Assessing Latent Dimensionality in Psychological Research

Melissa Prince – BPsych (Hons I)

Thesis submitted for Doctorate of Philosophy, March 2013
Statement of Originality

This thesis contains no material which has been accepted for the award of any other degree or diploma in any university or other tertiary institution and, to the best of my knowledge and belief, contains no material previously published or written by another person, except where due reference has been made in the text. I give consent to this copy of my thesis, when deposited in the University Library, being made available for loan and photocopying subject to the provisions of the Copyright Act, 1968.

Acknowledgement of Collaboration

I hereby certify that the work embodied in this thesis has been done in collaboration with researchers. I have included as part of the thesis a statement clearly outlining the extent of collaboration, with whom and under what auspices.

Thesis by Publication

I hereby certify that this thesis is in the form of a series of published papers of which I am a joint author. I have included as part of the thesis a written statement from each co-author, endorsed by the Faculty Assistant Dean (Research Training), attesting to my contribution to the joint publications.

Melissa Prince
Publications Included in Thesis


Statement of Contribution

FACULTY OF SCIENCE AND
INFORMATION TECHNOLOGY

12th August, 2012

To Whom it may concern,

This letter outlines Melissa Prince’s contribution to the series of papers that are submitted as a part of her PhD. All papers that are contributing to her thesis are listed below, with a statement of her contribution for each.

Regards,

Professor Andrew Heathcote                Associate Professor Scott Brown

Mr Guy Hawkins                            Mr Jonathon Love

Endorsed By:

Associate Professor Jenny Cameron
Assistant Dean (Research Training)

*All authors contributed to all stages of the development of this paper. Melissa in particular was responsible for all data collection for the examples presented and manuscript refinement, 60% contribution. Other authors contributed as follows: S. Brown (10%), A. Heathcote (30%).*


*Melissa contributed to the background experimental work, design of the simulations and the write-up of this article, 30% contribution. Other authors contributed as follows: G. Hawkins (50%), S. Brown (10%), A. Heathcote (10%).*


*This project was jointly led by Melissa Prince and Andrew Heathcote. Melissa was responsible for developing the GUIs (with Guy Hawkins) and testing the package. She also took the lead in writing this article, 50% contribution. Other authors contributed as follows: G. Hawkins (20%), J. Love (5%), A. Heathcote (25%)*

This vignette was developed in conjunction with the previous article. Melissa was responsible for preparing this manuscript, the extensive help documentation included in the package as well as testing the written tutorial with undergraduate research students, 95% contribution. Other authors provided feedback on the final document.


Each of the above three papers, were led by Melissa, 70% contribution (A. Heathcote 30%). Melissa contributed to the experimental design, coordinated and supervised all data collection, completed data analyses and took the lead role in manuscript preparation.
Acknowledgements

I would first like to thank Professor Andrew Heathcote for all of your help and guidance over the years. Without such an enthusiastic supervisor, who was always willing to respond to my questions and concerns in such a collaborative way, I would not be at the stage of completing this thesis. Also to Associate Professor Scott Brown and Dr. Kerry Chalmers, thank you for your continued support and encouragement as well as for your invaluable feedback at various stages of preparing this manuscript.

Thank you to all of the people who have contributed in some way to this thesis – from the numerous participants who were willing to sit through hours of testing, to the students enrolled in Psyc4815/Psyc6781 at the University of Newcastle for assistance with stimulus preparation and data collection. Thanks also to the other members of the Newcastle Cognition Lab, for your advice, support and the much needed coffee trips that kept me motivated – in particular, David Elliott, for programming the experiments.

To my friends, thank you for your patience. And finally to my family, I am forever grateful for all your love and support, which has allowed me to achieve all that I have.
# Table of Contents

Abstract ................................................................................................................................. 17 

Assessing Latent Dimensionality ...................................................................................... 19 
  
  *Overview of Section One: State-Trace Analysis* ................................................................. 23 
  *Overview of Section Two: The Differential Face Inversion Effect* ............................... 26 

Section One: State-Trace Analysis ...................................................................................... 31 
  
  *Chapter One* ...................................................................................................................... 33  
  State-Trace Analysis: What is it and why use it? .............................................................. 35  
  The State-Trace Plot .......................................................................................................... 39  
  State-Trace Experiment Design ........................................................................................ 47  
  Diagnosing Dimensionality ............................................................................................... 48  
  Choosing a Trace Factor ................................................................................................. 52  
  Refining a State-Trace Experiment: A Case Study ......................................................... 53  
  Summary and Recommendations for Experiment Design .......................................... 60  

Statistical Analysis of State-Trace Data ............................................................................. 61 
  
  Existing Statistical Methods ............................................................................................. 61  
  Proposed Method ............................................................................................................ 66  
  Example Analysis ........................................................................................................... 76  
  Summary and Recommendations for Statistical Analysis .......................................... 82  

General Discussion ............................................................................................................ 85 
  
  Future Directions ............................................................................................................ 87  

Technical Details for Bayesian State-Trace Analysis ...................................................... 89 
  
  Prior Proportions ........................................................................................................... 89  
  Posterior Proportions ..................................................................................................... 91  
  Numerical Accuracy ....................................................................................................... 95  

Chapter Two ..................................................................................................................... 97  
  
  A Bayesian Approach to State-Trace Analysis .............................................................. 99  
  Simulations .................................................................................................................... 102  
  Bayesian State-Trace Results ........................................................................................ 103  
  The Trace Model ........................................................................................................... 104  
  The No-Overlap Model ............................................................................................... 106
Chapter Five

A Memory Retrieval Effect?

Method

Participants

Stimuli

Apparatus

Procedure

Results

Discussion

Chapter Six

A Strategic Explanation?

Method

Participants

Stimuli

Apparatus

Procedure

Results

Discussion

Chapter Seven

Refining Loftus, Oberg and Dillon’s (2004) Design

Experiment

Method

Participants

Stimuli and Apparatus

Procedure

Results

Discussion

State-Trace Analysis

General Discussion

Bibliography
## Appendices  
<table>
<thead>
<tr>
<th>Appendix A</th>
<th>...</th>
<th>295</th>
</tr>
</thead>
<tbody>
<tr>
<td>Chapter One</td>
<td></td>
<td>295</td>
</tr>
<tr>
<td>Chapter Two</td>
<td></td>
<td>319</td>
</tr>
<tr>
<td>Chapter Three</td>
<td></td>
<td>327</td>
</tr>
<tr>
<td>Chapter Four</td>
<td></td>
<td>341</td>
</tr>
<tr>
<td>Chapter Five</td>
<td></td>
<td>381</td>
</tr>
<tr>
<td>Chapter Six</td>
<td></td>
<td>389</td>
</tr>
<tr>
<td>Appendix B</td>
<td>...</td>
<td>397</td>
</tr>
<tr>
<td>Method</td>
<td></td>
<td>400</td>
</tr>
<tr>
<td>Participants</td>
<td></td>
<td>400</td>
</tr>
<tr>
<td>Stimuli</td>
<td></td>
<td>400</td>
</tr>
<tr>
<td>Apparatus</td>
<td></td>
<td>402</td>
</tr>
<tr>
<td>Procedure</td>
<td></td>
<td>402</td>
</tr>
<tr>
<td>Results and Discussion</td>
<td></td>
<td>405</td>
</tr>
<tr>
<td>Appendix C</td>
<td>...</td>
<td>409</td>
</tr>
<tr>
<td>Method</td>
<td></td>
<td>412</td>
</tr>
<tr>
<td>Participants</td>
<td></td>
<td>412</td>
</tr>
<tr>
<td>Stimuli and Apparatus</td>
<td></td>
<td>412</td>
</tr>
<tr>
<td>Procedure</td>
<td></td>
<td>412</td>
</tr>
<tr>
<td>Results and Discussion</td>
<td></td>
<td>413</td>
</tr>
<tr>
<td>Linear Analyses</td>
<td></td>
<td>413</td>
</tr>
<tr>
<td>State-Trace Analysis</td>
<td></td>
<td>416</td>
</tr>
<tr>
<td>Summary</td>
<td></td>
<td>418</td>
</tr>
</tbody>
</table>
Abstract

Research in cognitive science and neuroscience has the shared goal of understanding how cognitive and neural representations and processes mediate the observed relationships between stimuli and responses in different experimental paradigms. The almost ubiquitous basis for inferences about the number of processes or latent dimensions involved, is the observation of a dissociation; an interaction due to an unequal or opposite effect of one independent variable on the levels of another independent variable. However, it has been clearly shown that dissociations do not provide strong evidence for the need of an extra dimension. In this thesis, which is a collection of published and submitted papers, we describe an extension of the dissociation methodology – state-trace analysis (Bamber, 1979) – that does provide a rigorous basis for this inference. In the first section, an informal introduction to state-trace analysis is provided. We also develop Bayesian methods suitable for quantifying state-trace evidence in favour of a one-dimensional or multi-dimensional explanation, as well as for refining state-trace experiments. In the second section of this thesis, an application of state-trace analysis is presented that examines the question of whether human face recognition is special in the sense that faces can be encoded in terms of a dimension or dimensions additional to those available to most other objects. Over a series of experiments, and using the new methods developed in Section One, we confirm that the encoding of unfamiliar faces is special and discuss the need to extend this type of analysis to other psychological phenomena.
Assessing Latent Dimensionality

Cognitive science and neuroscience have long been focussed on fundamental questions concerning latent dimensionality: does a single latent (i.e., not directly observable) variable, or dimension, mediate the relationship between two or more experimental factors? Such questions often encompass an interest in determining whether one, or more than one cognitive process, module, representation or brain region is involved in a particular behaviour, and whether this involvement differs between participants, stimuli, tasks, experimental manipulations or dependent measures. For example, it has been questioned whether recognition memory is mediated by one (memory strength) or two (familiarity and recollection) processes (Dunn, 2004, 2008)? It has also been asked whether there is a common basis for recall and recognition performance (Haist, Shimamura & Squire, 1992), recognition confidence and accuracy (Busey, Tunnicliff, Loftus & Loftus, 2000; Reintz, Séguin, Peria & Loftus, 2012), prospective and retrospective confidence judgements (Jang & Nelson, 2005), as well as recognition accuracy for faces and non-faces (Loftus, Oberg & Dillon, 2004). Others have sought to determine if the representation of multiplication facts changes during development (DeBrauwer, Verguts & Fias, 2006) and whether the basis for performance in various divided attention tasks changes in older age (Verharghen & Cerella, 2002). Indeed, Dunn and Kirsner (2003) suggest it has become the joint aim of cognitive psychology and neuropsychology to identify and characterise the fundamental processes that underlie human behaviour.

Although these questions seem straightforward, we cannot directly observe the latent variable(s) of interest. Instead we must measure some response variable and assume that the state of the latent variable is reflected in the variation observed in the
response measure; that is, that the latent variable “maps” on to the response measure. Moreover, even in simple paradigms there is little doubt that a plethora of mediating variables could play a causal role. Nevertheless, we seek to discover low-dimensional explanations of psychological phenomena in terms of key latent variables. Success in this approach requires that Occam’s razor be wielded rigorously; elaborating a theory by adding an extra dimension requires strong evidence, otherwise genuine explanation becomes mere re-description of the data.

Consequently, researchers typically seek patterns of data, called dissociations, in order to characterise latent dimensionality (Shallice, 1988) and hence to infer the existence of functionally independent neural (Teuber, 1955) or cognitive (Glanzer & Cunitz, 1966) systems. A single dissociation is said to occur when an experimental factor selectively affects performance under one condition or in one task or group, or by one measure but not another. A double dissociation, is said to occur when a second experimental factor selectively affects the second condition/task/group/measure but not the first (see Dunn, 2003, for a more detailed discussion). Although stronger evidentiary value is often placed on a double dissociation, in either case a one-dimensional or one-system account is typically rejected when a dissociation is confirmed by a significant interaction test.

However, there is a long history of literature showing that dissociations quantified by an interaction are problematic (e.g., Bogartz, 1976; Busemeyer & Jones, 1983; Dunn, 2003; Dunn & Kirsner, 1988; Henson, 2006; Poldrack, 2006; Loftus, 1978, 1996; Wixted, 1990). This is because they cannot compel the rejection of a one-dimensional explanation without making strong assumptions that are difficult, if not impossible, to directly test. For example, by definition a dissociation approach must assume selective influence of experimental factors on latent variables; that is, a
dissociation depends on an experimental factor affecting performance on one task but having no effect on performance for another. The latter of these requirements depends on confirming the null, which is impossible via standard interactions tests (Dunn, 2003).

Additional assumptions are also made about the form of the ‘response function’ mapping the latent variable to the response measure (Loftus, 1978). Linear analyses (e.g., ANOVA interaction tests) assume this response function is linear. However, for bounded response measures, which are commonly used in psychological research (e.g., accuracy calculated from binary data) the response function may be non-linear. Under such circumstances an observed interaction (or equally the failure to observe an interaction) may be scale dependent (e.g., confounded by floor and ceiling effects). Hence an apparent interaction can occur even when behaviour is mediated by only one underlying dimension, or an interaction can fail to occur when there is more than one dimension controlling behaviour. Even when an experiment is calibrated to avoid extreme performance, or the data is transformed in an effort to remove the bounds, confounding still cannot be ruled out without making further debatable assumptions (see Prince, Brown & Heathcote, 2012 – Chapter One – for further details).

An alternate, but relatively unknown, approach to addressing questions of latent dimensionality is able to avoid these and other caveats that plague dissociation methods. State-trace analysis (Bamber, 1979), which is also known as dimensional analysis (Loftus et al., 2004; see also Newell & Dunn, 2008, for a concise non-technical treatment), only makes the weak and arguably plausible assumption that the response function is monotonic; that is, that the latent variable and response measure consistently change in the same or in opposite directions. Latent dimensionality can then be easily assessed using a state-trace plot; graphing one type of dependent variable against another (a dependent variable state-trace analysis) or by graphing the same dependent
measure taken under different conditions or from different tasks or groups. Hence this approach is as general and flexible as dissociation analysis, and can be applied to “virtually any area of psychology” (Loftus, 2002, p.382). In all cases, it can be inferred that a single latent variable mediates performance if the state-trace plot is monotonic (i.e., is always increasing or always decreasing); otherwise performance must be controlled by more than one latent variable.

Originally developed in the late 70’s (Bamber, 1979), state-trace analysis has since been applied to a diverse range of topics and areas from basic research (e.g., short-term and long-term memory as well as perception) to more applied settings (e.g., legal, aging and developmental psychology) – an exhaustive summary is provided in Chapter One (Prince, Brown & Heathcote, 2012). State-trace evidence indicating that more than one latent dimension is at play can support inference about there being separate brain regions, cognitive representations or processes (e.g., is forgetting in short-term memory due to decay as well as interference processes; Oberauer & Lewandowsky, 2008). However, it can also indicate that a single representation or process has a multivariate structure and hence is characterised by more than one latent dimension. For example, single process models of recognition memory assuming recognition decisions are based on a single memory-strength dimension, can be either one-dimensional (e.g., equal variance signal detection theory, where memory strength is defined by a mean parameter) or can be multi-dimensional (e.g., unequal variance signal detection theory, where memory strength is defined by both a mean and variance parameter).

Unfortunately, state-trace analysis is unfamiliar to many researchers who could benefit from its use. The aim of this thesis (which is a collection of seven papers) is to make this technique more accessible to those researchers and also provide a specific application in the domain of recognition memory for faces. Please note that each of
these papers has been modified to include only non-redundant information (unedited versions of the published manuscripts are included in Appendix A).

**Overview of Section One: State-Trace Analysis**

The first section of this thesis is comprised of four papers:


Although state-trace analysis is as flexible and generally applicable as a dissociation approach, the experimental methodology for state-trace experiments does differ in its requirements from more familiar factorial designs. Chapter One (Prince, Brown & Heathcote, 2012 – *The design and analysis of state-trace experiments*), therefore, begins by providing an introduction to state-trace analysis and explains why this approach can support more certain inference about dimensionality, particularly for
bounded response measures. In an attempt to make this technique more accessible, we also provide guidance on the design and fine-tuning of state-trace experiments. In particular we demonstrate previously undocumented limitations of state-trace analysis, and suggest how they can be avoided using an iterative process of design refinement. In the later part of this chapter we also develop a new application of Klugkist, Kato and Hoijtink’s (2005) Bayesian encompassing prior method to the state-trace analysis of binary response data. The mathematical and computational details of this application are provided at the end of this chapter; however, the focus of Chapter One is to outline the rationale for the proposed analysis and illustrate how it can guide the fine-tuning of experimental designs as well as inference about monotonicity, and hence latent dimensionality.

A Bayesian approach is particularly suited to state-trace analysis for several reasons. First, the only additional assumption required is that the binary data are binomially distributed and so this approach detracts very little from the relatively assumption free nature of state-trace analysis. Second, Bayesian methods take into account that models of different dimensionalities vary greatly in their ability to fit data by chance and hence do not inappropriately favour more flexible models. Finally, in contrast to null hypothesis testing, Bayesian analyses can quantify evidence in favour of a simpler “null” (e.g., one-dimensional) model as well as evidence against it. This type of even-handed approach offered by Bayesian analysis has proven fruitful in many other areas of science, and stands in stark contrast to the focus on often inconsequential “significant” differences (i.e., Meehl’s, 1990, “crud factors”) encouraged by the widely acknowledged limitations of null-hypothesis statistical testing.

In addition to these theoretical advantages, we also empirically assessed the model recovery properties of Prince, Brown and Heathcote’s (2012) Bayesian methods
through a series of simulations. The results of these simulations are reported in Chapter Two (Hawkins et al., 2012 – *Designing state-trace experiments to assess the number of latent psychological variables underlying binary choices*). We first simulated an individual participant analysis, where results indicated that for large sample sizes (i.e., number of trials) the Bayesian methods perform well in identifying one-dimensional and multi-dimensional models. Second, we examined a method of aggregating participant results to select the best characterisation of the underlying dimensionality for a group of participants. Although aggregate results were mixed across simulated experimental designs, we use these findings to further provide guidance on designing state-trace experiments to maximise the probability of correctly classifying dimensionality.

Despite success in implementing Prince, Brown and Heathcote’s (2012) methods, these Bayesian procedures have the potential to narrow the focus of state-trace applications to only researchers who are familiar with the required sampling and estimation techniques. We, therefore, provide users with a software package, **StateTrace** (Prince, Hawkins, Love & Heathcote, 2012b) to aid the broader adoption of these methods. **StateTrace** is written for the freely available R language (R Development Core Team, 2011) and is designed for users with minimal experience implementing Bayesian analyses. In particular, we provide a guided user interface (GUI) for each of the functions contained in **StateTrace**, making the procedures more user-friendly for those researchers who are also unfamiliar with R. Each of the available GUIs provides users with a description of the arguments included in each function and allows argument values to be entered via widgets including text boxes, slider bars, true/false check boxes and multi-option lists. However, most functions require very minimal input in order to execute the analysis and still obtain meaningful results; the
other arguments then enable parameter values to be customised to the individual’s needs.

In Chapter Three (Prince, Hawkins, Love & Heathcote, 2012a – *An R package for state-trace analysis*) we describe the scope and capabilities of StateTrace. However, due to space limitations in the journal format, we were restricted in the level of direct instruction that could be provided in this paper. Hence we also provide a detailed step-by-step tutorial on how to analyse two example data sets in the ‘vignette’ document included in the StateTrace package. Although not in itself a peer reviewed publication, Chapter Four (Prince, Hawkins, Love & Heathcote, 2012b – *StateTrace: An R package for state-trace analysis*) presents this vignette, which was examined by reviewers of Chapter 3. It was developed based on our experience using the package with undergraduate research students.

**Overview of Section Two: The Differential Face Inversion Effect**

The second section of this thesis is comprised of three papers:


The aim of section is to use state-trace analysis to address a question that has long been of interest to experimental psychologists: Are faces special?

Faces are a rich and complex class of stimuli that have been the focus of many research areas including perceptual learning, object recognition and decision making as well as social cognition and applied areas such as eyewitness identification (Riesenhuber & Wolff, 2009). The common procedure used to assess the “special” status of faces has been a simple transformation of the stimuli: presenting the image upside-down. Traditionally the magnitude of the *inversion effect* (i.e., the decrement in performance due to inversion) for faces is compared to the inversion effect for a non-face control stimulus, and a dissociation is observed whereby the inversion effect is disproportionately stronger for faces: referred to as the *differential face inversion effect*. When confirmed by a significant stimulus type (faces vs. non-face) by orientation (upright vs. inverted) interaction, this dissociation is taken as evidence that upright faces are processed differently to inverted faces and to upright and inverted non-face items (Yin, 1969).

However, when Loftus et al. (2004) used state-trace analysis to re-examine this robust empirical effect they found that, in contrast to results from traditional linear analyses, unfamiliar (not previously seen) faces were not special relative to other classes of objects when inversion was only manipulated during the encoding stage of a recognition memory task. They, therefore, suggested that inversion only affects face processing when the face is being retrieved from memory. However, this was a surprising conclusion given the widely held belief that inversion affects the early perceptual stages of face processing (Friere, Lee & Symons, 2000). The second section of this thesis hence aimed to further test the evidence for Loftus et al.’s memory retrieval hypothesis. It should also be noted that the progression of experiments in this
section reflects our growing understanding of how to optimally calibrate an experimental design suited to state-trace analysis as well as the potential limitations of this technique, which ultimately helped to develop the guidance provided in Chapter One.

Our initial exploration of Loftus et al.’s (2004) memory retrieval hypothesis is reported in Chapter Five (Prince & Heathcote, 2009 – *State-trace analysis of the face inversion effect*). Here we sought to determine if Loftus et al.’s unexpected one-dimensional result could be due to caveats associated with their experimental methodology (e.g., use of unrealistic computer generated faces instead of realistic photographs) as well as design calibration for state-trace analysis. Nevertheless, even after addressing these limitations and refining the experimental design, we still replicated Loftus et al.’s one-dimensional encoding of unfamiliar faces.

In Chapter Six (Prince & Heathcote, 2010 – *The disproportionate face inversion effect in recognition memory*) we examined an alternate explanation of these one-dimensional results that was more compatible with the widely held perceptual view of the inversion effect: participants may be able to strategically use the face-specific dimension when they know it will benefit performance for all face trials. However, despite further refinement of the experimental methodology and more precise individual measurement, results were neither consistent with the memory retrieval hypothesis nor with the strategic explanation. In particular, we observed strong multi-dimensional evidence when inversion was only manipulated at initial encoding.

Given the inconclusive evidence presented in Chapter’s Five and Six, the final chapter of this thesis (Prince & Heathcote, 2013 – *Is the encoding of unfamiliar faces special?*) reports a more robust exploration of Loftus et al.’s (2004) memory retrieval hypothesis and the possibility that unfamiliar faces are not special. To test this memory
retrieval explanation, in addition to implementing Prince, Brown and Heathcote’s (2012) Bayesian state-trace methods, we also assessed whether manipulating stimulus inversion at retrieval alone (i.e., during the test phase of a recognition memory test) is sufficient to produce a disproportionately strong inversion effect for faces. Results showed that with more precise individual measurement and an experimental design fine-tuned for state-trace analysis, multi-dimensional state-trace evidence was observed when orientation was manipulated at (a) encoding alone, (b) retrieval alone and (c) both encoding and retrieval phases of a recognition memory task. Hence, we report strong support indicating that inversion does not only disrupt face processing when a face is retrieved from memory; that is, that the encoding of unfamiliar faces is special.
Section One: State-Trace Analysis

Included Papers:


Chapter One

The Design and Analysis of State Trace Experiments

Copyright © 2012 by the American Psychological Association. Adapted with permission. The official citation that should be used in referencing this material is:


No further reproduction or distribution is permitted without written permission from the American Psychological Association.
State-Trace Analysis: What is it and why use it?

Loftus (2002) provides a historical context for state-trace analysis, describing it as one of several equivalence based techniques for determining the rules by which different combinations of independent variables lead to equivalent states in one or more latent variables. He notes that such techniques can be applied to experimental data from “virtually any area of psychology” (p.382) and cites the colour-matching experiments that led to the tri-chromatic (i.e., three-dimensional) theory of colour vision as an exemplary application of equivalence methods to behavioural data.

Throughout this thesis, state-trace analysis is illustrated in a paradigm used by Loftus et al. (2004) to investigate whether cognitive representations of unfamiliar faces provide a basis for accurate responding in a recognition memory task that is not available to other non-face stimuli. However, a variety of examples demonstrate the importance of the question of dimensionality to many other areas of psychology, both theoretical and applied.

Applications of state-trace analysis have been particularly prominent in memory research. In the domain of short-term memory, Lewandowsky, Geiger, Morrell and Oberauer (2010) used state-trace analysis to determine whether a single latent variable can account for the effects of different types of distracters on the accuracy of short-term recall in complex-span tasks. In the domain of long-term recognition memory, Dunn (2004, 2008) investigated whether accuracy for responses classified by participants as being based on remembering or knowing (Tulving, 1985) are a function of different latent variables (see also Henson, 2006, for a discussion of related issues that arise with functional magnetic resonance imaging, fMRI, data). Brainerd, Wright, Reyna and Payne (2002) used state-trace analysis to investigate whether separate direct retrieval
and reconstruction processes jointly determine free recall and associative recall. These examples illustrate the relevance of state-trace analysis to key theoretical debates in memory research; such as whether forgetting in short-term memory is due to decay as well as interference (e.g., Oberauer & Lewandowsky, 2008), and whether memory has a single process or dual process architecture (e.g., Wixted, 2007).

Applications of state-trace analysis have not been limited to theoretical issues or effects on the accuracy of memory. For example, state-trace analysis has been applied to response time data to determine whether a single general-slowing factor can explain age-related differences in location-based and identity-based negative priming (Verhaeghen & De Meersman, 1998), selective and divided attention tasks (Verhaeghen & Cerella, 2002) and single vs. dual task performance (Verhaeghen, Steitz, Sliwinski & Cerella, 2003). Newell, Dunn and Kalish (2010) used a state-trace analysis of accuracy data to investigate dual (explicit and implicit) system theories of perceptual classification. In the domain of meta-cognitive skills, Jang and Nelson (2005) applied state-trace analysis to confidence ratings about recognition memory accuracy in order to investigate whether ratings made prospectively (i.e., during or shortly after study) and retrospectively (i.e., during testing occurring some time after study) have a common basis. Relevant to our main example, Loftus and Harley (2005) discussed the implications for the accuracy of eyewitness testimony of determining whether a common latent variable mediates the effects of priming, familiarity, viewing distance and spatial filtering on face recognition.

The foregoing examples explored differences in dimensionality as a function of tasks, stimuli and other experimental manipulations that can, at least in principle, be investigated at the individual participant level. State-trace analysis has also been used with both accuracy and response time measures in order to investigate differences
between groups of participants. Several examples underline the relevance of this approach to applied areas.

In the domain of clinical psychology, Haist, Shimamura and Squire (1992) used state-trace analysis to investigate whether the same declarative memory process mediates recall and recognition accuracy for both normal and amnesic participants. In the areas of problem solving and development, De Brauwer, Verguts and Fias (2006) used a state-trace analysis of response time data to investigate whether the representation of multiplication facts differs among children of different ages, and discussed the importance of the answer to this question for the design and evaluation of educational curricula. In the area of perceptual category learning, Newell, Dunn and Kalish (2011) showed through state-trace analysis that deficits displayed by Huntington’s and Parkinson’s disease patients can be explained by a deficit in a single underlying factor. Brainerd, Reyna and Howe (2009) used reversed associations (a technique closely related to state-trace analysis discussed further below) to investigate the role of dual memory processes in recall over the lifespan (early development, adult and aged) and in neurocognitive impairments. In a recent development with the potential for application to a wide range of disorders, Van den Broeck and Geudens (submitted) showed that state-trace analysis is better able to test for specific deficits in reading than traditional methods making comparisons to matched control groups.

In the examples cited so far state-trace analysis was applied to the same dependent variable measured under different experimental conditions, or for different types of tasks, stimuli or groups of participants (e.g., reading performance for clinical vs. control groups). A less common type of application, which we will call dependent-variable state-trace analysis, investigates whether different types of dependent variables are functions of a common underlying latent variable. For example, long-term
memory researchers have used state-trace analysis to investigate the relationship between retrospective confidence ratings and recognition accuracy (Busey, Tunnicliff, Loftus & Loftus, 2000; Heathcote, Freeman, Etherington, Tonkin & Bora, 2009), and between retrospective confidence ratings and judgments of the number of occasions on which an item was studied (Hintzman, 2004).

This type of dependent-variable state-trace analysis has also extended beyond recognition memory and behavioral measures. For example, Loftus and Irwin (1998) applied state-trace analysis to investigate whether a common perceptual process mediates different measures of visible and informational persistence. In the domain of neuroscience, Freeman, Dennis and Dunn (2010) applied state-trace analysis to investigate the relationship between the magnitudes of different evoked-response potential (ERP) components, as well as between the frequency of higher and lower confidence responses, in a recognition memory paradigm.

Taken together, these examples demonstrate the relevance of the question of dimensionality, and the applicability of state-trace analysis, to a variety of areas ranging from basic research in perception, attention, short-term and long-term memory, categorization, problem solving and meta-cognition to applications in aging, legal, clinical, educational, human factors and developmental psychology. These examples also make use of most of the dependent measures employed by psychologists and neuroscientists (i.e., accuracy, ratings, response time, ERP and fMRI). This chapter will focus on the sort of design used by Loftus et al. (2004); within-subjects designs with accuracy as the dependent variable. However, we also discuss implications for other types of state-trace analysis.
The State-Trace Plot

State-trace analysis can be explained with reference to a state-trace plot (see Bamber, 1979, for a formal treatment). In dependent-variable state-trace analysis, where the question of interest is whether two different dependent variables are both a function of the same latent variable, each axis of the state-trace plot corresponds to one of the dependent variables. For example, one axis might represent accuracy and the other confidence ratings (e.g., Busey et al., 2000; Heathcote et al., 2009). When the question of interest is whether measurements of the same dependent variable under different conditions (e.g., for different types of participants, stimuli, or levels of an experimental manipulation) are a function of the same latent variable, each axis represents the results for the dependent variable in one of the conditions. For example, the axes might represent recognition accuracy for face stimuli and non-face stimuli, such as houses (e.g., Loftus et al., 2004).

The axes of the state-trace plot can be thought of as defining a space within which the states of the system under examination can be depicted (i.e., a state space). We describe the two conditions or two different dependent variables constituting the axes of the state-trace plot as forming a state factor. A point on the state-trace plot is defined by two measurements, one for each level of the state factor. Each measurement is based on responses from multiple experimental trials. For example, suppose in a particular experimental condition with two-alternative forced choice testing the studied face was selected on 67% of test trials and the studied non-face item was selected on 58% of test trials. Results for this condition are represented as a single point, plotted at \{x = .67, y = .58\} in a state-trace plot where face accuracy (i.e., the proportion of correct responses) is represented on the x-axis and non-face accuracy on the y-axis. Although the state space may have more than two dimensions (e.g., when three or more
dependent variables or experimental conditions make up the levels of the state factor; we focus on the two-dimensional case as it has been most widely used in previous state-trace applications.

We illustrate state-trace analysis using data we collected in a recognition memory paradigm similar to that examined by Loftus et al. (2004). These experiments aimed to investigate whether faces are encoded on an extra dimension, commonly called a configural dimension (Maurer, Le Grand & Mondloch, 2002), not available to non-face stimuli. In particular, it was hypothesized that both face and non-face stimuli can be encoded in terms of their component features (i.e., on a featural dimension), but only faces can be encoded on a configural dimension (i.e., in terms of the relationships among features).

In the original experiment, and in our versions, houses were used as the non-face stimuli, because houses match faces on a range of characteristics, such as being mono-oriented (i.e., usually seen in one particular orientation), familiar and complex. The question addressed by state-trace analysis in this paradigm is whether measurements of memory accuracy for different types of items (houses and faces) arise from a single latent variable, sometimes called memory strength, or from two latent variables, which might be characterized as featural memory strength and configural memory strength.

In these experiments items were presented for study either upright or inverted. Inversion of a mono-oriented object usually results in decreased performance (the inversion effect; Rock, 1974). However, a number of lines of evidence indicate that inversion particularly impedes configural encoding of faces (Rakover, 2002). Hence, inversion potentially changes the number of encoding dimensions for face stimuli from two (when upright) to one (when inverted). Consistent with these findings, inversion reduces memory performance for face and non-face stimuli, but the reduction is greater
for faces (Valentine, 1988, Yin, 1969). We call the interaction between orientation and stimulus type the *differential face-inversion effect*. Figure 1.1a plots hypothetical data illustrating this effect; inversion causes a large decrease in performance for faces but only a small decrease for houses.

*Figure 1.1.* (a) Hypothetical results from a traditional dissociation experiment on the differential face-inversion effect (inversion has a large effect on performance for faces, but only a small effect for houses). (b) Hypothetical results from a corresponding state-trace experiment. This experiment takes the four conditions from the dissociation experiment (represented by the upper two points) and four new conditions, formed by manipulating the overall task difficulty (the lower two points). The state-trace plot in (b) graphs performance for house stimuli vs. performance for face stimuli, with lines (data traces) joining points from the upright condition and points in the inverted conditions. The dotted line shows that a monotonic (always-increasing) line can join all of the data points with small amount of misfit.
We call the second factor manipulated in Loftus et al.’s (2004) paradigm (i.e., inverted vs. upright study presentation) the dimension factor, as dimensionality is potentially influenced by its interaction with the state factor. The variation among points on a state-trace plot induced by manipulation of the dimension variable can be seen as providing a trace of the behaviour of the system in the corresponding state space. As with the state factor, the dimension factor may have more than two levels, but again we focus on the two-level case, as it is most common in previous state-trace applications.

It is important to note that in state-trace paradigms examining only one dependent variable it is sometimes possible to exchange the attribution of experimental manipulations to state and dimension factors with no material effect on the outcome of the analysis. For example, in Loftus et al.’s (2004) paradigm we could plot accuracy for upright stimuli against accuracy for inverted stimuli. However, in some cases, of which Loftus et al.’s paradigm is one, this exchange threatens validity of the state-trace analysis, and so there is a unique best attribution that should always be used. We examine this issue in the second major section of the chapter.

**Scale dependent interactions**

The differential face-inversion effect is usually tested by an interaction between the state (stimulus type: face vs. house) and dimension (orientation: upright vs. inverted) factors in a general linear model. When significant, such interactions are often labelled as dissociations and taken as support for a multi-dimensional process. A variety of types of dissociations have been enumerated (Shallice, 1988). For example, a single dissociation occurs when an experimental manipulation affects a dependent measure in one experimental condition but not in another condition. Figure 1.1a illustrates a single dissociation, where the face-house difference is apparent for inverted but not upright
stimuli. Less commonly, double dissociations are reported, where two experimental manipulations affect measures of each process in opposite directions.

Double dissociations are usually assumed to support stronger inference about dimensionality than single dissociations, particularly when they result in a crossover interaction. However, Dunn and Kirsner (1988) showed that both crossed and uncrossed double dissociations can occur when the experimental manipulations have opposite effects on a single latent variable. They pointed out that a single latent variable could only be ruled out if a third experimental manipulation affects the dependent measures in the other conditions in the same direction. This outcome, which they labelled a reversed association, is equivalent to observing a state-trace plot indicative of more than one latent variable.

In order to provide a concrete illustration of these issues, we focus on one reason for uncertainty about dimensionality inference common to all types of dissociation; *scale-dependent* interactions, with the most commonly recognized causes being floor or ceiling effects. Scale dependent interactions can occur when the function mapping latent to dependent variables (sometimes called the *response function*) is non-linear. Loftus (1978) details the effect of several types of response functions that account for bounds on dependent variables (e.g., an S shaped function accounting for the floor and ceiling in an accuracy measure, see Figure 1.2a). Figure 1.1a illustrates a scale-dependent interaction occurring because performance (i.e., accuracy) for both upright houses and faces is near ceiling.

One approach to floor and ceiling effects is to transform the dependent measure in a way that removes the bounds, such as a logit or probit transformation (i.e., inverse cumulative logistic or normal probability transformations) of a probability correct measure. Although easily implemented, this approach requires a strong assumption
about the exact mathematical form of the response function, and that the form of the function is the same for different levels of the state factor. Another approach is to manipulate the overall difficulty of a task to move the observed data to the middle of the range of the dependent variable. However, even where it is possible to avoid floor and ceiling effects in this manner, scale dependent interactions may still occur. This is illustrated by the response function shown in Figure 1.2b, which adds to floor and ceiling effects lower sensitivity to changes in the latent variable in a region that maps to the middle of the response range.

![Figure 1.2. Two hypothetical mappings (response functions) between a latent variable (on an arbitrary scale) and a response variable on a 0-1 scale (e.g., hit rate).](image)

**Monotonicity**

State-trace analysis avoids these problems because it tests an ordinal hypothesis that makes only a weak assumption about the response function: that it is monotonic. For example, monotonicity implies that an increase or decrease in the strength of a memory trace should lead to a corresponding increase or decrease in recognition
accuracy, without making any assumptions about the magnitudes of these changes, or to no change in recognition accuracy (e.g., when performance is at ceiling or floor respectively). The ordinal hypothesis is based on the fact that, if both state and dimension effects are mediated by a single latent variable mapped monotonically to the dependent variable, a plot of results for one level of the state factor against results for the other level of the state factor for each level of the dimension factor (i.e., a state-trace plot) must also be monotonic (Bamber, 1979).

Consequently, determining whether there is one or more than one mediating latent variable is accomplished by determining if the state-trace plot is monotonic. This determination can be accomplished graphically, as monotonicity implies that an always-increasing or always-decreasing line can join all points in the state-trace plot. Equivalently, monotonicity implies that points on one axis of the state-trace plot have the same order as the order of the points on the other axis (and so can be joined by an always-increasing line), or have the opposite order (and so can be joined by an always-decreasing line). Note that a state-trace plot must contain more than two points to be diagnostic of dimensionality, as these conditions are always true for two points. That is, the order of two points on one axis must always be the same as, or opposite to, the order of two points on the other axis.

The graphical approach to state-trace analysis has the weakness that it takes no account of measurement error: violations of monotonicity due to measurement error may be mistaken for evidence for more than one latent variable. Conversely, measurement error may cause observed monotonicity when there is mediation by more than one latent variable. Several statistical methods have been used to account for measurement error in the evaluation of state-trace monotonicity (see Newell & Dunn, 2008). In some cases these methods require further assumptions beyond those made by
state-trace analysis. In the third major section of this chapter we review some of these methods and propose a new method that aims to minimize added assumptions by taking the same ordinal approach as state-trace analysis.

**The trace factor**

One of the key methodological aspects differentiating state-trace (and reversed association) experiments from traditional dissociation experiments is the addition of a third factor, which we call a *trace factor*. In Loftus et al.’s (2004) design the trace factor was the amount of time spent studying each item. In designs with a two-level dimension factor a trace manipulation must be included to measure more than two points on the state-trace plot, and hence to enable inference about dimensionality. If the dimension factor has more than two levels a trace factor is not necessarily required, but it can be desirable for reasons we discuss in the next major section.

Figure 1.1b illustrates a state-trace plot for a design with a two-level trace factor. The figure was created by taking the data plotted in Figure 1.1a and adding data from new conditions that allow a shorter time for the study of each item (and, hence, result in poorer performance). The four pairs of coordinates defining the four points in Figure 1.1b correspond to results from eight experimental conditions created by a factorial crossing of the state factor (face vs. house stimuli), the dimension factor (inverted vs. upright orientation) and the trace factor (short vs. long study duration). Typically, lines are drawn on the state-trace plot to join results for conditions differing only on the trace factor in order to visually group them together. We will call such lines (e.g., the solid lines in Figure 1.1b) *data traces*.

The dotted line in Figure 1.1b is a monotonic curve that almost manages to connect all of the points in the state-trace plot. There is some error, because the curve does not quite pass through the lower right or upper left points (short-study-time upright
items and long-study-time inverted items, respectively). These imperfections illustrate the statistical problem we address in the third major section of this chapter: given that the data in Figure 1.1b contain some measurement error, should we conclude that a monotonic curve describes the data well, and hence that a single latent variable underlies memory for faces and for houses, or conclude that the monotonic curve misses the data, and hence that more than one latent variable is required?

Note that in some applications of state-trace analysis the levels of the trace factor correspond to individual participants (a random effect) rather than to experimental conditions (a fixed effect). For example, Haist et al. (1992) plotted the probability of recall against the probability of recognition (the state factor) for amnesic and control groups (the dimension factor), with each point on the plot representing results for one participant. Similarly, in De Brauwer et al. (2006) the state factor was response time for different types of multiplication problems, the dimension factor was age group and each point on the plot represented results for an individual participant (for similar applications see also Verhaeghen & Cerella, 2002; Verhaeghen & De Meersman, 1998, and Verhaeghen et al., 2003). Although we do not focus on such applications in this chapter, we do briefly address them in light of our findings in the General Discussion.

**State-Trace Experiment Design**

In this section we focus on the design of state-trace experiments where the trace-factor corresponds to a fixed effect. This makes the trace factor amenable to experimental manipulations that can be used to refine an experimental design to suit state-trace analysis. It is our experience that design refinements are often necessary to produce state-trace results that are clearly diagnostic of dimensionality. The
experiments whose results we report here were run for just such a purpose, in an attempt to refine Loftus et al.’s (2004) design.¹

**Diagnosing Dimensionality**

Even when a state-trace plot contains more than two points and there is no measurement noise it still may not be diagnostic of dimensionality, as we illustrate in Figure 1.3. The figure contains state-trace plots of hypothetical data from the same design as illustrated by Figure 1.1b (i.e., a design in which state, dimension and trace factors all have two levels). The points $a$ and $b$ are from the first level of the dimensional factor (e.g., inverted study items) for both levels of the trace factor (e.g., shorter and longer study time respectively), and their data trace is a thick solid line. Points $A$ and $B$ are from the second level of the dimension factor (e.g., upright study items), for both levels of the trace factor (e.g., shorter and longer study times respectively), and their data trace is a thick dashed line. In discussing this figure we assume that there is no measurement error, and so the data points fall exactly on their true values. We return to the issue of measurement error in the next major section.

¹ Although not included in the published paper, Appendix B of this thesis reports the methods and results of these pilot experiments.
Figure 1.3. State-trace plots showing illustrative data patterns from a $2 \times 2 \times 2$ design. Letters indicate data points. Thick lines are data traces, which join groups of data points that differ only on the trace factor. Thin dotted lines show the state-trace of the underlying system; panels in the upper row show a one-dimensional system and panels in the lower row show a multi-dimensional system.
If we were able to systematically vary all latent variables across all of their ranges, we would observe all possible values of the dependent variables corresponding to each axis of the state-trace plot. If this entire set of observations were represented on the state-trace plot and measured without error, the result would be the true state-trace of the system. Under very general assumptions, any data-generating process mediated by just one latent variable will always have a monotonic true state-trace. That is, the set of points created by moving the underlying latent variable across its whole range will make a curve in the state-trace plot that always increases or always decreases. For example, the thin dotted lines in Figures 1.3a and 1.3b show monotonic true state-traces. In Figures 1.3a and 1.3b, the data points are also monotonic, having the same order on both axes: in Figure 1.3a this order is \{a, b, A, B\}, and in Figure 1.3b it is \{a, A, b, B\}.

Note that in Figure 1.3b a monotonic line joining all data points does not follow the data traces. In terms of our recognition memory example, the monotonic line joins the short-study-time inverted data, followed by the short-study-time upright data, long-study-time inverted data, and then the long-study-time upright data. This illustrates that the data traces (i.e., the solid and dashed lines) are only a graphical convenience and do not necessarily reflect the true state-trace.

Note that neither the true state-trace of a one-dimensional system nor the curve joining points with the same order on both axes need not be straight, or have constant curvature, or satisfy any other condition related to smoothness. Such conditions may be important due to theoretical considerations outside the scope of state-trace analysis. However, state-trace analysis itself makes no such assumptions; it is based purely on monotonicity.

If the data points in a state-trace plot are non-monotonic a one-dimensional system cannot have produced them. However, the converse does not necessarily hold; if
the data points in a state-trace plot are monotonic this does not necessarily imply that a one-dimensional system generated them, even if there is no measurement noise. As illustrated by Figure 1.3, one further condition is required: overlap of the data traces on at least one axis.

Figures 1.3c and 1.3d show results for a two-dimensional system, illustrated by two true state-traces (thin dotted lines). The first true state-trace applies for the first level of the dimension factor (e.g., inverted stimuli) and the second applies for the second level of the dimension factor (e.g., upright stimuli). The data points in Figure 1.3d are non-monotonic; the order of points on the x-axis, \{A,a,B,b\} differs from the order on the y-axis, \{a,A,b,B\}. In contrast, the data points in Figure 1.3c are monotonic; they have the same order \{a,b,A,B\} on both axes, despite the fact that the data in this figure come from a two-dimensional data generating process.

In fact, the data points in Figure 1.3c are identical to those in Figure 1.3a – just the interpretation is different. In Figure 1.3a, we interpreted the data points as belonging to a single monotonic true state-trace (supporting a one-dimensional system), but Figure 1.3c shows they can equally belong to two true state-traces, and so are consistent with a multi-dimensional system. Figures 1.3a and 1.3c illustrate that when data traces fail to overlap, state-trace analysis fails to diagnose dimensionality.

Such failures can usually be remedied by changing the levels of the trace factor. For example, suppose that the trace factor in Figure 1.3 was the duration for which items are studied, as it was in our experiments. Because increased study duration increases accuracy, the accuracy for point \(b\) in Figure 1.3a can be increased by increasing study duration, and the accuracy of point \(A\) can be decreased using a shorter study duration. This leads to the diagnostic state-trace plot shown in Figure 1.3b, in which the data traces overlap on both axes. Even though overlap on just one axis is
sufficient, overlap on both is desirable as it increases the number of data points that can contribute to the detection of violations of monotonicity.

In Figure 1.3d the non-diagnostic result in Figure 1.3c was rectified by decreasing performance for both conditions within the second level of the dimension factor and increasing performance for both conditions within the first level of the dimension factor. Once again the resulting state-trace plot is strongly diagnostic of dimensionality, with data traces that overlap on both axes, and in this case it clearly indicates more than one dimension. If the trace factor were study duration in these examples, this result could be achieved by using longer study durations for level one than level two of the dimension factor (i.e., longer study for inverted than upright items). Note that there is no requirement to use a fully factorial design in state-trace analysis. Indeed, as we discuss below, a fully factorial design can be very inefficient.

**Choosing a Trace Factor**

Much of the art of running a successful state-trace experiment is in choosing and calibrating an appropriate trace factor. We consider three design issues related to the trace factor. First, if non-monotonicity in a state-trace plot is to be unambiguously attributed to the interaction of the state and dimension factors, the trace factor must have a monotonic effect within each level of the dimension factor. In our recognition memory example this means that increased study time must lead to increased recognition accuracy in all four conditions. Other state-trace analyses of memory have used retention interval as the trace factor (Haist et al., 1992) or study-item repetitions (Bamber, 1979; Hintzman, 2004; Jang & Nelson, 2005), which are also assumed to have a monotonic effect on accuracy. The statistical procedures that we develop in the next major section provide a method of testing whether the trace factor effect is monotonic.
The second design issue requires trace levels to be chosen so they maximize overlap. Commonly, the dimension factor affects results for each state in the same direction. In such cases, different trace-factor levels can be used within each level of the dimension factor to compensate for the effect of the dimension manipulation, and hence to maximize overlap. For example, in our experiments inverted items were harder to remember than upright items. Hence, it makes sense to give generally longer study durations to the inverted items, so that accuracy on the inverted and the upright items becomes about equal, and so data traces overlap.

Thirdly, the effects of potential interactions between the trace factor and other factors must be considered. For example, it has been suggested that inversion and very brief exposure have similar effects (Valentine, 1988), so that upright faces may be encoded only in terms of features (i.e., in a one-dimensional manner) below some minimum study duration. If upright and inverted data traces overlap only when study duration for the upright condition is below this minimum, the state-trace plot will be monotonic, even when face processing is multi-dimensional at longer durations. This issue may account for Loftus et al.’s (2004) finding in their first experiment of a monotonic state-trace plot, as only the shortest study duration (17ms) for upright items overlapped with the longest duration (250ms) for inverted items. Our alterations to Loftus et al.’s design aimed to address this issue by increasing the minimum study duration for upright items.

**Refining a State-Trace Experiment: A Case Study**

As previously discussed, a good state-trace design produces data traces that overlap on one, and ideally both, axes. When the dimension factor causes a large performance difference, using the same trace levels within each dimension level is
inefficient, as only data in the overlapping region contributes evidence about
dimensionality. In such cases it is better to use different levels of the trace factor for
each dimension level (i.e., a non-factorial design).

For example, the design used by Loftus et al. (2004) in their first experiment
produced a very strong dimension effect – items studied upright were much more
accurately recognized than those studied inverted. In order to achieve data trace overlap,
Loftus et al. used six study durations (from 17ms to 250ms) in a fully factorial design
with the state and dimension factors. Although this range of study durations was
sufficient to cause overlap for both houses and faces, the overlap was minimal.

We largely replicated this design with our first experiment, which we will call
the *upright-test* experiment. However, in contrast to Loftus et al. (2004), we used longer
study durations for inverted (256-2048ms) than upright (33-256ms) items. As shown in
Figure 1.4a (see caption for further experimental details), the longer study times for
inverted items overcame the performance decrement caused by inversion, and so
produced almost perfect data-trace overlap. These durations were selected based on an
earlier pilot study that was not so successful, illustrating the process of iterative
refinement of a design.
Figure 1.4. State-trace plots of accuracy as measured by hit rate minus false alarm rate (HR – FAR) for each participant then averaged over participants, for the (a) upright-test (18 participants), (b) inverted-test (16 participants) and (c) matched-test (14 participants) experiments. Each participant’s FAR was based on 128 test trials and each HR on 32 test trials. Data points are plotted with least significant ($p = .05$) difference bars: ± ($SE \times 1.96 \times \sqrt{2} ) / 2$, (Saville, 2003), where SE is a within-subject standard error (Loftus & Masson, 1994). Spearman’s rank based measure of association ($\rho$) for each condition is given above its panel. The lines are *data-traces*, which join groups of data points that differ only on the trace factor.
Selection of an appropriate trace factor is subject to trade-offs enforced by restrictions on the range and number of trace levels. If too many levels are used it is practically difficult to get enough trials at each level for precise measurement. For this reason we used four trace levels rather than Loftus et al.’s (2004) six. However, if too few levels are used it becomes difficult to detect non-monotonicity. For example, Figure 1.4c shows a state-trace plot from our third experiment (described further below) that is clearly non-monotonic with the four points per data trace that were measured. In contrast, if only the lowest and highest points in each data trace had been measured we might have come to a quite different conclusion, because a monotonic line can join these four extreme points.

Although it is difficult to specify how many trace levels are required in general, the example in Figure 1.4c suggests a minimum of three. Once the number is chosen, the spacing of the levels must be selected. The optimal spacing depends on the nature of any non-monotonicity that is present, but as this is not known in advance we recommend an iterative strategy, starting from a spacing that results in an approximately equal change in performance between each level. For example, Loftus et al. (2004) found that accuracy increased linearly with the logarithm of study duration, so we selected durations that increased by a constant multiple in our experiment, resulting in the relatively even spacing displayed in Figure 1.4. Unequal changes in performance between data points are undesirable because very closely spaced points generally make a smaller contribution to exploring the behaviour of the underlying system than more widely spaced points. Closer spacing also makes observing non-monotonicity within a data trace due to sampling error more likely.
Reducing the dimension effect

An alternative approach to maximizing data-trace overlap is to reduce the magnitude of the dimension factor effect. Confounding the dimension manipulation with another manipulation that has the opposite effect can do this. In fact, the reason that Loftus et al. (2004) obtained such a large difference between upright and inverted stimuli was that they confounded their inversion manipulation with the encoding-specificity effect (Tulving & Thomson, 1973), but in a way that increased rather than reduced the dimension effect. Encoding-specificity is a well-known and robust effect (Nilsson & Gardiner, 1993) whereby memory for a stimulus (including faces, Rakover & Teucher, 1997; Yin, 1969) improves when there is a better match between its study and test encodings. Loftus et al. had participants study stimuli either upright or inverted, but always tested them upright. Hence, the upright condition was advantaged even more than usual, because study and test encodings had a better match than in the inverted condition.

Loftus et al. (2004) applied inversion at study but not at test to investigate the hypothesis that evidence for a multi-dimensional face representation emerges only when inversion affects the processing of faces that are already stored in memory. As the faces they used in their first experiment were unfamiliar, they were not in memory before study, and so their hypothesis predicts a monotonic state-trace plot for this design. Arguably the same prediction should hold whether all images are tested upright or all are tested inverted, so in our second experiment, which we call the inverted-test experiment, we tested all stimuli inverted. This reverses the confounding effect of encoding specificity, and so should reduce the difference in accuracy between upright and inverted stimuli.
We took advantage of the reduced inversion effect in the inverted-test experiment by using longer study durations for the upright condition (67-512ms) in order to avoid the potentially confounding effects of very brief study durations discussed earlier. We also kept the same study durations for the inverted condition (267-2048ms) as in our upright-test experiment. However, as shown by the state-trace plot of data from this experiment (Figure 1.4b), our choice of upright durations under-estimated the power of the encoding specificity effect relative to the inversion effect. With the durations we used, performance was actually better in the inverted than upright condition. These results suggested that the inversion and encoding specificity effects are about equal in magnitude in our design, and so the best overlap would have been obtained using equal durations for the upright and inverted conditions. Prince and Heathcote (2009 – Chapter Five of this thesis) used this design and obtained greatly improved data-trace overlap, again illustrating the process of iterative design refinement.

**Accuracy measures and multiple baselines**

Within each level of the state factor in the upright-test and inverted-test experiments, accuracy was measured relative to a common baseline condition for all levels of the trace and dimension factors. That is, accuracy for houses was measured relative to a false alarm rate (FAR; i.e., the probability of wrongly saying a test item was studied) specific to houses, and accuracy for faces was measured relative to the FAR for faces.

In Figure 1.4 we used HR – FAR as the accuracy measure, where HR is the hit rate (i.e., probability of correctly saying a test item was studied). When there is a common baseline the order of accuracy results will be the same for any reasonable measure, such as \((HR – FAR) / (1 – FAR)\), which is similar to the measure used by
Loftus et al. (2004), or the $d'$ measure used in signal detection theory (i.e., $z(\text{HR}) - z(\text{FAR})$, Macmillan & Creelman, 2005, where $z()$ is a probit transformation). Note that all of these measures are monotonically related. It is a notable strength of state-trace analysis that the dimensionality indicated by a state-trace plot can be unaffected by using different monotonically related accuracy measures.

However, state-trace analysis may not be invariant in this way for a design where accuracy is assessed against different baselines for different dimension factor levels. Our third experiment, with results shown in Figure 1.4c, is a case in point. As has been standard in the differential face-inversion effect literature since its inception (Yin, 1969), study and test orientations were matched in this experiment. That is, in this matched-test experiment, an item that was studied upright was tested upright and an item that was studied inverted was tested inverted. As the difference between upright and inverted stimuli is clear to participants, they might decide to use different criteria to make recognition decisions for upright and inverted test items.

The usual method of addressing such potential differences in response bias is to measure accuracy relative to separate baselines for upright and inverted conditions. That is, some test items that were not previously studied are presented inverted and some upright, and their separate false alarm rates are used to calculate accuracy for inverted and upright conditions respectively. However, different baselines for the dimension (or trace) factors mean that state-trace inference may no longer be invariant across different accuracy measures. In the matched-test experiment, for example, this can occur because the order of results among upright and inverted conditions can depend on the scale on which accuracy is measured.

As an existence proof, suppose $\text{FAR} = 0.2$ and $\text{HR} = 0.85$ for upright items and $\text{FAR} = 0.35$ and $\text{HR} = 0.95$ for inverted items. If we measure performance using $\text{HR} -$
FAR, upright performance (0.65) is greater than inverted performance (0.6), but the order is reversed in $d'$ (1.88 vs. 2.03 for upright and inverted respectively) and $(HR – FAR) / (1 – FAR)$ (0.81 vs. 0.92). If the two baselines happen not to differ empirically this problem might not arise. Our third experiment was designed to check if this state of affairs held in the matched-test design; unfortunately, it did not, so we do not consider this experiment further. Note that a baseline that collapses over upright and inverted false-alarm rates does not address the problem, as a criterion shift would still affect hit rates.

Differing baselines for each state factor level (e.g., for houses and faces) do not compromise a state-trace analysis of accuracy. This is because the order of accuracy results within each state-factor level is unaffected by the baseline, and so monotonicity, which depends only on the order of conditions, is unaffected. Hence, when only one of the state and dimension factors has a common baseline, the factor with the common baseline should always be used as the dimension factor. Alternatively a different type of test that does not require a baseline, such as a two-alternative forced-choice test, can be used. The analysis methods we develop in the third major section of this chapter work with both two-alternative forced choice and single item testing.

Summary and Recommendations for Experiment Design

In a state-trace experiment the trace factor is typically of lesser theoretical interest, as it is included in the design only to sweep out different levels of performance. However, it has an important role to play in obtaining overlapping data traces. Overlapping data traces are essential for enabling a state-trace plot to be diagnostic of dimensionality. Hence, a successful state-trace experiment requires careful selection and calibration of the trace factor. Often, the most efficient way of ensuring data-trace
overlap is to use different levels of the trace factor between levels of the dimension factor in order to counteract the effect of the dimension factor.

The number and spacing of the levels of the trace factor must also be calibrated in order to best detect any non-monotonicity in the state-trace plot. At least three trace levels per dimension level, with each set of trace levels having approximately evenly spaced effects on performance, are recommended. In order to unambiguously interpret any non-monotonicity (and hence evidence of multi-dimensionality) as a result of the interaction of state and dimension factors it is also desirable that the trace factor is known to have, or can be shown to have, a monotonic effect on performance.

**Statistical Analysis of State-Trace Data**

In this section we propose a new method for analyzing state-trace plots of accuracy data, using Klugkist, Kato and Hoijtink’s (2005) Bayes-factor method of selecting among models defined by inequalities. Technical details are provided at the end of this chapter. We also provide an example of how the method can be applied, using data from our upright-test and inverted-test experiments. First, however, we motivate our new approach by discussing the potential shortcomings of some existing methods.

**Existing Statistical Methods**

Perhaps the most common method for state-trace analysis is visual inspection of a state-trace plot based on data averaged over participants. Measurement error is quantified by confidence intervals, with small intervals, and hence a clear conclusion, obtained by averaging over a large number of participants. For example, Loftus et al., (2004) averaged over 366 participants in their first experiment. They also quantified
evidence for monotonicity using Spearman’s Rho (ρ), a rank based measure of association (monotonicity implies ρ = 1). They showed that a criterion of ρ = 1 reliably identifies a one-dimensional system when results are averaged over participants simulated by varying the system’s parameters. Their method results in state-trace plots that are monotonic for every simulated participant.

However, when we repeated their analysis with added measurement error, which creates more realistic data where some participant’s simulated state-trace plots are non-monotonic, this test became unreliable. A value of ρ = 1 rarely occurred, and even when an optimal ρ criterion was used, the monotonic vs. non-monotonic classification was unreliable unless measurement error was small. Moreover, even when measurement error was small, optimal criteria on ρ varied substantially across the simulations, indicating that this test might be difficult to apply in practice.

Figure 1.4 exemplifies the type of analysis used by Loftus et al. (2004), as applied to our three experiments. To facilitate congruency between inference-by-eye and conventional standards for null-hypothesis statistical testing (NHST), we added to Figure 1.4 least-significant difference intervals (see figure caption for details), so that non-overlapping bars indicate a significant difference at the .05 level. Based on this method, a one-dimensional system cannot be rejected for the upright-test (Figure 1.4a) and inverted-test (Figure 1.4b) experiments. However, the trends in the upright-test experiment suggest non-monotonicity, so perhaps the non-significant results are attributable to a small sample size (see figure caption), and hence a lack of power. In the matched-test experiment (Figure 1.4c) this method supports a multi-dimensional system, although, as we have noted, interpretation of accuracy results for this experiment are problematic. All of these conclusions are consistent with the Spearman’s
\( \rho \) values in the figure, which are relatively close to one for the upright-test and inverted-test experiments and substantially less than one for the matched-test experiment.

**Averaging and monotonicity**

Unfortunately, the averaging that underpins the reduction in measurement error necessary for the preceding types of analyses can be problematic. First, averaging can produce results that depend on the type of accuracy measure used. More fundamentally, neither monotonicity nor non-monotonicity is necessarily preserved under averaging. For example, Figure 1.5, which uses the same design and notation as Figure 1.3, shows hypothetical data from two participants (left and middle columns) and the average over those two participants (right column). The first row shows that, even when both participants have non-monotonic state-trace plots, the average plot can be monotonic. The second row shows the opposite; both participants are monotonic but the average is non-monotonic (see Newell et al., 2010, for an example of this phenomenon). In all cases the individual participant data are plausible – data traces overlap on at least one axis and an increase in the trace factor within each level of the dimension factor causes an increase on both axes. Similarly plausible examples can be constructed where a mixture of monotonic and non-monotonic participants leads to either a monotonic or non-monotonic average.

These examples do not prove that a state-trace analysis performed on average data is always, or even very often, misleading. However, they do show that averaging has the potential to be misleading. In our experiments, where all factors are within-subjects, a complete state-trace analysis of each individual participant’s data provides a possible solution. Results of individual analyses can then be safely aggregated, except where there is evidence for strong individual differences, such as individual participants with outlying results or larger sub-groups with different dimensionality. Eliminating
outlying participants from aggregates or aggregating over only sub-groups of participants can address such problems. In the following Example Analysis section we present results for individual participants using figures that make it easy to detect such issues.

Figure 1.5. State-trace plots illustrating averaging distortions. Letters indicate data points and the lines are data-traces, which join groups of data points that differ only on the trace factor. The dimension factor has two levels, with data from one level (a and b) joined by a solid data trace and data from the other level (A and B) joined by a dashed data trace.
Although individual analysis avoids distortion due to averaging, it can introduce other issues. First, it cannot be used for state-trace designs with between-subjects factors. Second, higher levels of measurement noise usually attend individual analysis, necessitating more observations per participant to get precise estimates. Finally, within the NHST framework, a non-significant test result due to a lack of power can be confused with support for a null hypothesis. Although this confusion must be avoided in all contexts, it is particularly salient for individual analysis, due to higher levels of measurement noise. For NHST analyses of state_TRACE data, where a one-dimensional model is the null model, this confusion leads to a bias against a multi-dimensional finding.

**Null hypothesis statistical testing**

A range of model-selection techniques that have been applied to average state-trace data can also be applied to individual data. Newell and Dunn (2008) used monotonic regression (Barlow, Bartholomew, Bremner, & Brunk, 1972), with goodness-of-fit tests based on bootstrap estimates of null-hypothesis distributions (McLachlan & Peel, 2000). Other NHST approaches, such as comparing maximum likelihood fits amongst order-constrained models, can also be applied to individual data. For example, a particular one-dimensional model (i.e., a model that enforces a particular order for all points on both axes) could be used as the null, and its fit compared to a model without order constraints, or to models that enforce only the order dictated by the trace factor within each data trace.

We do not pursue the NHST approach here because it is not clear how it can take account of differences in model complexity. The class of multi-dimensional models is clearly much more flexible than the class of one-dimensional models, as it has fewer order (inequality) restrictions, and so is likely to fit data better by chance. Methods of
accounting for model complexity due to differences in number of parameters (see Pitt & Myung, 2002) cannot be applied, as one-dimensional and multi-dimensional models have the same number of parameters. As Klugkist, Laudy and Hoijtink (2005) point out: “For inequality constrained hypotheses the number of parameters does not reflect the complexity or size of the parameter space of the model” (p. 478).

A second reason that we did not pursue an NHST approach is that we believe it is inherently unsuited to state-trace analysis. Loftus (2002) suggested that state-trace analysis should be used to determine the simplest explanation of a phenomenon. Hence, state-trace analysis needs a statistical method that provides estimates of the probability that the simpler model (e.g., a one-dimensional model) is sufficient, rather than focusing only on estimating the probability that a more complex model (e.g., a multi-dimensional model) is required. NHST casts the simpler model in the role of the null hypothesis. A significant result can provide evidence against the null model. However, NHST cannot provide evidence for the null model, as a failure to reject the null hypothesis might either be due to the null being true or to a lack of power, which is particularly likely in individual participant analysis (see Wagenmakers, 2007, for other problems with NHST). Gallistel (2009) summarizes these and related issues and suggests that Bayesian analysis provides a potential method for solving both problems – confirming the null and addressing model complexity.

**Proposed Method**

We propose a method of state-trace analysis based on quantifying evidence for different models of orderings of results over experimental conditions. The models assume independent binomial distributions for the binary choice data within each condition, and all have the same number of parameters. That is, there is one binomial
probability parameter for each condition. For our upright-test and inverted-test experiments with four levels of study duration, for example, there are 18 parameters, corresponding to the $2 \times 2 \times 4 = 16$ hit rates in the conditions testing studied items and two false-alarm rates for conditions testing unstudied items (i.e., false-alarm rates for faces and houses).

The models differ in terms of inequality constraints (i.e., orders) placed on the set of binomial probability parameters ($\theta$). In our application, where each state has a separate baseline condition, $\theta = \{\theta^F_i, \theta^H_{i,j,k}\}$, where the superscript indicates a false-alarm (F) or hit (H) rate parameter, and the subscripts indicate the state ($i = 1, 2$), dimension ($j = 1, 2$) and trace ($k = 1…4$) factor levels. Although the labelling of the state and dimension factor levels is arbitrary, the trace factor levels are assumed to be labelled in the order predicted to produce increasingly accurate performance (e.g., in order of increasing study duration). In a design without baselines (e.g., two-alternative-forced choice testing), $\theta = \theta^H_{i,j,k}$.

We examine four ordinal models that are diagnostic either of dimensionality or of the need for design refinement. The first diagnostic model, the non-trace (NT) model, is defined as violating the order dictated by the trace factor. In our application, for example, the order predicted by the trace factor (study duration) is $\theta^F_i < \theta^H_{i,j,1} < \theta^H_{i,j,2} < \theta^H_{i,j,3} < \theta^H_{i,j,4}$. The non-trace model violates this order for one or more of the state and dimension conditions. Evidence for the non-trace model suggests that the trace factor manipulation did not have a monotonic effect, and so the trace factor manipulation may be in need of revision. The same logic applies to designs with no baseline, except the constraint involving $\theta^F_i$ is dropped.

The remaining three diagnostic models are special cases of the complement of the non-trace model, which we call the trace (T) model (i.e., a model in which trace-
factor order applies for all state and dimension factor levels). As the trace model implies that \( \theta_F^i < \theta_H^{i,1,1} \) and \( \theta_F^i < \theta_H^{i,2,1} \) we need only consider constraints on the order of the hit-rate parameters for the remaining three diagnostic models. Hence, designs with and without baselines for each state can be treated the same. The second diagnostic model, the multi-dimensional (MD) model, is defined as having a different order for the parameters associated with each state-factor level. Evidence for this model suggests a system with more than one latent variable.

The final two diagnostic models are special cases of the complement of the multi-dimensional model within the trace model, which we call the monotonic (M) model (i.e., a model in which parameters have the same order within each state-factor level). In particular, the third diagnostic model, which we call the no-overlap (NO) model, is defined as having one of the two orders indicating that there is no overlap between data traces (e.g., \( \theta_H^{i,1,4} < \theta_H^{i,2,1} \) or \( \theta_H^{i,2,4} < \theta_H^{i,1,1} \)). Evidence for this model suggests that monotonicity is not diagnostic of dimensionality, and hence the need for a refinement in the experimental design to increase overlap. The fourth diagnostic model, which we call the uni-dimensional (UD) model, is the complement of the no-overlap model within the monotonic model, so is defined as having any of the remaining monotonic orders. Evidence for this model suggests a system with one latent variable.

The union of the four diagnostic models contains all possible parameter orders. Each model differs tremendously in its complexity, and hence in its ability to fit data by chance. One way of quantifying the complexity of ordinal models is by counting the number of orders they contain. A model with no order constraints on its \( p \) parameters, which we call the unrestricted (U) model, contains \( p! \) orders. For example, consider a simple design with no baseline and two levels for the state, dimension and trace factors, in which \( p = 8 \). There are 40,320 possible orders in the unrestricted model, but only 36
of these satisfy the constraints of the trace model. Of those 36, only four satisfy the uni-dimensional model, 30 satisfy the multi-dimensional model, and the remaining two satisfy the no-overlap model. In this case the non-trace model is more complex than the multi-dimensional model by a factor of greater than 1000. The multi-dimensional model is 7.5 times more complex than the uni-dimensional model, which in turn is twice as complex as the no-overlap model. These factors increase and diverge for more complicated designs (see final section of this chapter), underlining the need to take account of model complexity.

**Bayes-factor model selection**

A Bayes factor (BF; Kass & Raftery, 1995) is the ratio of the marginal probability of the observed data given one model \( M_i \) divided by the marginal probability of the observed data given another model \( M_k \):

\[
BF_{i,k} = \frac{m(D | M_i)}{m(D | M_k)}. 
\]

The marginal probability equals the likelihood of the data given a model with parameters \( \theta \), \( f(D | M, \theta) \), integrated over the prior distribution of the parameters, \( p(\theta | M) : m(D | M) = \int f(D | M, \theta)p(\theta | M)d(\theta) \). We use Bayes factors to select among the four diagnostic models because, as Myung, Karabatsos and Iverson (2008) state: “Bayes factor-based model selection automatically adjusts for model complexity” (p.314). The aim of Bayes factor model selection is to find the model with the highest posterior model probability. This is the probability, given observed data \( D \), that one model \( M_i \) amongst a set of two or more models, is the true model: \( p( M_i | D) \).
For a set of models $M_i$, $i = 1 \ldots m$, that are assumed to have an equal prior probability of being the true model\(^2\), the posterior model probability for $M_i$ is calculated using each model’s Bayes factor all relative to the same model, $M_k$:

$$p(M_i | D) = \frac{BF_{i,k}}{\sum_{j=1}^{m} BF_{i,k}}.$$ Note that, as the marginal probability for $M_k$ cancels in this ratio, the posterior model probability is the same for different choices of $M_k$. Note also that, unlike NHST probabilities, both high and low posterior model probabilities are interpretable, as evidence for and against a model respectively.

Bayes factors can be difficult to estimate because the integration required to calculate marginal probabilities is over a high-dimensional space (e.g., 18 dimensions for our experiments). However, when the models being compared have the same parameters, differing only in that one ($M_i$) is an order-constrained special case of the other ($M_k$), Klugkist, Kato and Hoijtink (2005) showed that the Bayes factor can be estimated based only on Monte-Carlo (MC) methods that generate random samples from the prior and posterior of the less restricted or encompassing model (see Andrieu, de Freites, Doucet & Jordan, 2003, for an overview of methods used to obtain MC samples). All that is required are estimates of the proportion of the prior ($\hat{\pi}$) and posterior ($\hat{\Gamma}$) MC samples from the encompassing model (i.e., $M_k$) that obey the order constraints dictated by the more restricted model (i.e., $M_i$). The Bayes factor is approximated by the ratio of these two proportions: $BF_{i,k} \approx \hat{\Gamma}/\hat{\pi}$.

This method of estimating Bayes factors is made both conceptually and computationally simpler by assuming independent uniform priors for each parameter in the unrestricted model. Such a uniform prior corresponds to the assumption that, before

---

\(^2\) This assumption does not imply that differences in model complexity (i.e., in the ability of one model to fit data generated by other models) are not taken into account. An increase in the prior probability of a model being true (e.g., based on theoretical grounds or previous findings) causes an increase in its posterior probability, whereas an increase in the probability that a model fits the data by chance (given the prior) causes a reduction in its posterior model probability, all other things being equal.
the data are observed, all orders are equally likely. Other priors could be used, but as long as they allow some non-negligible probability for all orders, they will produce the same Bayes-factor estimates (Klugkist, Kato & Hoijtink, 2005) making results insensitive to the choice of prior (see Liu & Aitkin, 2008, for a discussion of the problem of prior sensitivity in other applications of Bayesian methods).

A uniform prior has the computational advantage that prior proportions can be computed analytically (see final section of this chapter), avoiding the computational expense of using MC samples. A second computational advantage follows from the fact that a uniform distribution is a special case of the beta distribution, which is the conjugate prior for the binomial distribution. This implies that the posterior also has a beta distribution (see final section for details). Fast algorithms are available in most statistical packages for obtaining independent MC samples from beta distributions, minimizing the computational expense of estimating posterior proportions based on MC samples from the posterior distribution of the unrestricted model.

Figure 1.6 illustrates the procedure by which we estimate posterior proportions for the four diagnostic models. First, MC samples are generated from the posterior of the unrestricted model (represented by a unit rectangle in Figure 1.6a). A count is made of the number that violate the trace-model order, enabling estimation of the posterior proportion \( \hat{\Pi}_{NT,U} \), and hence the Bayes-factor \( BF_{NT,U} = \hat{\Pi}_{NT,U} / \pi_{NT,U} \), for the non-trace model relative to the unrestricted model (note that by definition \( \pi_{NT,U} = \Pi_{NT,U} = BF_{U,U} = 1 \)). In theory, estimates of the Bayes factors for the remaining three diagnostic models can be based on counts of the number of trace-model MC samples (represented by the oval in Figure 1.6a) that follow the orders dictated by each. However, this method is usually impractical, because posterior trace-model MC samples may be obtained only rarely.
To avoid this inefficiency we use a type of Markov Chain Monte Carlo (MCMC) algorithm, an order-constrained Gibbs sampler (Gelfand, Smith & Lee, 1992, see final section of this chapter for details), to approximate a sequence of MC samples from the posterior of the trace model. The trace model is represented by the unit rectangle in Figure 1.6b, with the proportion of monotonic posterior trace-model MC samples represented by the oval. The area outside the oval represents the proportion of posterior multi-dimensional model MCMC samples relative to the trace model ($\hat{\Pi}_{MD,T}$). The two regions within the oval represent the proportions of posterior no-overlap model ($\hat{\Pi}_{NO,T}$) and uni-dimensional model ($\hat{\Pi}_{UD,T}$) MCMC samples.

![Diagram of nesting models](image)

**Figure 1.6.** Illustrations of the nesting relationship between samples from (a) the unrestricted (U) and trace (T) models, and (b) the trace model and the monotonic (M) model, which is represented by an oval divided into areas corresponding to monotonic orders with overlapping (O) and non-overlapping (NO) data traces. The $\Pi$ values indicate proportions of samples corresponding to the model indicated by the subscript.
These three posterior proportions relative to the trace model are multiplied by the posterior trace-model proportion obtained from unrestricted posterior model MC samples (Figure 1.6a) to obtain estimates of proportions relative to the unrestricted model ($\hat{\Pi}_{MD,U}$, $\hat{\Pi}_{NO,U}$, and $\hat{\Pi}_{UD,U}$), and hence Bayes factors relative to the unrestricted model ($BF_{MD,U} = \hat{\Pi}_{MD,U} / \pi_{MD,U}$, $BF_{NO,U} = \hat{\Pi}_{NO,U} / \pi_{NO,U}$, and $BF_{UD,U} = \hat{\Pi}_{UD,U} / \pi_{UD,U}$).

Klugkist, Laudy and Hoijtink (2005) describe Bayes factors relative to the unrestricted model as measuring model fit in a way that takes into account model complexity. Intuitively, the adjustment for complexity occurs because, as a model’s prior proportion ($\pi$) increases (indicating that the model is a priori more complex because it is able to fit a greater variety of data by chance), the Bayes factor decreases. Similarly, the Bayes factor increases as the model’s posterior proportion increases, indicating that it provides a better fit to the data. When the Bayes factor for a model relative to the unrestricted model is greater than one it provides evidence that the model is preferred over the unrestricted model with associated posterior model probability

$$p_{m\in[U]} = BF_{m,U} / (1 + BF_{m,U}).$$

Model-selection strategies

Posterior model probability estimates for each diagnostic model ($m = 1...4$) within the set of four diagnostic models ($\hat{P}_{m(U)}$) are given by

$$\hat{BF}_{m,U} / \left(\hat{BF}_{NT,U} + \hat{BF}_{MD,U} + \hat{BF}_{NO,U} + \hat{BF}_{UD,U}\right),$$

where \{U\} = \{NT, MD, NO, UD\}.

These posterior model probabilities quantify the relative evidence for and against each diagnostic model when no order is assumed more likely before observing the data and one of these models is assumed to be the true model. We will refer to model selection based on these probabilities as using a simultaneous exhaustive strategy.
The expensive computations (sampling and counting orders) implementing the simultaneous exhaustive strategy can be re-used to quickly explore many other model-selection strategies. For example, if the trace model is assumed to be certainly true, posterior model probabilities can be calculated relative to the remaining three diagnostic models. \( \hat{p}_{m,T} = BF_{m,U} \left( BF_{MD,U} + BF_{NO,U} + BF_{UD,U} \right) \), where \( \{T\}=\{MD, NO, UD\} \). We will describe model selection based on these probabilities as following a simultaneous trace-true strategy. Simultaneous trace-true probabilities can also be calculated from only posterior trace-model MCMC samples,

\[
\hat{p}_{m,(T)} = BF_{m,T} / \left( BF_{MD,T} + BF_{NO,T} + BF_{UD,T} \right),
\]

where \( BF_{m,T} = \hat{\Pi}_{m,T} / \pi_{m,T} \).

Selection could also be applied sequentially, based on posterior probabilities for pairs of models at each step. For example, first trace and non-trace models are compared, then the multi-dimensional and monotonic models and finally uni-dimensional and no-overlap models. The model selection sequence terminates if the non-trace model probability \( (p_{NT,(NT,T)}) \) is high, supporting a need for design refinement. Termination also occurs if the multi-dimensional model probability \( (p_{MD,(MD, M)}) \) is high, supporting more than one latent variable. If the monotonic model is selected the final comparison determines whether there is support for a single latent variable (higher \( p_{UD,(UD,NO)} \)) or the need to calibrate the trace factor to improve overlap (higher \( p_{NO,(UD, NO)} \)).

Different selection strategies can answer similar questions, but when they are based on different assumptions they can give different answers. For example, the simultaneous trace-true strategy typically produces more decisive selection among the multi-dimensional, uni-dimensional and no-overlap models than the simultaneous exhaustive strategy, reflecting the certainty with which it assumes that the non-trace
model is not the true model. Similarly, posterior probabilities calculated in the sequential strategy reflect the added assumptions for each step after the first (i.e., the non-trace model is false, then both non-trace and multi-dimensional models are false).

Although the three methods we have outlined often produce consistent results we generally prefer the simultaneous exhaustive strategy, both because of its weaker prior assumptions and the usefulness of each of its potential outcomes. However, where prior experimental evidence in favour of the trace model (i.e., evidence for a monotonic effect of the trace factor) is deemed sufficiently strong, the simultaneous trace-true strategy has the advantage of typically being more decisive, although the difference may be small. The same is often the case for later steps in the sequential strategy, but the conditional nature of these probabilities can also sometimes be confusing.

**Group Bayes factors**

Bayes factors for individual participants can be combined by multiplication to provide evidence about the best model for a group of participants. For example, individual participant Bayes factors for the trace and non-trace models (BF\textsubscript{T,U,i} and BF\textsubscript{NT,U,i} for \(i = 1\ldots n\) participants) yield corresponding group Bayes factors

\[ GBF_{T,U} = \prod_{i=1}^{n} BF_{T,U,i} \quad \text{and} \quad GBF_{NT,U} = \prod_{i=1}^{n} BF_{NT,U,i}. \]

These group Bayes factors can be combined to yield a posterior probability comparing a model in which the trace model is true for all participants to a model in which the non-trace model is true for all participants: \(gpt,(T,NT)=GBF_T / ( GBF_T + GBF_{NT} )\). Similar calculations produce group Bayes factors and posterior model probabilities that can be used in any of the model selection strategies just outlined.

It is important to note that the group Bayes factors do not provide evidence about the population of participants, as they assume that participants are unrelated,
rather than being samples from a common population distribution. Group Bayes factors may not be appropriate when the group of participants is heterogeneous with respect to the model comparison (i.e., when the true model differs among sub-groups of participants), as they make the assumption that all participants are of one type or another. However, they are preferable to analysing state-trace plots averaged over participants, which not only share the limitations of group Bayes factors, but which can also potentially be misleading due averaging distortion, as discussed previously.

**Example Analysis**

We illustrate the proposed analysis method by applying it to our inverted-test and upright-test experiments. We use letters to identify individual participant results; upper case for the inverted-test experiment and lower case for the upright-test experiment. Figure 1.7 graphically represents, for one participant in each experiment, the results of MC sampling from the unrestricted model posterior. The large points in the plots are posterior modes (i.e., points of highest posterior density) for the HR – FAR accuracy measure. Unrestricted model posterior MC samples can also be used to calculate other measures of central tendency, such as the mean or median (all three measures are very similar in our experiments). The plots also show 50% credible regions (the Bayesian analogue of NHST confidence regions), indicating the degree of uncertainty about posterior parameter estimates (see Figure 1.7 caption for details).³

³Note that the posterior means of the binomial probability parameters with a uniform prior can be directly estimated as they have a simple analytic form, \( \hat{\theta} = (n + 1) / (N + 2) \), where, for example, \( n \) is the number of hits and \( N \) the number of trials with studied items. However, this is not the case for accuracy measures, even in the simplest case, the HR – FAR measure. In contrast, estimates of posterior central tendency statistics for any accuracy measure are easily estimated by applying the accuracy measure to posterior MC samples. Similarly, credible regions are easily obtained from the appropriate quantile of the posterior accuracy distributions.
Figure 1.7. Modes of the posterior binomial probability parameter estimates from the unrestricted model (large symbols) with 50% credible regions (ellipses) for (a) participant G in the inverted-test experiment and (b) participant j in the upright-test experiment. The numbers 1…4 indicate shorter to longer study durations. The lines in (a) join posterior modes of MC samples from the trace model for each dimension factor level. The line in (b) joins posterior modes of MC samples from the best (i.e., most frequently sampled) monotonic model. Modes and credible regions were obtained using linear binned two-dimensional kernel density estimates (see, e.g., Wand & Jones, 1995) with a bivariate normal kernel, with covariance matrix given by Sheather and Jones’ (1991) direct plug-in algorithm.
Figure 1.7 shows that posterior MCMC samples can be used to visualize order-restricted models. For example, Figure 1.7a shows posterior modes for the trace model MCMC samples (small symbols), with separate lines joining points from different dimension-factor levels. Although the trace model plot for participant $G$ is consistent with multi-dimensionality the large credible regions suggest this may be attributable to estimation error.

Posterior MCMC samples can also be used to choose the monotonic order that best describes a state-trace plot and the level uncertainty about this choice. For example, Figure 1.7b shows the modes of the best monotonic model for participant $j$, where best is defined as the most frequently occurring monotonic model in posterior MCMC samples. In this case the best monotonic model is non-overlapping, as all inverted-condition modes are less than all upright-condition modes on both axes. Visual inspection of the unrestricted model modes in Figure 1.7b suggests that the monotonic model reversing the middle two points (i.e., triangle 1 before circle 4) is almost as good as the non-overlapping model. Reflecting this uncertainty, the Bayes factor for the non-overlapping model vs. the model with the middle points reversed, as estimated by the ratio of posterior counts for each model (as both models have equal prior probability), is only 1.05, with a corresponding posterior model probability of 0.53 for the no-overlap model.

Figure 1.8 plots estimated simultaneous exhaustive posterior model probabilities for individual participants. Figure 1.8a plots inverted-test experiment estimates and Figure 1.8b upright-test experiment estimates. Each of the four panels within a plot displays results for one of the four diagnostic models. In each panel participants are sorted in order of their probability estimates to make it easy to identify extreme cases.
The titles for each panel give group posterior model probabilities for the corresponding diagnostic model.

In order to aid interpretation of posterior model probabilities horizontal dotted lines in each panel indicate ranges which Raftery (1995) characterized as providing: a) equivocal evidence (against the model when \(0.25 \leq p < 0.5\) and for the model when \(0.5 < p \leq 0.75\)), b) positive evidence (against the model when \(0.05 \leq p < 0.25\) or for the model when \(0.75 < p \leq 0.95\)) or c) strong evidence (against the model when \(p < 0.05\) or for the model when \(p > 0.95\)). For example, Figure 1.8a shows the evidence for both the uni-dimensional and multi-dimensional models is equivocal for participant \(G\), confirming the impression given by the large credible regions in Figure 1.7a.

Figure 1.8a shows positive or better individual-participant evidence for the trace model in the inverted-test experiment. The exception is participant \(O\), who has positive evidence for the non-trace model. Because there is generally little evidence for the non-trace model, simultaneous trace-true posterior model probabilities for the other three diagnostic models are little changed from the values displayed in Figure 1.8a, except for \(O\), where \(\hat{p}_{NO,\{T\}} = 0.8\), \(\hat{p}_{MD,\{T\}} = 0.106\) and \(\hat{p}_{UD,\{T\}} = 0.094\).

The failure of the trace model for participant \(O\) is likely due to them having the lowest overall accuracy of any participant (an average HR – FAR = 0.13 averaged over conditions), which means the trace factor necessarily has only a small effect. This conclusion is consistent with small Bayes factors for participant \(O\) (\(\hat{BF}_{NT,\{U\}} = 1.00\), \(\hat{BF}_{NO,\{U\}} = 1.21\), \(\hat{BF}_{UD,\{U\}} = 0.014\) and \(\hat{BF}_{MD,\{U\}} = 0.016\)), indicating that no order-restricted model is preferred over the unrestricted model. In contrast, other participants have at least one order-restricted model with a Bayes factor of 8 or greater, and in most cases much greater.
Figure 1.8. Posterior model probability estimates for each participant (denoted by letters) in (a) the inverted-test and (b) the upright-test experiments. Each panel shows estimates for one of the four diagnostic models indicated by the title above the panel, which also provides the group posterior probability for each model. Within each panel participants are sorted in order of increasing probability estimates.
Taking into account their low accuracy, positive evidence for the non-trace model, and positive evidence for a failure of overlap (and hence non-diagnostic dimensionality results) even when the trace model is assumed true, censoring of participant O is indicated. However, when this is done, there is little change in the group posterior model probabilities for the inverted-test experiment ($\hat{gp}_{UD,(U)} = .94$, $\hat{gp}_{MD,(U)} = .06$, and $\hat{gp}_{NT,(U)} = \hat{gp}_{NO,(U)} = 0$). Individual analyses also favour a uni-dimensional model for all participants (except O), with participants A and D being possible exceptions. In contrast to other participants, these participants have positive evidence against the uni-dimensional model, and equivocal evidence favouring the multi-dimensional model. With participants A and D, as well as participant O, censored, $\hat{gp}_{UD,(U)} = .996$, strongly supporting the homogenous uni-dimensional model for this subset of participants.

Figure 1.8b shows positive or better evidence supporting the trace model for all participants in the upright-test experiment. In contrast to the average results in Figure 1.4a, a failure of data-trace overlap is indicated for participant j (see Figure 1.7b), d and possibly f. However, censoring these participants has little effect on the group posterior model probabilities ($\hat{gp}_{MD,(U)} = .999$, $\hat{gp}_{UD,(U)} = .001$, and $\hat{gp}_{NT,(U)} = 0$, $\hat{gp}_{NO,(U)} = 0$). Overall, a strong conclusion in favour of a multi-dimensional model for all participants is supported for the upright-test experiment.

In order to further examine the effect of censoring, Figure 1.9 displays results averaged over uncensored participants. Modes and confidence regions are based on participant averages obtained by bootstrap methods (Efron & Tibshirani, 1998) applied in a way that reflects the uncertainty about each participant’s parameters quantified by their posterior distributions (see Figure 1.9 caption for details). The results shown in
Figure 1.9 are similar to results obtained when this method is applied to all participants (not shown). Both of these results are similar to Figure 1.4, which is to be expected, as Bayesian estimates based on a uniform prior are asymptotically equivalent to the maximum-likelihood estimates (i.e., $\hat{\theta} = n/N$) used to construct Figure 1.4. That is, given the number of test trials in our experiments the estimates are almost numerically equivalent.

**Summary and Recommendations for Statistical Analysis**

We recommend the use of Bayes-factor based analyses of individual participants for appropriate experiments (binary choice data and a within-subjects design). Generally the simultaneous exhaustive strategy is recommended, although when there is strong prior evidence for the trace model the trace-true strategy can be both more appropriate and diagnostic. Although we have emphasized posterior model probabilities, users uncomfortable with the underlying assumption that one model is the true model may prefer to rely on Bayes factors, which quantify the relative evidence for two models without making this assumption. In any case, we recommend examination of Bayes factors relative to the unrestricted model as a way of assessing absolute goodness-of-fit taking into account model complexity.
Figure 1.9. Modes of the posterior binomial probability parameter estimates from the unrestricted model averaged over participants (large symbols) with 50% credible regions (ellipses) for (a) the inverted-test experiment with participants A, D and O censored and (b) the upright-test experiment with participants d, f and j censored. Averages over participants were obtained by bootstrapping based on 10,000 posterior estimates from the unrestricted model for each participant. For each participant one of the 10,000 posterior samples was randomly selected and the selected samples averaged over participants. This procedure was repeated 10,000 times, and the methods described in Figure 1.7’s caption applied to obtain posterior modes and credible regions.
We recommend plots of individual posterior model probabilities (e.g., Figure 1.8) be examined to check for outliers or sub-groups that differ in terms of evidence for different models. We recommend group Bayes factors and/or posterior model probabilities and plots such as Figure 1.9 for presenting summary results. However, group statistics should be used with caution\textsuperscript{4}, with the reasonableness of aggregation checked and censoring applied, or separate group statistics based on sub-groups reported, as indicated by plots like Figure 1.8. With accuracy data it is also important to check for performance at floor or ceiling. Although state-trace analysis can address potential confounding of dimensionality estimation attending these effects, such data contain little useful information, making model selection equivocal (e.g., participant \textit{O} in the example analysis).

All of the types of statistics and figures presented in the example analysis, and extensions, can be conveniently obtained using software freely available at http://www.newcl.org (see Prince, Hawkins, Love & Heathcote, 2012a – Chapter Three). To explore extensions, users may also, for example, access raw posterior MC and MCMC sample-order counts to compute the Bayes factors required to explore alternative model-selection strategies, plot posterior means and medians as well as modes, and construct average state-trace plots by bootstrap methods that treat participants as samples from a population.

\textsuperscript{4}A simulation study reported by Hawkins, Prince, Brown and Heathcote (2010; Chapter Two) found that group Bayes factors can fail to identify the dimensionality correctly when the number of trials in each condition for each participant is small. Hence, we recommend that experiments be designed to maximize the number of trials in each condition (see the last major section for recommendations about how to achieve this outcome). Note, however, that the simulation study assumed a worst-case scenario, where all participants were samples from an identical population model with no between-subjects variation, whereas group Bayes factors assume participants are unrelated. In ongoing work we are testing the operating characteristics of group Bayes factors in more realistic situations where participants are samples from a population distribution with between-subjects variation.
General Discussion

A key methodology in many areas of psychology involves studying the interplay between two factors in order to determine whether their effect can be explained by a single latent variable. The traditional approach to this issue, rejecting a one-dimensional model when a significant interaction is found in an ANOVA, requires strong assumptions that are difficult to check (Dunn & Kirsner, 1988) and the fortuitous finding of a particular data pattern such as a crossover interaction (Loftus, 1978). State-trace analysis (Bamber, 1979) is an alternative approach that places control back in the hands of the experimenter. Its assumptions are minimal and a diagnostic pattern of data can usually be guaranteed through systematic manipulation of a third (trace) factor. Although the experimental methodology for state-trace analysis differs in its requirements from familiar factorial designs, once the differences are recognized it is straightforward to develop and refine a state-trace experiment. The present chapter provides guidelines for this process, which apply to almost all of the types of state-trace experiments in the existing literature.

Although we advocate the wider use of state-trace analysis it is important to recognize its limitations. Although state-trace analysis can provide strong evidence for more than one latent variable that does not necessarily imply the involvement of more than one cognitive module, representation, process, or brain region if these entities have a multivariate nature. For example, parametric single-process models of recognition memory, which assume recognition decisions are based on a single memory-strength dimension, are usually multivariate as memory strength is characterized by both mean and variance parameters. If experimental manipulations have independent effects on both parameters state-trace plots will be non-monotonic because the two parameters of
the model act as two latent variables and so, in Loftus et al.’s (2004) terminology, there are two dimensions.

In general, we see state-trace analysis and parametric model fitting as complimentary approaches. State-trace analysis is a relatively assumption free method of identifying the number of model parameters that should be freely estimated in order to fit data. For example, Heathcote, Bora and Freeman (2010) found that quite different (single and dual process) models of recognition memory required the same number of free parameters as indicated by Heathcote et al.’s (2009) state-trace analysis of the same paradigm. As another example, Brainerd et al. (2009, Figure 12) applied state-trace analysis to direct access and reconstruction parameter estimates from a fitted model of recall in order to check whether they measure distinct processes.

We believe that inference about dimensionality requires an alternative to traditional null-hypothesis statistical testing, which can provide evidence against but not for a one-dimensional (null) model. Further, we believe it is important to take into account the greater flexibility (i.e., the ability to fit data by chance) of higher-dimensional models. Bayes factors represent one method to address such differences in functional form complexity (Pitt & Myung, 2002) and the limitations of null-hypothesis based analyses. When combined with Klugkist, Kato and Hoijtink’s (2005) method for approximating Bayes factors, the result is a statistical methodology compatible with the relatively assumption-free nature of state-trace analysis. We developed and implemented this approach for experiments with a binary dependent variable (e.g., accuracy), enabling users not only to assess dimensionality, but also to help refine experimental methodology. However, further work remains to extend this approach to the full range of existing state-trace applications.
**Future Directions**

One set of extensions relates to experiments with ratings or response time as the dependent variable. These extensions require the assumption of different data distributions, but are otherwise conceptually straightforward. Dunn (2008) describes an application of dependent-variable state-trace analysis to three-level rating data, which works with the binary-response analysis developed here, where the state factor is made up of cumulative probabilities for two of the three ratings levels. A second set of extensions is to designs with a dimension factor that has more than two levels or multiple dimension factors. Once again this extension is conceptually straightforward, although computational cost increases rapidly with the number of experimental conditions.

Kleigel, Maayr and Krampe (1994) state that: ‘state-trace analysis can be thought of as a factor-analysis for experimental research; it yields the minimum number of mechanisms required for description of ordinal interactions’ (p.153). For example, Heathcote et al. (2009) used state-trace analysis in way analogous to identifying manipulations that load on different factors. They showed that the effect of one manipulation was non-monotonic, indicating the need for at least two latent variables, but that the effects of two other manipulations were monotonic within each level of the non-monotonic manipulation, suggesting the effect of the latter two manipulations load on the same latent variable.

However, state-trace analysis with only two state levels is not fully analogous to factor analysis as it only determines if one or more than one latent variable is required. In order to determine the number of dimensions above one, extra axes must be added to the state-trace plot. That is, to identify the number of dimensions up to \( D \), a state factor with \( D + 1 \) levels is required (see Dunn & James, 2003, for a general formulation of this
problem). The potential importance of pursuing this extension is indicated by typical results from structural equation modeling, where reliable estimation of latent mediators usually requires converging evidence from several indicator variables. It seems likely that state-trace analysis will also more reliably identify dimensionality using a state factor with more than two levels (where levels are analogous to indicator variables).

A key area for future extension concerns addressing differences between participants. We applied our model-selection procedures to individual participant data because of concerns about averaging. Multilevel models provide an alternative method of group analysis that does not require averaging, and which can also be applied to designs with between-subjects factors. For example, Verhaeghen and Cerella (2002, see also Verhaeghen et al., 2003) used maximum-likelihood multilevel modeling, assuming a linear relationship between levels of a dependent-variable state factor, to a meta-analysis where the trace factor represented different studies, and Van Den Broeck and Geudens (submitted) where the trace factor represented different participants. Dimensionality was diagnosed by selection between a model with different linear relationships for levels of the dimension factor (i.e., a multi-dimensional model) and a model with a common relationship (i.e., a uni-dimensional model).

Further discussion and development of this approach is beyond the scope of the present chapter, but we note that Klugkist, Kato and Hoijtink’s (2005) Bayes factor method can be implemented in a multilevel framework (Myung et al., 2008), avoiding the need for assumptions about a particular parametric form for the relationship between levels of the state-factor. Multilevel models also have an advantage for purely within-subjects designs in providing group-level summaries based on the plausible assumption that participants come from a common population. The individual level methods
proposed here provide a basis for the development and validation of such multilevel models (Borsboom, Mellenbergh & van Heerden, 2003).

**Technical Details for Bayesian State-Trace Analysis**

**Prior Proportions**

We assume both state and dimension factors have two levels and the trace factor has \( t \) levels. The no-baseline design is denoted as B0 and a design with one baseline for each state, as B2. Analytic prior proportions \( (\pi) \) are obtained based on an encompassing prior that assumes identical and independent uniform distributions for all binomial probability parameters \( (\theta) \), making every ordering of these parameters equally likely. In the following, \( \theta \) for a baseline condition is denoted \( \theta_0 \) and for the remaining conditions \( \theta_i, i = 1, 2, \ldots t \). For brevity we refer to draws from the prior and posterior distributions of an order-constrained model, whether obtained by MC or MCMC methods, as prior and posterior samples.

**Proportions within the unrestricted model**

The trace model assumes that the same order holds within a data trace for both states. Given there are \( t \) levels within a trace and \( t! \) possible orders:

\[
\pi_t(B0) = [t!]^{-4}
\]  

(1.1a)

For a B2 design we include the extra constraint that \( \theta_0 < \theta_i \) within each state \( \times \) dimension condition. This constraint has probability \( 1/(2t+1) \) as it applies simultaneously to all \( 2t \) non-baseline \( \theta \). In combination with the probability of obtaining the required order of the remaining \( \theta \) parameters within each state \( \times \) dimension condition:
\[ \pi_T(B2) = \left( (t!)^2 (2t+1) \right)^{-2} \quad (1.1b) \]

**Proportions within the trace model**

To determine the prior probability of the monotonic model within the trace model, we first require the number of possible orderings for trace models. In general for \( k \) traces of equal length there are \( \left( k \times t \right)! \left/ \left( t \times \left( k \times t - t \right)! \right) \right. \) (i.e., \( k\ C_t \)) ways of choosing a trace model order, so for \( k = 2 \) the number of orders is:

\[ O_t(t) = (2t)!(t!)^{-2} \quad (1.2) \]

Given there are \( 2t\theta \) values for each state in the B0 design, the probability of any one trace model order is \( 1/(2t)! \), and so the proportion of prior samples with a monotonic ordering is the product of this value for each state with the number of trace models:

\[ O_T(t) \times \left[ (2t)! \right]^{-2}. \]

Similarly, given \( 2t + 1 \) values of \( \theta \) for each state (including \( \theta_0 \)), for the B2 design the proportion of prior samples with a monotonic ordering is:

\[ O_T(t) \times \left[ (2t+1)! \right]^{-2}. \]

Note that in this case, although the order is on the \( d = f(\theta_i) - f(\theta_0) \) differences measuring accuracy (where \( f \) is a monotonic function), we need only consider the ordering on the \( \theta_i \) as this necessarily entails the same ordering on \( d \) given the common baseline \( \theta_0 \).

The proportion of prior MCMC samples from the trace model that have a monotonic order is given by dividing the proportion of unrestricted prior samples with a monotonic ordering by the proportion that conform to the trace model: this proportion is the same for B0 and B2 designs:

\[ \pi_M = (t!)^2 / (2t)! = 1 / O_T(t) \quad (1.3) \]
The proportion of monotonic but non-overlapping data traces in prior MC samples from the trace model (i.e., samples not diagnostic of dimensionality), $\pi_{NO}$, is obtained from the fact that, for any number of trace levels, there are always exactly two non-overlapping orders and that each of the $O_T$ monotonic model orders is equally likely:

$$\pi_{NO} = 2^{2^{O_T(t)}}$$  \hspace{1cm} (1.4) \hspace{1cm}

This proportion is the same for B0 and B2 designs and can be used to get the proportion of monotonic MC samples that are overlapping, and thus diagnostic of a uni-dimensional system: $\pi_{UD} = \pi_{M} - \pi_{NO}$.

**Posterior Proportions**

The uniform distribution is a special case of the Beta distribution, Beta $(a, b)$, where $a = b = 1$ for the uniform distribution. The Beta distribution is conjugate to the binomial distribution, meaning that the marginal posteriors also have a Beta distribution. For example, if $s$ studied responses are observed from $S$ recognition test trials, the marginal posterior has a Beta $(s + a, S - s + b)$ distribution. Samples drawn from the latter Beta distribution are used to estimate posterior model proportions. One possible proportion estimate for a given model with a count of $n$ out of $N$ posterior samples is $\hat{\Pi} = n / N$. However, in practice the estimate corresponding to the posterior mean proportion under a uniform prior, $\hat{\Pi} = (n+1) / (N+2)$, is preferable, as it has lower estimation variance (Rouder & Lu, 2005).

Computation of the trace model proportion can be done efficiently by taking advantage of the independence of accuracy measures between state and dimension factor conditions. Proportions are determined separately for independent parts of the
posterior sample then combined by multiplication. Where accuracy is measured by proportion correct (e.g., for two-alternative forced choice testing), all four conditions are independent, so \( \hat{\Pi}_i = \hat{\Pi}_{i,1,1} \times \hat{\Pi}_{i,1,2} \times \hat{\Pi}_{i,2,1} \times \hat{\Pi}_{i,2,2} \), where \( \hat{\Pi}_{i,j} \) is the proportion of MC samples from the \( i \)'th state level and \( j \)'th dimension level following the trace order. Where accuracy is measured relative to a baseline (e.g., the HR-FAR measure) that is common to all conditions within a level of the state factor (e.g., one FAR for houses and one for faces) only the state levels are independent, as the trace model contains both dimension levels relative to their shared baseline (i.e., \( \theta_{0,0} < \theta_{0,2,1} \) and \( \theta_{0,1} < \theta_{1,2,1} \)). Hence, \( \hat{\Pi}_i = \hat{\Pi}_{i,1} \times \hat{\Pi}_{i,2} \), where \( \hat{\Pi}_{i,j} \) is the estimate for the proportion of posterior MC samples in state \( i \) conforming to the trace order.

**Posterior trace model proportions**

Samples can be obtained directly from the posterior of the trace model using Gibbs sampling (Gelfand et al., 1992). The computational cost of Gibbs samples is around an order of magnitude greater than directly sampling from the Beta distribution. Also, samples from the initial iterations of the MCMC algorithm must be discarded because they do not provide a good approximation to the posterior distribution. Further inefficiency occurs because the sequence of MCMC samples is not independent, so an MCMC sample contributes less information than a direct sample. However, in practice we have found that this method remains computationally superior to estimating trace-model proportions by directly sampling from the unrestricted model posterior.

The Gibbs algorithm can be specified in terms of a set of marginal Beta cumulative distribution functions, \( F_k \), and their inverses, \( F^{-1}_k \). Denote the \( i \)'th sample by a vector, \( x_i \), with \( n \) elements \( (x_{i1}, x_{i2} \ldots x_{in}) \) – these elements represent posterior
probability samples over each of the 2t conditions in the B0 design and 2(t + 1)
conditions in the B2 design.

For the B0 design let k = 1 ... t correspond to increasing trace conditions within a
level of the dimension factor. Marginal posteriors are distributed as Beta
\( (n_k + a, N_k - n_k + a) \), where \( n_k \) and \( N_k \) are the number of successes (e.g., correctly
responding that the test item was studied) and the number of trials, respectively, for the
k’th condition. Given an arbitrary initial sample \( x_1 \) which respects \( x_{11} < x_{12} < \ldots < x_{1n} \) a
sequence \( x_i, i = 1...S \), of samples from the trace model is obtained, after sufficient burn-
in samples are discarded, by randomly selecting and updating an element \( k \) of \( x_i \) using:

\[
x_{i(k+1)} = F_{k}^{-1}\left[F_k\left(x_{(k+1)}\right) + u\left(F_k\left(x_{(k+1)}\right) - F_k\left(x_{(k-1)}\right)\right)\right]
\]

In (1.5) \( u \) is a uniform random deviate on the unit interval and we define 
\( F_k(x_{i0}) = 0 \) and \( F_k(x_{i(t+2)}) = 1 \). The Gibbs sampler is run four times to produce four independent
sets of samples, one for each state \( \times \) dimension condition.

In the B2 design a single Gibbs sampler must be defined to obtain samples from
both levels of the dimension factor at once, as both levels of the dimension factor share
a baseline. For generality we define a Gibbs sampler for the case where there are
possibly different numbers of trace levels for each state, \( t_1 \) and \( t_2 \) and let \( t = t_1 + t_2 + 1 \).
Let \( k = 1 ... t \), where \( k = 1 \) corresponds to the baseline condition, \( k = 2...\left(t_1+1\right) \)
corresponds to increasing trace conditions in the first dimension level and
\( k = \left(t_1+2\right)...t \) corresponds to increasing trace conditions in the second dimension
level.
Let \( x_{ik} \) represent a sample for condition \( k \) on iteration \( i \). The samples must respect the joint order:

\[
\begin{bmatrix}
  x_{i_1} < x_{i_2} < \ldots < x_{i_{(n+i)}} \\
  x_{i_1} < x_{i_{(n+i-1)}} < \ldots < x_{i_{(n+i+n+1)}}
\end{bmatrix}
\]

This is achieved by the following update rule:

\[
x_{(i+1)k} = F^{-1}_k \left[ p + u \left( P - p \right) \right]
\]

(1.6)

Where:

\[
p = 0 \quad \text{if} \quad k = 1
\]

\[
p = F_k \left( x_{it} \right) \quad \text{if} \quad k = t1 + 2
\]

\[
p = F_k \left[ x_{i(k-1)} \right] \quad \text{otherwise}
\]

and:

\[
P = \min \left[ F_k \left( x_{i2} \right), F_k \left( x_{i(n+2)} \right) \right] \quad \text{if} \quad k = 1
\]

\[
P = 1 \quad \text{if} \quad k = t1 + 1 \text{ or } k = t
\]

\[
P = F_k \left[ x_{i(k-1)} \right] \quad \text{otherwise}
\]

Finally, we define a sequence of accuracy measures for each dimension as

\[
d_{ij} = d \left[ x_{i(j+1)}, x_{ii} \right], \text{ where } j = 1 \ldots n1 \text{ and } d_{2ij} = d \left[ x_{i(n+j+2)}, x_{ii} \right], \text{ where } j = 1 \ldots n2 \text{ (e.g.,}
\]

\[
d(x,y) = x - y \). The sampler is run twice to produce two independent sets of accuracy samples, one for each state.

Having obtained a sufficiently large set of samples from the posterior of the trace model, \( \Pi_M \) can be estimated by counting the number which obey the constraints of the monotonic model, which can be expressed as \( \text{rank}(d_1) = \text{rank}(d_2) \), where \( \text{rank} (y) \) is a function that returns the ranks of the elements of the vector \( d \) and the equality is in the sense of corresponding vector elements being equal. The issue of ties can be ignored as \( d \) is real valued, so ties occur with probability zero. When ties do occur to the limits of
the machine’s precision, they can be broken randomly with little influence on the outcome.

Adjacent samples generated by (1.5) and (1.6) are highly correlated, as they share all but one value. Where independent posterior samples are required they can be obtained by keeping only one sample in every $T$ when $T$ is sufficiently large. We set $T$ equal to the length of a sample (i.e., on average every element had been updated between samples that were kept) and found this produced sufficiently independent samples for our purposes (in particular, so that we could estimate numerical accuracy as described in the next section). Given these specifications a burn-in period discarding only the first 100 samples was also found to be sufficient.

**Numerical Accuracy**

Monte-Carlo estimates become more precise as the sample size, $S$, on which they are based increases. We choose $S$ such that the $(1-\alpha)$ credible interval for an estimate of a posterior proportion, $\Pi$, was less than $\delta$, i.e.,

$$F^{-1}(1-\alpha/2, a, b) - F^{-1}(\alpha/2, a, b) < \delta,$$

where $F^{-1}(x,a,b)$ is the inverse of the cumulative distribution function for the Beta distribution with $a = \Pi S + 1$ and $b = S - \Pi S + 1$. Calculation of this interval requires knowledge of $\Pi$. We bootstrapped this knowledge by taking a sample of size $S_0$, estimating $\Pi$, calculating the credible interval. We then iterated this procedure until the credible interval criterion was fulfilled.

This process was carried out separately with posterior samples from the unrestricted and trace models. For the unrestricted model credible intervals were calculated for each of the independent proportion estimates used to calculate $\hat{\Pi}_T$. For
the trace model credible intervals were calculated for estimates of $\Pi_{MD}$, $\Pi_{NO}$ and $\Pi_{UD}$.

In both cases sampling was terminated when all estimates fulfilled the criteria. Monte-Carlo error in the $\Pi_T$ estimate affects calculation of Bayes factors for the multi-dimensional, no-overlap and uni-dimensional models relative to the unrestricted model (as the trace model proportions are multiplied by $\hat{\Pi}_T$). Hence, we used a stricter criterion for the unrestricted model ($\delta = .0005$) than the trace model ($\delta = .005$) sampling, with $\alpha = .05$ in both cases.
Chapter Two

Model Recovery Properties of Bayesian State-Trace Analysis

Adapted from:

A Bayesian Approach to State-Trace Analysis

In Chapter One, state-trace analysis (Bamber, 1979) was introduced as a method for determining whether a single latent variable is capable of explaining the joint effect of two experimental factors. State-trace analysis is largely able to avoid the caveats associated with traditional dissociation methods by assessing the ordinal relationship between the effects of experimental factors: if all of the points in a state-trace plot fall on a single monotonic function then a single latent variable must be in play. The state-trace plot is created by plotting one level of the state factor against the other (e.g., recognition accuracy for faces and for houses). The dimension factor defines sets of points within the plot and it is the interaction between the state and dimension factors that can potentially reveal the underlying latent dimensionality (e.g., manipulating orientation can potentially change face processing from being two-dimensional when upright to one-dimensional when inverted). The final factor in the plot, the trace factor, is of lesser theoretical importance, however, is used to ensure diagnostic state-trace results (i.e., sweeping out a set of points within each level of the dimension factor, and helping to achieve data-trace overlap).

Given an observed state-trace plot, where the effects of the underlying latent variable(s) are perturbed by measurement error, how can we determine whether a monotonic curve best describes the data? Although a number of statistical methods for assessing departures from monotonicity have been suggested (see Loftus, Oberg & Dillon, 2004; Newell & Dunn, 2008), we proposed a Bayes factor approach to state-trace model selection (Prince, Brown & Heathcote, 2012 – Chapter One), based on Klugkist, Laudy and Hoijting’s (2005) encompassing prior methods. The encompassing prior method uses Bayes factors to select among models defined by inequalities (i.e.,
order constraints). The advantage of this approach is that it automatically accounts for differences in flexibility amongst models, which is a key issue in state-trace analysis as a one-dimensional model is far less flexible than a multi-dimensional model.

For our Bayesian state-trace approach we assume binomially distributed data (e.g., a binary two-alternative forced choice response), with state-trace models being defined by sets of inequality constraints on binomial probability parameters. We select among four mutually exclusive models, which together constitute the encompassing (unrestricted) model:

1. **Non-trace model**: the trace factor does not always have a monotonic effect and so any multi-dimensional evidence cannot be clearly attributed to the state-by-dimension factors interaction.

2. **No-overlap model**: the data traces do not overlap on either axis and so the state-trace plot cannot provide diagnostic evidence of the underlying dimensionality.

3. **Unidimensional model**: the state-trace plot provides diagnostic monotonic evidence; that is, a single latent variable is at play.

4. **Multi-dimensional model**: the state-trace plot provides diagnostic non-monotonic evidence; that is, more than one latent variable is at play.

When model $M_i$ is an order constrained version of an encompassing model $M_k$, Bayes factors (Kass & Raftery, 1995) can be estimated from prior and posterior samples from the encompassing model (Klugkist, Kato & Hoijtink, 2005). The proportion of prior $(\hat{\pi})$ and posterior $(\hat{\Pi})$ samples that adhere to the order constraints of the more restricted model $M_i$ are used to estimate a Bayes factor from the ratio of the two sample counts,

$$BF_{ik} \approx \frac{\hat{\Pi}}{\hat{\pi}}$$  \hspace{1cm} (2.1)
This Bayes factor indicates the strength of evidence in favour of $M_i$ over $M_k$. Intuitively this is the case because it is the ratio of the probability that the model fits the data before the data are observed (which is proportional to the complexity of the model; e.g., the maximally complex encompassing model will always fit any data pattern), to the actual fit of the model to the data. If this ratio is greater than one it indicates that the model fits better than chance. For example, if the two models are initially considered equally likely, $BF = 10$ implies $M_i$ is ten times more likely than $M_k$ after observing the data.

A set of such Bayes factors, assuming the same encompassing model, can be used to compare a set of order-restricted models by calculating each models’ posterior model probability, $p(M_i)$, which is the probability that model $M_i$ is the ‘true’ (data generating) model, based on the assumption that one model in the set is the true model. Even if the true-model assumption is not made, model selection based on $p(M_i)$ can also be justified on other grounds (e.g., it selects the model that is most likely to minimise a measure of error in predicting new data), and so we refer to it simply as a method of selecting the ‘best’ model. For a set of models $M_i$, $1 \ldots m$, the posterior model probability for $M_i$ is

$$p(M_i) = \frac{BF_{ik}}{\sum_{j=1}^{m} BF_{jk}}$$

(2.2)

for any $j = 1 \ldots m$, which includes $i$. Throughout we assume each model is equally likely to be the best model before observing the data.

In this chapter we aim to assess, via simulation, how often Prince, Brown and Heathcote’s (2012) Bayesian state-trace analysis selects the correct number of latent variables, either one or more than one. We begin by simulating an individual participant
analysis. We then examine a method of aggregating participant results to select the best characterisation of dimensionality for a group of participants.

**Simulations**

Figure 2.1 shows state-trace data consistent with a single latent variable model (1D) and a two latent variable model (2D). In both cases the trace factor has a clear monotonic effect on performance; that is, as the level of the trace factor increases so too does the dependent variable. The two models also both exhibit moderate and equal data trace overlap. These two patterns were used to generate simulated data (by using their coordinates to specify binomial probability parameters) and we will refer to them as the 1D and 2D models.

*Figure 2.1.* The two models on which simulations were based. $p$(State 1) and $p$(State 2) refer to the proportion of correct responses for the first and second level of the state factor, respectively. The two lines on each plot represent data traces, one for each level of the dimension factor. The solid lines are identical for both models, and the dashed line for the 2D model is the same as the dashed line for the 1D model but transposed downward by 0.1.
We next elaborated the 1D and 2D models with 2 trace levels shown in Figure 2.1, which we call the $T_2$ designs, by creating variants with three and four trace levels, $T_3$ and $T_4$ designs respectively. In $T_2$ designs the two levels of the trace factor provided data for the end points of the data traces. For $T_3$ and $T_4$ designs the additional levels were evenly spaced between the end points of each data trace. One purpose of these simulations was to provide guidance on experimental design in terms of the trade-off between number of trials contributing to the estimates of each point in the state-trace plot and the number of levels in the trace factor. For a fixed sample size (number of trials) there is a trade-off between these two factors, with more trace levels resulting in fewer trials per point. For each model and each $T$ we explored 6 total trial numbers ($n$) with total $n$ conserved across each $T$ at 192, 384, 768, 1536, 3072 and 6144. For example, a model with $n = 192$ had 24 observations per coordinate of each point for $T_2$, 16 observations for $T_3$, and 12 observations for $T_4$. In total we performed 36 simulations ($2 \times 3 \times 6$ sample sizes). For each simulation 1000 Monte Carlo replicates were sampled from binomial distributions with parameters determined by the design and model. Sufficient posterior samples were obtained so that posterior proportions of monotonic samples had 90% credible intervals less than 0.025; prior proportions were determined analytically assuming a uniform prior (see Prince, Brown & Heathcote, 2012 – Chapter One – for details).

**Bayesian State-Trace Results**

For each simulation we estimated Bayes Factors to test four mutually exclusive models, which we refer to as the non-trace (NT), no-overlap (NO), unidimensional (UD) and multidimensional (MD) models. Together these models account for all possible orders (i.e., together they constitute the encompassing model). Posterior model
probabilities were calculated for each Monte Carlo replicate for each model by dividing each Bayes Factor by the sum of all four Bayes Factors (i.e., Equation 2.2), which we refer to as $p(NT)$, $p(NO)$, $p(UD)$ and $p(MD)$, respectively. Figure 2.2 illustrates results in terms of the proportion of comparisons selecting one of the four models (i.e., where the models posterior probability was greatest amongst the set of four models). Figure 2.2 can be interpreted by comparing the height of corresponding points across the panels in each row. In particular, the `highest' point indicates which of the four models is most often supported.

**The Trace Model**

An important first check in any state-trace analysis is to determine whether the trace model is supported. For example, in Chapter One study duration was manipulated as the trace factor. In this case the trace model indicates that accuracy increased as study durations became longer for both levels of the state and dimension factors. In contrast, support for the non-trace model indicates that the order dictated by the trace factor was violated (i.e., for one or more combinations of the state and dimension factor levels, accuracy did not increase with longer study durations). Even when the trace model is the data generating model, measurement noise can cause violations of the trace model (i.e., support for the non-trace model) to arise more frequently when differences between levels of the study duration factor produce only small changes in accuracy. Support for the non-trace model clouds any conclusions about underlying dimensionality of the state factor since the effects of the dimension and trace factors are confounded, and can suggest that the experimental design needs to be improved by using more widely spaced trace factor levels.
Figure 2.2. Model selection results for both data generating models, type of comparison, number of trace levels, \( T \), and number of `trials', \( n \). Columns correspond to each of the mutually exclusive models being tested and rows to the type of the data generating model. On each plot the x axis represents the six levels of \( n \) and the y axis represents the proportion of simulations in which posterior model probability favoured the model specified for each column. The lines group designs with the same \( T \).
The non-trace model results are shown in the left column of Figure 2.2. The figure demonstrates a number of key points. As expected, evidence for the trace model is similar across both 1D and 2D simulations, since the trace factor should have a consistent effect irrespective of underlying dimensionality. Secondly, as total sample size increases the lines always approach zero, indicating consistent selection of the trace model. That is, Prince, Brown and Heathcote’s (2012) Bayesian state-trace analysis recovers the trace model with increasing reliability as measurement error decreases due to an increase in sample size. Finally, the probability of selecting the non-trace model approached zero with lower total trials for $T_2$ compared to $T_3$ and $T_4$. As seen in Figure 2.2, selection is approximately zero for $T_2$ at $n = 768$, whereas this increased to $n = 1536$ for $T_3$ and $T_4$ in the 1D model, and to $n = 3072$ for $T_4$ in the 2D model. Thus, for smaller $n$, the trace model had a greater chance of being supported in $T_2$ designs compared to $T_3$ and $T_4$ designs. This occurs because the combination of a smaller sample size (and hence greater measurement noise) and closer spacing between results for adjacent trace levels as $T$ increases makes a violation of monotonicity within a data trace more likely.

**The No-Overlap Model**

When the trace model holds it implies that one of the three remaining models best describes the data, as they are each order constrained versions of the trace models. A monotonic state-trace plot is a special case of the trace model where all data points have the same ordering for both levels of the state factor. A non-overlapping monotonic plot is a case where data traces for both levels of the dimension factor do not cross over at any point along either axis of the state-trace plot. In this case, monotonicity is not diagnostic of dimensionality, as both one-dimensional and multi-dimensional data...
generating models produce monotonic state-trace plots when there is a failure of data trace overlap. Hence, an important second check in a state-trace analysis is to determine whether the no-overlap model holds.

Results for the no-overlap model differed between the 1D and 2D data generating models. The 1D simulation results generally give some support for the no-overlap model, which is perhaps not surprising given the 1D model produces monotonic data. Of more concern is the fact that this support was inconsistent as a function of sample size, \( n \), for \( T4 \) and to a lesser degree for \( T3 \). That is, support for the no-overlap model initially increased with \( n \), but then decreased, from \( n = 1536 \) for the \( T4 \) design and from \( n = 768 \) for the \( T3 \) design. In contrast, the no-overlap model consistently received little support across all \( T \) and \( n \) in the 2D simulations. Overall, these results suggest that when there is in fact trace overlap in a one-dimensional data generating model, the no-overlap model is more often rejected in designs with fewer trace levels.

**The Unidimensional and Multidimensional Models**

For both data generating models the unidimensional and multidimensional posterior model probabilities provided support for the true model dimensionality. For the 1D case support for the unidimensional model (middle right column of Figure 2.2) increased with sample size, but the level of support was smaller for larger \( T \). For the 2D case support for the multidimensional model (right column of Figure 2.2) also increased with sample size. In contrast to the 1D case, the level of support was similar for all \( T \), although it was slightly less for the \( T4 \) design for smaller \( n \) (likely reflecting the larger level of support for the non-trace model) and slightly less for the \( T2 \) design for the second and third largest value of \( n \), with all \( T \) designs perfectly selecting the true model for the largest sample size. Across the 1D and 2D data-generating models, support for
the wrong dimensionality was generally low and decreased with sample size, although there was some inconsistency for the three smallest sample sizes.

Overall, the results of the simulation study indicate that accurate results for all comparisons can only be guaranteed for quite large sample sizes. This suggests that analysis of individual participant data may not produce clear results in applications where it is not possible to measure performance on a large number of trials for each individual. In such situations it would be desirable to have a method of combining results over participants in a way that improves correct identification at the group level. In the next section we extend the analysis of our simulation results to assess the performance of one such method suggested by Prince, Brown and Heathcote (2012), the group Bayes Factor.

**Group Bayes Factors**

A Bayes Factor for a group of participants, assuming each participant contributes independent evidence, can be obtained by taking the product of each participant’s Bayes Factor. Hence, a group Bayes Factor for model $M_i$ (relative to encompassing model $M_k$) is given by $GBF_i = \prod_{n=1}^{N} BF_n$, where $N$ is the number of subjects. Group Bayes Factors can then be combined to obtain a posterior model probability for model $M_i$ at the group level. Again we assume each model is equally likely to be the best model before observing the data, and so:

$$g_{p}(M_i) = \frac{GBF_i}{\sum_{j=1}^{m} GBF_j} \quad (2.3)$$

for a set of $j = 1 \ldots m$ models that includes model $i$.

We examined the utility of group Bayes Factors using the simulations from the previous section. For each simulation we sampled with replacement (i.e., resampled)
sets of individual Bayes Factors from the 1000 available. The sets were of sizes \(N\) 8, 16 and 32, representing experiments with different numbers of participants. These \(N\)'s cross with total trials \(n\) in a balanced manner. For example, a set of \(N = 32\) with \(n = 192\) trials provides results from a total of 6144 trials, equivalent to the set \(N = 16\) with \(n = 384\) trials, and \(N = 8\) with \(n = 768\) trials. The resampling procedure was repeated 500 times for each possible grouping: two data generating models (1D, 2D), with three trace levels \((T_2, T_3, T_4)\), three total trial sizes \((n = 192, 384, 768)\), and three participant sample sizes \((N = 8, 16, 32)\), for each of the four comparisons (non-trace, no-overlap, unidimensional, multidimensional), a total of 216 combinations \((33 \times 2 \text{ models} \times 4 \text{ comparisons})\). For each of the 500 repetitions of the 216 combinations we estimated group Bayes Factors, and then calculated the proportion of comparisons selecting one of the four models (i.e., where the models posterior probability was greatest amongst the set of four models), with results shown in Figure 2.3.

For the no-overlap model the group Bayes Factors results were much the same as for the individual analysis, except that the inconsistent effect of sample size for the individual analysis of the 1D data generating model disappeared in the group analysis. For the trace model performance was excellent when \(n = 768\) but the wrong (non-trace) model received increasing support when there were fewer observations per participant for all but the \(T_2\) design. These problems with the trace model caused corresponding failures to identify the correct dimensionality for lower values of \(n\), whereas for \(n = 768\) performance in identifying dimensionality was similar to that of the largest samples sizes in the individual analysis. In particular, the 2D data generating model was almost perfectly identified, but with higher \(T\) designs being slightly better, whereas performance in classifying the 1D data generating model was very good for \(T_2\) designs but decreased markedly for the \(T_3\) and \(T_4\) designs.
Figure 2.3. Group level results for the 216 comparisons. The rows and columns represent the same data generating models and comparisons as Figure 2. On each plot, the $x$ axis represents the three levels of $N$ that were resampled for each of $n = 192, 384, 768$, and the $y$ axis represents the proportion of cases in which the posterior model probability at the group level favoured the model specified for each column.
Conclusions

We aimed to investigate the ability of Prince, Brown and Heathcote’s (2012) Bayesian state-trace analysis to identify latent dimensionality. The results of individual participant data indicated that large sample sizes produced strong support for the correct outcome for both 1D and 2D data generating models across designs with two, three and four levels in the trace factor. Classification for the 1D data generating model was most reliable in designs with two trace levels, whereas the opposite tendency was evident for the 2D data generating model; dimensionality assessment was more accurate with larger numbers of trace levels. Overall these results indicate that a design with three trace levels provides the best compromise for accurate diagnosis of both single and multiple latent variable data-generating models.

We also explored a group analysis procedure that is advantageous where it is practically difficult to obtain a large number of responses from each individual participant, such as in cases where the number of available stimuli is limited, but where larger numbers of participants are available. Generally, this method was found to be very effective in identifying the 2D data generating model. However, our results indicate that it should be used with caution as it could be biased against detecting cases in which only one latent variable is present in certain experimental designs. When each participant contributed a smaller number of responses (192 or 384) results could be inaccurate even for the largest number of participants (32). For 768 observations per participant performance was more accurate and improved with group size for the 1D data generating model. In contrast to the individual participant results, the group level analyses indicate that designs with two levels in the trace factor produce the best compromise of most accurate classification across number of trials per participant and
different numbers of participants for both 1D and 2D data generating models. However, these results should be used with some caution given the three and four trace level designs demonstrated a large proportion of cases supporting the non-trace model (possibly due to the small experimental effects of the trace factor in these larger trace level designs), which had strong consequences for the correct classification of dimensionality.

Our individual and group analyses indicate that the ideal number of trace levels in a state-trace experiment is dependent on the intended approach to data collection. If only a small number of trials per participant are obtainable it seems wiser to use a trace factor with few levels so as to maximise data per point, and then combine across participants with group Bayes Factors. In contrast, if many trials per participant can be obtained, correct classification of dimensionality is possible with a three level trace factor through individual participant analysis, which confers additional benefits such as the exploration of individual differences in performance.

In summary, these results indicate that the success of a Bayesian state-trace analysis, and likely any state-trace analysis method, depends strongly on the particular model producing the state-trace plot. This highlights a caveat on our group analysis, which assumes all participants have an identical underlying model (rather than just having the same dimensionality but possibly different magnitudes of the effects of experimental factors). As well as being unrealistic, this assumption likely magnifies the effects of a particular data pattern. It would, therefore, be advantageous for future research to simulate groups of participants that vary in the effects of experimental factors (while maintaining a consistent dimensionality) in order to check the generality of the group analysis results reported here.
Chapter Three

An R Package for State-Trace Analysis

Adapted from:

StateTrace

Although the previous two chapters attempted to make state-trace analysis more accessible to researchers who could benefit from this technique, it would be remiss to ignore the fact that those who are unfamiliar with state-trace analysis are likely also to be unfamiliar with Bayesian methods. In an effort to aid the broader adoption of these methods we developed a freely available package, StateTrace, that implements Prince, Brown and Heathcote’s (2012) methods using computationally intensive posterior sampling to perform Bayesian estimation and model selection. In this chapter (Prince, Hawkins, Love & Heathcote, 2012a) we describe the scope and capabilities of StateTrace, which is written for the R language (R Development Core Team, 2011). R is freely available for Windows, Mac OS X and Linux, and can be downloaded from http://www.r-project.org/. The StateTrace package can be installed within R by typing install.packages("StateTrace") on the command line and the package functionality is made available by typing library("StateTrace"). StateTrace has been tested using the standard Rgui on PCs running Windows 7 and XP, as well as the standard R console under Mac OS X and Linux systems. Note however, that problems can sometimes occur with other commonly used interfaces (e.g., RStudio) and therefore we do not recommend their use with this package.

StateTrace is designed for users with no experience implementing Bayesian analyses. Functions can be accessed through a guider user interface (GUI), so it is also suitable for users with relatively minimal experience of R. The GUI functionality is provided by Hoffman and Laird's (2009) fgui package. It is important to note however, that the aim of this chapter is to provide an overview of the capabilities of StateTrace, including the type of statistical summary results and graphical output that can be
obtained. Before using the package we strongly recommend that users work through the detailed instructional tutorial provided in the “vignette” document that accompanies the StateTrace package (Prince, Hawkins, Love & Heathcote, 2012b – Chapter Four). This vignette provides a step-by-step guide on how to analyse two example data sets, including explanations of each argument available in a function as well as the argument values required to reproduce the example output presented and further detail regarding how to interpret the various output as one progresses through the analysis. This vignette was developed based on our experience using the package with undergraduate research students. Once the package is installed the vignette can be accessed within R by typing vignette(topic = "StateTrace", package = "StateTrace").

Although designed for users unfamiliar with Bayesian techniques, experienced users otherwise competent in these sampling methods may still benefit from StateTrace in that it provides facilities that manage potentially large demands on computer time and memory. With these facilities, analysis of appropriately designed experiments is practical on commonly available personal computers. StateTrace also provides convenient tabular and graphical methods for examining and summarising results, including customizable state-trace plots, that address both the individual and group levels of analysis. In the next section we discuss the types of experimental designs and data types that StateTrace accommodates. We then describe its statistical and graphical capabilities.

**Design and Data**

StateTrace can analyse three-factor (2x2xN) repeated-measures experiments that yield a binary dependent measure, including measures calculated either with or without reference to a measured baseline. Accuracy quantified by the difference
between a hit rate and false alarm rate are examples of the former type of measure. Proportion correct is an example of the latter type of measure. As state-trace analysis is not affected by monotonic transformations, equivalent results are produced for related measures, such as $d'$ from signal detection theory. The example data sets presented in this chapter are taken from recognition memory experiments and address both types, as measured by testing single items that were either studied or not (“yes-no” testing), or by two-alternative forced choice (2AFC) testing.\footnote{Although not included in the published manuscript, see Appendix C for details regarding the methods for the yes-no experiment. The 2AFC data is reported in Chapter Seven (Prince & Heathcote, 2013) as the ‘Matched Test’ design.}

Prince, Brown and Heathcote (2012) label the three factors as state, dimension and trace factors. In this section we explore the nature of each type of factor and the types of designs that StateTrace can and cannot analyse. A global limitation is that all three factors must be of the repeated measures type, as Prince, Brown and Heathcote’s (2012) methods analyse each participant’s data separately, and then combines the individual results to address the group level. In the final section of this chapter we discuss these limitations in more detail and describe how they can currently be addressed using StateTrace.

**State and Dimension Factors**

The state factor has two levels that constitute the axes of the state-trace plot. In our example data (as in the previous two chapters) the two levels are based on recognition memory decisions made about pictures of houses versus pictures of faces. In the case of a dependent-variable state trace analysis these levels correspond to different binary dependent measures. For example, they might be the outcome of a recognition decision and a classification of the decision as being made with high vs. low confidence.
The dimension factor has levels corresponding to different experimental manipulations. In our examples, the manipulation corresponds to whether the study and/or test presentations used upright or inverted images. The dimension factor can be thought of as interacting with the state factor in a way that changes the dimensionality of the processes underlying performance. For example, upright faces are thought to be able to be encoded both in terms of their constituent features and in a more holistic or relational way, whereas only the former encoding is available for inverted faces (Maurer, LeGrand & Mondloch, 2002; Valentine, 1988; Yin, 1969).

The state and dimension factors correspond to the two factors examined in a dissociation analysis. For example, traditional dissociation-based evidence for faces being processed in a qualitatively different way to other visual stimuli is provided by an interaction between the class of stimuli (e.g., face vs. house) and presentation orientation (e.g., upright vs. inverted). For a dependent-variable state-trace analysis, assignment of the two factors to state and dimension roles is clear, but otherwise these roles can be interchangeable. An exception concerns measures assessed against a baseline. For such measures the levels of the dimension factor must have a common baseline for state-trace plot monotonicity to be diagnostic of dimensionality. As this restriction does not apply to the state factor, assignment can be made accordingly (as was the case with our baseline example), or a measurement method that does not include a baseline must be used (e.g., 2AFC). One of our examples has no baseline (a “B0” design) and the other example has a separate baseline for each level of the state factor (a “B2” design). In the latter case the baseline conditions correspond to test house or face images that were not studied.

StateTrace assumes the dimension factor has two levels based on practical motivations related to the size of (i.e., number of cells in) a design. First, the
computational method used by **StateTrace** for the Bayesian analysis rapidly becomes prohibitive as size increases. Second, for large designs it is difficult to run enough trials per cell to obtain estimates that are sufficiently precise to support individual participant analysis. Hence, using a two-level dimension factor is prudent, perhaps after piloting to select appropriate levels, as this is typically sufficient to test dimensionality.

**Trace Factor**

The third experimental manipulation, the trace factor, has no analogue in dissociation analysis. In Prince, Brown and Heathcote’s (2012) method, the effect of the trace factor on dimensionality is not of interest. Indeed, the trace factor is chosen specifically because past research indicates it is unlikely to affect dimensionality, so that dimensionality evidence can be unambiguously interpreted in terms of the state and dimension factor effects. In our examples the trace factor was the time spent studying each item. Given past research indicates that increasing study time increases accuracy for houses and faces both when they are upright and when they are inverted, study time should have a monotonic effect in the state-trace plot. For the same reasons that apply to the dimension factor, the trace factor cannot have different baselines for different levels.

The trace factor’s role is to ensure that state-trace analysis is diagnostic of dimensionality. A state-trace plot can be non-diagnostic for two reasons. When state and dimension factors have only two levels and there is no trace manipulation the state-trace plot has only two points, and so is always monotonic. However, if the plot has more than two points it can still be non-diagnostic if the dimension factor effect is so large that results do not overlap on either axis of the plot. The remedy is to induce overlap using the trace factor. In our examples, greater accuracy for the upright
condition is counteracted by a longer study time for the inverted condition, inducing overlap.

**StateTrace** can accommodate any number of trace-factor levels. Prince, Brown and Heathcote (2012) recommend three to four levels with evenly spaced effects. Too many levels can cause the sort of design-size related problems discussed previously. Too few levels, and unevenly spaced effects, risk the state-trace plot being non-diagnostic due to an unlucky configuration of estimated points even when overlap is achieved. Choosing an appropriate number and spacing of trace factor levels can require some calibration through pilot testing. When doing so it is important to recognize that it may be most efficient to use different trace-factor levels within each level of the dimension factor (i.e., the design need not be fully factorial). For example, we used generally longer study times in the inverted than upright condition in order to counteract generally greater accuracy for upright than inverted items. A trace factor can sometimes be created by the post-hoc construction of a factor according to criteria set to obtain the most diagnostic outcome. For example, Heathcote, Freeman, Etherington, Tonkin and Bora (2009) constructed a post hoc trace factor in a recognition memory experiment by dividing test trials on study-test interval.

**Input Data Formats**

**StateTrace** reads in text data files made up of equal numbers of entries on each line. Data files may contain *trial data*, with a column indicating the response on each trial, or *summarized data*, with columns indicating the number of correct responses and number of trials for each condition. **StateTrace** accepts either a single file with all participants’ data or individual participant files. Columns can be delimited in a variety of common ways, and relevant columns can be selected from a larger set either by name
or number. The relevant columns indicate the response (number of trials for summarized data), and the state, dimension and trace levels, for each row.

StateTrace accepts data files formatted as coming from designs either with no baseline (B0) or with a different baseline condition for each state level (B2). As baseline conditions cannot differ over dimension and trace levels a B2 design is automatically detected through the presence of blank or ‘NA’ entries in these columns. Designs where accuracy in all conditions is measured relative to a single baseline can be treated as coming from either a B0 or B2 design. In the former case the baseline data are omitted, whereas in the latter case they are included twice. The outcome of state-trace analysis is the same in both cases, but the B2 format may be preferred as accuracy results can be displayed by differences between non-baseline and baseline results.

**Statistics and Graphics**

Figure 3.1 shows example state-trace plots for our 2 (state: face, house) x 2 (dimension: upright, inverted) x 3 (trace: study duration) design for both individual participants (Figures 3.1a and 3.1b) and the group aggregate for the no baseline example (Figure 3.1c) and for the baseline example (Figure 3.1d). Plots such as these can be visually inspected to assess whether all of the points fall on a monotonic function: that is, whether the order of points on the x-axis is the same as the order of points on the y-axis. However, inference about dimensionality based on visual inspection of state-trace plots can sometimes be misleading due to measurement noise. This is particularly the case for individual participant data (e.g., Figures 3.1a and 3.1b) where levels of measurement noise can be high. However, individual analysis is required to make strong inferences based on state-trace analysis because neither the monotonicity nor non-monotonicity of state-trace plots is necessarily preserved when they are averaged.
over participants (Prince, Brown & Heathcote, 2012). Hence, a quantitative approach to state-trace analysis provides an important complement to visually assessing a state-trace plot.

StateTrace obtains evidence about questions relevant both to refining an experimental design and diagnosing dimensionality by selecting among four mutually exclusive models. In defining these models it is convenient to refer to ‘data traces’, lines joining data points from the same level of the dimension factor (e.g., the solid and dashed lines in Figure 3.1a and 3.1b):

1. **Non-trace model**: the trace factor does not always have a monotonic effect (i.e., one or more data traces are non-monotonic)
2. **No-overlap model**: data traces do not overlap on either axis
3. **Unidimensional model**: a single latent variable mediates performance
4. **Multi-dimensional model**: more than one latent variable mediates performance

Each model specifies a set of order restrictions on the points in a state-trace plot. Evidence for each model is quantified using a Bayes factor (BF; Kass & Raftery, 1995). Bayes factors for the set of models can be combined to calculate posterior model probabilities, $p$, which quantify relative evidence. Additionally when the trace factor is selected based on strong prior evidence that it has a monotonic effect it makes sense to exclude the non-trace model when computing posterior model probabilities.
Figure 3.1. Accuracy state-trace plots showing modes of the posterior estimates from the encompassing model (large symbols) for the 2AFC data set for (a) participant ‘J’, (b) participant ‘R’, and (c) on average, as well as (d) on average for the yes-no data set. Accuracy is indicated by plotting the hit rate (HR) for the 2AFC data and \( d' \) (i.e., \( z(HR) - z(FAR) \), where FAR is the false alarm rate) for the yes-no example. For (a) and (b) the lines are data traces, joining posterior modes of the encompassing model samples, and ellipses represent the 50% credible regions. For (c) lines with small symbols join posterior modes of the trace model and for (d) they join the highest posterior probability (i.e., most frequently sampled) monotonic model. The ellipses in (c) and (d) represent 68% credible regions.
StateTrace can output both Bayes factors to quantify absolute evidence about whether the data provide sufficient support for clear conclusions about each model, and posterior model probabilities (calculated either including or excluding the non-trace model) to quantify relative evidence. We suggest (after Raftery, 1995) that evidence for the numerator model can be considered weak for a BF < 3, positive for a BF between 3 and 20, strong for a BF between 20 and 100, and very strong for a BF > 100. Table 3.1 provides similar conventions for posterior model probabilities. The two sets of conventions are closely related because $p = BF / (1 + BF)$, given BF = 1 by definition for the encompassing model. Like all conventions, these suggestions should not be used uncritically; Bayes factors and posterior model probabilities have a natural scale making their interpretation straightforward.

Table 3.1.

Conventions to aid interpretation of the posterior model probabilities (after Raftery, 1995).

<table>
<thead>
<tr>
<th>Favouring model</th>
<th>Against model</th>
</tr>
</thead>
<tbody>
<tr>
<td>$p &gt; .99$</td>
<td>Very strong evidence</td>
</tr>
<tr>
<td>$.95 &lt; p \leq .99$</td>
<td>Strong evidence</td>
</tr>
<tr>
<td>$.75 &lt; p \leq .95$</td>
<td>Positive evidence</td>
</tr>
<tr>
<td>$.25 \leq p \leq .75$</td>
<td>Equivocal evidence</td>
</tr>
</tbody>
</table>
Computing Bayes Factors

Prince, Brown and Heathcote (2012) used posterior sampling methods proposed by Klugkist, Kato and Hoijtink (2005) and Klugkist, Laudy and Hoijtink (2005) to compute Bayes factors for the four models. Although conceptually straightforward (i.e., they simply count the frequency with which different orders occur in a set of simulated samples), this method is computationally expensive. In order to make it practical,

StateTrace uses two types of sampling:

1. **Encompassing sampling**: Monte-Carlo (MC) sampling, used to obtain samples from a model that makes no order constraints, and

2. **Trace model sampling**: Markov-Chain Monte-Carlo (MCMC) sampling, used to obtain samples under the trace model constraint.

In principle only MC methods are needed, but the yield of samples relevant to models (2) – (4) is often so low that it would be far too inefficient. Although slower per sample, the MCMC method always yields samples from models (2) – (4) after an initial “burn-in” period, and so is better in practice.

For trace model sampling, initial (burn-in) samples are discarded because it can take some time for the MCMC process to converge to the target distribution (see Gilks, Richardson, & Spiegelhalter, 1996). Our experience is that convergence is very fast and that at most 100 initial samples need to be discarded. However, this may not be the case in all applications, so StateTrace provides facilities to check. StateTrace uses Plummer, Best, Cowles and Vine's (2006) coda package to automatically calculate one check, Gelman’s “R-hat” statistic (where values close to one indicate convergence), and also allows MCMC samples to be exported in a format suitable for further checks provided by coda.
**StateTrace** manages the sampling process, enabling estimates to be automatically refined to a specified level of accuracy, and for the time spent sampling to be limited to convenient periods (e.g., overnight runs). Statistics to support simultaneous selection among all four models (an *exhaustive strategy*) or only models (2) – (4) (*a trace-true strategy*) are calculated automatically and raw counts can also be accessed to support other approaches, such as sequential model selection (see Prince, Brown & Heathcote, 2012).

**Overview of Functions**

Typing `guista()` at the R command line invokes the main **StateTrace** GUI, which provides access to the main **StateTrace** functions: `stFirst`, `stSample`, `stSummary`, `stProbplot`, `stBootav`, `stPlot` and `staManage`. Alternatively, the GUIs for each separate function can be called by typing “gui” followed by the function name and parentheses (e.g., `guistFirst()`). In general, command line users can remove "gui" from the start of the function and enter argument values within the parentheses. Details on available arguments can be obtained by consulting the function’s help, called by typing a ? prior to the function name. Each of the GUIs provided allows users to view default values for function arguments and to enter alternate values via widgets such as text entry boxes, slider bars, true/false check boxes and multi-option lists. In all GUIs, argument values that require input for a function to run are marked by an ‘*’ and argument descriptions that contain the term “character string” must have their values contained in double quotation marks.

The **stFirst** function performs an initial analysis, first reading in data for one or more participants from one or more text files, and then making a quick preliminary assessment of the results based on a limited number of posterior samples. It then creates
an object of class \textit{sta} in the R environment, which is named by the user. The \textit{sta} object encapsulates the data and the numerical results based on sampling. Once \texttt{stFirst} is complete the \textit{sta} object can be saved from the R environment to a file in a compressed format and restored in a later R session using the R \texttt{save} and \texttt{load} functions respectively.

Three \texttt{StateTrace} functions allow the contents of an \textit{sta} object to be displayed. The \texttt{stSummary} and \texttt{stProbplot} functions provide, respectively, tabular and graphical summaries of the Bayesian analysis. The \texttt{stPlot} function makes state-trace plots (e.g., Figure 3.1). All three functions work, to some degree, with an \textit{sta} object created by \texttt{stFirst}. However, for more accurate results from the Bayesian analysis, and to access the full range of state-trace plot options, two other functions may have to be run: \texttt{stSample} and \texttt{stBootav}. These functions perform time-consuming computations whose results are stored in the \textit{sta} object. The \texttt{stSample} function collects enough extra samples to reach a specified level of accuracy in the Bayesian analysis. Some of these extra samples can also be stored in the \textit{sta} object so that a line representing the trace (e.g., Figure 3.1c) or monotonic (Figure 3.1d) model that is best supported by the data can be added to a state-trace plot. However, to enable these lines to be added to state-trace plots averaged over participants, the \texttt{stBootav} function must also be run.

The main GUI also provides access to the \texttt{staManage} function, which allows users to manage and export posterior samples stored in an \textit{sta} object. An \textit{sta} object is an R list that can be directly accessed by users, but \texttt{staManage}, and the other functions, are designed so this should not be necessary. Samples are stored in an \textit{sta} object to enable efficient generation of graphical summaries of uncertainty in estimation (i.e., credible regions, the Bayesian analogue of confidence intervals, e.g., the ellipses in Figure 3.1). However, this can sometimes cause an \textit{sta} object to become so large that it
is slow to load and save. The `staManage` can be used to remove samples after the
graphics have been generated. The `staManage` function also allows users to export a list
containing posterior samples with one entry for each participant. Each entry in the list
contains samples in a format (the `mcmc.list` class) for which `coda` provides many easy to
use analysis methods (e.g., `plot` and `summary` functions).

**First Steps**

Before `stFirst` initiates sampling, checks are run to ensure the selected data files
are compatible with `StateTrace` (e.g., state and dimension factors each have only two
levels) and the multiple files are compatible with each other (i.e., all have either a B0 or
a B2 design). It then obtains 100,000 MC samples for each independent set of design
cells (four sets for B2 designs and two for B0 designs) and determines the proportions
that follow the order specified by the trace factor. Estimates of 95% credible intervals
for each proportion estimate are obtained, and where the precision of the interval is less
than 0.0005, it marks sampling as complete. Otherwise it estimates the time required to
get enough samples to narrow the interval sufficiently, based on the time required for
the initial 100,000. These time estimates are only approximate and will vary, especially
if sampling is completed on a different computer that is faster or slower.

Next an order constrained Gibbs sampler (Gelfland, Smith & Lee, 1992) is used
to draw two sequences of 5,000 MCMC samples (“chains”) from the trace model for
each participant. The proportions of samples following the no-overlap, uni-dimensional
and multi-dimensional model orders are tabulated, and the corresponding 95% credible
intervals calculated. Sampling for a participant is marked as complete if all intervals are
less than 0.005. Otherwise the additional time required to complete is estimated. A
smaller interval criterion is used for encompassing than trace sampling as encompassing
proportion estimates have greater potential to reduce precision overall because they multiply the trace proportions in the calculation of Bayes factors. Once the two chains are complete, stFirst reports whether the MCMC process has worked properly (i.e., has “converged”) using Gelman’s multivariate R-hat statistic.

In the final stage of computation stFirst draws 10,000 bootstrap average samples and uses the two-dimensional density estimator provided by Wand and Ripley’s (2009) KernSmooth package (with its default parameters) to calculate the posterior modes (measures of central tendency) and 68% credible regions (i.e., the analogue of a standard error) around the modes. These calculations are used to display an average state-trace plot, which provides the user with an immediate view of results averaged over participants, as well as state-trace plots for each individual participant. Additionally, the corresponding posterior mode estimates are output in a tabular form to the R console. The stFirst function can be called repeatedly to add additional participants to the sta object. A warning will be issued if duplicate data sources are mistakenly specified; however, this data will still be added to the object without replacing the old entries.

Refining Estimates

Further sampling may be required if the analysis did not reach completion during the initial pass or the user wishes to alter the credible interval precision criteria from those used by stFirst. This is achieved using stSample, which allows fine-grained control over the defaults used by stFirst. The only required input is the name of an existing sta object. Because obtaining enough samples to fulfil stricter criteria can be time consuming stSample has a ‘refresh’ mode, which allows the predicted time to completion to be calculated for different criteria. This refresh mode is fast to run as no
actual sampling is done, unless none has been done yet, in which case a single pass is made. As the time information will vary depending on the computer used, if the most recent pass (e.g., obtained by running \texttt{stFirst}) was on a different computer it is useful to turn off the refresh mode but leave the maximum run time at zero; this will cause a single pass to be run and to update the timing information for the new computer.

Once a sampling plan is determined the refresh mode can be turned off, a suitable maximum run time entered and sampling initiated. Sampling is completed in a series of passes, and users may choose to sample only from the encompassing model, trace model, or both. Sampling terminates when precision criteria are satisfied, so \texttt{stSample} may require additional time than that estimated from previous passes (e.g., after running \texttt{stFirst}), it may also complete before the maximum time elapses and one type of sampling may complete before another. The \texttt{verbose} argument (with values 0, 1, or 2) controls information printed to the R console during sampling: 0 is silent, 1 prints the estimated total time remaining after each run for all participants, and 2 adds timings per participant.

The number of samples for each pass of each type of sampling is chosen to satisfy a trade-off. Using a small number per pass inherits a cost in housekeeping between passes and initial \textit{burn-in} samples on each trace-model run are lost. A large number uses more RAM and can result in more samples being taken than required to achieve the required precision. A larger value is advisable for encompassing than trace model sampling, as encompassing sampling is usually an order of magnitude faster. In our applications we have found the defaults work well and that little is gained in particular cases by altering them. Similarly, we have found the default criteria (credible interval type, e.g., 95%, and precision) strike an appropriate balance between
computation time and the accuracy of Bayes factor and posterior model probability estimates.

A second reason for running `stSample` is to collect samples that enable visualization of each model; that is, samples that follow the order(s) dictated by a model. We have found the default value of 10,000 encompassing samples collected by `stFirst` is sufficient for accurate visualization of central tendencies and credible regions, although large regions (which require estimation of distribution tails) can require more. The 10,000 trace samples collected by `stFirst` are also usually more than sufficient given that they are only used to estimate central tendency. The `stSample` function also collects a particular type of trace sample, monotonic model samples, which may be relatively rare, especially when the data are far from monotonic. Monotonic samples are used to plot the central tendencies of the uni-dimensional or no-overlap models, with the latter type of sample often being extremely rare unless the data are strongly non-overlapping. Given this, by default `stSample` keeps all monotonic samples.

Storing large sets of samples for each participant can greatly increase the size of `sta` objects. The default values (assuming stored monotonic samples are not allowed to grow too large) are rarely problematic. The `staManage` function can be used to reduce the number of stored samples where problems arise, including the ability to keep only samples for the “best” (i.e., most frequently occurring, and hence most probable) monotonic order rather than all samples with monotonic orders. The `staManage` function can also be used to join multiple `sta` objects; for example, it can be computationally efficient to divide a very large set of participants and then run the sampling for sub-groups of participants on separate machines, after which the `sta` objects can be combined and the group aggregate results examined.
Extracting Results

The `stSummary` function produces tabular model-selection results to the R console. It also provides information about the status of sampling for an `sta` object (i.e., whether it is complete or if not, how much more computation time is required according to the criteria stored in the object). Results can be output as Bayes factors or posterior model probabilities based on either exhaustive or `trace-true` strategies. By default, `stSummary` reports results summarized over participants based on group Bayes factors, which are the product of each participant’s Bayes factors and assumes each participant contributes independent evidence (Prince, Brown & Heathcote, 2012). However, in some cases the group results can be inappropriately influenced by outlying individual participant results (e.g., most participants are uni-dimensional but a few strongly multi-dimensional participants dominate the group results). We recommend that users also output and examine individual participant results in order to check this possibility. This can be done, and outlier participants can then be excluded from the calculation of group results, using `stSummary` options. Users may also output a large range of additional results, including the prior probabilities for each model (calculated analytically, see Prince, Brown & Heathcote, 2012), the total number of encompassing and trace model samples and counts of the number of times the orders specified by each model were sampled.

The `stProbplot` GUI allows the distribution over participants of posterior model probabilities to be inspected graphically. Outlying participants can be excluded, and the annotation within the plots customised. Additionally, the posterior probability for the group (based on the group Bayes factor) can be displayed numerically in the title and as a line on the plot. As shown in the output for the 2AFC example data in Figure 3.2, the `stProbplot` plot includes a panel for each model and each letter in a panel represents the
posterior model probability for a single participant. Participants in each panel are sorted by their results, allowing those with extreme values to be easily identified. Figure 3.2 was made using defaults, which produce appropriate annotations in most cases, and by choosing the option to use letter plot symbols in order to easily identify each participant’s results.

Inspection of Figure 3.2 suggests that participant ‘J’ provides an outlying result in favour of the non-trace model. However, follow-up analysis excluding participant ‘J’ revealed little influence on the group posterior model probability ($gp$ in the panel titles). Overall these results show positive or greater evidence for the trace model and for data-trace overlap (i.e., low probabilities for the non-trace and no-overlap models). Evidence is weaker and individual variability greater in relation to the dimensionality results, but the group evidence supports a multi-dimensional outcome. The $stProbplot$ plots can also reveal a potential mixture of uni-dimensional and multi-dimensional sub-groups, but that is not indicated in Figure 3.2.
Figure 3.2. Posterior model probabilities for each participant (denoted by letters) for the 2AFC example data for each of the four diagnostic models (panels). Group posterior model probabilities for each model are indicated in panel titles and plotted as a heavy dashed line. Within each panel, participant results are sorted in ascending order of their probability estimates and faint dashed lines demarcate the categories in Table 3.1.
Figure 3.1 shows examples of state-trace plots produced by stPlot. The stPlot function can represent accuracy data by the mode, mean or median measures of central tendency applied to samples from the encompassing model. As the encompassing model makes no order assumptions, these measures (e.g., large symbols in Figure 3.1) provide a model-free estimate of the observed data. The default choice used to create Figure 3.1 (the mode) produces estimates that are usually equivalent to the familiar maximum-likelihood estimator (e.g., $n/N$ for the 2AFC hit rate, where $n$ is the number of correct responses in $N$ trials).

The other central tendency measures usually produce similar results, at least for reasonable sample sizes not subject to floor or ceiling effects. For example, for the hit rate and uniform prior used by StateTrace, the mean of a large sample from the encompassing model is equivalent to $(n+1)/(N+1)$. For other accuracy measures such simple formulae are not available. This is also the case for any accuracy measures for any of the order-restricted models. Hence, estimates based on samples have the advantage of providing an easily applied and general approach.

The stPlot function uses the same type of approach to display the degree of uncertainty in central tendency estimates, by drawing contours around regions containing a specified percentage of the posterior encompassing-model samples. Estimating Bayesian credible regions in this way works with all accuracy measures in a way that takes account of any floor and ceiling effects, which can be very influential when contours are near bounds in an accuracy measure. The regions, and modes, are estimated using the bkde2D function in Wand and Ripley’s (2009) KernSmooth package, which is included by default with R. Users can choose the percentage contained by the regions and the degree of smoothing, as a multiple of the maximum over data points of the values provided by the dpik function: this KernSmooth
function, and \texttt{bkde2D}, are called with default values. The default multiplier of five used by \texttt{stPlot} was chosen to produce very smooth contours even for large regions, which can otherwise be irregular because they require estimation of distribution tails; users are encouraged to experiment with the multiplier as appropriate for their application.

Figure 3.1a and 3.1b plots results for participant ‘J’, who had the strongest evidence for a violation of the trace model in Figure 3.2, and participant ‘R’, who had the strongest evidence for the multi-dimensional model; both state-trace plots are clearly consistent with the model-selection analyses. Individual participant data are typically quite noisy, so for clarity the credible regions in these plots contain only 50\% of the posterior samples. Both data sets display strong data-trace overlap, consistent with the results for the no-overlap model in Figure 3.2.

Figure 3.1c is a state-trace plot of the average over all participants in the 2AFC example data. Reflecting the reduction in uncertainty associated with an average, the credible regions are much smaller in this case, even though they contain the default value of 68\% of the posterior samples. The lines in Figure 3.1c join the modes of the average of samples from the trace model. The points joined by the lines are different from the large symbols, which are estimated based on encompassing model samples, as the encompassing model admits samples that violate the trace model. However, the difference is not large, reflecting the fact that for most participants the trace model provides an excellent description of this data.

Figure 3.1d plots average results for the yes-no example using the signal-detection theory \(d'\) measure of accuracy. The lines in Figure 1d join the modes of the most commonly occurring order\(^6\) for monotonic MCMC samples (the \textit{best} monotonic

\(^6\) It is important to note that the best order may differ between participants. Before interpreting the best (most frequently occurring) monotonic model in the average, such as is plotted in Figure 3.1d, it is advisable to use \texttt{stSummary} to examine the degree of variability in the best orders over participants, as strong individual differences may mean that taking an average is not sensible.
model); that is, MCMC samples from either the uni-dimensional or no-overlap models. Because this data is well described by a one-dimensional model the difference between the best monotonic model and encompassing model modes is relatively small.

Figures 3.1c and 3.1d were created after first running \texttt{stSample} to the default criterion then \texttt{stBootav} to average the trace and monotonic samples stored by \texttt{stSample}. The \texttt{stBootav} function allows users to choose to calculate bootstrap averages (based on the stored samples for each participant) from one or more of the encompassing, trace and monotonic models. A set of bootstrap averages is created by repeatedly randomly selecting with replacement one sample from each participant’s set of posterior samples for a given model and taking their mean. Each time \texttt{stBootav} is invoked it can compute averages for only one type of accuracy.

Monotonic samples may be relatively rare for some data (e.g., the 2AFC data, as it is strongly non-monotonic), and so the averages can be unreliable; to alert users to this possibility \texttt{stBootav} reports the number for participants who have less than 100 samples. A participant with no monotonic samples is automatically excluded from the average. The problem of a lack of monotonic samples might be addressed by calling \texttt{stSample} again with a stricter criterion, but usually a lack of monotonic samples indicates the monotonic model is not appropriate for the data and so there is no point in plotting it.

Once averages are stored in an \texttt{sta} object, \texttt{stPlot} can make a corresponding average state-trace plot (e.g., Figure 3.1d). Data points in average plots represent the central tendency of the set of bootstrap averages. Variability among the averages is used to construct credible regions in the same way as for individual participants. These credible regions reflect uncertainty in the estimated average over the particular set of participants in an experiment. The \texttt{stBootav} function also allows participants to be
selected at random with replacement on each bootstrap repetition; this produces a set of averages with the same central tendency but greater variability that is appropriate when the participants are treated as a sample from a population.

**Limitations and Future Directions**

*StateTrace* is limited in a number of ways that we plan to address: it requires a fully repeated measures design, allows only two levels for state and dimension factors, and works with only binary data. This is not to say that it is not both valid and useful to perform a state-trace analysis involving between-subjects factors (e.g., does the dimensionality of memory differ between amnesiacs and controls?). However, statistical analysis of such state-trace plots requires estimation of a population level model, and so would require a hierarchical extension of Prince, Brown and Heathcote’s (2012) approach. Bayesian methods are suited to this extension and we hope to pursue it in future work. Until then we recommend Verhaeghen and Cerella’s (2002) multilevel approach to address such designs, although it requires the assumption of a functional form (e.g., linear) for data traces.

In principle, both state and dimension factors may have more than two levels. We are currently testing a fast approximation for the Bayesian analysis, which removes the computational reason for having only two dimension levels. Hence, this restriction will likely be removed as part of an update incorporating the fast approximation. An extension beyond two state levels requires more fundamental changes, and although applicable methods have been developed (Dunn & James, 2003), further work is required to extend our Bayesian analysis.

For now, when users are interested in the dimensionality underlying the relationship between three state-factor levels (e.g., A, B and C) or more they can use
**StateTrace** by taking advantage of the transitivity of pairwise state-trace inference (e.g., test A vs. B and B vs. C, with one-dimensional results in both cases indicating a single latent variable explains variation in all three). A similar approach can be required if more than two dimension-factor levels are required. Finally, the limitation to binary data can be accommodated by collapsing (e.g., a 1-10 confidence rating could be collapsed to high vs. low), but subject to the usual caveats about loss of information. We plan to explore the extension of Prince, Brown and Heathcote’s (2012) Bayesian analysis, and **StateTrace**, to such finer-grained measures using multinomial data generating assumption.
Chapter Four

Using StateTrace

Adapted from:

Using StateTrace: An Example Analysis

In the previous chapter a general introduction to the StateTrace package was provided. However, we acknowledge that Chapter Three provides insufficient information to allow a user to make full use of the StateTrace functionality. The aim of this chapter is, therefore, to provide a detailed step-by-step tutorial that illustrates how to analyse the two example data sets presented in Chapter Three. These detailed instructions include points on how to use R, explanations of the arguments comprised in each function, the argument values required to reproduce the output presented in Chapter Three, as well as guidance for interpreting additional output that is produced as one progresses through the analysis.

As described in Chapter Three, the StateTrace package runs under R, a free software environment for statistical computing and graphics (R Development Core Team, 2007). R is available for Windows, Mac OS X and Linux, and can be downloaded from http://www.r-project.org/. When R expects an input command, it issues the > prompt. Experienced R users can execute the functions instantiated in StateTrace from the command line, call them from their own functions and modify and incorporate the code in their own functions as required, subject to the requirements of the software license agreement. Here we describe how to use StateTrace through a GUI (guided user interface) suitable for users less experienced with R, allowing them to view default values of function arguments and enter alternate values via widgets, such as text entry boxes, slider bars, true/false check boxes, and multi-option lists. The GUI functionality is provided by Hoffman and Laird’s (2009) fgui package.

StateTrace, fgui and coda are installed once into an instance of R by typing

> install.packages("StateTrace")
on the R command line. Subsequently, each time R is invoked `StateTrace`
functionality is made available by typing

```r
> library("StateTrace")
```

On some operating systems equivalent menu based methods can be used for both of these steps; for example, for Windows users a package can be installed by selecting

‘Packages > Install package(s) from local zip files…’ from the menu toolbar and then navigating to and selecting the downloaded file, while the functionality of `StateTrace` can be made available by selecting ‘Packages > Load Package…’ from the menu toolbar and selecting “StateTrace” from the displayed list.

The main `StateTrace` GUI (see Figure 4.1) can be invoked by typing

```r
> guista()
```

and provides access to each of the `StateTrace` functions: `stFirst`, `stSample`, `stSummary`, `stProbplot`, `stBootav`, `stPlot` and `staManage`. Clicking the corresponding function buttons will open further GUI windows. Alternatively, these GUlS can be called directly by typing “gui” followed by the function name and parentheses (e.g., `guistFirst()`). In general terms, command line users can remove “gui” from the start of the function and enter argument values within the parentheses (e.g.,

```r
stFirst(staname="DFIE.sta", fnams="DFIE.txt", multiparticipant=T)
```

Details on the available arguments can be obtained by consulting the function’s help documentation, which is called by typing a ? prior to the function name on the command line (e.g., `?guistFirst` or `?stFirst`).

---

7 Mac users, note that the default install of R does not include the required `tcltk` package, which will also need to be installed prior to running the next command to invoke the functionality of `StateTrace`. 

Overview of Functions

The `stFirst` function performs an initial analysis, first reading in data for one or more participants from one or more text files, and then making a quick preliminary assessment of the results based on a limited number of posterior samples. It then creates an object of class `sta` in the R environment, which is named by the user. The `sta` object is used to encapsulate data and the numerical results produced by posterior sampling as well as analyses of the posterior samples. Once `stFirst` is complete the `sta` object can be saved from the R environment to a file in a compressed format and then restored in a later R session using the R `save` and `load` functions respectively.

Three StateTrace functions allow the contents of an `sta` object to be displayed. The `stSummary` and `stProbplot` functions provide, respectively, tabular and graphical summaries of the Bayesian analysis. While the `stPlot` function makes state-trace plots. All three functions will work to some degree with an `sta` object just created by `stFirst`. However, for more accurate results from the Bayesian analysis, and to access the full range of state-trace plot options, two other functions may have to be run: `stSample` and
**stBootav**. These functions perform time-consuming computations whose results are stored in the *sta* object. The **stSample** function collects enough extra posterior samples to reach a specified level of accuracy in the Bayesian analysis. Some of these extra samples can also be stored in the *sta* object so that a line representing the trace or monotonic model that is best supported by the data can be added to a state-trace plot. Additionally, the **stBootav** function must also be run to enable lines representing trace and monotonic models to be added to state-trace plots averaged over participants.

The main GUI also provides access to the **staManage** function, which allows users to manage and export posterior samples stored in an *sta* object. An *sta* object is an R list that can be directly accessed by users, but **staManage**, and other functions, are designed so this should not be necessary. Samples are stored in an *sta* object to enable efficient generation of graphical summaries of uncertainty in estimation (i.e., credible regions, the Bayesian equivalent of confidence intervals). However, this can sometimes cause an *sta* object to become so large that it takes a long time to load and save. Hence, it can be necessary to use **staManage** to remove samples after the graphics have been generated. The **staManage** function also allows users to export a list containing posterior samples with one entry for each participant. Each entry in the list contains samples in a format (the **mcmc.list** class) for which **coda** provides many easy to use analysis methods (e.g., **plot** and **summary** functions).
Data Input Formats

**StateTrace** reads data from text files in two formats, both of which (a) can contain data from either one or more participants, (b) have the same number of columns in each row, (c) can contain a header row and (d) have initial columns containing numbers or character strings indexing the design cell referred to by that row. They differ in that:

1. *Trial data files* have a row for each trial ending with a binary response indicator (e.g., correct and error responses coded as 1 and 0 respectively)

2. *Summary data files* have a row for each design cell ending with the summed binary response frequencies and number of trials for each cell.

Figures 2a and 2b show our example 2AFC data in trial and summary formats respectively. For both formats, the first column \((P)\) contains identifiers unique to each participant, which may either be a numeric (e.g., \(1 : n\)) or character string (e.g., “MP”, “GH”) value. Where the file has data for only a single participant, this column is used to assign a participant identifier or it may be omitted. When omitted participants are given integer identifiers in the order files are read; however, file names are also stored and can be displayed later using the **stSummary** function. The next three columns contain indicators for the levels of the state \((S)\), dimension \((D)\) and trace \((T)\) factors. Again these levels may be specified as numeric values (e.g., \(1:3\) for a trace factor defined by three study duration levels) or as character values (e.g., “F” and “H” for a state factor defined by the type of stimulus presented, faces and houses). The only restriction for these indicators being that the trace-factor identifiers within each combination of state and dimension levels must sort (using R’s **sort** function) into the order assumed by the trace model to produce increasing performance.
Figure 4.2. Example 2AFC data files in (a) trial and (b) summary formats and (c) yes-no data in summary format. In (b) and (c), rows corresponding to one participant’s data have been highlighted.

Each of the two formats has a different structure for the remaining columns, with a total of six columns in summary files and five columns in trial files. For the trial format the final column \((C)\) contains the response made to each trial. For example, for 2AFC judgements (e.g., Figure 4.2a) this column records whether the response was correct (1) or an error (0). In contrast, if the experiment tested single items and therefore required a yes-no judgement, this column records a yes (1) or no (0) response choice.

For the summary format, the fifth column contains the number of “successes” \((n)\) per design cell. This corresponds to the summed number of correct identifications for 2AFC data (e.g., Figure 4.2b). For yes-no data this column contains the summed frequency for making a “yes” response (e.g., Figure 4.2c). In both cases, the sixth column for summary files contains the total number of observations \((N)\) per design cell. For example, in our 2AFC experiment participants attended three 1-hour sessions over which they completed 52 blocks of 18 study images and 18 test pairs, yielding 78 observations per design cell, as shown in Figure 4.2b.
For our 2AFC example data in summary format (Figure 4.2b), there is one row for each of the 2(state: face, house) x 2(dimension: upright, inverted) x 3(trace: study duration) = 12 design cells per participant. However, the summary yes-no data (Figure 4.2c) has an additional two rows per participant corresponding to the baseline conditions for each level of the state factor. In this experiment, images of faces and houses were presented either upright or inverted at study and then all tested upright in a yes-no recognition task. As a yes-no task was used, participants were required to respond to new items during the test phase and it is the false alarm rates (i.e., incorrect “yes” judgements; FAR) for the new faces and new houses recorded in the extra rows.8 Therefore, the yes-no summary data has a total of 2 x 2 x 3 + 2 = 14 rows per participant. This latter type of design is referred to by Prince, Brown and Heathcote (2012) as a state baseline, or B2, design. While the former type is referred to as a no baseline, or B0, design. As the baseline conditions do not have corresponding dimension or trace factor levels, these indicators can be left blank or specified as “NA”.

**StateTrace** uses the presence of such indicators to automatically distinguish B0 from B2 designs.

Note that no explicit provision is made for designs where accuracy in all conditions is measured relative to a single baseline. Such data can be treated as coming from either a B0 or B2 design. In the former case, the baseline data can be omitted (e.g., only the “yes” data is analysed). In the latter case, the baseline data are included twice. The outcome of state-trace analysis is the same in both cases, but the B2 treatment may be preferred as it enables accuracy to be displayed in graphs in terms of a difference between non-baseline and baseline results.

---

8 Note that unambiguous inference about dimensionality for yes-no responses requires that accuracy for both levels of the dimension factor be assessed against a common baseline (i.e., relative to the same new items). This is achieved in our yes-no example as all items were tested upright.
The data for our 2AFC example can be made available with a
\[ \text{data(DFIE)} \]
command, which will create an object called ‘DFIE’ in the R workspace and can be viewed using
\[ \text{fix(DFIE)} \]

To demonstrate StateTrace’s data input capabilities, this data needs to be saved in a text file outside of the R environment using
\[ \text{write.table(DFIE, file="DFIE.txt", sep="\t", row.names=F)} \]
which will save the data in a tab-delimited text file called ‘DFIE.txt’ in the current working directory. Note that the location of the current working directory can be obtained by typing \text{getwd()} on the command line. If the working directory needs to be changed, this can be done with a \text{setwd()} command and including the path of the desired directory within the parentheses (e.g., \text{setwd("C:/User/Desktop/statetrace")}). Alternatively, on some operating systems the working directory can be changed by selecting ‘File > Change dir…’ from the R menu toolbar and navigating to the desired directory.

**First Steps**

It is important to note that multiple passes of sampling from the data will likely be required, which will update user specified aspects of the analysis. The \text{stFirst} function therefore provides users with a function specifically designed for commencing the sampling process. \text{stFirst} is a wrapper for several other functions that are called with some arguments fixed at values minimally sufficient for an initial analysis. The GUI shown in Figure 4.3 illustrates the remaining arguments that can be set by the user. For this function (and all other GUIs presented) parameter values that must be set are
marked by a ‘*’. Note also that in all GUIs, parameter descriptions containing the term “character string” must have their entries enclosed in quotes. Chief among these is the sta object name, which must not begin with a number or contain any spaces; the stFirst function will create an sta object with this name in the R environment, overwriting any existing object without a warning. For this example analysis, we assigned the name “DFIE.sta”. It is not required for the name of this object to be followed by the extension ‘.sta’ however, we tend to use this naming convention for ease in distinguishing the sta object from other formats that may have the same name and be saved in the current workspace.

Figure 4.3. GUI for stFirst, which is used to read-in the raw data files, create an sta object, perform an initial pass of sampling, and create preliminary state-trace plots.

The following three stFirst arguments specify different ways of loading data. The first argument, Character string of directory + file name of data file/s, allows one or more directory and filename combinations to be entered as a character vector. If no directory is entered then R’s working directory is assumed. Otherwise, the R convention of a forward slash, /, should be used when specifying the path of a file; for example,
"datafiles/DFIE.txt" specifies a single file in the directory 'datafiles' below the working directory, or c("DFIE-P1.txt", "DFIE-P2.txt") specifies two files in the working directory. Alternatively, the next two arguments enable loading of all files contained in a specified directory (Character string of directory containing data file/s) that have a particular extension (Character string of file extension; by default *.txt). If all three of these arguments are left blank when stFirst is run, R’s choose.files() method of selection (through a file list dialog) is invoked for Windows and Mac users. Using this method, Windows users can specify multiple files by holding the ‘Ctrl’ key when selecting the data files, whereas Mac users may only select one file (note this method is not available on Linux).

The next five arguments specify the data file format. First, File delimiter, is used to indicate the way in which columns have been delimited (note this argument requires input for stFirst to run). This value can be specified by clicking the button and selecting an option (“tab”, “space” or “comma”) from the displayed list. The next button, Selected data files each contain data for multiple participants?, indicates whether the data file(s) contain data for multiple participants (“T”) or a single participant (“F”). It can also be specified whether the data files contain a header row by checking either true (“T”) or false (“F”) for the argument Header row in data files?

Data files will also often contain more information than required by StateTrace and therefore the argument, Columns to use from each data file, specifies the relevant columns in the order of participant identifier (if included), state, dimension, trace and response columns either by indicating the column position (as an integer vector) or column name (as a character vector). This can be done using R’s combine function: c(...), where … are the column numbers or names to be included; for example, c(1,2,4,7) could be used for a trial data file that does not have a participant identifier
column but does have additional non-relevant columns to be excluded, or $c(3, 1, 2, 4)$ could be used for a trial data file with no participant identifier column that had the state, dimension and trace columns in the incorrect order. The next argument, *String for empty cells in data file/s,* allows any non-relevant rows to be excluded from analysis depending on the contents of the last ($C$) or second last ($n$) columns for trial and summary formats respectively. For example, in a recognition memory task participants may be required to make study-trial responses that are recorded on separate rows to the test-trial responses, or participants may fail to make a test response on some trials. Such rows can be excluded by specifying the character or symbol that identifies them (by default “NA”).

At the end of the initial pass, **stFirst** will generate a state-trace plot for each individual participant and for the group average. The final argument, *Accuracy based on probabilities?,* specifies whether accuracy measures based on probabilities (“T”) or on the inverse cumulative normal ($z$) transformation of probabilities (“F”) should be plotted. For the former option the state-trace plot will contain an estimate of the proportion of correct responses for B0 designs and the hit rate (HR; the proportion of correct ‘yes’ responses) minus the false alarm rate (FAR; the proportion of incorrect ‘yes’ responses) for B2 designs. Although state-trace analysis largely avoids the scale dependent caveats that can confound bounded response measures, some users may still wish to normalise their data. The latter option, therefore, specifies $z$(proportion correct) for B0 designs and the signal detection theory measure $d' = z$(HR) $- z$(FAR) for B2 designs.

For this example analysis, most of the **stFirst** arguments can remain at their default values. However, we assigned “DFIE.sta” as the *sta* object name, entered the file name “DFIE.txt” for the second argument value (assuming that the text file is located in
the working directory) and specified that a probability measure should be used when
generating the state-trace plot (i.e., the proportion correct as the example data comes
from a B0 design). Clicking the ‘OK’ button will then run \texttt{stFirst}. Note that when
\texttt{stFirst} is executed it may appear that the screen has ‘frozen’ even though the sampling
has begun (the same is also true for \texttt{stSample}). In this situation it is often necessary to
turn off the "buffered output" option in R (select \textit{Misc} > \textit{Buffered output} from the R
console) in order to view the produced output. If this is not done the output will still be
printed; however, the user will have to wait until the \texttt{stFirst} is complete rather than
receiving updates and results as the pass is running.

Before sampling begins, a number of checks are run to ensure the selected data
files are compatible with \texttt{StateTrace} (e.g., that the state and dimension factors each
have only two levels) and that multiple data files are compatible with each other (e.g.,
all have either a B0 or B2 design). If any of the files are incompatible, an error message
will direct the user to the aspect of the data file(s) that produced that problem. For
example, if one participant’s data had three trace levels and another had four trace levels
the message “Participants have different numbers of trace levels” would be printed.

Using the raw data, the initial pass will then create the \textit{sta} object. This object is
essentially a list of lists, which are empty by default but will hold information about the
raw data (e.g., whether the design is a B0 or B2, as well as the number of and expected
order for the trace levels) as well as the sample counts and prior and posterior estimates
(see \texttt{?staMake} for a detailed breakdown of the \textit{sta} object). \texttt{StateTrace} will then
complete two “runs” per participants sampling from the encompassing posterior. This
sampling is done for each independent set of design cells; four sets for B0 designs
corresponding to each combination of the state by dimension factor levels and two sets
for B2 designs as there is a baseline condition effectively tying the dimension levels together for each state level.

For each run, **StateTrace** draws $S = 100,000$ samples from the encompassing model assuming no orders and counts the number of these samples that respect the trace model order $s_T$. It then calculates the posterior proportion for the trace model (T) relative to the unrestricted (U) encompassing model using

$$\hat{\Pi}_{T,U} = \frac{(s_T + 1)}{(S + 2)},$$

estimates the 95% credible interval for the posterior estimates and where the precision of the interval is less than 0.0005 marks sampling for a set as being complete. Otherwise it estimates the time required to get enough samples to narrow the interval sufficiently, based on the time taken for the initial 100,000 samples and using a line search between 100,000 and $10^{14}$ samples. Note the time estimates are only approximate and will vary if sampling is completed on a different computer.

Next an order constrained Gibbs sampler (Gelfland, Smith & Lee, 1992) is used to draw two sequences of 5,000 MCMC samples (“chains”) from the trace model per participant. For each chain $S_T = 5,000$ samples are drawn from the trace posterior and **StateTrace** counts the number of these samples that have a monotonic order, $s_M$ and that are non-overlapping, $s_{NO}$. The proportion of trace samples following the no-overlap, $\hat{\Pi}_{NO,T} = \frac{(s_{NO} + 1)}{(S_T + 2)}$, uni-dimensional, $\hat{\Pi}_{UD,T} = \frac{(s_M - s_{NO} + 1)}{(S_T + 2)}$, and multidimensional, $\hat{\Pi}_{MD,T} = \frac{(S_T - s_M + 1)}{(S_T + 2)}$, model orders are tabulated and the corresponding 95% credible intervals calculated. Next the precision of these intervals is assessed, with sampling for a participant marked as complete if all intervals are less than 0.005 and the additional time required estimated otherwise. Note that a smaller interval criterion is used for the encompassing than trace sampling as the encompassing proportion estimates have greater potential to reduce precision overall because they multiply the trace proportions in the calculation of Bayes factors.
Although all of the above sampling is essentially happening “behind the scenes” **StateTrace** prints a number of details in the R console, which allows the progress of the sampling to be monitored. First the data sources of each data file loaded are reported:

```r
> guistFirst()
[1] "stFirst"
Close 'Generate sta object and complete the first pass ' window to allow entering commands in the R console; that window has gone modal. Note that the ' Generate sta object and complete the first pass ' window may be hidden from view (esp. in windows), and you may have to find it in the taskbar and close it.

Reading in data file DFIE.txt
```

If the data files are compatible and compiled to create the *sta* object, a summary of the design is then reported:

Object DFIE.sta has state levels F,H dimension levels I,U and 3 trace levels, with baseline design B0

As seen above, this summary notes that our DFIE example, has “F” and “H” (i.e., faces and houses) as the state factor levels, “I” and “U” (i.e., inverted and upright) are the dimension factor levels, there are three levels for the trace factor and the data has a B0 design.

Next updates are printed relating to the sampling from the encompassing model including (a) the current precision of the credible interval of the trace posterior proportion for each independent set as well as the (b) estimated time remaining (in minutes) for each participant:

<table>
<thead>
<tr>
<th>UPDATING REMAINING TIME FOR ENCOMPASSING MODEL SAMPLING (minutes)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Participant 1 (Chain CI-Size: 1=0.0046 2=0.006 3=0.0059 4=0.006 ) : 0.18</td>
</tr>
<tr>
<td>Participant 2 (Chain CI-Size: 1=0.0059 2=0.0023 3=0.0062 4=0.0055 ) : 0.11</td>
</tr>
<tr>
<td>Participant 3 (Chain CI-Size: 1=0.0053 2=0.006 3=0.0055 4=0.0054 ) : 0.14</td>
</tr>
<tr>
<td>.</td>
</tr>
<tr>
<td>Participant 17 (Chain CI-Size: 1=0.0059 2=0.0052 3=0.0043 4=0.0059 ) : 0.11</td>
</tr>
<tr>
<td>Participant 18 (Chain CI-Size: 1=0.006 2=0.0058 3=0.0055 4=0.0062 ) : 0.13</td>
</tr>
<tr>
<td>TOTAL TIME REMAINING FOR ENCOMPASSING SAMPLING: 2.31</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>WORKING ON ENCOMPASSING MODEL FOR REMAINING PARTICIPANTS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Participant 1 (Chain CI-Size: 1=0.0032 2=0.0042 3=0.0042 4=0.0042 ) : 0.2</td>
</tr>
<tr>
<td>Participant 2 (Chain CI-Size: 1=0.0042 2=0.0016 3=0.0044 4=0.0039 ) : 0.13</td>
</tr>
<tr>
<td>Participant 3 (Chain CI-Size: 1=0.0038 2=0.0042 3=0.0039 4=0.0038 ) : 0.17</td>
</tr>
<tr>
<td>.</td>
</tr>
<tr>
<td>Participant 17 (Chain CI-Size: 1=0.0042 2=0.0037 3=0.0031 4=0.0042 ) : 0.15</td>
</tr>
<tr>
<td>Participant 18 (Chain CI-Size: 1=0.0042 2=0.0041 3=0.0039 4=0.0044 ) : 0.17</td>
</tr>
</tbody>
</table>
It should be noted that when sampling from the encompassing posterior the time remaining relates to the appropriate computation time required for the set with the “worst” precision to meet the specified criterion. For example, for participant 2 above, the time remaining of 0.11 minutes at the end of the first run relates to the third set which has the widest credible interval. These details are printed for each individual participant, followed by an estimate of the total time remaining at the end of the first run (here 2.31 minutes) and then the individual participant updates are printed for the second run of sampling from the encompassing posterior.

Similar updates are then printed for the MCMC sampling from the trace posterior including (a) the current precision of the credible intervals for the no-overlap, uni-dimensional and multidimensional posterior proportions and (b) the estimated time remaining for all three estimates to satisfy the required precision criterion:

```
UPDATING REMAINING TIME FOR TRACE MODEL SAMPLING (minutes)
Participant 1 (CI-Size: NoOverlap=0.0031 ID=0.0135 MD=0.0138): 0.35
Participant 2 (CI-Size: NoOverlap=0.0043 ID=0.0129 MD=0.0135): 0.35
Participant 3 (CI-Size: NoOverlap=7e-04 ID=0.0068 MD=0.0068): 0.35
...
Participant 17 (CI-Size: NoOverlap=7e-04 ID=0.0137 MD=0.0137): 0.35
Participant 18 (CI-Size: NoOverlap=0.0011 ID=0.0004 MD=0.0004): 0
TOTAL TIME REMAINING FOR TRACE SAMPLING: 5.62

WORKING ON TRACE MODEL FOR REMAINING PARTICIPANTS
Participant 1 (CI-Size: NoOverlap=0.0025 ID=0.0094 MD=0.0097): 0.46
Participant 2 (CI-Size: NoOverlap=0.0031 ID=0.0092 MD=0.0096): 0.46
Participant 3 (CI-Size: NoOverlap=4e-04 ID=0.0048 MD=0.0048): 0
...
Participant 17 (CI-Size: NoOverlap=4e-04 ID=0.0098 MD=0.0098): 0.46
Participant 18 (CI-Size: NoOverlap=5e-04 ID=0.0023 MD=0.0023): 0
```

It can therefore be seen that at the end of the second run participant 2 also requires a further 0.46 minutes of computation time for the three estimates to satisfy the precision criterion. Finally, at the end of the second run of trace sampling, an estimate is reported for the overall time remaining for sampling from the encompassing posterior, the trace posterior as well as all sampling:

```
TOTAL TIME REMAINING FOR ENCOMPASSING SAMPLING (minutes): 2.96
TOTAL TIME REMAINING FOR TRACE SAMPLING (minutes): 5.53
TOTAL TIME REMAINING FOR ALL SAMPLING (minutes): 8.49
```
For the trace model sampling, initial (burn-in) samples are discarded because it can take some time for the MCMC process to converge to the target distribution (see Gilks, Richardson & Spiegelhalter, 1996). Our experience is that for Prince, Brown and Heathcote’s (2012) method, convergence is very fast and that at most 100 initial samples need to be discarded. However, this may not be the case in all applications, so StateTrace provides facilities to check. Once the two runs sampling from the trace model have completed, stFirst automatically calculates one check for whether the MCMC process has worked properly (i.e., has “converged”) using Gelman’s multivariate “R-hat” statistic as provided by Plummer, Best, Cowles and Vine’s (2006) coda package:

The calculation of this statistic requires there to be more than one set, or ‘chain’, of samples. Therefore, to ensure convergence can always be assessed at the end of stFirst for all participants, this first pass will force two runs from the trace posterior for all participants (cf. sampling from the encompassing posterior, where the second run is only completed for those participants where sampling is not complete). As seen above, the first row of the output corresponds to the participant number and the second row is Gelman’s R-hat statistic, where values close to one indicate good convergence.

In the final stage of computation stFirst draws 10,000 bootstrap average samples and uses the two-dimensional density estimator provided by Wand and Ripley’s (2009) KernSmooth package (with its default parameters) to calculate the posterior modes (measures of central tendency) and 68% credible regions around the modes:
These calculations are used to display an average state-trace plot, which provides the user with an immediate view of results averaged over participants, as well as state-trace plots for each individual participant. Additionally the corresponding posterior mode estimates are output in a tabular form to the R console; for example:

Participant 10 posterior mode
p(posterior region)=0.68

, , State = F (HR)

<table>
<thead>
<tr>
<th>Dimension</th>
<th>Trace</th>
<th>I</th>
<th>U</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
<td>0.6407435</td>
<td>0.5386162</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>0.4630981</td>
<td>0.6809448</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>0.6132080</td>
<td>0.6370516</td>
</tr>
</tbody>
</table>

, , State = H (HR)

<table>
<thead>
<tr>
<th>Dimension</th>
<th>Trace</th>
<th>I</th>
<th>U</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
<td>0.5523747</td>
<td>0.4844325</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>0.5285221</td>
<td>0.5976829</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>0.6480587</td>
<td>0.5489388</td>
</tr>
</tbody>
</table>

Average
0.577806

Once the first pass has completed, stFirst suggests directions for the next stage of analysis:

First pass completed!
Run stSample to obtain accurate posterior model probability estimates then examine results with stSummary and stProbplot.
Run stBootstrap to obtain participant results for average trace and monotonic plots with stPlot
Use staManage to join sta objects and manage stored samples

Note also that stFirst can be called repeatedly to add additional participants to the sta object; it should just be ensured that the argument sta object name is provided with the name of the existing object to which the additional data should be appended to. A warning will be issued if duplicate data sources are mistakenly specified however, this data will still be added to the object without replacing the old entries.
Saving and Loading \texttt{sta} Objects

Before running any subsequent passes, the user may wish to save the \texttt{sta} object to the current stage of analysis. This can be done by typing

\begin{verbatim}
> save.image()
\end{verbatim}

which will save the \texttt{sta} object (and everything else in the environment) in the current R workspace. However, there may be times where it is more appropriate to save the \texttt{sta} object outside the R workspace. This can be done using the \texttt{save()} command by providing the name of the object to be saved and assigning a file name within the parentheses. For example, typing

\begin{verbatim}
> save(DFIE.sta, file="DFIE.sta")
\end{verbatim}

will save the example \texttt{sta} object as a compressed file called “DFIE.sta” in the working directory. Note this command behaves in much the same way as ‘zipping’ a file and so can also be used to save multiple \texttt{sta} objects to a single external file (e.g.,

\begin{verbatim}
save(c(DFIE1.sta, DFIE2.sta), file="sta objects").
\end{verbatim}

When saved as an external file, the \texttt{load()} command can be used to load the \texttt{sta} objects into any R workspace. If the file has been copied to the new working directory, then simply including the file name within the parentheses will direct R to the file to load; for example

\begin{verbatim}
> load("DFIE.sta")
\end{verbatim}

Similarly, \texttt{load("sta objects")} will ‘unzip’ and load both of the \texttt{sta} objects.

Alternatively, if the file is not located in the working directory, typing

\begin{verbatim}
> load(choose.files())
\end{verbatim}

will open a file list dialog from which the user can navigate to and select the file to load.
Refining Estimates

Further sampling may be required if the analysis did not reach completion during the initial pass or the user wishes to alter the credible interval and precision criteria from those used by \texttt{stFirst}. This is achieved using \texttt{stSample} (see Figure 4), which allows fine-grained control over the defaults used by \texttt{stFirst}. The only required input (* \textit{sta object name(character string)} *) is the name of an existing \textit{sta} object (e.g., “DFIE.sta”). Because obtaining enough samples to fulfil stricter criteria can be time consuming, \texttt{stSample} has a ‘refresh’ mode (\textit{Refresh sta object calculations} = \textit{T}), which allows the predicted time to completion to be calculated for different criteria; that is, it will re-assess the observed credible interval and corresponding precision against the new criteria and then estimates how much more computation time is required. This refresh mode is fast to run as no actual sampling is done, unless none has been done yet, in which case a single pass is completed to get the necessary timing information. Note that as this timing information will vary depending on the computer used, if the most recent pass (e.g., running \texttt{stFirst}) was run on a different computer it is useful to turn off the refresh mode but leave the maximum run time at zero, which will cause a single pass to be run and update the timing information for the new computer.
Figure 4.4. GUI for the \texttt{stSample} function, which is used to refine the posterior estimates and sampling parameters used.

Once a sampling plan is determined the refresh mode can be turned off, a suitable maximum run time entered and sampling initiated. The \textit{Maximum run time (hours)} can be set to any feasible number of hours by entering an integer value in the text box. As noted above, by leaving this argument at the default ‘0’ value, \texttt{stSample} will complete a single pass of sampling per participant. Alternatively, the timing estimated provided by \texttt{stFirst} can be used to inform this parameter value. For example, the output from the initial pass of the DFIE data estimated that a further 8.49 minutes of computation time was required; hence the maximum run time could be set to 0.14 hours. However, given that this value is only an estimate that may fluctuate as the sampling progresses, it is typically wiser to set a larger value if it is desired for the sampling to run to completion (when there are a large number of participants we have found overnight runs to be a good solution; i.e., maximum run time $= 8$ hours).
**stSample** will divide the maximum time allowed between the types of sampling (from the encompassing and/or trace model) and then further between each participant that has not been completed. Note however, that sampling will terminate when the precision criteria are satisfied, and so **stSample** may complete before the maximum run time has elapsed and one type of sampling may complete before the other. The progress of this sampling can be monitored using the final **stSample** argument (**verbose**), which controls information printed to the R console during sampling. The slider for this argument can either be dragged along the width of the bar or a numeric value (0, 1, 2) entered in the adjacent text box: 0 is silent, 1 prints the estimated total time remaining after each run for all participants and 2 adds estimated timings per participant (the **stFirst** output is provided by a verbose value of 2).

Sampling is completed in a series of runs, and users may choose to sample only from the encompassing model (**Run encompassing model**), the trace model (**Run trace model**) or both. The number of samples for each run of each type of sampling (**Samples per run for encompassing model** and **Samples per run for trace model**) is chosen to satisfy a trade-off between optimising computational speed and obtaining a sufficient number of samples to count those respecting a model’s order. Although fast, small values inherit a small cost in housekeeping between runs and initial **burn-in** samples on each trace model run are lost (**Number of burn-in samples**). In contrast, large values cost more in memory and can result in more samples being taken than is necessary to achieve the required precision. A larger value is advisable for encompassing than trace model sampling, as encompassing sampling is usually an order of magnitude faster. In our applications we have found the defaults work well (100,000 samples per run for the encompassing model and 5,000 per run for the trace model sampling with 100 burn-in
samples) and that little is gained in particular cases by altering them. Nevertheless, these values can be modified by entering a different integer value in the appropriate text box.

Similarly the accuracy criteria (Credible interval (0-100%), as well as Credible interval precision for encompassing samples (0-1) and Credible interval precision for trace samples (0-1)) can be set by altering the integer value in the appropriate text box. Again we have found the default criteria (95% credible interval and precision of 0.0005 for encompassing samples and 0.005 for trace samples) strike an appropriate balance between computation time and the accuracy of Bayes factor and posterior model probability estimates.

A second reason for running `stSample` is to collect samples that can be used in visualising each model; that is, samples that follow the order(s) dictated by a model. We have found the default value of 10,000 encompassing samples (Number of encompassing model samples to keep for plotting), which were collected by `stFirst`, are sufficient for accurate visualisation of central tendencies and credible regions, although large regions (which require estimation of distribution tails) can require more. The 10,000 trace samples (Number of trace model samples to keep for plotting) collected by `stFirst` are also usually more than sufficient given they are only used to estimate central tendency. The `stSample` function also collects a particular type of trace sample, monotonic model samples (Number of monotonic model samples to keep for plotting), which may be relatively rare, especially when the data are far from monotonic. Monotonic samples are used to plot the central tendencies of the uni-dimensional or no-overlap models, with the latter type of sample often being extremely rare unless the data are strongly non-overlapping. Given this, by default `stSample` keeps all (i.e., Inf) monotonic samples.
As for **stFirst** once the desired parameters have been set, clicking ‘OK’ will initiate **stSample**; for the current example all defaults were used except the refresh mode was turned off and the maximum run time set to ‘8’ (although sampling was complete within 20 minutes). This process of running a subsequent pass can be repeated as many times as is necessary. When all sampling is complete, the overall time remaining for both the encompassing and trace models will be recorded as ‘0’.

**Managing the sta Object**

Storing large sets of samples for each participant can greatly increase the size of **sta** objects. The **stSample** defaults (assuming stored monotonic samples are not allowed to grow too large) do not cause problems, but if an object contains data from a large number of participants, issues may arise, such as very slow loading and saving times for **sta** objects. The **staManage** function (see Figure 4.5) can be used to reduce the number of stored samples in such cases (Number of encompassing model samples to keep, Number of trace model samples to keep and Number of monotonic model samples to keep). It also includes the option to keep only samples for the “best” (i.e., most frequently occurring, and hence most probable) monotonic order rather than all samples with monotonic orders (Keep only samples for the best monotonic model?).
As **stSample** can add extra stored samples **staManage** also allows for Gelman’s multivariate convergence analysis to be run on the expanded sets (*Check convergence of trace model MCMC chains*?). However, as **stSample** also provides the option of modifying the number of trace samples drawn per chain, the *Length of each MCMC chain* must be specified; this value must be some multiple of the total number of trace samples and result in there being at least two chains. **staManage** also allows the expanded sample sets to be exported as a list of **mcmc.list** objects

9 by specifying a name for the object (*Name of saved MCMC samples*; e.g., “DFIE.mcmc”). Exporting this object allows the user to run further convergence checks provided by Plummer et al.’s (2006) **coda** package (see the package documentation for details). For example, visual inspection of the chain convergence can be assessed using

```r
> plot(DFIE.mcmc[[1]])
```

which will plot the results for the first participant. If the chains have converged then this plot should look like a “fat hairy caterpillar” and the separate chains should not be

---

9 The **mcmc.list** format requires equal chain lengths for all of the chains it contains. This may not be the case if **stSample** is not run with the default value for *Samples per run of the trace model* used by **stFirst** (i.e., 5,000). In this case it is best to use **staManage** to remove the trace-model samples stored by **stFirst** before running **stSample** with new parameter values.
distinguishable from each other (see also `summary(DFIE.mcmc[[1]])`) to obtain additional summary statistics from `coda`).

Finally, `staManage` can be used to bind multiple `sta` objects. In contrast to the previous functions the required input of an `sta` object name (`* sta object name/s (character string)`) may either specify a single character name, in which case the stored samples are managed, or a vector of character names, in which case the objects are joined and saved to a new object. For example, it can be computationally efficient to divide a very large sample and run the sampling for sub-groups of participants on separate machines. Once completed these `sta` objects can be combined to examine the group aggregate results. By default, the combined object is saved back into the first element of the `* sta object name/s (character string)` value. However, a new object will be created if an alternate character name is entered for the `New name for sta object (optional, character string)` argument.

**Extracting Results**

The `stSummary` and `stProbplot` functions display model selection results, and the `stPlot` function creates state-trace plots. The `stSummary` function also provides information about the status of sampling for an `sta` object (i.e., whether it is complete or if not how much more computation time is required according to the criteria stored in the object). All three functions may be run as soon as `stFirst` is complete, but `stSample` should also be run to completion when final results are required. The `stBootav` function must also be run in order to calculate averages based on samples stored by `stSample` before making participant-average state-trace plots. The `stFirst` function runs `stBootav` on encompassing model samples, but `stBootav` must be run separately to plot participant averages for other models.
Model Selection Results

The stSummary GUI (Figure 4.6) controls output of tabular model-selection results to the R console. The only required argument for this function is the name of an sta object (* sta object name (character string)). Results can be output in terms of Bayes factors or posterior model probabilities using the first true/false argument, Report Bayes Factors (T) or probabilities (F). For the latter option, the next argument, Use Trace-true (T) or Exhaustive (F) strategy for probabilities, can specify if the probabilities are calculated based on all four models (i.e., an exhaustive selection strategy) or by excluding the non-trace model from the set (i.e., the trace-true strategy).

By default, stSummary reports results summarised over participants based on group Bayes factors, which are the product of each participants’ Bayes factors and assume each participant contributes independent evidence (see Prince, Brown & Heathcote, 2012). Note these group results are not obtained by averaging data over participants, as Prince, Brown and Heathcote showed that neither monotonicity nor non-monotonicity are necessarily preserved by averaging.

However, in some cases the group results can be inappropriately influenced by outlying individual participant results (e.g., most participants are uni-dimensional but a few strongly multi-dimensional participants dominate the group results). We therefore, recommend that users always output individual participant results in order to check this possibility using the Display values for individual participants option supplied in the stSummary GUI. When outliers occur the corresponding participant’s integer or character identifier can be entered for the argument Participants to exclude, to exclude them from the calculation of group results; for example, to exclude the third and seventh participant we would type 3 7 in the text box. For readability the number of decimal places printed (Round to how many decimal places?) and different ways of sorting
individual participant results (based on results for a particular model; *Sort values for individual participants by model*) can also be specified.

![GUI for the stSummary function and further customisations](image)

*Figure 4.6. (a) GUI for the stSummary function and further customisations that are available by clicking the (b) “Select additional results to display” button.*

Users may also output a large range of additional results, using a multi-option list GUI (Figure 4.6b) that opens after pressing the *Select additional results to display* button shown in Figure 4.6a. When each participant’s data were read from a separate file the “Data Sources” option outputs the file names, which can be useful in linking this information to the participant identifiers used by *StateTrace*. The remaining options output prior probabilities for each model (calculated analytically; see Prince, Brown & Heathcote, 2012), as well as the total number of encompassing and trace model samples and raw counts of the number of times the orders specified by each model were sampled. The latter results can be used to instantiate different model selection strategies without obtaining new samples (see Prince, Brown & Heathcote for details; a brief example is also provided below).

When the desired parameters have been set, clicking ‘OK’ will execute the stSummary function and the appropriate output will be printed to the R console. For our example analysis, we asked stSummary to output posterior probabilities using the
exhaustive model selection strategy, and to include the individual participant results.

We also asked for \texttt{stSummary} to output the prior probability for the trace model as well as the total number of samples from the encompassing model and the number of these samples that respected the order of the non-trace model.

The \texttt{stSummary} output first prints whether the sampling is complete and the accuracy criteria used, as well as the number of participants included in the group level results. Below we can see that for our DFIE example sampling is complete using a 95\% credible interval and precision of 0.0005 and 0.005 for encompassing and trace model sampling respectively. Moreover, all 18 participants have been included in the group results:

\begin{verbatim}
Sampling complete at 95\% credible interval with precision based on:
  Encompassing samples=3e-04 and Trace samples=0.005
  Number of participants=18
\end{verbatim}

If however, sampling were incomplete, \texttt{stSummary} would print the accuracy criteria specified as well as the estimated computation time remaining.

Next \texttt{stSummary} outputs the group followed by individual participant results and in both cases records that the posterior model probabilities are presented:

\begin{verbatim}
Group level analysis (posterior model probabilities)
Non-Trace No-Overlap Uni-dimensional Multi-dimensional
0 0 0 0

Individual participant analysis (posterior model probabilities)
Non-Trace 0.003 0.129 0.030 0.315 0.029 0.021 0.005 0.0005 0.0017 0.0017
No-Overlap 0.257 0.316 0.014 0.079 0.031 0.035 0.158 0.015 0.013
Uni-dimensional 0.434 0.313 0.236 0.249 0.600 0.218 0.315 0.203 0.526
Multi-dimensional 0.306 0.252 0.720 0.357 0.339 0.707 0.522 0.766 0.443
Non-Trace 0.667 0.001 0.280 0.017 0.136 0.006 0.010 0.041 0.018
No-Overlap 0.016 0.004 0.012 0.477 0.222 0.130 0.017 0.006 0.034
Uni-dimensional 0.133 0.467 0.054 0.175 0.271 0.187 0.383 0.561 0.065
Multi-dimensional 0.163 0.527 0.634 0.332 0.370 0.677 0.589 0.393 0.882
\end{verbatim}

For our DFIE example we see that with the exception of participant “10”, no participants showed positive support favouring the non-trace model, which is consistent with the group result, \(gp_{NT,U} < .001\). Therefore our manipulation of study duration was largely successful in producing a monotonic effect on performance. Similarly, no
support was found for the no-overlap model, \( g_{P_{NO,U}} < .001 \), suggesting that our use of longer study durations for inverted than upright presentations achieved strong overlap of the data traces and hence a diagnostic design. The final two models can inform us of the underlying latent dimensionality. For our DFIE data we see that the multi-dimensional model received very strong support, \( g_{P_{MD,U}} > .999 \), while the uni-dimensional model received little support, \( g_{P_{UD,U}} < .001 \). However, the individual participant results were not always as decisive in their evidence favouring either a multi-dimensional or uni-dimensional account.

Following the model selection results, stSummary outputs any additional results that were specified; here we asked for the trace model prior probability, the number of encompassing samples that respected the non-trace model and the total number of samples drawn from the encompassing model:

<table>
<thead>
<tr>
<th>Trace model prior probability</th>
<th>0.0007716049</th>
</tr>
</thead>
<tbody>
<tr>
<td>Non-Trace model samples from encompassing model</td>
<td></td>
</tr>
<tr>
<td>D1S1 1358738 9120721 8630085 9020214 9109822 6653134 8665316 8114288</td>
<td></td>
</tr>
<tr>
<td>D2S1 9073437 2121706 9090804 775329 7168134 8179696 7505780 7702157</td>
<td></td>
</tr>
<tr>
<td>D2S2 4764984 7873208 8888863 2488003 530354 4576328 7892338 603246</td>
<td></td>
</tr>
<tr>
<td>D2S2 9073527 88187000 3049042 4147921 7763561 6938374 2677432 9107318</td>
<td></td>
</tr>
<tr>
<td>D1S1 7478322 594330 280441 4978041 917771 1840771 7808696 8534738</td>
<td></td>
</tr>
<tr>
<td>D2S1 8412627 8925902 908049 5950154 8527622 6683316 2919946 3309888</td>
<td></td>
</tr>
<tr>
<td>D1S2 73000 9128638 9166930 1840517 6585839 8775368 9158786 2777556</td>
<td></td>
</tr>
<tr>
<td>D2S2 7799845 8122652 20184966 4302973 5221332 9042383 5129042 9905012</td>
<td></td>
</tr>
</tbody>
</table>

| Total samples from encompassing model |          |
| D1S1 8400000 14200000 11200000 12600000 1340000 7800000 15000000 |
| D2S1 14200000 22000000 14300000 6500000 8600000 15300000 15400000 |
| D2S2 13900000 15400000 12200000 2600000 5700000 13700000 15400000 |
| D2S2 14400000 12000000 11800000 4500000 9800000 15300000 11200000 |
| D1S1 10300000 9100000 600000 3900000 14100000 14200000 19000000 |
| D2S1 15400000 10900000 12400000 7000000 8800000 15200000 15200000 |
| D2S2 5800000 21000000 13700000 13900000 19000000 7700000 14500000 |
| D2S2 14000000 15400000 10300000 10300000 13400000 14300000 12700000 |
| D1S1 15400000 11200000 13900000 14500000 |
| D2S1 11600000 14400000 11000000 13600000 |
| D1S2 13800000 11400000 7500000 12000000 |
| D2S2 14200000 14300000 14000000 13400000 |
Although the latter two sample counts are reported for each independent chain, note that *StateTrace* is not interested in the probability that the non-trace model is true for ‘chain one’ but not ‘chain two’ and so on, but rather the probability that it is true for all chains. Therefore, the posterior proportions are first calculated for the trace model at the level of the independent chains and then combined to calculate the posterior proportion for the complementary non-trace model. For example, consider participant “1”: the posterior proportion of the trace model for chain one is 

$$\hat{\Pi}_{T,U} = \frac{(8,400,000 - 1,358,738 +1)}{(8,400,000 + 2)} = 0.8382$$

and the combined trace posterior proportion is 

$$\hat{\Pi}_T = 0.8382 \times 0.3610 \times 0.6572 \times 0.3698 = 0.0735$$

The complementary non-trace model posterior proportion is therefore, 

$$\hat{\Pi}_{NT,U} = 1 - 0.0735 = 0.9265$$

and the non-trace Bayes factor is 

$$BF_{NT,U} = 0.9265 / (1-0.00077) = 0.927.$$ 

As noted above these expensive computations (sampling and counting orders) can be re-used to assess many other model selection strategies. For example, Prince, Brown and Heathcote (2012) also suggested a sequential strategy, which first compares the trace and non-trace models. Here a Bayes factor would be calculated for both the trace model ($BF_{T,U} = 0.0735 / 0.00077 = 95.45$, for participant “1”) and the non-trace model ($BF_{NT,U} = 0.927$), and then the non-trace posterior model probability calculated, 

$$p_{NT,(NT,T)} = BF_{NT,U} / (BF_{NT,U} + BF_{T,U}) = 0.0096.$$ 

Given that the non-trace model is not supported, a similar comparison could then be made between the multi-dimensional and monotonic models and then (if warranted) between the uni-dimensional and no-overlap models (see Prince, Brown & Heathcote, for further details).

In addition to the tabular output of *stSummary*, the *stProbplot* GUI (Figure 4.7) allows the distribution over participants of posterior probabilities for each model to be inspected graphically. Again the only required input for this function is the name of
the *sta object* (*sta object name (character string)) and outlying participants can be excluded using the *Participants to exclude* parameter. *stProbplot* also allows the annotation within the plots to be extensively customised. A title can be added above each panel using the next four arguments (*Non-Trace title, No-Overlap title, Uni-dimensional title, Multi-dimensional title*), while the following five text-box arguments can customise the labels for y-axes (*Non-Trace y axis label, No-Overlap y axis label, Uni-dimensional y axis label, Multi-dimensional y axis label*) and for the x-axis (*x axis label*), which is common to all four panels. Note that if no label is desired, the default text should be replaced with a pair of double quotation marks. The *Select plotting symbols* button will link to a multi-option list GUI, which contains a range of filled and unfilled shapes, participant numbers and letters that can be used to denote each participant’s value.

![GUI for the stProbplot function, which creates a graphical display of posterior model probabilities.](image)
Dotted horizontal lines corresponding to the evidence categories in Table 4.1 can be customised using the next two true/false arguments, with *Insert dashed lines at* 
\( p(0.05, 0.25, 0.5, 0.75, 0.95) \) including the criteria for equivocal, positive and strong evidence and *Insert dashed lines at* \( p(0.01, 0.99) \) including the criterion for very strong evidence. The posterior probability for the group (based on the group Bayes factors) can also be displayed as a heavy dashed line on the plot (*Insert line at group posterior probability*) and as a numeric value in the title (*Insert group posterior probability in title*). Finally the range of values on the y-axis can be modified by dragging the slider bars corresponding to the *Minimum y axis value* and *Maximum y axis value* arguments.

Table 4.1.

*Conventions to aid interpretation of the posterior model probabilities (after Raftery, 1995).*

<table>
<thead>
<tr>
<th>Favouring model</th>
<th>Against model</th>
</tr>
</thead>
<tbody>
<tr>
<td>( p &gt; .99 )</td>
<td>Very strong evidence</td>
</tr>
<tr>
<td>( .95 \leq p \leq .99 )</td>
<td>Strong evidence</td>
</tr>
<tr>
<td>( .75 \leq p \leq .95 )</td>
<td>Positive evidence</td>
</tr>
<tr>
<td>( .25 \leq p \leq .75 )</td>
<td>Equivocal evidence</td>
</tr>
</tbody>
</table>
Unlike the GUIs for non-graphical functions, the `stProbplot` GUI does not close after ‘OK’ is clicked. This allows the user to progressively customise the plots without re-calling the function and re-entering the parameter values each time\(^\text{10}\). However, it is important to note that each time the plot is generated, the new plot will over-write the previous unless it is specified that R should record all of the plots generated in the graphics device, which is done by selecting `History > Recording` from the R console.

Clicking the ‘Cancel’ button will dismiss the `stProbplot` GUI but does not “cancel” the most recent parameter values assigned or output produced.

As shown by the output for the example DFIE data in Figure 4.8, the `stProbplot` plot includes a panel for each model and each number (corresponding to the subject number) represents the posterior model probability for a single participant. Participants in each panel are sorted by their results, allowing those with extreme values to be easily identified. Figure 4.8 was made using the default values, which produces appropriate annotation in most cases, and by choosing the option to use integer plot symbols in order to easily identify each participant’s result.

\(^\text{10}\) Mac users please note that there is a known issue in the GUI version of both plotting functions (`stProbplot` and `stPlot`) that does not allow for this progressive customisation to always work. Three work-around options are described in the help documentation for these functions (e.g., `?stProbplot` or `?stPlot`).
Figure 4.8. Posterior model probabilities for each participant (denoted by their participant number) for the DFIE example data for each of the four diagnostic models (panels). Group posterior model probabilities for each model are indicated in panel titles and plotted as a heavy dashed line. Within each panel, participant results are sorted in ascending order of their probability estimates and faint dashed lines demarcate the categories in Table 4.1.
For example, Figure 4.8 clearly suggests that participant “10” provides an outlying result in favour of the non-trace model, as was also noted from the stSummary output. However, follow-up analysis excluding participant “10” revealed little influence on the group posterior model probability ($gp$ in the panel titles). As previously noted, overall these results show positive or greater evidence for the trace model and for data-trace overlap (i.e., low posterior probabilities for the non-trace and no-overlap models). Evidence is weaker and individual variability greater in relation to the dimensionality results, but the group evidence clearly supports a multi-dimensional outcome. With respect to individual variability in dimensionality, the stProbplot plots can also easily reveal a potential mixture of uni-dimensional and multi-dimensional sub-groups, but this is not indicated in Figure 4.8.

The State-Trace Plot

Figure 4.9 shows examples of state-trace plots produced using the stPlot GUI. These plots show results for our 2 (state: face, house) x 2 (dimension: upright, inverted) x 3 (trace: study duration) design for both individual participants (Figures 4.9a and 4.9b) and the group aggregate for the 2AFC example (Figure 4.9c) and for the yes-no example (Figure 4.9d). Plots such as these can be visually inspected to assess whether all of the points fall on a monotonic function: that is, whether the order of points on the x-axis is the same as the order of points on the y-axis.
Figure 4.9. Accuracy state-trace plots showing modes of the posterior estimates from the encompassing model (large symbols) for the 2AFC DFIE data set for (a) participant 10, (b) participant 18, and (c) on average, as well as (d) on average for the yes-no data set. Accuracy is indicated by plotting the hit rate (HR) for the 2AFC DFIE data and $d'$ (i.e., $z(HR) - z(FAR)$, where FAR is the false alarm rate) for the yes-no example. For (a) and (b) lines are data traces, joining the posterior modes of the encompassing model samples, and ellipses represent 50% credible regions. For (c) lines with small symbols join posterior modes of the trace model and for (d) they join the highest posterior probability (i.e., most frequently sampled) monotonic model. The ellipses in (c) and (d) represent 68% credible regions.
Although we could create state-trace plots using the current example \textit{sta} object, which has now been run to completion using \texttt{stSample}, in order to illustrate the full range of state-trace plot options \texttt{stBootav} should also be run to obtain bootstrap averages. Again the only required input for the \texttt{stBootav} GUI (Figure 4.10) is the name of the \textit{sta} object (\texttt{*sta object name (character string)}) and the option is also provided to exclude participants from the bootstrap averages (\textit{Participants to exclude}). The \texttt{stBootav} GUI allows users to choose to calculate bootstrap averages (based on the stored posterior samples for each participant) for one or more of the encompassing, trace and monotonic models (\textit{Generate bootstrap average for encompassing model}, \textit{Generate bootstrap average for trace model}, and \textit{Generate bootstrap average for monotonic model} respectively). Note the bootstrap average for the encompassing model is calculated at the end of \texttt{stFirst}, which enables preliminary state-trace plots to be generated prior to running \texttt{stBootav}.

![GUI for the \texttt{stBootav} function, which is used to obtain the bootstrap participant averages.](image)

\textit{Figure 4.10.} GUI for the \texttt{stBootav} function, which is used to obtain the bootstrap participant averages.
A set of bootstrap averages is created by repeatedly randomly selecting with replacement one sample from each participant’s set of posterior samples for a given model and taking their mean. By default, these averages are based on 10,000 bootstrap samples (Number of bootstrap samples to draw). Each time stBootav is invoked it can compute averages for only one type of accuracy measure (i.e., based on probabilities (“T”) or z-transformed probabilities (“F”); Accuracy based on probabilities?). Finally, the stBootav function also allows participants to be selected at random with replacement on each bootstrap repetition (Resample participants = “T”); this produces a set of averages with the same central tendency but greater variability that is appropriate when the participants are treated as a sample from a population.

For our DFIE example, stBootav was run using the default values, with the exception of setting the Resample participants argument to “F”. Once executed, stBootav prints a number of updates in the R console, including the participant identifiers for those included in the bootstrap average and a record of the models for which the bootstrap averages were calculated:

```
Calculating bootstrap average over participants:
[1] 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18
Calculating encompassing bootstrap average
Calculating trace bootstrap average
Calculating best monotonic bootstrap average
```

Note that monotonic samples may be relatively rare for some data, and so the averages can be unreliable; to alert users to this possibility, when asked to calculate the bootstrap average for the monotonic model, stBootav also reports the number of participants who have less than 100 samples:

```
Less than 100 monotonic samples available for participants:
   6  8 12 18
  39 55 22  8
```

A participant with no monotonic samples is excluded from the average. The problem of a lack of monotonic samples might be addressed by calling stSample again with a
stricter criterion, but usually a lack of monotonic samples indicates the monotonic model is not appropriate for the data and so there is no point plotting it (e.g., monotonic samples may be rare for our DFIE data as it is non-monotonic).

The **stPlot** GUI (Figure 4.11) operates much like the **stProbplot** GUI. The name of an *sta object (*sta object name (character string)) must be provided and participants may again be excluded from the generated plots by specifying the corresponding participant identifiers (*Participants to exclude*). Note that this latter value must match the participants excluded when **stBootav** was run, as must the accuracy measure to use in the state-trace plots (*Accuracy based on probabilities*?).

**Figure 4.11.** GUI for the **stPlot** function used to generate state-trace plots.

**stPlot** also allows for extensive customisation of the annotation in the state-trace plots. A title can be included above the state-trace plot (**Main title**) and labels can be specified for each level of the state factor (**x axis label** and **y axis label**) as well as for each level of the dimension factor (**Dimension 1 label** and **Dimension 2 label**), which
correspond to the axes and legend entries respectively. By default, `stPlot` will fill these latter four values with the appropriate level obtained from the raw data files (e.g., ‘F’, ‘H’, ‘I’, and ‘U’ respectively for the DFIE data as shown in Figure 4.2). However, alternate labels that are more informative of the experimental design can be specified by modifying the value in the appropriate text box (e.g., for the DFIE plots in Figure 4.9, we entered ‘Faces’, ‘Houses’, ‘Inverted’ and ‘Upright’ in the four text boxes respectively). Note also that the accuracy measure specified will automatically be included in parentheses after the axis label (e.g., HR or $z_{HR}$ for 2AFC data and HR – FAR or $z_{HR} - z_{FAR}$ for yes-no data).

Additionally, the symbols used to represent each level of the dimension factor can be specified using the buttons `Plot symbols for dimension 1` and `Plot symbols for dimension 2`. Note there is no option to use a filled symbol as `stPlot` will include a numeric value within each shape to identify the corresponding trace levels. The range of values for the x and y axes can also be modified by entering a value in the text boxes corresponding to the arguments `Minimum x axis value`, `Maximum x axis value`, `Minimum y axis value` and `Maximum y axis value`. However, leaving these values as the default “NA”, `stPlot` will automatically scale the axes to fit the data and credible regions.

The `stPlot` function represents accuracy data using any one of three measures of central tendency (the mode, mean or median) applied to posterior samples from the encompassing model (`Statistic to use for model plotting`). As the encompassing model makes no order assumptions these central tendency measures (the large symbols in Figure 4.9) provide a model-free estimate of the observed data. The default choice used to create Figure 4.9 (the mode) produces estimates that are usually equivalent to the
familiar maximum-likelihood formula (e.g., $n / N$ for the 2AFC hit rate, where $n$ is the number of correct in $N$ trials).

The other central tendency measures usually produce similar results, at least for reasonable sample sizes not subject to floor or ceiling effects. For example, for the hit rate and uniform prior used by StateTrace, the mean of a large sample from the encompassing posterior is equivalent to $(n + 1) / (N + 1)$. For other accuracy measures such simple formulae are not available. This is also the case for any accuracy measures for any of the order restricted models. Hence, estimates based on posterior samples have the advantage of providing an easily applied and general approach.

The stPlot function uses the same approach to display the degree of uncertainty in central tendency estimates, by drawing contours around regions containing a specified percentage of the posterior encompassing-model samples (Plot credible $p$ regions). Estimating Bayesian credible regions in this way works with all accuracy measures in a way that takes account of any floor and ceiling effects, which can be very influential when contours are near bounds in an accuracy measure. The regions, and modes, are estimated using the bkde2D function in Wand and Ripley’s (2009) KernSmooth package, which is included by default in R.

The stPlot GUI allows users to choose the percentage contained by the regions (Width of credible $p$ regions) and the degree of smoothing (Smoothing factor for $p$ regions), as a multiple of the maximum over data points of the values provided by the dpik function: this KernSmooth function, and bkde2D, are called with default values. The default multiplier of five used by stPlot was chosen to produce very smooth contours even for large regions, which are otherwise often irregular because they require estimation of the tails of the posterior distributions; users are encouraged to experiment with the multiplier in their application.
Users are also able to create either an average state-trace plot or a plot for each individual participant (Plot average (T) or each participant (F)). Figure 4.9a and 4.9b were generated by setting this argument to “F” and plot the results for participant “10”, who had the strongest evidence for a violation of the trace model in Figure 4.8, as well as participant “18”, who had the strongest evidence for the multi-dimensional model; both state-trace plots are clearly consistent with the model-selection analyses. Individual participant data are typically quite noisy, so for clarity the credible regions in these plots contain only 50% of the posterior samples. The lines in these plots are data traces (the default Line type value), which join points with the same dimension factor level (upright or inverted; NB., the weight of the plot lines can be specified using Width of plot line/s).

Both data sets display strong data-trace overlap, consistent with the results for the no-overlap model in Figure 4.8. Note that plots such as 4.9a and 4.9b can be made as soon as stFirst is run, as stFirst stores sufficient samples from the encompassing model for each participant (10,000 for each) and runs the stBootav function necessary to produce averages.

Figure 4.9c was generated by setting the Plot average (T) or each participant (F) argument to “T” and is a state-trace plot of the average over all participants in the example DFIE data. Data points in average plots represent the central tendency of the set of bootstrap averages. Variability among the averages is used to construct credible regions in the same way as for individual participants. These credible regions reflect uncertainty in the estimated average over the particular set of participants in an experiment. Reflecting the reduction in uncertainty associated with an average, the credible regions are much smaller in Figure 4.9c, even though they contain the default value of 68% of the posterior samples (corresponding to the proportion of a normal distribution contained by conventional standard error bars).
The lines in Figure 4.9c join the modes of the average of samples from the trace model. The points joined by the lines (Place points on plot line/s = “T”) are different from the large symbols, which are estimated based on encompassing model samples, as the encompassing model admits samples that violate the trace model. However, in this case the difference is not large, reflecting the fact that for most participants the trace model provides an excellent description of this data.

Figure 4.9d plots average results for the yes-no experiment using the signal-detection theory $d'$ measure of accuracy (i.e., $z(\text{HR}) - z(\text{FAR})$). The lines in Figure 4.9d join the modes of the most commonly occurring order$^{11}$ for monotonic MCMC samples (the best monotonic model); that is, MCMC samples from either the uni-dimensional or no-overlap models. Because this data is well described by a uni-dimensional model the difference between the best monotonic model and encompassing model modes is relatively small. The order in which the points are joined, according with increasing trace factor levels, reflects almost perfect overlap between data traces in the average data. Note that Figures 4.9c and 4.9d were created after first running stSample to the default criterion then stBootav to average the trace and monotonic samples stored by stSample.

**Summary and Conclusions for StateTrace Package**

The question of latent dimensionality (i.e., whether a single latent variable mediates the relationship between the effect of two experimental factors) has pervaded not only basic research in areas of memory, perception and categorisation but it is also an important consideration in applied settings such as clinical psychology, human

$^{11}$ It is important to note that the best order may differ between participants. Before interpreting the best (most frequently occurring) monotonic model in the average, such as is plotted in Figure 4.9d, it is advisable to use stSummary to examine the degree of variability in the best orders over participants, as strong individual difference may mean that taking an average is not sensible.
factors, aging and development (see Prince, Brown & Heathcote, 2012). Traditionally, these questions are addressed using an ANOVA interaction test, with any one-dimensional account rejected when a significant interaction is observed. However, it is widely known that this approach requires strong assumptions to be made, which cannot always be directly tested.

In contrast, state-trace analysis provides a graphical method for assessing latent dimensionality, which makes only ordinal assumptions and so avoids confounds from range effects that can distort other methods, particularly when performance is measured on a bounded response scale. Although state-trace analysis requires researchers to consider different methodological issues Prince, Brown and Heathcote (2012) have provided detailed guidance to help develop and refine a state-trace experiment. Furthermore, they proposed an inferential method for state-trace data, based on Klugkist, Kato and Hoijtink’s (2005) and Klugkist, Laudy and Hoijtink’s (2005) encompassing prior method for estimating Bayes factors. These inferential tests can be used to not only assess dimensionality (i.e., to estimate the probability that a one-dimensional or multi-dimensional model is best able to account for the data) but also to help refine experimental methodology and to check the validity of the dimensionality assessment. Nevertheless, these Bayesian procedures do have the potential to narrow the focus of state-trace applications to only researchers that are familiar with the sampling and estimation techniques required for calculating Bayes factors and corresponding posterior model probabilities. In Chapters Three and Four, we therefore provided a software package, StateTrace, to aid the broader adoption of these methods.

The StateTrace package offers very general purpose data input capabilities, such that very little parsing should be required of a raw data file before it is ready for analysis. The stFirst function allows users to specify the relevant columns from a data
file, the order in which columns should be read-in, and any non-relevant rows that should be excluded. Moreover, **stFirst** is customised for initiating an analysis and so will also perform the necessary procedures to allow for a preliminary examination of results: calculating posterior proportions for the four models and estimating how much additional computation time may be required, as well as checking the convergence of the MCMC trace chains and generating preliminary state-trace plots for the individual participants and group average.

All of the relevant information (sample counts and posterior estimates) required for a range of model selection strategies is stored in the *sta* object. However, **StateTrace** also provides users with a number of summary functions that will not only extract this information but also allow for a customised presentation of results. The **stSummary** function, for example, can output tabular model selection results as Bayes factors or posterior model probabilities, and if probabilities are selected, can use an exhaustive or trace-true model selection strategy. **stSummary** can also simply print the raw sample counts and prior probabilities if the user desires to explore other model selection strategies (e.g., a sequential model selection strategy). Alternatively, **stProbplot** provides a graphical display of the posterior model probabilities and can also plot lines at the critical $p$ values to aid visual assessment of these results. Similarly, for researchers who prefer a graphical approach over inference, the **stPlot** function can be used to create a customised state-trace plot; including drawing a line to represent the best trace or monotonic model to visually examine a model’s fit to the data. The GUIs for these two plotting functions are particularly user-friendly as they do not close on execution. This allows the user to go back and forth between modifying parameter values and generating a plot as many times as is necessary.
Although the range of applications of the **StateTrace** package is restricted (described in Chapter Three; e.g., it is limited to completely repeated measures designs as well as state and dimension factors that have only two levels), the advance provided by Prince, Brown and Heathcote’s (2012) Bayesian analysis and the current package within this domain is substantial. Rather than judging state-trace plots “by eye” these Bayesian procedures not only quantify evidence about dimensionality but also aid design refinement. Moreover, the **StateTrace** package enables the broader adoption of these methods by automating many aspects of a state-trace analysis of binary response data, including the output of both tabular and graphical summary results and customisable state-trace plots. Additionally, **StateTrace** is a GUI-driven package and so is accessible to users who are also not familiar with the R language.
Section Two: The Differential Face Inversion Effect

Included Papers:


In this section state-trace analysis is used to examine the question of whether human face recognition is mediated by a dimension or dimensions additional to those available to most other objects. The common experience of effortlessly recognising a familiar face in a crowd or of involuntarily imagining faces in scenic features, such as clouds, seems to indicate that humans possess an innate aptitude for face processing. It is, therefore, not surprising that there has been a longstanding interest in determining whether faces are “special”. Although the answer to this question may intuitively seem obvious, there is a long history of debate surrounding exactly how and why the perception of, and memory for, faces is special. For example, how do humans manage to discriminate between faces when their homogenous structure makes them so easily confusable (Maurer, Le Grand & Mondloch, 2002)? Is face recognition qualitatively different from the recognition of other visual stimuli because it recruits unique brain regions and/or cognitive processes (Kanwisher, 2000; Kanwisher, McDermott & Chun, 1997)? Is our aptitude for face processing an inherited genetic ability (Phelps & Roberts, 1994), or controversially (Robbins & McKone, 2007), is expertise in face processing learnt from the vast experience with faces accrued from an early age; that is, is face processing just a manifestation of general purpose category learning (Schyns & Rodet, 1997) and so not really ‘special’ (Diamond & Carey, 1986)?

This question of whether faces are special is of broad interest due to the potential importance of this skill (Gould, 1917). Faces provide vital information about gaze direction and emotional expressions, and distinguishing between facial identities is critical for identifying a known ally from an enemy, and hence for our survival (McKone & Robbins, 2011; Richler, Mack, Palmeri & Gauthier, 2011). Indeed over the course of a human lifetime, thousands of faces can become so familiar that they can be recognised after only a glance, when seen in an unfamiliar context, or after undergoing
significant physical changes due to natural aging or cosmetic procedures (Maurer et al., 2002). Yet obtaining identity information is hard because faces have a homogenous structure that makes them so easily confusable (Maurer et al.; McKone & Yovel, 2009).

In particular, our expertise for recognising faces is not as strong when the faces are unfamiliar (Hancock, Bruce & Burton, 2006; Lui & Chaudhuri, 2000; Megreya & Burton, 2006). This difference in performance can have important consequences in applied settings, such as eyewitness identifications. For example, recognition of familiar people can be remarkably good even in low quality video; however, recognition performance becomes quite low when the face is obscured or when the person is unfamiliar (Burton, Wilson, Cowan & Bruce, 1999). Moreover, this poor performance for unfamiliar faces can persist even after extended study of high quality video (Bruce et al., 1999). Both in video and in still images (Bruce, 1982) performance for unfamiliar faces is particularly sensitive to a mismatch between encoding and retrieval images due to a transformation of expression or viewpoint (see Shapiro & Penrod, 1986, for a meta-analysis). Furthermore, although performance is generally low for unfamiliar faces Megreya and Burton (2006) also note that there is a considerable range of individual differences. Hence, because of this difficulty, and the importance of good performance for survival, there does seem to be strong face validity for the ‘special’ status of faces.

In this section we report a series of experiments that examine recognition memory for unfamiliar faces and manipulate the match between study and test presentations using a transformation that has been central to understanding and testing the special status of faces: inversion (i.e., rotation through 180°). Inversion has a detrimental effect on perceptual and memory performance for a wide variety of monoriented objects (i.e., objects with a canonical “upright” orientation; Rock, 1974). However, this effect is often larger for faces than other objects, even when these objects
come from a category with a preferred orientation and fairly equivalent familiarity and complexity to faces (e.g., houses). We call this dissociation the *Differential Face Inversion Effect* (DFIE) and when confirmed by a significant interaction test, this effect is one of the primary pieces of evidence to suggest that faces are processed along an extra dimension not available to non-face stimuli.

Nevertheless, relying on dissociation evidence to infer the existence of a face specific dimension is unfortunate because, as discussed in Section One, it is clearly the case that dissociations do not provide strong evidence for dimensionality (e.g., Bogartz, 1976; Busemeyer & Jones, 1983; Dunn & Kirsner, 1988; Henson, 2006; Poldrack, 2006; Wixted, 1990). It is also unnecessary, because Bamber’s (1979) state-trace analysis does provide a rigorous basis for this type of inference. Loftus, Oberg and Dillon (2004) used state-trace analysis to examine the dimensionality of face recognition and arrived at a conclusion at odds with the vast majority of the research literature: that familiar (famous) faces but not unfamiliar faces have access to qualitatively different encoding from non-face control stimuli.

In this section, we revisit Loftus et al.’s (2004) study based on advances in state-trace methodology developed by Prince, Brown and Heathcote (2012 – Chapter One) in an effort to address concerns about both the generality and validity of Loftus et al.’s study. This section begins by briefly reviewing the brain and behavioural evidence for the DFIE and multi-dimensional face processing. We then describe state-trace analysis and evidence that perhaps only familiar faces can be encoded in a multi-dimensional manner. In Chapters Five through Seven (Prince & Heathcote, 2009, 2010, 2013 respectively) we report a series of experiments over which we were able to refine our experimental design and address various caveats to Loftus et al.’s (2004) state-trace
analysis and subsequent conclusion that the DFIE may be restricted to memory retrieval.

**The Differential Face Inversion Effect**

Although the upright and inverted images used in experimental tasks are strictly identical with the exception of their orientation, it is widely found that an inverted presentation has a strong detrimental effect on the perception of, and memory for, all mono-oriented stimuli; called the inversion effect (Rock, 1974). The inversion effect for faces, for example, has been found to begin in early development (5-7 months; Rose, Jankowski & Feldman, 2008; 3-5 years; Sangrigoli & de Schonen, 2004) and to increase with age (Brace et al., 2001; Carey & Diamond, 1997; Schwartzer, 2000). Inversion has also been shown to affect performance in a number of tasks, ranging from the perceptual matching of body postures (e.g., Reed, Stone, Grubb & McGoldrick, 2006) to recognition memory for races (e.g., Valentine, 1988; but see Farah, Tanaka & Drain, 1995, for an exception).

The detrimental effect of inversion, while evident for all mono-oriented stimuli, is disproportionately stronger for faces; called the Differential Face Inversion Effect (DFIE). This result was first reported by Yin (1969) who observed a DFIE in recognition memory accuracy even when control stimuli (houses, airplanes and men in motion) were matched as closely as possible to faces in terms of their complexity, familiarity and difficulty in applying a verbal label. Since Yin’s seminal paper, the DFIE has been replicated in both perceptual and mnemonic paradigms with various procedural variations and with a wide range of dependent variables from bizarreness ratings (Murray, Yong & Rhodes, 2000), multi-dimensional scaling of response time (Sergent, 1984) and sequential matching (Leder, Candrian, Huber & Bruce, 2001).
The DFIE has been attributed to inversion either greatly weakening, or completely denying, access by faces to the qualitatively different encoding dimension or dimensions that normally enable good recognition performance despite the confusability of faces. That is, inverted faces suffer not only from the inversion decrement common to all mono-oriented objects but also from a decrement due to the loss of the extra-dimensional advantage they enjoy when upright. As a result, faces exhibit a much larger decrement in performance due to inversion than control objects.

**Dimensions of Face Processing**

Yin’s (1969) participants reported using two alternate strategies to aid their memory, namely “searching for some distinguishing feature” or “attempting to get a general impression of the whole picture” (p. 145), with the latter strategy used mostly with faces. The first of these strategies has since been referred to as *featural, piecemeal* or *part-based processing*, and is common to all mono-oriented stimuli. It is based on the isolated features of an object that can be specified without reference to its other parts (Rakover, 2002); for example, the shape of a nose on a face, or the window in a house. However, debate surrounds the details and nomenclature of the second strategy, which is mostly or only available to faces (McKone & Yovel, 2009; Singer & Sheinberg, 2006).

This face-specific dimension has been referred to by some as the *holistic processing* of a face, which has been defined as capturing the overall look of a face (Leder & Bruce, 2000) and as representing the face as a whole without an internal part structure (Tanaka & Farah, 1993). It has also been described as *configural processing* of two types of spatial relations among features. *First-order relations* refer to the arrangements of features that define a face; for example, two eyes positioned above the
nose, which is above the mouth (Rhodes, Brake & Atkinson, 1993). This type of information does not vary greatly between faces and hence first-order relations are useful in determining class membership (i.e., face vs. non-face) but not in discriminating one member from another. When a task involves identifying particular class members, we instead rely on second-order relations. Second-order relational information refers to the distances between the internal features of a face (Diamond & Carey, 1986) and may be defined relative to norms contained in facial prototypes (Goldstein & Chance, 1980).

It is also important to note that in addition to these differential definitions, the terms ‘holistic’ and ‘configural’ (among others) are often used interchangeably to refer to the perceptual process available to faces. Furthermore, the term ‘configural’ is used to describe the metric distances between features that can be measured and manipulated on an individual face. However, our focus is not on differentiation within the second dimension, or on the correspondence of the dimension to measureable stimulus information or to processing strategies. Rather we seek to address a more fundamental question, whether inversion differentially affects two broad, largely independent12 dimensions, which we hereafter refer to as featural and configural. Although either dimension is sufficient to support face recognition, performance in generally better when study conditions emphasize configural rather featural processing; for example, asking participants to make a trait judgement (e.g., classing a face as being honest or dishonest, which encourages processing the face as a whole) versus a featural judgement (e.g., classing a face as having a wide or narrow nose; Winograd, 1981). Of

---

12 Complete independence of featural and configural (i.e., spatial relational) information has been questioned by results such as those of Tanaka and Sengco (1997) who found that the ability to recognize a feature depended on the facial configuration in which it occurred. Tanaka and Sengco thus argued that such interdependency between featural and configural information is a result of holistic encoding.
particular importance is the finding that the benefits offered by configural processing are not observed when the face is presented upside-down.

**Dimensions and Inversion**

Both featural and configural dimensions are affected by inversion, however, typically the extraction of configural information is particularly disrupted. For example, no participants in Yin’s (1969) experiment reported being able to gain a general impression of the entire image when presented with an inverted face. Lewis and Glenister (2003) similarly found that inversion had a stronger detrimental effect on memory for whole faces (which have configural information available) than for face parts (which do not). An enhanced ability to selectively attend to the parts of inverted faces has also been attributed to reduced interference from configural processing that occurs with upright faces (Michel, Rossion, Han, Chung & Caladara, 2006; Young, Hellawell & Hay, 1987). Additionally, inversion is more detrimental to the discrimination of face pairs differing in configural properties (e.g., distance between eyes) than featural properties (e.g., replacing the eyes on one face; Freire et al., 2000; Leder & Bruce, 2000; Leder & Carbon, 2006; Rhodes et al., 1993) and the detrimental effect of inversion on recognition memory accuracy is stronger when study emphasises the configural properties of faces (McKelvie, 1996). It has, therefore, been suggested that different strategies are employed for encoding upright and inverted faces that extract both featural and configural information from upright faces but only featural information from inverted faces and from upright and inverted non-face stimuli (although see Yovel & Kanwisher, 2004, for an exception).

Consistent with this behavioural evidence implicating face-specific processing, both Positron Emission Tomography (PET) and Magnetic Resonance Imaging (MRI)
studies have identified a region of the fusiform gyrus – the face fusiform area (FFA) – that is selectively activated by tasks requiring processing of the configural properties of a face (e.g., see Haxby, Hoffman & Gobbini, 2000, for a review). Although functional MRI (fMRI) studies have found that inversion does not reduce activation in the FFA (e.g., Kanwisher, Tong & Nakayama, 1998), it does increase activation in areas typically activated by other mono-oriented stimuli (Haxby et al., 1999). Similarly, Evoked Response Potential (ERP) studies have identified N170 and N200 components associated with face processing, that are delayed and/or increased by inversion (e.g., Betin, Allison, Puce, Perez & McCarthy, 1996; Jacques, d’Arripe & Rossion, 2007; Jacques & Rossion, 2010; James, Johnstone & Hayward, 2001; Rebai, Poiroux, Bernard & Lalonde, 2001; Rossion et al., 1999). However, the specificity of these neuroimaging results, and configural processing in general, to faces has been questioned. McKone and Robbin’s (2011) noted that two theoretical perspectives have challenged the special status of faces. First, the within-class discrimination hypothesis proposed that ‘face-specific’ processes may actually be engaged in any task that involves identifying an object at the individual level. That is, in the studies reviewed thus far, the task is almost always to identify individual faces rather than to make a face versus non-face judgement. Similarly, outside experimental settings we naturally tend toward individual identifications for faces (e.g., “that’s John”). In contrast, non-face objects are typically identified at a more general level (e.g., “that’s a dog”). Although this hypothesis is generally no longer accepted (McKone & Robbins), it is from this proposal that the second challenging perspective is derived: rather than all within-class discrimination tasks recruiting the apparent face-specific dimensions, these processes may only be engaged when a person holds expertise in identifying members within a class of objects. In other words, our
expertise for face processing may be learnt from our vast experience with this stimulus class.

Gauthier and Tarr (1997), for example, examined the effects of expertise using “greebles” – a set of artificial stimuli that like faces are homogenous and differentiated by the relations amongst a small set of features. They found that after extensive practice (which focused on identifying particular members of the set) greeble experts displayed the behavioural hallmarks of configural processing for greebles. Later studies also found that greeble expertise led to similar effects of inversion for greebles as has been found for faces both in fMRI (e.g., Gauthier, Tarr, Anderson, Skularski & Gore, 1999) and ERP results (e.g., Rossion, Gauthier, Goffaux, Tarr & Crommelinck, 2002). Additionally, Rossion et al. (2000) found an N170 to greebles even in unpracticed participants. Their results thus suggested that the N170 may be associated with a general perceptual processing strategy that is engaged when identifying members of a homogenous stimulus class, which are defined by configural differences.

Consistent with the N170 being specific to a particular type of processing rather than to faces per se, Gauthier, Curran, Curby and Collins (2003) observed for car experts that car and face perception tasks interfered with each other behaviourally by reducing the N170 and that these effects increase with level of car expertise. Similarly, Boggan, Bartlett and Krawczyk (2011) reported that chess experts showed evidence of selective attention failures (due to interference of configural processing), that was of the same magnitude for chessboard displays as for faces. They further argued that as chessboards hold no physical resemblance to faces, this evidence of face-like processing could not be attributed to domain-specific (i.e., face-specific) factors. Nevertheless, the conclusion that configural processing and the inversion effect are not face specific but rather are the result of developing expertise remains controversial (see Gauthier &
Bukach, 2007; McKone & Robbins, 2007; Robbins & McKone, 2007) and should not be taken as an indication that all hallmarks of configural processing are due to expertise (see Boggan et al., 2011, for further discussion).

It has additionally been questioned whether inversion has a quantitative effect on face processing rather than a qualitative (i.e., separate strategies for processing upright vs. inverted faces) effect. For example, Barton, Keenan and Bass (2001) proposed a graded difference in processing strategies, whereby inversion decreases the rate at which both featural and configural information can be extracted but that this rate of decline is particularly marked for configural information (see also Sekular, Gaspar, Gold & Bennett, 2004; Valentine, 1988). Similarly, the differences in degree rather than kind for inverted versus upright face fMRI activation and ERP components suggests that inversion has a quantitative rather than a qualitative effect on face processing (e.g., Bentin et al., 1996; Kanwisher et al., 1998).

The quantitative perspective thus suggests that configural information may be extracted from an inverted face if given sufficient time. Yang and Schwaninger (2010), for example, concluded that inversion disrupts the efficiency of configural processing but does not make it completely inaccessible (see also Richler et al., 2011). They found that a subtle configural difference could not be detected between either upright or inverted face pairs, whereas more extreme configural changes could be detected in both. Furthermore, in contrast to previous evidence showing inversion eliminates the attention advantage for faces over non-faces, Bindemann and Burton (2008) found that both upright and inverted faces were equally efficient in affecting the allocation of visual attention.

Inversion may also not have the same effect on different types of configural information. For example, Goffaux and Rossion (2007) found that inversion disrupts the
processing of vertical relations (i.e., the vertical position of the eyes above the nose), but
the effect of inversion on horizontal relations (i.e., inter-ocular distance) was equal to
that observed for a featural change (i.e., replacement of the eyes). Goffaux and Rossion
therefore argued that perhaps the vertical relations are more important for upright face
discrimination, while horizontal relations largely define the bilateral symmetry of a
face.

A Robust Phenomenon?

Since Yin’s (1969) demonstration, the DFIE in recognition memory accuracy for
unfamiliar faces has been replicated many times and with various procedural variations
and dependent measures. A DFIE has been found with line drawings as well as realistic
pictures of faces, with control stimuli ranging from houses to stick figures, when the
face stimuli are harder and easier to recognise than the control stimuli overall, and when
the face and control stimuli and/or upright and inverted images are presented
randomised together or in blocks. Many of these studies followed Yin in testing images
in the same orientation as they were studied. However, a DFIE has also been found
when test items were presented from a different viewpoint than during study (e.g.,
Valentine & Bruce, 1986) and when all images are studied upright and tested either
upright or inverted (e.g., Scapinello & Yarmey, 1970; Yarmey, 1971). Additionally, the
effect has been found when memory is tested by two-alternative forced choice
procedures – asking participants to choose between a studied and unstudied image
presented simultaneously – and when memory is tested by yes-no decisions about
whether a single test item has been studied (e.g., Carey & Diamond, 1977; Diamond &
Caret, 1986; Valentine & Bruce; Valentine, 1988).
A further commonality in these studies is the method of analysis. Traditionally, the DFIE is quantified by a dissociation that is tested by an interaction comparing a face inversion effect (FIE) to the inversion effect for a mono-oriented control stimulus (e.g., a house inversion effect; HIE). Loftus et al. (2004) noted that numerous reports of this interaction “appear to have formed a robust and reasonably harmonious pattern that has assisted us in understanding face processing” (p. 960). However, it was not long after Yin’s (1969) seminal work that it was shown – particularly for accuracy based measures – that dissociation logic cannot compel the rejection of a one-dimensional explanation without making strong assumptions that are difficult, if not impossible, to test (Bogartz, 1976).

Loftus (1978) provides a very clear statement on this point with several examples. He reasoned that theoretical interest focuses on latent dimensions, such as configural processing, that cannot be directly observed. Hence, inferences relevant to theory rest on assumptions about the mapping between these latent variables and directly observable (manifest) responses. If this mapping is linear an interaction in the manifest variable is a veridical reflection of an interaction in the latent variables (apart from an unknown scale factor; i.e., a common value with the same multiplicative effects on all conditions). However, if the mapping is non-linear, as seems particularly likely for bounded response variables such as accuracy, a manifest interaction may occur when there is no latent interaction or no manifest interaction may occur even when the latent variables interact strongly. Experimenters are familiar with these possibilities in the form of floor and ceiling effects and do their best to calibrate their experiments to avoid them. Even so, away from the bounds of the measurement scale the possibility of non-linearity, and hence the confounding of interaction tests, remains (see Prince, Brown & Heathcote, 2012 – Chapter One – for an explicit example).
In contrast, state-trace analysis (Bamber, 1979), can determine whether one or more than one latent dimension mediates the joint effects of two experimental factors no matter what form the mapping takes as long as it is monotonic (i.e., the manifest and latent variables consistently change in the same or in opposite directions). As described in Section One, the number of dimensions required to explain performance is assessed by plotting one response measure against another (a state-trace plot): if both response measures map onto (i.e., are mediated by) a single latent dimension all points on the plot will fall on a single monotonic function (i.e., the two response variables consistently change in the same or in opposite directions).

Loftus et al. (2004) performed a state-trace analysis using houses and faces (the state factor defining the axes of the plot) presented either upright or inverted during study (the dimension factor). All items were tested upright in a yes-no recognition memory task. This design was utilised to examine Valentine’s (1988) assertion that to produce a DFIE “the orientation of the inspection series does not appear to be critical… [and that]… the disproportionate effect of face inversion only emerges when the task involves recognising a face as one stored in memory” (p. 474). Consistent with this assertion, Loftus et al. found a monotonic state-trace plot for unfamiliar faces versus unfamiliar houses (Experiment 1), but a non-monotonic state-trace plot for familiar (famous) faces versus unfamiliar houses (Experiment 2). Loftus et al. thus concluded that a DFIE only emerges at the time of memory retrieval; that is, familiar faces cause memory retrieval at study and so produce a DFIE when orientation is manipulated at study, whereas unfamiliar faces do not, and so manipulating orientation at study cannot produce a DFIE.
Are Unfamiliar and Familiar Faces Qualitatively Different?

Loftus et al.’s (2004) memory retrieval hypothesis might appear surprising given it is widely believed that inversion affects the early perceptual stages of face processing (Freire, Lee & Symons, 2000). However, a difference in results for unfamiliar versus familiar faces is less surprising. Most obviously, famous faces can be named and so the extra dimension that Loftus et al. detected in their second experiment could be a verbal one, in which case their overall results would indicate that only a single (featural) visual dimension was at play in both experiments. A second possibility is that the visual encoding of familiar faces is different from that of unfamiliar faces.

Indeed, there is mounting behavioural evidence that the expertise humans display in recognising familiar faces does not generalise to unfamiliar faces (Hancock, Bruce & Burton, 2000; Kemp, Towell, Pick, 1997; Lui & Chaudhuri, 2000). Recognition memory for unfamiliar faces is significantly affected by transformations of viewpoint, expression and lighting, yet familiar face recognition is relatively invariant across such manipulations. Burton, White and McNeill (2010), for example, found that a change in the camera angle for photographs taken under otherwise identical conditions was sufficient to produce a strong detrimental effect on accurately matching pairs of unfamiliar faces. Moreover, no improvement in performance has been found under more realistic conditions, including matching a live person to their photograph (Megreya & Burton, 2008) or to CCTV footage even when it included high quality close-up views (Davies & Valentine, 2009). There is a long history in the area of eyewitness identification of poor identification accuracy for unfamiliar faces – results that are often attributed to the fallibility of memory (Burton, Wilson, Cowan & Bruce, 1999; Wells & Hasel, 2007). However, even when memory load was eliminated from the task (e.g., asking participants to determine whether or not two unfamiliar faces are
the same person), Burton and Jenkins (2011) found that performance was still remarkably poor, implicating different perceptual processing for familiar and unfamiliar faces.

A number of studies indicate that unfamiliar face recognition relies on image specific processing strategies rather than the face-specific (configural) processing available to familiar faces. Young, Hay, McWeeny, Flude and Ellis (1985) tested matching of whole faces to faces with only internal or external features. Participants were faster and more accurate at matching internal features for familiar faces even when the images differed in their viewpoint and expression. For unfamiliar faces, in contrast, there was no difference between internal and external matches, with both at the same low level of performance found for external matches with familiar faces. Osborne and Stevenage (2008) found that the magnitude of this internal feature advantage increased with greater familiarity, but was eliminated when the images were presented upside-down. Megreya and Burton (2006) found that performance in matching the identities of unfamiliar faces was strongly predicted by performance in the equivalent inverted task, whereas for familiar faces there was no correlation. Moreover, performance for both upright and inverted unfamiliar faces was positively associated with performance for inverted familiar faces. Megreya and Burton proposed that unfamiliar faces are processed differently to familiar faces in much the same way that inverted faces are processed differently to upright faces; that is, unfamiliar faces may not be supported by configural processing.

Neuroimaging evidence is less supportive of a distinction between unfamiliar and familiar faces. For example, Bentin and Deouell (2000) found that familiarity did not modulate the N170 face component and Haxby et al. (2000) did not find an effect on activation in the FFA. Burton and Jenkins (2011) noted that even when familiarity does
affect neuroimaging responses it is unclear whether these effects are quantitative (i.e., reduced access to configural processing for unfamiliar faces) or qualitative (i.e., configural processing is not available to unfamiliar faces).

**Testing the Unfamiliar-Face DFIE with State-Trace Analysis**

The same problem identified by Burton and Jenkins (2011) also plagues behavioural studies – null effects (e.g., no inversion effect or no internal feature advantage) might result from insufficient power to detect weaker effects produced by unfamiliar faces. In both cases the problem is accentuated by the generic inability of null-hypothesis statistical testing to quantify evidence favouring a simpler model (Wagenmakers, 2007). These problems also occur if state-trace plots are analysed using null-hypothesis statistical methods. As we continue to discuss across Chapters Five through Seven, these caveats were particularly apparent in Loftus et al.’s (2004) experiment with unfamiliar faces.
Chapter Five

Is the Differential Face Inversion Effect
Restricted to Memory Retrieval?

Adapted from:

Loftus, Oberg and Dillon (2004) noted a caveat to their results: the familiar face stimuli were photographs of celebrities but their unfamiliar face stimuli were Identikit (i.e., computer generated) images. This is problematic for two reasons. First as previously noted, these stimuli differ in that familiar celebrity faces can be given a verbal label. Second, and of particular importance to the unfamiliar-face experiment, Leder (1996, 1999) found that configural information is reduced for line drawings compared to photographs of faces. Although Yin (1969) observed a strong DFIE interaction with line drawings, each drawing was studied for three seconds, whereas Loftus et al. allowed between 17ms and 250ms for the study of each Identikit face. Valentine (1988) suggested that inversion and brief presentation time affect encoding in similar ways, and Hancock et al. (2000) summarised findings that configural information is difficult to obtain from brief presentations. The likely reason is that the configural encoding of faces is demanding both in terms of attention and time (Palmero & Rhodes, 2002), perhaps because face information must be present in visual short-term memory long enough for its parts to be integrated (Anaki, Boyd & Moscovitch, 2007).

Problems related to short study times were exacerbated by the fact that Loftus et al.’s (2004) design produced a very strong inversion effect. This occurred in their design because inversion was confounded with the encoding specificity effect (Tulving & Thomson, 1973), a robust improvement in memory performance that occurs for a range of stimuli when study and test conditions match (Nilsson & Gardiner, 1993). The encoding specificity effect has been found using several different matching manipulations with faces, including the match between study and test for the recognition of face parts and whole faces (Leder & Carbon, 2004) as well as the context in which faces are studied (Watkins, Ho & Tulving, 1976; Winograd & Rivers-
Bulkeley, 1977). Moreover, the encoding specificity effect can have a large magnitude (e.g., for Rakover & Teucher, 1997, it was larger than the face inversion effect). Because study and test orientations matched in Loftus et al.’s (2004) upright condition but mismatched in the inverted condition, the inverted condition was disadvantaged both by the inversion effect and the encoding specificity effect.

Consequently, in Loftus et al.’s (2004) unfamiliar-face experiment only the worst performance in the upright condition, for the 17ms study condition, overlapped the best performance in the inverted condition, for the 250ms study condition. A state-trace plot can only be non-monotonic – and so provide evidence for encoding in terms of more than one dimension – in a region of overlap (Prince, Brown & Heathcote, 2012). So, it was in the very region where evidence for configural processing could be observed that it was most unlikely to have occurred, because even in upright faces configural information is difficult to obtain from very briefly presented stimuli. Richler, Mack, Gauthier and Palmeri (2009) reported directly relevant evidence on this point from a sequential same-different matching task; a marker of holistic processing was absent when exposure duration was 17ms, emerging only for durations of 50ms or more. Thus, even if the longer study durations used by Loftus et al. were sufficient to allow configural processing for upright faces, their analysis would not detect it because it does not affect results in the region of the state-trace plot where overlap occurred.

We addressed this caveat in complimentary ways using two between-subjects conditions. The first of these conditions closely replicated Loftus et al.’s (2004) original design where items could be studied upright or inverted and all items were tested upright; hereafter referred to as the Upright-Test condition. It differed in that inverted study items had longer study durations (267-2048ms) than the upright study items (33-
267ms). These longer inverted-study times were selected based on a pilot experiment\(^{13}\) in order to counteract the deleterious effect of inversion on accuracy and so to maximise overlap between upright and inverted conditions.

We also attempted to use generally longer study durations but were limited in our ability to do so because the inversion effect is confounded with the encoding specificity effect in this Upright-Test design. Consequently, very large differences in study duration are required to compensate for the large deleterious effects on memory for inverted stimuli. Practical limitations did not allow us to use presentation times longer than 2 seconds for inverted stimuli and, so we had to use short study times for the upright items. We attempted to address this issue in our second condition where all items were tested inverted; hereafter called the Inverted-Test condition. Pilot testing revealed that the study-test match advantage for inverted-study items in this design almost exactly counteracted the inversion effect. Hence we were able to use the same longer set of study durations (267-2048ms) for both study-orientation conditions.

**Method**

**Participants**

Participants (75 in Upright-Test and 65 in the Inverted-Test conditions) were recruited from members of the wider community with the only restriction on participation that they had normal or corrected-to-normal vision and were comfortable completing a computer-based task. No demographics were recorded and participants did not receive incentives.

\(^{13}\) Although not included in the publication, Appendix B presents the methods and results for this pilot design (see also Chapter One for discussion of results).
Stimuli

Stimuli were black and white bitmap images (120x105 pixels) displayed at twice their original size. To address concerns about Loftus et al.’s (2004) use of Identikit faces, we used realistic photographic stimuli in all conditions. A total of 384 face stimuli were sourced from the FERET database (Phillips, Wechsler, Huang & Rauss, 1998), excluding images with glasses, averted gaze, distinctive facial expressions or natural or photographic blemishes. The faces were divided into homogenous blocks based on race and gender. In total there were 144 African American and 240 Caucasian, with half male and half female. An additional 12 Caucasian male faces were used in a practice phase.

A total of 384 house stimuli (with an additional 12 for practice) were sourced using real estate websites and internet search engines. Houses were excluded if located in New South Wales in order to reduce potential familiarity effects given that participants were drawn from this region. Pilot testing revealed significantly greater accuracy for house than face stimuli. Participant feedback suggested that certain house characteristics made them distinctive within the context of a particular study list. Therefore, house stimuli were presented in homogenous blocks based on their most distinctive feature (e.g., drive-way, fence, etc.).

Apparatus

Testing was completed either at individual computer terminals equipped with 17” LCD monitors or using laptop computers. All stimuli and text were presented on a black background with white font. Prospective and retrospective confidence judgments were made using the computer keyboard with the keys z x . / labelled 1 2 3 4 respectively.
**Procedure**

Testing sessions began with the experimenter reading through the instructions displayed on the participants’ screen. During these instructions it was emphasized that the orientation of a stimulus at study and test was irrelevant to their recognition decision. That is, that they should respond “old” even if the test item was studied in a different orientation. Participants then completed two half-length practice blocks, one using faces and one using houses with order counterbalanced over participants.

The start of a study list was marked by the warning “*Prepare for study. Place your fingers on the keys*” displayed for 2000ms. For each study trial a centrally placed fixation cross was displayed for 1000ms followed by a 300ms blank screen. The target stimulus was then presented for its designated duration. Participants then had a maximum of 2500ms to rate their prospective confidence by responding to the question “*How confident are you that you will remember this image later on?*” using a four-point scale (1 = ‘definitely no’, 2 = ‘probably no’, 3 = ‘probably yes’ and 4 = ‘definitely yes’). As in Loftus et al. (2004), the purpose of this prospective confidence judgment was to encourage participants to attend to the stimulus and the data from this response was not considered further.

After the study list a 300ms blank screen was followed by the warning “*Prepare for testing. Place your fingers on the keys*”, which appeared for 2000ms. Each test trial was preceded by a 300ms blank screen followed by the test stimulus and retrospective confidence rating scale. The test image was centrally positioned above the question “*How confident are you that you have seen this image earlier?*” and again participants responded using a four-point rating scale where 1 = ‘definitely new’, 2 = ‘probably new’, 3 = ‘probably old’, and 4 = ‘definitely old’. The next trial commenced as soon as the participant responded or if the 5000ms time limit expired. For the entire length of
the study and test lists the words “STUDY” and “TEST” were displayed respectively in
the top left corner of the screen.

Following the practice trials, participants received feedback on the number of
times they used each confidence level. The purpose of this feedback was to encourage
participants to use the full confidence scale. No feedback regarding accuracy was
provided. Participants then commenced the main experiment, which consisted of 32
study-test cycles (16 using face stimuli and 16 using houses). The order of testing face
and house stimuli was identical to the practice phase order, such that faces were tested
first for half of the participants and houses tested first for the remaining participants.
Each study list included 16 images (8 presented upright and 8 inverted), while test lists
included 24 images (16 previously studied and 8 new). A 10 second break occurred at
the end of each cycle and a 5 minute break occurred after 16 cycles.

**Results**

Overall, participants failed to respond on 0.31% of trials. A further 0.64% of test
responses were excluded for being faster than 150ms. Accuracy was defined using
Loftus et al.’s (2004) “p” measure. This measure was obtained by first transforming the
1-4 confidence rating (CR) by \((CR – 1) / 3\), then averaging to produce for each
participant what Loftus et al. refer to as a hit rate (HR) and false alarm rate (FA), where:
\[ p = (HR – FA) / (1 – FA). \]

We first report a preliminary analysis to ensure the longer durations used in the
present study were able to replicate Loftus et al.’s (2004) finding that accuracy was
linear as a function of the logarithm of study duration. One-way repeated measures
ANOVAs were performed on the effects of the logarithm of study duration for upright
and inverted houses and faces in each condition, with polynomial trend analysis. This
was followed up by two-way factorial ANOVAs examining the effect of stimulus type (house vs. face) and its interaction with duration.

Linear trends were all highly significant ($p < .001$) and accounted for almost all of the variance in accuracy (see Table 5.1 and Figure 5.1). No quadratic or cubic trends approached significance, with the exception of inverted faces (quadratic trend, $p = .04$) and upright faces (cubic trend, $p = .03$) in the Upright-Test design.

Table 5.1. 

*Proportion of variance in accuracy accounted for by a linear trend in the logarithm of study duration.*

<table>
<thead>
<tr>
<th></th>
<th>Inverted Houses</th>
<th>Inverted Faces</th>
<th>Upright Houses</th>
<th>Upright Faces</th>
</tr>
</thead>
<tbody>
<tr>
<td>Upright-Test</td>
<td>99.2</td>
<td>96.1</td>
<td>98.5</td>
<td>95.4</td>
</tr>
<tr>
<td>Inverted-Test</td>
<td>99.0</td>
<td>96.1</td>
<td>97.8</td>
<td>97.4</td>
</tr>
</tbody>
</table>
In the Upright-Test condition accuracy was greater for houses than faces for both inverted ($M_H = 0.30, M_F = 0.23$), $F(1, 74) = 21.66, p < .001$, and upright items ($M_H = 0.31, M_F = 0.26$), $F(1, 74) = 12.23, p = .001$. Accuracy also increased more quickly with study duration for houses than faces. This effect was reliable for upright, $F(3, 222) = 3.67, p = .01$, but not inverted, $p = .27$. Similarly, in the Inverted-Test condition, accuracy was higher for houses than faces for both inverted ($M_H = 0.34, M_F = 0.23$), $F(1, 67) = 57.77, p < .001$, and upright images ($M_H = 0.33, M_F = 0.24$), $F(1, 64) = 51.21, p < .001$. The stronger effect of study duration for houses than faces was reliable
for both inverted, $F(3, 192) = 5.79, p = .001$, and upright, $F(2.7, 173.8) = 3.71, p = .02$
(using a Huynh-Feldt correction to degrees of freedom).

We tested for the DFIE as traditionally defined by the interaction between orientation and stimulus type. In each condition the corresponding ANOVA used only study durations that were common to upright and inverted conditions. Table 5.2 also shows, for each duration, estimates of the inversion effect (i.e., difference between upright and inverted) for faces and houses, the corresponding DFIE estimates (i.e., inversion effect for faces minus the inversion effect for houses) and the results of associated tests.

Table 5.2.

*Estimates of the inversion effects (Upright – Inverted) for faces (FIE) and houses (HIE), differential face inversion effects (DFIE = FIE – HIE) and associated test results.*

<table>
<thead>
<tr>
<th>Duration (ms)</th>
<th>Upright-Test</th>
<th>Inverted-Test</th>
</tr>
</thead>
<tbody>
<tr>
<td>267</td>
<td>0.206***</td>
<td>0.061***</td>
</tr>
<tr>
<td>512</td>
<td>-0.005</td>
<td>0.021</td>
</tr>
<tr>
<td>1024</td>
<td>0.003</td>
<td>-0.003</td>
</tr>
<tr>
<td>2048</td>
<td>-0.004</td>
<td>-0.051**</td>
</tr>
</tbody>
</table>

Note: *** $p < .001$, ** $p < .01$, * $p < .05$

In the Upright-Test design, 267ms was the only common duration for upright and inverted items. The DFIE was, therefore, tested by a two-way (orientation by stimulus type) ANOVA using only the 267ms data. Accuracy was reliably greater for houses ($M = 0.32$) than faces ($M = 0.26$), $F(1, 74) = 15.97, p < .001$ and for upright ($M$
= 0.41) than inverted (\(M = 0.18\)), \(F(1, 74) = 225.88, p < .001\). The DFIE interaction test was marginally significant \(F(1, 74) = 3.21, p = .08\), but as shown in Table 5.2 the effect was in the opposite direction (a greater inversion effect for houses than faces).

The Inverted-Test condition was fully factorial, so the DFIE was examined using a three-way ANOVA that included a duration factor with all four levels. Accuracy was again reliably greater for houses \((M = 0.33)\) than faces \((M = 0.23)\), \(F(1, 64) = 73.49, p < .001\). However, there was no reliable difference in accuracy for upright \((M = 0.29)\) and inverted items \((M = 0.28)\), \(p = .48\). An overall DFIE of 0.017 was observed, but the corresponding interaction was not reliable, \(p = .24\), and, as shown in Table 5.2, neither were the DFIE estimates at any individual duration.

State-trace plots for each condition are shown in Figure 5.2. Results for upright study are joined, as are points for inverted study, and these lines are clearly monotonically increasing, consistent with the requirement that the trace factor have a monotonic effect. The plots also show excellent overlap between the two traces in both conditions.
Figure 5.2. State-trace plots for the (a) Upright-Test and (b) Inverted-Test conditions with Loftus and Masson (1994) standard errors.
Following Loftus et al. (2004) we examined the monotonicity of the overall plots in two ways. First, we calculated Spearman’s $\rho$, a measure of rank order correlation, where for $\rho = 1$ perfect monotonicity holds (i.e., the same ordering for points on both axes). Both conditions had the same value of $\rho$ which is close to one because there were only two inversions in the order for each axis. In the Upright-Test condition these were between the inverted and upright conditions for the two shortest study durations and in the Inverted-Test condition they were between the middle two durations. The second method involved adding standard errors appropriate for a within-subjects comparison (Loftus & Masson, 1994) to the plots. This aids a visual assessment of whether an inversion is likely to be reliable. For the Test Inverted design neither inversion appears reliable as the standard error bars for the inverted points overlap markedly. This is also clearly the case for the lower left pair in the Test Upright condition and even for the other pair, where the inversion is more marked, a decrease for inverted houses and an increase for upright houses of less than one standard error would be sufficient to remove the inversion.

**Discussion**

We replicated Loftus et al.’s (2004) finding of a linear increase in accuracy for study durations up to one quarter of a second and extended this result for durations up to two seconds. The fact that duration effects showed no discontinuity suggests that there is no abrupt change in strategy associated with longer study presentations (i.e., no switch from featural to configural processing). Given evidence that duration and inversion have similar memory effects (Valentine, 1998) this result is consistent with Barton, Keenan and Bass’ (2001) suggestion that inversion does not cause a sudden
change in encoding but rather reduces the rate at which featural and configural
information are extracted.

Like Loftus et al. (2004) we found little evidence for a DFIE using the
traditional interaction measure. Although Loftus et al.’s tests were statistically reliable,
the magnitude of their effect was very small (0.042) and was not much different from
our results for some durations, which were not reliable. As Loftus et al. (2004)
demonstrated in an extensive set of simulations, such inconstancy in the interaction
measure of the DFIE is to be expected. In contrast, our state-trace results were largely
consistent with Loftus et al.’s, although they observed no evidence of non-monotonicity
(\(\rho = 1\)) whereas we obtained some weak evidence for the occasional inversion of data
points along each axis. Likely this was due to the much greater overlap between the
data-traces in our experiment, which increased the likelihood of chance inversions
occurring.

These results are not only consistent with Loftus et al.’s (2004) assertion about
the ineffectiveness of study inversion for unfamiliar faces, but also further strengthens
their conclusion in three ways. First, it shows the prediction is not dependent on the
orientation at which all items are tested, as our state-trace results were essentially the
same when all items were tested upright and when all were tested inverted. Second, the
Inverted-Test condition extended Loftus et al.’s finding to much longer study durations
where it is unlikely that insufficient study time was available to engage configural
encoding. Finally, through the use of photographic stimuli, the current experiment
showed that Loftus et al.’s finding cannot be attributed to their use of computer-
generated Identikit faces.

Despite results consistent with Loftus et al.’s (2004) conclusion that the DFIE
occurs only when recognizing faces already stored in memory, this conclusion is
surprising given the widely held view that the “face inversion effect is really a perceptual phenomenon rather than a memory phenomenon” (Freire, Lee & Symons 2000; p.160). An alternate explanation more compatible with this view would be possible if participants can strategically use the results of configural processing. That is, if configural encoding is not an automatic process but rather one that participants will utilize only when they know it will improve performance for all items. For example, if an item was encoded using purely featural information (e.g., studied inverted) it might be detrimental to use configural information at test, as suggested by the encoding specificity effect; that is, if only featural information is available from study, performance would benefit from a matched (featural) test encoding but hurt by a mismatched (configural) encoding.

Participants in the Inverted-Test condition may have relied completely on featural processing because configural information at test was not available (or was too difficult to extract) from the inverted test images. In contrast, participants in the Upright-Test condition had configural information available but may have instead relied on featural processing because they had no way of knowing for which items the configural information would be detrimental (i.e., those studied inverted). We examine this possibility in Chapter Six (Prince & Heathcote, 2010).
Chapter Six

The Strategic Use of Configural Processing

Adapted from:

Using state-trace analysis, Loftus, Oberg and Dillon (2004) reported an apparent exception to the otherwise robust differential face inversion effect (DFIE); evidence for a single dimension in accuracy averaged over participants in recognition memory for unfamiliar faces (Experiment 1). In contrast, when the faces were famous (i.e., familiar; Experiment 2), they found evidence for more than one dimension. As images were studied upright or inverted and all tested upright in both experiments, Loftus et al. concluded that a DFIE would only emerge when inversion was present at the time of memory retrieval – a conclusion that opposes the general opinion in the literature which would suggest the DFIE “is really a perceptual … phenomenon” (Freire, Lee & Symons, 2000; p.160).

Prince and Heathcote (2009 – Chapter Five) addressed a number of caveats to Loftus et al.’s (2004) unfamiliar-face experiment: the overlap problem was addressed using longer study times for inverted (267-2048ms) than upright (33-267ms) stimuli, and they also used photo-realistic pictures of unfamiliar faces rather than Identikit stimuli. The longer study times in particular, counteracted the double disadvantage of the inverted condition (due to the detrimental effect of inversion and encoding specificity) and produced strongly overlapping state-trace plots. Yet like Loftus et al., they observed monotonic state-trace plots both when all items were tested upright (following Loftus et al.’s original design) and when all items were tested inverted.

In contrast to Loftus et al.’s (2004) memory retrieval hypothesis, Prince and Heathcote (2009) proposed an alternate explanation of these one-dimensional state-trace results that was more compatible with the widely held perceptual view – participants may be able to strategically use configural information when they know it will improve
performance for all items. Thus the use of configural encoding may not be automatic in recognition memory. Here we aim to further examine the one-dimensional state-trace evidence for unfamiliar faces, as well as Prince and Heathcote’s (2009) strategic hypothesis.

Our first condition partly replicated Prince and Heathcote’s (2009) *Upright-Test* design, with both upright and inverted study trials mixed in each study list and all items tested upright. However, we used a two-alternative forced choice (2AFC) recognition memory test, rather than the single-item testing used in the original study (i.e. on each test trial participants chose between a studied and unstudied face, or between a studied and unstudied house). We also greatly increased the number of observations obtained from each participant (78 observations per design cell), by increasing the number of trials and reducing the number of duration levels that an item could be studied for. These modifications were included to assess whether Prince and Heathcote’s (2009) one-dimensional result was replicable with a different testing procedure and with a slightly different, and more powerful, design in terms of more precise measurement at the individual participant level. Although Prince and Heathcote (2009) also attempted to achieve this by collecting more than six times as many observations as Loftus et al. (2004), overall the number of trials collected per participant was still low, as was the overall level of performance observed. Therefore, we further doubled the number of trials completed per participant here. This first condition is referred to as the *Mixed Upright-Test* design.

In the Mixed Upright-Test condition (and thus also Prince and Heathcote’s, 2009, Upright-Test design), an old item can either be studied and tested upright or studied inverted and tested upright. The former case has a matched (configural) encoding available at study and retrieval. However, when an image is studied inverted it
only (or at least mostly) can be encoded using featural information, yet configural information is available from the upright test presentation. As suggested by the encoding specificity effect (i.e., the improvement in memory when study and test conditions match; Tulving & Thomson, 1973), if only featural information was available at study, performance would benefit from a matched (featural) test encoding and be hurt by a mismatched (configural) test encoding. Hence it may be detrimental for participants to use configural information at test when an item has been studied inverted.

In these Upright-Test conditions, upright and inverted items were mixed together at study. Therefore, when all items are presented upright at test, participants have no way of knowing for which test items the use of configural information may be detrimental (i.e., those studied inverted). As these experiments used multiple study-test cycles participants would quickly become aware that all test items were upright. Hence it is possible that they decided to rely purely on featural information, either by not encoding upright study items along a configural dimension, or choosing not to use the configural information available at test. In either case, both faces and houses would only be encoded along a single (featural) dimension, producing the one-dimensional state-trace plots observed by Loftus et al. (2004) and Prince and Heathcote (2009).

Riesenhuber, Jarudi, Gilad and Sinha (2004) reported evidence consistent with the proposal that participants may not be able to switch between processing strategies in a perceptual paradigm. In their study, participants were required to make same-different judgements about pairs of faces that could differ either in their features or configural properties. Riesenhuber et al. found that when feature trials were completed first, performance on subsequent configural trials was poor, as would be expected if participants continued to use a feature-based strategy.
We aimed to test Prince and Heathcote’s (2009) strategic explanation of their state-trace evidence in our second condition, where participants viewed two types of study-test lists: (a) all items studied and tested upright or (b) all items studied inverted and tested upright. By blocking study orientation in this manner we hoped that participants would become aware of when configural encoding was advantageous (in type ‘a’ lists) and hence make use of it. If this occurred, we should observe multi-dimensional state-trace evidence, and hence support against Loftus et al.’s (2004) memory retrieval hypothesis. We denote this condition Blocked Upright-Test design.

Method

Participants

The 38 participants were recruited from members of the wider community, who had normal or corrected-to-normal vision. They received cash reimbursement for their time (total AUD$30.00). Two subjects in Mixed Upright-Test were excluded due to their raw percentage correct falling below 55%, leaving 18 subjects in Mixed Upright-Test and Blocked Upright-Test designs.

Stimuli

Stimuli were black and white bitmap images (120 x 105 pixels) displayed at twice their original size. A total of 936 face stimuli were sourced from the FERET database (Phillips, Wechsler, Huang & Rauss, 1998), excluding images with averted gaze, distinctive facial expressions or blemishes (either natural or the result of photographic process). These face stimuli were divided into homogenous blocks based on race, gender and any other distinctive feature (i.e., glasses or facial hair). An
additional 36 Caucasian males without facial hair or glasses were included for the practice phase.

A total of 936 house stimuli (with an additional 36 for practice) were sourced using real estate websites and internet search engines. Houses were excluded if located in New South Wales in order to reduce potential familiarity effects given that participants were largely drawn from this region. Following Prince and Heathcote (2009), house stimuli were also divided into homogenous blocks based on their most distinctive feature (e.g., fence, two-storey).

**Apparatus**

Testing was completed either at individual computer terminals equipped with 17inch LCD monitors or at an external location using laptop computers. All stimuli and text were presented on a black background with white font. Prospective and retrospective confidence judgments were made using the computer keyboard with the keys $z$ $x$ $.$ $/$ labelled 1 2 3 4 respectively.

**Procedure**

It was emphasised during the instructions for the task, that the orientation of a stimulus was irrelevant to a recognition decision; that is, participants should identify an image as being “old” even if the test item had been studied in a different orientation. In the Blocked Upright-Test condition, participants were further informed that study lists would be comprised of either all upright or all inverted images and a warning was displayed prior to each study list indicating the study orientation to be used. Before commencing the main experiment, participants completed two full length practice blocks; one for faces and one for houses, with order counterbalanced over participants.
A study list (comprised of 18 trials) was initiated by pressing the space bar, following which the warning “Prepare for study … of … Place your fingers on the keys” was displayed for 2000ms. For each study trial a centrally placed fixation cross was displayed for 1000ms, followed by a 300ms blank screen. The target stimulus was then presented for its designated duration (upright: 33, 100, 267ms; inverted: 267, 800, 2048ms), with durations selected to maximize data trace overlap and each duration level used equally often in every study list. After each study presentation, participants had a maximum of 2500ms to rate their prospective confidence by responding to the question “How confident are you that you will remember this image later on?” using a four-point scale from “definitely no” to “definitely yes”. The purpose of this prospective confidence judgment was to encourage participants to attend to the stimulus and this data will not be considered further.

The test list (again comprised of 18 trials) was marked by a 300ms blank screen, followed by the warning “Prepare for test … of … Place your fingers on the keys” displayed for 2000ms. Each test trial was preceded by a blank screen following which the test item and retrospective confidence response scale were presented for a maximum of 5000ms. For our 2AFC design, a pair of test images (one old and one new, with the old item appearing equally often on the left and right) were presented above the question “Which image was previously studied and how confident are you that you have seen this image earlier?” Again participants responded using a four-point scale from “definitely left” to “definitely right”. For the entire length of the study and test lists, the words “STUDY” and “TEST” were respectively displayed in the top left corner of the screen.
Following the practice study-test lists, participants received feedback on the number of times they used each of the confidence levels. The purpose of this feedback was to encourage participants to use the full range of the confidence scale.

Participants were required to attend three one hour sessions, preferably on consecutive days. Participants completed 12 study-test lists in their first session and 20 study-test lists in the latter two. At the end of each list participants were able to take a self-paced break (minimum of 30s), while three longer breaks (minimum of 5min) occurred within each one hour session.

**Results**

The retrospective confidence rating was used to determine a participant’s proportion correct (i.e., the number of trials correct divided by total number of trials). Accuracy was then quantified by the inverse cumulative normal (z) transformation of the probability correct.

We first report a preliminary analysis to ensure the present study was able to replicate previous findings that accuracy is linear as a function of the logarithm of study duration. One-way repeated measures ANOVAs were performed on the effect of the logarithm of study duration for upright and inverted houses and faces in each condition with polynomial trend analyses. Linear trends were all statistically reliable (p < .05) and accounted for almost all (minimum 88%) of the variance in accuracy as a function of study duration. The only quadratic trends to approach significance were for upright faces (p = .045) and upright houses (p = .075) in the Mixed Upright-Test design.

Evidence for the DFIE was first assessed by the traditional test of an interaction between orientation and stimulus type. As the 267ms duration level was the only study duration common to both upright and inverted stimuli, the DFIE was tested by a two-
way (orientation by stimulus type) ANOVA using only the 267ms data. Table 6.1 also shows estimates of the inversion effect (i.e., the difference between upright and inverted) for faces (FIE) and houses (HIE), the corresponding DFIE estimates (DFIE = FIE – HIE) and the results of associated tests.

Table 6.1.

*Estimates of the FIE, HIE, and DFIE and results associated tests, for the 267ms data.*

<table>
<thead>
<tr>
<th></th>
<th>FIE</th>
<th>HIE</th>
<th>DFIE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mixed Upright-Test</td>
<td>0.281**</td>
<td>0.270***</td>
<td>0.011</td>
</tr>
<tr>
<td>Blocked Upright-Test</td>
<td>0.274***</td>
<td>0.215***</td>
<td>0.059</td>
</tr>
</tbody>
</table>

Note: ***p < .001, **p < .01, *p < .05

For the Mixed Upright-Test condition, there was no reliable difference in accuracy between houses (M = 0.479) and faces (M = 0.411), p = .152. However, accuracy was reliably higher for upright items (M = 0.582) than inverted items (M = 0.307), F(1, 17) = 30.50, p < .001. Although a slightly greater inversion effect was observed for faces than houses (DFIE = 0.01), this effect was not statistically reliable, p = .91. Similarly for the Blocked Upright-Test condition, accuracy was higher for houses (M = 0.421) than faces (M = 0.409), but not reliably so, p = .83. Upright items were again reliably more accurate (M = 0.537) than inverted items (M = 0.293), F(1, 17) = 41.93, p < .001. However, there was no reliable DFIE (DFIE = 0.059; p = .42).

State-trace plots for each condition are shown in Figure 6.2. We also report results from Prince, Brown and Heathcote’s (2012) Bayesian state-trace procedures which select among four mutually exclusive models: (a) a *non-trace* (NT) model, which
assumes the trace factor does not have a monotonic effect on performance; (b) a no overlap (NO) model, which given that the trace model holds, assumes the data traces do not overlap and hence cannot be considered diagnostic of dimensionality; (c) a uni-dimensional (UD) model, which assumes the state-trace plot is monotonic, given that both trace monotonicity and data trace overlap hold; and (d) a multi-dimensional (MD) model, which assumes the state-trace plot is non-monotonic.

In Figure 6.2, results for upright study are joined, as are the results for inverted study. These lines are clearly monotonically increasing, and consistent with the requirement that the trace factor has a monotonic effect, both conditions’ group posterior model probabilities favoured the trace model being true, \( gp(NT) < .001 \). The plots also show excellent data trace overlap; for both the Mixed and Blocked designs \( gp(NO) < .001 \). In assessing the overall dimensionality, the Mixed Upright-Test design showed positive evidence for a multi-dimensional model, \( gp(MD) = 0.910 \), however, the Blocked Upright-Test design showed equivocal evidence suggestive of a one-dimensional account, \( gp(UD) = 0.733 \).
Figure 6.2. State-trace plots showing the 50% credible regions for the (a) Mixed Upright-Test and (b) Blocked Upright-Test conditions. The numbers 1…3 indicate shorter to longer study durations.
Discussion

We replicated Loftus et al.’s (2004) and Prince and Heathcote’s (2009) finding of a linear increase in accuracy consistent with the suggestion that there was no abrupt change in strategy (i.e., no switch from featural to configural processing) associated with longer study durations. Additionally, we replicated the lack of evidence for a DFIE using the traditional interaction measure (although the DFIE estimates were of the same magnitude as Loftus et al., and Prince & Heathcote). Our state-trace findings, however, were mixed.

We found clear multi-dimensional evidence consistent with the use of both featural and configural information for the Mixed Upright-Test design, where inversion was only manipulated during initial encoding and upright and inverted items were mixed together at study. However, for the Blocked Upright-Test design, where study lists were blocked by orientation, we observed evidence suggestive of a single underlying dimension (although at an equivocal level). In this blocked condition, participants were informed of an item’s study orientation if it was old and therefore, according to Prince and Heathcote’s (2009) strategic hypothesis, may have been able to reinstate the use of the configural dimension. The observed one-dimensional result, however, does not offer support for this proposal. This is not to say however, that our results are consistent with Loftus et al.’s (2004) memory retrieval hypothesis. Indeed the strong multi-dimensional result for the Mixed Upright-Test design cannot be explained by a memory retrieval interpretation, as orientation was only manipulated during initial encoding.

It is important to note that the posterior model probabilities on which we are basing our interpretations, are not simply the average result over participants. Rather they examine, for example, the probability that all individual state-trace results are one-
dimensional versus all being multi-dimensional. Hence these probabilities can sometimes be influenced by outlying subjects. This is of particular concern given the large range of individual differences that can occur in memory performance for unfamiliar faces (see Megreya & Burton, 2006). To ensure our results were not influenced in this way, we re-examined the model selection results, excluding participants with poor evidence ($p > .5$) for trace monotonicity and data trace overlap (four participants from the Mixed design and seven from the Blocked design were excluded using this criteria). However, both conditions revealed the same pattern of results; that is, multi-dimensional evidence for the Mixed Upright-Test design and equivocal evidence for the Blocked Upright-Test design. Although the Blocked condition did show a decrease in the probability supporting a one-dimensional model, $p(UD) = 0.691$.

One possible explanation for observing multi-dimensional evidence, even though inversion was only manipulated during the initial stimulus encoding, is that our more precise individual measurement also produced higher accuracy performance overall and consequently an improved effect size. Although state-trace analysis is not affected by floor and ceiling effects to the same degree as traditional dissociation analyses, if accuracy is not high enough to reveal the decrement caused by inversion then it will also not be able to reveal the underlying dimensionality. Consistent with this suggestion, we can observe from the Mixed Upright-Test state-trace plot that the data traces do not depart from monotonicity (indicating multi-dimensional evidence) until the longer study duration levels (where accuracy is also higher). This same pattern can also be seen to a lesser degree in the Blocked Upright-Test design.

We also ran these same mixed and blocked conditions using a yes/no testing procedure (i.e., participants were shown a single test item and asked to indicate if that
item had or had not been studied), and in contrast to our 2AFC results, observed equivocal one-dimensional evidence\textsuperscript{14}. Interestingly, it has been found that memory performance is advantaged by a 2AFC testing procedure over a yes/no procedure (Deffenbacher, Leu & Brown, 1981), which could explain why our 2AFC conditions tended towards multi-dimensional evidence. It should also be noted that recognition memory studies in general tend to show a smaller inversion effect than perceptually based studies (e.g., the Mixed Upright-Test condition showed a 9.97% drop in accuracy due to inversion, but perceptual tasks can show a decrement double this magnitude; see McKone & Yovel, 2009). Hence evidence for more than one dimension underlying face processing may only emerge when performance is high enough to reveal the decrement caused by stimulus inversion.

\textsuperscript{14} Although not reported in the published manuscript, these experiments are included in Appendix C.
Chapter Seven

Is the Encoding of Unfamiliar Faces Special?

Adapted from:

Refining Loftus, Oberg and Dillon’s (2004) Design

The results presented in the previous two chapters (Prince & Heathcote, 2009, 2010, respectively), did not provide conclusive evidence regarding Loftus, Oberg and Dillon’s (2004) memory retrieval hypothesis and hence the possibility that unfamiliar faces may not be “special”. The aim of this chapter is to provide a more robust exploration of this proposal. We do so by further refining Loftus et al.’s design and examining the conditions under which evidence for the multi-dimensional encoding of unfamiliar faces does occur.

Loftus et al.’s (2004) experiments were exemplary in terms of collecting data from hundreds of participants, and so both had the power to detect a small differential face inversion effect (DFIE) interaction and to produce small confidence intervals on state-trace plots of data averaged over participants. However, this strategy is only effective if data can be averaged without introducing distortions. Prince, Brown and Heathcote (2012 – Chapter One) provided examples where average state-trace plots were monotonic even though all participants had non-monotonic plots, and vice versa. Prince and Heathcote (2009 – Chapter Five) attempted to avoid this distortion by obtaining more precise results at the individual-participant level. To do so they simplified Loftus et al.’s design, by using a smaller number of different times for which each item was studied (study duration), and collected more than six times the number of trials per participant. Prince and Heathcote (2010 – Chapter Six) doubled the number again to 78 trials per condition (936 test trials per participant in total), the level used in the experiment reported here. A simulation study conducted by Hawkins, Prince, Brown and Heathcote (2010 – Chapter Two), which is discussed further in the following section describing the state-trace analysis of the present experiment, suggested that this
relatively large sample size (i.e., number of observations per participant) is required to obtain clear results about dimensionality using the methods developed by Prince, Brown and Heathcote (2012).

Prince and Heathcote (2009 – Chapter Five) addressed the problem of poor data-trace overlap in Loftus et al.’s (2004) unfamiliar face experiment by using longer study durations for inverted (267-2048ms) than upright (33-267ms) stimuli, and they also used pictures of unfamiliar faces rather than Identikit stimuli. Although, the longer study durations produced strongly overlapping state-trace plots, like Loftus et al. they found monotonic state-trace evidence. However, also like Loftus et al., overall accuracy was quite low and the DFIE interaction (measured at the one time common to inverted and upright conditions, 267ms) was very weak. The problem with low overall accuracy is that floor effects blunt measurement sensitivity. Although state-trace analysis deals with confounding due to differential floor effects (i.e., differences in the compression of measurements among conditions), it is just as subject to the deleterious effects of insensitive measurement as any other analysis.

Prince and Heathcote (2010 – Chapter Six) achieved better overall performance with a two-alternative forced choice recognition procedure, which is known to improve performance over yes-no recognition (e.g., Deffenbacher, Leu & Brown, 1981). They also made it possible to obtain more observations per condition by reducing the number of study duration levels to three: (33ms, 100ms, 267ms) for upright and (267ms, 800ms, 2048ms) for inverted (Loftus et al., 2004 used 5 levels and Prince & Heathcote, 2009, four levels). State-trace plots from the same type of mixed upright and inverted study as used by Loftus et al. (2004) showed a clear trend to be non-monotonic, particularly for higher levels of performance, but the DFIE interaction remained relatively weak.
Experiment

The original design used by Yin (1969), where study and test orientations are matched, is by far the most common in the literature, and has consistently produced much stronger DFIE interactions than observed by Loftus et al. (2004). In order to obtain a strong DFIE interaction we used this Matched-Test design as one of three between-subjects conditions in the present experiment. Pilot work indicated a reduction in the inversion effect relative to Loftus et al.’s design, as inverted items were no longer disadvantaged by a study-test mismatch. This allowed us to obtain good state-trace overlap with more similar study times for upright (66ms, 200ms, 600ms) and inverted conditions (200ms, 600ms, 1800ms) than used by Prince and Heathcote (2009). Prince, Brown and Heathcote (2012) showed that state-trace analysis of accuracy in the Matched-Test design is valid with a two-alternative forced choice procedure, but not with yes-no responding, so we used the former response procedure here.

The remaining between-subjects conditions used the same stimuli and testing procedure as the Matched-Test condition. The Upright-Test condition followed Loftus et al.’s (2004) design, manipulating inversion only during study. The Upright-Study condition used Scapinello and Yarmey’s (1970) design, where all items were presented upright at study and then tested either upright or inverted. In both designs the inversion manipulation is confounded with encoding-specificity, and so a strong inversion effect was expected. To ensure overlap in the state-trace plots, these conditions used Prince and Heathcote’s (2010) study durations.

Loftus et al.’s (2004) memory retrieval hypothesis predicts a weak or absent DFIE interaction in the Upright-Test condition. In contrast, a stronger and equivalent DFIE interaction is predicted in the Upright-Study and Matched-Test conditions where inversion is applied to images that have a representation stored in memory due to study.
However, if a strong memory representation is required, such as with the famous faces examined by Loftus et al., the brief study exposure used in the present experiment may not be sufficient, in which case no DFIE interaction should occur in any between-subjects condition.

Alternatively, if the locus of the DFIE is purely perceptual, the strongest interaction should occur in the Matched-Test design, where perceptual processing of inverted images occurs twice, whereas it should be present but weaker in the other two conditions. Further, if the perceptual effects are equivalent at study and test the DFIE would be equal for the Upright-Study and Upright-Test conditions. If however, both perceptual and mnemonic effects mediate the DFIE the magnitude of the interaction will be largest in the Matched-Test condition, intermediate in the Upright-Study condition and smallest in the Upright-Test condition. We focus on these predictions about the DFIE interactions in our initial analysis, and then report the results of the state-trace analysis.

**Method**

**Participants**

Participants had normal or corrected-to-normal vision and were recruited from 4th year psychology students participating in a class exercise at the University of Newcastle, New South Wales, Australia, as well as from the wider University community, who received a cash reimbursement for their time (total AU$30). Three participants from the Upright-Test condition, and two from the Upright-Study condition, were excluded due to their overall accuracy being less than 55%, leaving 25
participants in the Matched-Test condition, and 24 each in the Upright-Test and
Upright-Study conditions

**Stimuli and Apparatus**

Face and house stimuli were black and white bitmap images (150 x 120 pixels) displayed at twice their original size. The 936 unique face stimuli were sourced from the FERET database (Phillips, Wechsler, Huang & Rauss, 1998) and excluded images with averted gaze, distinctive facial expressions or blemishes (either natural or the result of photographic process). To avoid the occurrence of distinctive visual features within the context of a study-test block and the use of distinctive verbal labels, the face images were divided into categories based on race, gender and the presence of glasses and facial hair (see Figure 7.1 for examples). An additional 36 images of Caucasian males without facial hair or glasses were used for the practice phase.

*Figure 7.1. Example upright face stimuli, from left to right (top row) Caucasian female (n = 288), Caucasian male (n = 264), Caucasian male with facial hair (n = 72), Caucasian male with glasses (n = 36), (bottom row) African American female (n = 72), African American male (n = 72), African American male with facial hair (n = 36), Asian female (n = 48) and Asian male (n = 96). Inverted images were rotated 180°.*
The 936 unique house stimuli (plus an additional 36 images for practice) were sourced from Australian real estate websites and excluded properties located in New South Wales to minimize any potential familiarity. The house images were also divided into homogenous categories based on their most distinctive feature (see Figure 7.2 for examples).

*Figure 7.2. Example upright house stimuli, from left to right (top row) angular roof \( (n = 36) \), centre drive-way \( (n = 48) \), left drive-way \( (n = 36) \), right drive-way \( (n = 72) \), (middle row) fence \( (n = 48) \), grand \( (n = 36) \), modern \( (n = 36) \), elaborate multi-storey \( (n = 144) \), standard multi-storey \( (n = 72) \), (bottom row) road \( (n = 48) \), set back \( (n = 108) \), standard \( (n = 144) \), standard front-on view \( (n = 120) \) and vegetation \( (n = 108) \). Inverted images were rotated 180°.*

Testing was completed at individual computer terminals equipped with 17 inch LCD monitors or using laptop computers. Screen background was black and fonts white. Responses were made using a standard QWERTY keyboard, with the keys z x . /
labelled 1 2 3 4 respectively. Whenever a response was required a visual analogue of the appropriate confidence scale was arrayed at the bottom of the screen.

**Procedure**

The testing session began with the experimenter reading through instructions, which were also displayed on the participants’ screen. Participants were informed of the different study-test combinations that would occur for their specific between-participant condition. For the Matched-Test condition, it was emphasised that a study image would never be tested in the opposite orientation to its study presentation. For the Upright-Study and Upright-Test conditions it was emphasised that a study image may or may not be tested in the same orientation, but that the orientation of a stimulus was irrelevant to the recognition decision; that is, an image should be identified as “old” even if it had been studied in a different orientation. Before commencing the main experiment participants completed two full-length practice blocks; one for faces and one for houses, with order counterbalanced over participants.

Each study list had 18 items and was initiated by pressing the space bar, following which a message asking participants to prepare for study and to place their fingers on the response keys was displayed for 2000ms. A centrally placed fixation cross marked the start of a study trial and was displayed for 1000ms, followed by a 300ms blank screen. The target stimulus was then presented for its designated duration, which was determined by the experimental condition and the item’s orientation (see Table 7.1). Study times were approximately logarithmically spaced as Loftus et al. (2004) found a linear effect of the logarithm of study time on accuracy, and Prince, Brown and Heathcote (2012) recommend evenly spaced effects of the trace factor to
obtain the most sensitive state-trace design. Each duration level was used equally often in every study list.

Table 7.1.
Summary of experimental conditions and corresponding study durations.

<table>
<thead>
<tr>
<th>Orientation</th>
<th>Study Duration (ms)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Study</td>
</tr>
<tr>
<td>Upright-Test</td>
<td>Upright</td>
</tr>
<tr>
<td></td>
<td>Inverted</td>
</tr>
<tr>
<td>Matched-Test</td>
<td>Upright</td>
</tr>
<tr>
<td></td>
<td>Inverted</td>
</tr>
<tr>
<td>Upright-Study</td>
<td>Upright</td>
</tr>
<tr>
<td></td>
<td>Upright</td>
</tr>
</tbody>
</table>

After each study presentation, participant had a maximum of 2500ms to rate prospective confidence by responding to the question ‘How confident are you that you will remember this image later on?’ using a four-point scale: ‘definitely no’, ‘probably no’, ‘probably yes’, ‘definitely yes’. The question and corresponding visual analogue scale remained on the screen until participants’ responded or the time limit expired. The purpose of this prospective confidence response, also used by Loftus et al. (2004), was to encourage participants to attend to the stimulus, and prospective responses are not considered further.

Each test list had 18 item pairs, and was preceded by a 300ms blank screen followed by a message displayed for 2000ms requesting participants to prepare for testing and to place their fingers on the response keys. A blank screen preceded each
test trial for 300ms, following which the test pair and retrospective confidence response scale were presented for a maximum of 5000ms. Test pairs had one old (studied) and one new (not studied) image, with the old item appearing equally often on the left and right, and were presented simultaneously above the question ‘Which image was previously studied and how confident are you that you have seen this image earlier?’. Participants responded using a four-point scale, ‘definitely left’, ‘probably left’, ‘probably right’, ‘definitely right’, that remained on the screen with the test stimuli until participants made a response or the time limit expired.

For the entire length of the study and test lists, the words ‘STUDY’ and ‘TEST’ were respectively displayed in the top left corner of the screen. Following the practice study-test lists, participants received feedback on the number of times they used each of the confidence levels. The purpose of this feedback was to encourage participants to use the full range of the confidence scales. No accuracy feedback was provided.

Participants were required to attend three one-hour sessions on consecutive days. They completed 12 study-test blocks in their first session and 20 study-test blocks in the latter two. The preparation messages marking the start of each list informed participants of their progress through these blocks within a testing session. At the end of each block participants were able to take a self-paced break (minimum 30 seconds); three longer breaks (minimum 5 minutes) occurred approximately equally spaced within each one-hour session.

**Results**

Overall participants failed to respond on 0.5% of trials. A further 0.4% of trials were excluded for being faster than 200ms. The retrospective confidence rating was used to determine if a participant identified the image on the left (i.e., ratings 1 and 2) or
right (i.e., ratings 3 and 4) as being old. These recognition decisions were then used to
determine accuracy; overall accuracy for each participant ranged from 57% to 82%,
with an average of 68.4% (see Figure 7.3). Inferential tests of accuracy used the inverse
cumulative normal ($z$) transformation of the proportion correct.

*Figure 7.3.* Proportion correct for faces (top row) and houses (bottom row) as a function of the
logarithm of study duration for the Upright-Test, Matched-Test, and Upright-Study conditions,
with standard error bars calculated using Morey’s (2008) method for within-subject designs.

One-way repeated measures ANOVAs were performed on the effect of the
logarithm of study time for upright and inverted faces and houses, with polynomial
trend analyses. This was followed by two-way factorial ANOVAs examining the effect
of stimulus type (houses vs. faces) and its interaction with study duration. As shown in
Figure 7.3, the increase in accuracy with the logarithm of study time was mostly linear. All linear trends were statistically reliable (all $p$’s < .001) and accounted for almost all of the variance in accuracy (see Table 7.2), with the only one quadratic trend significant, for inverted faces in the Upright-Study condition ($p = .008$).

Table 7.2.
Proportion of variance in accuracy (%) accounted for by a linear trend in the logarithm of study duration.

<table>
<thead>
<tr>
<th></th>
<th>Upright</th>
<th>Inverted</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Faces</td>
<td>Houses</td>
</tr>
<tr>
<td>Upright-Test</td>
<td>93.71</td>
<td>94.85</td>
</tr>
<tr>
<td>Matched-Test</td>
<td>97.46</td>
<td>97.18</td>
</tr>
<tr>
<td>Upright-Study</td>
<td>93.55</td>
<td>94.19</td>
</tr>
</tbody>
</table>

For the Upright-Test condition, accuracy was reliably greater for houses ($M = 68.3\%$) than faces ($M = 64.3\%$) when presented inverted, $F(1, 23) = 9.01, p = .006$, but there was no reliable difference in accuracy for upright items ($p = .741$). The linear increase in accuracy with the logarithm of study duration was also stronger for houses than faces when presented inverted, $F(2, 46) = 5.61, p = .007$, but there was no difference in the linear trends when presented stimuli were upright ($p = .95$). For the Upright-Study condition, accuracy was reliably greater for houses ($M = 69.9\%$) than faces ($M = 65.4\%$) for inverted stimuli, $F(1, 23) = 14.21, p = .001$, but did not reliably differ between faces and houses for upright presentations ($p = .44$), nor was there a reliable interaction between stimulus type and study duration for either upright ($p = .86$) or inverted ($p = .14$) stimuli.
For the Matched-Test condition, house accuracy ($M = 71.3\%$) was greater than face accuracy ($M = 64.9\%$) for inverted stimuli, $F(1, 24) = 30.26, p < .001$. In contrast, accuracy was significantly higher for faces ($M = 72.6\%$) than houses ($M = 69.6\%$) for upright stimuli, $F(1, 24) = 4.45, p = .015$. Moreover, the linear increase in accuracy with the logarithm of study duration was stronger for houses than faces when stimuli were presented inverted, $F(2, 48) = 4.34, p = .019$, but not when presented upright ($p = .25$).

DFIE interactions for each experimental condition were tested using data from duration levels that were common to both upright and inverted items. Table 7.3 shows, for each common duration, estimates of the inversion effect (i.e., the difference in accuracy for upright and inverted items) for faces (FIE) and houses (HIE), the corresponding DFIE estimates ($DFIE = FIE – HIE$) and the results of associated tests.

Table 7.3.

*Estimates of the inversion effects (upright minus inverted) for faces (FIE) and houses (HIE), the DFIE estimates (FIE - HIE) and associated test results. Estimates are expressed as the proportion correct (%).*

<table>
<thead>
<tr>
<th>Study Duration</th>
<th>FIE</th>
<th>HIE</th>
<th>DFIE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Upright-Test</td>
<td>267ms</td>
<td>9.2***</td>
<td>8.0***</td>
</tr>
<tr>
<td>Matched-Test</td>
<td>200ms</td>
<td>12.6***</td>
<td>5.8***</td>
</tr>
<tr>
<td></td>
<td>600ms</td>
<td>14.3***</td>
<td>4.2**</td>
</tr>
<tr>
<td></td>
<td>Overall</td>
<td>13.4***</td>
<td>5.0***</td>
</tr>
<tr>
<td>Upright-Study</td>
<td>267ms</td>
<td>12.9***</td>
<td>8.4***</td>
</tr>
</tbody>
</table>

Note: * $p < .05$, ** $p < .01$, *** $p < .001$

In the Matched-Test design, there were two common durations (200ms and 600ms), so the DFIE was examined using a three-way (stimulus type by orientation by
study duration) repeated-measures ANOVA. Overall, house ($M_H = 70.3\%$) and face ($M_F = 69.4\%$) accuracy did not differ statistically ($p = .93$). Accuracy was reliably greater for upright ($M_U = 74.4\%$) than inverted ($M_I = 65.2\%$) items, $F(1, 24) = 82.76, p < .001$. The overall DFIE was 8.4%, $F(1, 24) = 22.03, p < .001$, and as shown in Table 7.3, a reliable DFIE was also observed at each of the individual durations. The interaction between these DFIE estimates and study duration did not achieve significance ($p = .161$).

For the Upright-Test and Upright-Study conditions, the only common duration was 267ms, so the DFIE was assessed by a two-way (stimulus type by orientation) ANOVA. House and face accuracy did not differ statistically in either the Upright-Test ($M_H = 66.8\%, M_F = 66.3\%; p = .73$) or the Upright-Study ($M_H = 69.2\%, M_F = 68.6\%; p = .73$) conditions. Accuracy was reliably greater for upright than inverted items in both the Upright-Test ($M_U = 70.9\%, M_I = 62.2\%), F(1, 23) = 37.45, p < .001$, and Upright-Study ($M_U = 74.2\%, M_I = 63.6\%)$ conditions, $F(1, 23) = 49.41, p < .001$. Although a stronger inversion effect was observed for faces than houses in both the Upright-Test (DFIE = 1.2%) and Upright-Study (DFIE = 4.5%) conditions, neither difference approached statistical significance ($p = .71$ and $p = .093$ respectively).

A comparison was also conducted between the Upright-Study and Upright-Test conditions to examine if manipulating orientation during the encoding phase (i.e., Upright-Test) or retrieval phase (i.e., Upright-Study) modulated inversion effects using only the common 267ms duration. For the FIE and HIE estimates, this interaction was assessed using two-way (orientation by design) mixed ANOVAs for faces and houses respectively. In both cases, there was no reliable difference between manipulating orientation at encoding or retrieval for the FIE ($p = .167$) or HIE ($p = .590$). For the DFIE, assessed using a three-way (stimulus type by orientation by design) mixed
ANOVA, the interaction between the DFIE estimate and encoding vs. retrieval orientation manipulations was not significant \((p = .346)\).

**Discussion**

The most accurate overall performance, and the strongest evidence for a DFIE interaction, was found in the Matched-Test condition. This is consistent with Yin’s (1969) original finding as well findings in the many other studies that have presented items in the same matched orientation at study and test. In the Matched-Test condition there was a trend for the DFIE to increase with study duration, with a 7% DFIE estimate for 200ms study increasing to a 10% estimate for 600ms study, supporting the idea that configural information is easier to obtain from longer presentations (Hancock et al., 2000).

In the Upright-Study condition the estimate of the DFIE interaction was about half as large as in the Matched-Test condition, but was only marginally significant. However, given Yarmey (1971) and Scapinello and Yarmey (1970) found significant DFIE interactions in their Upright-Study designs it seems likely the marginal significance is due to a lack of power in our experiment.

The smallest DFIE interaction effect was measured in our Upright-Test condition, which although was in the predicted direction (i.e., faces were more affected by inversion than houses), did not achieve significance. Once again the lack of significance is likely due to a lack of power, given Loftus et al. (2004) reported a similar magnitude DFIE interaction to the Upright-Study design that was significant for the more than an order of magnitude greater number of participants tested.

The ordering of the DFIE interaction estimates – greatest in the Matched-Test condition, intermediate in the Upright-Study condition and smallest in the Upright-Test
condition – is as predicted by there being both a perceptual (Freire et al., 2000) and mnemonic (Loftus et al., 2004; Valentine, 1988) locus for the DFIE, although the difference between the Upright-Study and Upright-Test DFIE interactions did not achieve significance. However, it is clearly the case that these results reject Valentine’s assertion that “the orientation of the inspection [study] series does not appear to be critical … [and that] … the disproportionate effect of face inversion only emerges when the task involves recognising a face as one stored in memory” (p. 474).

In all three between-subject conditions we replicated Loftus et al.’s (2004) finding of a linear increase in accuracy with the logarithm of study time for durations up to two seconds. Plots of the study time effect (Figure 7.3) showed no discontinuity that might indicate an abrupt change in processing strategy associated with longer study times (i.e., use of configural information). An even spacing between accuracy at each study time was shown by Prince, Brown and Heathcote (2012) to improve the power of state-trace analysis, so these results suggest our data is well suited for that purpose.

In the next section we use the data from our experiment for the main purpose for which it was designed, state-trace analysis. Because our DFIE interaction analysis could only use a subset of the data (i.e., study durations common to upright and inverted conditions), and because our design emphasised a large number of observations per participant rather than a large number of participants, it lacked power. State-trace analysis, in contrast, uses all of the data and takes advantage of the differences between upright and inverted study times to counteract lower accuracy in the inverted condition. Figure 7.3 shows our choice of study durations were successful in obtaining a strong overlap in performance for inverted and upright conditions (i.e., the solid and dashed lines cover a similar range of accuracy in each column of panels).
State-Trace Analysis

The results of a state-trace analysis can be represented by a scatterplot, usually constructed from a three-factor design. The state factor defines the axes of the plot and, as shown in Figure 7.4, in the present case it is constituted of accuracy for faces and houses. A second factor, the dimension factor, defines sets of points on the plot and is manipulated with the aim of differentially influencing the underlying dimensionality. In our case image orientation is the dimension factor, with points in Figure 7.4 joined by solid lines corresponding to the upright condition and points joined by dashed lines corresponding to the inverted condition.

The feature of a state-trace plot that is diagnostic of latent dimensionality concerns whether or not all points in the plot can be joined by a single monotonic function. When the state and dimension factors each have only two levels – for example the 2 (stimulus type) x 2 (orientation) design traditionally used to assess a DFIE interaction – a state-trace plot has only two points and so is necessarily monotonic. In these situations, a third factor, called the trace factor, such as study time, is introduced to trace out a set of points within each level of the dimension factor. Once the state-trace plot has more than two points non-monotonicity, and hence a test of dimensionality, is possible.

The trace factor is chosen because it has a monotonic effect on performance, so that any non-monotonicity in the state-trace plot can be unambiguously attributed to the interaction of the state and dimension factors. Both Figure 7.3 and Figure 7.4 (where increasing study time within inverted and upright conditions correspond to the numbers 1-3) show that the trace factor had a monotonic effect in our experiment.
Figure 7.4. State-trace plots of two-alternative forced choice accuracy (i.e., hit rate, denoted HR) for the (a) Upright-Test, (b) Matched-Test and (c) Upright-Study conditions. Symbols represent the modes of the posterior estimates from the encompassing model (posterior means and medians were almost identical), lines are the data traces, and ellipses represent the 68% credible regions (i.e., regions containing 68% of posterior samples).
Con conventionally lines called a *data traces* join the points within each level of the dimension factor (i.e., the solid and dashed lines in Figure 7.4). Prince, Brown and Heathcote (2012) noted that, although a non-monotonic plot rejects a one-dimensional explanation, the converse is not necessarily true. In particular, a monotonic plot is only diagnostic of one latent dimension when data traces overlap on at least one axis. When data traces fail to overlap, a state-trace plot will be monotonic – assuming the effect of the trace factor is monotonic – even if performance is a function of multiple latent dimensions. Figure 7.4 shows that we obtained good data-trace overlap in all between-subject conditions.

Each point in Figure 7.4 is accompanied by a representation of the region of measurement uncertainty, the Bayesian equivalent of a standard-error region. A common approach to state-trace analysis is simply to observe whether a monotonic line can join all regions of uncertainty. Clearly this was not the case in any panel of Figure 7.4. The Upright-Test condition (Figure 7.4a) comes closest, but fails due to the points for the two longest durations. Prince and Heathcote’s (2010) Upright-Test condition also showed the clearest evidence of non-monotonicity where responding was more accurate. Figure 7.4a is also consistent with Loftus et al.’s (2004) results, in that the four lowest accuracy points, two each from the upright and inverted conditions, are monotonically increasing. That is, based on lower accuracy points alone the state-trace analysis indicates a one-dimensional outcome.

Much stronger non-monotonicity was displayed in the Matched-Test condition (Figure 7.4b). Face accuracy was much greater for upright than inverted under the same conditions that upright and inverted accuracy are the same for houses, resulting in the upright condition data trace being shifted to the right of the inverted condition data trace. Intuitively, this shift to the right can be thought of as due to the improved
accuracy for faces afforded by configural encoding. The same pattern is evident in the Upright-Study condition (Figure 7.4c), although the shift to the right is weaker overall and negligible for the two shortest study duration in each orientation condition, consistent with the lower accuracy results in Figure 7.4a.

Unfortunately, this informal analysis of data averaged over participants by graphical means has a number of potential weaknesses. Besides the possibility of distortion caused by averaging mentioned previously, outcomes can depend on how the region of uncertainty is defined. One approach could be to define the regions so overlap indicated a failure to obtain a significant difference, such as is provided by a least-significant difference interval (Saville, 2003). However, this method still has all of the many weakness of null-hypothesis testing (Wagenmakers, 2007), with the most important in the context of state-trace analysis being that it cannot provide positive evidence for monotonicity, due to an inability to prove the null.

Prince, Brown and Heathcote (2012) proposed an alternate method for assessing dimensionality using a Bayes-factor method of selecting amongst models defined by order restrictions in the state-trace plot. Bayes factors (Kass & Raftery, 1995) quantify the change in relative beliefs (odds) about two models caused by observing the data; that is, the change from prior odds (odds before seeing the data) to posterior odds (odds after seeing the data). The change for one model contributes to the numerator of the Bayes factor and the odds for the other to the denominator. A Bayes factor close to one indicates relatively equal evidence for both models, and so the data is unable to adjudicate between the models (similar to the idea of a lack of power). Large Bayes factors favour the numerator model and small Bayes factors the denominator model.

Prince, Brown and Heathcote’s (2012) analysis quantifies evidence relevant both to refining a state-trace experiment and to diagnosing dimensionality by selecting
among four mutually exclusive models, which together account for all possible orders in a state-trace plot:

1. **Non-trace model**: The trace factor does not always have a monotonic effect and so any non-monotonicity in the state-trace plot cannot be unambiguously attributed to the state by dimension interaction.

2. **No-overlap model**: Data traces do not overlap on either axis, and so no conclusions can be made about dimensionality from a monotonic state-trace plot.

3. **One-dimensional model**: The state-trace plot is monotonic and provides a valid basis for inference about dimensionality; that is, performance is mediated by a single latent variable.

4. **Multi-dimensional model**: The state-trace plot is non-monotonic; that is, performance is mediated by more than one latent variable.

Bayes factors for each member of this set of models relative to a model containing all possible orders, the encompassing model (Klugkist, Laudy & Hoijtink, 2005), are combined to obtain values quantifying the relative evidence for each model called *posterior model probabilities*. If it is assumed that one member of the set is the true (i.e., data generating) model these values can be interpreted as the probabilities that each model is the true model.

Prince, Brown and Heathcote (2012) argued that their Bayesian procedure is particularly suited to state-trace analysis as: (a) it only requires a minimal additional statistical assumption for accuracy data, that the binary data are binomially distributed; (b) it takes into account that hypotheses about different dimensionalities vary greatly in their ability to fit data by chance, and so does not inappropriately favour more flexible models; and (c) in contrast to null hypothesis statistical testing, it can quantify evidence
in favour of a simpler “null” (e.g., one-dimensional) model as well as evidence against
the null model and evidence that the data are equivocal.

We used Prince, Hawkins, Love and Heathcote’s (2012 – Chapter’s Three & Four) StateTrace package for R (R Development Core Team, 2012) to perform Prince, Brown and Heathcote’s (2012) analysis and to produce Figures 7.4 and 7.5. For each participant we calculated Bayes factors weighing the evidence for four models, which were then combined to calculate posterior model probabilities that are shown in Figure 7.5. In Figure 7.5 letters denote each participant within a condition. The light dotted lines on the plot indicate descriptive terms applied to posterior model probabilities by Raftery (1995): .25 < p < .75 indicates equivocal evidence, .05 < p < .25 and .75 < p < .95 positive evidence and p < .05 or p > .95 strong evidence. High probabilities indicate evidence for a model and low probabilities evidence against a model.
Figure 7.5. Individual participant (each denoted by a letter) posterior model probabilities for the (a) Upright-Test, (b) Matched-Test and (c) Upright-Study conditions. The light dotted lines are Raftery’s (1995) conventions for interpreting posterior model probabilities and the dark dotted line is the aggregated group result. Note that the probabilities for each participant sum to one across the four models.
For all but a few participants it is clear that both the non-trace and no-overlap models can be rejected, consistent with the average results in Figures 7.3 and 7.4 indicating a monotonic effect of study time and good overlap of data traces. Selection between the one-dimensional and multi-dimensional models at the individual participant level is more equivocal. These equivocal results might be taken as supporting heterogeneity amongst participants, with some using configural encoding and some not, consistent with the strong individual differences in performance found for unfamiliar faces (e.g., Megreya & Burton, 2006). However, Hawkins et al. (2010) showed this is not necessarily the case. In a study simulating results from the design used in the present experiment they found a similar spread of posterior model probabilities at the individual participant level when the data-generating model was multi-dimensional for every participant.

Hawkins et al.’s (2010) study also showed that – as long as the sample size per condition per participant was as large as was used in the present experiment – group Bayes factors can be used to perform a valid test between the hypothesis that all participants use one-dimensional processing and the hypothesis that all individuals use multi-dimensional processing. A group Bayes factor equals the product of the individual-participant Bayes factors (Prince, Brown & Heathcote, 2012). Combining Bayes factors calculated from each participant’s data in this way provides a powerful test at the group level that does not risk the potential distortion caused by performing state-trace analysis on group average data. Group Bayes factors can then be used to calculate posterior model probabilities at the group level in the same way as at the individual level.

In the Upright-Test condition the group posterior probability favouring the multi-dimensional model was 0.946. This increases to 0.953 if participants R and W
with positive evidence for the non-trace model are excluded, indicating strong evidence for the multi-dimensional model. In the Matched-Test and Upright-Study conditions both group posterior probabilities were greater than .999, indicating very strong evidence in favour of the multi-dimensional model. Hence, it appears that averaging did not distort the outcome in Figure 7.4, with all analyses converging on the conclusion that faces can be encoded using an extra dimension or dimensions not available to the house stimuli in every between-subjects condition in the present experiment (even if not necessarily for every participant).

**General Discussion**

Psychologists aim to understand how cognitive and neural representations and processes mediate observed relationships between stimuli and responses in different experimental paradigms. Even in simple paradigms there is little doubt that a plethora of mediating variables play a causal role. However, just as the coordinated macro-scale behaviour in a range of natural systems (e.g., molecules in a gas) can be characterized by a simple functional relationship among the appropriate variables (e.g., pressure, volume and temperature), psychological researchers seek to discover low-dimensional explanations of psychological phenomena in terms of key latent (i.e., not directly observed) variables. Success in this approach requires that Occam’s razor be wielded rigorously; elaborating a theory by adding an extra latent dimension requires strong evidence, otherwise genuine explanation becomes mere re-description. Our primary aim in the present chapter has been to determine whether such strong evidence supports faces being special in the sense that they can be encoded along a latent dimension not available to most other objects.
Faces are a complex class of stimuli that have been of interest in many areas of cognitive psychology and neuroscience including perceptual learning, object recognition and decision making, as well as social cognition and applied areas such as eyewitness identification (Riesenhuber & Wolff, 2009). This interest in the perception of, and memory for, faces has been driven by the practical significance of being able to identify faces that are familiar to us. Indeed, mature humans usually display remarkable expertise when identifying familiar faces, overcoming the considerable variation that can occur in the physical appearance of a face over time and the contexts in which they are seen.

One of the primary methods of assessing the special status of faces has been to compare the magnitude of the inversion effect for faces to that for a mono-oriented control stimulus class. The robust empirical finding that faces are differentially affected by stimulus inversion (i.e., the Differential Face Inversion Effect; DFIE) is typically taken as evidence that upright faces are processed differently – using both featural and configural dimensions – to inverted faces and to upright and inverted non-face stimuli, where the extraction of configural cues is either particularly difficult or not available.

We examined the DFIE in a recognition memory task using photographs of unfamiliar faces and houses. Our investigation focused on the methods used by Loftus et al. (2004) to examine evidence for the DFIE (State-Trace Analysis; Bamber, 1979), their conclusion based on state-trace results that completely unfamiliar faces cannot be encoded on a configural dimension even when upright, and their explanation based on Valentine’s (1988) hypothesis that the DFIE only emerges through retrieval from memory. All of our stimuli were unfamiliar to participants prior to the experiment, with the memory retrieval hypothesis being tested by comparing a condition in which stimuli were only inverted during study (the Upright-Test condition, and the same design as
used by Loftus et al.) to conditions in which inversion occurred only during the test phase of the recognition memory task (the Upright-Study condition), or during the study phase but also in a matched orientation for old (studied) items during the test phase, and in both upright and inverted orientations for new (not previously studied) items (the Matched-Test condition).

The strongest evidence for a traditional DFIE interaction was found in the Matched-Test condition, with the interaction only half the size, and only marginally significant, in the Upright-Study condition and smaller again and non-significant in the Upright-Test condition. Based on these dissociation (interaction) results we might be tempted to conclude that configural processing only emerges when orientation is manipulated during both the encoding and retrieval phases of a recognition memory task; that is, that participants do not use configural encoding when stimulus orientation is manipulated either at encoding or retrieval alone. However, as well as making the statistical error of accepting a null hypothesis, this conclusion is both difficult to justify theoretically and is inconsistent with our state-trace results. These state-trace results tell quite a different, and theoretically coherent, story, and a comparison of the two methods of analysis makes an important general point that a weak dissociation does not necessarily imply that performance is mediated by a single latent dimension.

In line with the strong DFIE interaction in the Matched-Test condition, we observed very strong multi-dimensional state-trace evidence when orientation was manipulated at both encoding and retrieval. This result is consistent not only the widely held view that inversion disrupts the perceptual encoding of faces but it is also consistent with Loftus et al.’s (2004) memory retrieval hypothesis. The latter of these explanations would argue that it is the manipulation of orientation during the test phase
of this condition that enabled configural processing or, as Valentine (1988) stated, that the orientation of the inspection series is not critical to the DFIE.

Our Upright-Study condition only manipulated orientation during the retrieval phase and although the DFIE interaction was not statistically reliable, the state-trace plot provided evidence supporting the multi-dimensional model that was just as strong as that observed for the Matched-Test condition. These state-trace results are consistent with Yarmey (1971) and Scapinello and Yarmey’s (1970) original finding of a significant DFIE interaction using the Upright-Study design with a between-subjects manipulation of inversion and both immediate and delayed recognition tasks. They are also consistent with Loftus et al.’s (2004) memory retrieval hypothesis, in that inversion at retrieval alone was associated with evidence of configural face processing. However, this result could be equally well explained by inversion disrupting the perceptual processes that support configural processing.

The Upright-Test design, which was based on Loftus et al.’s (2004) original paradigm, allows us to distinguish between these opposing memory and perceptual explanations. In contradiction to the predictions of the memory retrieval hypothesis, we again observed multi-dimensional state-trace evidence, even though, like Loftus et al., we observed only a very weak DFIE interaction. These state-trace results do clearly differ from the one-dimensional state-trace results observed by Loftus et al. (2004). However, this difference can be explained by the combined effects of a number of methodological improvements: (a) use of photographic stimuli for both faces and houses; (b) use of a two-alternative forced-choice recognition procedure, which is known to benefit memory performance; (c) more precise individual measurement; and (d) a design calibrated for state-trace analysis. Some of these issues were addressed by Prince and Heathcote (2009) and one-dimensional results were still obtained. The key
difference with the present results, and those of Prince and Heathcote (2010), appears to be avoiding floor effects, as in these latter studies evidence of non-monotonicity in state-trace plots was most evident for more accurate responding. Even with reduced floor effects, however, it does appear that it is hard to get a strong DFIE interaction in any of the recognition memory paradigm variants we studied except the Matched-Test paradigm.

Nevertheless, in the current experiments we have clearly shown that multi-dimensional processing of unfamiliar faces can occur in recognition memory regardless of whether orientation is manipulated during only encoding, only retrieval or both the encoding and retrieval phases. Our overall results are, therefore, consistent with the widely held view that inversion disrupts early perceptual stages of processing. However, as our paradigm does involve a strong memory component, we cannot (based on these results alone) assert that the DFIE is purely perceptually driven. Indeed, our finding of a numerically larger DFIE interaction estimate in the Upright-Study than Upright-Test condition suggests a mnemonic component. Future research could investigate if our multi-dimensional state-trace results are replicable in paradigms that minimise the involvement of retrieval processes. For example, the current design could be modified to reduce the delay between a study and test item by presenting the test pair immediately after the target image (e.g., Freire et al., 2000). Alternatively, the memory component could be removed completely, as in Freire et al., using a perceptual discrimination task where simultaneously presented images differ in their features or configuration.

Much research highlights strong differences between familiar and unfamiliar faces, with some going as far as to suggest that unfamiliar faces are unable to access the face-specific configural dimension that is available to familiar faces (e.g., Burton et al.,
1999; Hancock et al., 2000; Megreya & Burton, 2006; Young et al., 1985). Our consistent multi-dimensional results for unfamiliar faces do not support this proposal, and rather are suggestive that familiarity might cause a quantitative rather than qualitative difference. Our results also reject the idea that a single presentation at study is enough to make an unfamiliar face familiar. Under this view, it would be reasonable to expect multi-dimensional evidence for only the Study Upright and Matched Test conditions. However, our multi-dimensional Test Upright results cannot be explained by this proposal, as in this condition orientation was manipulated at study when our face stimuli were entirely unfamiliar.

Although our results indicated that faces as a stimulus class can support configural encoding, they do not speak to another issue related to familiarity: whether configural encoding of faces is an inherited genetic ability (Phelps & Roberts, 1994), or a manifestation of general-purpose perceptual category learning (Schyns & Rodet, 1997) due to the extensive experience with faces that is accrued from an early age (McKone & Robbins, 2007). Consistent with the latter view, Gauthier and Tarr (1997) found that after extensive practice identifying particular greebles – a set of artificial stimuli that like faces are homogenous and differentiated by the relations amongst a small set of features – participants displayed the behavioural hallmarks of configural processing. Greeble inversion effects similar to those found with faces have also been found in fMRI (e.g., Gauthier, Tarr, Anderson, Skularski & Gore, 1999) and ERP (e.g., Rossion, Gauthier, Goffaux, Tarr & Crommelinck, 2002) data recorded from greeble experts (see also Gauthier et al., 2003, and Boggan, Bartlett & Krawczyk, 2011, for related findings in car and chess experts respectively).

State-trace analysis could be used to address this controversial issue by examining whether dissociations found with objects of expertise, such as differential
inversion effects, are indicative of qualitative (multi-dimensional) encoding, or consistent with the special status of faces, due to quantitative differences in encoding on a single dimension. Absent a state-trace analysis it remains possible that the dissociations associated with highly practiced non-face stimuli that have been reported are not indicative of multi-dimensional encoding but rather are due to confounds such as range effects.

In closing, we wish to emphasise that our results in no way detract from Loftus et al.’s (2004) overarching conclusions about the uncertainties associated with evidence based on dissociations. We strongly caution against taking the implication that the same reassuring consistency between dissociation results would necessarily emerge if a rigorous state-trace methodology were to be applied to other paradigms, whether they use behavioural or neuroimaging (e.g., Freeman, Dennis & Dunn, 2010) dependent variables. We believe the example provided here of how to apply state-trace analysis using the methods developed by Prince, Brown and Heathcote (2012), and other recent examples of applications of state-trace methodology to face processing (Busey, Tunnicliff, Loftus & Loftus, 2000; Reinitz, Séguin, Peria & Loftus, 2012; Singer & Sheinberg, 2006), have general relevance for many fields of psychological investigation beyond the study of face processing (see Loftus, 2002 for further discussion). Indeed, the ability of state-trace analysis to combine evidence from disparate behavioural measures, such as confidence and accuracy (Busey et al.), and even behavioural and neuroimaging measures (e.g., response time and ERP amplitudes, see Provost, Johnson, Karayandis, Brown & Heathcote, in press), makes it seem an ideal method to enable rigorous inference based on converging evidence from a wider variety of measurement approaches.
Bibliography


Loftus, G.R. (1996). Psychology will be a much better science when we change the way we analyse data. *Current Directions in Psychological Science, 5*, 161-171. doi: 10.1111/1467-8721.ep11512376


