varying noise? According to our model, traditional model modification approaches (i.e., those based on modification indices) do not help much in improving model fits, because causal relationships have been specified. In this case, any model modification seems to yield a more complex model, which brings up additional difficulties to estimation and interpretation. As confirmatory approaches, factor structure modelling techniques usually aim to determine whether theoretical psychological construct can describe the dataset at hand, and the factor loadings are key parameters researchers would care about. Therefore, we claim that, once time-varying fluctuations are controlled, researchers can safely interpret the "corrected" factor structure and treat time-varying fluctuations as nuisance effects that have been controlled.

The approach of fixing identical factor loadings across classes and allow residual structures to be freely estimated has some similarities to conducting a measurement invariance test—both methods control possible noise sources. However, the aim of our methods is to recover a factor structure that is closer to the truth and the trivial heterogeneity is usually not the effect of interest; the measurement invariance test focuses on the grouping variable that causes the changes of factor structure—the heterogeneity is the topic of interest. Based on this fact, we see the mixture model as a calibration method to improve measurement, and measurement invariance test a comparison method. We therefore encourage the combination of these two methods in the future research since they have potential advantages to observe more realistic factor structures across groups and lend more precise interpretation to the heterogeneity caused by the target grouping variable.

As an early exploration to improve measurement by controlling time-varying fluctuations, mixture models have the advantage of being conceptually understandable. However, one major challenge and limitation of our simulation study is from the modelling approach itself—specifically, the computation difficulty. In our simulation, analytical mixture models rarely converged when true factor loadings were high, the sample size was large, and when there were three latent clusters. We acknowledge that in practice, before selecting an appropriate model, researchers might have to pay even more time and effort to achieve "healthy models", say, those that convert to the global maximum. Modelling processes involve the selection and modification of starting values and convergence criteria, and researchers who are less familiar with mixture models would easily be lost in the forest of options. Without a clear guideline, analysts would have no better solution but simply increasing starting values and narrowing down the convergence criteria. This modification would possibly slow down the model estimation speed as new models usually take more iteration runs, and could be impractical on computers with limited capacity.

Another limitation is, our simulations generated "simple and perfect datasets" where autocorrelations are identical and significant across all nearby items, and autocorrelation patterns are identical within classes. In practice, however, such assumptions are still unrealistic. Instead, the autocorrelation might behave as a random effect, varying in strength and direction (e.g., reverse scoring items) across item series and across participants. Combining these situations might lead to increased difficulties in controlling temporal component in questionnaire datasets and reduce the prospect for meaningful interpretation of the resulting models. However, unlike time-series data modelling methods in experimental psychology, controlling time-varying fluctuations in cross-sectional psychometrics dataset is in a very early stage. Experiences and references in this field is limited, and further developing modelling methods to control time-varying fluctuations is helpful to demonstrate the role of psychological fluctuations in item response process.

Now we know that modeling time-varying fluctuations could improve measurement in psychometrics, but they are quite challenging to implement. As such, could we turn to some design-based strategies to control time-dependent noise, as we have done in experimental designs? In next chapter, I would address this question by proposing some potential approaches based on questionnaire design, and provide a tool for investigations in this field.

Chapter 7

RandomiSur: A Platform for Randomizing and Counterbalancing Psychometric Measurement

7.1 Background

In the previous chapter, we have seen how a model-based strategy can increase validity and improve measurement in psychometric surveys. However, for researchers who lack the experience of (advanced) modelling, MAR-CFA would be quite challenging due to its model complexity and computational burden. And in most cases, model-based approaches would also require specialized software and only be applicable to certain types of data. But the success of a design-based strategy in cognitive experiments (e.g., the PSR algorithm in previous chapters) brings hope that a way can be found to extend the strategy to the psychometric context. If randomization can help control effects of time-varying fluctuations in laboratory tasks, perhaps it can bring similar improvements to questionnaire administration. Answering the question of whether design-based approaches can improve measurement to psychometrics is beyond the scope of this thesis. Instead, we describe a questionnaire platform that we have developed to allow researchers to flexibly randomize and counterbalance item orders when deploying surveys.

The vast majority of survey instruments present test items in fixed presentation orders. In the past, this was reasonable as surveys were often done with paper and pencil, and individual sequence manipulation would require researchers to have hard copies with different presentation orders. But now that online survey tools are available along with computer algorithms for randomization and counterbalancing, in principle, researchers have more flexibility in survey construction.

Unfortunately, how item order impairs validity is still under studied, and the small amount of work on this topic does not yield consistent conclusions. Schell & Oswald (2013) investigated different randomization approaches for the Big Five personality survey and concluded that its psychometric properties were independent of item order. Using survs, an online survey

platform, researchers utilized three order arrangements (putting items under the same factor together, cycling items under different factors, one-off randomization) in the survey and did not find significant improvement in CFA modelling. A contribution of Schell & Oswald (2013) is the idea of "planned" design in scale design for improving measurement. Specifically, putting items under the same factor together bears some similarities to the block design in experimental context, while "cycling" that keeps presenting items from distinct factors and it is therefore similar to our PSR_{max}. However, all three sequencing methods were conducted on the sample level, which means participants under the same sequencing method still complete the test with the same orders.

But as we have seen in the experimental context, design-based strategies improve measurement by controlling time-varying fluctuations *within individual participants*, which is important because temporal variations are likely to be idiosyncratic. In contrast, the planned designs in Schell & Oswald (2013) are no different from generating three new fixed orders and delivering them to participants, and thus this approach failed to provide an adequate test of the potential benefits of randomization. A more recent study conducted a fully randomized psychological measurement to investigate measurement quality against fixed-order measurement Şahin (2021). Results suggested fixed item order led to biases in mean scores and factor structure, while randomization increased the construct validity, indicated by better model fit. But as with the randomization process in experiments, if an item order is subject to simple restricted randomization without any further constraints, it is likely that we would observe runs of items measuring the same underlying factor. Grouping items from the same factor has been shown to overestimate the scale validity, giving rise to short-term item effects and anchoring (Gehlbach & Barge, 2012; Weijters et al., 2014).

To avoid unplanned runs in item presentation, transferring design-based strategies from the experimental context to the psychometric context would be helpful. To do so, using online survey platforms might benefit to efficiently randomize or counterbalance our surveys. Anwyl-Irvine et al. (2020) have summarized the most popular online survey platforms, which come with built-in functions and user-friendly configurations to efficiently generate online psychological experiments and surveys. However, they only provide limited support for item order management. Most platforms and questionnaire builders listed in Anwyl-Irvine et al. (2020) like Qualtrics, PsyToolKit, and SurveyMonkey, have built-in functions to fully randomize presentation orders of uploaded questionnaires, and Gorilla additionally supports Latin Square design that exports each row of a Latin Square as an individual presentation order. Maybe, these platforms provided randomization and counterbalance options because they want to support researchers to better design surveys; and indeed, some studies have utilized these functionalities to discuss whether simple restricted randomization is useful to improve measurements (e.g., Buchanan et al., 2018; Loiacono & Wilson, 2020). However, if we want to develop and deploy better design-based methods with online platforms, just as we did in experimental context with `explan`, there is little flexibility for us to do. This is because, firstly, they do not offer much opportunity for further control sequences beyond simple restricted randomization and Latin

Square. Second, even if researchers want to customize the ordering functions, the closed-source nature of online platforms often does not allow them to do so.

But researchers dissatisfied with existing platforms who wish to build their own custom platforms would face a large set of programming challenges. To develop a standalone online platform requires familiarity with website programming languages (e.g., HTML, JavaScript, PHP), and even database management skills (e.g., MySQL). Such programming abilities are likely to exceed the knowledge of many psychologists and data analysts.

7.2 RandomiSur

To overcome the shortcomings of existing platforms and help researchers better design their surveys, we introduce our newly developed computer-based design builder and online survey platform, RandomiSur. RandomiSur is an open-source questionnaire builder and data collection platform and can be retrieved from https://github.com/Jinghui-Liang/RandomiSur. Driven by jsPsych, a `JavaScript` framework which is supported by major web browsers (e.g., Chrome, Firefox, Safari), RandomiSur can set up items directly from a source file. Additionally, RandomiSur has comprehensive default settings that allow users to build a web-based online survey with minimal programming, and no additional browser dependency or plug-in is needed.

One of the key advantages of RandomiSur is that it gives researchers great flexibility to manipulate item orders. It includes eleven built-in algorithms to design surveys, including three fixed-order-based methods and eight randomization strategies. Researchers can use all possible combinations to generate item sequences customized for each participant, and all randomization strategies are at the individual level rather than the group level.

One of the differences between RandomiSur and the other platforms is that it uses a sequential presentation, with only one item is presented per page, instead of showing all items on the same page. This sequential presentation order prevents respondents from jumping backward and forward in the survey. In addition, a default timing function is implemented that enables researchers to observe item-level response times in addition to collecting response measurements. Response time at the item level has received strong interest among survey researchers and has become recognized as an informative factor illuminating individual differences (Bean & Bowen, 2021; De Boeck & Jeon, 2019; Liu & Liu, 2021).

All these properties make RandomiSur a powerful tool to design and conduct professional online surveys. Although RandomiSur was originally developed in Linux, all dependencies have been containerized into a Docker image (a lightweight virtual software environment that promotes easy and consistent deployment across machines with different operating systems). Researchers can access the full functionalities of RandomiSur via any operating system that supports Docker. In production, data can be stored in a hosted database by simply delivering source files to the host. In what follows, we will describe an example of how to build and deploy a questionnaire with RandomiSur, together with a detailed introduction to its features.

7.3 Applied example: Deploying an online personality survey with RandomiSur

Suppose you wish to deploy an online survey aiming to measure Big-Five personality by the 20 items sampled from the 50-item International Personality Item Pool (Goldberg, 1992; IPIP-20 henceforth, Goldberg et al., 2006), together with two demographic items (age, gender). The IPIP-20 contains 20 items from the original 50-item scale, measuring the Big-Five factor markers reported in Goldberg (1992), including Extroversion (E), Agreeableness (A), Conscientiousness (C), Neuroticism (N). and Openness to experience/Intellect (O). Each factor has four items.

7.3.1 Installation

Since RandomiSur is containerized with Docker, before downloading RandomiSur, users should first install Docker. After Docker is installed, on the local machine, users can access RandomiSur via `https://github.com/Jinghui-Liang/RandomiSur.git`, by downloading a compressed archive from the web page, or executing the following command line on the shell prompt locally, if Git is locally installed.

```
git clone https://github.com/Jinghui-Liang/RandomiSurgit
```

7.3.2 Prepare the test

Next, the IPIP-20 should be prepared and saved as comma separate values (csv) format. RandomiSur has specific requirements for column arrangements and their contents. A template for IPIP-20 is presented on Table 7.1.

question	choices	required	label	demographic
What is your age?		n	age	y
What is your gender	male/female	n	gender	y
I am the life of the party.	Inaccurate/ Neutral/ Accurate	y	E	
I feel little concern for others.		n	A	
I am always prepared.		y	C	
I get stressed out easily.			N	
I have a rich vocabulary.			O	

Table 7.1: A simplified template of questionnaire format

According to Table 7.1, a standard questionnaire should contain five mandatory columns. In the first column, **question**, users should place their scale items and demographic items, with each row containing a single item. The second column, **choices**, should contain the set of available choices for each item. To generate Likert-type choices, users should separate options with slashes. For example, "Inaccurate/Neutral/Accurate" would be treated as three possible options for an item. For scale items, users can provide choices for the first item and leave the column blank for all remaining rows. By doing so, RandomiSur will map this choice to the rest

of the items unless otherwise specified. For demographic items, however, blank cells mean text input will be accepted instead of choices. The third column, **required**, specifies whether or not an item must be answered. Both letter "y" in a cell or a blank cell mean that answering this item is mandatory, and participant will be prevented seeing the next item until they provide a response. The user can put "n" in this cell to disable the forced-answer function. The fourth column, **label**, contains labels for every item. For demographic items, labels presented in this column will be transferred to the name of a variable storing corresponding information. For scale items, labels represent which factor (subscale) a given item belongs to, and will be used as an important index for arranging items. The fifth column, **demographic**, tells RandomiSur whether or not the item is a demographic item. Typing "y" in cells of this columns makes the corresponding item on the same row a demographic item, while typing "n" or leaving this cell blank takes the corresponding item as a questionnaire item. RandomiSur behaves differently in handling demographic items and questionnaire items.

7.3.3 Item sequence arrangement

In RandomiSur, item sequences of a questionnaire are pre-defined and stored as a property of surveys, and they are easy to build with our built-in shiny app, `inject-order.R`. Executing the R code below will take users to the test builder page:

```
## NB: install "shiny" first using:
## install.package("shiny")

shiny::runApp("./RandomiSur/inject-order")
```

On the sidebar of `inject-order`, users can upload `ipip20.csv` to the shiny app and have a preview on "data" panel. To start arranging data, users can access the "builder" panel and randomize the questionnaire order by selecting desired strategies on the left sidebar, together with setting sample sizes for each selected method Currently, RandomiSur has eleven possible algorithms to arrange presentation orders. Table 7.2 details their mechanisms.

Methods	Label	Description
Fixed	fx	No sequence manipulation.
Simple Randomization	rd	Randomize the sequence order without constrains.
Permuted-Subblock Randomization (Maximal)	psrmax	Implementing PSR to sequences with maximal Nr, factor as conditions.
Latin Square	ls	Generate a $k \times k$ Latin Square, k equals to number of items. Each row is seen as an item sequence and presented to participants.
Fixed Grouping	gff	Clustering items under the same factors, fix item sequences, fix factor sequences. E.g., EEEECCCCAAAANNNNOOOO
Random Grouping, factor	grf	Similar to gff, but randomizing factor sequences.
Random Grouping, item	gfr	Similar to gff, but randomizing item sequences within groups.
Random Grouping, both	grr	Clustering items under the same factors, randomize item and factor sequences.
Fixed Cycling	cff	Cycling items based on a fixed factor cycle order, fixed item allocations in each cycle. E.g., EACNOEACNO...
Random Cycling, factor	crf	Cycling items, but randomizing factor cycle orders for each participant, within each cycle, fix item allocations.
Random Cycling, item	cfr	Cycling items, fix factor cycle order across participant, but randomize item allocations within cycles for each participant.
Random Cycling, both	crr	Cycling items, randomize both factor cycle orders and item allocations within cycles.

Table 7.2: Randomization methods in RandomiSur

These methods contains common ways to arrange item sequences. For example, the "Fixed" algorithm presents items as they are originally listed in the questionnaire; "Simple Randomization" conducts full randomization to the item order with no additional constrain; and "Latin Square" appears as a counterbalancing method to cancel possible order effects over participants. We also included two algorithms, fixed grouping and fixed cycling, that were mainly discussed in previous research (Schell & Oswald, 2013), and their randomization versions: random grouping and random cycling. For grouping algorithms, items under the same factor are placed together. Participants do not receive items from other factors until they have responded to items for a given factor. In the cycling method, participants repeatedly respond to items from different

factors under a test, but no two items from the same factor would sequentially appear. Both fixed grouping and fixed cycling would not randomize item positions in the original sequence, nor would they shuffle the presentation order of factors. If we use E_1, E_2, E_3, A_1, A_2, A_3, C_1, C_2, C_3 and so on to represent the i_{th} item under a given factor, then the fixed grouping order would be $E_1-E_2-E_3-A_1-A_2-A_3-C_1-C_2-C_3-N_1-N_2-N_3-O_1-O_2-O_3$, while the fixed cycling order would be $E_1-A_1-C_1-N_1-O_1-E_2-A_2-C_2-N_2-O_2-E_3-A_3-C_3-N_3-O_3$. On top of these principles, randomization on the factor level (Random Grouping, Factor and Random Cycling, Factor) would randomize the factor presentation so that factors do not always appear in the same order, e.g., $EACNO$, but with items still appearing from the first to the ith factor. In contrast, randomization on item level (Random Grouping, Item and Random Cycling, Item) randomizes the item orders but keeps factor presentation the same for every participant. Combining factor-level and item-level methods yields the comprehensive (quasi)randomization variations of grouping and cycling methods. That is, both factor order and item order would be independently randomized and different across participants. An item sequence under the *Random Grouping, Both* method could be $O_3-O_1-O_2-A_2-A_1-A_3-N_1-N_2-N_3-E_1-E_3-E_2-C_1-C_2-C_3$, and one under *Random Cycling, Both* could be $A_2-C_1-E_3-N_1-O_2-A_3-C_2-E_2-N_3-O_1-A_1-C_3-E_1-N_2-O_3$. All algorithms can be used in conjunction in order to compare capacities of sequence arrangement strategies.

Once users finish deploying desired randomization strategies, they can visit the "plan" panel to inspect names of generated orders, and determine how many participants each order is to be assigned to. After the inspection, users can download the compressed "data documents" by clicking "download" button on the top of this shiny app. A compressed file, `df-order.zip` would be downloaded and users should move and store this file at **scalepool** directory. Decompressing is not needed since the scripts later on will automatically do so for users.

7.3.4 Launch an online survey

Now we have all necessary background to deploy RandomiSur. To initialize it, users need to launch the terminal on their own machines, and change the working directory to where RandomiSur is stored.

```
cd path/to/RandomiSur
```

Then, users need to create the Docker image to build up a virtual development environment by the command below.

```
docker-compose up -d --build
```

```
...
Creating survey_php ... done
Creating survey_db  ... done
```

The first time the user executes this command, Docker downloads all necessary dependencies and builds up the environment for RandomiSur. This might take a while. Once the initialization

is finished, users should be able to access the the environment by the code below.

```
docker exec -it -w /var survey_php sh
```

```
...
#
```

The first configuration step is to set up the database. To do so, users should execute the binary `ConfigDB` by putting the command below.

```
./ConfigDB
```

Then, users will see a prompt asking if they are running a local test or uploading the questionnaire to a survey.

```
Are you running local test or uploading your platform to a server? (local/server)
```

In our example, we will put `local` as the first argument to generate a sample survey. After hitting return, users will see the second prompt on the terminal.

```
The name of target questionnaire you would like to use, extension required:
```

At this point, we use the filename of our questionnaire, `ipip20.csv`, as the argument to tell RandomiSur which is the target questionnaire and allow it to automatically generate the database, create a connection from the survey to the database, and write randomization rules based on the target questionnaire and item sequences generated by `inject-order`. When the prompt 'Initialization done' is displayed on the terminal, users can inspect the resulting online survey by accessing 127.0.0.1:8080 from a web browser. In this generated online survey, items will be presented on the webpage one by one, and participants will not know the presentation orders they received. Demographic items will always appear at the beginning of surveys, and randomization would not be applied to these items. As they finish the survey and click the "submit" button, which is displayed last, their responses and trial-level RT will be submitted and stored in the (virtual) database.

Presentation orders will be automatically and randomly assigned to participants. Once a presentation order reaches the maximal assignment frequency as planned, this order will not be selected. After all sequences have been assigned as many times as planned, the entire survey ends, and new visitors to the survey webpage will not be able to see any questions. By default, RandomiSur conducts anonymous surveys and participants will only be assigned random IDs only for the purpose of data tidying.

7.3.5 Accessing the dataset and data structure

Users can access the dataset anytime, even when a survey is not finished, by calling the built-in R script `download_rawdat.R` from the command line.

```
Rscript path/to/RandomiSur/R/download_rawdat.R
```

Executing this script from the terminal allows reading the environment file on the root directory of this platform (detailed below). Successful execution downloads three `csv` files on the local directory: a `response.csv` file which is the main response dataset for every participant, a `demo.csv` that records the demographic information for each participant, and an `order-pid.csv` that records which participant received which presentation order. Table 7.3 presented a template of pre-processed data from `response.csv`.

p_id	rt	response	Q_num	trial_index	order_index
"gso0"	1065	2	1	2	"cff"
"gso0"	1147	3	2	3	"cff"
"gso0"	1015	1	3	4	"cff"
"gso0"	948	2	4	5	"cff"
"gso0"	931	2	5	6	"cff"

Table 7.3: A simplified data template generated by RandomiSur.

The resulting dataset presented on Table 7.3 has six columns: (1) **p_id** represents the random, anonymous id assigned to a participant; (2) **rt** represents the participant's reaction time for this item; (3) **response** shows the participant's response to an item; (4) **Q_num** represents which item it is in the original item sequence; (5) textbf{trial_index indicates this item's position in the arranged sequence; and (6) textbforder_index indicates which presentation order was assigned to this participant. Looking at the the first row of this dataset, we can obtain the following information: the participant "gso0" received an item sequence under the Fixed Cycling (cff) manipulation method, and in this sequence, the first item presented to this participant was the seventh item in the original questionnaire, "I am interested in people". This participant chose the second option for this item, "Moderately Inaccurate", and the reaction time of this item is 1,065ms.

7.3.6 Hosting a formal online survey

To launch an online survey, users can change the (hidden) configuration file located on the root directory, i.e., `/RandomiSur/.env`. First, edit the host names, port, usernames, database name, and password to the corresponding information for the server. Then, users should access the virtual environment and execute the `ConfigDB` binary again, with `server` as the argument of the first prompt. By doing so, RandomiSur will automatically configure the online database according to the configuration on `.env`, and initialize the remote online database. The default setting of this Docker container enables files within the virtual environment to be accessed from outside (i.e., users' local machines), so that they can send the resulting public html folder `/RandomiSur/server/www` to the server in order to put the survey online. Dataset inspection follows the same procedure illustrated above.

7.4 Discussion

This chapter outlined a new platform, RandomiSur, for conducting online surveys and manipulating item sequences. RandomiSur takes advantage of multiple programming languages and

serves as a powerful test builder and online survey platform. It also minimizes the deployment difficulties through containerization and web-browser access, allowing researchers to build up and test an online survey with little programming. Most importantly, RandomiSur provides a large array of possibilities for arranging item sequences. Hence, researchers have more flexibility to explore design-based strategies in order to control time-varying fluctuations and improve measurement,

Compared to model-based approaches, using design-based strategies to improve measurement quality is more accessible and easier to implement, especially in the area of psychological measurement, were basic factor models are already fairly difficult to estimate even without introducing time-based components. That said, how much could controlling the presentation sequence improve the quality of measurement? At this point we are unable to draw any conclusion. In fact, according to the existing literature, the benefits of randomization are debatable. It is still unclear whether the benefits to controlling order arrangement in psychometrics are anywhere near those that we have demonstrated in the experimental context. By introducing RandomiSur, we have at least provided a tool that makes it possible to begin addressing these questions empirically. RandomiSur also offers new item sequence arrangements that could stimulate future research in psychometrics. Given its open-source nature, researchers can generate new R functions for ordering items and add them into the R shiny app. As a starting point, we are confident that a developed test builder and data collection platform can help researchers gain a further understanding of temporal characteristics in psychometrics.

In psychometrics research, it is very common to investigate relationships across multiple latent traits, and researchers need to conduct a joint measurement that uses multiple questionnaires. An interesting question is whether design-based methods help to clean up time-varying fluctuations in such datasets and improve measurement. If so, how might these methods work in such a complex scenario? A future development goal, would be to fit RandomiSur into general application scenarios, such as allowing simultaneously arranging presentation orders of multiple questionnaires and independently changing their the presentation orders.

Chapter 8

General Discussion

8.1 Summary of key findings

In Chapter 1, we discussed how time-varying fluctuations impair measurement and lead to reduced power/validity; we also suggested the possibilities to tap into time-varying fluctuations as a potential source of better measurement. In our first methodological part, we introduced Permuted Subblock Randomization (PSR), an algorithm to boost power by better randomizing trials in one-factor experimental designs. Then we compared PSR to a model-based approach, Generalized Additive Mixed Models (GAMMs), to compare power gains from design-based approaches to those from model-based approaches. Results suggested that the design-based approach yielded gains often comparable to model-based approaches. But the approaches are not in conflict, and combining them can optimize power even more in certain situations. At the end of this part, we extend PSR to 2x2 factorial designs, and introduced two variations, PSR-C and PSR-E. Our simulation results confirmed that the PSR algorithm can boost power in the factorial context. In this case, traditional PSR achieved general power increase; while PSR-C and PSR-E allowed for power gains for the interaction effect and column effect, respectively.

In the second part of this thesis, we introduced a model-based method, mixture autoregressive confirmatory factor analysis (MAR-CFA), to improve the validity in psychometrics research by clustering latent heterogeneity. This model-based approach showed that modelling time-varying fluctuations in psychometrics datasets helps to recover the true factor loadings and thereby increases validity. But deploying MAR-CFA is challenging because it imposes a computational burden and involves complicated model selection. Finally, we introduced RandomiSur, an online test builder and data collection platform, to provide a tool for researchers to randomize and counterbalance their surveys. This platform offers the possibility for future studies to use design-based approaches to increase psychometric studies validity, which is easier than using model-based approaches.

The key finding in this thesis is, time-varying fluctuations can be—and should be—used as a source to improve measurement. By dealing with such fluctuations with design-based and model-based approaches, or even by combining them, the statistical power and validity of

psychological research can be improved. Improving measurement may help address the long-standing reproducibility crisis in psychology (Pashler & Wagenmakers, 2012).

8.2 Experimental studies

In Chapter 3, even under the simple one-factor design, variations on the PSR approach are possible, and should be explored in future work. Although our simulations considered a situation in which every subblock contained the same number of repetitions, this need not be the case. Indeed, an appealing way to balance power gains against predictability would be to vary subblock sizes throughout the experiment. The general principle, which can be extended beyond the exponential decay scenario, is to use fewer repetitions per subblock at times where temporal effects are strongest, and more repetitions per subblock when temporal effects are weakest. For instance, in situations where learning effects with a characteristic "exponential decay" pattern are expected, one could use an *expanding subblock* strategy, where the sequence begins with a subblock with one repetition per level, followed by a subblock with two repetitions per level, followed by a subblock with three repetitions per level, and then a final subblock with all remaining repetitions. Also, learning effects can occur after each break during a session, as Thul et al. (2021) demonstrated in their reanalysis of a large Stroop dataset. Thus, to improve power, it would be advisable to use a subblock with one repetition of each level following each break, and then use PSR with a higher number of repetitions throughout the remainder of the session.

Given our results in Chapter 4, PSR could be used as a control method to deal with time-dependent error structures in datasets. Even when temporally structured residuals are already modelled by GAMMs, PSR could still add extra power gains by cleaning up the remaining time-dependent components while maintaining reasonable false positive rates. But the performance of PSR and GAMMs together depends strongly on sample sizes and error structures. According to our results, when the sample size is small and a strong, obvious time-varying fluctuations pattern is expected, using PSR on the basics of GAMMs could bring maximal power advantages. In more general situations where time-varying fluctuations are mixed in different time scales (e.g., as the simulated mixed error structure), using PSR on the field-standard analysis can achieve similar power gains as using so on GAMMs. For researchers who are not familiar to to GAMMs, PSR can be confidently applied to eliminate time-dependent noise, without changing the analysis they are familiar with.

In Chapter 5, we extended PSR to 2x2 factorial designs. But it also seems possible to extend the approaches here to designs that are even more complex than 2x2, such as 2x3 or even 2x2x2 designs. However, we acknowledge that PSR-C and PSR-E would be difficult to deploy under such cases. As the simplest extension, even adding one factor level to generate a fully within-subject 2x3 factorial design, there would be 15 possible paths to move across all conditions within a subblock. Representing all these variations would require a large sample size in order to counterbalance the sequence order if condition positions are always fixed. However, researchers can always consider PSR_{max} in complex designs given its general power advantages.

According to Chapters 3 to 5 in part I, by discretizing temporal characteristics, our PSR algorithms achieved power improvements in experiments. Our approach bears some similarities to the m-sequence approach for improving the efficiency of event-related functional magnetic resonance imaging (Buračas & Boynton, 2002), but these two design-based approaches to improving power have distinct aims. M-sequences aim to optimize efficiency of estimation in the face of strong carryover effects arising from the sluggish nature of the hemodynamic signal, but are otherwise temporally invariant. To counteract these sequential dependencies, quasirandom m-sequences are mean to minimize their own temporal autocorrelation; i.e., that strive to be as orthogonal as possible to time-shifted versions of themselves. In contrast, our approach is intended to deal with measurements that are assumed to be fluctuating in time. M-sequences are less concerned about repetition of labels (event types in fMRI nomenclature) and more concerned about counterbalancing their ordering. In contrast, our approach seeks to avoid repetition of labels and does not concern itself with higher-order properties (e.g., bigram or trigram probabilities). It may be profitable to further investigate the relationship between these two approaches in future work.

Our claims about PSR are based on the outcome of Monte Carlo simulation, but there are two main challenges with this approach: (1) determining whether the parameter space is representative of real world data; (2) ensuring important edge cases have not been missed. As to (1), we have attempted to simulate data with realistic variance components and that reflect representative designs in the experimental literature. We have also sought to use time-varying structures that are theoretically and empirically justified. There is abundant evidence in the literature for the existence and pervasiveness of time-varying patterns and repeated warnings about their potential impact on analysis (Altman & Royston, 1988; Amon & Holden, 2021; Baayen et al., 2017). However, research is needed to develop tools for diagnosing these patterns in real data, as well as efforts to catalog the functional forms and how much variance is attributable to temporal structure. Research is also needed to characterize the timescale of these various effects. Until such research is available, the true benefits of design-based (or even model-based) strategies for accounting for temporal structure will be difficult to estimate.

8.3 Psychological measurements

It is well-known that time-varying fluctuations exist in psychological measurement datasets. Some have suggested they need not be controlled (Shimada & Katahira, 2023), but then constructs capture temporal noise rather than true systematic patterns and thereby reduce the validity of the measurement (Myszkowski & Storme, 2024). Finally, if the local independence assumption is violated because of autoregressive residuals, analyses to estimate item capacity (e.g., difficulty parameter and discrimination parameter) or human parameters (e.g., ability parameter) would be warped (Tang et al., 2020). In Chapter 6, our findings with MAR-CFA dispute the conclusion that time-dependent noise can be ignored. We found that considering time-dependent noise in psychometric datasets can help recover true factor loadings and increase validity.

But, as mentioned earlier in this thesis, MAR-CFA might not be the best approach to deal with time-varying fluctuations in questionnaire datasets, or not the best possible model-based approach at least. If the autoregression pattern is fixed, then time-dependent variances will decay, and eventually account for small total variances. Especially when scores are standardized, identifying different autoregression patterns amounts to distinguishing distributions that are highly overlapping. On top of this, successfully fitting a mixture model requires complex model selection criteria, alongside the extra computational cost of comparing models based on these criteria. In addition, unequal time-varying effect sizes, missing data, or even requiring more complex models to deal with special item settings (e.g., three-point scales where responses are discrete versus five-point scales where responses can be seen as continuous) bring many challenges to using MAR-CFA, or even most of psychometric analytical approaches that consider time-varying fluctuations (Asparouhov et al., 2018, 2023). But given the powerful insight and high validity improvement model-based approaches could bring, we encourage future developments in this field.

So, with model-based methods in their early stage and challenging to use, how well might design-based methods improve measurement and validity in survey datasets? Answering this question needs some time, because there was formerly no tool for using design-based methods in psychometric surveys. By introducing the tool RandomiSur (Chapter 7), we introduce the possibility of controlling time-varying fluctuations through the design, as the package `explan` does for experimental datasets. But design-based approaches may be more challenging to develop in the psychometric context, given that sequences involving sensitive items or reversed-scoring items could induce unexpected adaptations or temporal effects (Myszkowski & Storme, 2024). In addition, some order arrangement algorithms (e.g., grouping, cycling) seem to increase validity by intentionally inducing "blocking effects" in each subset of items. In real questionnaire surveys where true validity is impossible to determine, it is unclear whether the apparent "improvement" is a statistical artifact or a true gain from organized item sequences. It is interesting to consider that when researchers develop a new measurement, they determine its validity using a fixed item order. Therefore, from some researchers' perspectives, this "factory setting" sequence is seen as a property of the measurement itself (Schell & Oswald, 2013). Following this opinion, would questionnaire validation with independently and properly randomized sequences yielding measurements with higher quality, as it is more "sample-free" and more aligned to local dependence assumptions? RandomiSur makes it possible to address these and further questions.

8.4 Closing remarks

In closing, let us come back to the main theme of this thesis: measurement. Although studies in this thesis are still in early stage, they provided an encouraging prospect and an exciting opportunity for gaining better measurement quality. As mentioned in Chapter 1, once we acknowledge that human beings adapt and change over time during an experiment or investigation, there is tremendous potential to improve the quality of measurement.

Indeed, there is much to be done to optimize measurements in the future. One can focus on extending design-based and model-based methods to scenarios that are close to real psychological research environments. To name a few, participants could take rests in between experimental sessions; task-takers can complete a combination survey which is combined by many questionnaires, or even an integrated task alternatively involving experiments and surveys. These settings could yield a number of fluctuation patterns that are more complicated than our cases. However, a major contribution of this thesis is to point out the possibility to use such fluctuations as the source of better measurements. Given that our design-based and model-based methods are developed on the most general situations, we are optimistic that these approaches can be widely generalized to many other psychological studies. By further developing methods to improve measurements under these scenes, the efficiency of psychological studies can be hugely benefited by removing the masking caused by time-varying fluctuations.

Bibliography

Altman, D. G., & Bland, J. M. (1999). How to randomise. *Bmj (Clinical Research Ed.)*, *319*, 703–704. https://doi.org/10.1136/bmj.319.7211.703

Altman, D. G., & Royston, J. P. (1988). The hidden effect of time. *Statistics in Medicine*, *7*, 629–637.

Amon, M. J., & Holden, J. G. (2021). The mismatch of intrinsic fluctuations and the static assumptions of linear statistics. *Review of Philosophy and Psychology*, *12*, 149–173.

Anwyl-Irvine, A. L., Massonnié, J., Flitton, A., Kirkham, N., & Evershed, J. K. (2020). Gorilla in our midst: An online behavioral experiment builder. *Behavior Research Methods*, *52*(1), 388–407. https://doi.org/10.3758/s13428-019-01237-x

Asaoka, R., & Wada, Y. (2023). Mechanism of the compression effect on visual duration perception caused by temporally sandwiching sounds. *Journal of Experimental Psychology: Human Perception and Performance*, *49*(4), 573–587. https://doi.org/10.1037/xhp0001100

Ashwood, Z. C., Roy, N. A., Stone, I. R., International Brain Laboratory, Urai, A. E., Churchland, A. K., Pouget, A., & Pillow, J. W. (2022). Mice alternate between discrete strategies during perceptual decision-making. *Nature Neuroscience*, *25*(2), 201–212. https://doi.org/10.1038/s41593-021-01007-z

Asparouhov, T., Ellen L., H., , & Muthén, B. (2018). Dynamic Structural Equation Models. *Structural Equation Modeling: A Multidisciplinary Journal*, *25*(3), 359–388. https://doi.org/10.1080/10705511.2017.1406803

Asparouhov, T., , & Muthén, B. (2023). Residual Structural Equation Models. *Structural Equation Modeling: A Multidisciplinary Journal*, *30*(1), 1–31. https://doi.org/10.1080/10705511.2022.2074422

Atroszko, P. A., Atroszko, B., & Charzy
'nska, E. (2021). Subpopulations of Addictive Behaviors in Different Sample Types and Their Relationships with Gender, Personality, and Well-Being: Latent Profile vs. Latent Class Analysis. *International Journal of Environmental Research and Public Health*, *18*(16), 8590. https://doi.org/10.3390/ijerph18168590

Baayen, H., Vasishth, S., Kliegl, R., & Bates, D. (2017). The cave of shadows: Addressing the human factor with generalized additive mixed models. *Journal of Memory and Language*,

94, 206–234. https://doi.org/10.1016/j.jml.2016.11.006

Baayen, R. H., Fasiolo, M., Wood, S., & Chuang, Y.-Y. (2022). A note on the modeling of the effects of experimental time in psycholinguistic experiments. *The Mental Lexicon*, *17*(2), 178–212. https://doi.org/10.1075/ml.21012.baa

Babu, A. S., Scotti, P. S., & Golomb, J. D. (2023). The dominance of spatial information in object identity judgments: A persistent congruency bias even amidst conflicting statistical regularities. *Journal of Experimental Psychology: Human Perception and Performance*, *49*(5), 672–686. https://doi.org/10.1037/xhp0001104

Barbosa Escobar, F., Velasco, C., Byrne, D. V., & Wang, Q. J. (2023). Assessing mechanisms behind crossmodal associations between visual textures and temperature concepts. *Journal of Experimental Psychology: Human Perception and Performance*, *49*(6), 923–947. https://doi.org/10.1037/xhp0001131

Barnes, L., Rangelov, D., Mattingley, J. B., & Woolgar, A. (2023). Fractionating distraction: How past- and future-relevant distractors influence integrated decisions. *Journal of Experimental Psychology: Human Perception and Performance*, *49*(5), 737–752. https://doi.org/10.1037/xhp0001081

Barr, D. J., Levy, R., Scheepers, C., & Tily, H. J. (2013). Random effects structure for confirmatory hypothesis testing: Keep it maximal. *Journal of Memory and Language*, *68*(3), 10.1016/j.jml.2012.11.001. https://doi.org/10.1016/j.jml.2012.11.001

Bauer, D. J. (2016). A More General Model for Testing Measurement Invariance and Differential Item Functioning. *Psychological Methods*, *22*(3), 507. https://doi.org/10.1037/met0000077

Bean, G. J., & Bowen, N. K. (2021). Item Response Theory and Confirmatory Factor Analysis: Complementary Approaches for Scale Development. *Journal of Evidence-Based Social Work*, *18*(6), 597–618. https://doi.org/10.1080/26408066.2021.1906813

Bence, J. R. (1995). Analysis of Short Time Series: Correcting for Autocorrelation. *Ecology*, *76*(2), 628–639. https://doi.org/10.2307/1941218

Berger, V. W., Bour, L. J., Carter, K., Chipman, J. J., Everett, C. C., Heussen, N., Hewitt, C., Hilgers, R.-D., Luo, Y. A., Renteria, J., Ryeznik, Y., Sverdlov, O., Uschner, D., Beckman, R. A., & for the Randomization Innovative Design Scientific Working Group. (2021). A roadmap to using randomization in clinical trials. *Bmc Medical Research Methodology*, *21*(1), 168. https://doi.org/10.1186/s12874-021-01303-z

Bissett, P. G., Jones, H. M., Hagen, M. P., Bui, T. T., Li, J. K., Rios, J. A. H., Mumford, J. A., Shine, J. M., & Poldrack, R. A. (2023). A dual-task approach to inform the taxonomy of inhibition-related processes. *Journal of Experimental Psychology: Human Perception and Performance*, *49*(3), 277–289. https://doi.org/10.1037/xhp0001073

Blackwell, D., & Hodges, J. L. (1957). Design for the control of selection bias. *The Annals of Mathematical Statistics*, *28*, 449–460.

Bogon, J., Köllnberger, K., Thomaschke, R., & Pfister, R. (2023). Binding and retrieval of temporal action features: Probing the precision level of feature representations in action planning. *Journal of Experimental Psychology: Human Perception and Performance*, *49*(7), 989–998. https://doi.org/10.1037/xhp0001136

Bollini, A., Cocchi, E., Salvagno, V., & Gori, M. (2023). The causal role of vision in the development of spatial coordinates: Evidence from visually impaired children. *Journal of Experimental Psychology: Human Perception and Performance*, *49*(7), 1042–1052. https://doi.org/10.1037/xhp0001122

Borchers, H. W. (2022a). *Numbers: Number-theoretic functions* [Manual].

Borchers, H. W. (2022b). *Pracma: Practical numerical math functions* [Manual].

Buchanan, E. M., Foreman, R. E., Johnson, B. N., Pavlacic, J. M., Swadley, R. L., & Schulenberg, S. E. (2018). Does the delivery matter? Examining randomization at the item level. *Behaviormetrika*, *45*(2), 295–316. https://doi.org/10.1007/s41237-018-0055-y

Buračas, G. T., & Boynton, G. M. (2002). Efficient design of event-related fMRI experiments using M-sequences. *Neuroimage*, *16*(3), 801–813.

Button, K. S., Ioannidis, J. P. A., Mokrysz, C., Nosek, B. A., Flint, J., Robinson, E. S. J., & Munafò, M. R. (2013). Power failure: Why small sample size undermines the reliability of neuroscience. *Nature Reviews Neuroscience*, *14*(5), 365–376. https://doi.org/10.1038/nrn3475

Chan, H. M., & Saunders, J. A. (2023). The influence of valence and motivation dimensions of affective states on attentional breadth and the attentional blink. *Journal of Experimental Psychology: Human Perception and Performance*, *49*(1), 34–50. https://doi.org/10.1037/xhp0001060

Chang, S., Dube, B., Golomb, J. D., & Leber, A. B. (2023). Learned spatial suppression is not always proactive. *Journal of Experimental Psychology: Human Perception and Performance*, *49*(7), 1031–1041. https://doi.org/10.1037/xhp0001133

Chen, C.-T., & Wang, W.-C. (2007). Effects of Ignoring Item Interaction on Item Parameter Estimation and Detection of Interacting Items. *Applied Psychological Measurement*, *31*(5), 388–411. https://doi.org/10.1177/0146621606297309

Chen, L., Yang, X., Ge, Z., Liu, L., Yang, X., Yang, P., & Li, L. (2023). High visual perceptual load reduces prepulse inhibition induced by task-unrelated and task-related sound. *Journal of Experimental Psychology: Human Perception and Performance*, *49*(4), 496–511. https://doi.org/10.1037/xhp0001085

Chen, M. S.-Y., Cave, K. R., & Chen, Z. (2023). Learning not to attend to distractors if the task

is demanding: Constraints on the attentional white bear effect. *Journal of Experimental Psychology: Human Perception and Performance*, *49*(4), 523–536. https://doi.org/10.1037/xhp0001099

Cheng, S., Ai, H., Ge, Y., Luo, Y., & Chen, N. (2023). Visual statistical learning of naturalistic textures. *Journal of Experimental Psychology: Human Perception and Performance*, *49*(12), 1579–1590. https://doi.org/10.1037/xhp0001152

Chin, W. W. (1998). Commentary: Issues and Opinion on Structural Equation Modeling. *Mis Quarterly*, *22*(1), vii-xvi. https://www.jstor.org/stable/249674

Cohen, J. (1962). The statistical power of abnormal-social psychological research: A review. *The Journal of Abnormal and Social Psychology*, *65*(3), 145–153. https://doi.org/10.1037/h0045186

Colvett, J. S., Weidler, B. J., & Bugg, J. M. (2023). Revealing object-based cognitive control in a moving object paradigm. *Journal of Experimental Psychology: Human Perception and Performance*, *49*(11), 1467–1484. https://doi.org/10.1037/xhp0001158

Coulacoglou, C., & Saklofske, D. H. (2017). Chapter 3 - Validity. In C. Coulacoglou & D. H. Saklofske (Eds.), *Psychometrics and Psychological Assessment* (pp. 45–66). Academic Press. https://doi.org/10.1016/B978-0-12-802219-1.00003-1

Cui, A. Y., Lleras, A., Ng, G. J. P., & Buetti, S. (2023). Complex background information slows down parallel search efficiency by reducing the strength of interitem interactions. *Journal of Experimental Psychology: Human Perception and Performance*, *49*(7), 1053–1067. https://doi.org/10.1037/xhp0001130

De Boeck, P., & Jeon, M. (2019). An Overview of Models for Response Times and Processes in Cognitive Tests. *Frontiers in Psychology*, *10*. https://doi.org/10.3389/fpsyg.2019.00102

Dempster, A. P., Laird, N. M., & Rubin, D. B. (1977). Maximum Likelihood from Incomplete Data Via the EM Algorithm. *Journal of the Royal Statistical Society: Series B (Methodological)*, *39*(1), 1–22. https://doi.org/10.1111/j.2517-6161.1977.tb01600.x

Durgin, F. H., & Portley, M. (2023). Is the approximate number system capacity limited? Extended display duration does not increase the limits of linear number estimation. *Journal of Experimental Psychology: Human Perception and Performance*, *49*(4), 483–495. https://doi.org/10.1037/xhp0001106

Efron, B. (1971). Forcing a sequential experiment to be balanced. *Biometrika*, *58*, 403–417.

Eggleston, A., Cook, R., & Over, H. (2023). Are upside-down faces perceived as 'less human'? *Journal of Experimental Psychology: Human Perception and Performance*, *49*(12), 1503–1517. https://doi.org/10.1037/xhp0001167

Fagan, A. A., Horn, M. L. V., Hawkins, J. D., & Jaki, T. (2013). Differential Effects of Parental

Controls on Adolescent Substance Use: For Whom Is the Family Most Important? *Journal of Quantitative Criminology*, *29*(3), 347. https://doi.org/10.1007/s10940-012-9183-9

Fang, W., Galusca, C. I., Wang, Z., Sun, Y.-H. P., Pascalis, O., & Xiao, N. G. (2023). Facial dominance augments perceived proximity: Evidence from a visual illusion. *Journal of Experimental Psychology: Human Perception and Performance*, *49*(5), 635–648. https://doi.org/10.1037/xhp0001102

Fitzmaurice, G. M., Laird, N. M., & Ware, J. H. (2012). *Applied longitudinal analysis*. John Wiley & Sons.

Garnier-Allain, A., Pressnitzer, D., & Sergent, C. (2023). Retrospective cueing mediates flexible conscious access to past spoken words. *Journal of Experimental Psychology: Human Perception and Performance*, *49*(7), 949–967. https://doi.org/10.1037/xhp0001132

Gehlbach, H., & Barge, S. (2012). Anchoring and Adjusting in Questionnaire Responses. *Basic and Applied Social Psychology*, *34*(5), 417–433. https://doi.org/10.1080/01973533.2012.711691

Gibson, B. S., Trost, J. M., & Maxwell, S. E. (2023). Top-down attention control does not imply voluntary attention control for all individuals. *Journal of Experimental Psychology: Human Perception and Performance*, *49*(1), 87–107. https://doi.org/10.1037/xhp0001068

Gilden, D. L., Thornton, T., & Mallon, M. W. (1995). 1/F Noise in Human Cognition. *Science*, *267*(5205), 1837–1839. https://www.jstor.org/stable/2886349

Goldberg, L. R. (1992). The development of markers for the Big-Five factor structure. *Psychological Assessment*, *4*(1), 26–42. https://doi.org/10.1037/1040-3590.4.1.26

Goldberg, L. R., Johnson, J. A., Eber, H. W., Hogan, R., Ashton, M. C., Cloninger, C. R., & Gough, H. G. (2006). The international personality item pool and the future of public-domain personality measures. *Journal of Research in Personality*, *40*(1), 84–96. https://doi.org/10.1016/j.jrp.2005.08.007

Good, P. (2013). *Permutation tests: A practical guide to resampling methods for testing hypotheses*. Springer Science & Business Media.

Goodridge, C. M., Billington, J., Markkula, G., & Wilkie, R. M. (2023). Error accumulation when steering toward curves. *Journal of Experimental Psychology: Human Perception and Performance*, *49*(6), 821–834. https://doi.org/10.1037/xhp0001101

Guitard, D., & Cowan, N. (2023). The tradeoff between item and order information in short-term memory does not depend on encoding time. *Journal of Experimental Psychology: Human Perception and Performance*, *49*(1), 51–70. https://doi.org/10.1037/xhp0001074

Gutzeit, J., Weller, L., Kürten, J., & Huestegge, L. (2023). Intentional binding: Merely a procedural confound? *Journal of Experimental Psychology: Human Perception and Performance*, *49*(6), 759–773. https://doi.org/10.1037/xhp0001110

Hallquist, M. N., & Wiley, J. F. (2018). MplusAutomation: An R package for facilitating large-scale latent variable analyses in Mplus. *Structural Equation Modeling*, 621–638. https://doi.org/10.1080/10705511.2017.1402334

Hill, A. B. (1952). The clinical trial. *The New England Journal of Medicine*, *247*, 113–119.

Holden, J. G. (2005). Gauging the fractal dimension of response times from cognitive tasks. *Contemporary Nonlinear Methods for Behavioral Scientists: A Webbook Tutorial*, 267–318.

Honda, C., Pruitt, T. A., Greenspon, E. B., Liu, F., & Pfordresher, P. Q. (2023). The effect of musical training and language background on vocal imitation of pitch in speech and song. *Journal of Experimental Psychology: Human Perception and Performance*, *49*(10), 1296–1309. https://doi.org/10.1037/xhp0001146

Hoversten, L. J., & Martin, C. D. (2023). Parafoveal processing in bilingual readers: Semantic access within but not across languages. *Journal of Experimental Psychology: Human Perception and Performance*, *49*(12), 1564–1578. https://doi.org/10.1037/xhp0001161

Hu, Y., Yang, Y., Huang, P., Ai, D., Sun, H., Zhou, D., Huangliang, J., & Yin, J. (2023). More evidence, greater generalization? The relation between the prevalence of observed action and the strength of generalization depends on action properties. *Journal of Experimental Psychology: Human Perception and Performance*, *49*(3), 306–326. https://doi.org/10.1037/xhp0001097

Kang, W., & Longo, M. R. (2023). Tactile localization on stretched skin. *Journal of Experimental Psychology: Human Perception and Performance*, *49*(8), 1175–1179. https://doi.org/10.1037/xhp0001142

Kershner, A. M., & Hollingworth, A. (2023). Category-specific learning of color, orientation, and position regularities guide visual search. *Journal of Experimental Psychology: Human Perception and Performance*, *49*(6), 907–922. https://doi.org/10.1037/xhp0001098

Kinoshita, S., Amos, A., & Norris, D. (2023). Diacritic priming in novice readers of diacritics. *Journal of Experimental Psychology: Human Perception and Performance*, *49*(3), 370–383. https://doi.org/10.1037/xhp0001084

Kirk, R. E. (2013). *Experimental design: Procedures for the behavioral sciences (fourth edition)*. Sage. https://doi.org/10.4135/9781483384733

Klassen, N. R., Bamford, L. E., & Karl, J. M. (2023). Peri-hand space: A helping hand for faster object recognition in children. *Journal of Experimental Psychology: Human Perception and Performance*, *49*(4), 512–522. https://doi.org/10.1037/xhp0001111

Kolbe, L., Molenaar, D., Jak, S., & Jorgensen, T. D. (2024). Assessing measurement invariance with moderated nonlinear factor analysis using the R package OpenMx. *Psychological Methods*, *29*(2), 388–406. https://doi.org/10.1037/met0000501

Lavelle, M., Luria, R., & Drew, T. (2023). Incidental recognition reveals attentional tradeoffs

shaped by categorical similarity. *Journal of Experimental Psychology: Human Perception and Performance*, *49*(6), 893–906. https://doi.org/10.1037/xhp0001128

Lee, Y. S., & Cho, Y. S. (2023). The congruency sequence effect of the Simon task in a cross-modality context. *Journal of Experimental Psychology: Human Perception and Performance*, *49*(9), 1221–1235. https://doi.org/10.1037/xhp0001145

Lerebourg, M., de Lange, F. P., & Peelen, M. V. (2023). Expected distractor context biases the attentional template for target shapes. *Journal of Experimental Psychology: Human Perception and Performance*, *49*(9), 1236–1255. https://doi.org/10.1037/xhp0001129

Liang, J., & Barr, D. J. (2024). Better power by design: Permuted-subblock randomization boosts power in repeated-measures experiments. *Psychological Methods*. https://doi.org/10.1037/met0000717

Liang, J., Wang, M., Luo, J., Liang, J., Xintong, Z., & Gao, Y. (2022). Elaborating on the Construct Validity of the Antisocial Process Screening Device in Chinese Children and Adolescents: Across-Informants and Across-Samples. *Current Psychology*, *41*. https://doi.org/10.1007/s12144-020-00777-2

Ligges, U., Krey, S., Mersmann, O., & Schnackenberg, S. (2023). *tuneR: Analysis of music and speech* [Manual].

Liu, Y., & Liu, H. (2021). Detecting Noneffortful Responses Based on a Residual Method Using an Iterative Purification Process. *Journal of Educational and Behavioral Statistics*, *46*(6), 717–752. https://doi.org/10.3102/1076998621994366

Lo, Y., Mendell, N. R., & Rubin, D. B. (2001). Testing the Number of Components in a Normal Mixture. *Biometrika*, *88*(3), 767–778. https://www.jstor.org/stable/2673445

Loiacono, E., & Wilson, E. (2020). Do We Truly Sacrifice Truth for Simplicity: Comparing Complete Individual Randomization and Semi- Randomized Approaches to Survey Administration. *Ais Transactions on Human-Computer Interaction*, 45–69. https://doi.org/10.17705/1thci.00128

Lubke, G., & Muthén, B. O. (2007). Performance of factor mixture models as a function of model size, covariate effects, and class-specific parameters. *Structural Equation Modeling*, *14*(1), 26–47. https://doi.org/10.1207/s15328007sem1401_2

Luke, S. G. (2017). Evaluating significance in linear mixed-effects models in R. *Behavior Research Methods*, *49*(4), 1494–1502. https://doi.org/10.3758/s13428-016-0809-y

Ma, X., & Abrams, R. A. (2023). Ignoring the unknown: Attentional suppression of unpredictable visual distraction. *Journal of Experimental Psychology: Human Perception and Performance*, *49*(1), 1–6. https://doi.org/10.1037/xhp0001067

Mainka, T., Ganos, C., & Longo, M. R. (2023). Skin stretch modulates tactile distance perception without central correction mechanisms. *Journal of Experimental Psychology: Human*

Perception and Performance, *49*(2), 226–235. https://doi.org/10.1037/xhp0001063

Manzone, J. X., & Welsh, T. N. (2023). Modulation of response activation leads to biases in perceptuomotor decision making. *Journal of Experimental Psychology: Human Perception and Performance*, *49*(7), 1090–1109. https://doi.org/10.1037/xhp0001140

Marsh, H. W. (1996). Positive and negative global self-esteem: A substantively meaningful distinction or artifactors? *Journal of Personality and Social Psychology*, *70*(4), 810–819. https://doi.org/10.1037//0022-3514.70.4.810

Marszalek, J. M., Barber, C., Kohlhart, J., & Holmes, C. B. (2011). Sample size in psychological research over the past 30 years. *Perceptual and Motor Skills*, *112*(2), 331–348. https://doi.org/10.2466/03.11.PMS.112.2.331-348

Marzola, G., & Cohen, D. J. (2023). Mirror numbers activate quantity representations, but show no SNARC effect: A working memory explanation. *Journal of Experimental Psychology: Human Perception and Performance*, *49*(4), 465–482. https://doi.org/10.1037/xhp0001090

Masyn, K. E. (2013). Latent Class Analysis and Finite Mixture Modeling. In T. D. Little (Ed.), *The Oxford Handbook of Quantitative Methods in Psychology: Vol. 2: Statistical Analysis* (p. 0). Oxford University Press. https://doi.org/10.1093/oxfordhb/9780199934898.013.0025

Matts, J. P., & Lachin, J. M. (1988). Properties of permuted-block randomization in clinical trials. *Controlled Clinical Trials*. https://doi.org/10.1016/0197-2456(88)90047-5

Maxwell, S. E. (2004). The Persistence of Underpowered Studies in Psychological Research: Causes, Consequences, and Remedies. *Psychological Methods*, *9*(2), 147–163. https://doi.org/10.1037/1082-989X.9.2.147

McNeish, D. (2018). Thanks coefficient alpha, we'll take it from here. *Psychological Methods*, *23*(3), 412–433. https://doi.org/10.1037/met0000144

Milligan, S., Nestor, B., Antúnez, M., & Schotter, E. R. (2023). Out of sight, out of mind: Foveal processing is necessary for semantic integration of words into sentence context. *Journal of Experimental Psychology: Human Perception and Performance*, *49*(5), 687–708. https://doi.org/10.1037/xhp0001121

Mirman, D. (2016). *Growth Curve Analysis and Visualization Using R*. Chapman and Hall/CRC. https://doi.org/10.1201/9781315373218

Muthen, L., Muthen, B., & Angeles)., M. &. M. (. (2017). *Mplus version 8 user's guide*. Muthen & Muthen.

Myszkowski, N., & Storme, M. (2024). Modeling Sequential Dependencies in Progressive Matrices: An Auto-Regressive Item Response Theory (AR-IRT) Approach. *Journal of Intelligence*, *12*(1), 7. https://doi.org/10.3390/jintelligence12010007

Narhi-Martinez, W., Chen, J., & Golomb, J. D. (2023). Probabilistic visual attentional guidance triggers 'feature avoidance' response errors. *Journal of Experimental Psychology: Human Perception and Performance*, *49*(6), 802–820. https://doi.org/10.1037/xhp0001095

Nedergaard, J., Skewes, J. C., & Wallentin, M. (2023). 'Stay focused!': The role of inner speech in maintaining attention during a boring task. *Journal of Experimental Psychology: Human Perception and Performance*, *49*(4), 451–464. https://doi.org/10.1037/xhp0001112

Negen, J., Bird, L.-A., Slater, H., Thaler, L., & Nardini, M. (2023). Multisensory perception and decision-making with a new sensory skill. *Journal of Experimental Psychology: Human Perception and Performance*, *49*(5), 600–622. https://doi.org/10.1037/xhp0001114

Newell, A., & Rosenbloom, P. S. (1981). Mechanisms of skill acquisition and the law of practice. In J. R. Anderson (Ed.), *Cognitive skills and their acquisition* (pp. 1–55). Erlbaum.

Nguyen, H. B., & van Buren, B. (2023). May the force be against you: Better visual sensitivity to speed changes opposite to gravity. *Journal of Experimental Psychology: Human Perception and Performance*, *49*(7), 1016–1030. https://doi.org/10.1037/xhp0001115

Nguyen, T. H., Han, H.-R., Kim, M. T., & Chan, K. S. (2014). An Introduction to Item Response Theory for Patient-Reported Outcome Measurement. *The Patient*, *7*(1), 23. https://doi.org/10.1007/s40271-013-0041-0

Nylund, K. L., Asparouhov, T., & Muthén, B. O. (2007). Deciding on the Number of Classes in Latent Class Analysis and Growth Mixture Modeling: A Monte Carlo Simulation Study. *Structural Equation Modeling: A Multidisciplinary Journal*, *14*(4), 535–569. https://doi.org/10.1080/10705510701575396

Orzek, J. H. (2024). *Mxsem: Specify 'OpenMx' models with a 'lavaan'-style syntax* [Manual].

Overkott, C., & Souza, A. S. (2023). The fate of labeled and nonlabeled visual features in working memory. *Journal of Experimental Psychology: Human Perception and Performance*, *49*(3), 384–407. https://doi.org/10.1037/xhp0001089

Ozkok, O., Zyphur, M. J., Barsky, A. P., Theilacker, M., Donnellan, M. B., & Oswald, F. L. (2019). Modeling Measurement as a Sequential Process: Autoregressive Confirmatory Factor Analysis (AR-CFA). *Frontiers in Psychology*, *10*.

Pashler, H., & Wagenmakers Eric–Jan. (2012). Editors' Introduction to the Special Section on Replicability in Psychological Science: A Crisis of Confidence? *Perspectives on Psychological Science*, *7*(6), 528–530. https://doi.org/10.1177/1745691612465253

Pedregon, C. A., Farley, R. L., Davis, A., Wood, J. M., & Clark, R. D. (2012). Social desirability, personality questionnaires, and the "better than average" effect. *Personality and Individual Differences*, *52*(2), 213–217. https://doi.org/10.1016/j.paid.2011.10.022

Pedziwiatr, M. A., von dem Hagen, E., & Teufel, C. (2023). Knowledge-driven perceptual organization reshapes information sampling via eye movements. *Journal of Experimental*

Psychology: Human Perception and Performance, *49*(3), 408–427. https://doi.org/10.1037/xhp0001080

Peker, A. T., Böge, V., Bailey, G. S., Wagman, J. B., & Stoffregen, T. A. (2023). Perception of higher-order affordances for kicking in soccer. *Journal of Experimental Psychology: Human Perception and Performance*, *49*(5), 623–634. https://doi.org/10.1037/xhp0001108

Pocock, S. J. (1979). Allocation of patients to treatment in clinical trials. *Biometrics*, *35*(1), 183–197.

Pornprasertmanit, S., Miller, P., Schoemann, A., & Jorgensen, T. D. (2021). *Simsem: SIMulated structural equation modeling* [Manual].

Qiu, R., Möller, M., Koch, I., Frings, C., & Mayr, S. (2023). The influence of event segmentation by context on stimulus–response binding. *Journal of Experimental Psychology: Human Perception and Performance*, *49*(3), 355–369. https://doi.org/10.1037/xhp0001093

R Core Team. (2023). *R: A language and environment for statistical computing* [Manual]. R Foundation for Statistical Computing.

Ramgir, A., & Lamy, D. (2023). Distractor's salience does not determine feature suppression: A commentary on Wang and Theeuwes (2020). *Journal of Experimental Psychology: Human Perception and Performance*, *49*(6), 852–861. https://doi.org/10.1037/xhp0001119

Robertson, I. H., Manly, T., Andrade, J., Baddeley, B. T., & Yiend, J. (1997). 'Oops!': Performance correlates of everyday attentional failures in traumatic brain injured and normal subjects. *Neuropsychologia*, *35*(6), 747–758. https://doi.org/10.1016/S0028-3932(97)00015-8

Rosenbaum, D., Mama, Y., & Algom, D. (2017). Stand by Your Stroop: Standing Up Enhances Selective Attention and Cognitive Control. *Psychological Science*, *28*(12), 1864–1867. https://doi.org/10.1177/0956797617721270

Rosenthal, R., & Rosnow, R. L. (2008). *Essentials of behavioral research: Methods and data analysis (third edition)*. McGraw-Hill.

Savino, G. E., & Kahan, T. A. (2023). Target-mask similarity affects both object substitution masking and object recovery. *Journal of Experimental Psychology: Human Perception and Performance*, *49*(2), 263–275. https://doi.org/10.1037/xhp0001072

Schaaf, M., Wirth, R., & Kunde, W. (2023). Time expectancies in dual tasking: Evidence for proactive resource sharing? *Journal of Experimental Psychology: Human Perception and Performance*, *49*(8), 1123–1131. https://doi.org/10.1037/xhp0001141

Scheibel, M., & Indefrey, P. (2023). Top-down enhanced object recognition in blocking and priming paradigms. *Journal of Experimental Psychology: Human Perception and Performance*, *49*(3), 327–354. https://doi.org/10.1037/xhp0001094

Schell, K. L., & Oswald, F. L. (2013). Item grouping and item randomization in personality

measurement. *Personality and Individual Differences*, *55*(3), 317–321. https://doi.org/10.1016/j.paid.2013.03.008

Schirmer, A., Cham, C., Lai, O., Le, T.-l. S., & Ackerley, R. (2023). Stroking trajectory shapes velocity effects on pleasantness and other touch percepts. *Journal of Experimental Psychology: Human Perception and Performance*, *49*(1), 71–86. https://doi.org/10.1037/xhp0001079

Schmalbrock, P., Liesefeld, H. R., & Frings, C. (2023). Increased display complexity reveals effects of salience in action control. *Journal of Experimental Psychology: Human Perception and Performance*, *49*(10), 1345–1359. https://doi.org/10.1037/xhp0001151

Sears, D. R. W., Verbeten, J. E., & Percival, H. M. (2023). Does order matter? Harmonic priming effects for scrambled tonal chord sequences. *Journal of Experimental Psychology: Human Perception and Performance*, *49*(7), 999–1015. https://doi.org/10.1037/xhp0001103

Severijnen, G. G. A., Di Dona, G., Bosker, H. R., & McQueen, J. M. (2023). Tracking talker-specific cues to lexical stress: Evidence from perceptual learning. *Journal of Experimental Psychology: Human Perception and Performance*, *49*(4), 549–565. https://doi.org/10.1037/xhp0001105

Shadish, W. R., Cook, T. D., & Campbell, D. T. (2002). *Experimental and quasi-experimental designs for generalized causal inference* (p. xxi, 623). Houghton, Mifflin and Company.

Shimada, D., & Katahira, K. (2023). Sequential dependencies of responses in a questionnaire survey and their effects on the reliability and validity of measurement. *Behavior Research Methods*, *55*(6), 3241–3259. https://doi.org/10.3758/s13428-022-01943-z

Shinozuka, M., & Deodatis, G. (1991). Simulation of stochastic processes by spectral representation. *Applied Mechanics Reviews*, *44*, 191–204.

Shireman, E., Steinley, D., & Brusco, M. J. (2017). Examining the effect of initialization strategies on the performance of Gaussian mixture modeling. *Behavior Research Methods*, *49*(1), 282. https://doi.org/10.3758/s13428-015-0697-6

Shukla, K., , & Konold, T. (2018). A Two-Step Latent Profile Method for Identifying Invalid Respondents in Self-Reported Survey Data. *The Journal of Experimental Education*, *86*(3), 473–488. https://doi.org/10.1080/00220973.2017.1315713

Siqi-Liu, A., & Egner, T. (2023). Task sets define boundaries of learned cognitive flexibility in list-wide proportion switch manipulations. *Journal of Experimental Psychology: Human Perception and Performance*, *49*(8), 1111–1122. https://doi.org/10.1037/xhp0001138

Sobrinho, N. D., & Souza, A. S. (2023). The interplay of long-term memory and working memory: When does object-color prior knowledge affect color visual working memory? *Journal of Experimental Psychology: Human Perception and Performance*, *49*(2), 236–262. https://doi.org/10.1037/xhp0001071

Stroop, J. R. (1935). Studies of interference in serial verbal reactions. *Journal of Experimental Psychology*, *18*(6), 643–662.

Tang, X., George, K., , & Chen, H. (2020). Detecting Local Dependence: A Threshold-Autoregressive Item Response Theory (TAR-IRT) Approach for Polytomous Items. *Applied Measurement in Education*, *33*(4), 280–292. https://doi.org/10.1080/08957347.2020.1789136

Tein, J.-Y., Coxe, S., & Cham, H. (2013). Statistical Power to Detect the Correct Number of Classes in Latent Profile Analysis. *Structural Equation Modeling : a Multidisciplinary Journal*, *20*(4), 640. https://doi.org/10.1080/10705511.2013.824781

Thul, R., Conklin, K., & Barr, D. J. (2021). Using GAMMs to model trial-by-trial fluctuations in experimental data: More risks but hardly any benefit. *Journal of Memory and Language*, *120*, 104247. https://doi.org/10.1016/j.jml.2021.104247

Van Geert, E., & Wagemans, J. (2023). What good is goodness? The effects of reference points on discrimination and categorization of shapes. *Journal of Experimental Psychology: Human Perception and Performance*, *49*(8), 1180–1201. https://doi.org/10.1037/xhp0001137

Van Lissa, C. J., Garnier-Villarreal, M., & Anadria, D. (2024). Recommended Practices in Latent Class Analysis Using the Open-Source R-Package tidySEM. *Structural Equation Modeling: A Multidisciplinary Journal*, *31*(3), 526–534. https://doi.org/10.1080/10705511.2023.2250920

Vandenberghe, A., & Vannuscorps, G. (2023). Predictive extrapolation of observed body movements is tuned by knowledge of the body biomechanics. *Journal of Experimental Psychology: Human Perception and Performance*, *49*(2), 188–196. https://doi.org/10.1037/xhp0001077

Veldre, A., Reichle, E. D., Yu, L., & Andrews, S. (2023). Lexical processing across the visual field. *Journal of Experimental Psychology: Human Perception and Performance*, *49*(5), 649–671. https://doi.org/10.1037/xhp0001109

Wang, J., Fang, S., Yang, C., Tang, X., Zhu, L., & Nie, Y. (2023). The Relationship Between Psychological Flexibility and Depression, Anxiety and Stress: A Latent Profile Analysis. *Psychology Research and Behavior Management*, *16*, 997–1007. https://doi.org/10.2147/PRBM.S400757

Wang, M.-C., Shou, Y., Liang, J., Lai, H., Zeng, H., Chen, L., & Gao, Y. (2020). Further Validation of the Inventory of Callous–Unemotional Traits in Chinese Children: Cross-Informants Invariance and Longitudinal Invariance. *Assessment*, *27*(7), 1668–1680. https://doi.org/10.1177/1073191119845052

Weijters, B., De Beuckelaer, A., & Baumgartner, H. (2014). Discriminant Validity Where There Should Be None: Positioning Same-Scale Items in Separated Blocks of a Question-

naire. *Applied Psychological Measurement*, *38*(6), 450–463. https://doi.org/10.1177/0146621614531850

Wentura, D., Gurbuz, E., Paulus, A., & Rohr, M. (2023). Emotional face expressions and group membership: Does affective mismatch induce conflict? *Journal of Experimental Psychology: Human Perception and Performance*, *49*(11), 1395–1406. https://doi.org/10.1037/xhp0001163

Wirth, B. E., Ramgir, A., & Lamy, D. (2023). Feature intertrial priming biases attentional priority: Evidence from the capture-probe paradigm. *Journal of Experimental Psychology: Human Perception and Performance*, *49*(8), 1145–1157. https://doi.org/10.1037/xhp0001135

Wirth, R., Tonn, S., Schaaf, M., Koch, I., & Kunde, W. (2023). Sequential adaptation to modality incompatibility. *Journal of Experimental Psychology: Human Perception and Performance*, *49*(10), 1360–1376. https://doi.org/10.1037/xhp0001149

Wo

'zniak, M., McEllin, L., Hohwy, J., & Ciaunica, A. (2023). Depersonalization affects self-prioritization of bodily, but not abstract self-related information. *Journal of Experimental Psychology: Human Perception and Performance*, *49*(11), 1447–1459. https://doi.org/10.1037/xhp0001153

Wood, S. N. (2017). *Generalized Additive Models: An Introduction with R, Second Edition* (2nd ed.). Chapman and Hall/CRC. https://doi.org/10.1201/9781315370279

Yan, N., Grindell, J., & Anderson, B. A. (2023). Encoding history enhances working memory encoding: Evidence from attribute amnesia. *Journal of Experimental Psychology: Human Perception and Performance*, *49*(5), 589–599. https://doi.org/10.1037/xhp0001096

Yarrow, K., Solomon, J. A., Arnold, D. H., & Roseboom, W. (2023). The best fitting of three contemporary observer models reveals how participants' strategy influences the window of subjective synchrony. *Journal of Experimental Psychology: Human Perception and Performance*, *49*(12), 1534–1563. https://doi.org/10.1037/xhp0001154

Yu, H., Allenmark, F., Müller, H. J., & Shi, Z. (2023). Asymmetric learning of dynamic spatial regularities in visual search: Robust facilitation of predictable target locations, fragile suppression of distractor locations. *Journal of Experimental Psychology: Human Perception and Performance*, *49*(5), 709–724. https://doi.org/10.1037/xhp0001120

Yu, M. E., Cooper, A., & Johnson, E. K. (2023). Who speaks 'kid?' How experience with children does (and does not) shape the intelligibility of child speech. *Journal of Experimental Psychology: Human Perception and Performance*, *49*(4), 441–450. https://doi.org/10.1037/xhp0001088

Zhang, Y., Ye, S., Chen, W., & Ding, X. (2023). When 'looking at nothing' imparts something: Retrospective gaze cues flexibly direct prioritization in visual working memory. *Journal of*

Experimental Psychology: Human Perception and Performance, *49*(11), 1407–1419. https://doi.org/10.1037/xhp0001160

Ziaka, L., & Protopapas, A. (2023). Cognitive control beyond single-item tasks: Insights from pupillometry, gaze, and behavioral measures. *Journal of Experimental Psychology: Human Perception and Performance*, *49*(7), 968–988. https://doi.org/10.1037/xhp0001127

van Casteren, M., & Davis, M. (2006). Mix, a program for pseudorandomization. *Behavior Research Methods*, *38*, 584–589. https://doi.org/10.3758/BF03193889

van der Linden, D., Frese, M., & Meijman, T. F. (2003). Mental fatigue and the control of cognitive processes: Effects on perseveration and planning. *Acta Psychologica*, *113*, 45–65. https://doi.org/10.1016/S0001-6918(02)00150-6

Şahin, M. D. g. (2021). Effect of Item Order on Certain Psychometric Properties: A Demonstration on a Cyberloafing Scale. *Frontiers in Psychology*, *12*, 590545. https://doi.org/10.3389/fpsyg.2021.590545